Anushka Rawat, 2026, 14:4
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Architecting Automated Resume Screening Systems

Anushka Rawat[1], Dr.Ajay Rana[2], Deepshikha Bhargava[3], Garima Panwar[4]
[1,2,3,4]Department of Computer Science and Engineering, Amity University, Greater Noida

Abstract- In the project, the aim is to create an automated system to help with resume screening. Typically, resumes do not come in a standard format. They vary in structure, writing style, and presentation. Therefore, processing resumes is far from easy compared to other texts. Unlike data in a neat and organized structure, as in academic data sets, processing a resume is a challenge. In academic data sets, the data is clean and in a neat organization. In order to manage these problems, certain textual processing techniques were applied, and the steps included normalization, tokenization, and feature selection. The system was tested numerous times in order to improve its ability to execute perfectly even with imperfect data. Over time, the scope extended beyond how to improve the precision of the program to include the influence of data quality in the real world. Aside from that, there are also ethical considerations involved in the making of the system, such as the implication that biases in the data might be inadvertently reflected in automated systems, where measures are being taken to make a more transparent and interpretable system, keeping recruiters involved by introducing flexible decision thresholds to keep human judgment in charge. Overall, however, it is a system meant to assist, not replace, the recruiter. In this manner, it can help speed up the hiring process and make it more efficient by reducing human error and quickly identifying potential candidates.

Keywords: Automated resume screening; Natural language processing (NLP); Text preprocessing; Recruitment technology; Algorithmic bias; Human-in-the-loop; Hiring efficiency.

I. INTRODUCTION

Resume screening is one of the most tiring and time-consuming aspects of the selection process. This is especially true for technical positions, as the number of resumes submitted in a matter of a few days may run into hundreds or even thousands. Careful scrutiny of every resume takes considerable time, concentration, and patience, which is taxing, as it is a challenging task even for recruiters with heavy workloads and deadlines to meet. In real-time situations, the recruiters might have to spend a short while on every resume. In fact, talented individuals with unusual skills and knowledge might be passed over for the position, not due to a lack of capability but due to the screening process.

However, manual processing also has a major drawback in the context of manual screening. This issue of inconsistency arises due to the diverse perspectives and approaches of various recruiters, which are largely based upon individual experiences and levels of busyness. In order to circumvent these

problems and address the increased number of applications, organizations are starting to look towards various automated tools that can aid recruiters in the processing of the first level of screening.

Machine learning provides a practical means of processing a large volume of resume data efficiently. The systems can analyze text material much faster than human beings and enable emphasizing potential candidates who may fit the needs. However, building such a system is not as straightforward as it might seem. Resumes seldom arrive in a standard format. The layout, style of writing, and structure vary greatly. Many contain tables, columns, graphics, or creative designs that make extraction difficult. Others use slang or domain-specific terminology that may not be picked up by the standard models. Prior to the implementation of any learning algorithm, the unstructured information must be cleaned and converted to an applicable form, which includes several sub-processes in the preprocessing of unstructured text data, such as tokenization, normalization, lemmatization, and feature representation. In practice, these steps often involve heuristics and trial-and-error methods. It is not always the case that techniques that are effective in theory are equally good in practice on actual texts. In the course of preparing this project, some

preprocessing techniques have been improved or replaced because of their inconsistent nature.

This article aims to illustrate the practical experience in developing an intelligent resume screening system in the context of real-world constraints. The work started from the simplest stage of using keywords in filtering resumes and continued toward using advanced techniques in feature extraction-based supervised learning methods. There were also experiences in dealing with the problems that occurred during the process, leading toward the final system, with the realization of the disparity between theory and reality in the data used for recruitment processes.

However, the main focus of this piece of research is, in an honest and practical manner, to report on building a resume screening system from the point of view of a student developer. Thus, it contrasts with the focus of many works in the literature that emphasize the study and discussion of the accuracy and performance of resume screening systems.

A. Keyword-Based and Rule-Based Systems

Through the early days of automated resume screening, most solutions relied heavily on simple keyword searches and basic rule-based filters. These scanning tools check out resumes for specific words related to the requirements for the job in skills, certifications, or educational qualifications, after which it shortlists candidates according to matches found. From an implementation point of view, this approach was easy, quick, and required very little computational effort. [11], [12] It was also the reason why this solution was so favored for organizations when big sets of job applications were being handled.

