SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I9P105

Volume 12 Issue 9, 63-71, September 2025

Original Article

Hybrid Transformer-Based Resume Parsing and Job Matching Using TextRank, SBERT, and DeBERTa

Madhu Bala Myneni[1], Fathima Sha Quadhari[2]

[1,2]*Computer Science and Engineering, VNR-Vignana Jyothi Institute of Engineering and Technology, Telangana, India.*

[2]*Corresponding Author : fathimashaq@gmail.com*

Received: 05 July 2025 Revised: 07 August 2025 Accepted: 06 September 2025 Published: 29 September 2025

***Abstract** - As the job markets have become very competitive, distinguishing oneself has become very important. Resume summarization and ranking have emerged as a crucial task for processing and handling large volumes of resumes. With the advent of Natural Language Processing (NLP) techniques, automated resume summarization has gained significant attention due to its potential to expedite the hiring process while ensuring fairness and objectivity. This paper focuses on hybrid transformer-based resume parsing and job matching by implementing TextRank, SBERT, and DeBERTa. DeBERTa, an advanced transformer, functions on a disentangled attention mechanism for contextual understanding of the words, TextRank, SBERT and PageRank algorithms for extractive summarization. The ranking of candidates is done by calculating a composite score, which includes evaluation metrics like cosine similarity for job description match based on understanding the context. Difflib- a sequence matcher for candidates' experience fit, and Jaccard similarity for skills match. These scores are weighted based on their importance to the job, creating a balanced and tailored ranking. This approach focuses on saving time, reducing labour costs, and making recruitment more efficient by identifying the best matches for each position.*

***Keywords** - Natural Language Processing, Recruitment automation, Resume, Sentence-BERT, TextRank.*

1. Introduction

It is a known fact that recruitment is the most important area of any organization, as all organization employees are major assets. Finding the right fit for the job helps the organization to achieve its organization goals. The first important thing that companies look for in a candidate is skills, and next is experience. These two factors help recruiters to pick the right candidate for the role. Organizations do not focus on filling positions but on finding the candidate who aligns with the job role so that they can help establish long-term success for the organization. This means that if there is a poor recruitment process, it has a direct impact on the attrition rate. The rise of digital recruitment platforms has a significant impact on recruitment, as manual screening is very time-consuming. There are various advanced techniques, such as Natural Language Processing and Machine learning techniques, that have revolutionized this process. These processes help in quickly analyzing large volumes of resumes and matching them to job descriptions based on skills, experience, and contextual relevance. By automation, labor cost is reduced, bias is minimized, through which companies can stay competitive in talent acquisition. The key component of the proposed model is the integration of advanced NLP techniques, such as the TextRank framework work, including Sentence-BERT for extractive summarization, DeBERTa for contextual understanding, and spaCy for entity recognition. The integration of these tools helps in evaluating critical information such as name, contact information, summaries and experiences. The combination of methods, like Regular expressions, helps in the extraction of specific information such as contact numbers and emails. The proposed model ranks the resumes based on the composite score, which combines cosine similarity, Sequence Matcher, and Jaccard similarity and ranks the candidates based on weighted factors generated by the weighted score model. The other important factor is the importance of few-shot training, a machine learning approach that allows the model to learn from a small number of examples. This enables the system to adapt quickly to new job roles with minimal data. By fine-tuning the model with just a few relevant examples, it is able to accurately identify patterns and prioritize candidates who are the best match for the role.

Over the course of the study, the following research gaps were identified:

Handling diverse resume formats with unstructured formats. Retrieving data from unstructured resumes often leads to incomplete data [1]. The keyword-based parsers failed to extract structured information from unstructured resume formats, resulting in a mismatch in extracting the education qualification. Similarly, traditional NER systems often fail to extract entities like skills or education due to the complex formats of resumes [2]. These gaps are addressed in the proposed model with the use of spaCy's NER and regex to robustly extract entities such as skills, education, and contact details from different resume formats.

The traditional system generally depends on keyword matching, as it may not address the job description's semantic meaning and contextual relevance, which can lead to inaccurate results [7, 25]. Further addressing that systems using basic similarity metrics like cosine similarity may give much importance to word frequency, neglecting the contextual understanding of the words. This gap is bridged by integrating SBERT to identify semantic similarity and DeBERTa to understand the deep contextual meaning of the text. The semantic overlap between Job descriptions and resumes is analysed by using SBERT, while DeBERTa captures contextual nuances, such as the relevance of a candidate's experience to specific job roles. The composite scoring system balances multiple factors such as role, skills, experience, and context, prioritizing job-specific criteria instead of depending on keyword matches only.

The proposed hybrid transformer model is able to streamline the resume ranking process using natural language processing and machine learning techniques. In the proposed hybrid approach, integration of various techniques, including TextRank and SBERT for extractive summarization, DeBERTa for understanding contextual analysis, and spaCy for Named Entity Recognition (NER), is used for the implementation of an intelligent resume parsing pipeline.

