AI-Based Resume Screening and Interview Question Generation Using Sentence-BERT and Controlled NLP

Harshit Juneja
Amity University, Greater Noida
Uttar Pradesh, India
harshitjuneja2014@gmail.com

***Abstract*—The volume of application received by recruitment processes in contemporary organizations is large, such that manual screening of resumes is slow, inconsistent, and likely to be subjective. In this paper, an AI-supported system is introduced that will automatically put the resumes through a job description-based screening and will produce corresponding interview questions based on the profile of a candidate. The recommended solution combines the resume parsing algorithm based on Natural Language Processing (NLP), extraction of skills and text normalization and semantic matching through Sentence-BERT (SBERT) embeddings to calculate a candidate-job relevance score. A hybrid scoring method is a combination of semantic similarity and skill coverage and experience metrics to enhance the reliability of rankings. The system produces structured interview questions through a managed skill-conscious pipeline that projects extracted abilities and project cues to a maintained assortment of questions. The Precision5 and Accuracy of 100 public resumes are shown to have Precision with 0.82 and Accuracy with 0.83 showing better results than the TF-IDF and Keyword-matching baselines. Mean relevance score of questions generated through human analysis is 4.2/5 which is a high role alignment and clarity. The solution suggested is readable, scalable and applicable in the actual recruitment processes.**

Keywords—Resume Screening, Natural Language Processing, Candidate Ranking, Semantic Similarity, Sentence-BERT, Interview Question Generation, Skill Extraction, Recruitment Automation.

I. INTRODUCTION

Hiring and recruitment are important processes to any organization regardless of the industry but more so in the engineering software and technology fields where skills change fast. As online job portals are expanded and the application process is simplified, the recruiter may get a hundred or a thousand resumes per advertised job. A resume must be read and judged manually, which is time-consuming and can also lead to inconsistency because one can be tired or have different interpretations and implicit bias. Old-fashioned screening mechanisms are based on rule-based screening or keyword matching. These methods are easy to employ but often fail in practice in practice. There are various names that candidates can give to the same talent and the content used in a resume can be irrelevant thus interfering with the appearance of keywords. As an illustration, a prospective employee who has experience in so-called Natural Language Processing can spell out Text Analytics, or Language Models, or NLP, and keyword matching will not always be effective in picking them up. Another significant problem in recruitment processes is interview preparation besides screening. The interview questions must be created to suit the job requirements and profile of the candidate by the recruiters and interviewers. Practically, interview questions tend to be general, and do not exhaustively take advantage of the resume background of the candidate. This compromises effectiveness of interviews particularly in technical jobs where specific questions enhance effectiveness of the interview.

This paper has made the following contributions:

1) Sentence-BERT based resume screening model: a semantic similarity-based model of resume screening scoring based on candidate-job match scores.
2) Semantically aligned scoring approach, a hybrid strategy that incorporates semantic similarity and skill coverage and experience measures.
3) A question generation interview method that is controlled to generate technical, project-based, and HR questions based on extracted resume information.
4) Testing of the system with available data of resumes in the public and relevance score of the generated questions in the human.

II. LITERATURE SURVEY

The high-speed development of the online recruitment system and the growing number of applicants have brought an automated resume screenings and interview questions generation to a prominent position. Companies are turning to AI-based approaches to enhance their efficiency in hiring and recruiting, decrease the number of people working manually, and streamline the decision-making process. In this section, the research work has been conducted on resume screening, semantic methods of similarity, finding information on resumes and automated interview questions generation.

