프로덕션 환경에서 연구하기

Sungjoo Ha

Hyperconnect

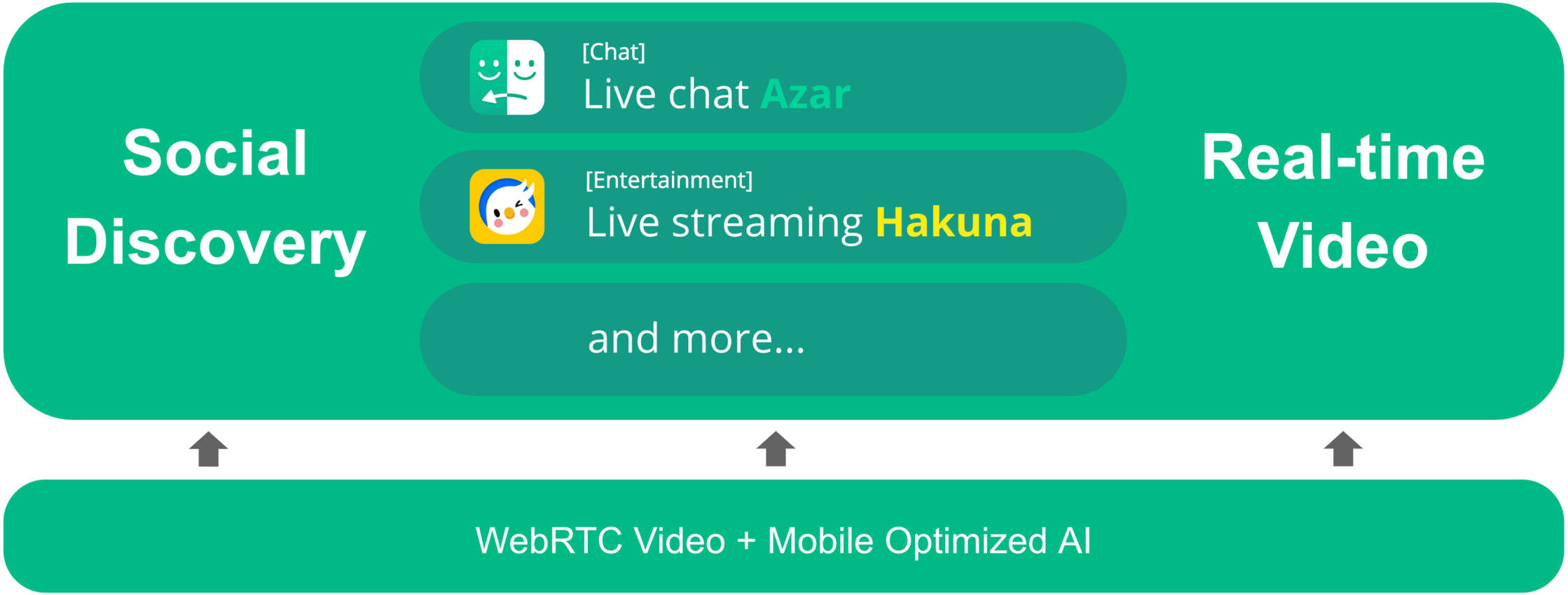
TensorFlow-KR 세 번째 오프라인 모임

October 20th, 2019

오늘의 이야기

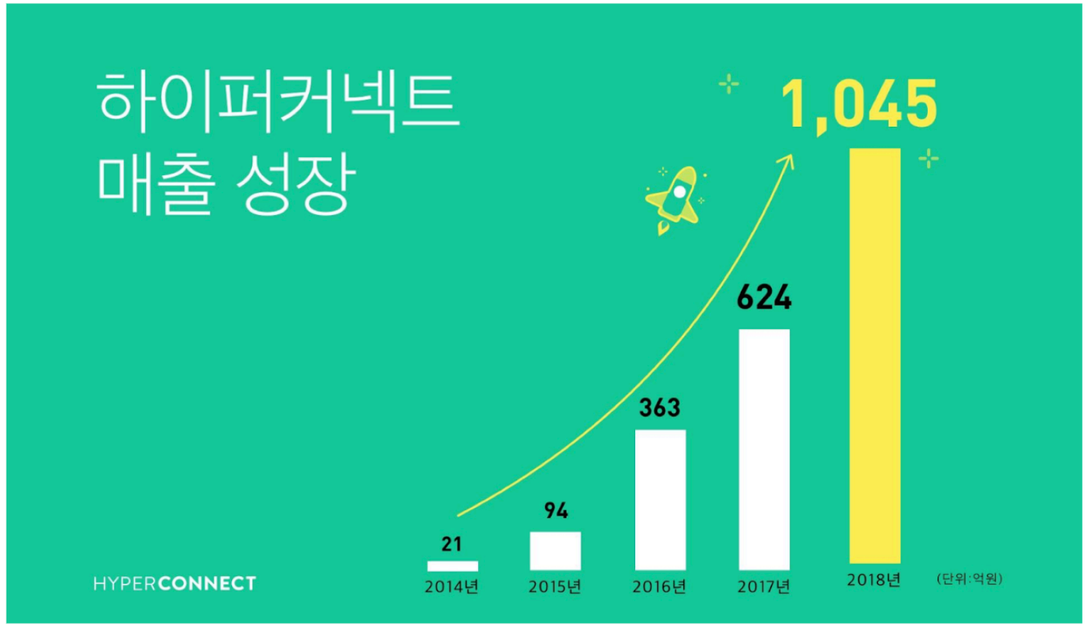
- 프로덕션서비스 중인/될 제품과 연구성공이 불확실한 기술 개발의 이야기
  - 팀에서 수행한 Keyword Spotting 연구의 예제를 중심으로
- 하이퍼커넥트 AI lab 이 일하는 방식의 이야기
- 회사와 팀과 팀원 사이의 합을 맞추는 이야기

Hyperconnect



