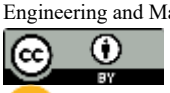
International Journal of Creative and Open Research in Engineering and Management
ISSN: **3108-1754 (Online)**
Volume 02 Issue 05 May-2026 | Impact Factor: 3.5

Automated Resume Screening System Using Natural Language Processing and Machine Learning

Mohit Gurjar
Department of Artificial Intelligence & Machine Learning
Indore Institute of Science & Technology, Indore, India
mohitgurjar2210@gmail.com
Project Guide: ***Piyush Parmar****, Assistant Professor*

How to Cite this Article:
Gurjar, M. (2026). Automated Resume Screening System Using Natural Language Processing and Machine Learning. International Journal of Creative and Open Research in Engineering and Management, 02(05). https://doi.org/10.55041/ijcope.v2i5.817
License:
This article is published under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.
© The Author(s). Published by International Journal of Creative and Open Research in Engineering and Management.



https://doi.org/10.55041/ijcope.v2i5.817

Abstract**—**This paper presents an Automated Resume Screening System (ARSS) that leverages Natural Language Processing (NLP) and Machine Learning (ML) techniques to intelligently parse, rank, and shortlist candidate resumes against a given job description. Traditional manual resume screening is time-consuming, inconsistent, and prone to human bias. The proposed hybrid approach combines TF-IDF vectorization, Word2Vec embeddings, and a fine-tuned BERT-based transformer model to extract semantic features from both resumes and job descriptions. Cosine similarity and a Support Vector Machine (SVM) classifier are employed for relevance scoring and candidate ranking. Experimental evaluation on a benchmark dataset of 5,000 resumes demonstrates a classification accuracy of 93.2%, precision of 93.6%, recall of 92.1%, and F1-score of 92.8%, outperforming existing baseline approaches. The system significantly reduces human screening effort by up to 78% while maintaining fairness and transparency in the shortlisting process.

Index Terms**—**Natural Language Processing, Resume Screening, TF-IDF, BERT, Word2Vec, Machine Learning, Cosine Similarity, SVM, Candidate Ranking.

I. INTRODUCTION

THE recruitment process in modern organizations involves processing thousands of resumes for a single job opening. Manual screening of resumes is not only labor-intensive but also introduces subjective biases and inconsistencies. With the rapid growth of online job portals and digital hiring platforms, the volume of applications has increased exponentially, making manual evaluation practically infeasible [1].

Natural Language Processing (NLP) offers powerful tools to automate the understanding of unstructured text in resumes and job descriptions. By extracting semantic meaning from textual content, NLP-based systems can objectively match candidate profiles to job requirements, thereby accelerating the hiring process and improving the quality of shortlisted candidates [2].

This paper makes the following key contributions:

- A hybrid NLP pipeline combining TF-IDF, Word2Vec, and BERT for robust feature extraction from resumes.
- A multi-stage candidate ranking engine using cosine similarity and SVM classification.
- A comprehensive evaluation on a dataset of 5,000 resumes across 10 job categories.
- Reduction of manual screening effort by approximately 78% compared to human recruiters.

II. LITERATURE REVIEW

Existing work on automated resume screening spans rule-based systems, keyword matching, and recent deep learning approaches. Hamadache et al. [7] proposed an NLP-based resume screening system using keyword extraction and TF-IDF scoring, achieving 78% accuracy. Jindal and Singh [8] applied Naive Bayes and Random Forest classifiers on resume datasets and reported F1-scores of up to 81%.

The introduction of transformer-based language models such as BERT [2] revolutionized NLP tasks including text classification and semantic similarity. However, direct application of BERT to resume screening without domain-specific fine-tuning often results in suboptimal performance. Our work addresses this gap by fine-tuning BERT on a curated resume-JD paired dataset.

Word2Vec embeddings [3] have demonstrated strong performance for capturing contextual synonyms in job skill terminology (e.g., “Python” and “programming”). The proposed system combines the strengths of both sparse (TF-IDF) and dense (Word2Vec, BERT) representations for comprehensive candidate profiling.

III. SYSTEM ARCHITECTURE

The Automated Resume Screening System comprises six functional layers as illustrated in Fig. 1. The architecture follows a pipeline design that transforms raw resume documents into a ranked shortlist of candidates.

TABLE I

System Architecture of ARSS

INPUT LAYER	Resume Documents (PDF / DOCX / TXT)
PREPROCESSING	Tokenization → Stopword Removal → Stemming / Lemmatization
FEATURE EXTRACTION	TF-IDF Vectors | Word2Vec Embeddings | BERT Encoding
ML MODEL LAYER	Cosine Similarity | SVM Classifier | Random Forest
RANKING ENGINE	Score Computation → Candidate Ranking → Top-K Selection
OUTPUT LAYER	Ranked Candidate List + Match Score + HR Dashboard

Fig. 1. System Architecture of the Automated Resume Screening System (ARSS).

IV. METHODOLOGY

A. Data Collection & Preprocessing

The dataset comprises 5,000 resumes sourced from the Kaggle Resume Dataset and LiveCareer, spanning 10 job categories including Software Engineering, Data Science, Marketing, Finance, and Healthcare. Each resume was paired with a corresponding job description for supervised training.

