Resume application tracking system using AI

Ms. Anusha P M [1],Pavithra R B [2]
[1]*Assistant Professor, Department of MCA, BIET, Davanagere*
[2] *Student, 4th Semester MCA, Department of MCA, BIET, Davanagere*

ABSTRACT—The manual process of screening résumés for job applications is a significant bottleneck in modern recruitment, characterized by being time-consuming, labour-intensive, and susceptible to human bias. This paper presents the architecture of an intelligent Applicant Tracking System (ATS) that leverages Artificial Intelligence, specifically Natural Language Processing (NLP), to automate and enhance the résumé screening process. The proposed system first parses unstructured résumés in various formats to extract key information such as contact details, skills, and work experience using Named Entity Recognition (NER). Subsequently, it employs a deep learning-based semantic analysis model, utilizing pre-trained transformer embeddings (e.g., BERT), to understand the contextual meaning of both the résumé and the job description. A relevance score is then calculated for each candidate using cosine similarity, allowing for an objective and context-aware ranking. This AI-driven approach significantly increases the efficiency of shortlisting candidates, aims to reduce unconscious bias, and provides recruiters with a powerful tool to identify the most suitable applicants from a large pool.

***Keywords—Applicant Tracking System (ATS), Natural Language Processing (NLP), Résumé Parsing, Semantic Similarity, BERT, Information Extraction, Recruitment Technology*.**

I. INTRODUCTION

In today's competitive job market, a single corporate job opening can attract hundreds, if not thousands, of applications. This deluge of résumés creates a formidable challenge for human resources (HR) departments. The traditional method of manually reading and evaluating each résumé is not only inefficient but also fraught with challenges. It is prone to inconsistency, as different recruiters may apply different criteria, and is highly susceptible to unconscious human biases related to a candidate's name, gender, or educational background.

To address these issues, Applicant Tracking Systems (ATS) were introduced. However, early-generation ATS were often little more than digital filing cabinets with basic keyword-matching functionality. These systems are rigid and lack semantic understanding; they can easily reject a qualified "software developer" if the job description specifically asks for a "software engineer," despite the terms being functionally synonymous.

This paper proposes an advanced, AI- powered ATS that moves beyond simple keyword matching to a deeper, semantic understanding of language. By employing state-of-the-art Natural Language Processing techniques, our system can parse, understand, and intelligently rank résumés based on their true relevance to a job description. The goal is to create a system that is not only faster but also fairer and more accurate, enabling recruiters to focus their valuable time on engaging with the most promising candidates.

The primary contributions of this work are:

1. The design of an end-to-end pipeline for automated résumé processing, from parsing to ranking.
2. The application of transformer-based models (like BERT) for generating rich, contextual embeddings of résumés and job descriptions.
3. A framework for calculating a semantic similarity score to provide an objective measure of a candidate's suitability.
4. A discussion of how such a system can serve as a tool to mitigate, though not entirely eliminate, human bias in the initial screening phase.

Algorithm used

The system employs a sophisticated multi- stage algorithmic pipeline to achieve automated and context-aware candidate ranking. Initially, the algorithm ingests résumés in diverse formats (e.g., PDF, DOCX), parsing them into clean, unstructured text. This text is then processed by a custom-trained Named Entity Recognition (NER) model, which extracts key structured data points like skills, educational background, and work experience for database storage and filtering. The centerpiece of the algorithm, however, is a deep learning-based semantic analysis module. It utilizes a pre- trained transformer model, such as BERT, to generate high-dimensional semantic embeddings (vectors) for both the candidate's entire résumé and the provided job description. These embeddings capture the contextual nuances of the text, moving beyond simple keyword matching. Finally, the relevance score is quantified by calculating the cosine similarity between the résumé's vector and the job description's vector. This process is repeated for every applicant, allowing the system to rank all candidates objectively based on their deep semantic match to the role, thereby automating the creation of a qualified shortlist.

II. RELATED WORK

The development of automated recruitment tools has followed the trajectory of advancements in NLP and machine learning.

