

청년 AI·빅데이터 아카데미 6기 프로젝트 결과보고서

A반 1조
김원배 연선경 오가영
이재윤 이준영 최우혁

프로젝트명

상호작용 접객 로봇, 딥 포닉스

요 약 본

프로젝트 정보	
주제영역	<input type="checkbox"/> 건강 <input type="checkbox"/> 생산성 <input type="checkbox"/> 생활 <input type="checkbox"/> 안전 <input checked="" type="checkbox"/> 엔터테인먼트
기술분야	<input type="checkbox"/> 이동통신 <input type="checkbox"/> 방송스마트미디어 <input checked="" type="checkbox"/> 기반SW컴퓨팅 <input type="checkbox"/> 디지털콘텐츠 <input type="checkbox"/> 융합서비스 <input type="checkbox"/> 네트워크 <input type="checkbox"/> 전파·위성 <input checked="" type="checkbox"/> SW <input type="checkbox"/> 정보보호 <input type="checkbox"/> ICT 디바이스
프로젝트명	딥 포닉스(Deep PONIX), 상호접객 로봇
프로젝트 소개	<p>딥 포닉스(Deep PONIX)는 상호작용이 가능한 접객로봇이다.</p> <p>최근 접객로봇은 로봇 분야의 중요한 개발 주제로 떠오르고 있다. AI 기술을 통해 음성 및 영상을 처리하고, 자율주행까지 가능한 접객로봇은 안내원을 대신할 수 있으며, 인천공항, 박물관 등에서 등장하고 있다. 접객로봇은 AI 기술의 총 집합체로 다양한 제조기업과 ICT 기업의 상호협력을 통해 개발 진행 중이다.</p> <p>포닉스는 포항공대의 마스코트이다. 마스코트 형상을 한 안내로봇 딥 포닉스를 3D 프린팅 기술과 우드락을 활용해 제작하였다. 딥 포닉스 얼굴과 몸체에 카메라, 라즈베리파이, KT 메이커스 kit, 디스플레이를 탑재해 청년 AI 빅데이터 과정과 정보통신연구소에 대한 안내기능을 제공하고자 한다.</p>
개발배경 및 필요성	<p>우리는 청년 AI 빅데이터 6기 교육생으로 입소한 후, 정보통신연구소, 포항공대, 교육과정에 대한 안내정보가 부족함을 느꼈다. 연구소에 대한 정보, 식당 위치, 교육 장소 위치 정보 등 모르는 정보에 대해 담당자에게 문의하거나 일일이 검색해야 했다. 이런 불편함 속에서 다음 기수와 방문객들을 위해 안내원 같은 존재가 필요하고, 많은 이들에게 편의를 제공하기 위한 안내로봇인 딥 포닉스를 개발하기로 결정했다.</p>
프로젝트 주요기능	<p>딥 포닉스의 기능은 크게 음성과 영상 파트로 나뉜다.</p> <p>음성 기술 부분은 KT 기가 지니 API를 활용하여 기능을 개선하고자 시도했다. '포닉스'라는 호출어를 인식하고, 정보통신연구소에 관한 질의응답이 가능하도록 정보통신연구소용 디렉너리를 제작했다.</p> <p>영상 기술 부분은 얼굴 인식을 중점적으로 시도했다. 내장된 카메라를 통해 얼굴 이미지를 인식하고, 얼굴 데이터가 학습된 사람과 학습되지 않은 사람을 분류해 다른 대사를 출력하도록 만들었다. 학습된 사람인 경우, 안내로봇이 "000님 안녕하세요?"라고 인사를 하고, 학습되지 않은 사람인 경우는 "제가 처음 뵈네요..처음뵙겠습니다..."라고 인사할 수 있도록 개발했다.</p>
작품의 기대효과 및 활용분야	<p>정보통신연구소의 방문객에게 편리성과 즐거움을 줄 수 있다. 포항공대 마스코트인 포닉스 형상을 한 안내로봇을 통해 포항공대와 정보통신연구소의 홍보 효과를 줄 수 있다. 실내지도를 매핑해 자율주행하는 기술, 자유 대화 및 정보통신연구소 안내기능 업그레이드, 딥 포닉스 전체 3D 프린팅 시도를 통해 '자율주행하며 안내하는 접객로봇'의 아이디어를 점차 완성시켜 나갈 수 있을 것으로 기대한다.</p>

(본문) 프로젝트 결과보고서

I. 프로젝트 개요

가. 프로젝트 소개

○ 프로젝트명

“ 음성인식과 영상인식을 활용한 상호작용하는接客로봇, 딥 포닉스 ”



○ 딥 포닉스는 보이고 움직이는 AI 스피커

→ ‘PIRL 실내 길 인식 통해 자율주행하는接客로봇’

- 우리가 만들고자 하는 말하는 서비스 로봇 딥 포닉스는 포항공과대학 정보통신연구소(PIRL)을 안내하는接客로봇임.

우리 팀의 궁극적 목표는 ‘연구소 내 길을 인식해 자율주행하는 로봇’ 만드는 것임.

- 초기 개발 단계에서는 연구소 안내 기능을 위해 STT(Speech To Text)-TTS(Text To Speech)를 통해 전체적인 음성 안내 시스템을 제공하

고자 함. 이를 위해 KT makers kit와 KT GIGA Ginie API를 사용함.

- AI 스피커 기능을 개발하고, 추후 실시간 얼굴 이미지를 인식하는 영상 처리, 마지막 단계에서는 PIRL 연구소 실내 지도를 자율주행을 통해 매핑하는 기술을 통해 실내 자율주행을 완성시키는 것이 목표

○ 서비스형接客로봇은 최근 1~2년 사이 아주 트렌디한 기술 분야로 매년 지능화 로봇 시장은 규모가 증가하고 있음

- 국내 대기업 간 서로 협력하는 합종연횡이 가속화되고 있음. 국내 제조기업이 하드웨어를 담당하고, ICT 기업이 소프트웨어를 담당하는 상호협력을 통해接客로봇 개발 경쟁 중임. 이처럼 4차 산업혁명 시대를 맞이해 자율주행 로봇 AI 등 4차 산업혁명의 핵심 기술을 선점하기 위해 이중산업 기업의 합종연횡이 가속화되고 있음.

나. 개발배경 및 필요성



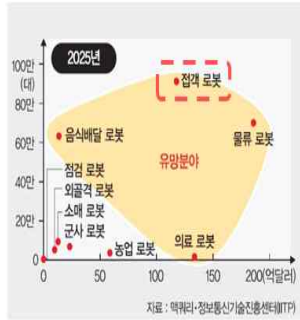
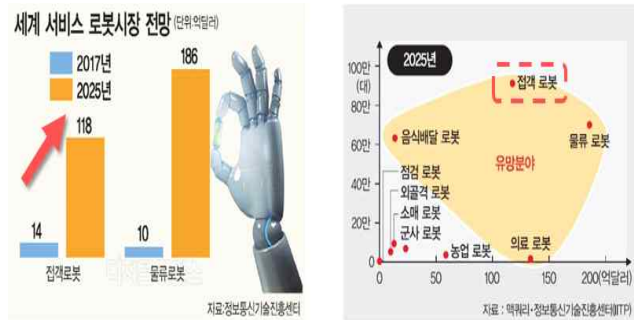
○ 청년 AI · 빅데이터 교육과정, 정보통신연구소에 대한 안내 필요성 느낌

- 우리는 청년 AI · 빅데이터 6기 교육생으로 입소한 후, 정보통신연구소, 포항공대, 교육과정에 대한 안내정보가 부족함을 느꼈다. 연구소에 대한 정보, 식당 위치, 교육 장소 위치 정보 등 모르는 정보에 대해 담당자에게

문의하거나 일일이 검색해야 했다. 이런 불편함 속에서 다음 기수와 방문객들을 위해 안내원 같은 존재가 필요하고, 많은 이들에게 편의를 제공하기 위한 안내로봇인 딥 포닉스를 개발하기로 결정했다.