Hyperconnect

5년 간의 매출 추이



2017 : 9th

Google Play Revenue

1	Tinder
2	Google Drive
3	LINE
4	Pandora
5	HBO NOW
6	Netflix
7	BIGO LIVE
8	KakaoTalk
9	Azar
10	LINE Manga

2018 : 7th

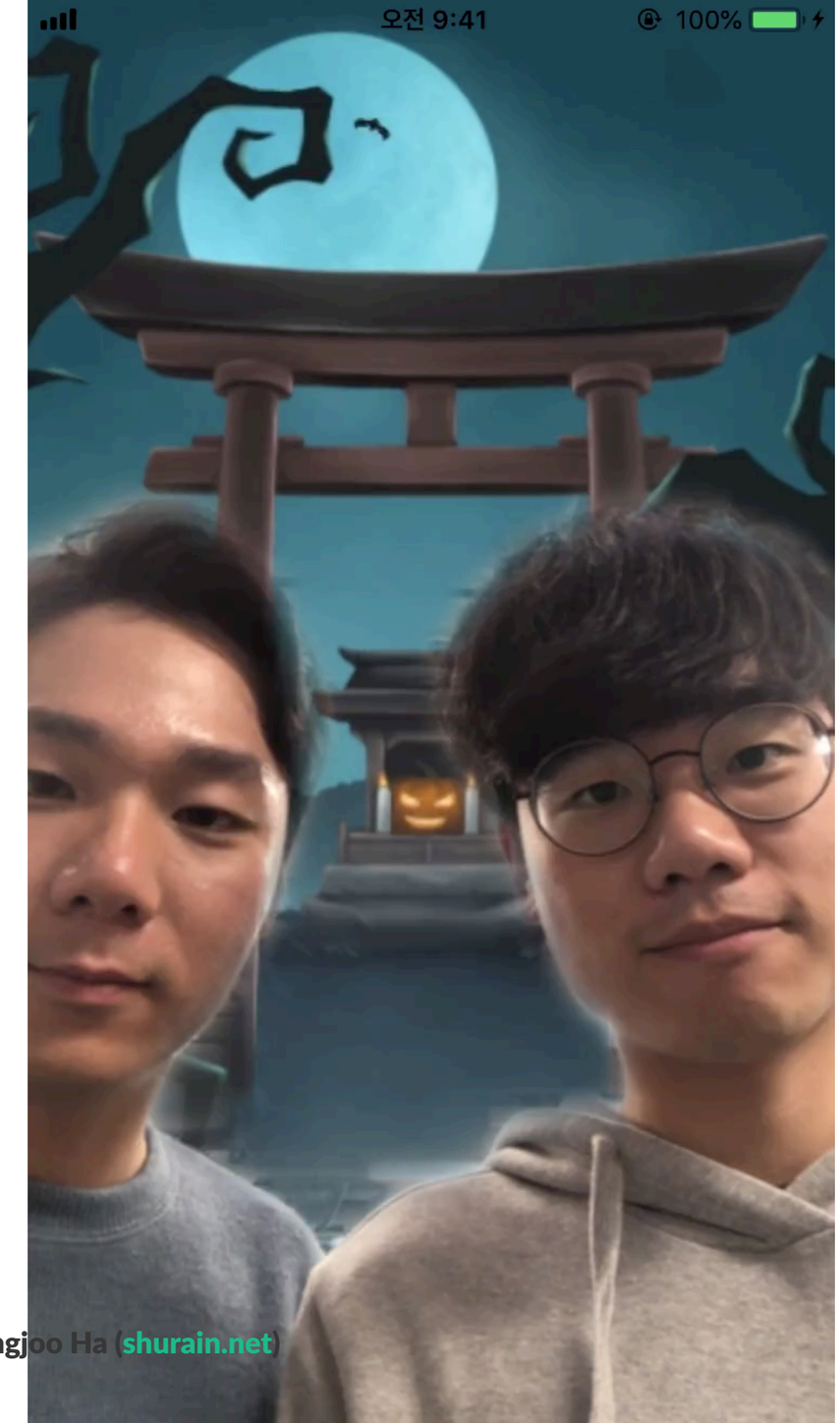
1	Tinder
2	Google Drive
3	Pandora
4	LINE
5	BIGO LIVE
6	Netflix
7	Azar
8	KakaoTalk
9	LINE Manga
10	Google One