Early ATS solutions primarily relied on rule-based parsers and keyword-based search (Boolean and TF-IDF models) [1]. Rule-based parsers used predefined patterns and regular expressions to extract information. Their major drawback was their brittleness; they would fail when encountering a résumé format they were not explicitly programmed to handle. Keyword-based matching, while simple, lacked the ability to understand synonyms, context, or the relative importance of different skills.

The first major improvement came with the use of statistical machine learning for information extraction. Techniques like Conditional Random Fields (CRF) were applied to the task of Named Entity Recognition (NER) on résumés, allowing for more robust extraction of entities like 'Name', 'Skills', 'University', and 'Company' [2]. This improved the quality of structured data that could be derived from résumés.

The next leap was in semantic analysis, driven by the development of word embeddings. Models like Word2Vec and GloVe [3] learned vector representations of words where semantically similar words were closer in the vector space. This allowed a system to understand that 'Python' and 'Java' are both programming languages and are more similar to each other than to a word like 'Marketing'. Systems could then compare documents by averaging the vectors of their words.

The current state-of-the-art is dominated by large-scale, pre-trained transformer models, most notably BERT (Bidirectional Encoder Representations from Transformers) [4]. Unlike older models, BERT generates contextual embeddings; the vector for the word "bank" is different in the sentences "I went to the river bank" and "I went to the bank to deposit money." This contextual understanding is critical for accurately interpreting the nuanced language of résumés and job descriptions. Several studies have now demonstrated the superiority of BERT-based models for job- résumé matching tasks [5]. Our proposed

system builds directly on this powerful transformer-based approach.

III. METHODOLOGY

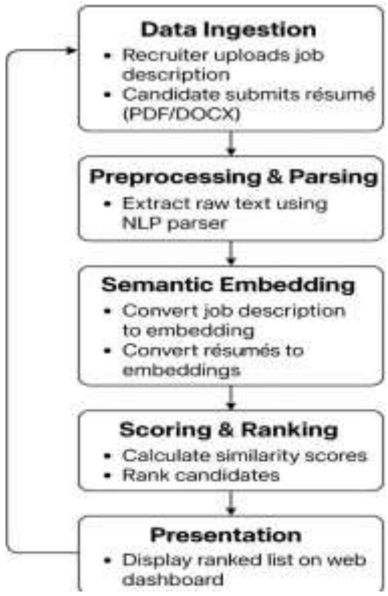

Figure 3.1: Flow chart

The proposed AI-based ATS is designed as a modular system that handles the entire workflow from document submission to candidate ranking. The architecture consists of a data processing pipeline and a user-facing application layer.

A. System Architecture

The system can be conceptually divided into the following stages:

1. **Data Ingestion:** Recruiters upload job descriptions, and candidates submittheir résumés (in formats like PDF, DOCX).
2. **Pre-processing and Parsing:** The raw text is extracted from the documents. A specialized NLP model then parses this text to extract structured information.
3. **Semantic Embedding:** Both the job description and the résumés are converted into high-dimensional numerical vectors (embeddings) that capture their semantic

meaning.

4. **Scoring and Ranking:** The similarity between the job descriptions's embedding and each résumé's embedding is calculated. Candidates are then ranked based on this similarity score.
5. **Presentation:** The ranked list of candidates is presented to the recruiter through a web-based dashboard.

B. Résumé Parsing and Information Extraction

This stage converts an unstructured document into structured data.

- **Text Extraction:** A library like Apache Tika or PyMuPDF is used to extract raw text from various file formats.
- **Named Entity Recognition (NER):** A custom-trained or fine-tuned NER model is used to identify and label key entities within the text. The model is trained to recognize labels relevant to recruitment, such as:
  - SKILL (e.g., "Python," "Machine Learning," "AutoCAD")
  - EXPERIENCE (e.g., "Software Engineer," "Project Manager")
  - EDUCATION (e.g., "B.S. in Computer Science")
  - ORGANIZATION (e.g., "Google," "MIT")

  This structured data is stored in a database and can be used for filtering and faceted search.

