STM JOURNALS

Journal of
Multimedia Technology &
Recent Advancements

ISSN: 2349-9028
Volume 12, Issue 1, 2025
January–April
DOI (Journal): 10.37591/JoMTRA

https://journals.stmjournals.com/jomtra

Review JoMTRA

Interview Preparation System Using AI

Shruti Patil[1], Smita Badarkhe[2,*], Neha Patil[3], Rutuj Nagargoje[4]

Abstract

The automated interview evaluation system leverages artificial intelligence (AI) and natural language processing (NLP) technologies to streamline and enhance the interview process. Through a web-based application built with the Flask framework, the system enables candidates to respond to dynamically loaded interview questions using both text and speech inputs. Questions, managed via a CSV (comma separated values) file for flexibility, are evaluated for similarity to expected answers using the rapid fuzz library, providing an objective assessment metric. The system employs Google's speech recognition API (application programming interface) to process spoken responses, ensuring accessibility and ease of use. Results, including average similarity scores, are presented to both candidates and interviewers, offering clear and actionable feedback. By integrating AI, NLP, and advanced web technologies, the system aims to provide a robust and user-friendly platform for consistent and fair interview evaluations, enhancing the overall efficiency and effectiveness of the interview process. This project presents an innovative interview system designed to enhance the efficiency and accuracy of evaluating candidates' responses in interview settings. Developed using the Flask web framework, this system leverages both textual and speech input to provide a versatile and accessible interview experience. The application consists of three primary components: the user interface, the backend processing, and the evaluation mechanism. The user interface, implemented with HTML and CSS (cascading style sheet), provides a seamless and intuitive platform for candidates to interact with. It includes pages for initiating the interview, answering questions, and viewing results. The backend, developed in Python with Flask, handles routing, data management, and speech recognition integration using the `speech recognition` library. Interview questions are dynamically loaded from a CSV file, ensuring flexibility and ease of updating content. Candidates can respond to interview questions either by typing their answers or by using speech input, which is processed using Google's speech recognition API. The system records these responses and evaluates them against expected answers using the "rapid fuzz" library, which calculates the similarity score between the provided and expected answers. This scoring mechanism provides an objective measure of the candidate's performance. The results page displays the final assessment, including the average similarity score, offering both the candidate and the interviewer a clear understanding of the interview outcomes. This system enhances the interview process by making it more efficient while ensuring all candidates are evaluated consistently and fairly. Overall, this project demonstrates the effective integration of web technologies and NLP tools to create a robust and user-friendly interview system. It stands as a testament to the potential of combining traditional web development with advanced speech and text processing capabilities to improve conventional processes.

***Author for Correspondence**
Smita Badarkhe
E-mail: smita.garde_skncoe@sinhgad.edu

[1,3,4]Student, Department of Electronics & Telecommunication, Smt. Kashibai Navale College of Engineering, Savitribai Phule Pune University, Pune, Maharashtra, India
[2]Assistant Professor, Department of Electronics & Telecommunication, Smt. Kashibai Navale College of Engineering, Savitribai Phule Pune University, Pune, Maharashtra, India

Received Date: July 03, 2024
Accepted Date: January 13, 2025
Published Date: March 07, 2025

Citation: Shruti Patil, Smita Badarkhe, Neha Patil, Rutuj Nagargoje. Interview Preparation System Using AI. Journal of Multimedia Technology & Recent Advancements. 2025; 12(1): 8–13p.

Keywords: Natural language processing (NLP), structured query language (SQL), artificial intelligence (AI), speech processing, interview

INTRODUCTION

In today's extremely competitive job market, securing your dream job often depends on your

ability to excel in interviews. However, mastering the art of interviews can be a difficult task. That is where this artificial intelligence (AI) interview preparation system steps in as a game-changer. Designed to empower job seekers and career enthusiasts alike, this platform uses the power of artificial intelligence to offer a new and different interview preparation experience. The AI interview preparation system uses machine learning algorithms to provide a personalized interview practice environment. With an extensive database of interview questions across various industries and roles, you can simulate real-world interview scenarios to perfection. The system's AI-driven feedback mechanism offers insightful, feedback on user responses, helping the user identify areas for improvement and refine his/her communication skills. This AI-based system is your trusted companion on the journey to interview success. This is made by using Python and SQL languages. The questions and answers are stored in database. Users can choose which topic they want to test their knowledge on topics like HR (human resources) or technological questions and accordingly the questions will be displayed. Then the user responses will be stored in database. After the test review will be given by judging the responses.