2019 1Q : 5th

1	Tinder
2	Google One
3	Pandora
4	BIGO LIVE
5	Azar
6	LINE
7	Netflix
8	LINE Manga
9	KakaoTalk
10	Google Drive

Note: Does not include revenue from third-party Android stores in China or other regions.

SensorTower Data That Drives App Growth sensortower.com



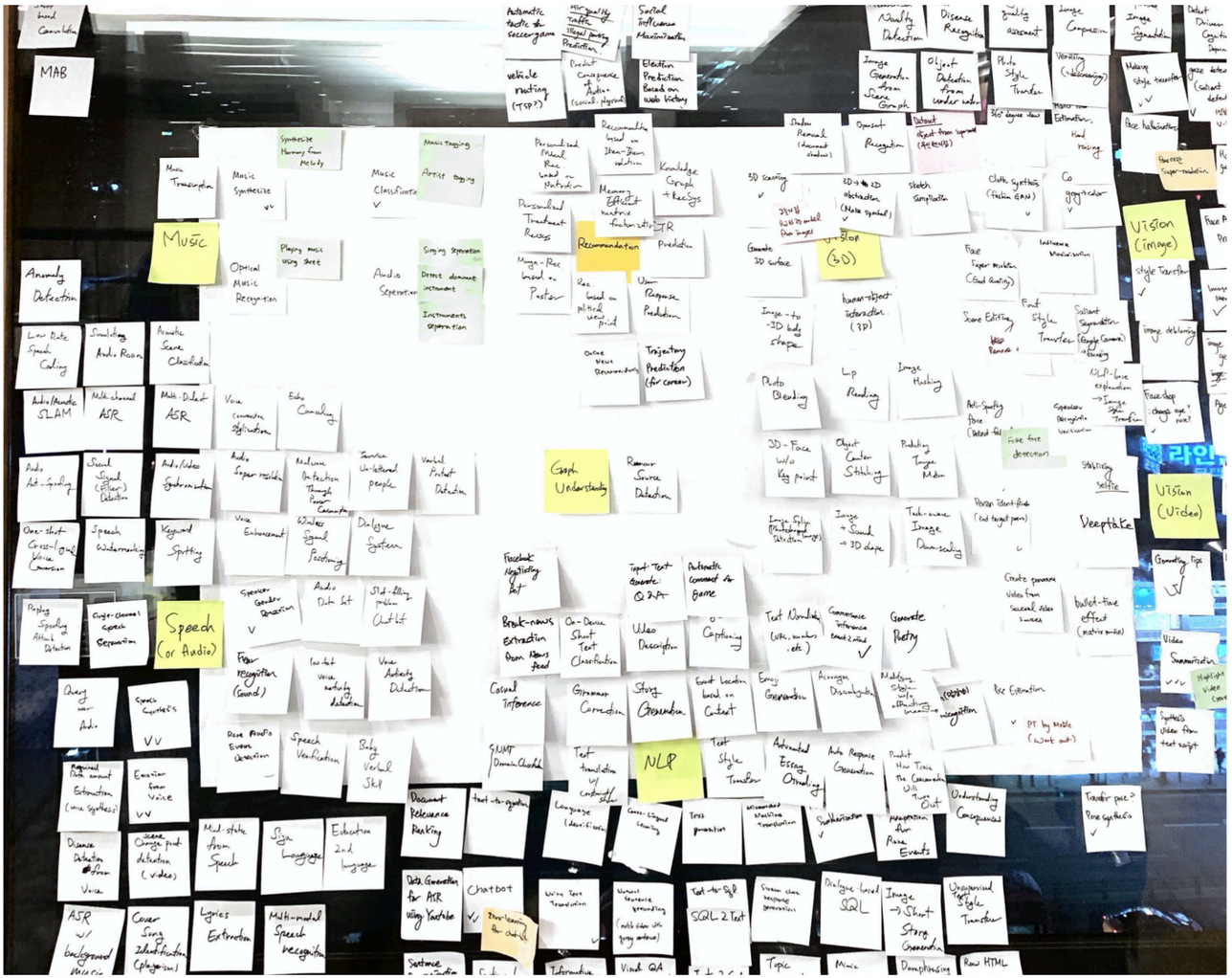
Hyperconnect AI Lab

- 기계학습 관련된 업무 전반의 담당
  - 프로젝트 선정
  - 데이터 수집
  - 모델 개발 및 실험
  - 논문화
  - 기획 참여
  - 데이터 QA
  - 배포

2019년 초

- 기존 팀의 포커스는 모바일 환경에서 실시간으로 이미지 다루기[1]