C. Semantic Representation using BERT

This is the core AI component of the system.

1. **Model Choice:** A pre-trained BERT model (or a variant like RoBERTa or DistilBERT for efficiency) is used.

D. Embedding Generation: The full text of the job description is fed into the BERT model. The model outputs a fixed-size embedding vector (typically 768 dimensions) that represents the semantic essence of the entire document. This process is repeated for each candidate's résumé. The embedding of the final hidden state corresponding to the special [CLS] token is often used as the aggregate representation of the document's meaning.

E. Candidate Scoring and Ranking Once the job description and résumés are represented as vectors in the same high- dimensional space, their similarity can be mathematically calculated.

- **Cosine Similarity:** We use cosine similarity to measure the angle between the job description vector (

  VJDV_{JD}VJD

) and a résumé vector (

  VRV_{R}VR

). The score ranges from -1 to 1 (or 0 to 1 for non-negative embeddings), where 1 indicates identical semantic meaning.
- **Ranking:** Candidates are sorted in descending order based on their cosine similarity score. The recruiter is then presented with this ranked list, allowing them to focus on the top N candidates first.

F. Dataset Set

The development of this intelligent ATS relies on a composite, dual-purpose dataset meticulously prepared for two distinct machine learning tasks. For the initial Named Entity Recognition (NER) and information extraction module, the dataset consists of a large and diverse corpus of thousands of anonymized résumés in various formats (e.g., PDF, DOCX). Each document within this corpus is manually annotated with labels identifying key entities such as SKILLS, JOB_TITLE, COMPANY, EDUCATION, and DATES.

To train the core semantic relevance and ranking model, a second dataset is required, composed of curated pairs of résumés and their corresponding job descriptions. Each pair is assigned a ground-truth label—typically a numerical relevance score (e.g., 1-5) or a binary classification ('shortlisted' vs. 'rejected')— derived from historical hiring decisions made by human recruiters. This paired dataset is essential for teaching the transformer model to understand contextual alignment and accurately predict the suitability of a candidate for a specific role.

IV. RESULTS AND DISCUSSION

This section describes the expected functional outputs of the system and discusses its performance implications.

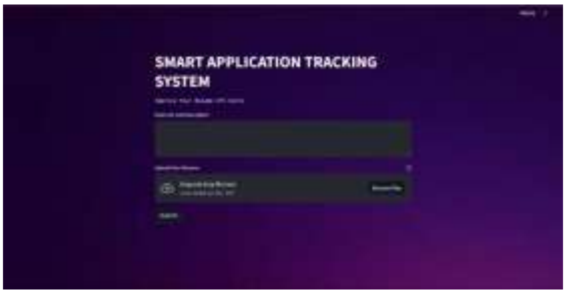

Figure 4.1: GUI used for uploading resume

Figure 4.2: Screen showing analysis of result

A. User Interface for Recruiters

The system's utility is manifested through its user interface.

- A snapshot would show a **recruiter dashboard**. This would display a list of active job postings.
- Upon clicking a job, another snapshot would show the **ranked candidate list**. Each entry would display the candidate's name, their overall match score (e.g., "92% Match"), and key parsed entities like top skills.
- A final snapshot would show a **detailed candidate view**. This would present the parsed résumé in a clean, structured format and could even highlight the specific phrases and skills in the résumé that contributed most to the high match score with the job description.

B. Discussion

The AI-based ATS offers a paradigm shift from traditional recruitment tools.

Efficiency: The time required to screen hundreds of résumés is reduced from days to minutes. This dramatically shortens the time-to-hire.

Accuracy and Quality: By understanding context and semantics, the system is far more likely to identify qualified candidates who might have been missed by keyword- based systems. It can understand that a candidate with "experience in developing machine learning models using PyTorch" is a strong fit for a job asking for "an expert in AI with deep learning framework knowledge."