A. Conventional Resume screening and information retrieval strategies

Early resume screening platforms were basically based on rule-based filtering and keywords matching. The strategies usually search for specified keywords which include programming languages, tools, certifications, or degree names, and rank the results by relevance which is determined by the frequency of the keyword. Although such systems are easy to apply and computationally efficient, they easily fail when the resumes have various names of the same thing. TF-IDF and other information retrieval techniques have been used extensively in ranking documents. Term-weighting schemes suggested by Salton and Buckley are still considered the basis of retrieval-based scoring schemes, and TF-IDF is still widely used as a baseline when it comes to text similarity on recruitment tasks. Moreover, a simple way to measure relevance in the resume-job description is by computing the cosine similarity of vectors of TF-IDF. The lexical methods are, however, usually not semantically understood and are vulnerable to vocabulary mismatch and formatting noises.

B. Screening and Classification by means of Machine Learning

As machine learning began to appear, scholars studied classification and ranking models to be trained on labeled hiring data. Candidate classification and ranking entailed the use of traditional machine learning algorithms like decision trees and ensemble algorithms like Random Forests because they are easily interpretable and resistant to overfitting. These approaches usually involve systematic feature derivation like the number of skills, experience, level of education, and keywords of projects. As much as these models provide better generalization in comparison to pure keyword systems, features engineering and access to labeled data is very dependent; this is usually scarce because of privacy issues regarding recruitment data.

C. Transformer-Based Semantic Matching

Recent developments of deep learning have made systems much better understand contextual meaning in text. Word embedding algorithms like Word2Vec and GloVe brought in the use of distributed representations which were used to establish semantic similarity among words as per the context. Nonetheless, polysemy (where the meaning of words varies with context) cannot be well represented using the form of static embeddings. Language models that use transformers overcame this drawback through generating contextual embeddings. BERT proposed by Devlin et al. shows high performance in the vast variety of NLP tasks and it does so by learning deep bidirectional representations. It is not only that BERT embeddings enhance the semantic understanding, but that BERT can be inefficient and unpredictable when it comes to the sentence-level similarity.

D. Parsing and Skill Extraction Techniques

Resume Parsing and Skill Extraction Methods One of the crucial parts of automated screening is the extraction of structured information of the unstructured resumes. Resume parsing is the process of finding such entities as skills, education, work experience, certifications and projects. This has been widely studied using named entity recognizers (NER) models and rule based parsers. In recent NLP systems like spaCy, tokenization, POS tagging and NER pipelines are efficient and are typical building blocks in systems extracting resumes. Nevertheless, resumes are very diverse in the way they are laid out and the sequence of the contents, and it is hard to extract them.

E. Automated Interview Question Generating

Another line of study is generation of interview questions that seek to enhance better evaluation of candidates and preparation of interviewers. The earlier strategies were based on templates of questions and curated question banks which were rule-based. Template-based systems produce grammatically correct and structured questions and tend to be efficient in cases when a domain control is needed. Such systems might however not be flexible and diverse in case templates are restricted. The development of neural question generation approaches has been facilitated by the development of sequence to sequence learning and transformer networks.

III. PROPOSED METHODOLOGY

The system under a proposal takes a resume of the candidate and a job description (JD) as inputs and outputs: (i) a candidate-job match score, (ii) identified and matched skills, and (iii) 10-12 recommended interview questions, which are classified as technical, project-based, and behavioral. Figure 1 represents the overall system architecture. The proposed framework is a modular framework with eight interacting elements as shown in Figure 1.

A. Data Entry

The initial step assumes the entry of data and text extraction in turn. The system is compatible with PDF and DOCX resumes. To extract the text of PDF resumes, PDFPlumber is used, and layout-aware text sequence is maintained. In the case of DOCX resumes, the documents are broken down into individual paragraphs with the help of python-docx. The job descriptions are in plain text. All the content extracted in the Skills, Education, Experience, and Projects sections are joined into one normalized textual representation and then preprocessed.

B. Text Preprocessing

Preprocessing distinguishes noise, as well as standardizing input towards further NLP work. The subsequent steps are performed in that order:

a) all tokens are made lowercase;
b) all punctuation marks and non-alphanumeric are removed;
c) NLTK corpora is used to remove all stopwords;
d) all whitespaces and deduplication of tokens are merged to build skills. The matching of skills in this pipeline is more precise, and formatting artefacts do not influence similarity scores.

