Enhancing Resume Analysis: Leveraging Natural Language Processing and Machine Learning for Automated Resume Screening Using KSA Parameters

B.A Thisara Dilshan
Department of Industrial Management
Faculty of Science
University of Kelaniya, Sri Lanka
dilshan-im19005@stu.kln.ac.lk

P.P.G Dinesh Asanka
Department of Industrial Management
Faculty of Science
University of Kelaniya, Sri Lanka
dasanka@kln.ac.lk

2025 5th International Conference on Advanced Research in Computing (ICARC) | 979-8-3315-3098-3/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICARC64760.2025.10963312

***Abstract*—In the digital age, the recruitment landscape has changed dramatically. This study aims to optimize recruitment by automating resume screening through advanced Natural Language Processing (NLP) and Machine Learning (ML) techniques. The purpose of this research is to address the limitations of traditional Applicant Tracking Systems (ATS) by introducing a framework employing Bidirectional Encoder Representations from Transformers (BERT). A comprehensive methodology was implemented, including data preprocessing, manual dataset annotation, feature extraction via TF-IDF, and cosine similarity for ranking resumes against job descriptions. Empirical findings reveal that BERT outperforms other ML technologies including SVM, achieving a classification accuracy of 78.6%, validating its effectiveness in KSA extraction and alignment. Additionally, the system's outputs were validated by industry experts, who confirmed the accuracy and relevance of the extracted KSAs in real-world scenarios. The practical implications of this work include providing actionable insights for HR professionals, enhancing candidate-job matching, and streamlining recruitment workflows. The originality of this study lies in integrating advanced ML models for dynamic KSA classification, paving the way for more efficient recruitment practices. Future research may explore ensemble techniques and transfer learning to further refine the methodology and adapt it to diverse industries.**

***Keywords*—Knowledge-Skills-Abilities, Machine Learning, Natural Language Processing, Software Engineering**

I. INTRODUCTION

Job searching has become more efficient and accessible in the digital age. The most important tool for representing a candidate when looking for job is the CV. The initial phase in the candidate filtering procedure is resuming screening, followed by an interview or test [1]. However, there are more than enough applicants for a single job, making it difficult for an employee to choose individuals solely based on their CV/ or resume. To address this challenge, many organizations have adopted automated systems to quickly identify competent candidates. For this, most companies utilize an ATS(Application tracking system) [2], which is software that organizations use to identify most significant elements or areas of interest in CVs and ignore the rest. In this scenario, advances in Natural Language Processing (NLP) and Machine learning (ML) have been useful. [3]. While the KSAM framework, which includes Knowledge, Skills , Abilities and Mindset, is often used in broader talent evaluations, this research specifically excludes the "Mindset" factor. Mindset is excluded as it is difficult to quantify and less directly relevant to technical requirements in software engineering roles.

A. KSA in Software Engineering

Knowledge, skills, and abilities are the three basic pillars that define the qualifications and potential for success of a person in any given field. Combined together, they become a potent combination that distinguishes individuals from one another and enables them to succeed in the professions of their choice [3]. In the field of software engineering, knowledge includes understanding of concepts, principles, and application areas, such as programming languages, software development life cycles, data structures, and security practices [2]. Skills, on the other hand, are competencies resulting from experience and training: project management, problem-solving, use of tools, communication, and adaptation. Abilities are innate qualities such as analytical thinking, attention to detail, innovative thinking, resilience, and team collaboration [2].

B. Background of the research

Traditional methods of resume screening rarely systematically measure the most important parameters, which subsequently results in the large mismatches between the qualifications of the candidate and the demands of the job. This gap in the recruitment process brings the need for a more structured method that could sort and set the KSA from the resumes on a higher level. Thus, the solution given would allow the recruiters and the HR managers to take a more informed decision and hence the matching between job roles and candidate competencies would be optimized. Current practices cannot parse the essential KSA parameters. This research

Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on July 12,2025 at 04:06:05 UTC from IEEE Xplore. Restrictions apply.

focuses on establishing a holistic framework in NLP and ML for candidate evaluation in the domain of software engineering. The purpose of this research is to design a flexible model for automating KSA extraaction from resumes, which will provide an objective and efficient means of assessment compared to traditional screening. This model learns dynamically from new data, ensuring that the extracted KSAs are relevant to the current industry standards and technological advancements.

