



International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 15, Issue 3, March 2026

NLP-Based Question Generation Model for Technology Specific Mock Interviews

Mrs. B. Sreelatha¹, B.Bharath Kumar², U.Sai Teja³, E.Augusteen⁴, M.Sumanth⁵

Associate Professor, Department of CSE (Data Science), ACE Engineering College, Hyderabad, India¹

IV B.Tech, Department of CSE (Data Science), ACE Engineering College, Hyderabad, India²⁻⁵

ABSTRACT: The NLP-Based Question Generation System for Technology-Specific Mock Interviews aims to improve interview preparation by providing an intelligent and adaptive platform for generating interview-style questions. The system processes domain-specific learning material and automatically extracts key concepts using Natural Language Processing techniques such as tokenization, lemmatization, and keyword extraction. Using methods like TF-IDF and similarity modeling, the platform generates relevant interview questions without relying on predefined question banks.

The generated questions are organized through a domain classification module that categorizes them into technical areas such as programming languages and database systems. The system further performs performance-based difficulty adaptation, where question levels are dynamically adjusted based on the candidate's responses and evaluation scores. Additionally, answer evaluation is performed using techniques such as cosine similarity and keyword matching to measure the semantic closeness between candidate responses and expected answers. This system enables personalized mock interview sessions that simulate realistic interview environments and provide adaptive practice opportunities. By helping candidates identify knowledge gaps and improve technical readiness, the platform supports more effective preparation for real-world technical interviews.

I. INTRODUCTION

Technical interview preparation is essential for students and job seekers entering the technology industry. Traditional preparation methods rely on static question banks and manual practice, which lack adaptability and personalized feedback. As a result, candidates often struggle to identify their weak areas and improve their interview performance effectively. To address this issue, the proposed system introduces an NLP-based question generation model that automatically generates domain-specific interview questions from learning materials. The system also classifies questions by domain and adjusts difficulty based on user performance, providing personalized mock interview sessions for better preparation.

II. EXISTING SYSTEM

Existing interview preparation systems mainly depend on traditional methods and semi-automated digital platforms. In many cases, candidates practice through **face-to-face mock interviews** with mentors, teachers, or peers, which helps improve communication and confidence but requires human availability and consistent scheduling. With the growth of online platforms, several **AI-based interview tools** such as automated mock interview systems provide practice sessions using predefined question sets and offer feedback based on speech analysis, response timing, or keyword detection. Additionally, **online video platforms** like virtual meeting tools are widely used to conduct remote mock interviews.

However, most of these systems rely heavily on **static question banks or manually prepared interview questions**, which limits their ability to adapt to different technical domains and evolving industry requirements. These systems generally do not analyze learning materials to automatically generate new questions, and they provide limited personalization for individual users. As a result, candidates often receive the same set of questions repeatedly, reducing the effectiveness of practice and making it difficult to simulate realistic interview environments.

III. PROPOSED SYSTEM

The proposed system, NLP-Based Question Generation for Technology-Specific Mock Interviews, intelligently analyzes domain-specific learning materials to automatically generate interview-style questions using Natural Language Processing and Machine Learning techniques. tokenization, lemmatization, and keyword extraction. Generated questions are organized using a domain classification module that categorizes them into relevant technical areas such as programming languages and databases. The system also evaluates candidate responses using techniques like TF-IDF, cosine similarity, and keyword matching to measure the similarity between user answers and expected answers. Additionally, a difficulty adaptation mechanism analyzes user performance and dynamically adjusts the question level to provide personalized mock interview sessions, helping candidates identify knowledge gaps and improve their technical interview readiness.

ADVANTAGES OF THE PROPOSED SYSTEM

- **Automatic Question Generation:**

The system automatically generates interview-style questions from domain-specific learning materials using NLP techniques, reducing the dependency on static question banks.

- **Domain-Specific Question Classification:**

Generated questions are categorized into technical domains such as programming languages, databases, and other subjects, enabling focused interview preparation.

- **Adaptive Difficulty Mechanism:**

The system dynamically adjusts the difficulty level of questions (easy, medium, hard) based on the candidate's performance, creating a personalized learning experience.

