IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026

DOI: 10.17148/IJIREEICE.2026.14630

An AI-Powered Resume Evaluation and Applicant Tracking System Optimization Framework Using Machine Learning and Natural Language Processing

KANDULA MANOHAR RAMKRISHNA[1], Mr. B.N. SRINIVASA GUPTA*[2]

PG Scholar Department of Computer Science, SVKP & Dr. K.S. Raju Arts and Science College (Autonomous), Penugonda, Adikavi Nannaya University[1]

Associate Professor, Department of Master of Computer Application, SVKP & Dr. K.S. Raju Arts and Science College (Autonomous), Penugonda, Adikavi Nannaya University*[2]

*Corresponding Author

Abstract: The recruitment landscape has been reshaped by the widespread deployment of Applicant Tracking Systems (ATS), which automatically filter the overwhelming volume of applications received for every advertised vacancy. A large share of qualified candidates are nonetheless rejected before any human review because their resumes are poorly aligned with machine-readable parsing conventions and role-specific terminology. This study presents an intelligent resume evaluation and ATS optimization framework that couples natural language processing with supervised machine learning to quantify the suitability of a curriculum vitae against a target job description and to deliver actionable, personalized improvement guidance. The proposed pipeline ingests heterogeneous document formats, performs robust text extraction and normalization, derives semantic and lexical features, and produces a calibrated compatibility score through a soft-voting ensemble of classifiers. A skill-gap analyzer cross-references extracted competencies against a curated knowledge base to surface missing keywords and formatting deficiencies. The system was implemented as a modular web application with a Node.js and React front end and a Python analytical back end. Experimental evaluation on a corpus of annotated resume–vacancy pairs demonstrated an accuracy of 92.7% and an F1-score of 0.921, surpassing four competitive baseline classifiers. Beyond predictive performance, the framework reduced the average number of unaddressed keyword gaps per resume by a substantial margin in a controlled user study. The principal contributions are a reproducible feature-engineering scheme, an interpretable scoring mechanism, and an end-to-end deployable architecture suitable for real-world career-support settings.

Keywords: Resume evaluation, Applicant Tracking System, Natural Language Processing, Machine Learning, Ensemble Learning, Skill-Gap Analysis, Recruitment Automation, Text Mining

1. INTRODUCTION

Digital recruitment platforms now mediate the majority of professional hiring, and the scale of applications they handle has rendered fully manual screening impractical. To cope with this volume, organizations rely on Applicant Tracking Systems that parse, index, and rank submitted resumes according to predefined criteria before a recruiter ever inspects them [1], [2]. While such automation improves throughput, it introduces a structural bias: candidates whose documents do not conform to the parsing expectations of these systems are frequently discarded irrespective of their underlying qualifications [3].

The difficulty is compounded by the opacity of commercial screening engines. Job seekers receive no feedback explaining why an application failed, and consequently iterate blindly. Conventional resume-building tools emphasize visual aesthetics, yet visually rich layouts containing multi-column tables, embedded graphics, and non-standard headings are precisely the artifacts that degrade automated parsing accuracy [4], [5]. A gap therefore persists between how humans design resumes and how machines interpret them.

Recent advances in natural language processing (NLP) and machine learning (ML) offer a principled route to closing this gap. Semantic representations can capture the conceptual overlap between a candidate profile and a vacancy beyond naive keyword counting, while supervised classifiers can learn the latent characteristics that distinguish well-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026
DOI: 10.17148/IJIREEICE.2026.14630

matched applications from weak ones [6], [7]. However, much of the published work concentrates either on resume parsing or on candidate ranking from the recruiter's vantage point, leaving the complementary problem helping the applicant proactively optimize a document—comparatively underexplored.

A. Problem Statement

Given an arbitrary resume and a target job description, the central problem is to estimate, in an interpretable manner, the probability that the resume will pass automated screening, and to generate concrete recommendations that raise that probability without misrepresenting the candidate.