But this approach clearly had its drawbacks. It centered around exact word matches and often failed to look at the bigger picture of a candidate's experience. If an applicant described their skills with different terms or phrasing, then that system would likely fail to recognize them. Thus, qualified candidates might get rejected because of a lack of keywords in their resumes. Another weakness was that such systems were highly susceptible to gaming: as applicants gradually learned to include popular keywords, or even embedded hidden text within their resumes, some applicants managed to 'trick' the filter into passing them. [12]A result was that sometimes resumes were ranked higher by virtue of optimization tricks rather than actual capability. Due to such shortcomings, today, keyword-based tools are little more than basic screening tools, unsuited to handle the level of complexity and nuance entailed in hiring.

B. Keyword-Based and Rule-Based Systems

[11], [12]

Due to the popularity of supervised learning, statistical machine learning started to make a significant contribution to resume ranking and screening operations. Techniques such as Naive Bayes, Support Vector Machines, and logistic regression started becoming popular due to their fair performance, interpretability, and computational efficiency.

[12] All these factors were highly suitable for academic environments where powerful computational tools are limited. Most commonly, the resumes are converted into numerical vectors using vector space models such as vector space representations and TF-IDF representations. Using these labeled datasets, these classifiers learn how to classify the resumes based on distinguishing features to ascertain the relevance of the candidate towards a specific role in the job. Research in this area, carried out by Ahmed and Baig [11], has shown that these models can produce satisfactory results using well-structured datasets.

C. Statistical and Traditional Machine Learning Approaches

Given the progress made in the application of machine learning, there is now the adoption of the new process of deep learning rather than the traditional methods aimed at understanding the resume process. Various researchers have worked on using deep learning techniques, such as recurrent neural networks, convolutional neural networks, and, most recently, transformer models, to enhance the analysis of the text used in resumes. Traditionally, techniques focused on comparing keywords.

However, deep learning techniques work by trying to grasp the association between words in different professional scenarios. [13], [20] Research has proven that the utilization of word embedding and attention has the capability to improve the overall classification output.

Transformer-based architectures are particularly effective, with the ability to look at relationships extending beyond a document. These relationships can then be used to determine overall patterns like career development, relating skills, and even backgrounds. As experiments show, deep learning techniques normally have higher accuracy compared to machine learning models. Yet, these advantages notwithstanding, the implementation of deep learning models in academic environments, or in small projects in general, is hard to accomplish. This is due to the fact that the creation of deep learning models requires large datasets, in addition to large computing power, for the purpose of tuning the parameters involved in the models. As such, it may be difficult to replicate the results obtained in the studies.

D. Commercial Applicant Tracking Systems

Commercial (ATS) unifies a host of recruitment functions into a single platform, including resume parsing, applicant evaluation, and hiring workflow management. Most of the modern solutions are offering semantic search, skill identification automatically, and simple predictive analytics. These help organizations manage such a volume of applications efficiently, which is essential in companies that conduct frequent or bulk recruitments. [7] In spite of convenience, most ATS platforms use proprietary technologies that are not publicly explained. Due to the fact that internal logics behind ranking or filtering candidates rarely make much sense for users, they can be suspicious about how the generated decisions have been made.

Furthermore, because of these limitations, independent assessment of these systems makes it hard to objectively measure their reliability or fairness. Many organizations must therefore depend on the claims of vendors regarding system effectiveness. [9]

E. Educational and Experiential Perspectives

From a learning perspective, hands-on experiences are also crucial for understanding how machine learning systems are applied. Learning theories, such as Kolb's Experiential Learning Theory, highlight that people learn better when they experience it firsthand. Instead of memorizing facts, people learn concepts better when they are using the knowledge to observe and understand what happens and reflecting on that experience. [4] Scholars like Garousi and Herstad also highlight that a disconnect exists between academic prototypes and real-world software systems, citing many factors students might encounter while using software, such as inconsistency in data, deployment, and maintenance, among others. Our experience is consistent with this, as several hurdles were met during implementation, such as handling messy data as well as software dependencies. [1]