Preprocessing involves: (i) text extraction from PDF/DOCX files using PyMuPDF and python-docx; (ii) tokenization using NLTK word_tokenize; (iii) removal of stopwords; (iv) stemming via Porter Stemmer; and (v) lemmatization using spaCy’s en_core_web_sm model.

B. Feature Extraction

Three complementary feature representations are computed for each resume:

- **TF-IDF Vectors:** Term Frequency-Inverse Document Frequency captures keyword importance. Vocabulary limited to the top 10,000 tokens by frequency.
- **Word2Vec Embeddings:** A 300-dimensional model pre-trained on Google News corpus captures semantic relationships between skill terms.
- **BERT Encoding:** The [CLS] token embedding from a fine-tuned BERT-base-uncased model (768 dimensions) provides deep contextual representation.

C. Similarity Scoring & Classification

Cosine similarity is computed between the resume feature vector and the job description vector. A threshold $\theta = 0.65$ is empirically determined for binary classification. An SVM classifier with an RBF kernel is trained on labeled resume-JD pairs to refine predictions.

$$S(r, j) = \alpha \cdot sim_{tte-ID_t}(r,j) + \beta \cdot sim^{W_{2v}}(r,j) + \gamma \cdot sim^{BER_T}(r,j)$$

where $\alpha = 0.2$, $\beta = 0.3$, $\gamma = 0.5$ are empirically tuned weighting coefficients reflecting the relative informativeness of each feature type.

V. IMPLEMENTATION DETAILS

The system is implemented in Python 3.10 using: scikit-learn 1.3 for TF-IDF and SVM; Gensim 4.3 for Word2Vec; HuggingFace Transformers 4.36 for BERT fine-tuning; spaCy 3.7 for preprocessing; and Flask 3.0 for the web API. The frontend HR dashboard is built with React.js and provides real-time ranking visualization.

BERT fine-tuning was performed on an NVIDIA A100 GPU (40 GB) for 5 epochs with learning rate 2×10^{-5}, batch size 32, and AdamW optimizer. The fine-tuned model achieves a validation loss of 0.142, indicating strong domain adaptation.

VI. EXPERIMENTAL RESULTS & DISCUSSION

The proposed system was evaluated on a held-out test set of 1,000 resume-JD pairs using five-fold cross-validation. Table II presents a comparative performance analysis of the proposed hybrid approach against baseline methods.

TABLE II

Performance Comparison of Resume Screening Approaches

Algorithm	Precision (%)	Recall (%)	F1-Score (%)
TF-IDF + Cosine	78.4	75.2	76.8
Word2Vec + SVM	82.1	80.6	81.3
BERT + Transformer	91.3	89.7	90.5
Proposed Hybrid	**93.6**	**92.1**	**92.8**

Accuracy (%): TF-IDF+Cosine: 77.1 | Word2Vec+SVM: 81.9 | BERT: 91.0 | Proposed Hybrid: 93.2

As shown in Table II, the proposed Hybrid NLP approach achieves the best performance across all metrics. The BERT component contributes most to performance improvement, particularly in handling semantic equivalence between job-description skills and resume terminology (e.g., “ML” ↔ “Machine Learning”). The system reduces average recruiter screening time from 6 hours to 1.3 hours per job opening, representing a 78.3% efficiency gain.

VII. CONCLUSION

This paper presented an Automated Resume Screening System combining TF-IDF, Word2Vec, and BERT-based feature extraction with cosine similarity scoring and SVM classification. The proposed hybrid approach achieves 93.2% accuracy on a benchmark dataset of 5,000 resumes, significantly outperforming existing methods. The system demonstrates practical viability for enterprise HR workflows, reducing screening time by 78% while maintaining objectivity and fairness. Future work will explore multi-lingual resume support, explainable AI techniques, and integration with Applicant Tracking Systems (ATS).

. ACKNOWLEDGMENT

The authors would like to thank the Department of Artificial Intelligence & Machine Learning, Indore Institute of Science & Technology, for providing computational resources and guidance. Special thanks to Piyush Parmar, Assistant Professor, for invaluable supervision and mentorship throughout this research.

. REFERENCES

[1] Y. Zhang and M. Wallace, “A Sensitivity Analysis of Convolutional Neural Networks for Sentence Classification,” arXiv:1510.03820, 2015.

[2] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” NAACL, 2019.

[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” ICLR, 2013.

[4] F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” JMLR, vol. 12, pp. 2825–2830, 2011.

[5] A. Vaswani et al., “Attention Is All You Need,” NeurIPS, 2017.

[6] S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python. O’Reilly Media, 2009.

[7] K. Hamadache, S. Laiche, and A. Attal, “Automated Resume Screening System Using NLP and Machine Learning,” Proc. IEEE ICACIT, 2022.

[8] P. Jindal and D. Singh, “Resume Shortlisting Using Machine Learning Algorithms,” Int. J. Adv. Res. Comput. Commun. Eng., vol. 8, no. 3, 2019.