LITERATURE SURVEY

The paper titled "AI-Based Mock Interview Evaluator: An Emotion and Confidence Classifier Model" by Mandal et al. [1], which is also a base paper of this system presents a novel approach using convolutional neural networks (CNNs), natural language processing (NLP), and audio processing libraries like Pydub. The methodology involves assessing candidates during mock interviews through three key components: facial emotion recognition, speech-based confidence evaluation, and knowledge assessment via web scraping. The goal is to enhance interview preparation and reduce stress for candidates. However, the paper acknowledges potential implementation complexities and raises concerns about accuracy limitations in the proposed model. The research aims to contribute to the field of interview evaluation by leveraging AI technologies for a more comprehensive assessment of candidates.

The paper “Automated HR Interview System”, by Tadvalkar et al. [2] focuses on automating HR interviews. The methodology includes the naïve Bayes classifier for text classification, the Python speech recognition module, Google speech API (application programming interface), and CNNs. Additionally, support vector machines (SVMs) are used. This system streamlines the HR interview process using a combination of natural language understanding and speech recognition technologies.

The paper by Bagaria et al. [3] focuses on keyword spotting (KWS) and utilizes recurrent neural network (RNN) transducer and text-to-speech (TTS) technology. The methodology includes the use of the LibriSpeech database and the Speech_recognition library for TTS. This paper explores the adaptation of RNN transducer technology to enhance keyword spotting capabilities.

The paper by Su et al. [4] involves NLP and fuzzy string matching, with the use of the Scapy framework. The fuzzy wuzzy library is employed to improve the precision of answer evaluations.

Purohit et al. [5] use dialog state tracking (DST) in combination with deep learning techniques in their study. The methodology includes LSTM + ANN (long short-term memory + artificial neural network) and utilizes questions and answer data for training. Additionally, deep reinforcement learning (deep RL) is applied, along with the Keras framework and Scikit-learn 0.18, to enhance interview coaching.

The paper by Kumar et al. [6] presents an interviewing chatbot system based on NLP. It involves techniques such as JARO, sentiment analysis, and database management. Google Firebase, a NoSQL database, is employed to store and manage data for this chatbot system.

The paper by Suen et al. [7] focuses on keyword spotting (KWS) and utilizes RNN transducer and TTS technology. The methodology includes the use of the LibriSpeech database and the Speech recognition library for TTS. This paper explores the adaptation of RNN transducer technology to enhance keyword spotting capabilities.

PROPOSED SYSTEM

The proposed system follows the flow as in Figure 1. First, we must gather lots of question from trusted websites from internet. We take only question from those website that users use for preparation for interview. Then all these questions are stored in .csv (comma separated values) files. Then these questions are divided into two categories: HR and technical questions. We created a user interface where the users can register themselves on website. Then the users have to select which interview: either HR interview or technical interview. Once the type of interview is selected, the interview system displays questions that the users have to answer. The response of the user is recorded using speech recognition. Then the user response is compared with the expected answer in the database and the score is given after attempting all the questions.