C. Skill Extraction

The hybrid model is used to extract skills first by matching and identifying them (dictionary-based) against a curated skill taxonomy of programming languages, programming frameworks, databases, programming algorithms and CS fundamentals; and second is extracting (NER-based) skills with the spaCy [5] tool in the case of skills that are emerging or domain-specific. Suppose that the skill set obtained out of the resume be $S= \{s_1, s_2, ..., s_n\}$ and the skill set obtained out of the job description be $S_{JD} = \{t_1, t_2, ..., t_m\}$ The skill overlap ratio is:

$$SkillMatch = \frac{|S_R \cap S_{JD}|}{|S_{JD}|} \qquad (1)$$

The ratio is a measure of the coverage of required skills and a normalized interpretable signal, that is independent of resume length and verbosity.

D. Semantic similarity Scoring by using S-BERT

Lexical matching is ineffective when the resumes employ different words to indicate similar skills. SBERT sentence embeddings are used to calculate semantic similarity [4]. R and JD, the entire text of the resume and job description, respectively, are separately represented in dense vectors of fixed length:

$$\vec{r} = \mathrm{SBERT}(\mathrm{R}), \vec{j} = \mathrm{SBERT}(\mathrm{JD}) \qquad (2)$$

Cosine similarity between both embeddings is then calculated:

$$\mathrm{Sim}(\mathrm{R}, \mathrm{JD}) = \frac{\vec{r} \cdot \vec{j}}{\|\vec{r}\| \|\vec{j}\|} \qquad (3)$$

All-MiniLM-L6-v2 is the model employed, which generates both 384-dimensional representations and is highly accuracy-computationally efficient, allowing to score in real-time with only normal CPU hardware.

E. Hybrid Candidate Ranking Score

A mixed-method scoring plan enhances the reliability of rankings higher than any other single-metric one:

$$Score = \alpha \cdot Sim(R, JD) + \beta \cdot SkillMatch + \gamma \cdot ExpWeight$$

Where, Sim(R, JD) is normalized SBERT cosine similarity, SkillMatch, required skill overlap ratio, ExpWeight an experience measure of resume patterns (year mentions, internship and project counts), and α, β, γ are tuning parameters, all of which add to 1.0. All experiments were done with default value a = 0.50, b = 0.30, g = 0.20.

F. Interview Question Generation

Question generation generates the questions by applying a controlled and skill-conscious pipeline to prevent hallucination:

a) Skills and project cues are learned off the resume.
b) Questions are extracted from a curated repository of skills to conceptual, practical and scenario-based questions, based on each extracted skill.
c) Extracted project names and technologies are used as instantiations of project-specific templates.
d) There are behavioral/HR questions of a fixed bank that are appended to holistically evaluate them. The system generates 10-12 equal opportunity questions per candidate.

IV. EXPERIMENTAL SETUP

A. Experimental Setup

There was a development of a working prototype in Python with the help of modern libraries of NLP and web-interfaces. The implementation is based on reproducibility, on-the-fly responsiveness, and simplicity to implement on a standard hardware with no graphics card needs. There is a question repository of 500 or more questions where the question category is organized according to the skills (conceptual, practical, scenario-based) and experience level. Skills are assigned to 3-6 questions each, which allows a variety of candidates of similar profile and prevents repetitive question sets.

B. Dataset

The Kaggle Resume Dataset was sampled by gathering resumes and anonymized. There were 100 resumes between four software engineering positions, namely Backend Developer, Full Stack Developer, Data Analyst, and Machine Learning Intern. Real job posting curated corresponding job descriptions. Three domain expert evaluators manually labeled the resume-JD pairs (shortlisted / not shortlisted) and majority decided on the disagreements.

