

## **An Explainable Resume Analyzer Using ESCO Ontology and Semantic Skill Mapping for Transparent Candidate Ranking**

Sushma Malik<sup>1</sup>, Anamika Rana<sup>2</sup>, Ayush Pal<sup>3</sup>, Ritham Gupta<sup>4</sup>

<sup>1</sup> Assistant Professor, <sup>2</sup> Associate Professor, <sup>3</sup> Research Scholar

<sup>1,2,3,4</sup> Department of Computer Applications, Maharaja Surajmal Institute, New Delhi, India

<sup>1</sup>sushmalik25@gmail.com, <sup>2</sup>anamika.rana@gmail.com, <sup>3</sup>ayush34492@gmail.com

---

**Abstract:** Traditional recruitment processes often rely heavily on manual resume screening, which is both time-consuming and prone to human bias. These challenges can lead to inefficiencies, inconsistent evaluations, and potential overlooking of qualified candidates. This study presents a locally deployed, desktop-based Resume Analyzer designed to automate the initial screening phase while ensuring enhanced data privacy for small and medium-sized organizations. Unlike cloud-based Applicant Tracking Systems (ATS), the proposed system operates entirely on a local machine, reducing the risk of sensitive applicant data exposure.

The system utilizes Natural Language Processing (NLP) techniques to extract and analyze textual content from resumes in PDF, DOCX, and TXT formats. A specialized algorithm is implemented to accurately calculate total work experience by interpreting employment durations and aggregating them systematically. To standardize skill identification and improve matching accuracy, the system integrates a mapping mechanism based on the ESCO (European Skills, Competences, Qualifications and Occupations) framework developed by the European Commission. This ontology-driven approach ensures alignment with recognized occupational and skills classifications, promoting consistency and transparency in candidate evaluation.

Candidates are ranked using a configurable weighted scoring model that evaluates three primary components: skills (50%), work experience (30%), and education (20%). The scoring system provides clear justifications for rankings, enhancing interpretability and recruiter trust. Experimental evaluation demonstrates that the system effectively identifies top candidates while maintaining fairness and transparency. The findings suggest that localized, ontology-based resume analysis tools can serve as a practical, cost-effective alternative to cloud-hosted ATS solutions, particularly in privacy-sensitive environments where data security and control are critical considerations.

**Keywords:** Resume Analysis, ESCO Ontology, Semantic Skill Mapping, Candidate Ranking, Explainable AI (XAI), Natural Language Processing (NLP), Skills Classification

---

### **1. Introduction**

Modern recruitment processes must handle an overwhelming number of resumes for a single job vacancy. With online job portals and digital applications becoming the norm, recruiters often receive hundreds or even thousands of resumes per posting. Manually screening such a large volume of documents is time-consuming, inefficient, and prone to inconsistencies or unconscious bias[1]. To address this issue, many organizations have adopted Applicant Tracking Systems (ATS) to automate resume parsing, indexing, and filtering. These systems help streamline hiring workflows by extracting structured information and shortlisting candidates based on predefined criteria[2].

However, most commercially available ATS platforms are cloud-based, subscription-driven, and feature-rich systems designed primarily for medium to large enterprises. While they offer scalability and advanced analytics, they often involve recurring costs, complex configuration processes, and dependency on internet connectivity. For small businesses, startups, academic institutions, and research laboratories, such solutions may not be practical or affordable. In addition, cloud-hosted systems raise concerns regarding data privacy and regulatory compliance, particularly when handling sensitive personal information such as employment history, contact details, and educational credentials. In privacy-sensitive environments, organizations may prefer to retain full control over applicant data by processing it locally rather than transmitting it to external servers[3].

To address these limitations, this study proposes a lightweight, desktop-based Resume Analyzer that operates entirely within a local environment. By functioning offline, the system ensures that candidate data never leaves the organization's infrastructure, thereby reducing risks associated with data breaches or unauthorized access.

This localized approach is especially suitable for small offices and educational labs that require cost-effective and secure recruitment tools. At the same time, the system aims to maintain a level of analytical capability comparable to more sophisticated ATS platforms.[4]

A key technical challenge in automated resume screening lies in transforming unstructured textual content into structured, machine-readable information. Resumes are typically submitted in various formats, such as PDF and Word documents, and may differ significantly in layout, design, and writing style. Extracting relevant details—particularly skills, work experience, and educational qualifications—requires robust parsing and text-processing techniques. Variations in section headings, inconsistent date formats, and graphical elements can further complicate information extraction. Therefore, developing a reliable parser capable of handling multi-format documents and diverse resume structures is a critical component of the proposed system[5].