This paper is organized into multiple sections. Section 2 discusses different resume parsing models utilized. Section 3 explains the methodology adopted for resume parsing and job description matching. And Section 4 demonstrates the entity extraction, summarization and selecting the optimal matched resume.

2. Review of Literature

Resume screening makes the process of recruitment easier by finding the most qualified candidate from large pools of applications [1]. The evolution of automated systems helps the hiring teams to work more efficiently by extracting structured information from resumes in many different formats [2]. The Early approaches using BERT showed strong accuracy in end-to-end resume parsing, helping to match candidates with job descriptions [3]. NER-based parsers further improved processing by identifying entities such as skills and education qualifications [4]. When combined with machine learning models, NER enhanced its ability to even predict suitable candidates. This helped in providing valuable insights for HR analytics [5]. Tools built with SpaCy made it simpler to summarize resumes and match them to job descriptions, which further improved the accuracy of decisions made in the recruitment process [6]. By utilizing the concept of S-BERT embeddings, which aligns resumes with job requirements, the hiring process becomes smarter [7]. Transformer techniques like BERT, whose ability to understand context in both directions has completely changed the concept of resume parsing [8]. There are advanced versions of transformers such as RoBERTa, which, when combined with multi-task learning, is able to extract relationships between complex resume elements [9]. Summarization plays a vital role in resume parsing, as brief summaries help recruiters to have a quick glance at the candidate's profile instead of going through the entire resume. Algorithms like TextRank and BM25+ help in generating extractive summaries by highlighting the most important sentences in resumes [10]. For large-scale screening, DeBERTa's unique attention mechanisms help enhance accuracy [11]. Transformer-based ranking systems further improved shortlisting by matching resumes more closely with job criteria [12]. Integrating transformer models like enhanced versions of RoBERTa, combined with GloVe features, provides better parsing for addressing the specialized needs in various fields [13]. Evaluation techniques like cosine similarity are used to measure if a candidate profile matches the job description [14]. Instruct-DeBERTa generated better results for advanced resume parsing, though it was originally adapted for sentiment analysis [15].

Certain techniques of machine learning, like Conditional Random Fields (CRFs), helped NER models better understand context in resumes [16]. Statistical models such as Hidden Markov Models (HMMs) and Support Vector Machines (SVMs) are used to segment the resume content using probability-driven transitions [17]. Abstractive summarization has been explored to generate brief summaries of resumes, though they are less common in hiring workflows [18]. RoBERTa's strong pre-training also made it effective for parsing large datasets of resumes [19]. Large language models further enhanced the area of abstractive summarization, helping produce meaningful candidate profiles [20]. Evaluation Metrics like Jaccard coefficients have been used to measure keyword overlap, which is useful in identifying how candidates can be matched to the Job Description [21]. Python implementations of cosine similarity have made large-scale matching faster and easier [22].

NER models available through Hugging Face have proven effective for extracting structured data from unstructured resume formats [23]. Hybrid methods that combine regular expressions with NER have also helped systems adapt to different resume formats [24]. Overall, NLP techniques have automated resume analysis across diverse formats [25]. Early parsers laid the groundwork for today's advanced screening systems [26]. S-BERT and cosine similarity have improved query relevance for matching candidates to jobs [27]. S-BERT has also been applied to extractive summarization to highlight key resume sentences

[28] and to ranking methods that optimize alignment between candidates and job descriptions [29]. Finally, addressing that the research on summarization helped in understanding the importance of abstractive and extractive summarizations in resume analysis [30].

3. Materials and Methods

3.1. Problem Statement

Consider the resume document to be D ={w1,w2,...,wN} with N total words.
1. Segmentation: The segmentation function:
D → {s1, s2, s3, s4}
where each segment
si = {wi1, wi2,...,wi Mi}⊂D and represents:
s1: Personal details
s2: Education
s3: Work experience
s4: Skills
Every word belongs to one segment:

$$\cup_{i=1}^{4} s_i = D, \sum_{i=1}^{4} |s_i = N$$

2. Entity Extraction
Entity types to identify as
θ={e1, e2 ,...,ek}

For each section si, form a dataset:
Di={(xi,y)∈Xi×Yi}
Where Xi are token sequences (from si), and Yi are corresponding labels.

The model is trained using supervised learning to minimize cross-entropy loss:
θ = argminθ (1/R) ∑ (r=1 to R) l(Yr, f(xr, θ))
Where:
R: Total training samples
l: Cross-entropy loss
f(xr, θ): Model predictions given input xr

The proposed approach formalizes automated resume parsing as segmentation plus sequence labelling. It is accuracy-driven as it minimizes the predictive error over labelled data using these mathematical notations.

