© May 2025 | IJIRT | Volume 11 Issue 12 | ISSN: 2349-6002

Resume Screening System

Kankanala Sahithi[1], Pinjari Sanjana[2], Sreeja Sumkara[3], Shyamala Akhil[4], Avinash Amaragani[5]
Department of Computer Science Engineering (Artificial Intelligence and Machine Learning), Sreyas Institute of Engineering and Technology, Hyderabad, India[1234]

Abstract—**The Resume Screening System is an innovative, AI-powered tool that aims to speed up the hiring process. Current resume screening process is time taking and prone to human bias, particularly when there are large number of applicants. This study focuses on the issues of efficiently assessing, retrieving, and evaluating important information from candidate resumes. It compares the system will ensure that only the most relevant profiles according to the job requirements are shortlisted. Present screening systems will only match exact keywords, our system understands the meaning behind the words to match candidates based on the context. The goal of this system is to provide an accurate and time saving tool for the recruiters that will improve decision-making during the hiring process. The system can be integrated into existing HR platforms or it can be utilized independently by small and medium scale organizations. In the end, this system will not only reduce the manual workload but also improves the quality of hiring by ensuring that best-fit candidates are prioritized.**

Keywords— **Resume Screening, Semantic Matching, Candidate Shortlisting, Skill-Based Filtering, Job-Profile Matching.**

I. INTRODUCTION

In recent decades, the companies hiring process has changed significantly. In early 1990s, organizations started focusing on resume screening to automate the hiring process. Back then, hiring teams would receive huge number of job applications for single post. Manually going through each resume was low and exhausting process and it also led to inconsistent and biased results. One of the first answers to these problems was presented by a business named Resumix. This company developed a software that is capable to analyze resumes and extract information about skills, education and experience.

In earlier systems, simple keyword-matching algorithms were used. This has reduced the burden on the HR teams, but this technology lacked the ability to understand context or meaning. As a result, qualified candidates were often overlooked since their resumes did not contain the exact terms that are specified in the job description. Even though, these early systems played a crucial role in shaping the recruitment process, setting the foundation for what we now know as Applicant Tracking Systems (ATS).

By the early 2000s, ATS platforms like Taleo, iCIMS, and Bullhorn gained huge popularity among software companies. These technologies have provided recruitment functions like posting jobs, collecting resumes, filtering them, and communication with candidates. Their primary purpose was to simplify the hiring procedure and organize all the candidate data. While applicant tracking systems (ATS) helped in streamlining the hiring process, the actual method of screening resumes remained largely unchanged. They still depended on simple keyword occurrences and struggled in identifying similar terms of an applicant's resume.

1.1. Problem Statement

Companies frequently receive hundreds of applications for a single position, and it takes a lot of time and effort to manually go through each resume. This slowdowns the hiring process, manual screening may result in unconscious bias. As a result, even right applicants are overlooked sometimes.

Many companies have begun to use Applicant Tracking Systems (ATS)to address this. By eliminating applicants who don't seem to fit the job description, these tools help in managing the resume flow. However, the majority of ATS systems only use precise keyword matching. Even if a candidate is eligible for the job but their resume is not containing the exact words from the job description then algorithm ignores them.

These systems don't understand the meaning or context behind the words. There's a big difference between matching words and matching what a person is actually capable of. This project is designed to overcome this issue using AI and Natural Language

Processing (NLP), so resumes are evaluated more fairly, more accurately, and in a way that identifies the candidate's potential.

1.2. Objective

This project's main goal is to create a resume screening system that increases the accuracy, speed and intelligence of the hiring process. For a single job post, recruiters receive a large volume of applications. It takes a lot of time and effort to manually go through each resume, and there is always a chance that right candidates will be missed because of human bias or the use of alternative terminology.

This system is built in a way that it understands the content from the resume instead of just searching for similar keywords. Using artificial intelligence and natural language processing. It identifies important details like a candidate's education, experience and skills before evaluating how well they are suitable for the job. This makes the whole process smarter and more accurate.

The goal is to help recruiters make better decisions by reducing manual effort, minimizing errors, and ensuring that candidates are shortlisted based on relevance, not just word matching. The system will also rank the applicants based on how closely on how closely their profile matches the job requirements. This system can be used individually or it can be added to existing HR tools.

II. LITERATURE

2.1 Literature Survey

Over the years, resume screening has gained a lot of attention from researchers, especially as job applications moved online and companies started receiving huge volumes of resumes. Manually reviewing every resume soon became unrealistic, especially for large companies. This is what pushed researchers and developers to create models and techniques that could make the shortlisting process easier, faster, and more accurate.

