

LLM 기반 ChatGPT를 활용한 컴퓨터 분야 면접 준비용 AI 모의 면접 시스템의 설계 및 구현에 대한 연구

A Study on the Design and Implementation of an AI Mock Interview System for Computer Science Interview Preparation Using LLM-based ChatGPT

천재성¹, 장희권², 김지혜³, 배창민⁴, 이동규⁵, 문일영^{1*}

¹한국기술교육대학교 컴퓨터공학과, ²인천대학교 컴퓨터공학부, ³중앙대학교 컴퓨터예술학부, ⁴순천대학교 멀티미디어공학과, ⁵강원대학교 컴퓨터공학과

Jae-Sung Chun¹, Hee-Kwon Jang², Ji-Hye Kim³, Chang-Min Bae⁴, Dong-Gyu Lee⁵, Il-Young Moon^{1*}

¹Department of Computer Engineering Korea University of Technology and Education, Cheonan 31253, Korea

²Department of Computer Engineering Incheon National University, Incheon 22012, Korea

³School of Computer Arts ChungAng University, Seoul 06974, Korea

⁴Department of Multimedia Engineering, Suncheon National University, Suncheon 57922, Korea

⁵Department of Computer Engineering and Engineering, Kangwon National University, Chuncheon 24341, Korea

[요약]

본 연구는 LLM(Large Language Model) 기반 ChatGPT를 활용하여 Computer Science(CS) 면접 준비를 위한 AI 모의 면접 시스템을 설계하고 구현하는 것을 목표로 한다. 이 시스템은 AI의 자연어 처리와 음성 인식 기능을 통해 면접자의 답변을 실시간으로 분석하고 피드백을 제공하여, 면접 준비 과정에서 학습자의 취약점을 개선할 수 있도록 지원한다. 설문조사 결과, 사용자의 90%가 실시간 피드백 기능이 면접 준비에 실질적인 도움을 주었다고 평가하였다. 주요 기능으로는 GPT 프롬프트 생성, 음성 데이터를 텍스트로 변환하는 Speech-to-Text 기능이 포함된다. 시스템은 응답 시간과 피드백의 정확성 면에서 긍정적인 평가를 받았으며, 향후 연구는 질문 유형의 확장과 다양한 산업 분야로의 적용 가능성을 모색할 예정이다.

[Abstract]

This study aims to design and implement an AI mock interview system for Computer Science (CS) interview preparation using LLM (Large Language Model) based ChatGPT. The system utilizes AI's natural language processing and speech recognition capabilities to analyze and provide real-time feedback on interview responses, helping users improve their weaknesses during the preparation process. According to a survey, 90% of users reported that the real-time feedback function provided substantial assistance in their interview preparation. Key features include GPT prompt generation and Speech-to-Text functionality, which converts voice data into text. The system received positive evaluations for its response time and feedback accuracy. Future research will explore expanding the range of question types and applying the system to various industries.

Key Words: AI interview system, ChatGPT, Computer Science interview preparation, Large Language Model (LLM), Real-time feedback

<http://dx.doi.org/10.14702/JPEE.2024.643>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 7 October 2024; Revised 14 October 2024

Accepted 21 October 2024

*Corresponding Author

E-mail: iymoon@koreatech.ac.kr

I. 서론

A. 연구 배경

Computer Science(CS) 분야의 면접은 개발자의 역량을 평가하는 중요한 요소로 자리잡고 있으며, 특히 대규모 IT 기업에서는 이론적 지식과 문제 해결 능력을 심도 있게 검증하는 과정이 필수적이다. 그러나 면접자의 준비 부족과 면접관의 주관적 평가로 인해 면접의 공정성에 문제가 발생할 수 있다. 최근 LLM(Large Language Model)과 같은 인공지능(AI) 기술의 발전으로, 면접자의 감정, 자세, 음성 및 텍스트를 분석하여 자동으로 면접 성과를 평가하고 면접자의 특성을 예측할 수 있는 AI 기반 모의 면접 시스템이 주목받고 있다. 이러한 시스템은 면접자의 성과를 정량적으로 평가하고, 면접 준비 과정에서 효과적인 피드백을 제공함으로써 면접자의 역량을 향상시키는 데 기여할 수 있다[1].

B. 연구 목적

본 연구의 목적은 LLM 기반 ChatGPT를 활용하여 CS 면접 준비를 지원하는 AI 모의 면접 시스템을 설계하고 구현하는 데 있다. 이 시스템은 AI의 자연어 처리 능력을 활용하여 면접자의 답변에 대해 실시간 피드백을 제공하고, 그에 따른 학습 효과를 증대시키는 것을 목표로 한다. 특히, 면접에서 자주 다루지는 이론적 질문과 문제 해결 능력을 평가하여 보다 효율적인 면접 준비 과정을 제공한다.