3.2. Methodology

The methodology consists of three key stages: segmenting the resume, in which entity extraction and summarization are done. Job Description parsing and finding the optimal match. The model is first trained with ground truth data using few-shot learning in DeBERTa to capture word context. spaCy extracts entities, and Regex identifies contact details like email and phone number. Once key entities and summaries are extracted, they are stored for later use. When the end user submits a job description, the system parses this input and compares it against the stored resume summaries and entities. Matching resumes are displayed based on their alignment with the job role and other extracted criteria, enabling efficient and targeted candidate selection.

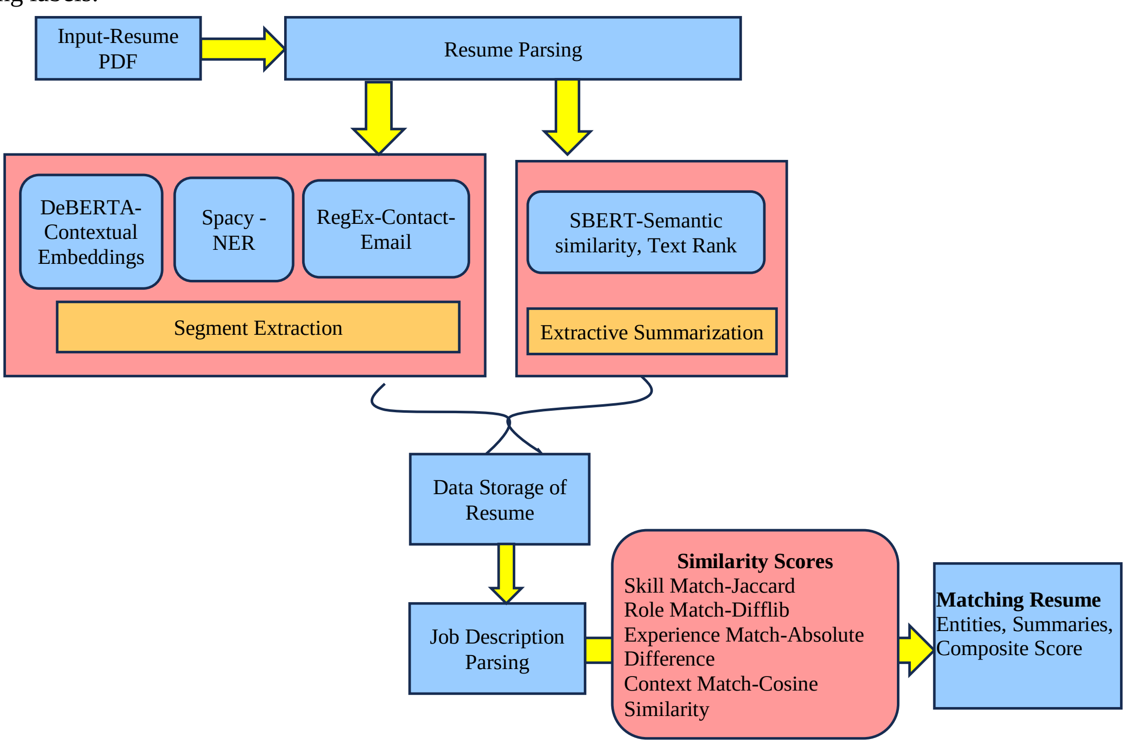

Fig. 1 Block diagram of hybrid transformer-based framework for resume parsing and job matching

Figure 1 represents a hybrid, automated resume parsing and matching system that leverages a blend of advanced NLP, embedding models, and classic computational techniques for end-to-end job matching and scoring.

Input: Resume PDF- The initial input is a resume document (PDF format). The system extracts raw text data as the foundation for further analysis.

Resume and JD Parsing and Extraction will be performed in two phases: segment extraction and extractive summary. Job descriptions undergo a similar extraction and embedding process, ensuring consistent representation for matching.

Data Storage **-** Parsed entities, summaries, and vector embeddings are stored in a structured database for fast retrieval and comparisons during job matching.

3.2.1. Composite Scoring & Output

All similarity and match scores are integrated into a composite score, representing the overall suitability of the resume for a specific job. Entities, summaries, and scores can be presented to recruiters for interpretability or used for automated filtering and ranking.

The proposed hybrid pipeline ensures robust, multifaceted evaluation of resumes for job matching. Combining advanced NLP, embedding models, and classic similarity algorithms offers both high accuracy in parsing and nuanced, fair scoring in matching processes.

3.3. Segment Extraction

The segment extraction focuses on extracting entities such as names with the help of SpaCy and DeBerta. The few-shot training provided to the model helps in understanding the contextual embedding from the DeBERTa and its impact on the text. DeBERTa builds word representations that consider the context in which each word is represented. Instead of treating every word independently, in a sentence, it separates the meaning of a word from its position and pays attention to the distance the words are from each other.