The earliest research focused on matching keywords. One of the most well-known approaches was TF IDF, a method that calculates the importance of a word based on how often it appears in a document. This means the resumes that contain more job-relevant words were given higher scores. It worked well for quick filtering but it had one drawback that is didn't understand the meaning behind the words. For example, two candidates might say the same thing in different ways, but only one will get picked because their words were matching more closely.

As technology evolved, researchers experimented with machine learning techniques, like Naïve Bayes, SVM, and Decision Trees. These systems are trained and tested on the resumes that are reviewed by HR teams. This method was able to identify patterns instead of checking for keywords. However, it only Functioned effectively when the date was reliable. The algorithm will make mistakes when it encounters anything new if the training data does not cover a few job roles or industries.

Unsupervised learning methods, such as clustering approaches, were also used in several research. Even if different terminology is used, these methodologies will shortlist resumes based on common experiences or skill sets. This allowed recruiters to quickly review groups of relevant resumes. These models, on the other hand, could only sort based on broad similarities and were sometimes unable to select the best candidate from a group due to a lack of specific labels to learn from.

Simultaneously, researchers started to using Natural Language Processing (NLP) to extract important data from resumes. Named Entity Recognition (NER) is a widely used NLP technique that extracts important data. It identifies important data such as job titles, educational background, and work experience. This made it much easier to convert unstructured resume content into a more organized format that systems could actually work with. It as gave recruiters better control by letting them filter and group resumes based on specific criteria, making the entire screening process more clear and manageable.

In recent years, improved models have been developed to assist computers in understanding the meaning of resumes and job descriptions. Meaning of the words is analyzed using deep learning algorithms like Word2Vec and BERT. These models can identify if two different phrases are indicating the same thing. These new techniques are promising and have great accuracy. However, because to their complexity and the need for a significant amount of processing capacity, they are still not fully integrated into traditional recruiting systems.

Overall, there is an obvious change in research from simple keyword filters to more advanced, learning-based methods that understands and analyses the context first. Many of these ideas are still too advanced to be applied successfully in very industry, though. Individuals, their backgrounds, and their suitability for a certain role.

2.2 Literature Review

As more companies moved their hiring processes online, researchers started thinking about better ways to handle the growing number of resumes. One of the first problems they noticed was that older systems mostly just searched for keywords. Jain and Agarwal (2021) talked about this in their study. They indicated that tools such as TF-IDF may help filter resumes more quickly, but they didn't quite comprehend what the applicant was saying. So, even if someone has the necessary talents, they could ne overlooked just because their Phrasing did not quite fit the job description.

Later, Patil and Sharma (2019) took a different strategy. They trained a machine learning model using resumes already vetted by hiring teams. The model learned to identify which resumes were a good fit and which were not. It performed better than keyword searches, but it had several limits.

If the system saw a job role it hadn't been trained on, it didn't always know what to do. So, while it was helpful, it didn't work well for every type of job.

Other researchers looked into grouping similar resumes together using clustering methods. These systems didn't need labeled data; they just grouped resumes that appeared similar. This saved time, especially because there were so many resumes to sort through. However, they were unable to determine which person was most suited for a particular position. Recruiters still had to hand check everything.

In recent years, there has been a greater emphasis on using natural language processing (NLP) to extract relevant information from resumes. Named Entity Recognition (NER) is a mechanism for identifying key things such as abilities, qualifications, job titles, and work experience. This makes it easy to divide down a CV into meaningful sections, allowing the system to better evaluate it and recruiters to discover what they need more quickly.

Some recent studies have also used deep learning models such as Word2Vec and BERT. These models attempt to understand the context of what someone is saying, rather than just the words they use. For Example, even if someone writes "worked on mobile apps" rather than "developed Android applications," the algorithm may recognize the same concept. These techniques are quite effective, yet they are still primarily employed in research. They have yet to be widely implemented in real-world hiring tools due to their complexity.

From all this, it's clear that resume screening has improved a lot, but most systems still don't fully understand the candidate's message. Many are fast, but they miss the point. That's why our project focuses on understanding meaning, not just matching words. The goal is to create something that helps recruiters make better choices by really understanding what each resume is saying.

III. RELATED WORK

As, more people began looking for employment online, businesses began to receive an overwhelming volume of resumes. It became very difficult for recruiters to carefully review each one. To make the process easier, developers and researchers began creating technologies that may assist with resume screening. Many ideas were tested over time; some were simple, while others relied on more modern technology such as machine learning. They all have the same goal: to assist recruiters locate the right candidates faster while not missing out on prospective talent.

In the beginning, most resume screening tools just looked for specific keywords from the job description. A method called TF-IDF was often used, which checks how often a word appears and how important it is in that context. This method was fast and easy to use, but it didn't really understand what the person was saying. For example, two people might have the same skill but mentioned by Jain and Agarwal in a 2021 study, where they said that keyword systems are quick but not always fair or accurate.

