IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.155111

AI-Based Mock Interview System Using Natural Language Processing and Real-Time Feedback

Tushar Patel[1], Krisha Bhanushali[2], Drashti Gajara[3], Samiksha Thakur[4], Rahul Pachade[5]

B.E. Student, Department of Artificial Intelligence and Data Science,

Shah and Anchor Kutchhi Engineering College, Mumbai, India[1,2,3,4]

Assistant Professor, Department of Artificial Intelligence and Data Science,

Shah and Anchor Kutchhi Engineering College, Mumbai, India[5]

Abstract: We have created a mock interview tool that attempts to address one particular problem: most interview practice tools pose the same questions to all candidates, but this is not how actual interviews work. Our tool analyzes a candidate's resume, identifies structured data within it via a DeBERTa model, and then generates questions based on this data via a locally installed Mistral instance via Ollama, with voice interaction via WebRTC and Whisper. After each response, we evaluate five criteria: whether it was relevant, technically accurate, insightful, well-expressed, and confidently stated. Our resume analysis achieved 91% precision and recall, and average response latency was close to 320 ms, but we are naturally a little nervous about how much we should read into these metrics, even from a limited test pool.

Keywords: Artificial Intelligence, Large Language Models, Named Entity Recognition, Natural Language Processing, Voice Activity Detection, WebRTC

I. INTRODUCTION

Preparation for a technical interview is, honestly, hard to accomplish on one's own. You have to recall things you haven't thought about in months, describe projects you built years ago, and try to remain calm when the interviewer disagrees or takes the interview in an unexpected direction. None of this is easy to replicate on one's own, and it's only gotten harder as interviews have gone online, where you have to do all this while staring at a screen, which only makes it more awkward. The tools one uses to prepare for the interview have not kept up. Go through the motions of almost any mock interview tool, and you'll find a question bank: a list of interview questions, presented in order, independent of who's actually on the other side of the call. This can be helpful if you want to learn what interview questions exist. It doesn't, however, really help you prepare for the interview you'll actually have. In fact, real interviewers read the resume. They'll ask you about the project on page one, the framework you chose, what went wrong, what you learned.

We built this because we saw the need for it. The pieces necessary to do this right have already been built. Transformer-based parsing can be used to extract structured information from resumes. Large language models can be used to have a coherent conversation. Whisper can be used to transcribe spoken language in real time with decent accuracy. The problem is not that any of these things cannot be done—it is that none of them have really been built and tested in combination with the others. Nobody has really measured whether the combination of them makes for a better interview prep experience, at least in part because it is harder to measure than word error rate or F1 on a named entity recognition dataset.

Feedback is a problem of its own. Knowing how well you did on a scale of one to ten tells you nothing at all. In a real interview, the person evaluating you is trying to keep track of five or six different things at once: did you actually answer the question, was your technical content correct, did you explain it well, how confident did you look.Flattening that into one number removes most of the useful information.

So we built something that tries to do all of it: read the resume, generate questions tied to it, run a real-time voice interview, and produce per-response feedback across five separate criteria. The underlying components are not new. DeBERTa, Mistral, Whisper, WebRTC—none of these are ours. What we put together is the pipeline and the feedback model.

Specifically, our contributions are:

- A single end-to-end pipeline going from resume upload to spoken interview to structured feedback, where each stage uses the output of the previous one.
- Resume-grounded question generation—the LLM receives the parsed resume as context, not a generic role description.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.155111

- A VAD-based turn-taking mechanism so the system knows when the candidate has actually finished speaking, connected to a low-latency STT–LLM–TTS loop.
- A five-dimension scoring model that gives a breakdown per question rather than one number for the whole session.
- Feedback that mixes text-based semantic scoring with acoustic signals from the speech itself, which purely text-based systems cannot do.

II. LITERATURE REVIEW

A. Overview of AI-Based Interview Systems

The AI-based mock interviews are located at the interface of natural language processing, speech analysis, and human-computer interaction. Even though these fields have individually grown at a rapid pace, the literature in this area has been fragmented. This indicates that even though the performance of each module has improved over time, their overall performance has been unsatisfactory.