F. Limitations in Existing Literature

Despite the broad research on resume screening and the automation of the hiring process, there are some gaps in the existing studies. Most of the research has focused on the effectiveness of the proposed models, paying little attention to the relevant factors that are significant in practice. For instance, the handling of different document layouts is given little consideration, even though the layout may affect the reliability of the resume parsing process. In addition, the ethical implications of AR-systems have not been taken into consideration more frequently than their effect on the success of the hiring process. Additionally, little attention is given to failed attempts or unsuccessful projects, even though they could have provided valuable information for future studies. [4], [15]

To overcome these drawbacks, in the present study, a developmental approach is undertaken that simulates realistic working conditions. Rather than conflating results based on accuracy maximization alone, consideration is given to the effective performance in situations involving incomplete information, [4], [15] unstructured formatting, as well as computationally intensive tasks. Moreover, there is an emphasis on moral responsibility and reliability rather than accuracy maximization alone.

II. DATASET DESCRIPTION AND PREPARATION

A reliable dataset is the most basic requirement for creating a successful machine learning system. To develop a machine learning-based resume screening system, the process of creating the required dataset is very challenging because of privacy concerns, the disorganized nature of the resumes, and the clear and incomplete nature of the information. [8] Different resumes tend to differ greatly from each other, creating a challenging environment for the screening process. It is the objective of this section to highlight the process undertaken to prepare the data before the testing phase.

A. Data Collection

The set of data used in this research comprises approximately 2,000 resumes collected from different sources, including publicly accessible databases, academic research conducted previously, and final-year students who voluntarily participated by contributing their resumes. They were made aware that the resumes would be used for research purposes and gave their consent. [8]Information such as names, phone numbers, email accounts, and home locations, which are of a private nature, was removed from the resumes during the preprocessing stage.

B. Data Distribution

Our implementation experience only reinforces the fact that nothing is easy when working with practical data. Several implementation issues arose after the project began. These issues include inconsistencies in the resumes and software dependency problems. These were not obvious to us when we planned our project. [11] It is only after we began the project that we realized that more than knowledge is needed to take care of the issues. We actually have to troubleshoot.

C. Data Cleaning and Privacy Protection

A substantial portion of the collected resumes contained information that did not make meaningful contributions to the analysis. These contained decorations like images, tables, headers, footers, and icons. The process began with filtering out this information [15] types to ensure that relevant data was uniformly selected. Therefore, it improved consistency in data quality.

Another important issue during every preprocessing task was protecting personal information. Information such as names, contact numbers, email addresses, and residential details was identified using specific patterns and masked accordingly. Although this is a time-consuming process, it has to be carried out to maintain ethics. [9] There were also some technical difficulties encountered, such as pertaining to the text encoding errors, where the text was not easily recognizable. These were handled by normalizing the encoding of the texts, as well as by converting the texts once more from the original files.

D. Text Normalization and Annotation

Upon completion of the primary cleaning process, further standardization of the resume texts was carried out to ensure consistency. During the process, all texts were converted to one standard case, unnecessary punctuation was removed, and lemmatization and stop words were eliminated. Additionally, some phrases specific to resumes and consistently appearing in them with minimal relevance to personality, etc. [12], were excluded in order to enhance the quality of features.

To obtain reliable training data, a portion of the resumes was examined. Each resume was examined for defined categories of jobs to obtain scores for their relevance with regard to the target job. Assessing the suitability of the candidate involves room for subjectivity. As such, regular coordination of the team was necessary. Assessed differences were regularly compared. [17]

E. Dataset Limitations

Although the data was designed and pre-processed with care, it was recognized to have some limitations. Compared with the copious volumes of data that recruitment websites routinely work with, the number of resumes that were available for this study was considerable. [20]Data access limitations also caused difficulties with collecting data from some geographical locations and organizations.

Additionally, the data representation of technical fields was uneven but mainly depended on the data availability.

These limitations have an impact on the effectiveness of generalization that was achieved by the trained models. It should be noted that, as an alternative to ignoring them, these constraints are acknowledged with the aim of providing transparency with regard to an appropriate understanding of the system.

III. THE NEED FOR AUTOMATION IN RESUME SCREENING

[20] In recent years, with the rapid growth of the technology industry, there has been a sharp increase in applications that are received for each position that becomes available. For an organization, it is no longer a matter of perusing through the various resumes as they become overwhelming when there are hundreds, indeed thousands, of applications being received for a particular position. There is a great strain that can be felt when application processing becomes tedious, [20] where otherwise qualified candidates may be unintentionally passed over due to insufficient processing time.