II. PREVIOUS WORK

In the constantly moving information technology (IT) sector, there has been a growing call for competent candidates with a balanced set of hard and soft skills to adapt quickly to rapid technological changes and competitive market demands [4]. Employers are increasingly looking for people who not only have technical capabilities but also strong interpersonal and adaptive skills [1]. However, the IT industry also places a high premium on ”soft skills” such as communication, leadership and teamwork, without which projects cannot be managed and teams led effectively [4].The advertisements and extracting detailed information on required KSAs. This would help not only in identifying the common skills but also in understanding the necessary requirements across different IT specializations [1].

Automated resume screening technologies have revolutionized hiring across different industries by combining advanced algorithms with machine learning techniques that efficiently parse and evaluate huge volumes of candidate resumes [5] . Systems like ATS can now extract and analyze data from resumes more effectively, resulting in better matching between JD and profiles of job seekers [2]. Moreover, AI technologies have enabled the integration of more advanced algorithms, which can learn from input data and improve over time, hence making the systems smarter and more efficient.

One can note in particular that SVMs are effective in high-dimensional spaces, which makes them ideal with respect to the complex sets of features typical of resumes [6]. Integration of NLP and machine learning in the recent years has further largely impacted resume analysis, a way of enhancing accuracy as well as efficiency in candidate evaluation. [7] examined disciplinary in online engineering resumes and conformity to formatting conventions, with disciplinary discourse as a quality metric.

This nuanced understanding of the content of resumes is all the more important for technical fields, in which professional identity is codified in detailed documentation. Continuing with this, [8] developed an Automated Resume Parsing and Ranking System henceforth Automated Resume Parsing and Ranking System (ARRS) , which uses NLP algorithms to extract and rank candidates according to predefined criteria, a great stride over the erstwhile talent acquisition process by being integrated into existing Applicant Tracking Systems. Meanwhile, [9] introduced a system using models YOLOv5 and DistilBERT for parsing and ranking resumes, further showing the potential of deep learning models in increasing accuracy and relevance in resume analysis. Complementing these developments, [10] proposed a time-aware and semantic approach to resume analysis, incorporating temporal information and semantic context to improve talent similarity calculations.

In the context of skills extraction, “Topic Modeling” method (automatic approach for allocating unstructured information into groups with common semantic patterns) operated 27 with words(or sequence of words) and their occurrence in job advertisements [4]. Topic modeling is flexible and covers different research objectives . For example some authors analyze the popularity of job skills [11]. Moreover, the use of unsupervised models for topic modeling helps to match skill sets and job occupations [12]–[14].

The integration of automated resume screening tools with broader human resource management systems is in very nascent stages [15]. This could further streamline the recruitment process and make it holistic in approach, providing talent acquisition that includes screening and other recruitment, candidate engagement, and onboarding activities [5]. However, challenges potentially biased due to non-diverse training data have been identified. Recent research focuses on developing inclusive algorithms and employing balanced datasets to mitigate this issue [6]. The adaptability of these systems to various industries, which vary greatly in required skills and qualifications, remains a challenge, addressed by developing customizable screening algorithms [5]. The correctness of resume parser depends on several factors [16], such as writing style, choice of words and syntax of written text. A set of statistical algorithms along with a complex set of rules is needed in order to suitably know and fetch the correct information from resumes. NLP and ML have the capability to do so efficiently and accurately.

III. METHODOLOGY

This section discusses the methodology presented in fig. 1, used in this research to fulfill its objectives. The methodology consists of a systematic process involving data collection, preprocessing, feature extraction, model development and evaluation. Each step is detailed below, followed by a discussion of challenges and limitations encountered during the process.

A. Data Collection

For this research, we collected a diverse set of resumes through a combination of several approaches. First, we directly reached out to software development companies to gather resumes from professionals actively working in the field. To supplement this, we also distributed a Google form to individual software engineers, asking them to upload their resumes and provide additional data such as their years of experience, technical skills, and current job roles. Additionally, new resumes were created by browsing LinkedIn profiles of experienced software engineers, extracting necessary details (excluding personal data), and using that information to generate new resumes. This multi-channel approach allowed us to

Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on July 12,2025 at 04:06:05 UTC from IEEE Xplore. Restrictions apply.