C. 연구 의의

기존의 면접 준비 시스템은 객관적인 피드백 제공이 부족하고, 면접자가 자신의 약점을 명확히 파악하기 어려운 점이 있다. 본 연구에서 제안하는 LLM 기반 AI 모의 면접 시스템은 이를 보완하여, 면접자가 실전 면접과 유사한 환경에서 학습할 수 있도록 지원하며, ChatGPT는 피드백의 신뢰성을 향상시키고, 면접자의 성과를 정량적으로 분석하여 효과적인 학습을 지원함으로써 면접 준비 과정에서의 개선점을 발견할 수 있도록 돕는다[2].

II. 이론적 배경 및 관련 연구

A. LLM과 ChatGPT의 개요

대규모 언어 모델(LLM)은 방대한 양의 텍스트 데이터를

학습하여 다양한 주제에 대해 자연스럽고 유창한 답변을 생성할 수 있는 인공지능 모델로, 자연어 처리(NLP) 분야에서 뛰어난 성능을 발휘한다[3]. 이러한 LLM은 텍스트 생성, 질의응답, 문서 요약 등 다양한 응용 분야에 활용될 수 있다. 특히, ChatGPT는 LLM의 대표적인 활용 사례로, 사람과의 대화에서 논리적이고 일관된 답변을 제공하는 능력을 가지고 있다. 이 모델은 사전 훈련과정을 통해 문맥을 이해하고, 사용자의 요구에 맞춘 응답을 생성할 수 있도록 설계되었다. 이러한 특성을 통해 ChatGPT는 질의응답뿐만 아니라 글쓰기 학습 및 교육, 기술적 질문에 대한 피드백 제공 등 다양한 교육적 활용 가능성을 제시한다[4]. 본 연구에서는 ChatGPT의 질의응답 기능을 활용하여 컴퓨터 과학(CS) 면접에서 자주 출제되는 기술적 질문에 대해 심층적이고 적절한 피드백을 제공하는 시스템을 개발하고자 한다.

B. AI를 활용한 면접 준비 시스템의 발전

최근 AI 기술, 특히 자연어 처리(NLP)와 음성 인식 기술의 발전은 면접 준비에 혁신적인 변화를 가져왔다. AI 면접 시스템은 웹 기반 비디오 채용 인터뷰를 통해 지원자의 답변 내용을 분석하고, 지원자의 성향, 역량, 직무 적합도 등을 다각적으로 평가하여 면접자의 태도와 특성을 파악할 수 있다[5]. LLM 기반 AI는 면접자의 답변을 실시간으로 분석하고, 그에 따른 적절한 피드백을 제공한다. 특히, Computer Science 면접에서 요구되는 복잡한 문제 해결 능력을 평가할 수 있도록 ChatGPT와 같은 AI가 학습된 데이터를 바탕으로 정확한 답변을 생성하는 방식으로 사용된다.

C. 기존 모의 면접 시스템의 한계

기존의 모의 면접 시스템은 주로 고정된 질문을 제공하고, 사용자가 스스로 답변을 작성하는 방식으로 작동하였다. 이 과정에서는 피드백 제공이 제한적이었고, 면접자가 자신의 취약점을 파악하기 어려운 문제가 있었다. 특히 실시간 피드백 기능의 부재는 학습 효과를 제한하는 요소로 작용하였다. 본 연구에서 제안하는 시스템은 이러한 한계를 극복하고, 면접 준비 과정에서 실시간으로 학습 효과를 증대시키는 것을 목표로 한다.

III. 시스템 설계 및 구현

A. 시스템 아키텍처

본 시스템은 LLM 기반 ChatGPT API를 활용한 AI 모의

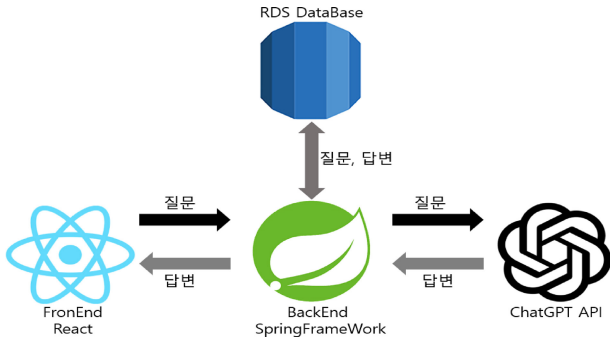


그림 1. 시스템 구성도
Fig. 1. System Architecture Diagram.

