**ABSTRACT**

**Problem:** We are developing interview simulation software for DRDO’s RAC. The aim is to create a solution that ensures an unbiased and objective interview process to find the right talent. In a typical interview, an interviewer asks questions and evaluates the candidates' responses. Therefore, the questions must be relevant to the candidate's area of expertise, and the responses should directly relate to those questions. Finally, the system should help calculate an overall score for each candidate's subject knowledge and assess their suitability for the job they are applying for.

**Solution:** We have developed a solution to address the problem statement. The first step involves a Resume Analyzer, which retrieves the candidate's resume and identifies skills and domains relevant to the job. We also ask the candidate whether they are a fresher or applying for a promotion to determine the difficulty level of the questions.

The context is stored in a warehouse and classified based on difficulty, initiating the entire process.

The identified skills and domains are then sent to the Easy Phase, where each skill is assessed individually. For instance, if the skill is React, a model retrieves related context from a knowledge warehouse, generates easy questions, provides answers through another model, and evaluates the responses. If the candidate meets the threshold score, they advance to the Difficulty Phase, where follow-up questions are asked to create a more challenging Board Room experience.

After the Difficulty Phase, a report is generated that outlines the candidate's skills along with their evaluation scores, ultimately providing a knowledge percentage for each skill. The final output will be the overall percentage of skills in that domain.

**Background Process:** We use data warehousing to gather, store, and update the information that supports the question-and-answer model. This includes using web scraping to make sure the data stays current and useful.

**Evaluation:** First, we check how well the model performed. If the score is below a certain level, we use web scraping to find the correct answer to the question. If the web-sourced answer is accurate, we save the question-and-answer pairs along with the answer in the data warehouse to improve the Answering LLM model. This creates a system that continuously learns and improves to maintain high standards.