Later on., researchers began using machine learning to make the process better. They trained models using resumes that were already reviewed by HR teams. One such model, developed by Patil and Sharama in 2019, could sort resumes into categories like "suitable" or "not suitable." It worked better than keyword filters but still had its limits. If the model saw a completely new type of job or role, it didn't always know what to do with it. It worked best when it was used on jobs it had already seen before.

Some teams also tried a different method called ontology-based filtering. This is where they map out how different job titles and skills are connected. For example, they would link roles like frontend developer and web engineer if they had similar tasks. While this method helped in some areas, it needed a lot of manual setup and constant updates, which made it hard to manage especially with how fast things change in the job market.

More recently, platforms like Zoho Recruit, Seek Out, and Hiretual have added AI features to their tools. These platforms are much better organized and quicker to use, which helps recruiters work more efficiently. But even with these upgrades, most of them still use the same old logic-searching for exact words or using fixed rules. This means a lot of resumes still need to be checked manually, and strong candidates can be missed if they don't use the "right" wording.

Our project tries to improve on this by focusing more on what the resume is actually saying, rather than just which words are used. We use Natural Language Processing (NLP) to look at the meaning behind the text. This helps us understand how well someone's experience and skills match a job even if they describe things in their own way.

One thing that makes our system different is that it doesn't just look for repeated words—it pays attention to the story a person is telling. A resume should show who the person is and what they've done, not just check off a list. Our system tries to read it the way a human would—looking at the full picture and not just matching terms.

Another important part of this system is fairness. Older tools usually work better for people who know how to write resumes in a certain format. But not everyone writes the same way, and that doesn't mean they aren't qualified. By focusing on the context of what they wrote, we try to give everyone an equal chance – especially those who might have been ignored by traditional systems.

Methodology

The main aim of this project is to help identify the right resumes from a large number of applications by matching them meaningfully with a given job description. We didn't want to rely only on keyword matching like most traditional systems. So we developed a solution that uses Natural Language Processing (NLP) to understand resumes and job descriptions. The goal is to read through the candidates' written responses, extract crucial details such as their abilities, experience and qualifications, and then compare that information to the job requirements. We divided this method into two basic stages:

(i)Preparation Stage, All resumes are cleaned up, and critical information is removed and sorted. This includes reading the resume, removing unnecessary words, and identifying sections like talents or past career roles.

(ii) In the Matching and Shortlisting Stages, the system compares each resume to the job description using similarity-based approaches. It aims to determine how closely a candidate's qualifications match those required for the position. After the comparison, each resume is given a score based on its relevance. Finally, the candidates with the highest scores are selected. This entire procedure reduces manual work and improves shortlisting accuracy and fairness, especially when there are too many resumes to go through one by one.

3.1 Preparation Stage

The preparation stage is the most crucial part in the resume screening procedure. The algorithm first understands the contents of each resume before it matches resumes to job descriptions or make decisions. Since, resumes can be produced in number of formats, styles, and tones, this process will focus on closely examining resumes, analyzing the language, and extracting only the most crucial information. The process begins with resume gathering. Most resumes are submitted in formats such as PDF or DOCX. Machines cannot read these files directly, so we utilize text extraction tools like Apache Tika or Python libraries to convert them to plain text. This stage allows the system to properly handle the resume content, regardless of how the original file was written or formatted. After extracting the text from the resume, the following step is to clean it so that the system can interpret it better. Preprocessing is the term used to describe this step in the process. It entails a few easy but critical processes, such as converting all of the text to lowercase and deleting items that aren't really relevant, such as punctuation marks, symbols, and needless numbers. We also delete common terms such as "the," "is," and "in," which add no meaningful significance to the content. The text is then broken into individual words or tiny phrases, a process known as tokenization.

Once we have clean, independent words, we use something called lemmatization or stemming. This helps to reduce words to their root form. For example, "running," "ran," and "runs" are all shortened to "run." This allows the system to treat comparable terms consistently, even if they are written differently. The entire preprocessing is visualized in Figure 1.