면접 시스템으로, 시스템 구성도는 그림 1과 같다. 사용자 인터페이스는 React를 기반으로 설계되었으며, React는 컴포넌트 기반 구조와 빠른 렌더링 성능을 제공하여 실시간 피드백과 같은 사용자 경험을 최적화할 수 있다. 특히, React는 가상 DOM(Virtual DOM)을 사용하여 상태 변화에 따라 선택적으로 UI를 렌더링함으로써 어플리케이션의 성능을 크게 향상시킨다. 이러한 방식은 컴포넌트 간 상태 변화로 인한 불필요한 전체 UI 업데이트를 줄여 성능을 최적화할 수 있으며, 코드의 재사용성과 유지보수성을 증가시킨다[6]. 또한, React의 풍부한 생태계를 통해 다양한 기능을 쉽게 통합할 수 있어 복잡한 인터페이스 설계에도 적합하다. 백엔드는 Spring Framework를 사용하여 서버 측에서 면접 데이터를 처리하는데, Spring은 대용량 애플리케이션의 복잡성 문제를 해결하기 위해서 만들어졌다[7]. 확장성과 안정성 면에서 대규모 애플리케이션에 적합하며, 다양한 내장 기능과 보안 지원을 통해 효율적인 개발 환경을 제공한다. 데이터 저장소로는 RDS(Relational Database Service)를 사용하여 질문 및 답변

데이터를 관리하며, RDS는 자동 관리 기능과 높은 확장성을 제공하여 안정적인 데이터 운영을 지원한다. 이를 통해 면접 시스템의 성능과 보안성이 보장된다[8].

B. AI를 활용한 주요 기능

1) 연습 모드: 사용자가 시간 제한 없이 질문에 답변하고, LLM 기반 AI를 통해 실시간 피드백을 받을 수 있는 기능이다. AI는 사용자의 답변을 즉시 분석하여, 답변의 정확성, 논리적 흐름, 그리고 보완할 부분을 평가한다. 이는 사용자가 자신의 약점을 즉시 파악하고 개선할 수 있도록 돕는다.

2) 실전 모드: 제한된 시간 안에 답변을 작성하며, 면접이 끝난 후에만 피드백을 확인할 수 있다. 실제 면접 상황과 유사한 환경을 제공하여 면접 대비 능력을 강화한다. 이 과정에서 AI는 면접자의 답변을 저장하고, 이후에 상세한 피드백을 제공한다.

3) GPT 프롬프트 구성

GPT 프롬프트는 AI가 면접자의 질문과 답변을 이해하고 적절한 반응을 생성하는 핵심 요소이다. 면접자가 답변을 완료하면, AI는 학습된 LLM 데이터를 활용해 답변의 질을 평가한다. 예를 들어, 그림 2-1과 같이 면접자의 답변을 바탕으로 GPT 모델이 답변의 정확성을 평가하고, 틀린 부분을 지적한다. 이를 위해, AI는 면접 질문과 답변 내용을 입력으로 받아 분석한 후, 피드백을 생성한다. 실제 서비스 화면에서 그림 2-2와 같이 “객체지향프로그램(OOP)에 대해서 설명해주세요”라는 질문에 대한 답변은 AI가 학습한 내용을 바탕으로 분석되고, 이에 대한 적절한 피드백이 제공된다.

4) Speech-to-Text 기능

음성 입력을 텍스트로 변환하기 위해 OpenAI의 Whisper API를 활용하여 면접자의 답변을 실시간으로 텍스트로 변환

```

const askQuestion = async (memberAnswer) => {
  try {
    const response = await fetch('https://api.openai.com/v1/chat/completions', {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json',
        'Authorization': 'Bearer ${apiKey}'
      },
      body: JSON.stringify({
        model: 'gpt-4o-mini',
        messages: [{
          role: 'user',
          content: `당신은 개발자 면접관이며, 다음 질문에 대해 면접자가 제대로 답변했는지 체크해야 합니다.\n
          다음 면접 질문과 답변에 대해 면접 답변이 정답인지, 틀렸다면 어느 부분이 틀렸는지 답변해주세요. 코드는 출력하지 않습니다.\n
          그리고 어느 부분이 틀린지 지적해주세요.\n
          면접질문: ${question}, 면접답변: ${memberAnswer}`
        }],
      })
    })
  } catch (error) {
    console.log(error);
  }
};
    
```

그림 2-1. ChatGPT API와 연동된 질문 평가 요청 코드
Fig. 2-1. Question Evaluation Request Code Integrated with ChatGPT API.