[1] https://github.com/hyperconnect/MMNet/



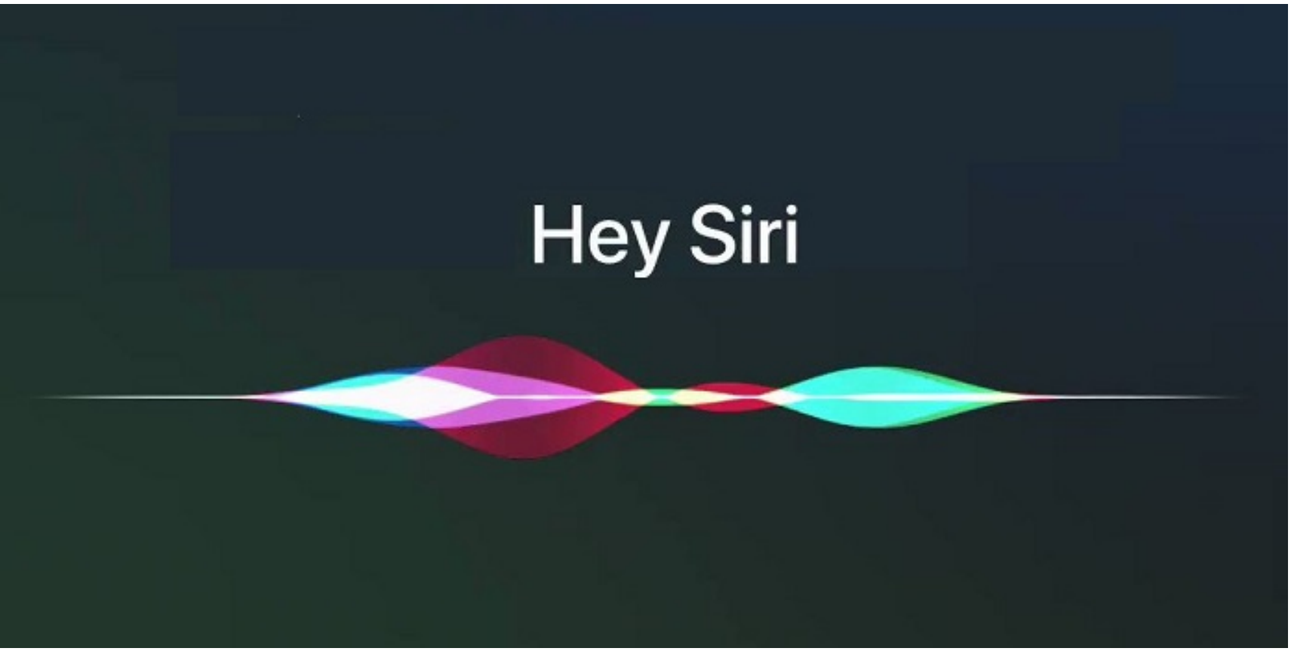
Workshop

- 10개 학회
  - 3700편 논문
- 유저 니즈/비즈니스 트렌드 기반 아이디어 브레인스토밍
- 300가지 정도의 잠재적인 활용처 고민
- 1년 간 로드맵 구성

Project Selection

- 다양한 요소를 고려하여 프로젝트 선정
  - 실현 가능성
  - 임팩트
  - 기술적 중요도
  - 트렌드
- 그 중 하나인 키워드 검출(keyword spotting)

Keyword Spotting



- 특정 핵심어가 발성되었는지 검출하는 문제
- 프로젝트 선택 시 고려했던 사항
  - 도메인 확장
    - CV 외의 도메인으로의 확장
  - 난이도
    - 분류 문제는 상대적으로 쉬우니 low hanging fruit 이라 판단
  - 기존 전문성
    - 경량 모델을 만들어 모바일 배포하는 것은 이미 잘 하기 때문에 좋은 성과를 금방 거둘 수 있으리라 판단

Literature Survey

A neural attention model for speech command recognition	2018-09-27	https://arxiv.org/abs/1808.08929				
Effective Combination of DenseNet and BiLSTM for Keyword Spotting	2019-01-10	https://ieee	Mengjun Zeng ; Nanf	- Google Speech Command로 SOTA 달성		
Short utterance based speech language identification in intelligent vehicles with t	2018-11-02	https://ieee	Zhanyu Ma, Hong Yu,	IEEE Transactions on Vehicular Technology 2018		
Low-bit quantization and quantization-awaretraining for small-footprint keyword spotting	2018-11-01	https://ope	Yuriy Mishchenko, Yu	NIPS 2018 workshop	- 파라미터 사이즈 50K	
Stochastic Adaptive Neural Architecture Search for Keyword Spotting	2018-11-01	https://arxi	Tom Véniat † , Olivier	Arxiv	- Google Speech Command로 NAS 시도	Google Spe
Keyword Spotting Based on Hypothesis Boundary Realignment and State-Level Confidence Weighting	2018-10-22	http://robot	Hong Liu, Yuezhao Cl	CSAE	- 음소를 맞추는 모델이 있다는 가정하에 어떻게 그 결과를 잘 조합할지에 대한 논문 - 음소 모델을 사용할 때 다시 보면 될 듯	
Efficient Voice Trigger Detection for Low Resource Hardware	2018-09-02		Siddharth Sigtia, Rob	Interspeech 2018	- Apple SIRI Research - 2-phase로 구성된 모델 - window = 25 ms & 13-dimensional MFCCs	