Beyond extraction, normalization of skills presents another significant challenge. Candidates often describe similar competencies using different terminology. For example, “Java programming,” “Java development,” and “object-oriented programming in Java” may refer to related skill sets but appear distinct in raw text. Without semantic standardization, automated ranking systems may fail to recognize equivalencies or closely related competencies. To address this issue, the proposed solution incorporates a skill normalization mechanism based on the ESCO (European Skills, Competences, Qualifications and Occupations) framework. ESCO provides a structured, multilingual classification of skills and occupations, enabling consistent mapping between resume content and standardized skill definitions. By leveraging this ontology-based approach, the system improves matching accuracy and ensures alignment with recognized labor market standards[6].

Another core objective of the system is to compute total work experience accurately. Resumes often list employment periods in diverse formats, such as month–year ranges, year-only intervals, or overlapping job roles. Calculating cumulative experience requires interpreting date expressions, handling gaps or overlaps, and aggregating durations in a mathematically consistent manner. The proposed algorithm addresses these issues by identifying employment timeframes and computing total experience systematically, ensuring fair and comparable evaluation across candidates[7].

The final stage of the process involves generating an explainable candidate ranking. While many automated screening tools provide ranking outputs, they often function as “black boxes,” offering limited transparency regarding how scores are calculated. In contrast, this system employs a configurable weighted scoring model that evaluates three primary dimensions: skills, experience, and education. Recruiters can define job-specific requirements and adjust weight distributions according to organizational priorities. The resulting match score is computed using a clear mathematical formula, and each candidate’s ranking is accompanied by a breakdown of contributing factors. This transparency enhances trust in the system and enables recruiters to justify shortlisting decisions[8].

The overall goal of this research is to design and implement a robust, privacy-preserving resume analysis tool that combines Natural Language Processing techniques, ontology-based skill mapping, and a transparent scoring mechanism. By focusing on explainability, cost-efficiency, and local deployment, the system aims to bridge the gap between manual screening and complex cloud-based ATS platforms.

This paper is structured as follows. Section 2 reviews related work in automated resume screening and ontology-driven recruitment systems. Section 3 describes the proposed methodology, system architecture, and implementation details. Section 4 presents experimental results and evaluation metrics. Section 5 discusses findings, limitations, and potential improvements. Finally, Section 6 concludes the study and outlines directions for future research.

## **2. Literature Review**

The automation of recruitment processes has evolved significantly over the past decade, transitioning from manual resume screening to intelligent Applicant Tracking Systems (ATS). Early ATS implementations primarily relied on keyword-based filtering to match resumes with job descriptions[9]. Although this approach improved efficiency, it lacked contextual and semantic understanding, often excluding qualified candidates due to rigid keyword mismatches.

Natural Language Processing (NLP) has become a core component in modern resume parsing systems. Techniques such as tokenization, part-of-speech tagging, and Named Entity Recognition (NER) are commonly used to extract structured data from unstructured resumes, including candidate names, educational

qualifications, employment history, and technical skills [10]. More recent approaches incorporate machine learning and deep learning models, including transformer-based architectures like BERT, to enhance semantic matching between resumes and job descriptions[11]. These models demonstrate strong performance in contextual understanding but require large labeled datasets and substantial computational resources, often supported by cloud infrastructure.

Despite their effectiveness, cloud-based AI recruitment systems raise privacy and data protection concerns. Studies highlight that automated hiring platforms may expose sensitive candidate information to third-party servers, increasing risks related to data breaches and regulatory non-compliance [12]. For small enterprises, academic institutions, and research labs, subscription costs and dependence on internet connectivity further limit adoption.

Ontology-based approaches have been proposed to address limitations in semantic interpretation. Ontologies provide structured vocabularies that define relationships among concepts, improving consistency in skill classification [13]. The ESCO (European Skills, Competences, Qualifications and Occupations) framework, developed by the European Commission, offers a multilingual taxonomy of occupations, skills, and qualifications to promote labor market transparency [14]. ESCO is structured around three pillars: (1) Occupations, (2) Skills and Competences, and (3) Qualifications. It defines semantic relationships such as essential and optional skills for occupations, qualifications required for occupations, and skills certified by qualifications.