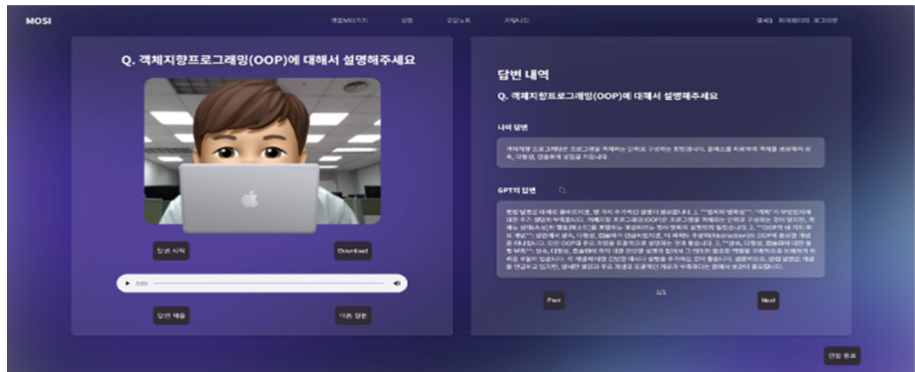


그림 2-2. AI 모의 면접 시스템의 실제 서비스 화면
Fig. 2-2. Actual Service Screen of the AI Mock Interview System.

```
useEffect(() => {
  async function getMicrophoneAccess() {
    try {
      const stream = await navigator.mediaDevices.getUserMedia({ audio: true });
      const mediaRecorder = new MediaRecorder(stream);
      let audioChunks = [];

      mediaRecorder.addEventListener('dataavailable', (event) => {
        if (event.data.size > 0) {
          audioChunks.push(event.data);
        }
      });
      mediaRecorder.addEventListener('stop', () => {
        const blob = new Blob(audioChunks, { type: 'audio/wav' });
        const audioUrl = URL.createObjectURL(blob);
        setAudioURL(audioUrl);
        setAudioBlob(blob);
        audioChunks = [];
        setWait(true);
      });
      setRecorder(mediaRecorder);
      mediaRecorder.start();
      setRecording(true);
    } catch (error) {
      console.error('mediaRecorder: $ {error}');
      setModalText('마이크를 인식할 수 없습니다. 면접을 진행할 수 없습니다. ');
      setModalOnClick(() => () => {
        setModalOpen(false);
        navigate('/interview/choice-stack');
      });
      setModalOpen(true);
    }
  }
});
```

그림 3-1. AI 모의 면접 시스템에서 음성 데이터를 수집 및 처리하는 코드
Fig. 3-1. Code for Collecting and Processing Audio Data in the AI Mock Interview System.

한다. Whisper API는 OpenAI에서 개발한 고급 음성 인식 시스템으로, 다양한 음성을 텍스트로 변환하며, AI가 음성 데이터를 처리하고 분석하는 데 필수적인 역할을 한다[9]. 이를 구현하기 위해 사용자는 브라우저의 마이크를 통해 음성을 입력하게 되며, 해당 음성은 Blob 형식으로 저장된다. Blob(Binary Large Object)은 이미지, 오디오, 비디오와 같은 대형 바이너리 데이터를 표현하는 객체로, 파일 업로드나 미디어 처리에 주로 사용된다[10]. 이 과정은 브라우저의 마이크 입력을 통해 음성을 녹음하고, 녹음된 음성을 Blob 형식으로 저장하여 Whisper API로 전송함으로써 텍스트로 변환하는 방식으로 이루어진다. 이러한 구현 방식은 그림 3-1에서

```
try {
  const formData = new FormData();
  formData.append('file', audioBlob, 'audio.wav');
  formData.append('model', 'whisper-1');
  const transcriptionResponse = await fetch('https://api.openai.com/v1/audio/transcriptions', {
    method: 'POST',
    headers: {
      'Authorization': 'Bearer ${apiKey}'
    },
    body: formData
  });
  if (!transcriptionResponse.ok) {
    const errorDetail = await transcriptionResponse.json();
    throw new Error('Error transcribing audio: ${transcriptionResponse.statusText}, ${JSON.stringify(errorDetail)}');
  }
  const result = await transcriptionResponse.json();
  return result.text;
} catch (error) {
  throw error;
};
```

그림 3-2. Whisper API를 활용한 음성 데이터를 텍스트로 변환하는 코드
Fig. 3-2. Code for Converting Audio Data to Text Using Whisper API.

설명된 흐름도와 같이 단계별로 처리된다. 이후, 저장된 Blob 형식의 음성 데이터를 FormData 객체로 구성된 후, Whisper API로 전송하여 텍스트로 변환한다. Whisper API 요청에는 file 필드와 model 필드가 포함되며, file 필드에는 녹음된 오디오 데이터가, model 필드에는 Whisper 모델의 이름이 지정된다. Whisper API로 전송된 오디오 데이터는 텍스트로 변환된 후, 이를 기반으로 면접자의 답변을 분석하게 된다. 이러한 음성 데이터를 Whisper API에 전송하여 텍스트로 변환하는 과정은 그림 3-2와 같은 흐름도로 이루어진다.