End-to-End Speech Command Recognition with Capsule Network	2018-09-01	https://www	Jaesung Bae, Dae-Sh	Interspeech2018	- Capsule Net 으로 Pose relationship between low-level relationship 문제를 해결 - Clean/Noisy dataset 에서 테스트	Google Speech Commands	Google Spe dataset
Data Augmentation for Robust Keyword Spotting under Playback Interference	2018-08-01	https://arxi	Anirudh Raju, Sankar	ArXiv	- Alex, Google Team 에서 진행한 연구 - 노이즈/불완전한 에코 켄슬링 환경 하에서 좋은 성능을 내도록 데이터 어그멘테이션 전략을 제안함 : AI 스피커로 음악을 듣다가 Alexa를 부르는 상황을 가정 [redacted] - DNN gives acoustic scores + HMM Decoder **- Weighted Cross-entropy 쓰는거 자연스러움**	3000 Hour datasets	
Efficient keyword spotting using time delay neural networks	2018-07-11	https://arxi	Samuel Myer, Vikrant	Interspeech 2018	- 2-stage : phone-NN + word-NN - transfer learning: phone 모델 먼저 다른 데이터로 학습한 이후에 고정시켜두고 word-NN을 학습함 - 속도 개선을 위한 아이디어: Frame Skipping & Caching to reduce computation - loss/regularization/batchsize등에 관한 설명이 없음 - https://www.fluent.ai/	- phone-NN: LVCSR dataset (132 phone targets) - word-NN: keyword datasets : 오디오 잘라붙여서 생성	Google spe In-house da
Attention-based End-to-End Models for Small-Footprint Keyword Spotting	2018-05-29	https://arxi	Changhao Shan, Junl	Interspeech2018	- 1.02% false rejection rate at 1.0 false alarm(FA) per hour - 아주 기본적인 모델을 Mi speaker 로 실험해봄 - CRNN + Soft attention + GRU is best	Mi AI Speaker(188.0K positive examples(~99.8h) / 1007.4K negative examples(~1581.8h))	same as tra
Deep Residual Learning for Small-Footprint Keyword Spotting	2018-04-15	https://arxi	Raphael Tang, Jimmy	ICASSP 2018		Google speech commands dataset	