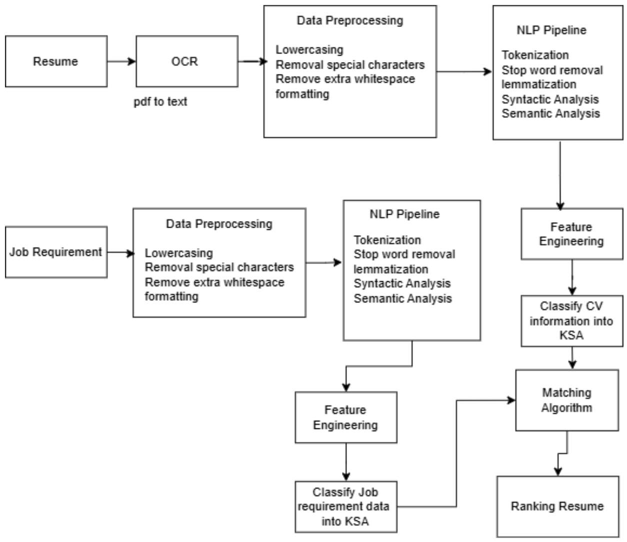

Fig. 1. System architecture

collect a total of 312 resumes from software engineers. While reviewing the dataset, several insights emerged regarding the demographic and experience levels of the candidates (fig. 2). To reduce potential biases, personal identifying information was removed, and resumes were collected from candidates across different regions and experience levels, ensuring a varied dataset. These steps aim to create an unbiased dataset for training and evaluation.

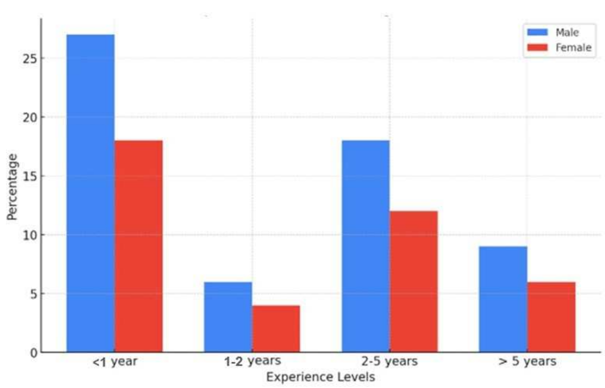

Fig. 2. Experience breakdown by gender

The data collection process faced challenges due to the variety of resume formats, inconsistent structure, and data diversity. Optical Character Recognition (OCR) was used to extract text from scanned resumes, but manual validation was necessary. The inconsistent structure of resumes made it difficult to standardize data for analysis. To address this, a detailed preprocessing stage was employed, ensuring uniformity across the dataset. Data diversity was crucial, as most resumes were from junior-level positions, skewing towards lower-experience candidates.

B. *Data preprocessing*

Data preprocessing was essential to prepare resumes and JD's for analysis using NLP techniques. Due to the variety of file formats, languages, and structures in the collected resumes, a systematic preprocessing approach was adopted to standardize the text for reliable analysis. This stage involved several steps.The first step, OCR and Text Extraction, addressed resumes submitted in non-text formats, such as scanned PDFs. Tesseract OCR, an open-source engine, was used to convert these image-based documents into machine-readable text. This step was critical, as scanned PDFs contain text in image form, which must be extracted for processing. Following OCR, a manual review of a sample of extracted resumes was conducted to verify text accuracy. This validation process was necessary, as OCR technology can sometimes misinterpret characters or fonts, particularly in low-quality scans or documents with non-standard fonts. Ensuring accurate text extraction was crucial to maintain data quality for subsequent stages.

After extracting text, the next phase, Text Cleaning and Normalization, involved preparing the content for NLP analysis by removing inconsistencies and unnecessary elements. Text cleaning tasks included converting all text to lowercase to ensure uniformity. Special characters and symbols—such as punctuation, hashtags, and currency signs—that did not contribute to KSA classification were also removed to focus on meaningful words and phrases. Additionally, whitespace formatting was standardized to address any extra spaces, new lines, or tabs that could introduce inconsistencies in analysis. This step ensures clean and uniform text structure. Python's NLTK (Natural Language Toolkit) and spaCy libraries were used for implementing these preprocessing steps, offering strong tools for text processing.

By the end of this preprocessing phase, the text from resumes and JD's was standardized and prepared for tokenization and feature extraction, ensuring that the data was ready for further analysis without residual formatting issues that could affect the accuracy of NLP-based evaluations.