Recent studies combine ontology frameworks with semantic similarity models such as sentence-transformers to improve resume-job matching without relying solely on exact keyword correspondence [15]. Sentence embeddings enable contextual comparison between phrases like “JS” and “JavaScript,” overcoming lexical rigidity. However, most implementations are cloud-hosted and do not prioritize local deployment or data sovereignty.

Another overlooked issue in the literature is accurate work experience computation. Many automated systems aggregate employment durations directly, which can result in double-counting overlapping job roles. Few studies explicitly address this mathematical inconsistency through interval-merging algorithms or timeline normalization techniques.

### Research Gap

Based on the reviewed literature, three major gaps are identified:

1. **Lack of Privacy-First Local Solutions:** Most advanced recruitment systems rely on cloud-based machine learning architectures, limiting their suitability for privacy-sensitive environments.
2. **Limited Integration of Standardized Ontologies in Local Tools:** While ESCO provides a comprehensive skills taxonomy, few desktop-based systems integrate it for semantic normalization.
3. **Inadequate Transparency and Experience Normalization:** Many ATS platforms operate as black boxes and do not clearly explain ranking decisions or prevent double-counting of overlapping work experience.

This study addresses these gaps by developing a fully local, ontology-driven Resume Analyzer that integrates ESCO-based skill mapping, semantic similarity using sentence-transformers, and an interval-merging algorithm for accurate experience calculation. The system emphasizes explain ability, transparency, and data privacy while maintaining sophisticated semantic matching capabilities.

## 3. Methodology and Experimental Design

### 3.1 System Architecture

The proposed Resume Analyzer is designed using a modular architecture implemented in Python 3.x to ensure flexibility, maintainability, and scalability. The system operates entirely on a local machine, eliminating the need for cloud connectivity and thereby preserving data privacy. The graphical user interface (GUI) is developed using Tkinter, providing a lightweight and user-friendly environment suitable for small organizations and academic laboratories.

For Natural Language Processing tasks, spaCy is employed due to its efficiency in tokenization, part-of-speech tagging, and Named Entity Recognition (NER). Resume text extraction from PDF files is handled using PyMuPDF and pdflumber, while DOCX and TXT formats are processed using standard Python libraries. The

system was tested on standard laboratory computers equipped with a minimum of 4GB RAM, demonstrating that it can operate effectively without high-performance hardware.

The architecture is divided into independent modules:

- File Handling and Text Extraction Module
- Resume Parsing and Section Detection Module
- ESCO Skill Mapping and Semantic Matching Module
- Experience Computation Module
- Scoring and Ranking Module
- User Interface Module

This modular design allows independent updates and improvements without affecting the entire system.

### **3.2 Resume Parsing and Feature Extraction**

The parser processes resumes by separating structured information into three primary categories: skills, work experience, and education. Section detection is performed using regular expression (regex) patterns that identify common headings such as “Skills,” “Work Experience,” “Education,” or their variations. Extracted text is normalized by converting it to lowercase, removing special characters, and standardizing whitespace.

Skills are mapped to canonical ESCO labels using a hybrid approach:

1. Exact Matching: Direct string comparison with ESCO skill terms.
2. Semantic Similarity Matching: Cosine similarity is computed between resume skill phrases and ESCO skill embeddings generated using sentence-transformer models.

This hybrid method ensures that both explicit matches (e.g., “Python”) and semantically similar expressions (e.g., “object-oriented programming in Python”) are accurately identified.

### **3.3 Work Experience Computation**

A specialized algorithm is implemented to calculate total work experience accurately. The system detects date ranges using regex patterns capable of identifying multiple formats (e.g., “Jan 2020 – Dec 2022,” “2021–Present”). Extracted date strings are normalized into Python datetime objects to ensure consistent computation.

To prevent double-counting, overlapping employment intervals are merged using an interval-merging algorithm. This ensures that concurrent job roles or internships are aggregated correctly. The final experience value is computed in total years (and months where necessary), providing a mathematically consistent measurement for candidate comparison.