Fig.1.roposed Architecture for AI-Based Resume Screening and Interview Question Generation

C. Baseline Methodologies and Measures of Evaluation

The suggested SBERT model is contrasted with two baselines (i) Keyword Overlap Baseline - the number of exact matches between preprocessed resume and JD based on keywords; (ii) TF-IDF + Cosine Baseline - implements TF-IDF vectors and sorts them by their cosines. Measures of evaluation Precision at 5 (percentage of relevant resumes in the top 5), binary shortlist with Overall Classification Accuracy, Skill Extraction F1-Score, Mean Question Relevance Score (1-5 human rating).

V. RESULTS

A. Quantitative Results

All the evaluation metrics in comparison with all the screening methods are expressed in Table I. Figure 2 visualizes precision of 5 and accuracy of the three methods. SBERT model has a relative improvement of 30.2% in Precision at 5 and 18.6% better accuracy than the TF-IDF baseline. Such gains can be attributed to the fact that semantic paraphrases, such as the one between "REST APIs" and "Backend Development," between "Pandas/NumPy" and Data Analysis," or between "Transformers" and "NLP models" can be reached using S-BERT but not using lexical techniques.

TABLE I. PERFORMANCE COMPARISON OF RESUME SCREENING METHODS

Method	Precision@5	Accuracy	Skill F1	Question Rel. (1-5)
Keyword Matching	0.55	0.65	0.61	3.1
TF-IDF + Cosine [1],[2]	0.63	0.70	0.68	3.3
Proposed SBERT Model [4]	0.82	0.83	0.87	4.2

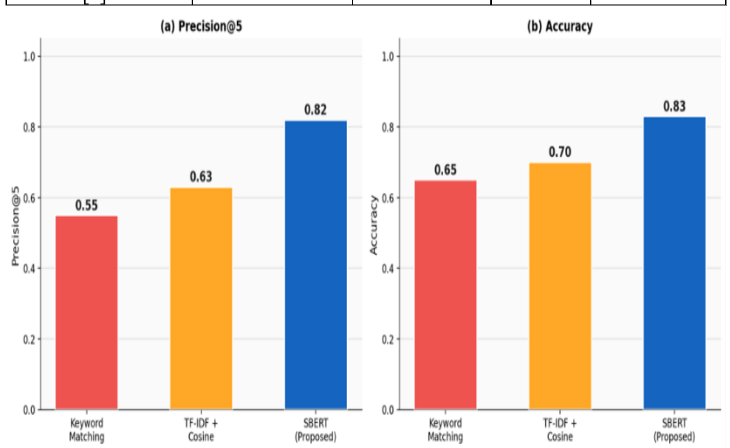

Fig.2. Performance Comparison across the methods of resumes screening

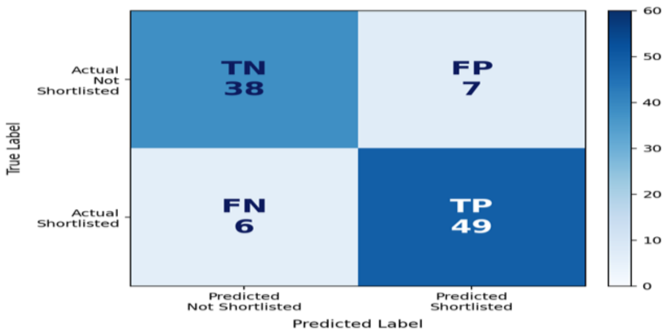

Fig.3. Confusion Matrix - SBERT based Shortlisting (n = 100 Resumes, Software Engineering JD)

B. Confusion Matrix Analysis

Figure 3 represents the confusion matrix of binary shortlisting choices of the SBERT model in the 100-resume test set. The model has 87 hits (38 TN and 49 TP), and there are only 13 misses (7 FP and 6 FN). The low false negative rate (6%) is of particular concern in the recruitment scenarios when not shortlisting a qualified candidate is a costly mistake.