○ 급성장 중인 서비스 로봇 시장

- 세계接客로봇 시장 규모는 2017년 14억 달러(75,000대)에서 2025년 118억 달러(91만 대)로 8.4배 성장할 것으로 기대되며,接客로봇은 유동인구가 많은 쇼핑몰·음식점·호텔·공항·병원·은행 등에서 도입이 활발하게 이루어질 것으로 예상되고 있음.



- 接客/안내 로봇은 전문/상업용 로봇 시장에서 판매대수가 가장 많은 로봇이며 성장 가능성이 높은 미래 유망분야
- 接客로봇 시장은 소프트뱅크 페퍼가 대중화시킨 분야

※接客(接客)로봇이란?

: 매장이나 기관을 방문하는 고객에게 사람을 대신해 정보제공 / 길안내 / 주문접수 / 외국어대응 등의 서비스를 제공하는 로봇으로, 소프트뱅크 페퍼가 대표적 사례

- 아직 시장 수요가 적어 키오스크보다 대중화되지 못해接客로봇의 가격은 대당 수천만 원 수준으로 다소 부담이 있는 편이며, 고객 환경에 맞는 커스터마이징이 필요

※ 소프트뱅크 법인용 페퍼의 경우 3년 동안 약 2천만 원 (월 55,000 엔 × 36개월) 정도의 비용이 발생

- 인건비가 상승하고 노동력이 부족해지면接客, 안내 서비스형 로봇의 수요가 늘어날 것으로 기대됨. 단순반복接客 업무는 로봇을 활용해 자동화하기 좋은 분야로 최근 패스트푸드점, 건물 등에서 키오스크형 안내데스크를 많이 사용하는 것을 예로 들 수 있음. 5만 원 지출은 비용 효과적일 수 있음

○ 국외 주요 업체 및 최근 동향

- 일본 소프트뱅크가 초기 시장을 선도하고 있는 가운데, 중국·미국·유럽·우리나라 기업들이 추격 중
- 가격이 저렴하면서도 인간과 감성을 나눌 수 있으며 앱 설치로 기능 추가도 가능한 소프트뱅크 페퍼가接客로봇 시장을 활성화시키고 있는 가운데, 최근 중국 스타트업들이接客로봇 분야에 적극적으로 진출을 모색 중
- 接客로봇은 대당 500만 원 수준의 단순 서비스 제공형에서부터 고객과 간단한 의사소통이 가능한 수천만 원짜리 제품까지 다양하며, 폼팩터 측면에서 고정정보보다는 바퀴를 사용하는 이동형이,接客용으로서 고객에게 친근감을 주기 위해 非인간정보보다 인간형 형태가 대세

○ 국내 선진 사례



- 接客/ 안내 로봇은 상업용 로봇 시장에서 판매대수가 가장 많으며, 성장 가능성 큰 분야임
- 19' 부터 LG전자-네이버, 현대자동차-SK 텔레콤 등 기업 공동연구를 통해 서비스형 안내로봇을 중심으로 로봇스틱 기술 개발에 힘쓰고 있음
(국내 AI 스피커의 진화 →接客로봇으로 나아가는 프로세스)
- 서비스 로봇 시장은 급성장하는데 국내는 청소 로봇에 머무르는 실정
- 최근 국내 IT업계와 전자업계 간 협업이 증가하고 있음

1. LG전자

- LG전자 로봇 개발에 가장 적극적인 기업, 가정용 로봇, 안내로봇, 청소로봇, 웨어러블 로봇, 산업용 로봇 등 다양한 분야에서 로봇사업을 추진
- LG전자는 이미 2017년 인천국제공항 안내로봇 '에어스타'와 '청소보트' 개발. 올해는 자체 개발하는 다양한 로봇에 네이버가 자체 개발한 고정밀 위치·이동 통합기술플랫폼 'xDM'을 적용하는 등 협업에 속도를 내고 있음.

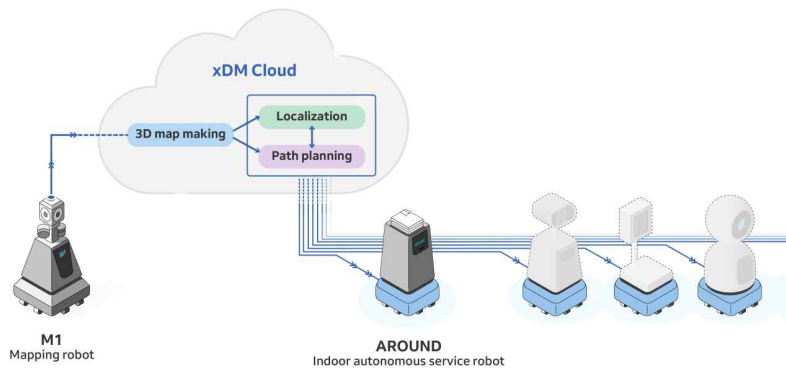


2. 네이버랩스의 AROUND

네이버랩스가 개발한 M1과 어라운드 로봇은 실내공간을 M1이 촬영, 정밀 지도를 클라우드에 저장하고, 이를 기반으로 어라운드가 자율주행하는 시스템을 갖추고 있다. 특히 센서 등 고가 부품을 최소화하고도 원활한 자율주행을 구현해 기존 자율주행 로봇의 약 10분의1 수준의 제작비로도 상용화가 가능할 것으로 예상된다.



AROUND Autonomous Service Robot Platform



3. 현대자동차 & SK텔레콤

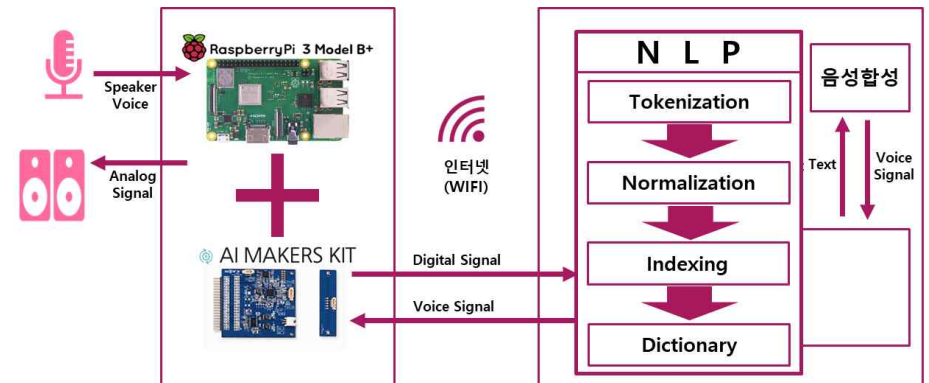
- 현대자동차 & SK텔레콤, 고객 응대용 AI 로봇 개발 추진
- 양사는 19년 3월 11일 ‘서비스 로봇 개발을 위한 양사의 인공지능 플랫폼 간 상호 협력 MOU’를 체결함. 이를 통해 서비스 로봇을 개발해 전시장 직원을 보조해 전시장에서 고객을 응대하는 역할을 수행하기를 기대함.