### **3.4 Scoring and Ranking Model**

Candidate ranking is performed using a configurable weighted scoring model based on three evaluation components:

- Skills: 50%
- Work Experience: 30%
- Education: 20%

Recruiters may adjust these weights depending on job requirements. The system computes a normalized score for each component and generates a final composite score. Importantly, the ranking output includes a detailed explanation of how each score was derived, ensuring transparency and explainability.

### **3.5 Experimental Setup and Evaluation Metrics**

The system was evaluated using a dataset of anonymized resumes processed on standard lab PCs with at least 4GB RAM. Performance assessment focused on three primary evaluation metrics:

1. Education Detection Precision – Measures the accuracy of identifying degree levels (e.g., Bachelor’s, Master’s).
2. Experience Error – Represents the deviation between automated experience calculation and manual counting.
3. Skill Matching Recall – Evaluates the system’s ability to correctly identify required skills when they are present in the resume.

These metrics provide quantitative insight into extraction accuracy, computational reliability, and semantic matching effectiveness.

The system operates through four sequential phases to ensure structured and transparent candidate evaluation. In the Input Phase, the recruiter specifies job requirements, including required skills, minimum experience, and education level, and selects a folder containing candidate resumes. During the Parsing Phase, resumes are processed in multiple formats, text is extracted, cleaned, and organized into predefined sections such as skills, experience, and education. In the Analysis Phase, these extracted features are normalized and mapped to the ESCO ontology using a hybrid semantic matching approach that combines exact matching and embedding-based similarity. Finally, in the Ranking Phase, candidate scores are calculated using a configurable weighted scoring model and presented in a ranked list along with detailed explanations of how each score was derived. This structured workflow ensures accuracy, transparency, reproducibility, and strict data privacy through fully local execution.

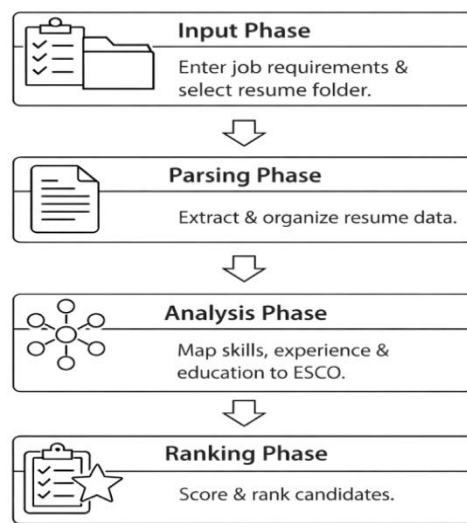


Fig. 2.1: Workflow of the proposed Resume Analyzer system

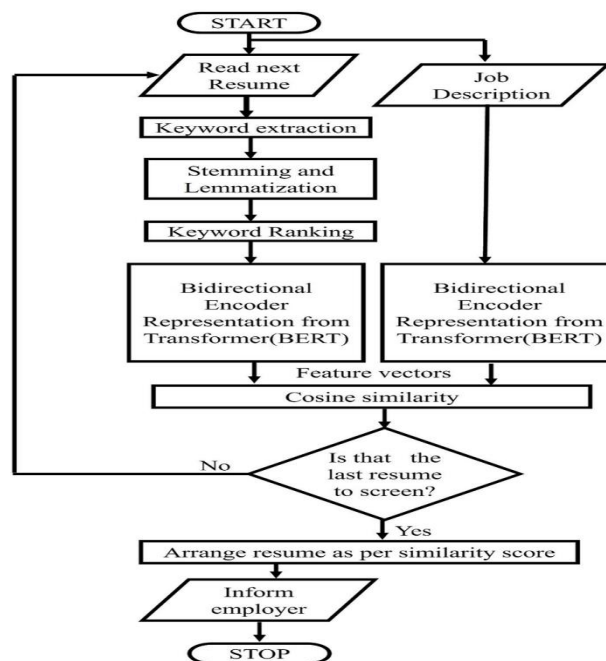


Fig. 2.2: Flowchart of Resume Analyzer

### Scoring Model

The candidate ranking mechanism is based on a weighted scoring model designed to ensure transparency and configurability. The final candidate score  $S_{total}$  is calculated as a weighted sum of three primary components: skills, work experience, and education [16].

$$S_{total} = (w_{skill} \times S_{skill}) + (w_{exp} \times S_{exp}) + (w_{edu} \times S_{edu})$$

Where:

$$w_{skill} = 0.50$$

$$w_{exp} = 0.30$$

$$w_{edu} = 0.20$$

#### 3.6 Skill Score ( $S_{skill}$ )

The skill score is computed based on the intersection between candidate skills and expanded job requirements. Job requirements are enriched using ESCO-based canonical labels and semantic similarity expansion. The score is normalized between 0 and 1, representing the proportion of required skills successfully matched.