C. Interview Question Relevance Report

Questions generated were rated on a scale of relevance on a 1-5 scale by human evaluators in five categories. As it is presented in Fig. 4, the SBERT-based system is always more efficient than the baselines in all question categories. The score on the skill-grounded pipeline shows the greatest relevance (4.4/5) when using the project-based questions, in which the content of the resume is candidate-specific. The types of questions produced by SBERT all have a high exceeding of the acceptable threshold of 4.0/5.

D. Score Distribution Analysis SBERT Score

Fig. 5 illustrates how SBERT cosine similarity scores of shortlisted and non-shortlisted candidates differ. There is a wide dispersion between the two distributions of 0.74 and 0.42 as the shortlisted candidates and the non-shortlisted candidates cluster around the mean distribution respectively. The most optimal precision-recall trade-off is obtained when the decision threshold is set at 0.60. The distinct bimodal divide indicates that SBERT cosine similarity is an excellent discriminative characteristic in binary resume screening.

V. COMPARISON WITH THE STATE OF THE ART METHODS

A systematic comparison of the suggested framework with the typical prior methods is given in Table II. The suggested approach is tested on the same 100-resume scale as in Section V. The proposed approach has the best Precision@5 (0.82) and accuracy (0.83) of all the compared ones, and at the same time, it can also generate controlled interview questions and has complete interpretability. Fine-tuned BERT [3] achieves accuracy of up to 0.78 though it takes considerably more labelled training data, a lot of computational power, and does not have built-in question generation. The RAG based method [9] is useful in generating questions but with less screening accuracy (0.71) and lower interpretability. The proposed framework will, therefore, provide optimal balance of accuracy, computation efficiency, interpretability and functional completeness.

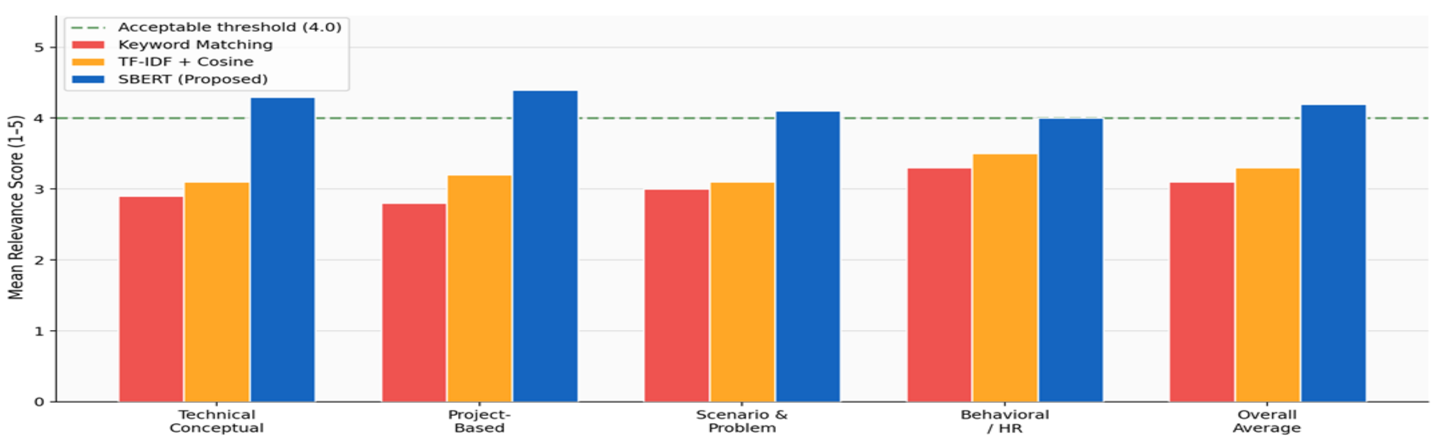

Fig.4. Mean Interview Question Relevance Score by Category (Scale 1-5; Threshold = 4.0) Human Evaluation

