|  |
| --- |
|  |
| **제19회 임베디드SW경진대회 개발완료보고서**  **[자유공모]** |
|  |

※ ‘파란색 안내 문구’는 삭제하고 검정색 글씨로 총 30page 이내로 작성하여 PDF로 변환하여 제출. (폰트 : 맑은 고딕 / 폰트 크기 : 11pt / 줄 간격 : 160%)

**□ 개발 요약**

※ 1page 이내로 작성

|  |  |
| --- | --- |
| **팀 명** | VVS |
| < 작품 사진 및 그림 > | |
| **작품명** |  |
| **작품설명**  **(요약)** |  |
| **소스코드** | < GitHub 링크 > |
| **시연동영상** | < YouTube 링크 / 1개의 동영상 링크만 삽입 > |

**□ 개발 개요**

※ 프로젝트 개발 배경, 동기, 목표, 필요성 등 / 2page 이내로 작성

○ 개발 작품 개요 / 제안하는 작품에 대한 개요를 자세히 기술한다.

- 농인들의 통화 서비스 이용 접근성 향상을 위해 실시간 통화 동시통역 시스템을 개발하였음.

- 각자의 통화 환경에 맞게 시스템 구축. 서비스 제공자는 음성을 활용하여 대화하고 서비스 대상자는 텍스트와 이모지를 활용해 대화함.

- 서비스 제공자의 음성이 대상자에게 텍스트로 변환되어 전달, 대상자가 작성한 텍스트는 제공자에게 음성으로 변환되어 다시 전달됨. 변환을 담당하는 하드웨어는 라즈베리파이4.

○ 개발 목표 / 개발 목표를 명확하게 제시한다.

- 음성을 텍스트로 바꾸어주는 STT 기술을 해당 서비스에 맞게 최적화 된 방법으로 개발한다.

- 텍스트를 음성으로 바꾸어주는 TTS 기술을 해당 서비스에 맞게 개발한다.

○ 개발 작품의 필요성 / 개발하는 작품이 왜 필요한지에 대해 상세히 서술한다.

- 현재 음성 통화를 텍스트로, 텍스트를 음성으로 실시간으로 바꿔주는 서비스가 제공되지 않고 있음.

○

-

-

**□ 개발 환경 설명**

※ 최대한 자세하게 기술 / 15page 이내로 작성

○ Hardware 구성

-

-

○ Hardware 기능 (제어 방법 등 서술)

-

-

○ Software 구성

-

-

○ Software 설계도 (흐름도 및 클래스 다이어그램 등 / 개발언어에 따라 선택)

-

-

○ Software 기능 (필요 시 알고리즘 설명 포함)

-

-

○ 프로그램 사용법 (Interface)

- 학습을 위한 파일 리스트들

-

○ 개발환경 (언어, Tool, 사용시스템 등)

- shell script, perl script, c++, python

- Tool: tornado, selenium, ws4py, json, tensorflow, pytorch

- 시스템: linux, Raspbian, code-server, Jupyter server

**□ 개발 프로그램 설명**

※ 최대한 자세하게 기술 / 8page 이내로 작성

○ 파일 구성

-

-

○ 함수별 기능

-

-

○ 주요 함수의 흐름도

-

-

○ 기술적 차별성

-

-

**□ 개발 중 발생한 장애요인과 해결방안**

※ 개발 과정에서 나타났던 모든 장애 요인(Risk)들을 나열하고, 이러한 장애요인들이 발생했던 경우 어떻게 해결했는지 구체적으로 제시 / 1page 이내로 작성

○

- 딥러닝 모델로 진행하려 했으나, 실시간 분석 성능이 좋은 모델을 사용하기 어려웠고, 있다해도 한국어 성능이 매우 조악했음 (RNN-T 모델의 경우 잘 나와야 WER 20%)

- sox 음성코덱 재현 문제

- 온라인 디코딩 문제 (한국어를 위한 nnet3 model 존재 x)

- Language Model 추가 (기존 interpolation 방식을 사용하는 것이 아닌 최신 phrase hint 기능을 사용하기 위해 다양한 연구)

(- 배달 관련 말뭉치 구하는 문제 (selenium을 이용한 웹 크롤링

- 말뭉치 하나 학습하는데도 많은 시간이 걸렸음 -> phrase hint

- 또한 interpolation 하고 나면 손상이 심해 다른 분야에 확장 어렵

)

그래서 arpa로 합치는데

○

- 낮은 computing 성능 (부족한 RAM을 해결하기 위해 parallel을 이용한 병렬 처리 추가 및 arpa파일 쪼개기 수행)

- kaldi documentation의 빈약한 부분을 각자 스스로 공부하여 채워넣어야했음

(kaldi binary 파일 분석 및 다양한 shell script 파일 분석, kaldi I/O, 문법 파악 등)

-

intel mkl 동적 라이브러리 설정 오류,

○

- ㅈㄴㄴㄴㄴㄴ방대한 시스템 파일간의 구조파악이 어려웠음

**1페이지 생각하지말고 그때그때 최대한 많이**

**□ 개발결과물의 차별성**

※ 개발한 결과물과 기존 발표된 유사작품(제품) 간 차별성 및 우수성 설명 / 1page 이내로 작성

○ API보다 이게 더 나은 이유

- 비용이 들지 않음 (실시간, 오프라인 구현)

- 우리가 사용하는 domain에 특화 가능

- 다양한 상용품에 준하는 성능(zeroth ee를 사용하는 기업 list 참조)

- 한국어 관련 Open source가 없는 kaldi tool을 이용

○ 최신 DEEP LEARNING 보다 나은 이유

- 최신 deep learning model을 사용하였을땐

- (facebook의 wav2vec2.0을 활용한 PORORO의 asr보다 우수한 인식능력을 보임)

- 또한 최신 모델에서는 실시간 음성인식이 어려운 문제를 해결하지 못함

○

-

-

**□ 개발 일정**

※ 실제 프로젝트 개발 일정 작성 / 1page 이내로 작성

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **내용** | **2020年** | | | | | | | | | | | | | | | |
| **6月** | | | | **7月** | | | | **8月** | | | | **9月** | | | |
| 1 | 인공지능 공부 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | 다양한 NLP 모델 학습 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 기능 구현 및 개발 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 시험 평가 및 테스트 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

- 인공지능 공부

- 다양한 NLP 모델 학습 (참고한 논문 및 github 첨부)

-

- STT, TTS 관련하여 학습

- KALDI 구조 (wFST(HCLG) 구조) 파악

- AM, LM train 방법 파악 (arpa 파일 구조 파악, 등등,)

- 한국어 음운 특징 파악

- fst 제작 방식 파악 openfst 분석

- decoding 방식 파악

- lattice 구조 파악

- 코덱 분석해서 최대한 비슷하게 만드는 작업

- (신지혜) API관련 분석 (우리가 직접 구현해서 비교해봤다라는걸 보여줘야함)

**□ 팀 업무 분장**

※ 프로젝트 개발 관련 팀원의 업무 분장 위주로 작성 / 1page 이내로 작성

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **구분** | **성명** | **참여인원의 업무 분장** |
| 1 | 팀장 | ○○○ |  |
| 2 | 팀원 | ○○○ |  |
| 3 | 팀원 | ○○○ |  |
| 4 | 팀원 | ○○○ |  |
| 5 | 팀원 | ○○○ |  |