C. *Proposed methodology*

In this study, a structured methodology was applied to classify and analyze the KSA parameters from resumes and JD's, facilitating an accurate match between candidates and job requirements. After cleaning and preparing the data, TF-IDF was utilized as the primary feature extraction method to create a comprehensive feature set [18].

A labeled dataset was then created, a critical step for training machine learning models in KSA classification. The cleaned text data from resumes and JDs was stored in a structured format (CSV file). Each entry was manually reviewed and annotated to label specific instances of knowledge, skills, and abilities. This manual annotation process yielded a labeled dataset that served as a foundation for training both SVM and BERT-based models, offering a precise and structured basis for accurately classifying each KSA category.

Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on July 12,2025 at 04:06:05 UTC from IEEE Xplore. Restrictions apply.

Following the dataset labeling, an NLP pipeline was implemented to further refine the data and extract relevant features. This pipeline began with tokenization, breaking down the text into individual terms, enabling detailed analysis. Stop words, which contributed little meaning, were removed to reduce noise and allow a focus on essential terms like job-specific skills and qualifications. Lemmatization followed, standardizing terms to their root forms, which reduced variations in word forms (e.g., pluralization and verb tenses) and enhanced consistency.This detailed processing produced a thorough representation of each resume and JD, capturing critical information for KSA classification.

The annotated data and features extracted through the NLP pipeline were then used to train the SVM and BERT models. SVM, known for handling high-dimensional data, proved effective in categorizing terms into Knowledge, Skills, and Abilities. The BERT model added semantic depth, capturing complex contextual meanings and relationships. During training, hyperparameters such as the penalty parameter (C) and kernel type (linear, RBF) were tuned using grid search to optimize performance.

The SVM classifier, known for its ability to handle high-dimensional data such as text, was used for training, providing an efficient approach to KSA classification. However, to further enhance the contextual understanding of textual data, the Bidirectional Encoder Representations from Transformers (BERT) model was also incorporated into the analysis. BERT, being a transformer-based model, excels in capturing semantic relationships and contextual nuances within text, offering a significant improvement in classification accuracy compared to traditional approaches like SVM.

The integration of both models allowed us to compare their strengths. Although SVM performed well in categorizing structured text data, the BERT architecture demonstrated superior ability to understand the deeper context of KSAs, especially in complex and unstructured resume data. The fine-tuned BERT model utilized pretrained weights, which were further optimized on the labeled KSA dataset to improve classification accuracy, precision, and recall. This dual-model approach provided comprehensive insights into KSA classification performance, paving the way for a more strong evaluation of resumes and JD's.

After classifying the KSA elements for both resumes and JDs, the final step involved calculating a matching score through cosine similarity. This similarity measure compared the vectorized representations of KSAs extracted from both resumes and JD's. Cosine similarity measured the cosine of the angle between the KSA vectors of a candidate's resume and a JD, producing a score that quantified the alignment between the candidate's profile and the job requirements, as represented in Eq. (1) [19].

$$\cos(\theta) = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \|\vec{b}\|} = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2} \sqrt{\sum_{i=1}^{n} b_i^2}} \tag{1}$$

A higher cosine similarity score indicated a closer match, thus ranking candidates based on their alignment with the job's required KSAs.

D. Final Output System

The developed system provides an interactive user interface designed to streamline the resume-ranking process by comparing resumes with JDs based on matching scores. The interface enables users to upload multiple resumes in PDF format and enter a specific JD for evaluation. The system utilizes the trained BERT model to extract KSAs from both resumes and JDs. It then calculates the cosine similarity score between the vector representations of the KSAs to determine the degree of alignment. As shown in fig. 3, the system ranks resumes in descending order of matching percentages, providing a clear and actionable ranking for HR professionals. This ensures that the most relevant candidates are prioritized based on their alignment with the JD. The figure demonstrates an example where two resumes are ranked with matching scores of 71.80% and 67.07%, respectively. These scores highlight the system's ability to differentiate and rank candidates effectively.

IV. Results and Discussion

The journey of classifying KSAs from resumes and JD's involved training and testing two models: SVM and BERT. Using a labeled resume dataset (80% for training), the models were designed to identify and categorize KSAs within the textual resume content. A 5-fold cross-validation was applied to mitigate overfitting and enhance generalizability. This dataset, meticulously prepared during data preprocessing, served as a foundation to evaluate the performance of the model in capturing KSAs.