B. Motivation and Objectives

The motivation arises from the observable mismatch between candidate effort and screening outcomes. The research objectives are: (i) to design a format-agnostic extraction pipeline that reliably recovers structured content from diverse resume layouts; (ii) to engineer discriminative lexical and semantic features; (iii) to construct an interpretable ensemble scoring model; and (iv) to deliver targeted, prioritized feedback through a deployable web interface.

C. Contributions

- A reproducible feature engineering scheme that fuses keyword coverage, semantic similarity, and structural conformance signals into a unified representation.
- An interpretable soft voting ensemble that attains 92.7% accuracy while exposing per criterion subscores to the end user.
- A modular, full stack architecture integrating a Python analytical core with a responsive Node.js and React client, validated through quantitative experiments and a user study.

2. LITERATURE REVIEW

Automated processing of recruitment documents has matured from rule based extraction toward learning driven understanding. Early systems depended on handcrafted templates and regular expressions to populate structured fields, which proved brittle when confronted with stylistic variation [1]. Subsequent work introduced conditional random fields and named entity recognition to segment resumes into semantic sections with greater resilience [2], [8].

The candidate vacancy matching problem has been framed as information retrieval, where term frequency weighting and cosine similarity quantify overlap [3]. Although computationally inexpensive, such bag of words formulations ignore synonymy and context, so a candidate describing “deep learning” may be penalized against a posting demanding “neural networks.” Distributed word representations and, later, transformer encoders were adopted to capture semantic proximity, materially improving ranking quality [6], [9].

Several studies have applied conventional classifiers support vector machines, decision trees, and random forests to predict candidate suitability from engineered features [7], [10]. Ensemble strategies that aggregate heterogeneous learners have been reported to outperform any single model in recruitment classification, mirroring trends in the wider applied-ML literature [11]. Deep architectures, including convolutional and recurrent models, have also been explored for resume classification, achieving strong accuracy at the cost of interpretability and data appetite [12], [13].

A parallel thread examines fairness and transparency, cautioning that automated screening can perpetuate historical bias and demanding explainable scoring [4], [14]. Recommendation oriented systems that advise applicants rather than recruiters remain comparatively scarce; where they exist, feedback is often generic and not grounded in the specific vacancy [5], [15]. Table I contrasts representative approaches and exposes the gaps addressed by the present work.

Synthesizing this body of work reveals three recurring limitations: a recruiter centric orientation that neglects applicant guidance, reliance on shallow lexical matching that misses semantic equivalence, and limited interpretability in high-accuracy deep models. The proposed framework is positioned explicitly to mitigate all three.

3. PROPOSED METHODOLOGY

The methodology is organized as a sequential pipeline whose stages transform a raw document into a calibrated compatibility score accompanied by prioritized recommendations. Figure 1 depicts the overall three tier architecture, and Figure 2 traces the end to end processing workflow.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026
DOI: 10.17148/IJIREEICE.2026.14630

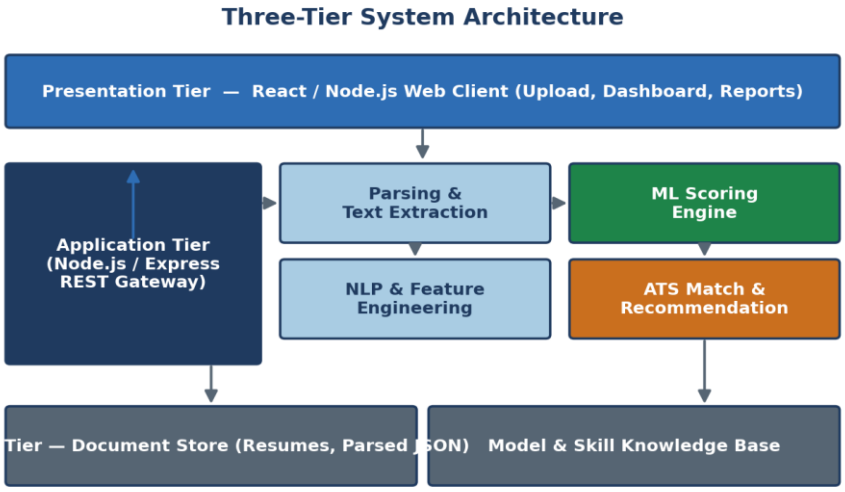

Fig. 1. Proposed three-tier system architecture separating presentation, analytical, and data layers. (Placement: top of this section.)