4. 삼성전자

- 삼성전자는 연내 로봇시장에 본격 진출. 삼성전자는 그간 로봇 제품을 출시

한 적이 없어 관심이 모아지는 상황임. 그간 삼성전자는 C랩과 삼성종합기술원을 주축으로 웨어러블 로봇 등 다양한 로봇을 연구·개발에 몰두해옴.

다. 작품 구성도



라. 작품의 특징 및 장점

○ 음성 처리 파트

▶ ‘포닉스’ 호출어 인식

- 내장 호출어 변경은 불가 → 호출어 변경된 것처럼 코딩
‘포닉스’ 음성 wave 데이터를 학습시켜, 그 말이 ‘포닉스’ 인지 인지하도록 학습시키는 모델 구현

- wave 파일(소리 데이터)을 train data set으로 만든다. 이에 해당하는 정답 데이터 셋[]도 작성한다.

세상에는 우리가 정답 데이터 셋으로 만들어두지 않은 수많은 단어, 소리가 존재한다. 이 문제를 해결하기 위한 방법을 고안한다. 모델 학습을 잘 시키기 위해, unknown 분류 카테고리도 만든다.

- Data agumentaion: 노이즈를 줘서 데이터를 엄청 불리는 방식이다.

- Noise label을 어디서 가져올 것인가? (Noise X 어떤 값)을 해서 노이즈가 합해진 데이터를 생성해낸다.

- Silence : 조용한 데이터 셋(소리없는 상태가 녹음된 wave 파일)도 만들어서 학습 잘 되게 도와준다.

- Test-Train set을 나눠서 Training을 시킨다.
Learning rate 같은 hyperparameter를 조정한다. (여기가 fine tuning)
학습을 시킨다.

Q. 오디오 파일을 어떻게 넣으면 좋을까?

A. 오디오 파일이 존재하면, 모델에 전송해 값을 출력하는 것은 OK.
하지만 ‘딥 포닉스’는 AI 스피커의 특성을 기본적으로 갖고 있기 때문에, 사용자가 실시간으로 말하는 음성을 처리해내야 한다. 이때, 실시간으로 입력되는 명령어를 wave 파일로 저장해야 하는 문제가 발생한다. 모델이 요구하는 파일 형식에 맞춰 넘겨줘야 우리가 학습시킨 모델을 사용할 수 있다.

* 이미 학습된 모델을 Inference만 해야 한다. 사용자가 말하면 듣고, 녹음해서 바로 모델로 전송.

→ 음성을 real time으로 저장할 수 있는 게 있는지?

아니면 1초마다 외부파일을 저장하는 형태?

※ 활용한 캐글 코드 링크

<https://www.kaggle.com/kcs93023/keras-sequential-conv1d-model-classification>

▶ 정보통신연구소 관련 딕셔너리 생성

- 자연어 처리 시, 우리가 신경써야 할 부분 : 풍부한 데이터셋이 중요

→ 청년 AI·빅데이터 아카데미 전용 사전이 필요해 직접 생성했다. 기존 KT API에 내장된 딕셔너리에서 저장되어 있지 않은 생소한 단어에 대한 STT가 잘 안 이루어짐을 확인했다.

정보통신연구소 사이트(<http://pirl.postech.ac.kr/>)의 내용을 우리가 필요한 부분만 선별해 정리하였다. 이를 통해 딥포닉스 사용자가 말한 질의 문장을 제대로 인식해 답변을 출력해 줄 수 있도록 셋팅하였다. 최대한 다양한 문장에 대해 딥 포닉스가 질의응답 가능하도록 다양한 문장형에 대해 고민해 작성하였다.

- 질의응답 기능 개선 : BERT의 한국버전인 KoBERT 기계독해 API 활용

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova

Google AI Language

{jacobdevlin, mingweichang, kentonl, kristout}@google.com

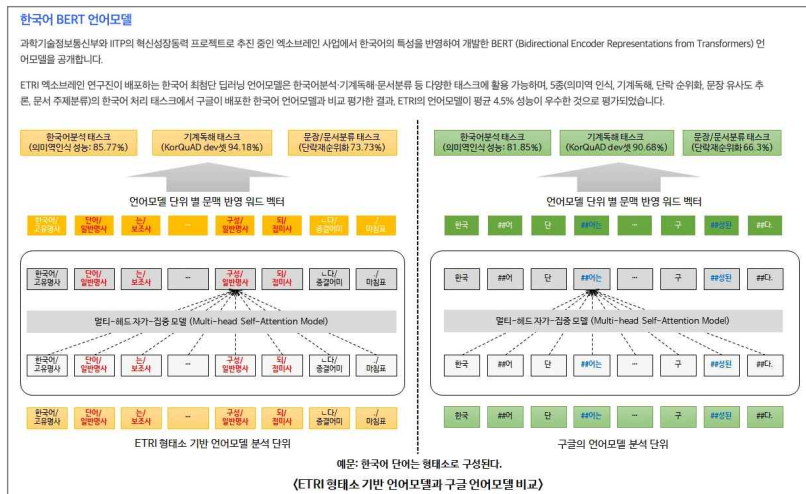
※ 기계독해 API 란?

자연어로 쓰여진 단락과 사용자 질문이 주어졌을 때, 딥러닝 기술을 이용하여 단락 중 정답에 해당하는 영역을 찾는 API 이다.

기계독해 API는 언어분석(예: 개체명 인식) 및 질문분석(예: 질문분류) 기술을 활용하여 더욱 정확한 정답을 제시할 수 있습니다. 현재 위키백과 기반의 단락과 질문에서 학습하였으며 뉴스 등 다른 도메인의 단락과 질문에서도 올바른 정답을 추출할 수 있다.

기계독해 API는 HTTP 기반의 REST API 인터페이스로 JSON 포맷 기반의 입력 및 출력을 지원하며 ETRI에서 제공하는 API Key 인증을 통해 사용할 수 있는 Open API 이다.

- 사이트 링크 : http://aiopen.etri.re.kr/demo_nlu.php



배포 모델	세부 모델	세부 내용	모델 파라미터
KorBERT	Korean_BERT_Morphology	<ul style="list-style-type: none"> 학습데이터: 23GB 원시 말뭉치 (47억개 형태소) 형태소분석기: 본 OpenAPI 언어분석 중, 형태소분석 API 딥러닝 라이브러리: pytorch, tensorflow 소스코드: tokenizer 및 기계독해(MRC), 문서분류 예제 Latin alphabets: Cased 	30349 vocabs, 12 layer, 768 hidden, 12 heads,
	Korean_BERT_WordPiece	<ul style="list-style-type: none"> 학습데이터: 23GB 원시 말뭉치 (47억개 형태소) 딥러닝 라이브러리: pytorch, tensorflow 소스코드: tokenizer Latin alphabets: Cased 	30797 vocabs, 12 layer, 768 hidden, 12 heads,