Baseline

Name	Smoothed	Value	Step	Time	Relative
ResNet8Model/log_mel_spectrogram_3010_0.001_mom_l1/test	0.9630	0.9630	30.00k	Wed Mar 13, 17:56:11	21m 37s
ResNet8Model/log_mel_spectrogram_3010_0.0005_mom_l1/test	0.9623	0.9623	26.50k	Wed Mar 13, 19:19:47	19m 10s
ResNet8Model/log_mel_spectrogram_3010_0.001_mom_s1/test	0.9611	0.9611	18.00k	Wed Mar 13, 17:49:26	15m 9s
ResNet8Model/log_mel_spectrogram_3010_0.001_adam_l3/test	0.9591	0.9591	30.00k	Wed Mar 13, 17:57:12	22m 26s
ResNet8Model/log_mel_spectrogram_3010_0.001_adam_l2/test	0.9555	0.9555	30.00k	Wed Mar 13, 17:03:51	20m 23s
ResNet8Model/mfcc_4020_0.001_mom_l1/test	0.9552	0.9552	24.50k	Wed Mar 13, 19:19:50	18m 28s
DSCNNLModel/mfcc_4020_0.0_adam_l2_dropout/test	0.9539	0.9539	19.50k	Wed Mar 13, 19:19:44	15m 43s
DSCNNLModel/mfcc_4020_0.0_mom_l1_dropout/test	0.9536	0.9536	19.50k	Wed Mar 13, 19:19:38	16m 3s
DSCNNLModel/mfcc_4020_0.0_mom_l1/test	0.9523	0.9523	30.00k	Wed Mar 13, 18:52:40	23m 15s
DSCNNLModel/mfcc_4020_0.0_adam_l2/test	0.9510	0.9510	30.00k	Wed Mar 13, 17:07:13	23m 56s
ResNet8Model/log_mel_spectrogram_3010_0.005_mom_l1/test	0.9484	0.9484	30.00k	Wed Mar 13, 18:49:53	19m 6s
DSCNNMModel/mfcc_4020_0.0_mom_l1/test	0.9455	0.9455	30.00k	Wed Mar 13, 18:50:25	21m 22s
DSCNNMModel/mfcc_4020_0.0_adam_l2/test	0.9445	0.9445	30.00k	Wed Mar 13, 17:05:08	22m 3s
DSCNNSModel/mfcc_4020_0.0_mom_l1/test	0.9422	0.9422	30.00k	Wed Mar 13, 18:49:08	20m 19s
DSCNNSModel/mfcc_4020_0.0_adam_l2/test	0.9026	0.9026	1.500k	Wed Mar 13, 18:24:22	1m 46s