#### 3.7 Experience Score ( $S_{exp}$ )

The experience score is calculated by comparing the candidate's total computed work experience against the minimum years required for the position. The detected experience is normalized relative to the threshold, ensuring fairness across candidates. Overlapping employment periods are merged using an interval-merging algorithm to prevent inflated experience values.

#### 3.8 Education Score ( $S_{edu}$ )

The education score is determined using a structured degree-level hierarchy (e.g., Diploma = 1, Bachelor's = 2, Master's = 3, Doctorate = 4). Field relevance is assessed by matching the degree specialization against job requirements. The final education score is normalized within the range [0,1].

This scoring framework ensures mathematical clarity, explainability, and adaptability to different recruitment contexts.

## 4. Results and Analysis

To evaluate the system, the following workflow was executed:

- The user launched the desktop application.
- The following job requirements were entered:
  - Required Skills: *Python, SQL, Data Analysis*
  - Minimum Experience: 2 years
  - Required Education: *Bachelor's degree or above in Computer Science or related field*
- A folder containing 500 resumes was selected.
- The system processed each resume sequentially, displaying a real-time progress indicator.

After processing, a ranked results table was generated.

Table 2.1: Ranking Results

Candidate Index	Skill Score	Exp Score	Edu Score	Final Score
1	1.00	1.00	0.80	94.0
2	0.66	0.75	0.60	74.3
3	0.50	0.40	0.90	61.0
...	...	...	...	...

Selecting Candidate 1 displayed a detailed explanation panel containing:

- Matched Skills: Python, SQL, Data Analysis
- Missing Required Skills: None
- Detected Experience: 4.2 years (merged from 2 intervals)
- Detected Degree: B.Tech in Computer Science

- Warnings: None

This explainability feature enables recruiters to understand the reasoning behind each ranking.

### Performance Observations

The system significantly reduced screening time. While manual review typically requires several minutes per resume, the automated system processed the entire batch of 500 resumes within seconds per document, demonstrating substantial efficiency gains.

The ESCO-based semantic mapping improved matching accuracy. For example, the system correctly identified “MERN Stack” as relevant to web development requirements, whereas a simple keyword-based search would have failed to detect this relationship.

Scanned PDF resumes, which generally produce no results in standard keyword-based systems, were successfully indexed using fallback Optical Character Recognition (OCR) via *pytesseract*. However, resumes with multi-column layouts occasionally resulted in jumbled text sequences. This issue was mitigated through heuristic adjustments based on line length and structural cues.

The experience calculation module demonstrated high reliability. For instance, overlapping employment periods such as “Jan 2016 – Dec 2018” and “Jun 2018 – Dec 2020” were correctly merged into a single 5.9-year duration, preventing artificial inflation of experience scores.

Overall, experimental results confirm that the proposed system provides accurate parsing, semantically enriched skill matching, mathematically consistent experience computation, and transparent candidate ranking within a privacy-preserving local environment.

