|  |
| --- |
| **1. 주제**  난청자를 위한 실시간 전화 통화 텍스트 변환 시스템  **나반 4팀 20241977 이환희** |

|  |  |
| --- | --- |
| **2. 요약**  - 목표: 청각 장애인 및 난청인을 위해 전화 통화의 음성을 실시간으로 텍스트로 변환해 주는 시스템을 개발하여 사용자들이 전화 통화 내용을 실시간으로 확인할 수 있도록 지원하는 것을 목표로 합니다.  - 핵심 내용: 이 시스템은 상대방의 음성을 텍스트로 변환해 화면에 표시하고, 사용자는 문자 대신 자신의 음성으로 응답합니다. 이를 통해 노인 사용자의 불편함을 줄이고, 문자 입력의 어려움을 해소할 수 있습니다.  - 중요성: 사회적으로 노인들의 디지털 소통 접근성을 확대하고, 전화 통화에서 겪는 제약을 해소할 수 있습니다. 특히, 일상적인 전화 통화에서의 불편함을 줄여 더욱 원활한 소통을 가능하게 합니다. | **3. 대표 그림**  - 개발 배경: 보청기를 사용해도 잘 안 들리시는 할아버지와의 통화가 너무 어려워서 개발하게 되었습니다.  설명: 통화 중 상대방의 음성을 실시간으로 텍스트로 변환해 화면에 표시하고, 노인 사용자는 음성으로 응답할 수 있는 시스템 프로세스 다이어그램입니다.  - 예상 결과    그림 1. 실시간 전화 통화 텍스트 변환 시스템 |

|  |
| --- |
| **4. 서론**  - 배경, 사례: 청각 장애인과 난청을 가진 노인들은 전화 통화에서 상대방의 말을 이해하는 데 어려움을 겪을 수 있습니다. 문자 입력을 통한 응답은 시간이 오래 걸리며 불편하기 때문에, 노인 사용자들이 문자 대신 음성으로 응답할 수 있는 솔루션이 필요합니다.  **-** 문제 정의: 기존의 음성-텍스트 변환 시스템은 문자로 답변해야 하는 방식이 대부분입니다. 그러나 문자 입력이 느리거나 불편한 노인 사용자를 위해 음성으로 응답할 수 있는 대안이 필요합니다.  - 극복 방안: 상대방의 음성을 실시간으로 텍스트화하여 노인 사용자가 내용을 쉽게 읽을 수 있도록 하며, 동시에 노인 사용자는 문자 입력 없이 음성으로 응답할 수 있는 시스템을 개발한다면 이 문제를 극복할 수 있습니다. |