This approach helps capture the whole meaning of the sentence instead of just focusing on each word independently. SpaCy, on the other hand, acts as a practical toolkit that works with text easily. It does so by dividing the sentences, identifying entities, and parts of speech tagging. In the proposed model, it has been integrated into powerful transformer models like DeBERTa. Integrating it together helps in creating a workflow where text is not just processed mechanically but understood in a way that reflects how they are interpreted.

Organizes the extracted information into structured "segments," like education, experience, skills, and contact details for easy processing.

3.3.1. Contextual Embeddings with DeBERTa (microsoft/deberta-v3-base)

Applies contextual transformer embeddings to represent resume sections with rich, context-aware vectors, ensuring nuanced meaning is retained for later similarity comparisons.

3.3.2. Spacy NER

Utilizes Named Entity Recognition to extract entities such as names, organizations, education, skills, roles, and dates from raw resume text.

3.3.3. RegEx (Email/Contact)

Regular expressions are used for pattern-based extraction of emails and contact numbers. Significantly uses predefined patterns to extract structured data from resume text.

3.3.4. Cosine Similarity

In the skills extraction process, Cosine similarity, integrated with DeBERTa embeddings, is used to semantically compare the sentences in the resume with the list of skills in the ground truth data. It helps the system identify the skills, even when phrased differently. For example, identifying machine learning in the resume text as matching Machine Learning in the skills list.

3.4. Extractive Summarization

The proposed model generates the extractive summaries process by encoding sentences with SBERT (using the all-MiniLM-L6-v2 model), which is used to generate meaningful vector representations. After this, the Cosine similarity is calculated between these embeddings, from which a similarity graph is constructed. It is generated by identifying the proximity of the sentences related to each other in the document. On this graph, NetworkX's nx. The PageRank algorithm is applied to determine the importance of each sentence, after which the ranking of the top three sentences is made. For the final output, these top-ranked sentences form a cohesive summary. The importance of cosine similarity lies in determining sentence-level relationships, enabling TextRank to focus on key insights of the content.

3.4.1. TextRank

A graph-based approach, integrated with PageRank, identifies and ranks the most important sentences in a resume for constructing a concise summary. This helps capture high-impact content for recruiters.

3.4.2. SBERT Semantic Similarity

Sentence-BERT generates embeddings for resumes and job descriptions. It computes their semantic closeness using cosine similarity, quantifying how meaningful the match is.

3.5. Job Description Parsing

The parsing of job descriptions is done to extract roles and experience details accurately from job descriptions. It is done by adopting several Natural Language Processing (NLP) techniques. Initially, Regular expressions (Regex) are applied to capture specific patterns like years of experience. The next important factor is the matching of roles; the proposed model uses a fuzzy logic technique, such as the difflib sequence matcher for role matching. This helps identify how two strings are related to each other; for example, if the role is for a senior software engineer and the resume only has software engineer, it identifies that both are similar and maps them accordingly. It also helps identify the abbreviations and short forms.

3.5.1. Skill Match (Jaccard)

Compares the set of skills from resumes and job descriptions using the Jaccard coefficient, which measures the intersection divided by the union of skill sets.

3.5.2. Role Match (Difflib)

Python's Difflib sequence matcher is used to check for string-based similarity between candidate job titles/roles and those in the job description, even if wordings vary.

3.5.3. Experience Match (Absolute Difference)

Calculates the absolute difference in years of experience between the candidate and job requirements.

3.5.4. Context Match (Cosine Similarity)

Measures vector similarity between the contextual embeddings of the resume section and the job description to capture deeper relevance. The system evaluates resumes against a job description by calculating a composite score, then combining them and finding an optimal match.

Role alignment uses difflib. A sequence matcher will be used to compare the role extracted from the job description with the role on the resume. This contributes 10% to the final score.

Skill matching, Jaccard similarity measures the overlap between skills in the Job Description and skills extracted from the resume. This is assigned the highest weight because skills are given much importance in the area of recruitment.

Context alignment using SBERT embeddings. Cosine similarity between the job description text and the resume summary captures the semantic relationships between the texts, identifying that 'Data Analyst' and 'Data Engineer' are closely related to each other. This context score contributes 20% to the overall evaluation process.

Experience compatibility is evaluated by comparing the resume's experience to the minimum experience in the Job Description. In the proposed model, a threshold limit of two years is implemented, making the score a good match. This part accounts for 30% of the total score. Finally, these scores are combined with the formula given in 3.

$$\text{composite\_score} = (0.4 * \text{skill\_score} + 0.3 * \text{exp\_score} + 0.2 * \text{context\_score} + 0.1 * \text{role\_match\_score}) \quad (3)$$

Resumes are filtered by comparing the JD with resume entities extracted and summaries stored in data storage. After Comparison, an optimally matched resume is given as the output. The proposed system ensures that resumes are selected not on the basis of keywords but by considering multiple factors. These factors are important for finding the right fit for the job, including skills, experience, role alignment, and contextual fit, resulting in a close fit match.