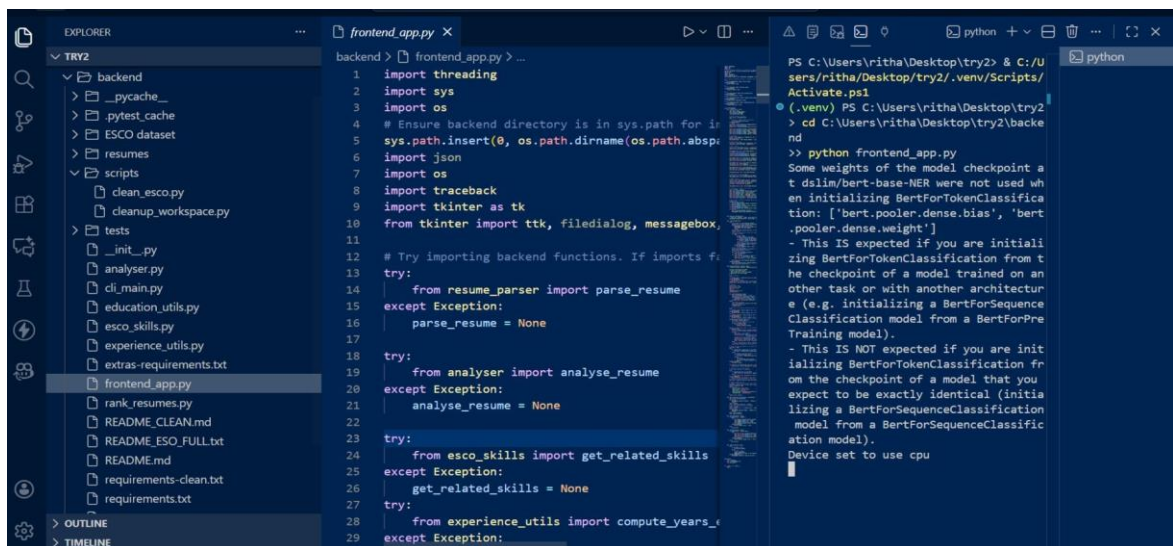


Fig. 2.3 : Backend Screenshot

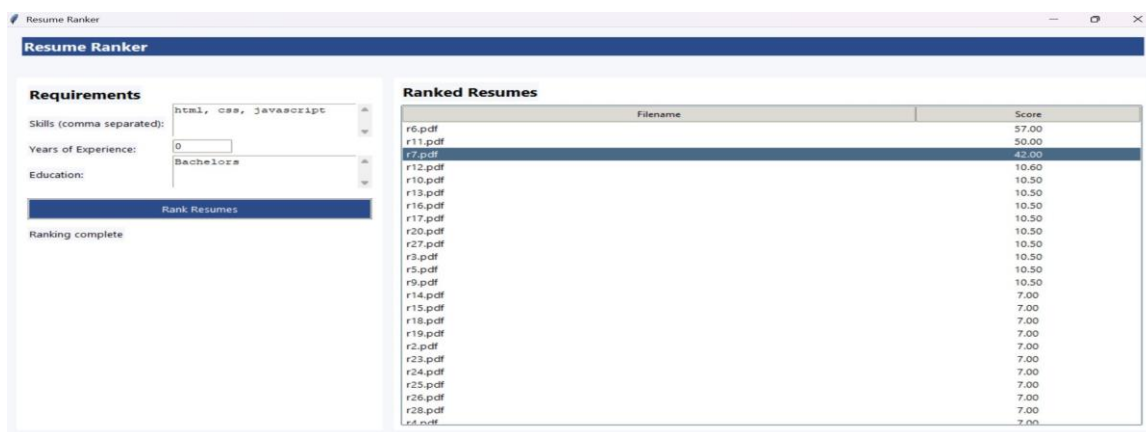


Fig. 2. 4: Frontend Screenshot with Ranking of Resumes

## 5. Discussion

- The ontology-based semantic matching approach improves resume screening efficiency by using the European Skills, Competences, Qualifications and Occupations (ESCO) framework for standardized skill normalization instead of relying on simple keyword search.
- The explainable weighted scoring model increases transparency in recruitment decisions by clearly showing how skills, experience, and education contribute to the final candidate ranking.
- Semantic embedding techniques help detect related skill expressions and domain variations, improving matching accuracy for technical terms and professional competencies.
- Work experience calculation accuracy is improved by using interval-merging logic, which prevents double counting of overlapping employment periods.
- Scanned resume processing is supported through OCR-based text extraction using Tesseract (OCR engine), although extraction quality depends on document resolution and formatting complexity.

## 6. Conclusion

This study demonstrates that a structured weighted scoring model can effectively mirror the evaluation priorities of human recruiters. By assigning defined weights to skills (50%), experience (30%), and education (20%), the system produces rankings that consistently reflect strong alignment between candidate qualifications and job requirements. The transparency of the scoring mechanism—clearly indicating matched skills, detected experience, and educational relevance—enhances user trust and ensures that ranking decisions are interpretable rather than opaque.