녹음된 음성 데이터가 Whisper API를 통해 텍스트로 변환되는 과정을 통해, AI 면접 시스템은 사용자의 음성 응답을 텍스트 데이터로 분석하고 피드백을 제공할 수 있다. 전체 과정에서 음성 데이터 수집과 변환은 각각 그림 3-1과 그림 3-2의 구현 방식을 통해 이루어지며, 이를 통해 면접자의 음성 응답

을 효과적으로 처리하고 분석할 수 있다.

C. 시스템 구현 환경

본 연구에서 제안한 AI 모의 면접 시스템을 구현하기 위해 다양한 개발 환경과 도구를 사용하였다. 아래 표 1은 시스템 구현에 사용된 주요 도구 및 환경을 요약한 것이다.

IV. 성능 평가 및 실험 결과

본 연구에서 개발된 AI 모의 면접 시스템의 성능은 10명의 Computer Science 면접 준비자를 대상으로 평가되었다. 실험은 연습 모드와 실전 모드로 나누어 이루어졌으며, 각 모드에서 사용자는 동일한 면접 질문에 답변하고 실시간 피드백을 제공받았다. 시스템 성능 평가는 응답 시간, 피드백의 정확성, 사용자 만족도를 중심으로 이루어졌다.

A. 응답 시간 측정 방법 및 결과

응답 시간은 LLM 기반 ChatGPT가 사용자 입력(질문 및 답변)에 대해 피드백을 생성하는 데 걸리는 시간을 측정하였다. 실험은 총 100회의 질문과 답변을 기반으로 수행되었으며, 각 응답에 걸리는 시간을 밀리초(ms) 단위로 기록하였다. 그림 4-1은 사용자 입력을 기반으로 피드백을 생성하는 데 걸린 시간을 측정하고 기록하는 코드 예제를 보여준다. 이 코드는 응답 시작 시간과 종료 시간을 비교하여 응답에 소요된 시간을 밀리초 단위로 계산하고, 그 결과를 출력하는 기

```
const handleGrade = async () => {
  try {
    setBtnBlock(true);
    const memberAnswer = await transcribeAudio();
    const gptAnswer = await askQuestion(memberAnswer);
    const endTime = new Date();
    const responseTime = endTime - startTime;
    console.log('사용자 ${userNumber} 질문 {questionId} 응답 시간: ${responseTime} ms');
    const newAnswer = {
      questionId: questionId,
      question: question,
      memberAnswer: memberAnswer,
      gptAnswer: gptAnswer
    };
    setQuestionAnswers(prevAnswers => {
      const updatedAnswers = [...prevAnswers, newAnswer];

      localStorage.setItem("answers", JSON.stringify(updatedAnswers));
      const newCount = updatedAnswers.length;

      localStorage.setItem("answersCount", newCount.toString());
      setCurrentPage(newCount - 1);
      return updatedAnswers;
    });
    await insertAnswer(memberAnswer, gptAnswer);
    setBtnBlock(false);
  }
}
```

그림 4-1. 응답시간 측정을 위한 코드 예제

Fig. 4-1. Example Code for Measuring Response Time.

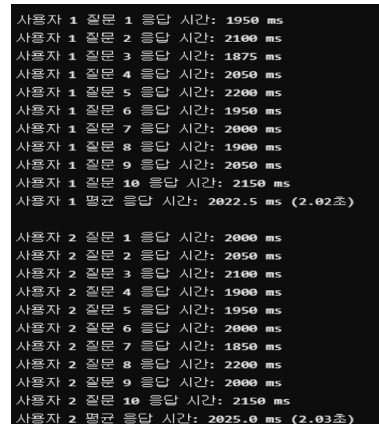


그림 4-2. 응답시간 측정 결과

Fig. 4-2. Response Time Measurement Results.