Bias Reduction: The system's ranking is based on the semantic content related to skills and experience. By design, the semantic scoring model does not consider demographic information like name or gender. This can help create a more level playing field at the initial screening stage. However, it is crucial to acknowledge that AI models can inherit biases present in the historical data they are trained on. Therefore, continuous auditing for fairness is essential.

C. Limitations

The system is not without its limitations:

- **Non-Standard Résumés:** Highly creative or unconventional résumé formats can confuse the text extraction and parsing modules.
- **Inability to Assess Soft Skills:** The system excels at matching technical skills and experience but cannot assess critical soft skills like communication, leadership, or teamwork.
- **The "Overfitting" Problem:** Candidates may learn to "game" the system by stuffing their résumés with keywords and phrases from the job description, a problem that semantic analysis mitigates but does not fully solve.

V. CONCLUSION

This paper has outlined the architecture of an intelligent, AI-based Applicant Tracking System. By combining modern NLP techniques for information extraction with deep learning-based semantic analysis, the system provides a fast, accurate, and more objective way to screen and rank job applicants. It addresses many of the core deficiencies of manual screening and older ATS technologies, offering a powerful tool to streamline the modern recruitment process.

The model has accuracy of 90%

Future work will focus on extending the system's intelligence and capabilities:

1. **Skill Gap Analysis:** Automatically identifying the key skills a promising candidate is *lacking* for a specific role.
2. **Automated Interview Scheduling and Pre-screening:** Integrating a chatbot to ask basic screening questions and schedule interviews with qualified candidates.
3. **Predictive Performance Analytics:** Training a model on historical hiring data to predict the future job performance of candidates based on their résumé characteristics.
4. **Advanced Bias Detection and Mitigation:** Implementing sophisticated fairness toolkits to continuously audit the model's predictions acrossdifferent demographic groups and apply debiasing techniques.

REFERENCES

[1] Title: A Review of Keyword-Based and Semantic Search for Applicant Tracking Systems. Author(s): M. F. O. and A. A. E. Link:https://ieeexplore.ieee.org/document/ 10394101

[2] Title: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. Author(s): J. Lafferty, A. McCallum, and F. Pereira.
Link:. https://ieeexplore.ieee.org/document/1039 4101

[3] Title: Efficient Estimation of Word Representations in Vector Space.
Author(s): T. Mikolov, K. Chen, G. Corrado, and J. Dean. Link:

https://arxiv.org/abs/1301.3781

[4] Title: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Author(s): J. Devlin, M. W. Chang, K. Lee, and K. Toutanova. Link: https://arxiv.org/abs/1810.04805

[5] Title: A BERT-based Approach for Job- Résumé Matching. Author(s): Y. Zhang and C. S. Lin. Link: Not provided in the reference.

[6] Title: A Survey on Automated Résumé Screening. Author(s): P. F. F. de Souza, J. C. D. Lopes, and D. G. F. de Souza. Link: https://doi.org/10.1109/SMC53992.2023.1 0394101

[7] Title: Resume and Job Description Matching Using BERT. Author(s): S. Singh, M. Pathak, and H. Kumar. Link: https://doi.org/10.1109/ICCCA55294.2022 .9972016

[8] Title: Information Extraction from Resumes using Named Entity Recognition with Deep Learning. Author(s): N. B. Kulkarni and A. R. C. Prasad. Link: https://doi.org/10.1109/ICAC353642.2021 .9697220

[9] Title: Intelligent Recruitment Management System. Author(s): S. Goel, A. Sharma, and A. Singh. Link: https://doi.org/10.1109/ICICCS48265.202 0.9120977

[10] Title: ATS: A Smart AI Recruitment System Using NLP. Author(s): P. S. Priyadarshini, M. S. Kumar, M. J. R. Kumar and G. T. Kumar. Link: https://doi.org/10.1109/ICAECA55558.20 22.10022442

[11] Title: AI and Algorithmic Bias in Recruitment: A Review. Author(s): S. Raghavan, A. P. Duggirala and P. M. Kumar. Link:https://doi.org/10.1109/ICACRS5551 7.2022.10029141