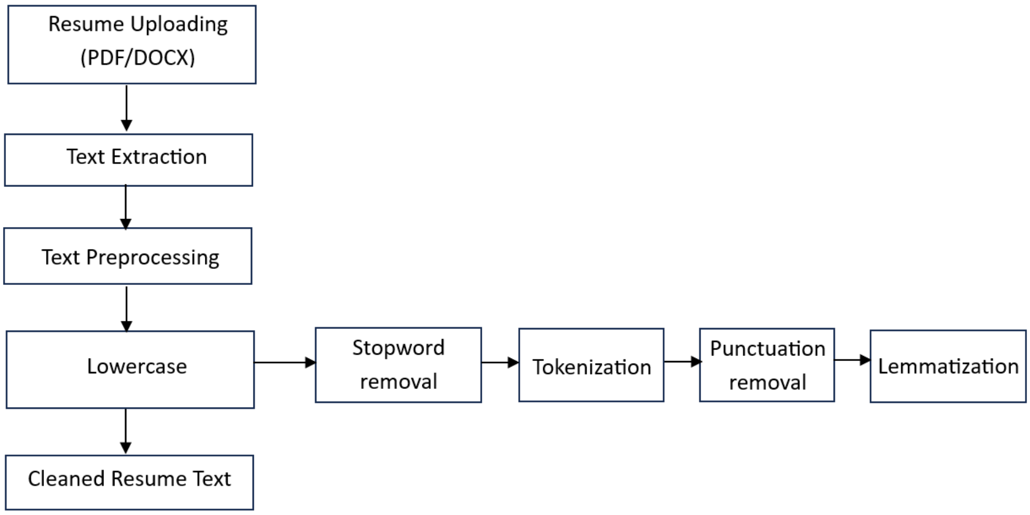


Figure 1: Preprocessing Workflow

After the text has been cleaned and ready, the system uses NLP to locate key details. For this, we employ a technique known as Named Entity Recognition (NER). This allows the system to identify useful information such as the candidate's talents, education, job titles, work experience, and even the tools or programming languages they have used. So, if someone has written “Graduated in B.Tech (IT)” the system will catch those and save them for further analysis. An example to visualize how NER extracts information is presented in Table 1.

Raw Resume Text	Extracted Info	Information type
Graduated in B.Tech (IT)	B.Tech (IT)	Education
Worked as a Python Developer for 2 years	Python developer	Job title
Skills: Python, SQL, Problem Solving	Python, SQL	Technical Skills

Table 1: Key information extracted via NER

All this data is maintained in a structured manner, similar to a basic table, where each resume is divided into neat portions. One resume might feature abilities such as Java, communication, and problem-solving, while another resume is listing their experience such as "Software Developer at A Company, 2 years." The system strives to organize everything nicely, regardless of how it is expressed on the CV. Another key aspect of this step is dealing with diverse resume styles. Everyone writes their resumes differently. Some people clearly mark their parts as "Education" or "Skills," whilst others may simply write it in a paragraph. This algorithm is designed to organize these portions using both keywords and context, ensuring that nothing important is overlooked. Personal information like their phone numbers or email addresses are not included in this procedure, we only focus on professional content. This also promotes justice and privacy.

By the end of the preparation stage, each CV has been cleaned, organized, and is ready to be matched to a job description. Without this stage, the system would be unable to determine what a candidate has accomplished or how well they suit a specific role. So, this stage ensures that we begin the following step with clean, meaningful data that is straightforward to compare.

3.2. Matching and Shortlisting Stage

After the resumes are cleaned and formatted during the preparation step, the system begins the process of comparing them to the job descriptions. This is where the real matching and shortlisting occurs. The goal at this stage is to determine which resumes are the best fit for the job profile, not just based on exact word matches, but also on a deeper comprehension of the content. Instead of relying solely on keyword presence, our approach employs natural language processing techniques to examine the resume and job description in a more relevant manner.

At the heart of this stage is semantic matching. This means the system compares the actual meaning of the words and phrases used in resumes with those in the job description. This method allows the system to recognize that people regularly use multiple sentences to communicate the same concept. For example, one person may write "software developer" on their resume whereas the job description says "software engineer." A simple system may treat these as entirely different duties. But our approach understands the meaning and sees that both actually refer to similar kinds of work.

The system accomplishes this by converting both resumes and job descriptions into numerical format utilizing language models such as Word2Vec or BERT. These models are trained to recognize how words are used in sentences and in what context they appear. Once the text has been converted into vectors, they are compared using a method known as cosine similarity. Cosine similarity determines how similar two texts are by comparing the vectors' direction. A higher score indicates that the resume is a better fit for the job description. Semantic matching workflow is presented in figure 2.