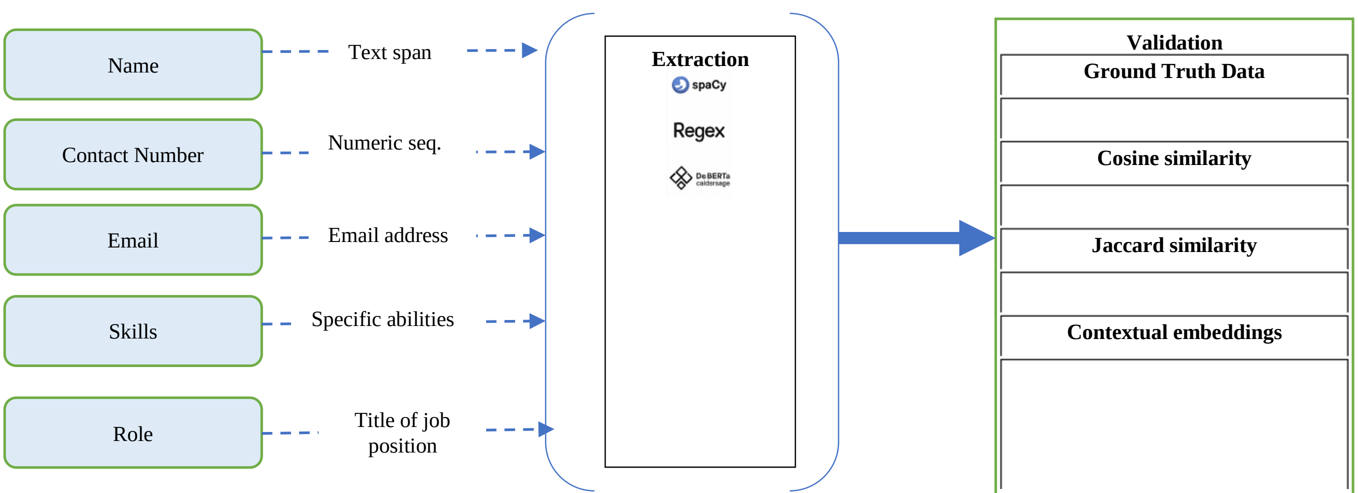

Fig. 2 Key resume components - validation

4. Results and Discussion

The results are demonstrated on a sample case that illustrates the end-to-end system output, including extracting and validating entities in raw resumes, summarizing key information, parsing the job description, computing similarity-based sub-scores, and generating a final composite score that determines candidate-job fit. The process ensures transparency and robustness (handling unstructured data), making selection evidence-based and reproducible.

The Results are analysed based on finding the appropriate match for the job role. When the job Description is matched to the resumes, a composite score is generated to understand the impact of different factors such as skills, experience and context.

It can be observed that the model is able to extract entities from unstructured resumes, which include tables and images. This can be illustrated in the results provided from Tables 1 to 4 and Figures 3 to 6.

4.1. Segment Extraction

The segment extraction focuses on performing entity extraction and extractive summarization.

4.1.1. Entity Extraction

The Following entities are extracted from the resumes: name, skills, educational qualifications, contact number, and email address. This can be illustrated in Figures 3 to 6 and Tables 1 to 4 below.

Entity	Extraction Tool	Validation
Name	spaCy NER	DeBERTa Match
Email	Regex	Ground Truth
Contact	Regex	Ground Truth
Skills	DeBERTa, Jaccard	Skill List
Exp.	Regex	DeBERTa
Role	difflib	DeBERTa

Extracted Entities from Sample Resume

- Name: Tarun Singh Chauhan (extracted via spaCy NER, verified with DeBERTa)
- Email: Chauhan.tarun897@hotmail.com (extracted by Regex, validated against ground truth pattern)
- Contact: 930650147 (Regex, validation for phone format)
- Skills: Flutter, Dart, MO Engage, Fresh Signal (extracted and matched via DeBERTa + Jaccard to the skill list)
- Experience: 4 years (Regex for years, DeBERTa verifies context)
- Role: Flutter Developer (difflib + DeBERTa for context)

Entity Extraction for different Job Roles

Table 1. Extracted entities for sample role1: Flutter developer

Entity	Value Extracted	Ground Truth
Name	Tarun Singh Chauhan	Tarun Singh Chauhan
Email	Chauhan.tarun897@hotmai.com	Chauhan.tarun897@hotmai.com
Contact	930650147	930650147
Skills	Flutter, Dart, MO Engage, Fresh Signal, PayPal	Flutter, Dart, MO Engage, Fresh Signal, PayPal
Exp.	4 years	4 years
Role	Flutter Developer	Flutter Developer

Table 1 represents the entities extracted for the role of Flutter Developer; all the extractions are validated with the ground truth data.