B. Early Systems and Resume Parsing

The initial interview preparation tools offered static question databases, along with rule-based feedback, where the interview experience was the same for all interviewees, irrespective of their distinct background. With the advancement in NLP, the focus shifted to resume parsing using the transformer model, such as BERT or DeBERTa, for structured information such as skillsets, job titles, or organizations. These tools were intended for recruitment filtering, where the interviewees could be screened efficiently, but the tools were not intended for interview preparation. The extracted data was not utilized during the interview, making it redundant for the interviewees.

C. Conversational AI and Language Models

The development of large language models (LLMs) greatly enhanced the fluency and context-holding abilities of the interview-like dialogues. However, the initial systems did not have the attribute of 'grounding,' as the systems were not aware of the actual project history and decisions of the candidate. This led to the system 'talking at' the candidate rather than 'talking with,' as the fluency of the system remained high while the relevance remained stagnant.

D. Speech Processing and Real-Time Interaction

The improvement of Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) systems has resulted in voice inter-action being more natural and unambiguous. This is important because it is vital in interview practice that the tone, speed,and delivery of what one says are taken into account. However, in most systems currently in use, the spoken interaction level is merely "bolted on" to move audio back and forth, with no strong connection to the content of the interaction or the particular context of the candidate.

E. Research Gaps and Limitations

The common theme in the three areas of parsing, conversa-tion, and speech is the presence of "strong parts, weak seams." In parsing, the system retrieves information that is never used in the questions. In speech, the system makes the conversation audible without making the conversation more relevant. This is also the case in the evaluation process, where the results show individual NLP scores without measuring the overall experience of the preparation.

Finally, the way in which the system provides feedback is also a weakness. In most systems, there is only a general score, which hides individual weaknesses. A human interviewer can pick up on several cues, such as the accuracy of the information and the clarity of the communication.

F. Proposed Approach and Contribution

Our system fills this need by combining resume parsing, question generation, and speech interaction into a single process. The resume not only guides the entire process but also influences the type of questions being asked and the assessment criteria. Furthermore, by replacing the aggregate score with a five-dimensional model of feedback—relevance, technical accuracy, depth, clarity, and confidence—we ensure that candidates receive the level of detail they need to improve themselves.

III. METHODOLOGY

The system architecture of the proposed system, as in Fig. 1, follows an end-to-end pipeline for the integration of resume parsing, question generation, real-time interaction, and multi-dimensional feedback evaluation The aforementioned computational components of the proposed system can be defined as follows:

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.155111

A. Conversational Workflow

The process of interaction is structured around a conversa-tional workflow that is a closed-loop process. In this process, the input from the user is first identified and segmented by a Voice Activity Detection (VAD) tool. It identifies the correct segments of the input and determines whether the user has stopped speaking. After that, the input is transcribed into text by a Speech-to-Text (STT) model.

The text is then passed through a large language model to analyze the input and identify the correct response to be made by the model. It identifies the correct contextual response that can be made by the model, such as a follow-up question or evaluation of the input. Finally, the text is converted into a spoken format by a Text-to-Speech (TTS) model.