- **Automated Answer Evaluation:**

Candidate responses are evaluated using techniques such as TF-IDF, cosine similarity, and keyword matching to measure the semantic similarity with expected answers.

- **Personalized Mock Interview Experience:**

By combining question generation, classification, and performance analysis, the system provides realistic and personalized mock interview sessions that help candidates identify knowledge gaps and improve interview readiness

IV. METHODOLOGY

4.1 Data Collection

The system collects domain-specific learning materials and interview-related textual data to generate meaningful interview questions. The dataset is created synthetically using predefined templates that simulate real-world interview questions and answers. The data includes fields such as Question_ID, Domain, Topic, Difficulty, Context, Keywords, Question, and Expected_Answer, which help organize and manage the interview preparation process.

4.2 Data Preprocessing

The collected textual data is cleaned and prepared using preprocessing techniques such as lowercasing, duplicate removal, punctuation removal, and text normalization. Additional NLP processes like stopword removal, tokenization, and lemmatization are applied to improve text quality and ensure consistency for further processing.

4.3 Feature Extraction

TF-IDF (Term Frequency–Inverse Document Frequency) vectorization is applied to convert textual data such as questions and answers into numerical feature vectors. This transformation enables the system to analyze the importance of words and perform similarity comparisons between candidate responses and expected answers.

4.4 Question Generation

Important concepts are extracted from the processed textual content using keyword extraction and NLP techniques. Based on these concepts, the system automatically generates interview-style questions, eliminating the dependency on predefined question banks and enabling dynamic question creation.

4.5 Domain Classification

The generated questions are categorized into their respective technical domains such as programming languages, databases, or other technology subjects. This classification helps organize the interview questions and enables users to practice within specific technical areas.

4.6 Answer Evaluation

Candidate responses are evaluated using cosine similarity and keyword matching techniques. The system compares the user's answer with the expected answer to determine the semantic similarity and calculate a relevance score for performance assessment.

4.7 Difficulty Adaptation

The system analyzes candidate performance and dynamically adjusts the difficulty level of questions (easy, medium, hard). This adaptive mechanism ensures that users receive questions that match their skill level and gradually improve their knowledge.

4.8 Mock Interview Simulation

The system combines generated questions, domain classification, and performance evaluation to deliver personalized mock interview sessions. This simulation helps candidates practice in a realistic interview environment and identify areas that require improvement.

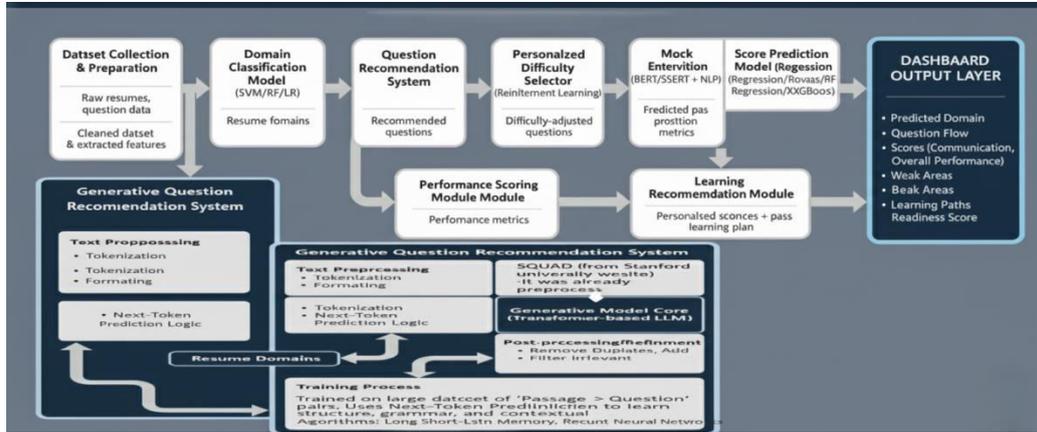
4.9 Visualization and Interface

The system uses dashboards to present interview questions, performance scores, and feedback in an interactive format. This allows users to easily navigate the platform, track their progress, and improve their interview preparation experience.