Table 2. Extracted entities for sample role 2: Project manager

Entity	Value Extracted	Ground Truth
Name	Vignesh	Vignesh
Email	Ragul@buzarg.in	Ragul@buzarg.in
Contact	9962599205	9962599205
Skills	Project Management, Product Management	Project Management, Product Management.
Exp.	4 years	4 years
Role	Project Manager	Project Manager

Table 2 represents the entities extracted for the role of Project Manager; all the extractions are validated with the ground truth data.

Table 3. Extracted entities for sample role 3: Electrical engineer

Entity	Value Extracted	Ground Truth
Name	Hardik T.Mehta	Hardik T.Mehta
Email	Mehta.hardik.13@gmail.com	Mehta.hardik.13@gmail.com
Contact	9924139147	9924139147
Skills	MS Office, Matlab, Etap, PSIM	MS Office, Matlab, Etap, PSIM
Exp.	2.1 years	2.1 years
Role	Electrical Engineer	Electrical Engineer

Table 3 represents the entities extracted for the role of Electrical Engineer; all the extractions are validated with the ground truth data.

Table 4. Extracted entities for sample role 4: AR/VR developer

Entity	Value Extracted	Ground Truth
Name	Souradeep Ash	Souradeep Ash
Email	Souradeepraj@gmail.com	Souradeepraj@gmail.com
Contact	918617557101	918617557101
Skills	AR, VR, Unity, Figma,c++	AR, VR, Unity, Figma, c++
Exp.	1.5 years	1.5 years
Role	AR/VR Developer	AR/VR Developer

Table 4 represents the entities extracted for the role of AR/VR Developer; all the extractions are validated with the ground truth data.

4.1.2. Summaries Generated for different Job Roles

The summaries are generated by implementing SBERT, Text Rank and Page Rank algorithms. SBERT is used to understand the semantic relation between the text, and Text Rank creates a graph by connecting the most similar sentences.

Job role 1: Flutter Developer

Role: Flutter Developer;
Experience: 4.0 years;
Skills: Flutter, Dart, Firebase, Mo-Engage, One-Signal, Freshchat, Maps, branch, Unbxd, Razorpay, PayPal, SQLite, NoSQL, Building Shop page, product banners, OTP login flow, Enhancing user experience, Zoom images, Cached image & Video Player, Application development: - Bug fixes, App optimization, rolling out app updates.; EDUCATION Bachelor of Technology (B. Tech) Mesky India Online Pvt. Ltd (Current) Gurgaon In Computer Science & Flutter Developer (SDE-2) July 2023 - current Engineering (CSE) Android Link: - Mesky Ap Play Store: - Mesky iOS Audiences: - 1k+ Amity University Haryana Integrating MOEngage SDK for Event tracking, Push Notification, InApps July 2015 - July 2019 Enabling Dep linking & user event tracking with Branch SDK Integration • unbx API for Auto Suggest, Searching, Filtering, Events Tracking; Play Store, Ap Store, Signed APK • Implemented Cubits with Clean Architecture HTTP

Fig. 3 Extractive summary for flutter developer

Figure 3 depicts the Extractive Summary generated for the role of Flutter Developer.

Job role 2: AR/VR Developer

Role: AR/VR Developer;
Experience: 1.5 years;
Skills: AR, VR, Unity, C#, Figma, UI Design, Project Management, IoT, Digital Twin; Institutions: HiDs Technology;
Education: B.Tech in relevant field 01 / 2024 – 07 / 2024 | Bengaluru, Karnataka, India. Managing Product like "weldverse" - VR Project, Created UI using Figma and Augmented Reality SDKs • managing all the documents (Gathered data) for this project, managed a Team of 4 Virtual Reality SDK's people, this project focuses on EdTech and Training Industry.; CERTIFICATE CAD-MECH Developing ar/vr/mr/xr aps AR-VR Developer with webxr, unity & unreal Courseura.08 / 2023 – 10 / 2023 | Remote Achievements/Tasks: Extended reality for • Build Augmented Reality application for their Product to showing case there working every body specialization Pune, India AR; • AR / VR Developer Technology in Stack: Threls •01 / 2023 – 07 / 2023 | Pune, Maharashtra, Indiaworked on "AR with IoT & Digital twin" Project called "MS 4.0" an Industry ARJs.

Fig. 4 Extractive summary for AR/VR developer

Figure 4 depicts the Extractive Summary generated for the role of AR/VR Developer.

Role: Electrical Engineer;
Experience: 2.1 years;
Skills: MS Office, MATLAB, PSIM, ETAP, PSCAD, Pro-E, JMAG, MAXWELL, Power System Analysis, Substation Designing, Transmission (Electrical Power Systems), Nirma University. Education: M.Tech in Electrical Engineering (Electrical Power Systems) from Nirma University, 7.46 PPT, B.E. (Electrical) from Sankalchand Patel College of Engineering, Visnagar, 7.72015 - 7.40 PI*GujaratSankalchand patel college of 7.71B.E. (Electrical) Technological 2012engineering, Visnagar; Company: Gujarat State Electricity Corporation Ltd, Thermal Power Station, Gandhinagar; Summer Internship SynopsisCompany: Gujarat State Electricity Corporation Ltd, Thermal Power Station, Gandhinagar

Fig. 5 Extractive summary for electrical engineer

Figure 5 depicts the Extractive Summary generated for the role of Electrical Engineer.