- 정보통신연구소 디렉터리를 사용해 KoBERT 테스트한 결과 화면

분석 시각화

단락	교로별 ICT 기술을 선도하고, 미래 ICT 기술의 연구개발을 제시한다. 컴퓨터가 인간과 비독을 거두고, 자동차가 스스로 운전할 하며, 모든 것이 서로 연결되어 있어서 스마트폰으로 집안일을 돌보는 세상이 되었습니. ICT 기술이 융합된 기업은 4차라고, 새로운 ICT 기술을 보유한 기업이 출현하여 기존 산업 판도를 크게 뒤집을 것이 예상된다. 정보통신연구소는 AI, Big Data, IoT 기반 미래 핵심 기술을 중점 연구로 ICT 기술이 전 산업 분야와 결합하여 더 빨리, 더 크게 세상을 변화시킬 수 있도록 그 방향을 제시합니다. ICT 기술과 기존 산업체와의 융합을 통하여 창조적 혁신을 주도합니다. ICT 기술은 기존의 모든 산업 영역에서 새로운 가치를 창출하여 창조적 혁신을 만들어내고 있습니다. 정보통신연구소는 교육과 연구의 융합 서비스 효과를 기반으로 한 새로운 산업 협력 모델을 중심으로 산업체와의 적극적인 협력을 주도합니다.
질문	정보통신연구소가 뭐야?
정답	교육과 연구의 융합 서비스 효과를 기반으로 한 새로운 산업 협력 모델을 중심으로 산업체와의 적극적인 협력을 주도
신뢰도	0.557840
단락 내 정답 시작 위치	415
단락 내 정답 마지막 위치	476

○ [코드] 음성 처리 파트

▶ STT부분

```
from __future__ import print_function

import grpc
import gigagenieRPC_pb2
import gigagenieRPC_pb2_grpc
import MicrophoneStream as MS
import user_auth as UA
import audioop
import os
from ctypes import *

HOST = 'gate.gigagenie.ai'
PORT = 4080
RATE = 16000
CHUNK = 512

ERROR_HANDLER_FUNC = CFUNCTYPE(None, c_char_p, c_int, c_char_p, c_int, c_char_p)
def py_error_handler(filename, line, function, err, fmt):
    dummy_var = 0
    c_error_handler = ERROR_HANDLER_FUNC(py_error_handler)
    asound = cdll.LoadLibrary('libasound.so')
    asound.snd_lib_error_set_handler(c_error_handler)

def generate_request():
    with MS.MicrophoneStream(RATE, CHUNK) as stream:
        audio_generator = stream.generator()

        for content in audio_generator:
            message = gigagenieRPC_pb2.reqVoice()
            message.audioContent = content
            yield message

            rms = audioop.rms(content, 2)
            #print_rms(rms)

def getVoice2Text():
    print("\n\n음성인식을 시작합니다.\n\n종료하시려면 Ctrl+\ 키를 누르세요.\n\n\n")
    channel = grpc.secure_channel('{}:{}'.format(HOST, PORT), UA.getCredentials())
    stub = gigagenieRPC_pb2_grpc.GigagenieStub(channel)
    request = generate_request()
    resultText = ''
    for response in stub.getVoice2Text(request):
        if response.resultCd == 200: # partial
            print('resultCd=%d | recognizedText= %s' % (response.resultCd, response.recognizedText))
            resultText = response.recognizedText
        elif response.resultCd == 201: # final
            print('resultCd=%d | recognizedText= %s' % (response.resultCd, response.recognizedText))
            resultText = response.recognizedText
            break
        else:
            print('resultCd=%d | recognizedText= %s' % (response.resultCd, response.recognizedText))
            break

    print("\n\n인식결과: %s \n\n\n" % (resultText))
    return resultText

def main():
    # STT
    text = getVoice2Text()

if __name__ == '__main__':
    main()
```

- (1)음성인식을 위해서 라이브러리 호출을 하면 음성 데이터를 입력하고 결과를 수신할 수 있는 객체가 전달된다.
- (2)error 이벤트, data 수신 이벤트, end 종료를 처리하기 위한 콜백 함수를 구현한다.
- (3)data 수신 이벤트에서는 음성인식된 결과가 전달 된다.
- (4)음성인식 설정값을 KT API 서버로 전달한다.
- (5)설정값을 전달하면 음성인식이 시작된다.
- (6)마이크에서 데이터가 생성되면 음성인식 데이터를 입력한다.

▶ TTS부분

```
import grpc
import gigagenieRPC_pb2
import gigagenieRPC_pb2_grpc
import MicrophoneStream as MS
import user_auth as UA
import os
from ctypes import *

HOST = 'gate.gigagenie.ai'
PORT = 4080

ERROR_HANDLER_FUNC = CFUNCTYPE(None, c_char_p, c_int, c_char_p, c_int, c_char_p)
def py_error_handler(filename, line, function, err, fmt):
    dummy_var = 0
    c_error_handler = ERROR_HANDLER_FUNC(py_error_handler)
    asound = cdll.LoadLibrary('libasound.so')
    asound.snd_lib_error_set_handler(c_error_handler)

# TTS : getText2VoiceStream
def getText2VoiceStream(inText, inFileName):
    channel = grpc.secure_channel('{}:{}'.format(HOST, PORT), UA.getCredentials())
    stub = gigagenieRPC_pb2_grpc.GigagenieStub(channel)

    message = gigagenieRPC_pb2.reqText()
    message.lang=0
    message.mode=0
    message.text=inText
    writeFile=open(inFileName, 'wb')
    for response in stub.getText2VoiceStream(message):
        if response.HasField("resOptions"):
            print("\n\nResVoiceResult: %d" % (response.resOptions.resultCd))
        if response.HasField("audioContent"):
            print("Audio Stream\n\n")
            writeFile.write(response.audioContent)
    writeFile.close()
    return response.resOptions.resultCd

def main():
    output_file = "./testTTS/testtts.wav"
    getText2VoiceStream("죄송하지만 알아듣지 못했어요. 다시 말씀해주시겠어요?", output_file)
    MS.play_file(output_file)
    print( output_file + "이 생성되었으니 파일을 확인바랍니다. \n\n\n")
```


- (1)TTS 스트림 데이터 출력을 위해서 speaker 모듈을 가져오고 생성한다.
- (2)합성하고자 하는 텍스트를 설정해서 API 를 호 출한다.
- (3)TTS 스트림을 받을 때 오류가 발생할때 호출되는 함수 설정이다.
- (4)TTS Stream 데이터가 전송될때 처리하는 코드인데, audioContent 가 수신되면 (1)에서 생성한 speaker 객체에 write 한다.
- (5)PCM 데이터가 전송 완료되었을 때 처리하는 코드를 콜백으로 등록한다.
- (6)PCM 데이터가 출력이 완료되기를 기다린다..