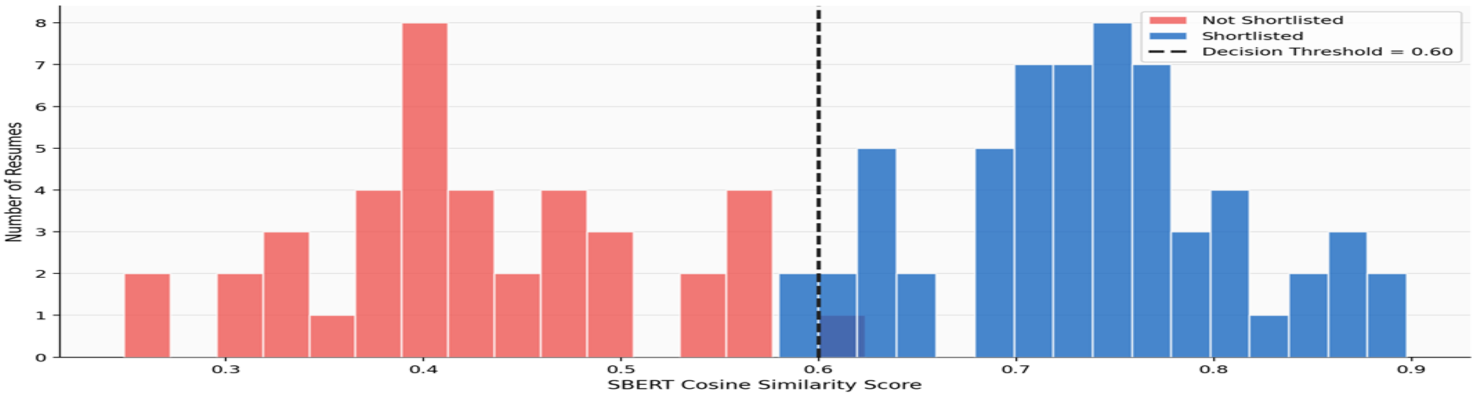

Fig.5. *SBERT Score Distribution: The Shortlisted vs. Non-Shortlisted (Decision Threshold = 0.60*

TABLE II. COMPARISON WITH STATE-OF-THE-ART RESUME SCREENING APPROACHES

Method	Approach	Prec@5	Accuracy	Q-Gen.	Interp.
Salton & Buckley [1]	TF-IDF Retrieval	0.54	0.63	No	Yes
Word2Vec+Cosine [12]	Static Embeddings	0.67	0.71	No	Partial
BERT Fine-tuned [3]	Contextual BERT	0.75	0.78	No	Partial
Random Forest [18]	Feature-Based ML	0.65	0.69	No	Yes
RAG-based Gen. [9]	Retrieval + LLM	0.71	0.74	Yes	No
Proposed SBERT+Ctrl	Semantic + Hybrid	0.82	0.83	Yes	Yes

VI. LIMITATIONS

The extraction module of the skills is also based on the integrity of the curated taxonomy; the emerging/niche technologies that are not present in the dictionary could be overlooked, which may prioritize the relevant candidates. The text extraction in the resume, which has dense graphics, multi-column formations or scanned images, can lead to the incomplete extraction of the text. The weight of experience (ExpWeight) is based on heuristic pattern matching, as opposed to organised timeline extraction. The existing assessment involves n = 100 resumes of one dataset; bigger multi-domain assessment should be conducted.

VII. CONCLUSION

The paper tried out an AI-based resume screening and interview question generation framework using Sentence-BERT semantic matching and controlled and skill-aware question generation pipeline. The system parses and normalizes resume text, finds skills through a hybrid dictionary-NER method, ranks candidates with job descriptions with the help of SBERT embeddings and the hybrid score formula, and creates structured interview questions based on the profile of a particular candidate. Experimental evidence on 100 public resumes indicates Precision at 5 of 0.82 and accuracy of 0.83 which is significantly better than TF-IDF and keyword baseline. The significance of the median relevance of the questions is proved by human evaluation of 4.2/5. The proposed approach delivers the optimal total balance between accuracy, interpretability, and functional completeness, including built-in question generation, which is not offered by any previous method related to semantic matching, which is confirmed by comparison with state-of-the-art methods.