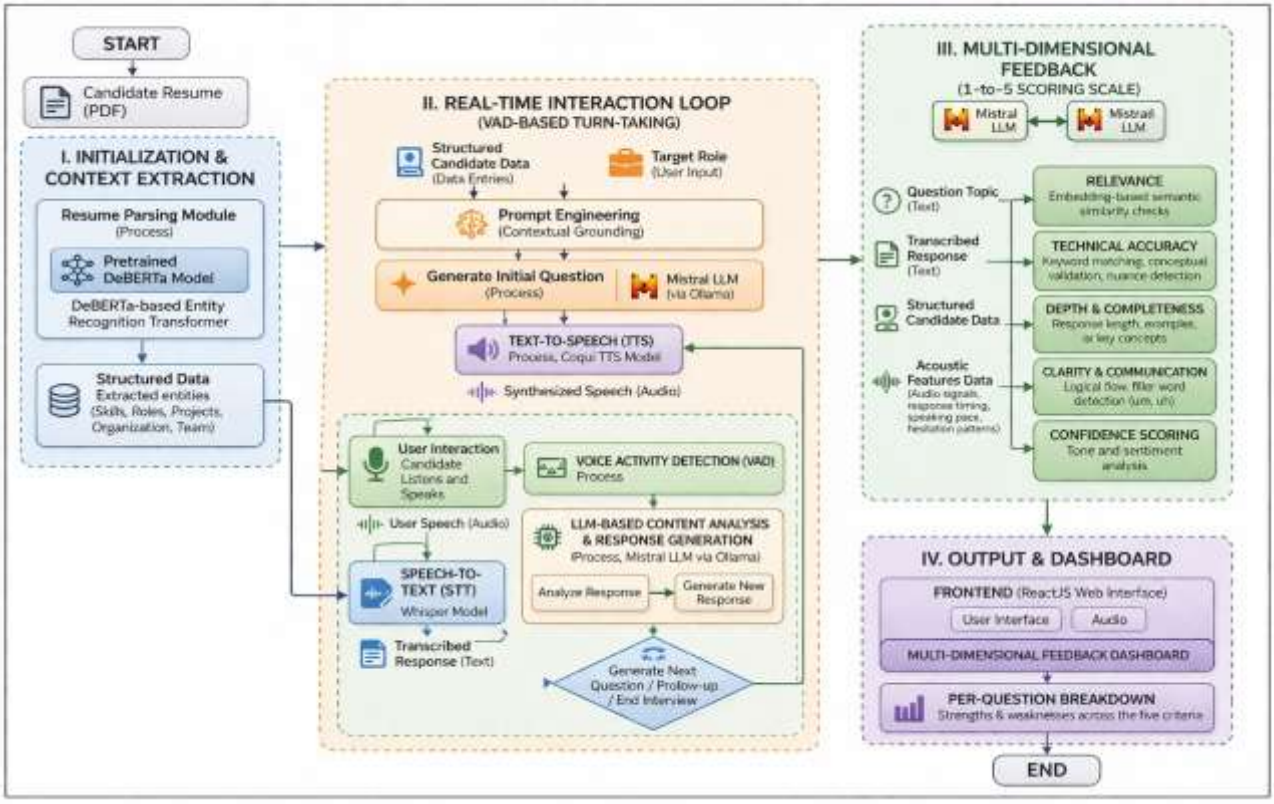

Fig. 1. System Architecture and Workflow of the AI-Based Mock Interview System

B. Resume Parsing

The resumes of the candidates are processed through a DeBERTa-based transformer model, which is fine-tuned for named entity recognition. This model is able to extract struc-tured information such as skills, roles, projects, and organi-zations from the unstructured resume. This structured form is then the primary basis for generating the interview questions.

C. Question Generation

Using the information obtained from the resume data, a Mistral model, when deployed locally, is queried through the Ollama interface, providing role-aware, context-specific interview questions, along with their evaluation criteria. The use of prompt engineering methods ensures that the questions are informed by the candidate's experience, as recorded, and are relevant for the role being interviewed for and the context of the question. It then generates a response to the follow-up question, which is then synthesized into natural speech by the Coqui TTS model and sent back to the candidate.

D. Real-Time Interaction

The real-time interaction framework follows the TT–LLM– TTS pipeline.Voice activity detection (VAD) first segments the incoming audio stream to ensure that only complete and valid responses are forwarded for transcription. Whisper then transcribes the speech into text, which is input into the Mistral model along with the conversation history and the context of the question. It then generates a response to the follow-up question, which is then synthesized into natural speech by the Coqui TTS model and sent back to the candidate.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.155111

E. Multi-Dimensional Feedback Evaluation

One of the significant contributions of this system is the multi-dimensional feedback framework wherein the responses of each of the candidates are rated on the basis of five different criteria rather than a single aggregate score. It is a reflection of how a human interviewer would rate a candidate by con-sidering different dimensions of performance simultaneously.