▶ NLP 부분

<pre>import ex2_getVoice2Text as stt import ex4_getText2VoiceStream as tts import MicrophoneStream as MS</pre>	STT, TTS 호출
<pre># first category pirl = ['정보통신연구소', '정보통신', '연구소', '정보 통신'] edu_1 = ['기업맞춤형', '기업'] edu_2 = ['지역기업 인재양성', '지역기업', '인재양성'] edu_3 = ['에이아이 빅데이터 아카데미', '에이아이', '빅데이터'] home = ['숙소', '잠은', '점자는'] eat = ['식당', '밥', '밥먹는'] rest = ['화장실', '화장'] study = ['휴게실', '스터디룸'] corp = ['협력업체', '협력기관', '협력'] vision=['vision', '비전', '위전'] teacher=['강수현선생님', '강수현', '수현', '강수'] # second category where = ['어디', '위치', '장소'] who = ['누구'] what = ['소개', '뭐', '무엇', '알려', '뭐야'] edu = ['교육', '과정'] when = ['언제', '설립년도'] # 이 부분 답 수정해주고, 위에 카테고리 좀 더 세부화할 사람 필요함 - 이게 원배랑 # 나랑 얘기하던 하드코딩 부분임 answer = [['경상북도 포항시 남구 청암로 77 포항공과대학교 정보통신연구소입니다', '서영주 소장님 입니다', '잘 모르겠어요. 다른 걸 물어보시겠어요', '잘 모르겠어요. 다른 걸 물어보시겠어요', '기업 니즈', '잘 모르겠어요. 다른 걸 물어보시겠어요', '잘 모르겠어요. 다른 걸 물어보시겠어요', '지역기업의', 'rist 연구4동 국제관 박대준학술정보관입니다', '잘 모르겠어요. 다른 걸 물어보시겠어요', '정보', '경상북도 포항시 남구 지곡로 120 포스코인재창조원에 있습니다', '잘 모르겠어요. 다른 걸 물어', 'rist식당입니다', '잘 모르겠어요. 다른 걸 물어보시겠어요', '구글에 rist식당검색하면 식당표', '1층에 2곳 있습니다', '잘 모르겠어요. 다른 걸 물어보시겠어요', '잘 모르겠어요. 다른 걸 물어', '1층에 1곳 4층에 2곳있습니다', '잘 모르겠어요. 다른 걸 물어보시겠어요', '잘 모르겠어요.', '잘 모르겠어요. 다른 걸 물어보시겠어요', '잘 모르겠어요. 다른 걸 물어보시겠어요', '포스코, 3', '잘 모르겠어요. 다른 걸 물어보시겠어요', '잘 모르겠어요. 다른 걸 물어보시겠어요', '4차 산업', '정보통신연구소 211호 입니다', '에이아이빅데이터교육을 담당해주시는 친절한 선생님입니다', ''] no_answer = '알아들지 못했어요, 다시 말씀해주세요?' cat1 = [pirl, edu_1, edu_2, edu_3, home, eat, rest, study, corp, vision, teacher] cat2 = [where, who, what, edu, when]</pre>	NLP 부분

<pre>while 1: while 1: flag = False input_text = stt.getVoice2Text() for name in ['포닉스', '올릭스', '코닉스', '코맥스', '보닉스', '오릭스']: if name in input_text: MS.play_file('../data/ponix_greeting.wav') flag = True break if flag: break</pre>	호출어 부분
<pre>while 1: score = [0] * len(cat1) score2 = [0] * len(cat2) reply = False query = stt.getVoice2Text() for idx, category in enumerate(cat1): for subcat in category: if subcat in query: score[idx] += 1 for idx, category in enumerate(cat2): for subcat in category: if subcat in query: score2[idx] += 1 if sum(score) == 0: MS.play_file('../data/ponix_not_understood.wav') continue if sum(score2) == 0: MS.play_file('../data/ponix_not_understood.wav') continue topic = score.index(max(score)) topic2 = score2.index(max(score2)) #print(topic, topic2, type(topic), type(topic2), len(answer), len(answer[0])) reply = True tts.getText2VoiceStream(answer[topic][topic2], 'initial_response.wav') MS.play_file('initial_response.wav') if reply: break</pre>	질문과 답매칭

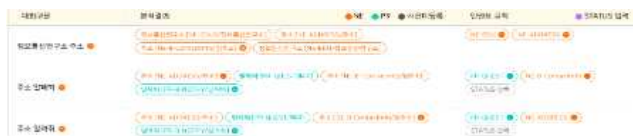
- 1) 호출어부분 : 기존의 KT API 코드파일로는 호출어를 우리가 원하는 '포닉스' 로 불가능하다는 것을 인지하여, 코드로 대체하였다.
- 질문 및 답변 부분 : STT부분과 TTS부분의 코드를 활용하여 원하는 질문의 Tokenization > Normalization > Indexing > Dictionary 작업을 통해 각각의 질문에 대응되는 답을 찾아주는 코드이다.

*필수는 아니지만, 미리 만들어 놓으면 아래 ‘3)’ 질문-답변 대응시킬 때 편리하다.

id	질문	답변
1	정보통신연구소가 뭐니	1991년 설립된 연구기관으로서, 인공지능, 빅데이터, 사물인터넷
2	정보통신연구소 교육 소개해줘	기업맞춤형 교육, 지역기업 인재양성 교육, ai-bigdata 아카데미
3	정보통신연구소 소장님	서영주 소장님입니다.

→ 3가지 방법으로 시도해 본 후, 그 중 성능이 가장 좋은 아래 ②, ③로 선택했다.

문제점: 답변이 잘려서 나오거나 제대로 나오지 않는다.



예를 들어, ‘정보통신연구소 어디야’라는 질문이 들어오면 first category에서는 pirl이 선택될 것이고, second category에서는 where이 선택될 것이다. 따라서 (pirl,where)=(1,1)의 답변인 ‘경상북도 포항시 남구 청암로 77 포항공과대학교 정보통신연구소입니다.’가 출력된다. (코드 설명은 [아래](#) 참고)

```
# first category
pirl = ['정보통신연구소', '정보통신', '연구소', '정보 통신']
edu_1 = ['지역기초대학', '기초대학', '지역기', '지역기초']
edu_2 = ['지역기초대학', '지역기초', '지역기', '지역기초']
edu_3 = ['에이마이데미아카데미', '에이마이', '에이마이', '에이마이']
home = ['숙소', '숙소', '숙소', '숙소']
eat = ['식당', '식당', '식당', '식당']
rest = ['환락장', '환락장', '환락장', '환락장']
study = ['대학', '대학', '대학', '대학']
corp = ['현역업체', '현역업체', '현역업체', '현역업체']
vision = ['vision', 'vision', 'vision', 'vision']
teacher = ['강수현', '강수현', '강수현', '강수현']

# second category
where = ['먼디', '위치', '장소']
who = ['누구']
what = ['소개', '뭐', '무엇', '알려', '뭐야']
edu = ['교육', '과정']
when = ['연제', '출판년도']

|
answer = [
    '정보통신연구소', '정보통신', '연구소', '정보 통신', '지역기초대학', '기초대학', '지역기', '지역기초', '에이마이데미아카데미', '에이마이', '에이마이', '에이마이', '숙소', '숙소', '숙소', '숙소', '식당', '식당', '식당', '식당', '환락장', '환락장', '환락장', '환락장', '대학', '대학', '대학', '대학', '현역업체', '현역업체', '현역업체', '현역업체', 'vision', 'vision', 'vision', 'vision', '강수현', '강수현', '강수현', '강수현', '먼디', '위치', '장소', '누구', '소개', '뭐', '무엇', '알려', '뭐야', '교육', '과정', '연제', '출판년도']
]

no_answer = '알아들지 못했어요, 다시 말씀해주실래요?'
cat1 = [pirl, edu_1, edu_2, edu_3, home, eat, rest, study, corp, vision, teacher]
cat2 = [where, who, what, edu, when]
```

③ KorBERT

※ 링크: <http://aiopen.etri.re.kr/>

-KoBERT에서 ‘언어처리’ api를 사용하였다.

문장을 분석하고 질문에 대한 답변을 찾아주기 때문에 문장으로 데이터셋(txt파일)을 만들었다.

데이터셋이 크면 문장에 대한 답변을 찾는 데 시간이 오래 걸려 아래와 같이 카테고리별로 나눴다.

[정보통신연구소 category]

정보통신연구소의 주소는 경상북도 포항시 남구 청암로 77 포항공과대학교 입니다.

정보통신연구소 소장은 서영주 소장님입니다.

정보통신연구소에서 가장 가까운 식당은 포항산업과학연구원 식당입니다.

정보통신은 1991년 설립된 연구기관으로서, 인공지능, 빅데이터, 사물인터넷 등의 각 분야 전문가들로 구성된 연구

정보통신연구소는 기업 맞춤형 교육, 지역기업 인재양성 교육, ai big data 아카데미 교육을 하고 있습니다.