The hyperparameter grid defined in this study was selected based on iterative experimentation with several configurations to optimize the performance of the SVM classifier in categorizing Knowledge, Skills, and Abilities (KSA) from textual data. Initially, broader parameter grids, such as 'C': [0.01, 0.1, 1, 10, 1000], 'kernel': ['linear', 'rbf'], 'gamma': ['scale', 'auto'], were tested, but they yielded inconsistent results, with either overfitting on the training data or failing to generalize effectively to unseen data. Subsequent refinement led to the parameter grid used in this analysis: 'C': [0.1, 1, 10, 100], 'kernel': ['linear', 'rbf', 'poly'], 'gamma': ['scale', 'auto']. This configuration produced superior results during cross-validation, as evidenced by the balanced F1-scores across the KSA categories. The inclusion of the 'poly' kernel added flexibility, while the tighter range of C values allowed for better control over model complexity. The results from grid search, with a best parameter set of 'C': 10, 'kernel': 'linear', 'gamma': 'scale', demonstrated a marked improvement in the classifier's performance, achieving high precision for "Knowledge" and strong recall for "Skills," as well as a balanced performance across "Abilities." This iterative approach underscores the importance of systematic parameter tuning in achieving a model configuration that balances accuracy and generalizability.

Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on July 12,2025 at 04:06:05 UTC from IEEE Xplore. Restrictions apply.

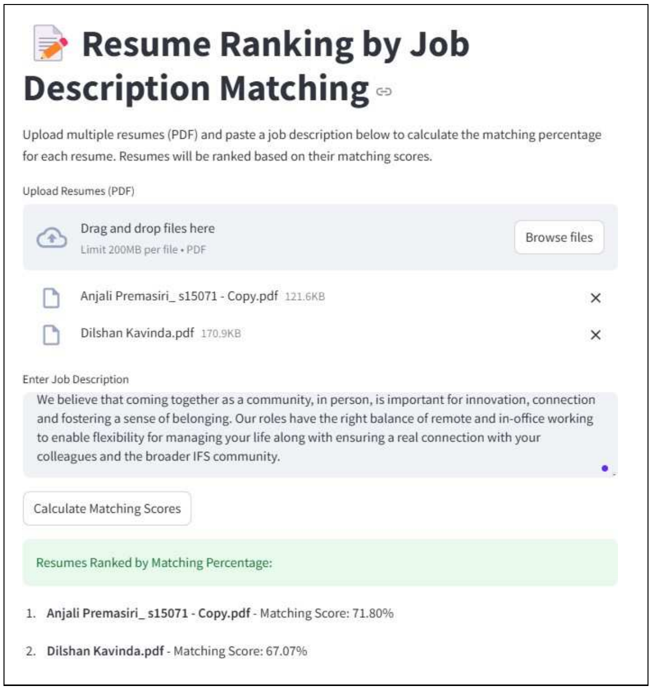

Fig. 3. Resume Ranking System Output Interface: Resumes ranked based on cosine similarity matching scores with a provided job description.

TABLE I. SVM MODEL EVALUATION

Class	Precision	Recall	F1-Score
Knowledge	0.72	0.72	0.72
Skills	1.00	0.43	0.60
Abilities	0.72	0.93	0.81

The SVM model's performance metrics, as shown in TABLE I, reveal distinct strengths and limitations. While the model achieved perfect precision (1.00) for "Skills," its recall (0.43) was significantly lower, indicating challenges in identifying all relevant instances. This imbalance resulted in a moderate F1-score (0.60) for this category. In contrast, the "Knowledge" and "Abilities" classes exhibited more balanced metrics, with F1-scores of 0.72 and 0.81, respectively. The high recall (0.93) for "Abilities" demonstrates the model's effectiveness in capturing a majority of relevant instances, while precision for this class remained consistent with the "Knowledge" category. These findings suggest that the SVM model excels in precision-driven classification but requires further improvement in recall for certain KSA categories, particularly "Skills." Such insights emphasize the need for advanced approaches to address this trade-off in performance.