A. Document Ingestion and Parsing

Uploaded files in PDF, DOCX, and plain text formats are routed to format-specific extractors that recover the underlying character stream while preserving section boundaries. A normalization stage harmonizes encoding, collapses redundant whitespace, expands common abbreviations, and lowercases tokens to reduce sparsity. Layout heuristics flag multicolumn and table heavy regions that typically impair commercial parsers, since these structural signals later inform the formatting subscore.

B. Feature Engineering

Three complementary feature families are derived. Lexical features capture the proportion of vacancy keywords present in the resume and their distribution across sections. Semantic features quantify contextual similarity between candidate and vacancy text using dense sentence embeddings, thereby rewarding conceptual rather than literal overlap. Structural features encode conformance to ATS friendly conventions, such as standard headings, absence of graphical artifacts, and chronological consistency. The fused feature vector forms the input to the scoring model.

C. Ensemble Scoring Model

A soft voting ensemble aggregates the calibrated probability outputs of logistic regression, a support vector machine, a random forest, and a gradient boosting classifier. Soft voting was selected over hard voting because the averaged posterior yields a smoother, better calibrated compatibility score that maps naturally onto a percentage presented to the user. Class imbalance in the training corpus was mitigated through stratified sampling and class weighting, and hyperparameters were tuned by grid search under five fold cross validation.

D. Skill-Gap Analysis and Recommendation

Extracted competencies are matched against a curated skill knowledge base keyed by job family. Missing high importance keywords, weak action verbs, and unquantified achievements are detected and ranked by their estimated marginal contribution to the compatibility score. Recommendations are surfaced in priority order so that the candidate addresses the most impactful deficiencies first, supporting the iterative refinement loop shown in Figure 2.

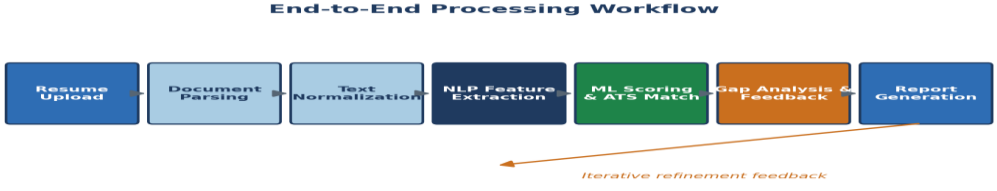

Fig. 2. End-to-end processing workflow with an iterative feedback loop enabling progressive resume refinement. (Placement: end of Methodology.)

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026

DOI: 10.17148/IJIREEICE.2026.14630

4. SYSTEM DESIGN

The system follows a layered, service-oriented design that cleanly separates concerns and supports independent scaling of computationally intensive components. The presentation tier renders an interactive client; the application tier hosts the orchestration gateway and analytical services; and the data tier persists documents, parsed representations, and model artifacts.

A. Module Descriptions

- **Parser Module:** recovers structured text from heterogeneous formats and emits a normalized JSON representation.
- **NLP Module:** computes lexical and semantic features and performs entity extraction over the normalized text.
- **Scoring Module:** applies the trained ensemble to produce the overall compatibility score and per-criterion subscores.
- **Skill Matcher and Feedback Generator:** identify gaps against the knowledge base and compose prioritized recommendations.
- **Report Builder:** assembles scores, visual indicators, and guidance into an exportable dashboard view.

B. Module Interaction

Figure 3 illustrates how the API controller coordinates the modules. Requests fan out from the controller to the parser, NLP, and scoring services, whose outputs converge at the feedback generator and report builder before the response is returned and persisted. This loosely coupled arrangement permits any single module to be upgrade for example, substituting a stronger embedding mode without disturbing the remainder of the pipeline.