Role: VR/AR Project Manager;
Experience: 4.0 years;
Skills: Unity, C#, VR, AR, XR, Project Management, Product Management, Mobile Game Development, PC Game Development, E-Learning
Education: Bachelor of Engineering from St Muthukumaran Institute of Technology. As a Developer, Project and Product manager in VR/AR, I bring a deep understanding of the technology landscape and an unwavering commitment to delivering high-quality immersive experiences.; VR/AR/XR Project Manager/Product Designer Product management Experience designing VR/AR products and experiences. Project delivery Strong project/Product management skills, including budgeting, Mobile game app dev, scheduling, and risk management. PC game dev Experience working with cross-functional teams, including designers, engineers, and stakeholders. E-Learning Strong communication skills, including the ability to effectively convey video learning technical information to non-technical stakeholders.

Fig. 6 Extractive summary for project manager

Figure 6 depicts the Extractive Summary generated for the role of Project Manager.

4.3. Job Description Parsing and Matching

Job description matching is done by parsing through the job description provided in the prompt and comparing it with the entities and summaries extracted.

Sample Job Description:
Job Role Sought: Flutter Developer
Skills Required (from JD): Flutter, Dart, Firebase
Min Experience: 3 years

4.4. Resume Selection using Composite Scoring

Using the formula:

$$composite\_score = 0.4 \times skill\_score + 0.3 \times exp\_score + 0.2 \times context\_score + 0.1 \times role\_match\_score$$

By considering,

Skill Match (Jaccard): 80% (significant skills overlap)

Experience Match: Full score (4 years > 3-year minimum)
Context Match (SBERT cosine): High (relevant project, tools, and framework overlap)

Role Match (difflib): 100% (exact title match)
Composite Score Produced: 0.74

Table 5. Composite Scores generated for job roles

Job Role	Composite Score
Flutter Developer	0.74
Electrical Engineer	0.82
AR/VR Developer	0.76
Project Manager	0.79

Table 5 represents the composite scores generated for different Job Roles. Based on the above, the resume for Tarun Singh Chauhan is ranked highly for the Flutter Developer role, since the extracted entities, summary, and composited scores indicate a strong fit in skills, experience, and context.

5. Conclusion

The Proposed model represents a robust system for finding the right fit for the job. It does not focus on keyword extraction but on the semantic and contextual understanding of the text. The model effectively extracts key entities like name, contact number, email, education, skills, experience, role, and summary from resumes. It also matches them against Job Descriptions (JDs). This is done by integrating rule-based and machine learning techniques.

The Precision, Recall and F1 score is 81.25 percent. Whereas for Extractive Summarization, SBERT and TextRank are implemented to generate brief summaries. Job Description parsing integrates regex, SpaCy tokenization, fuzzy matching, and SBERT's cosine similarity to extract roles, skills, and experience from different formats of resumes. For finding the right match, it combines difflib for role matching, Jaccard similarity for skills, and SBERT's cosine similarity for context, and for each of these components, weights are allocated respectively to prioritize top candidates. This approach helps in considering all the factors that are important for resume parsing; further, it can be used for resolving the complex recruiting process.

References

[1] Satyaki Sanyal et al., "Resume Parser with Natural Language Processing," *International Journal of Engineering Science and Computing*, vol. 7, no. 2, pp. 4484-4489, 2017. [Google Scholar]

[2] Alessandro Barducci et al., "An End-to-End Framework for Information Extraction from Italian Resumes," *Expert Systems with Applications*, vol. 210, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Vedant Bhatia et al., "End-to-End Resume Parsing and Finding Candidates for a Job Description using BERT," *arXiv Preprint*, pp. 1-6, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[4] G.O. Narendra, and S. Hashwanth, "Named Entity Recognition based Resume Parser and Summarizer," *International Journal of Advanced Research in Science Communication and Technology*, vol. 2, no. 1, pp. 451-456, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] B.V. Brindashree, and T.P. Pushphavathi, "HR Analytics: Resume Parsing Using NER and Candidate Hiring Prediction Using Machine Learning Model," *International Journal of Research in Engineering, Science and Management*, vol. 6, no. 12, pp. 217-221, 2023. [Publisher Link]