|  |
| --- |
| **5. 본론**  이 시스템은 전화 통화에서 상대방의 발언을 실시간으로 텍스트화하여 화면에 표시합니다. 동시에, 사용자는 문자 입력을 할 필요 없이 자신의 목소리로 응답하여, 노인 사용자의 불편함을 줄이고 편리하게 소통할 수 있도록 지원합니다.    - 그림 2. 시스템 흐름 개요  **- 필요한 기술 요소:** 음성 인식 (Speech-to-Text) 기술  상대방의 음성을 실시간으로 텍스트로 변환하는 기능은 Google Speech-to-Text API 또는 IBM Watson Speech-to-Text API와 같은 음성 인식 서비스를 사용하여 구현합니다. 서버에서 실시간 음성 인식을 수행한 후, 텍스트화된 내용을 클라이언트 화면에 실시간으로 전송하여 사용자가 읽을 수 있도록 합니다.  **통화 연결 및 데이터 전송**  WebSocket을 사용하여 클라이언트와 서버 간 실시간 데이터 전송을 구현합니다. WebSocket을 통해 지속적인 연결을 유지하면서 음성 데이터를 즉시 텍스트로 변환하고, 사용자 화면에 표시할 수 있습니다.  통화 중 상대방의 음성을 텍스트로 변환하고 노인 사용자는 음성으로 응답할 수 있게 하여 문자 입력의 불편함을 해소합니다.  **UI/UX 설계**  노인 사용자가 읽기 편하게 큰 글씨 크기와 높은 색상 대비를 가진 UI를 설계합니다. 텍스트는 대화 흐름을 놓치지 않도록 실시간으로 업데이트되며, 음성으로 응답할 수 있는 간단한 인터페이스를 포함합니다.  UI는 통화 중에 상대방의 발언이 끝나면 노인 사용자가 쉽게 마이크 버튼을 눌러 응답할 수 있도록 설계되어야 합니다.  **- 구현 방법 및 개발 방향: 1. 음성 인식 및 음성 전송 모듈 구현**  클라이언트 측  통화 중 상대방의 음성을 캡처하여 서버로 전송하고, 사용자가 마이크 버튼을 통해 자신의 음성을 상대방에게 전송할 수 있도록 합니다. 이를 위해 Twilio API 등을 사용하여 전화 통화 기능을 통합하고, PyAudio 라이브러리로 오디오 스트리밍을 관리할 수 있습니다.  서버 측  Flask 서버에서 Google Speech-to-Text API를 사용하여 실시간 음성 인식을 수행하고, 변환된 텍스트를 WebSocket을 통해 클라이언트에 전송합니다. 동시에, 사용자가 마이크를 통해 발언할 때, 해당 음성을 Twilio API를 통해 상대방에게 전송할 수 있습니다.  **2. 실시간 데이터 전송**  WebSocket 프로토콜 사용  Flask-SocketIO를 사용하여 클라이언트와 서버 간 실시간 데이터 전송을 구현합니다. 텍스트 업데이트를 통해 상대방 발언을 화면에 표시하고, 노인 사용자는 마이크 버튼을 눌러 자신의 발언을 시작할 수 있습니다.  **3. 사용자 인터페이스 (UI) 설계 및 구현**  프론트엔드 개발  HTML, CSS, JavaScript, React를 사용하여 노인 사용자를 위한 간단하고 직관적인 UI를 설계합니다. 상대방의 발언은 화면에 실시간으로 표시되며, 큰 글씨와 편리한 인터페이스가 제공됩니다.  마이크 버튼을 화면에 배치하여 사용자가 문자 입력 대신 음성으로 쉽게 응답할 수 있도록 하고, 화면에 상대방의 텍스트가 업데이트되면 마이크 버튼이 활성화되도록 합니다.  UI 반응성  WebSocket을 통해 실시간으로 텍스트가 업데이트되며, 텍스트가 업데이트될 때마다 UI가 자동으로 새 텍스트에 맞추어 이동하도록 하여 사용자가 대화 흐름을 놓치지 않도록 합니다.  **구현 일정**  1단계: 요구사항 분석 및 시스템 설계  프로젝트 목표와 기능 명세서를 작성하고, 시스템 다이어그램과 UI 목업을 설계합니다. (2주)  2단계: 음성 인식 모듈 및 전화 연결 개발  Google Speech-to-Text API와 Twilio API를 통해 음성 인식과 전화 연결 기능을 구현합니다. (2주)  3단계: 실시간 데이터 전송 모듈 개발  WebSocket을 사용하여 서버-클라이언트 간 실시간 데이터 전송을 구현합니다. (1주)  4단계: 프론트엔드 개발  사용자 인터페이스를 설계하고, 노인 사용자가 쉽게 사용할 수 있는 텍스트 표시 및 음성 응답 기능을 완성합니다. (2주)  5단계: 통합 테스트 및 최적화  시스템 통합 후, 응답 속도와 정확도를 확인하고 UI를 최적화합니다. (1주) |

|  |
| --- |
| **6. 결론**  이 보고서는 노인 사용자가 문자 입력의 불편함 없이 전화 통화 중 상대방의 말을 실시간 텍스트로 확인하고, 음성으로 응답할 수 있는 시스템을 제안했습니다. 향후 음성 인식의 정확도를 더욱 높이고 사용자 인터페이스를 개선하여 노인 사용자가 편리하게 사용할 수 있도록 발전시킬 계획입니다. |

**7. 출처**

[1] Google Cloud Platform, “Google Speech-to-Text Documentation,” 2024.

[2] Twilio API, “Twilio Programmable Voice,” 2024.

[3] Flask-SocketIO, “WebSocket-based communication for Python,” 2024.