A. Overcoming the Screening Bottleneck

A common problem that comes with these applications involves careful scanning of resumes due to large volumes of work. However, this means that detailed but critical information may not be given proper consideration. [9] Application automation provides powerful solutions that assist in eliminating such scenarios. This is because such systems permit detailed analyses of all resumes by using the same application procedure. This minimizes the chances of such information being missed.

B. Consistency in Candidate Evaluation

The other advantage that comes with automated screening is that it brings uniformity to the assessment process itself. It is known that when a screening process is conducted by a human, it is difficult to have assessments that are comparable from one individual to another. [13] There are factors like workload and even fatigue that may affect an individual while performing this task. Automation technologies, on the other hand, use uniform criteria for assessing all the individuals, thereby creating a certain uniformity to the screening process itself when handling many individuals.

IV. SYSTEM ARCHITECTURE

The automated resume evaluation tool is designed in a modular way to allow flexibility, testing, and improvement. Emphasis was also placed on keeping the automated resume screening system separate from the beginning. In simple terms, the developers focused on implementing the tool so that its different parts operate independently. [13], [14] As a result, any change could be made easily without impacting the entire process.

A. Overall System Overview

The overall architecture provides insight into the manner in which the major modules of the program cooperate during execution. The workflow of the program starts with the receipt of the resume on the platform.[14] The resume then goes

Fig. 1. High-Level System Architecture

through several stages of text preprocessing, generation of features, model-based evaluation, and presentation of results. Each module processes a single function while communi- cating with other components or parts of the system only when absolutely necessary. This helps to keep all components of the system independent of each other. In the development phase or during testing, these modules could also be modified or even

changed according to necessity without influencing the entire system.

B. Input and Parsing Module

The component designed specifically for input cases with resumes comes in handy in cases where resumes are sent through common formats like PDF or DOCX. However, as soon as the document is selected, it goes through a validation step that ensures the document is in an appropriate, readable state before the parsing step. It uses document processing libraries that work with both structured and semi-structured formats.

A couple of challenges were encountered with resumes having multi-column designs, tables, and other complexities. Under such circumstances, the output texts were not in the correct order. This created further complications in the output analysis. So, layout-aware heuristics were incorporated in the system, as they would help in approximating the actual reading order of the documents. Under circumstances where the output of the system could not provide the expected results, further steps were taken to clean up the output.

C. Preprocessing and Feature Extraction Layer

The preprocessing layer is responsible for transforming raw text data into a form that is understandable by machine learning. The process of preprocessing includes tokenizing the text, using standardized word forms, and applying lemmatization, which minimizes different forms of a word being used for a particular concept. [12] The curated list of vocabulary was also included to enhance the detection of technical text.

Following the preprocessing step, the resumes were converted to numerical representations via the TF-IDF method of weighting. Such an approach ensures that emphasis is placed upon words that carry critical information, thus diminishing the usefulness of frequently occurring yet less informative words. To ensure data efficiency as well as avoiding increased model complexities, the removal of less contributing features occurred via the filtering method.

D. Model Management Component

The model management layer was intended to support the entire life cycle of the classifiers, including training, validation, and deployment. Each of the different learning algorithms was processed through its own separate pipeline, which made it possible to do a comparison in terms of the experiment.

To further enable systematic experimentation, access to trained models along with their associated metadata, like hyperparameters, datasets, and performance metrics, is made available. The ability to keep track of these audit trails has also enabled researchers to reproduce previously conducted experiments, observe performance trends, and diagnose any unexpected model behavior. This well-structured approach came in handy while refining a model.

E. Decision and Ranking Engine

Once this classification task was completed, the outcomes were passed to a ranking module dedicated to calculating a score related to the general suitability of each applicant. This was based on different criteria, such as confidence levels and sets of skills/relevant experience.

There may be some setting of thresholds that allows the degree of constraint to be varied according to specific needs. To improve interpretability, additional explanations were also provided by the ranking engine for each result. These explanations briefly identified the skills, qualifications, or experience factors that were most significant in the final score calculation. [2] By providing such background information along with the rankings, users benefited more from the system outputs with increased trust in the system results.