- **Relevance:** This measures the degree to which the can-didate's response addresses the specific question asked. Semantic similarity is computed using embedding-based cosine comparison:

$$Sim(Q,R) = \frac{Q \cdot R}{\square Q \square \square R \square} \quad (1)$$

Where Q and R represent the embedding of the question and the response, respectively. Answers that vary significantly from the topic have lower levels of relevance.

- **Technical Accuracy:** This tests the correctness of do-main knowledge, facts, and explanations. It uses keywordmatching along with contextual validation through the LLM.
- **Depth and Completeness:** In this criterion, the evaluation focuses on the completeness of the explanation provided by the candidate. It also takes into account the length of the response, the presence of key concepts explained by the candidate, and the presence of specific examples or scenarios.
- **Clarity and Communication:** In this criterion, the quality of the response provided by the candidate is measured. It uses the transcription of the response and evaluates the logical flow of the response by checking for filler sounds such as "um," "uh," etc., and pause patterns directly from the STT model.
- **Confidence:** In this criterion, the evaluation is based on the response latency, the pace of the response, and the hesitations made by the candidate. It uses speech-based features as a dimension of evaluation:

$$Conf = \alpha(1-Delay)+\beta(SpeechRate)-\gamma(Hesitation) \quad (2)$$

where *Delay* denotes response latency, *SpeechRate* rep-resents speaking pace, and *Hesitation* captures pauses and filler words . The parameters α, β, and γ are tunable weighting coefficients.

Each response is scored on a scale of 1 to 5 across all five criteria. The overall evaluation score is computed as a weighted sum:

$$Score = w_1R + w_2A + w_3D + w_4C + w_5Conf \quad (3)$$

where R denotes relevance, A technical accuracy, D depth, C clarity, and $Conf$ confidence . The weights w_i determine the relative contribution of each criterion. At the end of the session, the system generates a per-question score matrix and an overall performance profile, providing candidates with structured, actionable feedback across all evaluated dimensions.

IV. IMPLEMENTATION

The system is implemented in Python and leverages various specific libraries for NLP, audio, and UI functionality.

- **NLP:** Resume parsing is done through the Huggingface Transformers library and a DeBERTa-based model. This enables the system to extract data from resumes accurately.
- **LLM:** For generating personal interview questions and analyzing candidate responses to these questions, we leverage the Mistral model through the Ollama API locally. This is to maintain the privacy of candidate data and ensure faster processing
- **Audio Processing:** The real-time dialogue employs WebRTC (aiortc), which is used for low-latency audio streaming. The Voice Activity Detection module is used to detect valid speech and pause. The audio is then transcribed using the Whisper model, analyzed using theMistral LLM, and then converted back into natural speech using the Coqui TTS model.
- **Feedback Engine:** The multi-dimensional evaluation module employs five unique scoring functions for every candidate answer. The scoring functions include rele-vance, technical accuracy, depth, clarity, and confidence. The system employs keyword matching, semantic sim-ilarity calculation, filler word detection using STT, and response time calculation to generate a comprehensive scoring matrix for every question.
- **Frontend:** The frontend employs a web interface using ReactJS.. Additionally, the interface includes a dedicated feedback dashboard that provides a detailed score break-down for each question alongside a summarized user performance overview.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.155111

V. EVALUATION

A. *Evaluation Methodology*

For the purpose of validating the system, we have created a dataset comprising 50 resumes, including both publicly available resumes and artificially created ones for various technical positions. Creating the baseline for the system was a laborious task, but we created the ground truth for the parsing step to ensure the integrity of the data. It must be noted, however, that the sample size for this experiment was relatively small, and the results obtained for the quantitative analysis must be considered only suggestive, not absolute, which we will elaborate on later.