REFERENCES

[1] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, 1988.

[2] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. NAACL-HLT*, Minneapolis, MN, USA, Jun. 2019, pp. 4171–4186.

[4] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence embeddings using Siamese BERT-networks,” in *Proc. EMNLP-IJCNLP*, Hong Kong, China, Nov. 2019, pp. 3982–3992.

[5] M. Honnibal and I. Montani, “spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing,” Explosion AI, Tech. Rep., 2017.

[6] S. Agarwal, H. Dua, and R. Agarwal, “Automated question generation using NLP: A review,” *Int. J. Eng. Adv. Technol.*, vol. 8, no. 6, pp. 1090–1095, Aug. 2019.

[7] X. Du, J. Shao, and C. Cardie, “Learning to ask: Neural question generation for reading comprehension,” in *Proc. ACL*, Vancouver, Canada, Jul. 2017, pp. 1342–1352.

[8] J. Zhou, Y. Liu, Z. Chen, and J. Tang, “Neural question generation from text: A preliminary study,” in *Proc. NLPCC*, Dalian, China, Nov. 2017, pp. 662–671.

[9] P. Lewis *et al.*, “Retrieval-augmented generation for knowledge-intensive NLP tasks,” in *Proc. NeurIPS*, virtual, Dec. 2020, pp. 9459–9474.

[10] A. Vaswani *et al.*, "Attention is all you need," in *Proc. NeurIPS*, Long Beach, CA, USA, Dec. 2017, pp. 5998–6008.

[11] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 3rd ed. Upper Saddle River, NJ, USA: Pearson, 2023.

[12] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. NeurIPS*, Lake Tahoe, NV, USA, Dec. 2013.

[13] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proc. EMNLP*, Doha, Qatar, Oct. 2014, pp. 1532–1543.

[14] C. Fellbaum, Ed., *WordNet: An Electronic Lexical Database*. Cambridge, MA, USA: MIT Press, 1998.

[15] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. ACL*, Philadelphia, PA, USA, Jul. 2002, pp. 311–318.

[16] A. M. Turing, "Computing machinery and intelligence," *Mind*, vol. 59, no. 236, pp. 433–460, Oct. 1950.

[17] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.

[18] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[19] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[20] L. Das, P. Anand, A. Anjum, M. Aarif, N. Maurya, and A. Rana, "The Impact of Smart Homes on Energy Efficiency and Sustainability," in *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, Gautam Buddha Nagar, India, 2023, pp. xx–xx.

[21] G. Parashar, A. Chaudhary, and A. Rana, "Systematic Mapping Study of AI/Machine Learning in Healthcare and Future Directions," *SN Computer Science*, vol. 2, no. 6, p. 461, 2021.

[22] A. Rana, V. Khurana, A. Shrivastava, D. Gangodkar, D. Arora, and A. K. Dixit, "A ZEBRA Optimization Algorithm Search for Improving Localization in Wireless Sensor Network," in *2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS)*, 2022, pp. xx–xx.

[23] R. Semwal, N. Tripathi, A. Rana, A. Chauhan, V. Bhutani, and K. Gupta, "Conceptual Integration of AI for Enhanced Travel Experience," in *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, Gautam Buddha Nagar, India, 2023, pp. xx–xx.

[24] L. Das, R. Salman, S. Sabeer, S. K. Ansari, M. Aarif, and A. Rana, "Customer Retention Using Machine Learning," in *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, Gautam Buddha Nagar, India, 2023, pp. xx–xx.

[25] V. Malik, R. Mittal, A. Rana, I. Khan, P. Singh, and B. Alam, "Coronary Heart Disease Prediction Using GKFCM With RNN," in *2023 6th International Conference on Contemporary Computing and Informatics (IC3I)*, 2023, pp. xx–xx.