정보통신연구소의 연구부문은 머신러닝, 비전엔지니어링, 빅데이터, 알고리즘 등을 연구하고 있습니다.

정보통신연구소는 현재 인공지능 및 빅데이터 분야를 연구하는 'AIBD 센터'와 사물인터넷 분야를 연구하는 'IoT 센

정보통신연구소의 협력기관은 포스코, 포스코인재창조원, 삼성전자, 에스케이하이닉스 등 여러 기업이 있습니다.

[aibigdata category]

청년에이아이빅데이터 교육장소는 1층입니다.

청년에이아이빅데이터의 담당자는 배강국교수님, 강수현선생님입니다.

강수현선생님의 사무실은 정보통신연구소 211호 입니다.

강수현선생님은 에이아이빅데이터교육을 담당해주시는 친절하신 선생님입니다.

배강국교수님 매일 든든한 간식을 드실 수 있습니다.

6기 선배들은 멋진 선배들입니다.

○ 영상 처리 파트

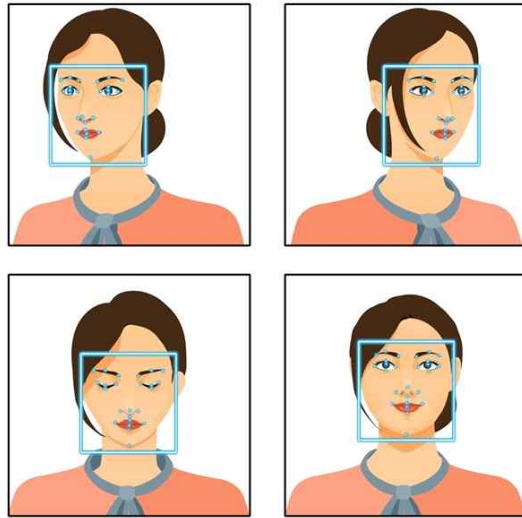
▶ CNN을 이용한 얼굴 분류기

학습한 얼굴의 경우, 사람처럼 알아보고 인사하는 기능
사람 알아보고, “누구님, 안녕하세요!”

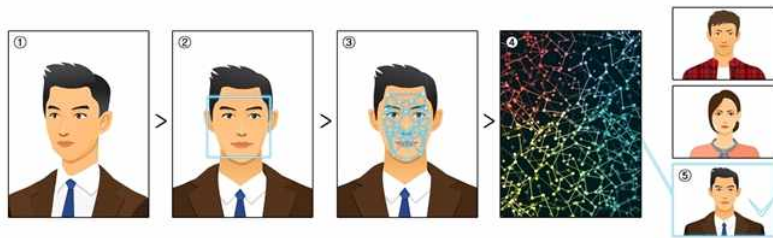
- CNN을 이용한 얼굴 분류기: 얼굴 인식 기술을 활용해 먼저 인사 건네는
딥 포닉스

※ 5기 YOLNIR 프로젝트 참조





◇일일이 손으로 찍던 특징점, 이제 AI가 대신



2D 얼굴인식 기술 적용 단계. (1)영상인식 (2)얼굴부위 탐지 (3)100여개 점 찍어 얼굴 특징 인지 (4)AI 분석 데이터화 (5)데이터 베이스와 대조. 에스원 제공



- 얼굴 분류기 모델에 A반 7명의 얼굴 사진 총 210을 학습시켰고, 학습시키지 않은 얼굴을 인식할 경우 ‘Unknown’ 으로 표시하도록 하였다.




```
shortcut = from -3 / activation_function = linear
```

```
cv6 = filter * 64 / activation_function = leaky_relu
```

```
shortcut = from -3 / activation_function = linear
```

yolo layer

등 108개의 layer로 이루어짐

[illegible]

- YOLO를 실행시키기 위해서는 Darknet이 필요하다. Darknet은 Joseph Redmon이 독자적으로 개발한 신경망 프레임워크(neural network framework)로서 dnn(deep neural network)들을 학습시키고 실행시킬 수 있는 틀(framework)이다. 그리고 yolo는 학습된 신경망(결과물) 중 하나이다. Darknet을 이용하면 yolo 뿐만 아니라 AlexNet, VGG-16, Resnet, Densenet 등 기존의 정통 주류의 dnn(deep neural network)들도 돌려 볼 수 있다.

– 26 –

– 25 –

```
cv1 = filter * 32 / activation_function = leaky_relu
```

```
cv3 = filter * 32 / activation_function = leaky_relu
```

```
cv4 = filter * 64 / activation_function = leaky_relu
```


Darknet과 yolo는 그 코드가 모두 공개되어 있으며 누구라도 사용할 수 있다.

※ Darknet & YOLO 사이트: <https://pjreddie.com/darknet/>

1. 서론

- 본 실험을 통해 CNN기수에 대한 이해와 인물 얼굴 분류기를 제작해 보려 한다.
- 본 실험은 배운 이론을 바탕으로 데이터 수집부터 keras을 이용해 구현까지 직접 수행해 보았다.

2. 실험 관련 이론

- 일반적인 FCNN과 CNN의 가장 큰 차이점이라 함은 이미지 인식에 있어 이미지의 모든 픽셀을 분석하지않고 지역적, 공간적 상관관계를 고려한 학습을 한다. 따라서 효과적으로 Data-representation을 수행하고 이를 통해 분류 작업에 있어 높은 성능을 보인다.
- 실험에서 단순 층계수, 노드수의 hyper-parameter말고도 optimizer로 Adam, activation function으로 relu/softmax, loss function으로 cross-entropy를 지정하여 수행하였다.

3. 실험방법 및 수행

- 데이터 전처리 : web crawler를 통해 얻은 이미지를 openCV API를 통해 얼굴만 cropping하여 데이터를 확보하였다.
- 모델 설계 : 분석을 위해 최소한의 CNN구조를 2conv 계층, 1fcnn 계층으로 구성하였다.
- 학습 과정 : over fitting을 방지하기 위해 early stopping을 하였다.

4. 결과 및 토론

- 평가 및 테스트 : test-set 기준 정확도 90%의 성능을 가진다
- 네트워크 분석 : 학습된 weight와 feature map 시각화를 통해 CNN의 원리를 분석해보려 한다.

Q. 왜 서버가 아닌 라즈베리파이에서 영상을 처리하는가?

A. 기존에 OpenCV를 사용하여 라즈베리파이가 아닌 서버에서 영상을 처리했지만, 테스트 개발환경에서 소켓 통신을 통한 스트리밍 후 프레임드롭과 (영상의 프레임이 떨어져서 영상이 뚝뚝 끊겨보이는 현상) 종종 생기는 화이트스크린(영상에 흰 화면만 나타나는 현상, 디버그를 위해서 검색해봤지만 정확한 이유는 찾을 수 없었음)이라는 문제점을 가지고 있었음.

하지만 이 문제점은 라즈베리파이 자체에서 영상을 처리할 때는 확인되지 않았고, 또한, 라즈베리파이에서 서버로 스트리밍하는 과정에서 어차피 영상을 찍고, 이것을 프레임단위로 서버로 보내주는 프로세스가 기존에 이미 있었기 때문에, 그렇다면 우리가 사용하는 두 개의 라즈베리파이 중 영상을 담당하는 기기가 영상 처리까지 맡을 수 있지 않을까? 라고 생각하게 되었음.

라즈베리파이 자체가 저전력, 저성능, 저용량 기기이기에 가벼운 모델과 올인원 라이브러리를 사용할 필요성이 있어서, 이 조건에 맞는 OpenCV의 built-in모델과 Raspberry Pi Face Recognition으로 알려진 모델을 사용했다.