표 1. 시스템 구현 환경

Table 1. System Implementation Environment

| 항목 | 도구/자료 | 설명 |
|-------|------------------------|-------------------------------------|
| 입력 장치 | 웹캠, 마이크 | 사용자 음성과 영상을 입력받기 위한 장치. |
| 서버 비용 | AWS EC2 | 시스템의 서버를 운영하기 위한 클라우드 기반 서버 인프라. |
| API비용 | ChatGPT API | 면접 질문 생성 및 피드백 제공을 위한 비용. |
| 개발도구 | IntelliJ IDEA Ultimate | 프로젝트 구현을 위한 통합 개발 환경(IDE). |
| 버전관리 | Git, GitHub | 코드의 버전 관리를 위한 도구. |
| 언어 | Java 17 | 백엔드 로직 구현을 위한 언어. |
| 빌드도구 | Maven | 자바 프로젝트의 빌드 및 관리 도구. |
| 프레임워크 | Spring Boot 3.3.0 | 백엔드 프레임워크로 확장성과 안정성을 제공. |
| 테스트도구 | JUnit | 유닛 테스트를 수행하여 각 컴포넌트의 기능과 정확성을 검증. |
| 자동화도구 | GitHub Actions | 자동 빌드 및 테스트 프로세스를 구현하여 코드 안정성을 보장. |
| 배포환경 | AWS EC2 | 서버 배포 및 운영을 위한 클라우드 서비스. |
| 빌드및실행 | Docker | 배포된 환경에서 어플리케이션을 실행 및 관리하는 컨테이너 도구. |

능을 수행한다.

또한, 그림 4-2에서는 각 질문에 대한 응답 시간이 실제로 기록된 결과를 보여주고 있으며, 사용자가 제출한 응답에 대해 평균 응답 시간을 산출한 내용을 확인할 수 있다. 평균 응답 시간은 2.034초(2034ms)로 측정되었으며, 모든 응답은 2.5초 이내에 처리되었다. 이는 면접 준비 과정에서 실시간 피드백을 제공하기에 충분히 빠른 성능을 나타내는 결과이다.

B. 피드백의 정확성 평가

피드백의 정확성은 사용자의 답변에 대해 LLM 기반 ChatGPT가 제공한 피드백이 실제 면접 평가 기준과 얼마나 일치하는지를 평가하였다. 이를 위해 IT 전문가 10명이 사전에 설정한 10개의 기술 면접 질문에 대한 표준 답변을 기준으로 피드백의 정확도를 측정하였다. 각 질문에 대해 정확성, 논리적 흐름, 기술적 오류, 상세도를 기준으로 1점에서 5점까지 점수를 부여하였다.

실험 결과, 표 2처럼 ChatGPT가 제공한 피드백의 평균 정확성은 4.45점으로 측정되었다. 특히, 객체지향 프로그래밍(OOP)(평균 4.6점), 데이터베이스 정규화(평균 4.6점), 스택과 큐(평균 4.5점)와 같은 주제에서 높은 정확성을 보였으며, 논리적 흐름 및 기술적 오류 탐지에서도 매우 정확한 피드백을 제공하였다. 다만, 알고리즘(평균 4.3점) 및 기술적 세부 사항에서는 추가적인 개선이 필요함이 발견되었다.

이러한 실험을 통해 ChatGPT의 피드백은 전반적으로 높은 신뢰도를 가지지만, 특정 분야에서 더 심화된 설명이 필요할 수 있음을 확인할 수 있었다.

표 2. 피드백 정확성 평가 설문 결과

Table 2. Feedback Accuracy Evaluation Survey Results

| 질문 번호 | 질문 내용 | 평균 점수 |
|-------|-----------------------------------|-------|
| 1 | 시스템이 제공한 피드백의 전반적인 정확성은 어땠습니까? | 4.5 |
| 2 | 피드백의 논리적 흐름은 적절했습니까? | 4.3 |
| 3 | 피드백에서 기술적 오류를 정확히 탐지하였습니까? | 4.2 |
| 4 | 피드백이 충분히 상세하고 구체적이었습니까? | 4.4 |
| 5 | 객체지향 프로그래밍(OOP)에 대한 피드백이 정확하였습니까? | 4.6 |
| 6 | 데이터베이스 정규화에 대한 피드백이 정확하였습니까? | 4.6 |
| 7 | 스택과 큐에 대한 피드백이 정확하였습니까? | 4.5 |
| 8 | 알고리즘에 대한 피드백이 논리적이었습니까? | 4.3 |
| 9 | ChatGPT의 피드백이 전체적으로 유용하였습니까? | 4.5 |