F. User Interface Layer

A light interface for the system was implemented using the Flask framework to allow for easy interaction with the system. This allowed users to upload resume documents, pick the most appropriate job categories, and even view a list of candidates. Basic GUI elements were also added to

allow the user to get a sense of how the candidates were distributed.

To elicit user feedback, testing sessions were conducted with a user base of peers who used the system with real- world use cases. These testing sessions revealed several user issues pertaining to the interpretation of error messages and upload time for users. [9] Further, based on the user feedback received during these testing sessions, several enhancements were incorporated.

G. Security and Reliability Considerations

Considering the fact that the recruitment systems deal with confidential information regarding applicants, the aspect of

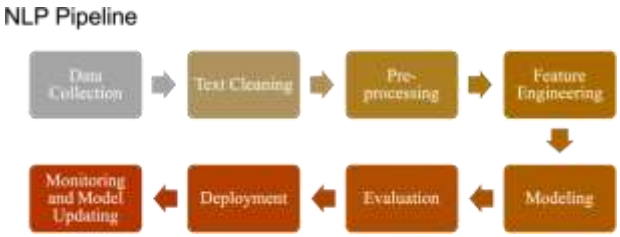

Fig. 2. Natural Language Processing Pipeline [12], [17]

data protection was one of the main concerns during the development of a recruitment system. Various measures, such as the use of safe file management procedures and encrypted storage of required information, were implemented in the creation of a recruitment system. [15] The reliability of the system was ensured through the implementation of various backup strategies, as well as various logging mechanisms that ensured all operational issues, including any system errors that may have been encountered during preprocessing and execution of the model, were well taken care of. Logs helped in efficiently diagnosing problems with the system, thus facilitating system performance while allowing issues to be addressed accordingly.

H. Advancements in Applicant Tracking Systems

Early systems for applicant tracking typically relied on simple keyword matching. However, this led to a problem whereby applicants could easily manipulate the system to their own advantage. They could easily do this by merely adding popular keywords to their resume. In this regard, the systems could easily be compromised by favoring keyword frequency over actual skill and experience evaluation. [7], [9]

It is also important to note that the success or failure of a resume filtering system is, in one way, associated with the possible ability to handle unstructured data. In other words, the information contained within the resume is not formatted homogeneously; there is a possibility that a resume may consist of both important pieces of information regarding the potential candidate and irrelevant data, [11] Technological Paradigms in Document Analysis

It is also important to note that the success or failure of a resume filtering system is, in one way, associated with the possible ability to handle unstructured data. In other words, the information contained within the resume is not formatted homogeneously; there is a possibility that a resume may consist of both important pieces of information regarding the potential candidate and irrelevant data, [11] like the decorative components included to beautify the resume.

I. Natural Language Processing (NLP) Integration

The proposed framework is based essentially on natural language processing techniques; such techniques are crucial to turn raw text from resumes into a structured, usable format. Basic preprocessing steps include tokenization, normalization, word removal based on frequency, but without information content, and reduction of terms to the base. [12] This is actually a standardisation step such that different word forms represent the same word.
Besides these primary steps, the system is also able to recognize domain-related words and phrases that accurately represent the skills and experience of candidates. The extraction of this data enables the model to concentrate on accurate indicators of expertise and represent candidate profiles in a better way.

J. Machine Learning Classifiers

Several supervised learning algorithms were being evaluated during the development process to identify an appropriate technique that provided an

effective trade-off between predictive accuracy and interpretability. Deep learning architectures are commonly considered effective but also less interpretable. However, the application of simpler techniques like the Random Forest and Support Vector Machines proved useful.

Interpretability was also considered to be particularly relevant in the context of recruitment, as it also needed to be interpretable and justifiable. To further aid this, feature importance analysis was carried out to identify the skills/experience attributes that had the greatest impact in the decision-making process. This also helped ensure that ranking was based on interpretable professional attributes rather than arbitrary correlations

V. LITERATURE REVIEW: BRIDGING ACADEMIC THEORY AND INDUSTRY NEEDS

While past research attempts to deal with resume screening by an automated system have a strong theoretical basis, many of the research works have exhibited a lack of practicality and understanding of student developers. This is because the solutions to resume screening by an automated system have generally been built without the financial and technological conveniences that corporate developers have access to. How- ever, it is important to consider such practical aspects when analyzing and critiquing the solution to the resume screening.