▶ 디스플레이 확장 → 화면 달린 AI 스피커 : [청각 + 시각]

- 듣는 AI 스피커에서 보는 제품으로 → 나아가 움직이는 AI 스피커로 진화
- 接客 로봇으로 발전 가능

II. 프로젝트 수행결과

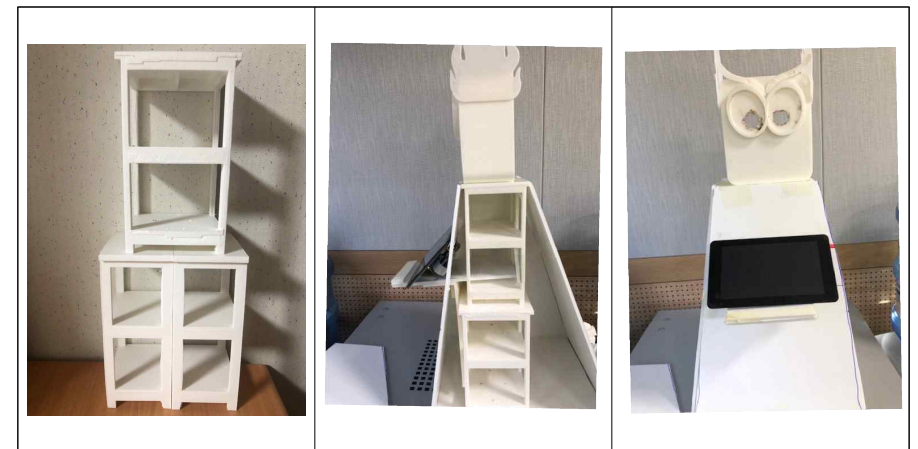
가. 프로젝트 개발환경

구분		내용
S/W 개발환경	OS	영상인식 및 디스플레이 : Raspbian OS 음성인식 : GIGA Genie AI-Makers-kit OS
	개발환경	Ubuntu 16.0.4
	개발언어	Python(2.7, 3.5, 3.7)
H/W 구성장비	디바이스	RaspberryPi, KT AI-Makers-Kit, DeskTop, Raspberry Pi Display, Pi camera, Replicator2(3D프린터)
	통신	HTTP, GRPC
	개발언어	Python(2.7, 3.5, 3.7)

구분		내용	
음성 인식	S/W	KT AI Server	음성인식, 대화해석, 음성합성 서비스처리
	H/W	KT AI-Makers-Kit	Mic : Speaker Voice 인식
			Speaker : Analog Signal 출력
			Analog Signal을 Digital Voice Signal로 변환 및 Raspberry-Pi와 통신
영상 인식	S/W	Open CV	Builtin 모델 사용 및 등록된 사용자 얼굴 인식
		YoLo	Darknet 사용을 통해 등록된 사용자 얼굴 인식
	H/W	Raspberry Pi	Pi Cam과 S/W와 연결, 무선랜을 통한 소켓통신, 영상처리
		Pi Cam	영상인식
디스플레이	S/W	Python Flask Framework	음성인식 및 답변 정보를 Display에 출력

	H/W	Raspberry-Pi Display	Raspberry-Pi에 mount하여 사용
3D 모델링	S/W	123D	3D도면 작업
		Makerbot	123D도면을 해당 Replicator2 프린터와 호환되는 파일로 변환
	H/W	Replicator2	모델링 된 도면을 출력

나. 결과물 상세 이미지



○ 하드웨어 제작 과정

▶ 준비물

베이스 (30x30)

3D 프린트 (몸통 뼈대 大:2 , 小: 2), (얼굴: 눈1, 눈썹2, ...)

전동드릴

라즈베리파이 키트 2세트

아두이노 키트 2 세트

▶ 중요 피쳐

LV 0. 자연어 처리 - (STT, TTS, API 사용 가능)

PIRL 소개

LV 1. HW - 움직이게 하기, 센서

움직이는 PIRL 소개

LV 1-1. 얼굴 인식 - 물체 인식, face recognition

직원보고 인사하는, 움직이는 PIRL 소개

LV 2. 소리 방향 파악

이름 듣고 돌아보고, 직원보고 인사하는, 움직이는 PIRL 소개

LV 3. 위치 추적

에스코트하는, 이름 듣고 돌아보고, 직원보고 인사하는, 움직이는 PIRL 소개

1) 도면제작

2) 3D 프린팅 얼굴 -> 몸통

3) 프린팅 한 것을 환경에 맞게 제작

ex) 카메라를 달기 위해 얼굴 눈쪽에 구멍을 내서 카메라가 필요할 수 있

도록 제작

4) 우드 보드로 외형 프레임 제작

5) 디스플레이 장착



o 3D 프린팅 장비 특징



Replicator 2 #1 (PLA)

- 대분류 3D프린터
- 중분류 Makerbot
- 제조사 1) 온라인 예약
- 이용방법 2) 신규 이용자 평일18:00이후, 휴일13:00이후 장비 예약 가능 (장비연도 상주시간)
- 신규 이용자 장비교육이 필수이며, 주간 장비 이용이 불가능합니다.
- 신규 이용자 문의 postech.makers@gmail.com으로 문의 바랍니다.
- 장비이용시간은 공간방문시간이 아닌 실 장비이용 시간입니다.
- 장비이용시작후 15분 이내 미방문시 노쇼처리 됩니다.
- 사용후 만족도 평가 작성 및 장비 이용 결과보고서 제출 (이메일 확인)
- 사용연령 제한 중학생 이상
- 이용료 5,000원/회
- 장비설명 작업가능 사이즈: 285 x 153 x 155 mm (가로 x 세로 x 높이)
- 필라멘트: PLA, 1.75 mm
- 노즐 지름: 0.4 mm

예약하기



- 3D 프린터를 활용한 하드웨어 제작과정

: 메이크올 무한상상실(<https://www.makeall.com/>)을 이용해 덩 포닉스 하드웨어를 프린팅하기로 하였다. 무한상상실을 이용하기 위해서는 처음 이용 시 교육을 받아야 하고, 3D 도면을 USB에 담아가야 한다. 우리 조원 중 3D 도면을 그릴 수 있는 인원이 없어 다른 조 팀원에게 캐드를 활용한 3D 도면을 부탁하였다. 3D 프린팅 교육 날, 우리의 도면대로 12등분 조각을 출력한 후 조립하게 되면 내구성이 약해진다는 피드백을 받았다. 덩 포닉스의 내구성을 갖추기 위해서 레고처럼 한쪽은 들어가고 한쪽은 나오는 식으로 조립형식(레고식)으로 도면을 출력을 해야 했다. 교육 멘토님이 디자인 기본 툴 중 무료이며 초급자가 다루기 쉬운 123D 프로그램을 추천해주셨다. 그 프로그램으로 도면을 그리려고 시도하였으나, 작업과정이 비효율적이고, 시간이 많이 소요되어 하드웨어 전체 프린팅에서 부분 프린팅으로 전환하였다. Thingiverse라는 사이트에서 얼굴 부분에 해당하는 도면을 구했다. 우리가 원하는 포닉스 도면은 없었지만, 다양한 3D 도면이 존재한다. 이 사이트에서 포닉스의 특징을 최대한 살릴 수 있는 부분을 찾아 부위별로 출력을 진행하였다. 몸통 부분은 처음에 아크릴을 고려하였으나, 배송기간/ 가격 등을 고려해 우드보드를 선택해 제작하였다.