V. SYSTEM ARCHITECTURE

System architecture provides a structured overview of how the different components of the **NLP-Based Question Generation System for Technology-Specific Mock Interviews** interact to perform interview preparation tasks. The architecture consists of multiple modules including **data input, preprocessing, question generation, domain classification, answer evaluation, difficulty adaptation, and user interface components**. Domain-specific learning materials and interview datasets act as the primary input to the system. These inputs are processed through NLP-based preprocessing techniques such as tokenization, stopword removal, and lemmatization to prepare the text for analysis.

After preprocessing, the system extracts important concepts and generates interview-style questions automatically. The generated questions are categorized into technical domains to maintain structured interview preparation. Candidate responses are then evaluated using techniques such as **TF-IDF vectorization, cosine similarity, and keyword matching** to determine answer relevance. Based on the candidate's performance, the system dynamically adjusts the difficulty level of questions to provide an adaptive learning experience. Finally, the **Streamlit-based interface** allows users to interact with the system, attend mock interview sessions, view feedback, and track their performance, ensuring an efficient and personalized interview preparation environment.



VI. RESULTS AND OUTPUT

STEP : 1 User Login Authentication

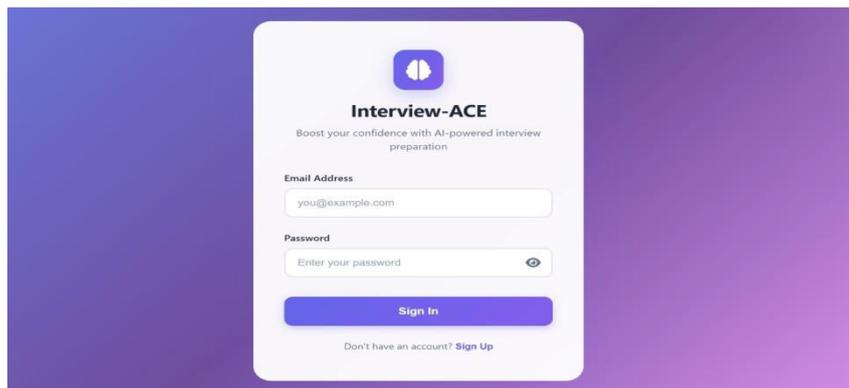


Fig: User Login Authentication

STEP : 2 Dashboard and Resume Upload.