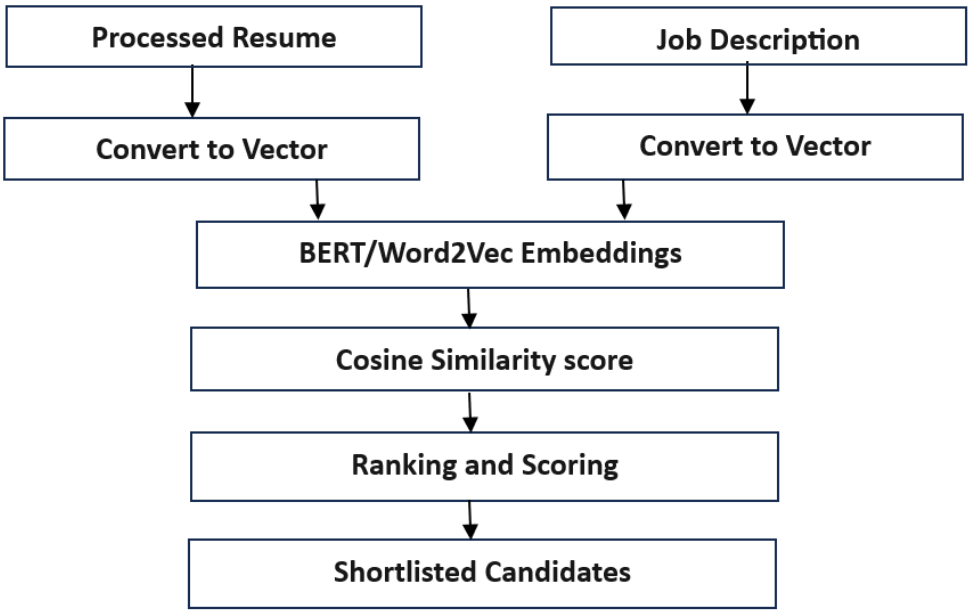


Figure 2: Semantic Matching Workflow

This comparison yields a similarity score for each resume. Resumes that incorporate abilities, experiences, and education that are relevant to the job requirements obtain higher marks. Others who are less relevant receive lower marks. This allows the machine to sort and rate all resumes from most to least suitable. Once all the scores are ready, the system moves to the shortlisting part. There are two ways this can be done. One method is setting a fixed cut-off score. For example, only resumes that score above 70 percent are considered. The other method is picking the top few resumes with the highest scores The similarity scores of different resumes, showing how closely each resume matches the job description based on semantic analysis is represented in Figure 3.

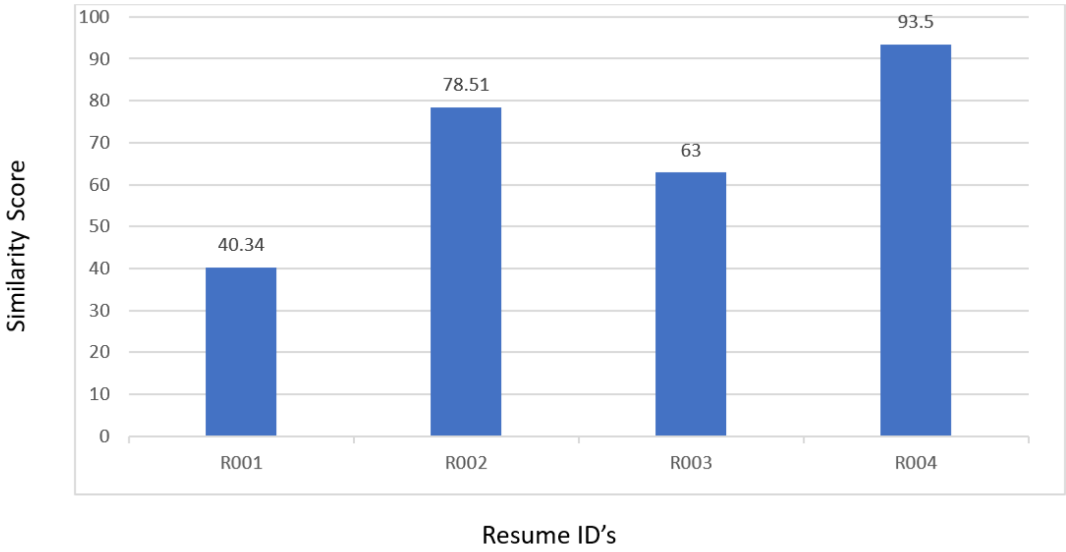


Figure 3: Similarity scores for sample resumes

One of the main advantages of this stage is that it doesn't punish candidates just because they described things differently. A lot of good candidates use unique ways to explain their work, and older systems may miss them. But since our model focuses on meaning, it is able to catch what they actually meant. This makes the shortlisting fairer and ensures that qualified people are not left out. This stage also saves a significant amount of manual effort. Recruiters do not need to open every single resume. They simply need to look at those that the system has already identified as relevant. This accelerates, streamlines, and simplifies the employment process for all parties involved. In short, the matching and shortlisting stage turns all the information gathered in the preparation stage into real results. By using smart comparisons and a deeper understanding of language, the system is able to fairly and efficiently identify the most suitable candidates for the job.