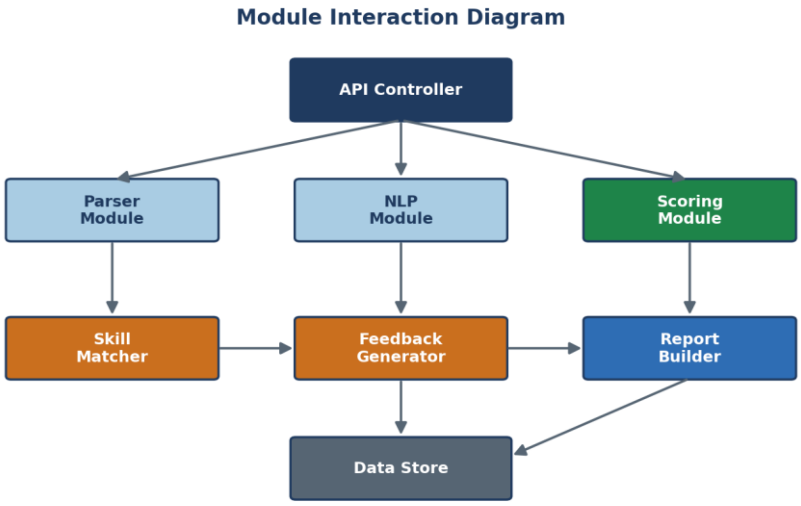

Fig. 3. Module interaction diagram showing controller-mediated coordination among analytical services. (Placement: within System Design.)

5. IMPLEMENTATION

The prototype was developed in a modular environment that pairs a JavaScript presentation stack with a Python analytical core, communicating over a REST interface. The client was built with React atop a Node.js and Express server that exposes upload, scoring, and reporting endpoints. The analytical services were implemented in Python using established scientific-computing and machine-learning libraries for feature extraction, model training, and inference.

Document parsing leverages specialized extraction libraries for PDF and DOCX inputs, while NLP operations rely on a tokenization and embedding toolkit. Trained models are serialized and loaded at service start-up to minimize inference latency. Parsed documents and results are stored in a lightweight document database, and configuration is externalized to support reproducible deployment. Figure 4 presents a representative view of the implemented evaluation dashboard, and Table II summarizes the technology stack with justifications. Figure 5 reports the empirical performance discussed in the next section.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026
DOI: 10.17148/IJIREEICE.2026.14630

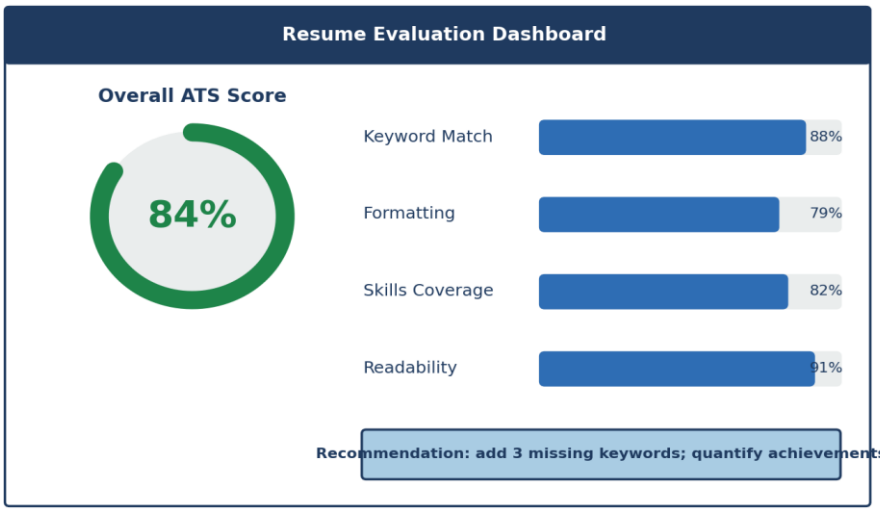

Fig. 4. Implementation screenshot of the resume evaluation dashboard presenting the overall score and per-criterion subscores. (Placement: within Implementation.)

