

2024년 전기

다중화자 탐지가 가능한 STT 기술 개발



부산대학교 전기컴퓨터공학부 정보컴퓨터 공학 전공

지도교수 : 권준호

팀 명 : *Untouchable*

201824492 변상윤

201724433 문성필

<목차>

1. 과제 배경 및 목적
2. 요구 조건 분석 및 제약 사항
 - 2.1. 요구 조건 분석
 - 2.2. 제약 사항 및 대책
 - 2.2.1. 제약 사항
 - 2.2.2. 대책
3. 개발 환경 및 사용 기술
 - 3.1. 개발 환경
 - 3.2. 사용 기술
4. 개발 일정 및 역할 분담
 - 4.1. 개발 일정
 - 4.2. 역할 분담

1. 과제 배경 및 목적

최근 인공지능 기술의 발전으로 음성인식 기술인 STT (Speech-To-Text) 모델이 다양하게 개발되었다. STT란 사람이 말하는 음성 언어를 컴퓨터가 해석하여 텍스트로 변환하는 처리를 의미한다. 이는 회의록 작성, 유튜브 자막 생성, 상담 기록, 음성 명령어 처리 등 다양한 분야에서 활용되고 있다.

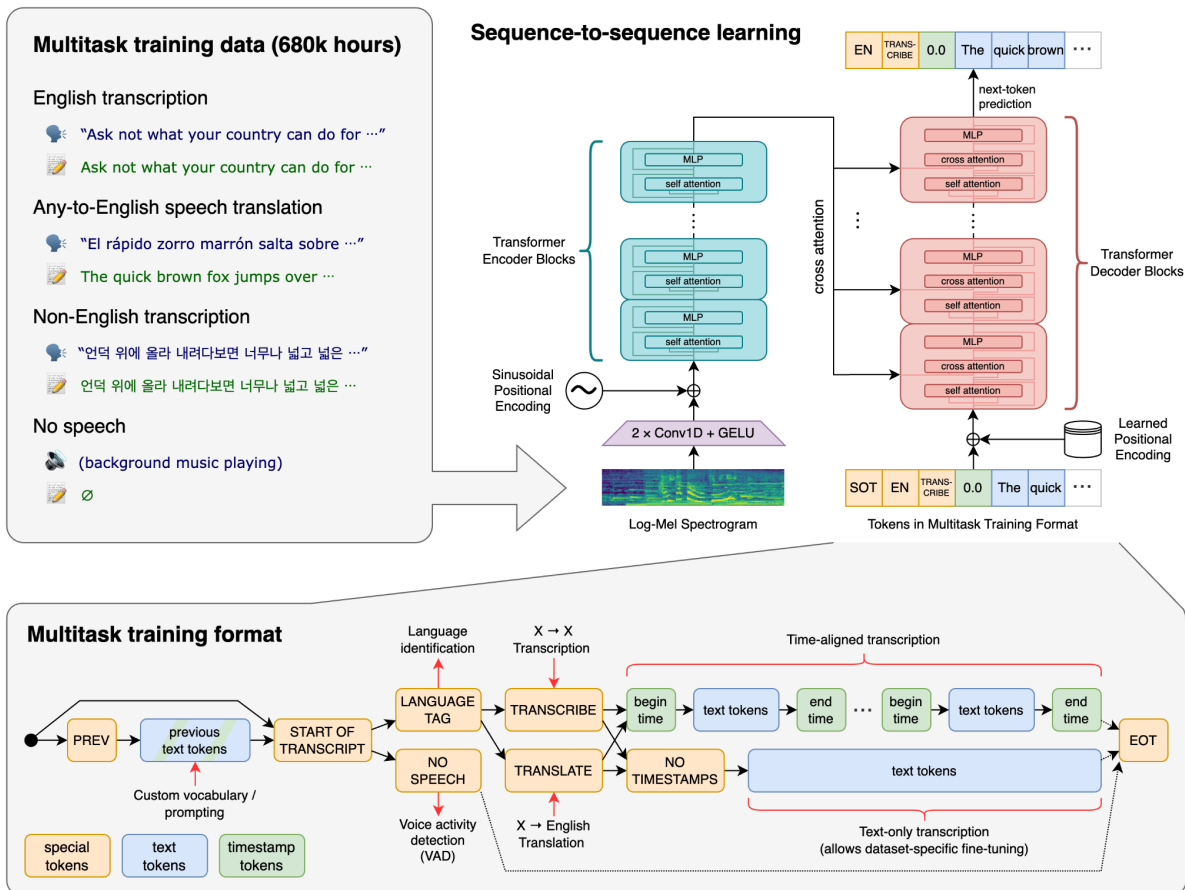
우리는 다중화자 인식이 되는 STT를 개발함으로써 위의 효과와 더불어 추가적인 효용을 기대할 수 있다. 먼저 다중화자 인식에 대한 알고리즘 학습과 그에 대한 심화를 생각해 볼 수 있다. 또한 다중화자 구현 이전에 STT 코드 분석이 선행되어야 하므로 STT 자체에 대한 이해도를 증가시킬 수 있다.