[6] Sai Tarun Boddu, Sujeeth Desu, and Sreekanth Puli, "Resume Summarizer and Job Description Matcher using Natural Language Processing and Spacy," *International Journal of Scientific Research in Engineering and Management*, vol. 7, no. 11, pp. 1-11, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Asmita Deshmukh, and Anjali Raut, "Enhanced Resume Screening for Smart Hiring using Sentence-Bidirectional Encoder Representations from Transformers (S-BERT)," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 8, pp. 1-9, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] Jacob Devlin et al., "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 4171-4186, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[9] Chaoyu Fan et al., "The Entity Relationship Extraction Method using Improved RoBERTa and Multi-Task Learning," *Computers, Materials & Continua*, vol. 77, no. 2, pp. 1719-1738, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[10] Vaibhav Gulati et al., "Extractive Article Summarization using Integrated TextRank and BM25+ Algorithm," *Electronics*, vol. 12, no. 2, pp. 1-17, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Pengcheng He et al., "DeBERTa: Decoding-Enhanced BERT with Disentangled Attention," *arXiv preprint*, pp. 1-23, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Vinaya James, Akshay Kulkarni, and Rashmi Agarwal, "Resume Shortlisting and Ranking with Transformers," *Intelligent Systems and Machine Learning*, pp. 99-108, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[13] Arwa A. Jamjoom et al., "RoBERTaNET: Enhanced RoBERTa Transformer-based Model for Cyberbullying Detection with GloVe Features," *IEEE Access*, vol. 12, pp. 58950-58959, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[14] Ylber Januzaj, and Artan Luma, "Cosine Similarity – A Computing Approach to Match Similarity Between Higher Education Programs and Job Market Demands Based on Maximum Number of Common Words," *International Journal of Emerging Technologies in Learning*, vol. 17, no. 12, pp. 258-268, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] Dineth Jayakody et al., "Instruct-DeBERTa: A Hybrid Approach for Aspect-based Sentiment Analysis on Textual Reviews," *arXiv Preprint*, pp. 1-12, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Wahab Khan et al., "Named Entity Recognition using Conditional Random Fields," *Applied Sciences*, vol. 12, no. 13, pp. 1-18, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Diksha Khurana et al., "Natural Language Processing: State of the Art, Current Trends and Challenges," *Multimedia Tools and Applications*, vol. 82, pp. 3713-3744, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Zijian Győző Yang et al., "Abstractive Text Summarization for Hungarian," *Annals of Mathematics and Computer Science*, vol. 53, pp. 299-316, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19] Yinhan Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv Preprint*, pp. 1-13, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[20] Gunjan Keswani et al., "Abstractive Long Text Summarization using Large Language Models," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 12s, pp. 160-168, 2024. [Google Scholar] [Publisher Link]

[21] Suphakit Niwattanakul et al., "Using of Jaccard Coefficient for Keywords Similarity," *Proceedings of the International MultiConference of Engineers and Computer Scientists*, Hong Kong, vol. 1, pp. 380-384, 2013. [Google Scholar] [Publisher Link]

[22] Ahmad Farhan AlShammari, "Implementation of Text Similarity using Cosine Similarity Method in Python," *International Journal of Computer Applications*, vol. 185, no. 2, pp. 1-4, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[23] Rimmalapudi Rajesh et al., "Resume Parsing Using Named Entity Recognition and Hugging Face Model," *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 13, no. 3, pp. 2585-2592, 2024. [CrossRef] [Publisher Link]

[24] G.R. Gurushantha Murthy, Shinu Abhi, and Rashmi Agarwal, "A Hybrid Resume Parser and Matcher using RegEx and NER," *2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)*, Faridabad, India, pp. 24-29, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[25] Mehtap Saatci, Rukiye Kaya, and Ramazan Ünlü, "Resume Screening with Natural Language Processing (NLP)," *Alphanumeric Journal*, vol. 12, no. 2, pp. 121-140, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[26] Irfan Ali et al., "Resume Classification System Using Natural Language Processing and Machine Learning Techniques," *Mehran University Research Journal of Engineering and Technology*, vol. 41, no. 1, pp. 65-79, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[27] K. Venkatesh Sharma et al., "Enhancing Query Relevance: Leveraging SBERT and Cosine Similarity for Optimal Information Retrieval," *International Journal of Speech Technology*, vol. 27, pp. 753-763, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[28] Fahd A. Ghanem et al., "Deep Learning-Based Short Text Summarization: An Integrated BERT and Transformer Encoder–Decoder Approach," *Computation*, vol. 13, no. 4, pp. 1-21, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[29] Neha Aidasani et al., "AI-Powered Resume Ranking System: Enhancing Recruitment Efficiency Through Natural Language Processing," *International Journal for Research Trends and Innovation*, vol. 10, no. 5, pp. 367-371, 2025. [Publisher Link]

[30] S. Divya et al., "Unified Extractive-Abstractive Summarization: A Hybrid Approach Utilizing BERT and Transformer Models for Enhanced Document Summarization," *PeerJ Computer Science*, vol. 10, pp. 1-26, 2024. [CrossRef] [Google Scholar] [Publisher Link]