TABLE II. PERFORMANCE COMPARISON of ML MODELS

Model	Accuracy	Precision	Recall	F1-Score
SVM	0.71	0.75	0.71	0.70
BERT	0.786	0.85	0.79	0.81
Logistic Regression	0.685	0.71	0.65	0.68
Random Forest	0.70	0.74	0.70	0.72

As observed in TABLE II, BERT achieved a significant improvement, particularly in precision and recall, reflecting its ability to better capture contextual nuances in KSAs compared to other models. To validate BERT's effectiveness, we conducted a manual testing phase. A sample resume and JD were classified to assess BERT's accuracy in real-world applications. The output was subsequently reviewed by industry experts, who confirmed that BERT accurately tagged KSAs relevant to JD's, providing a high degree of professional alignment. Further analysis involved comparing the Knowledge, Skills, and Abilities (KSAs) in the resume and JD using cosine similarity, which yielded a score of 79.64%, indicating strong alignment. Both focus on Java Back-End development, microservices, and database management, with shared expertise in technologies such as Spring Boot, Docker, Kafka, and REST APIs, confirming a strong match in skills and experience. This alignment suggests that the candidate is well-suited for the role. The results of our analysis reveal several key insights. First, the superior performance of BERT over other models confirms the effectiveness of transformer based models for KSA classification . BERT's architecture, which accounts for the contextual relationships between words, was able to accurately identify and categorize KSA in a way that aligned closely with human judgment. The manual testing and validation by industry experts further solidified this finding, providing confidence in BERT's practical applicability. Additionaly the performance of SVM and BERT models was compared with two additional techniques, Logistic Regression and Random Forest, as shown in table . II. While Logistic Regression achieved moderate accuracy and F1-scores, Random Forest outperformed it due to its ability to handle non-linear relationships in data. However, BERT significantly outperformed all other models, demonstrating its superior ability to capture semantic nuances in textual data. These results further validate the effectiveness of transformer-based models in the context of automated KSA extraction and resume screening.

In conclusion the ranking process based on the matching algorithm is designed to optimize candidate job fit by calculating and comparing KSA vectors between resumes and JD's. This process ensures that the most relevant candidates are identified and ranked according to their suitability for the role, greatly enhancing the efficiency and effectiveness of the recruitment process.

V. Conclusion and Future Work

While the results of this study are promising, certain limitations warrant further investigation. The dataset used for training and testing was specific to a particular industry, which may limit the generalizability of the findings across diverse sectors. Future research could address this by expanding the dataset to include a broader array of industries and job roles. Moreover, the reliance on KSAs as the sole metric for candidate evaluation overlooks other important attributes, such as personal values, cultural fit, and soft skills, which play a significant role in recruitment. Future iterations of this system could integrate additional functionality, such as real-

Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on July 12,2025 at 04:06:05 UTC from IEEE Xplore. Restrictions apply.

time analysis of social media profiles or automated feedback to candidates, further enhancing its usability. Moreover, the model can be enhanced by incorporating ensemble techniques, such as combining SVM and BERT, to leverage their individual strengths for improved KSA classification. Additionally, employing transfer learning could enable the model to adapt to diverse job roles and industries beyond software engineering. Exploring unsupervised clustering methods for identifying latent patterns in KSAs presents another promising avenue for advancing the methodology.

In conclusion, this study demonstrates the potential of BERT for automated KSA extraction and alignment in recruitment processes. The manual validation confirm that BERT can serve as a reliable tool for matching candidates to roles based on their KSAs, with potential applications in streamlining and enhancing recruitment workflow.

REFERENCES

[1] C. K. Chamithri and P. P. G. D. Asanka, "Identify the Current Demanded Knowledge, Skills, and Abilities (KSA) in IT through Analysis of Job Advertisements in Sri Lanka: A Framework Using Natural Language Processing Techniques," in *ICARC 2024 - 4th International Conference on Advanced Research in Computing*, 2024, pp. 229–234. doi: 10.1109/ICARC61713.2024.10499754.

[2] L. Mathew, N. C. George, N. Linet, and N. K. Thomas, "ATS BREAKER-A system for comparing candidate resume and company requirements," *International Journal of Engineering Research*, vol. 9, no. 6, pp. 591–594, 2020.

[3] AIHR, "Knowledge, Skills, and Abilities (KSA) Examples," [Online]. Available: https://www.aihr.com/blog/knowledge-skills-abilities-ksaexamples/. Accessed: Feb. 29, 2024.