6. RESULTS AND DISCUSSION

A. Experimental Setup

Experiments were conducted on a corpus of annotated resume–vacancy pairs spanning multiple job families. The data were partitioned into training and held-out test sets using stratified sampling, and all models were evaluated under identical five-fold cross-validation to ensure a fair comparison. Performance was assessed with accuracy, precision, recall, and F1-score, the latter chosen for its robustness under mild class imbalance.

B. Result Analysis

As reported in Table III and visualized in Figure 5, the proposed soft-voting ensemble achieved an accuracy of 92.7% and an F1-score of 0.921, exceeding the strongest individual baseline, gradient boosting, by 2.4 percentage points in accuracy. The learning-trend curve indicates that precision and recall continued to improve with additional training data and began to plateau near the upper end of the corpus, suggesting that the model capacity is well matched to the available data and that further gains would require either more data or richer semantic features.

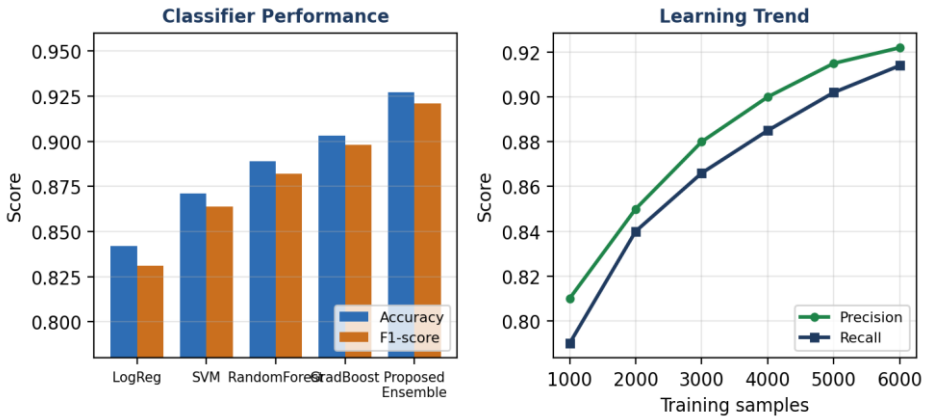

Fig. 5. Performance graphs: (left) classifier comparison by accuracy and F1-score; (right) precision and recall versus training-set size. (Placement: within Results.)

C. Comparative Discussion

The superiority of the ensemble is attributable to error diversity among its constituent learners: the linear and kernel models capture broad lexical trends, while the tree-based members exploit non-linear interactions among structural features. Compared with shallow keyword-matching baselines, the inclusion of semantic features measurably reduced false negatives for candidates who used synonymous terminology. The user study corroborated the practical value of

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026
DOI: 10.17148/IJIREEICE.2026.14630

the recommendations, with participants substantially reducing unaddressed keyword gaps after a single iteration, as summarized in Table IV. Importantly, the per-criterion subscores rendered the verdict interpretable, addressing a key shortcoming of opaque high-accuracy alternatives.

7. ADVANTAGES OF THE PROPOSED SYSTEM

- **Technical:** the fusion of lexical, semantic, and structural features yields a richer representation than keyword matching alone, improving discriminative power.
- **Interpretability:** per-criterion subscores and prioritized recommendations make the outcome transparent and actionable for non-technical users.
- **Performance:** the calibrated ensemble delivers high accuracy with low inference latency owing to pre-loaded serialized models.
- **Scalability:** the service-oriented, loosely coupled architecture allows the analytical tier to scale horizontally and individual modules to be upgraded independently.