[26] H. Basak, R. Hussain, and A. Rana, "DfeNet: A Novel Dimension Fusion Edge Guided Network for Brain MRI Segmentation," *SN Computer Science*, vol. 2, no. 6, p. 435, 2021.

[27] N. Kashyap, A. Rana, V. Kansal, and H. Walia, "Improve Cloud Based IoT Architecture Layer Security—A Literature Review," in *International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, 2021, pp. xx–xx.

[28] P. Kushwaha, A. Rana, F. Hassan, S. S. Hada, G. Bhardwaj, and V. Bhutani, "Energy Prediction in Urban Areas Using Machine Learning and Deep Learning," in *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, Gautam Buddha Nagar, India, 2023, pp. 190–195, doi: 10.1109/UPCON59197.2023.10434347.

[29] M. Chandra, P. K. Kushwaha, and S. Saxena, "Modified Fractal Carpets," in *2011 International Conference on Computational Intelligence and Communication Networks (CICN)*, Gwalior, India, 2011, pp. 537–540, doi: 10.1109/CICN.2011.115.

[30] B. Makkar *et al.*, "Map Reduce Concept-Based Sentiment Analysis Approach," *International Journal of Computer Sciences and Engineering*, vol. 7, no. 4, pp. 924–927, 2019.

[31] A. V. Srivastava, B. P. Lohani, P. K. Kushwaha, and S. Tyagi, "Dual-Layer Security and Access System to Prevent the Spread of COVID-19," in *Proceedings of International Conference on Machine Intelligence and Data Science Applications*, ser. Algorithms for Intelligent Systems. Singapore: Springer, 2021, doi: 10.1007/978-981-33-4087-9_28.

[32] P. K. Kushwaha, V. Singh, A. D. Gupta, D. K. Bhargava, and Meenakshi, "Explainable AI for hyper-local crop yield prediction in India," in *2026 2nd International Conference on Computing, Sciences and Communications (ICCSC 2026)*, 2026.

[33] P. K. Kushwaha, A. Srivastava, A. D. Gupta, M. Deshwal, and D. K. Singh, "AI-driven framework for automated browser security and threat detection," in *2026 2nd International Conference on Computing, Sciences and Communications (ICCSC 2026)*, 2026.

[34] N. K. Yadav, K. Meenakshi, and P. K. Kushwaha, "Design and implementation of a secure blockchain-enabled online voting system using multi-factor authentication," in *2026 4th IEEE International Conference on Power Electronics and IoT Applications in Renewable Energy and its Control (PARC 2026)*, 2026, pp. 223–227.

[35] Km Meenakshi, S. Negi, and P. K. Kushwaha, "Early detection of atrial fibrillation using real-time healthcare analytics," in *2025 IEEE International Conference on Modern Electronics Devices and Intelligent Communication Systems (MEDCOM 2025)*, 2025, pp. 106–110.

[36] P. K. Kushwaha, M. Divya Sree, D. K. Bhargava, and Mayank, "AI-powered safety & mental health companion for students," in *2025 International Conference on Emerging Technologies and Innovation for Sustainability (EmergIN 2025)*, 2025, pp. 724–730.

[37] P. K. Kushwaha, A. Srivastava, D. K. Bhargava, B. P. Lohani, and A. D. Gupta, "DeepfakeBench: A comprehensive benchmark of deepfake detection," in *2025 International Conference on Emerging Technologies and Innovation for Sustainability (EmergIN 2025)*, 2025, pp. 736–741.

[38] P. K. Kushwaha, A. Sharma, B. P. Lohani, A. D. Gupta, and D. K. Bhargava, "SUR SAMRAJYA: A youth-centric hybrid music player for low-cost

smartphones," in *2025 International Conference on Emerging Technologies and Innovation for Sustainability (EmergIN 2025)*, 2025, pp. 551–557.