A. The Shift Toward Experiential Learning

There are various theories of education that emphasize the importance of learning through practical experience rather than solely relying on theoretical methods. For instance, Kolb's model of experiential learning highlights that all learning takes place through a cyclical process of experimentation, reflection, and adaptation. The design of the resume screening system essentially followed this approach because when unrealistic outcomes were produced using this model on various unpredictable resumes, [1] It was deemed important to reconsider previous notions and adapt the entire process.

B. Critique of Current Algorithmic Approaches

While in most studies the proposed approach is reported to have a high level of accuracy using carefully prepared and structured datasets, the question remains for its usage in real- life conditions. Garousi had also pointed out that approaches that have shown the highest accuracy in a controlled environment are not sure to perform the same when applied to messy and inconsistent data. In order to get a better idea of the limitations that the proposed approach is likely to face, a test was carried out using resumes that have a non-standard layout. Such non-standard [4] resumes are more likely to reflect real-life documents.

C. The Ethics of Algorithmic Bias

The question of fairness has emerged as a key concern in AI-based recruitment tools. It has also become important in AI hiring technologies, which may, in their training data, contain a substantial amount of unintentional bias, even in cases where no intentional discrimination has taken place.

In order to address this problem, the issue of fairness has been taken into account. This has helped in understanding the impact of different features on the ranking and the possibilities of imbalances in the results. This has helped in ensuring that the results of the process are based on the skills and professional qualifications possessed by the person and not on other factors.

VI. METHODOLOGY: A PRACTICAL DEVELOPMENT FRAMEWORK

The system is developed using a structured approach divided into individual phases that are consistent with a particular phase model, ensuring that development remains easy to track at every point without compromising clarity. This will not only serve as a check on development but also allow further extension with minimal redesign by others at a later time.

Phase 1: Data Acquisition and Cleaning

Initially, there was a dataset of over 2,000 resumes obtained from different technical domains. The preparation of the initial dataset was a time-

consuming process, which proved to be the most challenging task of the project. Parsing the text from some PDF files was an inconsistent operation, while some files needed to be double-checked to ensure that the text was being appropriately parsed. [13] Similarly, the protection of the information remained of equal importance at this phase. Using the pattern-identification method, the information that identified the individual, such as their names, contact numbers, and email addresses, was eliminated. Some documents also contained character encoding errors, which resulted in garbled and undecipherable text, sometimes hampering the parsing processes.

Phase 2: Feature Engineering

After data cleaning, the text data needed to be transformed into an appropriate machine learning representation. For this purpose, the decision was made to use the TF-IDF method of vectorization. The most important words have more weight with this method, but less common and very common words are less important with this approach.

By focusing on technical skills and domain-specific words, the strategy helped the system distinguish the profiles better. Resumes with expert knowledge have thus been better represented in the training stage.

Phase 3: Model Training and Selection

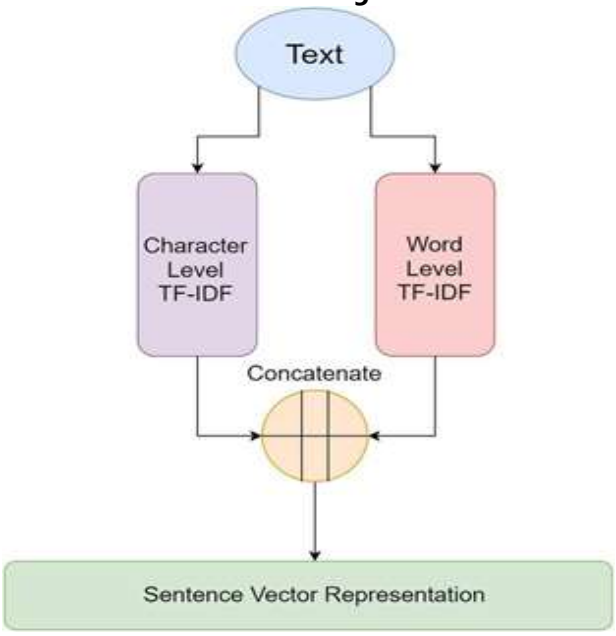

Fig. 3. Life Span [1]

Three main classifiers were used:

a) Naive Bayes, which may be employed as a baseline classifier for text classification tasks, is capable of handling more complex data interactions. Therefore, I chose the "Random Forest" model. The purpose of using "Random Forest" was to capture more complicated dependencies and is generally able to achieve higher accuracy and robustness.
b) K-Nearest Neighbors (KNN) algorithm for measuring similarities between resumes and between candidate profiles.