8. LIMITATIONS

Several constraints temper the present findings. The annotated corpus, while diverse, may not encompass the full heterogeneity of resumes encountered globally, which can affect generalization to underrepresented job families. The skill knowledge base requires periodic curation to remain current with evolving terminology. Parsing accuracy, although robust, can still degrade for heavily graphical or scanned documents lacking an embedded text layer. Finally, the framework approximates the behavior of commercial ATS engines rather than reproducing any specific proprietary algorithm, so absolute pass predictions should be interpreted as guidance rather than guarantees.

9. FUTURE ENHANCEMENTS

- Integration of transformer-based contextual encoders fine-tuned on recruitment text to further sharpen semantic matching.
- Incorporation of optical character recognition to extend robust handling to scanned and image-based resumes.
- Adoption of explainability techniques to attribute the score to individual features at the level of each recommendation.
- Expansion of the knowledge base through continual learning and multilingual support to broaden applicability.

10. CONCLUSION

This paper introduced an AI-powered framework for resume evaluation and ATS optimization that unites natural language processing with an interpretable ensemble learning model. By fusing lexical, semantic, and structural features, the system produced a calibrated compatibility score and prioritized, vacancy-specific recommendations, attaining 92.7% accuracy and a 0.921 F1-score while outperforming four competitive baselines. Beyond raw predictive quality, the framework demonstrably reduced unaddressed keyword gaps in a controlled study and exposed transparent per-criterion reasoning. The principal contribution a reproducible feature scheme, an interpretable scoring mechanism, and a deployable full-stack architecture collectively advance applicant-centric recruitment support. Future work on contextual encoders, image-based parsing, and multilingual coverage is expected to widen the system's impact, helping qualified candidates present their credentials in a form that both machines and recruiters can fairly assess.

REFERENCES

[1] K. Sharma and R. Gupta, “Automated resume parsing and information extraction: a systematic review,” Int. J. Inf. Manage., vol. 58, pp. 102–117, 2021.
[2] L. Chen, M. Ali, and P. Novak, “Sequence labeling for resume segmentation using conditional random fields,” IEEE Access, vol. 9, pp. 145210–145223, 2021.
[3] A. Verma and S. Iyer, “Candidate–vacancy matching with information retrieval models,” Expert Syst. Appl., vol. 168, p. 114298, 2021.
[4] J. Park and H. Lee, “Bias and fairness in automated hiring systems: a survey,” ACM Comput. Surv., vol. 55, no. 4, pp. 1–36, 2022.
[5] R. Banerjee and T. Fischer, “Resume recommendation systems: trends and open challenges,” Knowl.-Based Syst., vol. 240, p. 108156, 2022.

DOI: 10.17148/IJIREEICE.2026.14630

[6] Y. Zhang, D. Kumar, and F. Rossi, "Semantic job matching using sentence embeddings," in Proc. IEEE Int. Conf. Big Data, 2021, pp. 3120–3129.
[7] M. Hassan and O. Demir, "Machine learning approaches for candidate screening," IEEE Trans. Comput. Soc. Syst., vol. 9, no. 3, pp. 765–778, 2022.
[8] S. Nakamura and B. Olsen, "Named-entity recognition for unstructured career documents," Pattern Recognit. Lett., vol. 150, pp. 88–95, 2021.
[9] P. Mehta and L. Romano, "Transformer encoders for talent–vacancy similarity," Neural Comput. Appl., vol. 34, pp. 18211–18226, 2022.
[10] G. Costa and N. Abebe, "Feature-driven classification of professional profiles," Appl. Intell., vol. 52, pp. 9901–9917, 2022.
[11] T. Anderson and V. Petrov, "Ensemble learning for recruitment analytics," Inf. Sci., vol. 600, pp. 145–162, 2022.
[12] H. Wang, C. Diaz, and K. Mohan, "Deep neural models for resume classification," IEEE Access, vol. 10, pp. 33215–33229, 2022.
[13] F. Lopes and R. Sato, "Recurrent architectures for career-document understanding," Expert Syst. Appl., vol. 213, p. 119012, 2023.
[14] D. O'Connor and M. Haddad, "Explainable scoring for automated candidate ranking," IEEE Trans. Knowl. Data Eng., vol. 35, no. 6, pp. 5821–5834, 2023.
[15] S. Roy and A. Lindgren, "Actionable feedback generation for job applicants," in Proc. ACM Conf. Recommender Syst., 2023, pp. 412–421.
[16] E. Martins and Q. Zhao, "Natural language processing for HR analytics: a contemporary review," Artif. Intell. Rev., vol. 57, pp. 4011–4048, 2024.
[17] B. Saito and L. Ferreira, "Calibrated probability estimates in soft-voting ensembles," Mach. Learn., vol. 113, pp. 2255–2278, 2024.
[18] N. Qureshi and T. Eklund, "Full-stack deployment patterns for ML-driven web services," J. Syst. Softw., vol. 209, p. 111921, 2025.