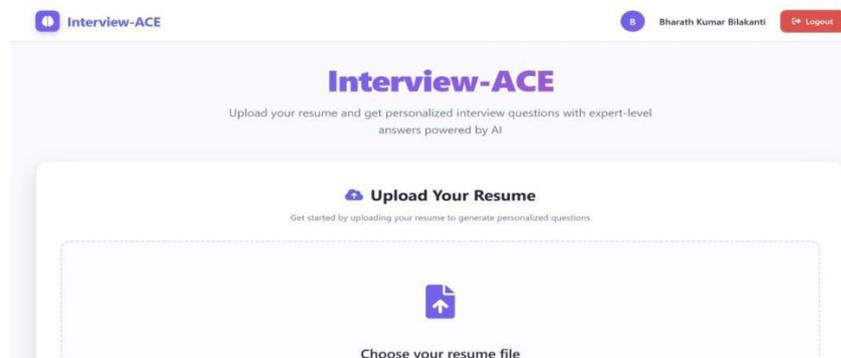


Fig: Dashboard and Resume Upload

STEP : 3 Manual Profile Completion

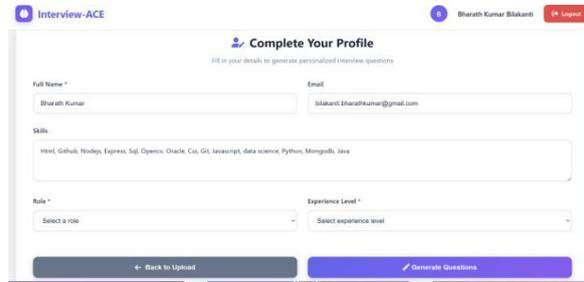


Fig: Manual Profile Completion

STEP : 4 Profile Summary Display



Fig: Profile Summary Display

STEP : 5 Live Interview Session

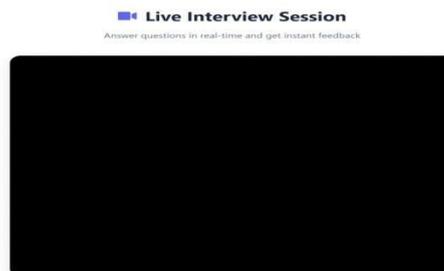


Fig: Live Interview Session

STEP : 6 Answer Evaluation.

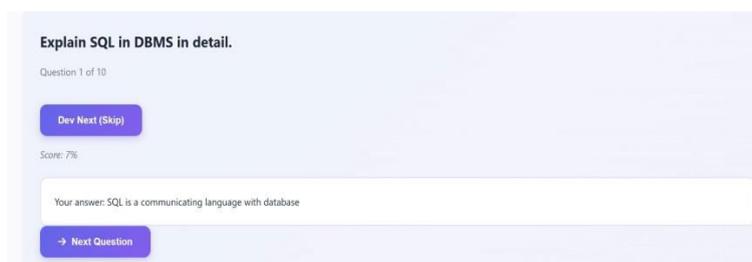


Fig: Answer Evaluation

STEP : 7 Interview Performance Report



Fig: Interview Performance Report

STEP : 8 Performance Visualization

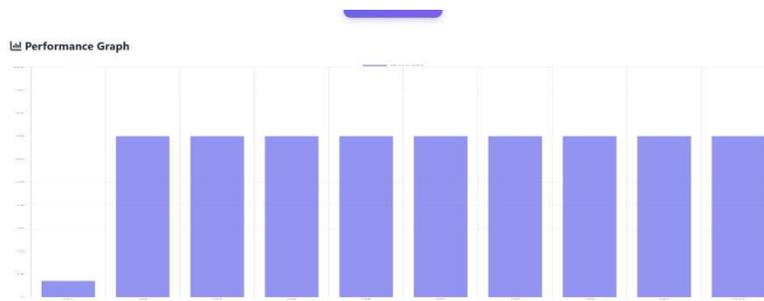


Fig: Performance Visualization

VII. CONCLUSION

The NLP-Based Question Generation System for Technology-Specific Mock Interviews demonstrates how Natural Language Processing and Machine Learning techniques can enhance interview preparation through intelligent automation. By processing domain-specific learning materials, the system automatically extracts key concepts and generates relevant interview-style questions without relying on static question banks. Techniques such as TF-IDF vectorization, cosine similarity, and keyword matching enable effective evaluation of candidate responses by measuring the semantic similarity between user answers and expected answers.

The system provides a structured and adaptive mock interview experience by categorizing questions into different technical domains and dynamically adjusting question difficulty based on user performance. This adaptive mechanism helps candidates identify their strengths and weaknesses while gradually improving their technical knowledge. Additionally, the use of an interactive interface allows users to practice interview questions, receive feedback, and track their performance effectively.

Overall, the proposed system offers a personalized and realistic interview preparation environment that supports candidates in improving their technical readiness for industry interviews. By combining automated question generation, intelligent answer evaluation, and performance-based difficulty adaptation, the system contributes to a more efficient and scalable approach to technical interview practice.

VIII. FUTURE SCOPE]

1. Integration of Deep Learning Models

The system can be enhanced by integrating advanced deep learning-based language models such as transformer architectures. These models can improve the accuracy of question generation by understanding complex technical concepts and contextual relationships within learning materials, enabling the generation of more context-aware and high-quality interview questions.

2. Integration of Real-Time Learning Resources

Future versions of the system can integrate real-time learning resources and larger datasets from online educational platforms, technical documentation, and industry interview repositories. This will help the system continuously update its knowledge base and generate questions based on the latest technologies and industry trends.