Phase 4: Evaluation and Refinement

For calculating the components of the effectiveness, usual parameters like Precision, Recall, and F1 Score were employed. F1 Score was given special importance since it is required to maintain a perfect balance so that the potential candidates can be correctly identified without the results being maximized in the wrong manner. From the results of the evaluation process, improvements to the code were made to perform better by gradually refining the pre-processing, features, and parameters to achieve consistent results.

VII. SYSTEM LIMITATIONS [20]

Although the expected system has demonstrated gratifying performance, some limitations should be noted. Most of these constraints stem from the scarcity of available data and the computing power required to develop the system. Compared with large-scale commercial recruitment platforms that handle vast volumes of data, the dataset used in this research is considered small. Though attempts were made to fetch resumes with varied technical backgrounds, their variation is still limited compared with what is actually out there in a real-world recruitment context.

The framework also relies mostly on text data sourced from resumes. Documents with heavy reliance on visual layout, like those with images, charts, and complex layouts, may not be successfully processed. Moreover, significant information may be lost during parsing, which will impact the classification. In the current scenario, the system is

designed to assess candidates for one role at a time. Cases like recruitment that involve multiple roles or multiple functions are beyond the modeling strategies that were examined for this research.

VIII. THREATS TO VALIDITY

There are a number of variables that could potentially affect the reliability and generalizability of the results. These variables include potential problems associated with data preparation, experimental design, and representativeness of evaluation data. Being aware of these concerns helps to look at the results realistically. [13]

A. Internal Validity

Internal validity could also be affected by imperfections in the preprocessing and labelling steps. The models could be adversely affected by any possible errors in text extraction, which might have changed their patterns. Though both automated and manual methods were used, perfecting all minor errors was impossible.

Model configuration choices also led to the presence of possible bias. This is driven by the fact that the computational capability was limited, and thus the hyperparameters were also limited. This means that it is possible that a better model was not included in the optimization process.

B. External Validity

The evaluation criteria were mainly resumes from students and individuals who were in their early professional careers. Hence, its efficacy in senior professionals, technical experts, and managerial personnel is not clear and may vary in their cases. [4] Moreover, the dataset featured a limited scope of geographic and cultural diversity, given that the majority of the resumes referenced a similar academic and professional environment. Such a factor may harbor the scope to influence the effectiveness of the proposed approach to a limited extent, considering a more accommodative scenario.

C. Construct Validity

It is also important to note that while it is good to have evaluated metrics like F1 SCORE for evaluating classification accuracy, they do not tell the full story of the effectiveness of a particular system for real-world recruiting scenarios. The quantitative measure alone is not enough to signify recruiter satisfaction levels or even the quality of hiring decisions made thereby. [9] Therefore, it can be concluded that depending only on statistical performance measures may not give a complete picture of the worth of this system, and it would be useful for further studies to obtain feedback from recruiters and longer- term studies to measure the effect of this system on hiring.

IX. DEPLOYMENT CONSIDERATIONS

Implementing an automated resume screening system in an organization involves challenges that go beyond simply testing it experimentally. It is important to recognize that the system may perform well during the experimental phase but fail to operate as expected in the real environment due to other factors. Planning is therefore crucial to ensure that the system is effective, suitable for the organization, and compliant with data protection regulations. These factors are vital for increasing the chances of successful deployment and usage.

X. DISCUSSION: REFLECTIONS ON TECHNICAL MATURITY

Within this development setting, practical solutions quickly proved to be more effective than unnecessarily complex ones. Rather, simple and easily understood methods tended to fare better than those that were richly intricate. Although such sophisticated methods may sound good in theory, [6] this does not necessarily mean that they will work well under practical constraints.

A. The Value of Simplicity

In spite of more sophisticated models being explored early on, in the end, simpler models were more secure. In this regard, while outcomes were comparable for more complex models with regard to the Random Forest model, this one was far more manageable and likely to be understood, fixed, and controlled. In this way, this experience also served to remind individuals of the notion that simple models

can also prove to be superior, depending on what is required.