표 3. AI 모의 면접 시스템에 대한 사용자 만족도 설문 결과

Table 3. User Satisfaction Survey Results for the AI Mock Interview System

| 항목 번호 | 설문 항목 | 평균 점수 |
|-------|--|-------|
| 1 | 시스템의 전반적인 사용 편의성 | 4.6 |
| 2 | 시스템의 응답 속도(피드백 제공 시간) | 4.7 |
| 3 | 실전 모드와 연습 모드의 차별성이 면접 준비에 도움이 되었는가? | 4.5 |
| 4 | 실시간 피드백이 답변의 정확성 및 논리적 흐름을 파악하는데 도움이 되었는가? | 4.8 |
| 5 | 피드백 기능이 실제 면접 준비에 긍정적인 영향을 주었는가? | 4.7 |
| 6 | 제공된 피드백이 적절하고 구체적이었는가? | 4.6 |
| 7 | 이 시스템이 면접 준비에서 자신의 취약점을 파악하는 데 도움이 되었는가? | 4.7 |
| 8 | 시스템 사용 후 면접 준비의 자신감이 향상되었는가? | 4.5 |
| 9 | 시스템이 제공한 질문이 면접 준비에 적합하다고 생각하십니까? | 4.4 |
| 10 | 이 시스템을 다른 면접 준비자에게 추천하고 싶은가? | 4.8 |

C. 사용자 피드백과 만족도 평가

사용자 만족도는 실험 종료 후 설문조사를 통해 평가되었다. 10명의 사용자가 시스템을 사용한 후 10개의 객관식 문항을 기반으로 시스템의 피드백과 사용성을 평가하였다. 평가 항목으로는 사용 편의성, 피드백의 구체성, 응답 속도, 피드백의 유용성이 포함되었다. 설문 결과, 사용자의 90% 이상이 피드백의 실시간성과 정확성에 대해 매우 만족한다고 응답하였다. 구체적인 설문 항목과 점수는 표 3과 같다.

V. 결론 및 향후 연구

A. 연구 요약

본 연구에서는 LLM(Large Language Model) 기반 ChatGPT를 활용한 Computer Science 면접 준비를 지원하는 AI 모의 면접 시스템을 설계하고 구현하였다. 이 시스템은 실시간 피드백 기능을 통해 학습자가 자신의 취약점을 파악하고 개선할 수 있는 기회를 제공하였으며, 특히 기술적 면접 준비 과정에서 객관적이고 공정한 피드백을 제공하여 면접 대비 능력을 높이는 데 기여하였다. 음성 인식을 통해 사용자 경험을 향상시키고, 연습 모드와 실전 모드를 통해 다양한 학습 방법을 지원하였다.

B. 연구 기여

면접 준비 시스템에서의 AI 기술 적용을 통해 몇 가지 주요 기여를 하였다:

- 1) 객관적 평가의 도입: LLM을 활용한 자동화된 피드백 시스템을 통해 면접자의 답변을 공정하고 객관적으로 평가할 수 있는 시스템을 구현하였으며, 이는 기존 면접 시스템에서의 주관적 평가 문제를 해결하였다.
- 2) 사용자 맞춤형 피드백: 학습자가 면접 준비 과정에서 실시간으로 피드백을 받음으로써, 개인 맞춤형 학습 경로를 설정할 수 있게 하였다. 이로 인해 학습자는 자신의 약점을 파악하고, 즉각적인 개선이 가능해졌다.
- 3) 사용자 경험의 최적화: 다양한 모드(연습 모드, 실전 모드) 제공을 통해 사용자들이 실제 면접과 유사한 상황에서 학습할 수 있도록 지원하였고, AI 기반 실시간 응답을 통해 사용자의 학습 효율성을 극대화하였다.
- 4) 기술적 정확성 및 학습 효과 강화: ChatGPT의 강력한 언어 모델을 기반으로 한 피드백은 논리적 흐름, 기술적 오류 탐지, 세부 사항의 정확성에서 높은 수준의 신뢰도를 보여주었다.

C. 향후 연구 방향

LLM 기반 AI 면접 시스템의 잠재력을 보여주었지만, 다음과 같은 확장 및 개선 가능성을 제시한다.

- 1) 질문 유형의 다양화 및 산업별 확장: 현재 연구는 주로 Computer Science 분야에 중점을 두었지만, 향후 연구에서는 의료, 법률, 금융 등 다양한 산업 분야에서의 면접 준비를 지원할 수 있는 질문 유형을 추가하여, 보다 폭넓은 산업군에 적용할 수 있는 AI 면접 시스템으로 확장할 계획이다.
- 2) 멀티모달 입력의 도입: 향후 연구에서는 음성 및 텍스트 기반 피드백 외에도 비언어적 요소(표정, 제스처) 등을 분석하여 면접자의 신체 언어를 평가할 수 있는 멀티모달 AI 시스템으로 발전시킬 수 있다. 이를 통해 더욱 종합적인 평가가 가능해질 것이다.
- 3) 피드백의 세밀함 및 맥락 분석 개선: LLM 모델을 지속적으로 업그레이드하여, 사용자의 답변 맥락에 따라 더 정교하고 세밀한 피드백을 제공할 수 있도록 연구가 필요하다. 특히, 기술적 질문에서 세부적인 피드백을 제공하는 알고리즘의 개선이 향후 과제가 될 것이다.