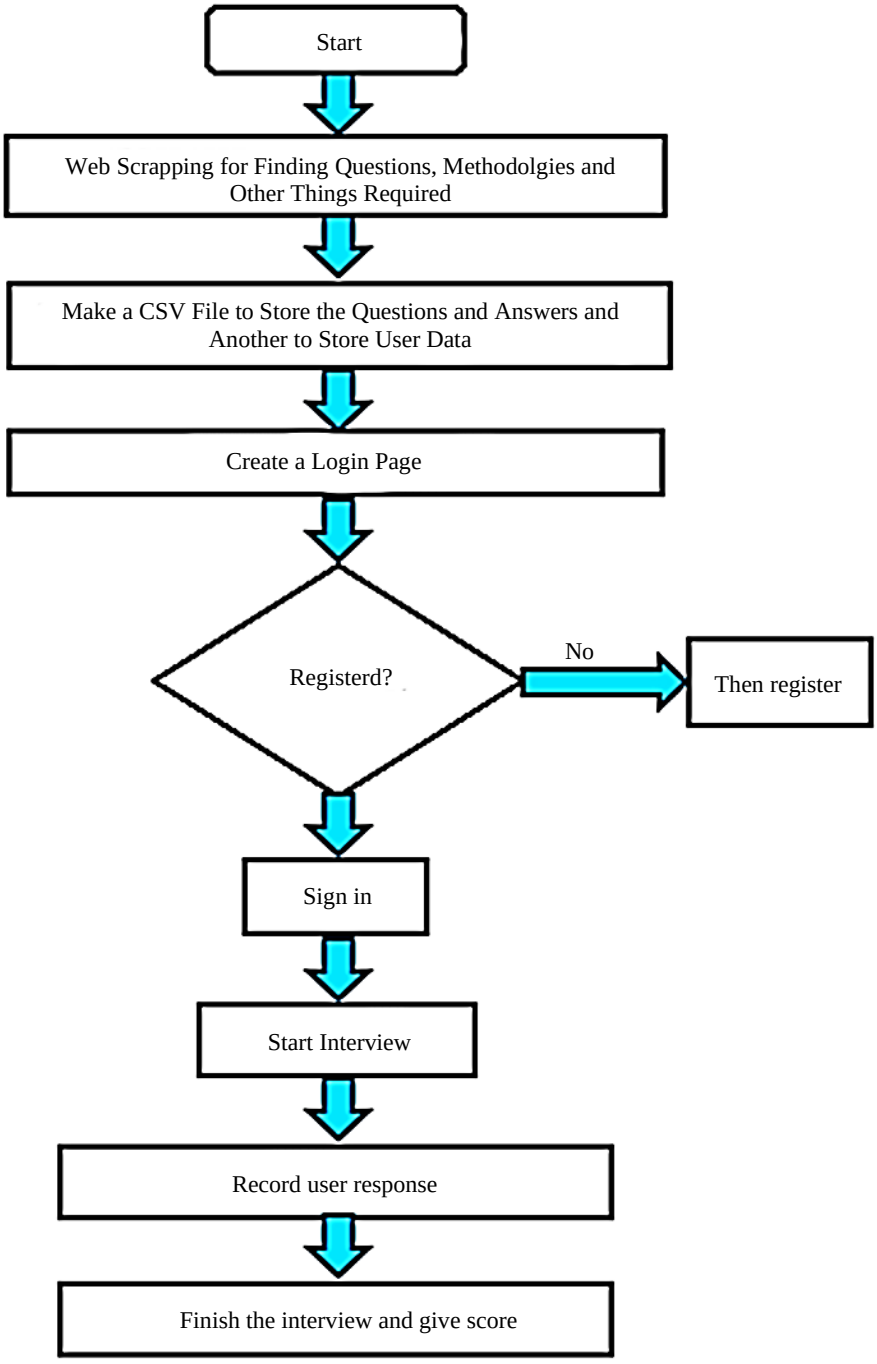

Figure 1. Flowchart of the proposed system.

Speech Recognition

The interview will commence with the presentation of a question, to which the candidate will respond orally. The Google Speech API and the Python Speech Recognition module will be used to identify the speech. The voice from the source will be heard by the listen (source, timeout) until the timeout expires. The Google Speech Recognizer will then identify the speech and translate it to text following the timeout! The goal of speech recognition, a crucial area of AI and NLP, is to translate spoken words into written language. The acoustic model, which examines the connection between sound patterns and linguistic units; the language model, which forecasts the probability of word sequences; and the pronunciation model, which manages variances in word pronunciations, are the main players in this process. Phases including preprocessing, feature extraction, acoustic modelling, decoding, and post-processing are involved in the speech recognition process. There are still issues, such as speech variability, terms outside of the lexicon, and context dependency, even with great progress. Contemporary methods make use of hybrid models, deep learning, and speaker variability adjustments. Applications such as transcription services and virtual assistants spur innovation in the direction of more precise and flexible systems.

Fuzzy String Matching Module

The Levenshtein distance is used in fuzzy string matching to determining the distinction between two sequences. The difference between two strings is computed using the Levenshtein distance. By calculating the least amount of changes required to equalize the strings. The score that is attained increases with fewer changes needed. The fuzzywuzzy Python library is used in the suggested technique to determine the estimated string-matching score. There are various functions in the fuzzywuzzy fuzz module for obtaining the partial string-matching ratio. The partial ratio obtained is used to compute the fuzzy score.

From Figure 2, we can say that from a structural perspective, it is evident that the expected answer and the user's answer are not exactly the same, but they both convey the same meaning.

Text to Speech

Text to speech is used in proposed system to give better experience to user. System use Pyttsx3, a flexible Python package. It is an essential part of adding text-to-speech functionality to these kinds of systems, making spoken instructions, question presentation, feedback, and accessibility features possible. With Pyttsx3, developers can give users an even more engaging and dynamic experience by asking questions aloud, giving verbal instructions regarding the interview procedure, and giving feedback on the users' answers. In addition, Pyttsx3 makes written material audible, making it possible for users with disabilities or vision impairments to interact with the system to the fullest extent possible.