3. Support for Multiple Technical Domains

The system can be expanded to support a wider range of technical domains such as cloud computing, cybersecurity, data science, and artificial intelligence. This would allow candidates from different technical backgrounds to practice domain-specific interview questions.

4. Mobile Application Development

A mobile application version of the system can be developed to allow users to practice mock interviews anytime and anywhere. This will increase accessibility and improve the usability of the platform for students and job seekers.

5. Integration with Learning Platforms

The system can be integrated with online learning platforms and educational resources to recommend study materials based on the candidate's weak areas identified during mock interviews.

REFERENCES

- [1] Zhou, X., Li, J., & Ren, X. (2022). Segmentation-Based End-to-End Summarization for Long News Articles. <https://arxiv.org/abs/2205.07847>
- [2] Nayeem, M., & Chali, Y. (2022). NEWTS: A Topic-Focused Summarization Corpus for Topic-Aware Text Summarization. <https://arxiv.org/abs/2201.09987>
- [3] Poudel, P., et al. (2022). Multilingual News Classification using Fine-Tuned T5-SMALL and SVM Models. <https://arxiv.org/abs/2204.12811>
- [4] Kumar, A., & Gupta, V. (2023). A Comparative Study of News Categorization using Machine Learning and Transformer Models. <https://arxiv.org/abs/2302.06712>
- [5] El Mahdaoui, A., et al. (2023). News Genre Classification using Ensemble Transformers (RoT5-SMALLa + Longformer). <https://arxiv.org/abs/2310.03455>
- [6] Liu, Y., et al. (2019). RoT5-SMALLa: A Robustly Optimized T5-SMALL Pretraining Approach. <https://arxiv.org/abs/1907.11692>
- [7] Lewis, M., et al. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation and Summarization. <https://arxiv.org/abs/1910.13461>
- [8] Raffel, C., et al. (2020). T5: Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. <https://arxiv.org/abs/1910.10683>
- [9] Devlin, J., et al. (2019). T5-SMALL: Pre-Training of Deep Bidirectional Transformers for Language Understanding. <https://arxiv.org/abs/1810.04805>
- [10] KUMAR, P. ASHOK, GSatish KUMAR, and SETTI NARESH KUMAR. "Improve the Capacity of Uniform Embedding for Efficient JPEG Steganography Based on DCT." (2015).
- [11] Kumarr, P. Ashok, et al. "In-vestigating Context-Aware Sentiment Classification Using Machine Learning Algorithms." Accelerating Discoveries in Data Science and Artificial Intelligence 1. ICDSAI 2023. Springer Proceedings in Mathematics & Statistics. Vol. 421. 2024.
- [12] Sunkavalli, Jayaprakash, et al. "Experimentation Analysis of VQC and QSVM on Sentence Classification in Quantum Paradigm." In 2024 International Conference on Computing, Sciences and Communications (ICCSC), pp. 1-5. IEEE, 2024.
- [13] Kumar, P. Ashok. "Event Based Time Series Sentiment Trend Analysis." ResearchGate.
- [14] Pandi, Chiranjeevi, et al. "A Survey: Recommender System For Trustworthy."
- [15] Kumar, P. Ashok, B. Vishnu Vardhan, and Pandi Chiranjeevi. "Correction to: Investigating Context-Aware Sentiment Classification Using Machine Learning Algorithms." In XVIII International Conference on Data Science and Intelligent Analysis of Information, pp. C1-C1. Cham: Springer Nature Switzerland, 2023.
- [16] Supraja, Thatikonda, et al. "Lstm-based cloud workload prediction using pso-ga hybrid algorithm." Security Issues in Communication Devices, Networks and Computing Models. CRC Press, 2025. 44-52.
- [17] Sunkavalli, Jayaprakash, et al. "Experimentation Analysis of VQC and QSVM on Sentence Classification in Quantum Paradigm." 2024 International Conference on Computing, Sciences and Communications (ICCSC). IEEE, 2024.
- [18] Supraja, Thatikonda, et al. "Exploring IoT and 5G in Smart Healthcare: Applications, Challenges and Future Directions." CVR Journal of Science and Technology 28.1 (2025): 12-2



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details