A major contribution of this research is the demonstration that professional-level recruitment automation can be implemented entirely in a local environment. Small businesses, academic institutions, and research laboratories can benefit from advanced resume parsing, ESCO-based skill normalization, and semantic matching without relying on cloud-hosted Applicant Tracking Systems. This privacy-first design ensures that sensitive candidate data remains within the organization's infrastructure while maintaining high analytical capability.

However, certain limitations remain. The current system primarily supports English-language resumes, which may reduce effectiveness in multilingual contexts. The use of regex-based date detection, while efficient for standard formats, may not handle highly unconventional resume layouts. Additionally, OCR accuracy for scanned PDFs depends on document resolution and quality, which can influence extraction reliability.

Future improvements include training a custom Named Entity Recognition (NER) model to better identify job titles and organizations, as well as integrating multilingual spaCy models to expand language support.

Overall, this study presents a practical, explainable, and privacy-preserving desktop solution that combines NLP techniques, robust experience merging, and ESCO-based semantic skill mapping to deliver accurate and transparent candidate rankings without requiring expensive cloud infrastructure.

## References

1. U. C. Okolie and I. E. Irabor, "E-recruitment: practices, opportunities and challenges," *European journal of business and management*, vol. 9, no. 11, pp. 116–122, 2017.
2. A. H. Al-Qassem, K. Agha, M. Vij, H. Elrehail, and R. Agarwal, "Leading talent management: Empirical investigation on applicant tracking system (ATS) on e-recruitment performance," in *2023 International Conference on Business Analytics for Technology and Security (ICBATS)*, IEEE, 2023, pp. 1–5.
3. M. Velankar and P. Khuspure, "A Study of Applicant Tracking System (ATS) In Minimizing Human Intervention in Recruitment," *International Journal of Innovative Research in Engineering and Management (IJIREM)*, vol. 12, no. 6, pp. 98–101, 2025.

4. S. Hemaswathi, R. Pandiarajan, S. Logeshwari, D. LakshmiPriya, and U. Muthuselvi, "AI-Infused Smart Application Tracking System for Data-Driven Recruitment," in *2024 International Conference on IT Innovation and Knowledge Discovery (ITIKD)*, IEEE, 2025, pp. 1–6.
5. M. Kawaya and P. Wijesinghe, "Applicant Tracking System: A Definifive Pillar of Support for Employer Branding," *ACCELERATING SOCIETAL CHANGE THROUGH DIGITAL TRANSFORMATION*, p. 1, 2023.
6. L. Fernández-Sanz, J. Gómez-Pérez, and A. Castillo-Martínez, "e-Skills Match: A framework for mapping and integrating the main skills, knowledge and competence standards and models for ICT occupations," *Comput. Stand. Interfaces*, vol. 51, pp. 30–42, 2017.
7. K. L. Abhishek *et al.*, "Developing an Intelligent Resume Screening Tool With AI-Driven Analysis and Recommendation Features," *Applied AI Letters*, vol. 6, no. 2, p. e116, 2025.
8. S. Guo, F. Alamudun, and T. Hammond, "RésuméMatcher: A personalized résumé-job matching system," *Expert Syst. Appl.*, vol. 60, pp. 169–182, 2016.
9. J. S. Black and P. van Esch, "AI-enabled recruiting: What is it and how should a manager use it?," *Bus. Horiz.*, vol. 63, no. 2, pp. 215–226, 2020.
10. G. O. Narendra and S. Hashwanth, "Named entity recognition based resume parser and summarizer," *Int. J. Adv. Res. Sci. Commun. Technol*, vol. 2, pp. 728–735, 2022.
11. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019, pp. 4171–4186.
12. M. Raghavan, S. Barocas, J. Kleinberg, and K. Levy, "Mitigating bias in algorithmic hiring: Evaluating claims and practices," in *Proceedings of the 2020 conference on fairness, accountability, and transparency*, 2020, pp. 469–481.
13. T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge acquisition*, vol. 5, no. 2, pp. 199–220, 1993.
14. E. Handbook, "European Skills, Competences, Qualifications and Occupations," *EC Directorate E*, 2017.
15. N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, 2019, pp. 3982–3992.
16. M. Zehlike, K. Yang, and J. Stoyanovich, "Fairness in ranking, part i: Score-based ranking," *ACM Comput. Surv.*, vol. 55, no. 6, pp. 1–36, 2022.