Database

The database stores user responses and are fetched while showing result. SQL is used for integrity control and ease of data management. It allows multiple users to access the data whenever required. The increased redundancy, security, and storage makes SQL the best choice [8–10].

RESULTS

Figure 3 shows the home page of the website. For the questions that are displayed on screen as shown in Figure 4 a csv file is used which stores questions and their respective answers. Any question from that dataset is randomly displayed. Figure 5 shows a screen shot from a live video as the candidate gives a video interview.

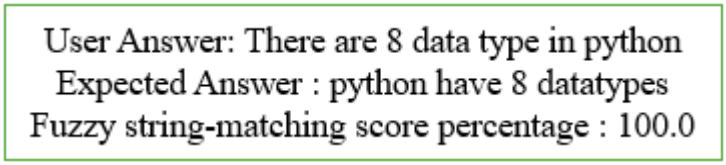

Figure 2. Fuzzy string matching score.

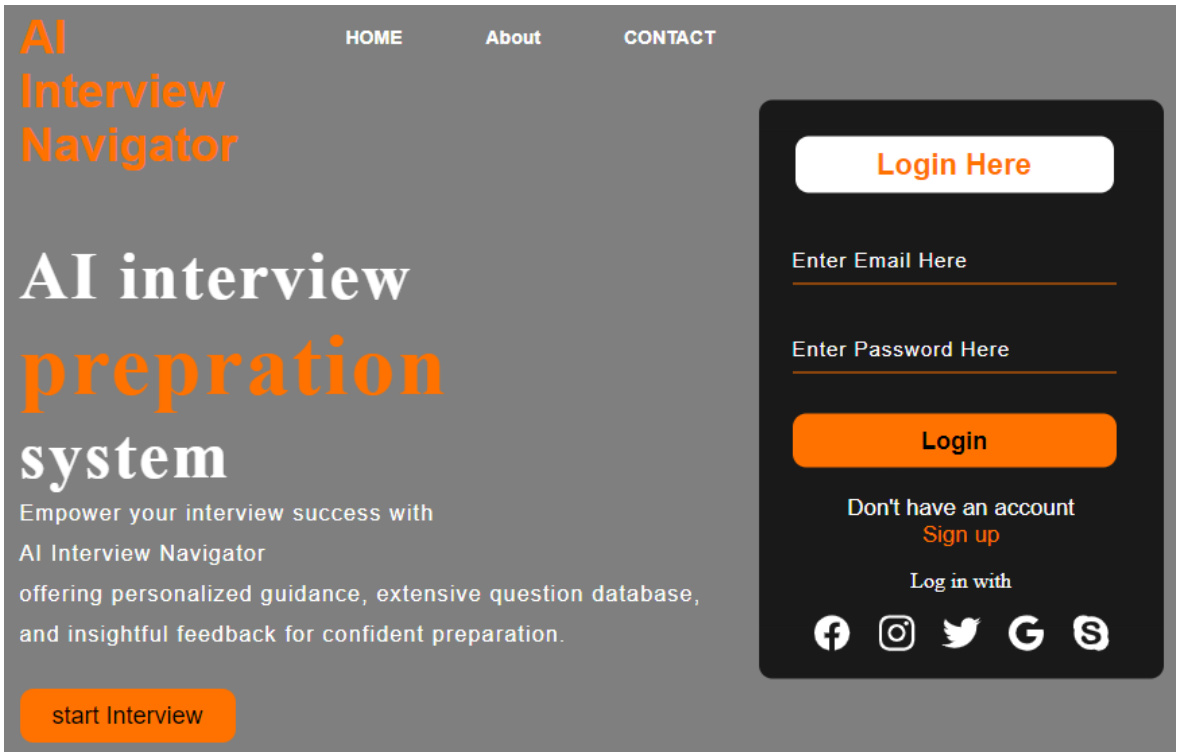

Figure 3. Home page.