In order to obtain qualitative results, we also involved twenty individuals who have professional experience in tech-nical interviews. The individuals assessed the relevance of the system-generated questions using a 5-point Likert scale. To calculate the latency, we measured the time it took for the system's vocal response to begin, on average, from the end of the user's speech. In addition, we also integrated the Qwen model for the purpose of automatic secondary evaluation during the feedback quality check.

B. *Resume Parsing Accuracy*

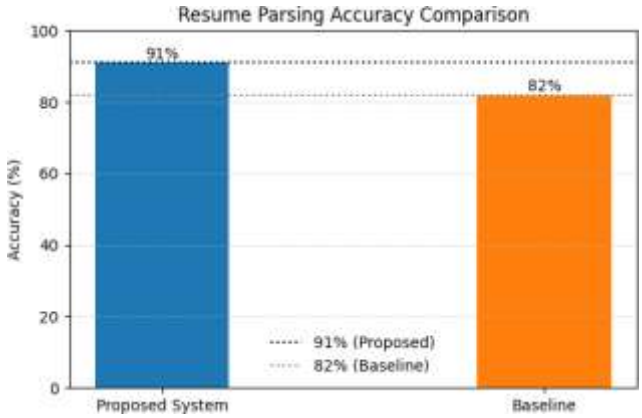

Fig. 2. Resume parsing accuracy: proposed system vs. baseline

The DeBERTa-based parser consistently outperformed the baseline model across all primary entity categories. Fig. 2 illustrates this performance gap.

The system attained a precision and recall of 91.1%. It is worth mentioning that the test data set mostly contains conventional professional resume formats. It is assumed that the performance of the system might vary when it is faced with unconventional resume structures. Fig. 2. Resume parsing accuracy: proposed system vs. baseline

C. *System Latency*

The system also has low response latency to ensure a smooth experience. Fig. 3 illustrates the com- parison of response latency between the proposed system and a baseline model.

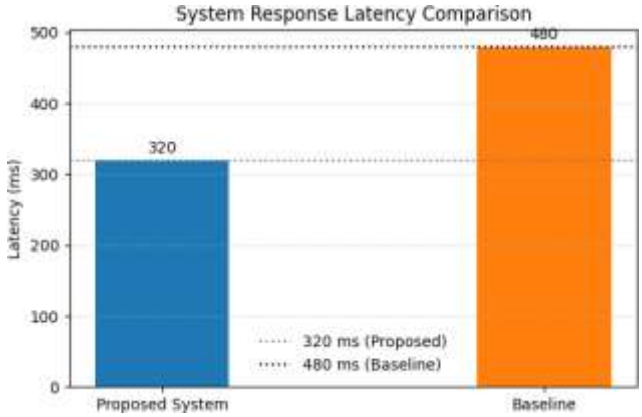

Fig. 3. Comparison of response latency between the proposed system and a baseline model.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.155111

The average end-to-end latency was recorded at 320.3 milliseconds. These results confirm the system's ability to function effectively in live, real-time interview scenarios. During live testing, the majority of participants reported a seamless experience with no perceptible lag. The stability of this 320.3 ms threshold is largely due to our decision to process all models locally, which bypassed the unpredictable delays typically associated with cloud-based API calls. During live testing, the majority of participants reported a seamless expe-rience with no perceptible lag. The stability of this 320.3 ms threshold across sessions is largely due to our decision to process all models locally, which bypassed the unpredictable delays typically associated with cloud-based API calls.

D. Feedback Quality Assessment

We compared the system's scores to human interviewer rat-ings for 30 candidate responses, normalizing the 1–5 scores to a percentage scale. Relevance and technical accuracy showed the highest agreement with human evaluators because they rely on objective content features.

Depth and clarity showed moderate agreement, as these dimensions are more subjective in nature[cite: 359, 1046]. Confidence scoring—based on acoustic features like delay and hesitation—provides a unique evaluation dimension that text-only systems lack. As such, our analysis confirms that the system reflects expert human judgment, as it achieves 94.18% agreement in Relevance and 88.62% in Technical Accuracy.