※ 3D 프린팅을 활용한 하드웨어 제작 시 TIP

우리는 처음에 바퀴달고 총 높이가 90cm를 예상하고 제작하려고 하였으나, 3D프린팅 장비 사이즈도 그렇고, 높이 20cm 하는 것도 대부분 뽑는 시간이 12시간이 넘어가기 때문에 장비를 많이 사용할 수 없었던 점과 장비를 사

용하는데 있어서 여러 실패도 있었기 때문에 많은 시간을 소비하였다. 그래서 시간적 여유가 있을때 빨리 3D프린팅 장비사용을 마무리 하는 것이 좋다. 그리고 큰 것을 뽑을 경우 밤에 출력해서 그 다음날 찾으러 가는 것이 더 좋다고 느꼈다.

III. 프로젝트 수행방법

가. 업무분담

번호	성명	역할	담당업무
1	김원배	팀 장	라즈베리 To 우분투서버 통신환경 만들기 아두이노로 바퀴굴리기 소스코드 알아보기 [다크넷 깔기, 디스플레이에 쓰일 웹]
2	연선경	팀 원2	- 딕셔너리 제작 / 보고서 작성 / api 사용법 숙달(구글, KorBERT)
3	오가영	팀 원3	- 보고서 작성 / api 사용법 숙달(구글, KorBERT) /
4	이재윤	팀 원4	- STT 학습 Process / 음성 데이터 -> 모델링 -> 학습 -> 텍스트
5	이준영	팀 원5	- 프로토타입을 위한 3D Printing , 외형제작, 물품 구매
6	최우혁	팀 원 6	- 구글 api, mic -> STT 영상처리, 서버-라즈베리파이 소켓통신

나. 문제점 및 해결방안

○ 프로젝트 관리 측면

- 기술을 배우면서, 바로 적용해 프로젝트를 진행하는 어려움
- 프로젝트에 필요한 적정 기술을 원활하게 서치하기 어려움, 논문 해석이 어려움
- 오픈소스에 있는 코드를 우리 프로젝트에 적합하게 변경하는 데 어려움이 있음

- 개발 역량에 개인차가 있음 → 프로그래밍 능력 차이로 인해 개발 파트가 일부 팀원에 몰리는 경향 있음

○ 작품 개발 측면

▶ STT-TTS 모델 활용시 문제점

첫째, STT 파트 → 우리가 어떤 문장을 말했을 때, 스피커가 정확한 문장을 입력받지 못하는 경우

ex) 유저가 ‘정통원(정보통신연구원)’ 이라고 스피커에 말함 → 스피커에서 스피치를 텍스트로 변환시킬 때 ‘정통원’ 이라고 바로 인식하는 경우 매우 드뭄. [정통, 정통 년, 정동원 등] 다양한 텍스트로 입력되는 케이스

해결법 : [정통, 정통 년, 정동원 등등등] ‘정통원’ 이라는 말을 스피커가 인식할 때, 입력받는 여러 가지 케이스를 생성해 저 리스트에 해당하는 입력값은 ‘정통원’ 에 해당한다고 학습시켜주는 방법

활용기술 : RNN, 문장 생성기, 어린왕자 데이터 셋을 학습시켜 챗봇을 만드는 모델

※ 텐서플로우(TensorFlow)를 이용해서 글자 생성(Text Generation) 해보기
- Recurrent Neural Networks(RNNs) 예제 - Char-RNN

<https://github.com/golbin/TensorFlow-Tutorials/tree/master/10%20-%20RNN/ChatBot>

<http://solarisailab.com/archives/1620>

둘째, 스피커가 말을 입력받아 텍스트로 변환했는데 학습한 데이터 셋에 해당되지 않는 경우

ex) 딥 포닉스에 “환을 알려줘” 라는 말을 했을 때, [환을 알려줘]로 텍스트로 입력받았으나, 저 문장에 학습한 대답 데이터가 없기 때문에 NA 값이 출력되는 경우

해결법 : 학습데이터를 최대한 다양하게 만들어줘야 함 → 시간적 한계

셋째, 스피커가 제대로 된 문장을 입력받음. 이와 유사한 질문과 답변이 학습 데이터에 존재하나, 스피커가 형태소 분석을 제대로 하지 못해 입력문장과 유사한 문장임을 인지하지 못해 NA 값을 출력하는 경우

ex) ‘정보통신연구소 어디야’ 라는 질문과 ‘해당 답변~~~’ 을 쌍으로 학습시킴. ‘정보통신연구소 어디야’ 와 유사한 ‘정보통신연구원 어디야’ 를 입력받으면, 해당 답변을 못찾는 문제 → 유사 질문끼리 매칭시켜 주는 방법 고민해야 함

해결법 :

- 1) 기가 지니 Dialog kit 활용해 디셔너리 생성 : 메인 질문-유사 질문 셋을 수작업으로 만들어, STT 모델에 적용하는 방법
- 2) 기가 지니 STT모델을 개선시키는 방법 . 질문을 입력받으면, 형태소 분석기에 넣어 형태소를 추출하고, 형태가 조금씩 다른 문장들은 동일한 문장임을 인식하도록 셋팅함

활용기술 : 카카오톡 형태소 분석기

※ khaiii는 “Kakao Hangul Analyzer III”의 첫 글자들만 모아 만든 이름으로 카카오에서 개발한 세 번째 형태소분석기

<https://github.com/kakao/khaiii>

※ 참고자료: 한국어 음성 인식 모델 구현 시 중요한 점

- language 모델부터 만들어야 한다

<https://m.blog.naver.com/PostView.nhn?blogId=fastcampus&logNo=221181060609&proxyReferer=https%3A%2F%2Fwww.google.com%2F>

IV. 기대효과 및 활용분야

○ 안내로봇 →接客로봇 → 인간과 소통 가능한 감성 로봇

※ 우리가 의도한 큰 프로세스

- 사람 알아보고 다가간다 (사람인지 아닌지 여부)
- 학습된 얼굴을 인식해, 아는 체를 한다(사람처럼 얼굴 인식)
- 실내 길을 안내한다 (네비게이션처럼 : 실내 매핑 기술 & 실내 자율주행)
- 이동 중 중간에 유저가 말을 걸면, 멈춰 대답을 한다 (*인천공항 안내로봇과의 차별점 → 길 안내 서비스가 끝나면 항상 start point로 이동함)

V. 참고자료

가. 참고 및 인용자료

- 포스코경영연구원, 2017, <지능형 로봇 활용, 어디까지 왔나?>
- ITFIND, <서비스 로봇 시장 동향>
- 융합연구정책센터소셜, <로봇(Social Robot) 산업동향>
- 한국로봇산업협회, <한·중·일 로봇 산업 및 정책 동향>
- KT AI 메이커스 키트 매거진
- 한국심리학회, 2018, <인공지능(AI) 스피커와의 상호작용이 소비자 심리에 미치는 영향>

- KEIT 김돈정PD, <음성인식 가상비서 기술 동향 및 전망>
- <음성 언어 처리 기술, 어디까지 왔나>,
https://www.korean.go.kr/nkview/nklife/2017_4/27_0405.pdf
- <etri 언어분석 API> , http://aiopen.etri.re.kr/demo_nlu.php
- <텐서플로우(TensorFlow)를 이용해서 글자 생성(Text Generation) 해보기>,
<http://solarisailab.com/archives/1620>
- <https://github.com/kakao/khain>
- <https://github.com/golbin/TensorFlow-Tutorials/tree/master/10%20-%20RNN/ChatB>