[4] A. Ternikov, "Soft and hard skills identification: Insights from IT job advertisements in the CIS region," *PeerJ Computer Science*, vol. 8, 2022. doi: 10.7717/PEERJ-CS.946.

[5] S. Bharadwaj, R. Varun, P. S. Aditya, M. Nikhil, and G. C. Babu, "Resume Screening using NLP and LSTM," in *5th International Conference on Inventive Computation Technologies (ICICT 2022)*, 2022, pp. 238–241. doi: 10.1109/ICICT54344.2022.9850889.

[6] G. Kamineni, K. A. Sai, and G. S. N. Rao, "Resume Classification using Support Vector Machine," in *Proc. 2023 3rd Int. Conf. on Pervasive Computing and Social Networking (ICPCSN 2023)*, 2023, pp. 91–96. doi: 10.1109/ICPCSN58827.2023.00021.

[7] C. Berdanier, M. McCall, and G. Fillenwarth, "Characterizing Disciplinarity and Conventions in Engineering Resume Profiles," *IEEE Trans. Prof. Commun.*, vol. 64, no. 4, pp. 390–406, 2021. doi: 10.1109/TPC.2021.3110397.

[8] B. Nisha, V. Manobharathi, B. Jeyarajanandhini, and G. Sivakamasundari, "HR Tech Analyst: Automated Resume Parsing and Ranking System through Natural Language Processing," in *2nd International Conference on Automation, Computing and Renewable Systems (ICACRS 2023)*, 2023. doi: 10.1109/ICACRS58579.2023.10404426.

[9] S. Kinger, D. Kinger, S. Thakkar, and D. Bhake, "Towards smarter hiring: Resume parsing and ranking with YOLOv5 and DistilBERT," *Multimedia Tools and Applications*, 2024. doi: 10.1007/s11042-024-18778-9.

[10] C. Huang, L. Cheng, and F. Yao, "Time-aware and Semantic Resume Analysis for Talent Similarity Calculation," in *2023 IEEE 3rd International Conference on Digital Twins and Parallel Intelligence*, 2023. doi: 10.1109/DTPI59677.2023.10365442.

[11] A. Aken, B. Prabhakar, and K. P. Arnett, "Skills in the MIS Job Market," [Online]. Available: https://www.researchgate.net/publication/220891106. Accessed: Jul. 19, 2024.

[12] M. Pejic-Bach, T. Bertoncel, M. Mesˇko, and Zˇ. Krsti, "Text mining of Industry 4.0 job advertisements," *International Journal of Information Management*, vol. 50, pp. 416–431, 2020. doi: 10.1016/j.ijinfomgt.2019.07.014.

[13] C. G. Cegielski and L. A. Jones-Farme, "Knowledge, Skills, and Abilities for Entry-Level Business Analytics Positions: A Multi-Method Study," *Decision Sciences Journal of Innovative Education*, vol. 14, 2016. doi: 10.1111/(ISSN)1540-4609.

[14] O. Mu¨ller and J. vom Brocke, "Comparing Business Intelligence and Big Data Skills: A Text Mining Study Using Job Advertisements," [Online]. Available: https://www.uni.li. Accessed: Mar. 12, 2024.

[15] M. Li, "The impact of artificial intelligence on human resource management systems: Applications and risks," *Applied and Computational Engineering*, vol. 48, no. 1, p. 716, 2024. doi: 10.54254/2755-2721/48/20241060.

[16] "Re´sume´ parsing," *Wikipedia*, [Online]. Available: https://en.wikipedia.org/wiki/ResumeParsing. Accessed: Aug. 17, 2024.

[17] G. Sidoro, A. Gelbukh, H. Go´mez-Adorno, and D. Pinto, "Soft similarity and soft cosine measure: Similarity of features in vector space model," *Computacio´n y Sistemas*, vol. 18, no. 3, pp. 491–504, 2014.

[18] S. Jabri, A. Dahbi, T. Gadi, and A. Bassir, "Ranking of text documents using TF-IDF weighting and association rules mining," in *2018 4th International Conference on Optimization and Applications (ICOA)*, 2018, pp. 1–6. doi: 10.1109/ICOA.2018.8370556.

[19] B. Stecanella, "What Is TF-IDF?" [Online]. Available: https://monkeylearn.com/blog/what-is-tf-idf/. Accessed: Aug. 07, 2024.

Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on July 12,2025 at 04:06:05 UTC from IEEE Xplore. Restrictions apply.