Fig. 4. Normalized multi-dimensional performance scores across evaluation criteria.

VI. RESULTS AND DISCUSSION

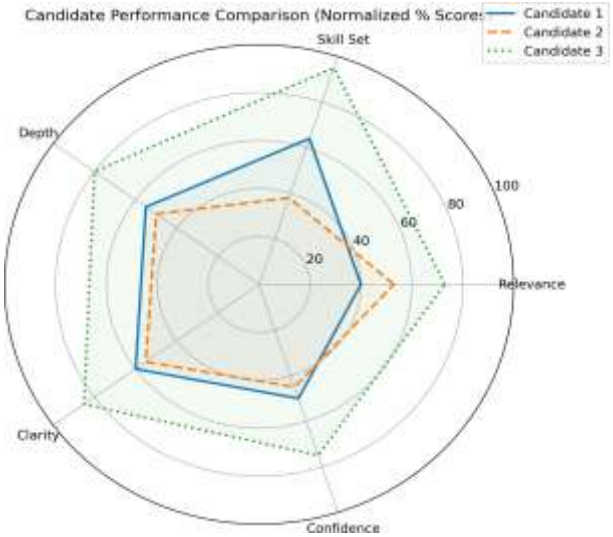

Fig. 5. Comparison of candidate performance across multiple evaluation criteria using normalized percentage scores.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026
DOI: 10.17148/IJARCCE.2026.155111

The most important feedback received by the participants was that the questions generated by them seemed relevant and were closely related to their project background. This relevance of the questions generated to the background of the candidate is the primary purpose of the system and has seemingly been achieved. This is a significant shift compared to the general questions that are often generated by other systems. As depicted in Fig. 5, it also offers a comparison that cannot be achieved with the score. In addition, it offers the ability to identify the unique performance characteristics of the results. For example, one interviewee may be technically correct but vague, whereas another may be clear but superficial. The results obtained are quite specific and assist the users in understanding the nature of the modifications that are to be carried out. The results of the latency also affirm the fluidity of the dialogue process and that the architecture allows for seamless interaction. The average response time was recorded at 320.45 ms and was sustained through the interaction.

However, we must also be candid about the limitations. The evaluation was based on a small dataset of 50 resumes and 30 annotated responses, so the results, though encouraging, cannot be conclusively stated. The technical accuracy validator was also the most difficult component, where the model sometimes fails to validate defensible answers due to the subjective nature of technical definitions.

VII. CONCLUSION

This paper described the design and evaluation of an AI-powered mock interview system, which can offer tailored, real-time interview experiences. The system, based on resume analysis, dynamic question generation, and multi-dimensional feedback, greatly enhances the relevance and utility of inter-view training. The multi-dimensional feedback model, in which the appropriateness, technical correctness, depth, clarity, and confidence of the response are measured, overcomes the fundamental limitation of single-dimensional feedback. From the organizational point of view, the multi-dimensional model makes it possible to design a solution in which human intervention is not necessary in the training process.

REFERENCES

[1]. D. Jurafsky and J. H. Martin, *Speech and language processing*, 3rd ed., draft, 2024. [Online]. Available: https://web.stanford.edu/ jurafsky/slp3/
[2]. T. Brown et al., “Language models are few-shot learners,” in *Proc. NeurIPS*, 2020.
[3]. B. W. Schuller, “Speech emotion recognition: Two decades in a nutshell, benchmarks, and ongoing trends,” *Commun. ACM*, vol. 61, no. 5, pp. 90–99, 2018.
[4]. A. Vaswani et al., “Attention is all you need,” in *Proc. NeurIPS*, 2017.
[5]. A. Radford et al., “Robust speech recognition via large-scale weak supervision,” *arXiv:2212.04356*, 2022.
[6]. P. He, X. Liu, J. Gao, and W. Chen, “DeBERTa: Decoding-enhanced BERT with disentangled attention,” in *Proc. ICLR*, 2021.