2. 요구 조건 분석 및 제약 사항

2.1. 요구 조건 분석

2.1.1. STT 구현

다중화자 인식이 되는 STT를 개발하려면 먼저 STT 모델을 구현해야한다. 아직 관련 지식이 부족한 상태이므로 이를 위해 관련 배경 지식을 학습하는 시간이 필요할 것이며, Whisper를 기반으로 구현하므로 이에 대한 코드 분석도 필요할 것이다. 아래는 Whisper 모델의 개요다.



¹ Whisper API Approach.

2.1.2. 다중 화자 식별 후 라벨링 구현

2.1.2.1. 전이학습을 이용하여 다중화자 구현

일반적인 ML의 방식을 사용하면 기존 STT의 학습된 모델을 이용할 수 없고 다중화자 구현시 새로 학습을 시켜야 한다. 따라서 기존 STT에서 전이학습을 시켜 다중화자를 구현함으로써 더 정확도를 높일 수 있고 더 적은 훈련 모델만으로도 학습을 시킬 수 있다.

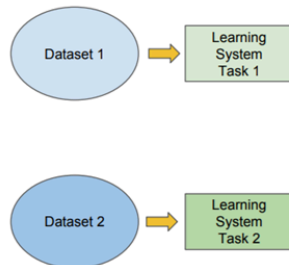
¹ <https://github.com/openai/whisper?tab=readme-ov-file>

Traditional ML

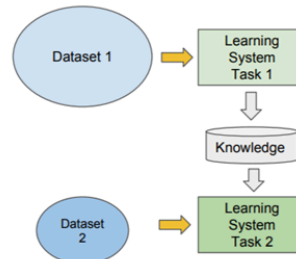
vs

Transfer Learning

- Isolated, single task learning:
 - Knowledge is not retained or accumulated. Learning is performed w.o. considering past learned knowledge in other tasks



- Learning of a new tasks relies on the previous learned tasks:
 - Learning process can be faster, more accurate and/or need less training data



² ML vs TL

2.2. 제약 사항 및 대책

2.2.1. 제약 사항

- 2.2.1.1. 화자가 말이 겹치거나 노이즈가 많이 낀 음성의 경우 따로 전처리를 해주어야 한다.
- 2.2.1.2. STT 모델을 위한 음성 모델 수집의 저작권
- 2.2.1.3. 다중 화자 식별을 위한 음성 데이터 수집

2.2.2. 대책

- 2.2.2.1. 순차적인 대화형식의 노이즈가 끼지 않은 음성 파일만을 대상으로 개발한다
- 2.2.2.2. 저작권 없는 음성 소스를 사용한다
- 2.2.2.3. 라벨링이 되어 있는 음성 소스를 사용한다.

3. 개발 환경 및 사용 기술

3.1. 개발 환경

Linux, Python

3.2. 사용 기술

openai Whisper

² <https://www.v7labs.com/blog/transfer-learning-guide>

4. 개발 일정 및 역할 분담

4.1. 개발 일정

월	6					7					8					9				
주	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
whisper 코드 분석																				
배경지식 학습																				
다중화자 구현																				
문제점 개선																				

4.2. 역할 분담

4.2.1. 문성필

4.2.1.1. Whisper 코드 분석

4.2.1.2. STT 모델 구현

4.2.1.3. 문제점 개선

4.2.2. 변상윤

4.2.2.1. Linux 환경 구현

4.2.2.2. 다중화자용 Transfer Learning 조사 및 구현

4.2.2.3. 학습 모델 선별

4.2.3. 공통

4.2.3.1. 배경 지식 학습

4.2.3.2. 보고서 작성

4.2.3.3. 발표 및 시연 준비