B. Societal Impact

The construction of the automated screening tool also demonstrated the importance of the responsibility that comes with a decision-support technology. These types of systems have a significant impact on real individuals, so the shape of their professional paths is also a factor to be considered. The project also demonstrated the importance of more than mere technological performance. [15] The need for explanations, transparency, and accountability was also observed to be significant.

XI. CONCLUSION

Creating an automated resume screening system requires far more than writing code and/or choosing machine learning algorithms. For a computer science student, such a problem is an exercise that bridges academia and reality by providing real- world expertise in dealing with untidy data, design tradeoffs,

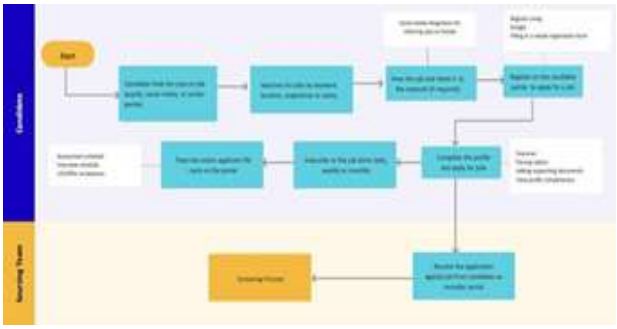

Fig. 4. Workflows for Conceptual Fairness [15], [18]

and problem-solving in an environment where questions never seem to have simple answers. During the entirety of the problem-solving process, it becomes immediately apparent that the mere technical feasibility of an answer is far from sufficient. Some of the most interesting learnings I came away with were focused on ethical dilemmas. This work is not just about effectiveness, but also about the role these tools will play in hiring processes. How these tools are designed to be used will eventually inform how fair these tools are.

A level of fairness in how candidates are treated by these technologies is critical so that candidates are not unfairly impacted. The reflection also brought to attention the need to incorporate improvements through iteration. There were certain design decisions that had to be changed during the process of testing, which revealed difficulties that were unforeseen. These improvements actually helped to make the model more stable. Therefore, through this process, the lesson that can be learned is that even technical expertise is not as significant as effective evaluation in machine learning.

REFERENCES

1. D. Kolb, Experiential Learning: Experience as the Source of Learning and Development, FT Press, 2015.
2. A. Ahmed and S. Baig, "Evaluating the Efficacy of ML in Recruitment Pipelines," IEEE Transactions on Education, vol. 62, 2019.
3. ACM/IEEE-CS, Computer Science Curricula: Guidelines for Undergraduate Degree Programs, 2013.
4. V. Garousi and T. Herstad, "Bridging the Gap Between Industry and Academia in Software Engineering," IEEE Software, 2019.
5. J. Zhao et al., "Gender Bias in Coreference Resolution," EMNLP, 2018.
6. S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, Pearson, 2020.
7. P. Fader, "The Future of Recruitment: Data Over Instinct," Harvard Business Review, 2021.
8. R. Sharma, "NLP Techniques for Professional Document Parsing," International Journal of Computer Applications, 2022.
9. L. Chen, "The Impact of Automation on HR Decision Making," Journal of Business Ethics, 2020.
10. K. Miller, "Algorithmic Transparency in Hiring," IEEE Technology and Society Magazine, 2022.
11. C. Manning, P. Raghavan, and H. Sch¨utze, Introduction to Information Retrieval, Cambridge University Press, 2008.
12. D. Jurafsky and J. Martin, Speech and Language Processing, Pearson, 2021.
13. C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

14. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
15. S. Barocas, M. Hardt, and A. Narayanan, Fairness and Machine Learning, MIT Press, 2019.
16. A. Ng, "Machine Learning and AI via Brain Simulations," Communications of the ACM, vol. 63, 2020.
17. J. Hirschberg and C. Manning, "Advances in Natural Language Processing," Science, vol. 349, 2015.
18. A. Caliskan, J. Bryson, and A. Narayanan, "Semantics Derived Automatically from Language Corpora Contain Human Biases," Science, 2017.
19. S. Pan and Q. Yang, "A Survey on Transfer Learning," IEEE Transactions on Knowledge and Data Engineering, vol. 22, 2010.
20. Z. Zhou, Machine Learning, Springer, 2021.