IV. SYSTEM ARCHITECTURE

Our resume screening technology is designed to work in a straightforward and logical manner, much like a recruiter would analyze resumes but much faster and without bias. Each component of the system has a distinct functithon, and together they manage everything from reviewing resumes to picking the best ones. This is how it works.

i. User input
The process begins when the recruiter uploads resumes and provides a job description. This is performed through a simple interface in which users may choose variables such as a minimum matching score or the number of applications to shortlist.

ii. Reading Resume Content
Once resumes are uploaded, the system reads the text inside them. Since people usually submit resumes in formats like PDF or Word, we use tools that can pull out the plain text from these files. This helps the system understand what's written in the resume, regardless of how it's formatted.

iii. Clean the text (preprocessing)
The next step is to polish up the resume text. Most resumes have excessive formatting, symbols, or superfluous words that are ineffective. The system simplifies the text by doing things like:

- Making all the text lowercase.
- Eliminating punctuation, symbols, and extra digits.
- Ignoring commonly used terms like "is," "the," or "and"
- Dividing the material into smaller chunks, such as words or brief phrases.
- Converting comparable words to a basic form.

This allows the system to focus on what is most important.

iv. Extracting Important Details
Now that the text is clean, the algorithm attempts to comprehend what the speaker is saying. Using language analysis technologies, it searches for important information such as skills, education, experience, and job duties. For example, if someone says "Skilled in Java and Python" or "Worked as a

Software Engineer," the system recognizes it and saves it as relevant information.

v. Matching Resumes with Job Descriptions

This is where things get smarter. The technology does more than just search for terms that match. It use language models to attempt to understand what the resume and job description mean. These models are transformed into a type of code (called vectors), which are then compared using a method known as cosine similarity.It defines how closely they relate to one another in meaning.

The better a resume matches the employment requirements, the higher its score. Even if someone uses different terms to describe their experience, the algorithm can still determine whether it is relevant or not.

vi. Matching the resume with the job description.

After reviewing all resumes, the system ranks them from best to least relevant.

The recruiter can then decide whether to set a minimum score (for example, only shortlist those with scores greater than 70%) or just select the top few. The selected resumes are then presented on the screen, ready for examination by the recruiter.

This method speeds up the resume evaluation process, makes it more equitable, and reduces stress. It ensures that good candidates are not ignored just because they used different wording and it saves recruiters a significant amount of time during the early stages of hiring.

V. RESULTS

To determine how well our resume screening system works, we divided the testing procedure into two major stages: (i) Categorizing resumes based on their domain or field, and (ii) Comparing resumes with job descriptions and shortlisting the most relevant ones. Both of these methods were tested using real-world resume data and machine learning models to ensure that the system is functional and generates relevant results.

5.1. Resume classification

Before comparing resumes and job descriptions, the system determines which domain each resume belongs in. This step is significant since it improves shortlisting accuracy by focusing solely on applicants who actually work in the chosen field. To accomplish this, we trained numerous machine learning models on a dataset of resumes that had previously been labelled with their respective fields, such as IT, HR, Management, Finance, etc.

Several classification models, including Naive Bayes, Logistic Regression, Random Forest, and Support Vector Machines (SVM) are tested. A TF-IDF vectorizer is used for transform the resume text to numerical values, which allowed the models to read and manage the data more effectively. Throughout the review, we used 10-fold cross-validation to ensure that our results are consistent and fair.

The Linear SVM yielded the most accurate results of any model we tested. It outperformed the other models in terms of classification accuracy, reaching around 79.2%.

As a result, we used Linear SVM as our final model for resume categorization. This stage helps in deleting resumes that did not fit the intended job domain before moving on to the next phase. Classification accuracies of each model are shown in the below table.

Classification accuracies of each model are shown in the below table.

Classifier	Accuracy
Random Forest	41.5%
Multinomial Navie Bayes	48.7%
Logistic Regression	60.3%
Linear SVM	79.2%

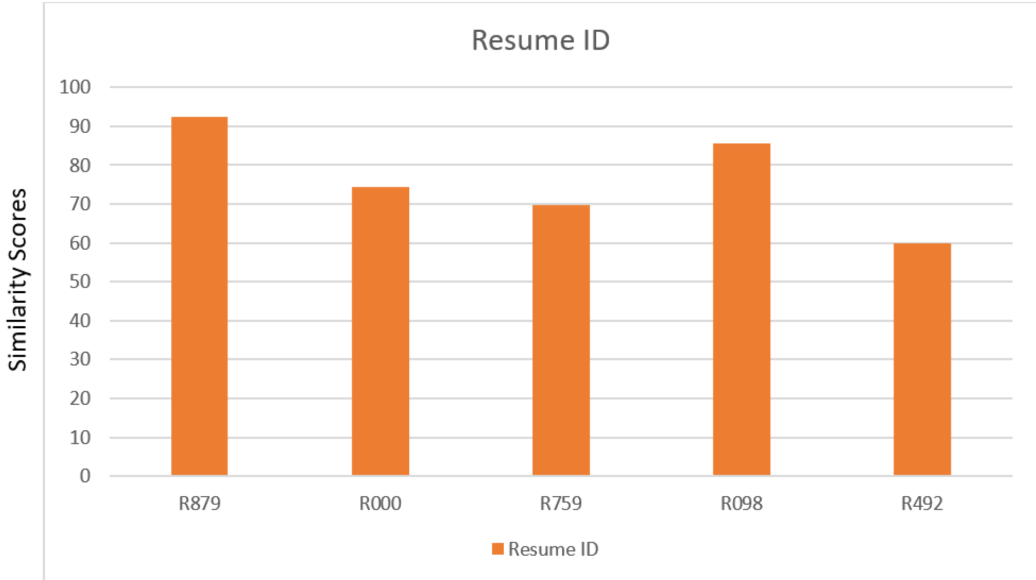
Table: Accuracy Comparison of different classifiers