D. 최종 결론

LLM 기반 AI 모의 면접 시스템은 기존 면접 준비 방식의 한계를 극복하고, 면접자의 학습 효과를 극대화할 수 있는 새로운 방법을 제시하였다. 실시간 피드백 제공을 통해 학습자는 자신의 취약점을 인식하고 즉각적인 개선이 가능하며, AI 기술을 통해 면접 준비 과정에서의 효율성 또한 크게 향상될 수 있다. 향후 연구는 다양한 산업 적용, 심리적 효과 분석, 개인 맞춤형 학습 경로 등을 포함하여 AI 면접 시스템의 발전 가능성을 더욱 넓힐 수 있을 것이다.

참고문헌

- [1] Y.-C. Chou, F. R. Wongso, C.-Y. Chao, and H.-Y. Yu, "An AI Mock-interview Platform for Interview Performance Analysis," in Proceedings of the 2022 10th International Conference on Information and Education Technology (ICIET), IEEE, pp. 37-41, Apr. 2022.
- [2] G. Bansal, V. Chamola, A. Hussain, M. Guizani, and D. Niyato, "Transforming Conversations with AI—A Comprehensive Study of ChatGPT," Springer US, vol. 16, no. 5, pp. 2487-2510, 2024.
- [3] P. Kumar, "Large Language Models (LLMs): Survey, Technical Frameworks, and Future Challenges," *Springer Netherlands*, vol. 57, no. 10, 2024, DOI: 10.1007/s10462-024-10888-y.
- [4] Y. Shin and J. Shin, "Structure and improvement strategies for TOPIK argumentative writing based on chatGPT," *Journal of East Asian Cultures*, no. 98, pp. 121-146, August, 2024.
- [5] Y. Kim, "A case study on experiencing an ai interview system," *Culture and Convergence*, vol. 45, no. 11, pp. 283-294, 2023, DOI: 10.33645/cnc.2023.11.45.11.283.
- [6] S. Choi, H. Kim, Y. Lee, H. Lee, D. Ryu, and C. Yoo, "React-based integrated service for collaborative support tools," *Journal of Korean Institute of Information Technology*, vol. 2022, no. 6, pp. 763-767, 2022.
- [7] Y. Park, H. Park, and H. Byun, "Design and Implementation of a User-Sharing-Oriented Online Library System Using Spring Framework," *Korean Institute of Information Scientists and Engineers*, vol. 36, no. 1B, pp. 383-387, June, 2009.

[8] “Amazon Relational Database Service,” Amazon Web Services, [Online]. Available: https://aws.amazon.com/ko/rds/?nc2=h_ql_prod_db_rds. [Accessed: Sep. 30, 2024].
[9] “Whisper,” OpenAI, [Online]. Available: <https://openai.com/index/whisper/>. [Accessed: Sep. 30, 2024].

[10] K. Choi, E. Kang, and H. Kang, “Providing multiple BLOB types for efficient storage of multimedia data,” *Journal of KIISE*, vol. 22, no. 10, pp. 1404-1415, October, 1995.



천재성 (Jae-Sung Chun)_정회원

2011년 2월 : 한국기술교육대학교 컴퓨터공학부 졸업
2022년 8월 ~ 2024년 8월 : 한국기술교육대학교 컴퓨터공학과 석사
2024년 8월 ~ 현재 : 한국기술교육대학교 컴퓨터공학과 박사 과정
<관심분야> 웹, 앱, AI, 빅데이터, LLM



장희권 (Hee-Kwon Jang)_정회원

2025년 2월 : 인천대학교 컴퓨터공학부 졸업
<관심분야> 서버 개발, DB, 인프라



김지혜 (Ji-Hye Kim)_정회원

2024년 8월 : 중앙대학교 컴퓨터예술학부 졸업
<관심분야> 웹, 앱, 기획, 디자인



배창민 (Chang-Min Bae)_정회원

2021년 2월 : 순천대학교 멀티미디어공학과 졸업
<관심분야> 웹, 앱, QA, 데이터분석



이동규 (Dong-Gyu Lee)_정회원

2024년 2월 : 강원대학교 컴퓨터공학과 졸업
<관심분야> AI, 웹, 데이터베이스



문 일 영 (Il-Young Moon)_종신회원

2005년 2월 : 한국항공대학교 항공통신정보공학과 공학박사
2005년 3월 ~ 현재 : 한국기술교육대학교 컴퓨터공학부 정교수
<관심분야> AI, 무선인터넷 응용, 무선 인터넷, 모바일 IP