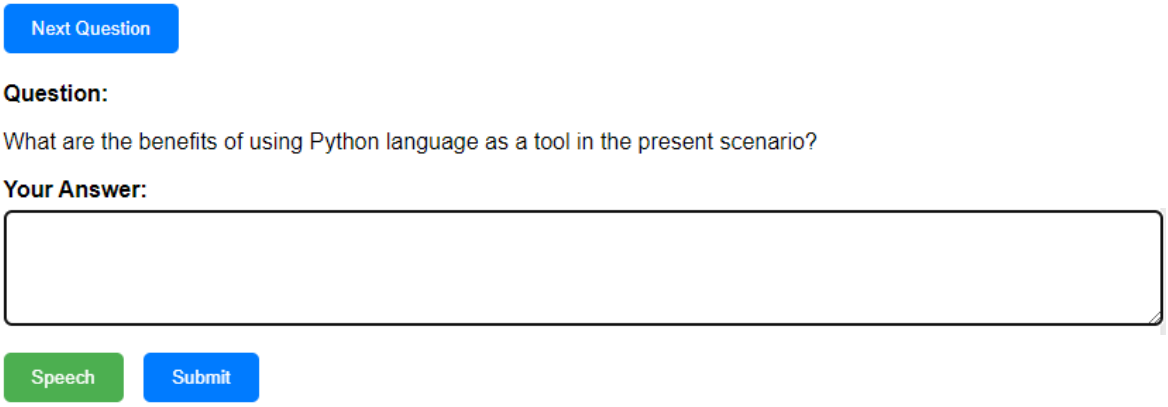

Figure 4. Interview interface.

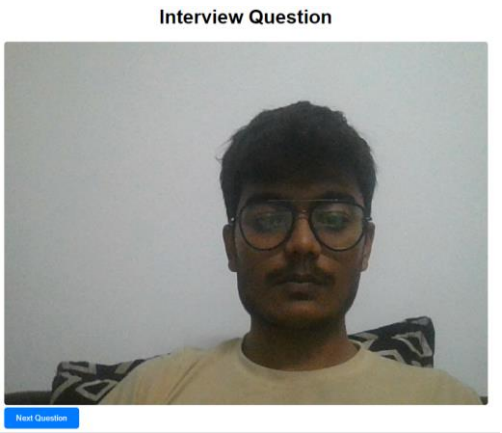

Figure 5. Screen shot from a live video.

CONCLUSION

In summary, there are important areas for ongoing improvement even if the project has made progress in data collection and interface development for question presentation. The use of machine learning models for dynamic feedback, the integration of a structured SQL database for effective data administration, and the application of NLP for question personalization should be the top priorities for future developments. These improvements will improve the system's capacity to produce customized questions, evaluate user answers, and deliver customized feedback, resulting in a more thorough and successful AI interview training experience.

Future Scope

AI interview preparation systems will have to update and adjust their material and algorithms on a regular basis to account for improvements in technology, job market trends, and interview habits. In order to monitor performance indicators, assess user input, and integrate new information and best practices into the system, this may entail utilizing machine learning techniques. Also, we can add face tracking and emotion analysis for users' emotional reactions.

REFERENCES

1. Mandal R, Lohar P, Patil D, Patil A, Wagh S. AI-based mock interview evaluator: an emotion and confidence classifier model. In: 2023 International Conference on Intelligent Systems for Communication, IoT and Security (ICISCoIS), Coimbatore, India, February 9–11, 2023. pp. 521–526.
2. Tadvalkar A, Ingale M, Chaudhari S, Chaudhari M. Automated HR interview system. Int Res J Eng Technol. 2021; 8 (5): 3645–3648.
3. Bagaria V, Badve M, Beldar M, Ghane S. An intelligent system for evaluation of descriptive answers. In: 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Thoothukudi, India, December 3–5, 2020. pp. 19–24.
4. Su MH, Huang KY, Yang TH, Lai KJ, Wu CH. Dialog state tracking and action selection using deep learning mechanism for interview coaching. In: 2016 International Conference on Asian Language Processing (IALP), Tainan, Taiwan, November 21–23, 2016. pp. 6–9.
5. Purohit J, Bagwe A, Mehta R, Mangaonkar O, George E. Natural language processing based Jaro – the interviewing chatbot. In: 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, March 27–29, 2019. pp. 134–136.
6. Kumar K, Kumar A, Abhishek K, Singh DK. Automation of HR interview system using JESS inference engine. In: 2014 Fourth International Conference on Communication Systems and Network Technologies, Bhopal, India, April 7–9, 2014. pp. 1102–1105.
7. Suen HY, Hung KE, Lin CL. TensorFlow-based automatic personality recognition used in asynchronous video interviews. IEEE Access. 2019; 7: 61018–61023.
8. Yang M, Tao Y, Cai H, Hu B. Behavioral information feedback with large language models for mental disorders: perspectives and insights. IEEE Trans Comput Soc Syst. 2024; 11 (3): 3026–3044.
9. Patil P, Patil S, Miniyar V, Bandal A. Subjective answer evaluation using machine learning. Int J Pure Appl Math. 2018; 118 (24): 1–3.
10. Ramamurthy M, Krishnamurthi I. Design and development of a framework for an automatic answer evaluation system based on similarity measures. J Intell Syst. 2017; 26 (2): 243–262.