5.2. Semantic Matching and Shortlisting

After the classification step, the next part of the process is to match the resumes with the job description. Our system goes beyond merely looking for matched keywords and performs semantic analysis. This means that the system tries to understand what is being said in the resume and job description, rather than simply seeing if the same terms appear.

To do this, we employed models such as Word2Vec and BERT to convert resumes and job descriptions into vectors, which are numerical formats that preserve the text's meaning and context. After conversion, we used cosine similarity to compare the vectors.

Cosine similarity measures how comparable the resume is to the job description based on the angle between their vector representations. Each resume was given a score based on how well it matched the job description. The higher the score, the better the match. All resumes were then ranked from highest to lowest based on their scores. We also set a threshold value (for example, 70%), and only those resumes that scored above this limit were shortlisted. Because of this semantic approach, the system was able to pick up on resumes that matched the job description in meaning, even if they didn't use the exact same words. For example, if the job description said "software engineer," a resume that mentioned "developer" or "programmer" with similar context would still be selected. The following graph represents the similarity scores of resumes ranked against job requirements.



VI. CONCLUSION

Manually reviewing resumes may take a significant amount of time and effort, especially when there are hundreds of candidates for a single position. Sometimes good applicants are overlooked simply because they did not utilize the appropriate phrases or because their résumé was written incorrectly. This program attempts to make the screening process quicker, smarter, and more equitable. The primary goal of this strategy was to reduce recruiters' effort by using technology into the first screening phases. Our approach analyzes and contrasts a résumé with a job description using machine learning and natural language processing (NLP), even when the terminology used are different. It initially classifies resumes by domain and then compares them semantically to job criteria. This improves the shortlisting process' accuracy.

Unlike traditional keyword matching systems, ours tries to look at the meaning behind the words. So even if someone uses a different term, their resume won't be ignored if they're actually a good fit. Even if a competent applicant did not include the perfect phrase on their resume, this method ensures that they all have an equal chance of getting shortlisted and prevents prejudice.

All things considered, this study indicates that resume screening may become considerably more successful and efficient with the correct technology. In addition to reducing the recruiter's effort, it improves the likelihood of finding the most qualified applicants. In the future, this kind of technology may be a standard part of recruiting practices for businesses.

REFERENCES

[1] Roy, Pradeep Kumar, Sarabjeet Singh Chowdhary, and Rocky Bhatia. "A Machine Learning approach for automation of Resume Recommendation system." *Procedia Computer Science* 167 (2022): 2318-2327. https://www.sciencedirect.com/science/article/pii/S187705092030750X

[2] Tejaswini, K., V. Umadevi, Shashank M. Kadiwal, and Sanjay Revanna. "Design and development of machine learning based resume

ranking system." *Global Transitions Proceedings* 3, no. 2 (2022): 371-375. https://www.sciencedirect.com/science/article/pii/S2666285X21001011

[3] Pisanelli, Elena. "Your resume is your gatekeeper: Automated resume screening as a strategy to reduce gender gaps in hiring." *Economics letters* 221 (2022): 110892. https://www.sciencedirect.com/science/article/abs/pii/S0165176522003664

[4] Pant, Dipendra, Dhiraj Pokhrel, and Prakash Poudyal. "Automatic software engineering position resume screening using natural language processing, word matching, character positioning, and regex." In *2022 5th International Conference on Advanced Systems and Emergent Technologies (IC_ASET)*, pp. 44-48. IEEE, 2022. https://ieeexplore.ieee.org/abstract/document/9765916

[5] Mishra, Sneha, Pallavi Mehta, and Sairaj Navalkar. "The Importance of Having a Screening Process During Recruitment." *Asian Journal of Multidimensional Research* 13, no. 7 (2024): 1-7. https://tarj.in/wp-content/uploads/paper/AJMR/2024/FULL-PDF/AJMR-JULY-2024/7.1,%20Sneha%20Mishra.pdf

[6] Mariappan, Prabha, Kali Krishna, Janmejaya Sahoo, and Sai Preetham. "Smart Resume Screening and Job Represents Recommendation System." *Available at SSRN 5141402*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5141402