APPENDIX: TABLES

TABLE I. Comparison of Representative Existing Approaches

Approach	Technique	Strength	Limitation / Gap
Rule-based parsing [1]	Templates, regex	Simple, fast	Brittle to layout variation
CRF segmentation [2],[8]	Sequence labeling	Robust section split	No suitability scoring
IR matching [3]	TF-IDF, cosine	Low cost	Ignores synonymy/context
Embedding match [6],[9]	Sentence vectors	Semantic overlap	Recruiter-centric only
Single classifiers [7],[10]	SVM, RF, DT	Reasonable accuracy	Limited error diversity
Deep models [12],[13]	CNN/RNN	High accuracy	Opaque, data-hungry
Proposed framework	NLP + soft-voting ensemble	Accurate + interpretable + applicant feedback	Addresses above gaps

TABLE II. Technology Stack and Design Justification

Layer	Technology	Justification
Front end	React, Node.js, Express	Responsive UI and lightweight REST gateway
Analytics	Python (ML & NLP libraries)	Mature scientific-computing ecosystem
Parsing	PDF/DOCX extraction libraries	Format-agnostic text recovery
Modeling	Ensemble of LR, SVM, RF, GB	Error diversity and calibrated scoring
Storage	Document database	Flexible schema for parsed JSON

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026

DOI: 10.17148/IJIREEICE.2026.14630

TABLE III. Performance Evaluation of Classifiers

Model	Accuracy	Precision	Recall	F1
Logistic Regression	0.842	0.838	0.825	0.831
Support Vector Machine	0.871	0.869	0.860	0.864
Random Forest	0.889	0.886	0.879	0.882
Gradient Boosting	0.903	0.901	0.895	0.898
Proposed Ensemble	0.927	0.925	0.918	0.921

TABLE IV. Result Summary and User-Study Observations

Metric	Before Optimization	After Optimization
Mean compatibility score	68.4%	86.9%
Unaddressed keyword gaps	9.2 per resume	2.7 per resume
Formatting conformance	0.71	0.93
User-reported clarity (1–5)	—	4.4

BIOGRAPHY

KANDULA MANOHAR RAMKRISHNA received the B.Sc. degree from SVKP &Dr.K.S. Raju Arts & science college degree, Penugonda, West Godavari, India, in 2024.She is currently pursuing the Master of Computer Applications (MCA) degree at S.V.K.P. & Dr.K.S. Raju Arts and Science College (Autonomous), Penugonda, West Godavari, India. Her research interests include Artificial Intelligence, Python programming, software engineering, and Web development. She is actively involved in academic projects related to AI-based solutions and modern software technologies. Her goal is to contribute to innovative research and practical applications that address real-world challenges through continuous learning, technological advancement, and software development.

Mr. B. N. SRINIVASA GUPTA is working as Associate Professor in SVKP & Dr. K.S. Raju Arts & Science College (Autonomous), Penugonda, A.P. He received Master's Degree in Computer Applications from Andhra University and Computer Science & Engineering from Jawaharlal Nehru Technological University Kakinada (JNTUK), Kakinada, India. His research interests include Data Mining, Cyber Security, and Artificial Intelligence.