[7] Amro, Belal, Arwa Najjar, and Mario Macido. "An intelligent decision support system for recruitment: Resumes screening and applicants ranking." (2022). https://dspace.hebron.edu/jspui/handle/123456789/1124

[8] Herold, Marcel, and Marc Roedenbeck. "Usage of AI in the Recruitment and Selection Process: A Systematic Review and Computational Analysis." *Authorea Preprints* (2024). https://advance.sagepub.com/doi/full/10.31124/advance.173458936.60463045

[9] Gorda, Anak Agung Ngurah Eddy Supriyadinata, Kadek Devi Kalfika Anggria Wardani, and I. Gusti Ngurah Widya Hadi Saputra. "Beyond Automation: A Systematic Review of AI in Employee Recruitment." In *2024 10th International Conference on Smart Computing and Communication (ICSCC)*, pp. 666-670. IEEE, 2024. https://ieeexplore.ieee.org/abstract/document/10690629

[10] Rojas-Galeano, Sergio, Jorge Posada, and Esteban Ordoñez. "A Bibliometric Perspective on AI Research for Job-Résumé Matching." *The Scientific World Journal* 2022, no. 1 (2022): 8002363.

https://onlinelibrary.wiley.com/doi/full/10.1155/2022/8002363

[11] Dixit, Sweta, Neha Sharma, Mohit Maurya, and Mridul Dharwal. "AI power: making recruitment smarter." In *Evolution of digitized societies through advanced technologies*, pp. 165-180. Singapore: Springer Nature Singapore, 2022. https://link.springer.com/chapter/10.1007/978-981-19-2984-7_14

[12] Albassam, Wael Abdulrahman. "The power of artificial intelligence in recruitment: An analytical review of current AI-based recruitment strategies." *International Journal of Professional Business Review: Int. J. Prof. Bus. Rev.* 8, no. 6 (2023): 4. http://agora.edu.es/servlet/articulo?codigo=8994309

[13] Hunkenschroer, Anna Lena, and Christoph Luetge. "Ethics of AI-enabled recruiting and selection: A review and research agenda." *Journal of Business Ethics* 178, no. 4 (2022): 977-1007. https://link.springer.com/article/10.1007/S10551-022-05049-6

[14] Madanchian, Mitra, Hamed Taherdoost, and Nachaat Mohamed. "AI-based human resource management tools and techniques; A systematic literature review." *Procedia Computer Science* 229 (2023): 367-377. https://www.sciencedirect.com/science/article/pii/S187705092302029X

[15] Basha, H. A. S., N. Rajitha, J. A. Roslin, K. K. Thoti, M. R. P. Mohammed Khalid, and B. R. Mishra. "AI-powered Recruitment and Employee Selection: Evaluating Bias and Fairness in Hiring Practices." *European Economic Letters (EEL)* 14, no. 1 (2024): 827-834.

https://www.researchgate.net/profile/Kiran-Kumar-Thoti/publication/378416093_AI-powered_Recruitment_and_Employee_Selection_Evaluating_Bias_and_Fairness_in_Hiring_Practices/links/666160eb85a4ee7261a8f737/AI-powered-Recruitment-and-Employee-Selection-Evaluating-Bias-and-Fairness-in-Hiring-Practices.pdf

[16] Cai, Fei, Jiashu Zhang, and Lei Zhang. "The impact of artificial intelligence replacing humans in making human resource management decisions on fairness: A case of resume screening." *Sustainability* 16, no. 9 (2024): 3840. https://www.mdpi.com/2071-1050/16/9/3840

[17] Black, J. Stewart, and Patrick van Esch. "AI-enabled recruiting: What is it and how should a manager use it?." *Business horizons* 63, no. 2 (2020): 215-226. https://www.sciencedirect.com/science/article/abs/pii/S0007681319301612

[18] D'Souza, Max Constance, and Mooon Paiithannkar. "AI-POWERED RECRUITMENT: TRANSFORMING THE HIRING PROCESS." https://ijoeete.com/wp-content/uploads/2024/09/29-max-d.pdf

[19] Golla, Hemanth Kumar Yadav, Usman Basha Peddinti, Koushik Kotla, Vandana Todimela, and Uday Kiran Uppara. "Resume classification and human-free virtual AI interview system." In *AIP Conference Proceedings*, vol. 3237, no. 1. AIP Publishing, 2025. https://pubs.aip.org/aip/acp/article-abstract/3237/1/020021/3337609/Resume-classification-and-human-free-virtual-AI

[20] Lacroux, Alain, and Christelle Martin-Lacroux. "Should I trust the artificial intelligence to recruit? Recruiters' perceptions and behavior when faced with algorithm-based recommendation systems during resume screening." *Frontiers in Psychology* 13 (2022): 895997. https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2022.895997/full