- 비교 대상이 되는 모델이 있어야 개선이 의미가 있음
  - 점진적으로 비교할 수 있는 모델을 늘려가며 다양한 컴포넌트를 확보
- 프로덕션에서는 이미 연구된 모델을 구현하는 것이 충분할 수도 있음

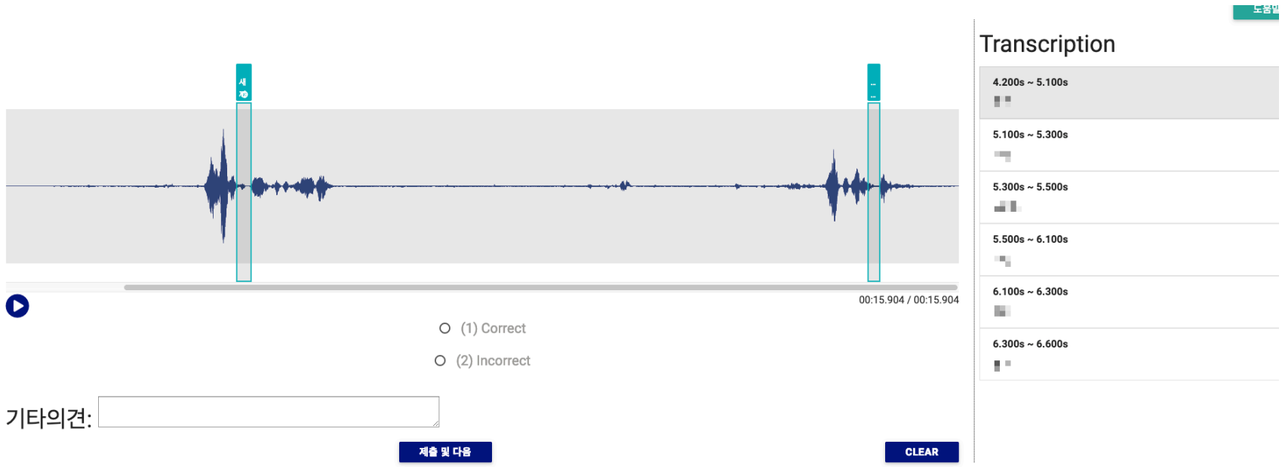
```
"""
Modified code from https://github.com/tensorflow/tensorflow/blob/master/tensorflow/examples/speech_commands/models.py
We tried to exactly implement model from 'Convolutional Neural Networks for Small-footprint Keyword Spotting' paper

- one_fstride4
- trad_fpool3

Our implemented models have same # of flops as 'Hello Edge: Keyword Spotting on Microcontrollers' paper reported
"""

import math
import tensorflow as tf


def create_model(fingerprint_input, model_settings, model_architecture,
                 is_training, runtime_settings=None):
  """Builds a model of the requested architecture compatible with the settings.
```

Baseline Selection

- 합리적인 성능이 나오는지 (SotA와의 비교)
- 구현 난이도 및 공식 코드 공개 여부
  - 재현이 까다로운 경우가 자주 있음
- 프로덕션에서 적용될 제약 조건을 얼마나 만족하는지
  - 모바일 CPU에서 실시간 수행 가능?[2]

[2] Convolutional Neural Networks for Small-Footprint Keyword Spotting (CNN)
Hello Edge: Keyword Spotting on Microcontrollers (DS-CNN)
Deep Residual Learning for Small-Footprint Keyword Spotting (Res)

Data



- 공개 데이터셋
  - 논문들이 공통적으로 사용하는 데이터셋을 최대한 확보
  - 공정한 비교를 해야함
  - 불필요하게 기존 모델보다 안 좋은 모델을 만드는데에 큰 노력을 기울이지 않도록
  - 아쉽게도 학외 소속에게는 데이터를 공개하지 않는 경우가 무척 많음[3]
- 비공개 데이터셋
  - 내가 관심있는 도메인에서의 모델 성능은 다를 수 있음
  - 데이터 수집에 대한 고민
    - 어노테이션
    - 정합성 확인
- 데이터 탐색이 필수적

[3] 오픈된 데이터이고 논문 작성을 위한 용도라 하더라도 거절하는 경우가 많음

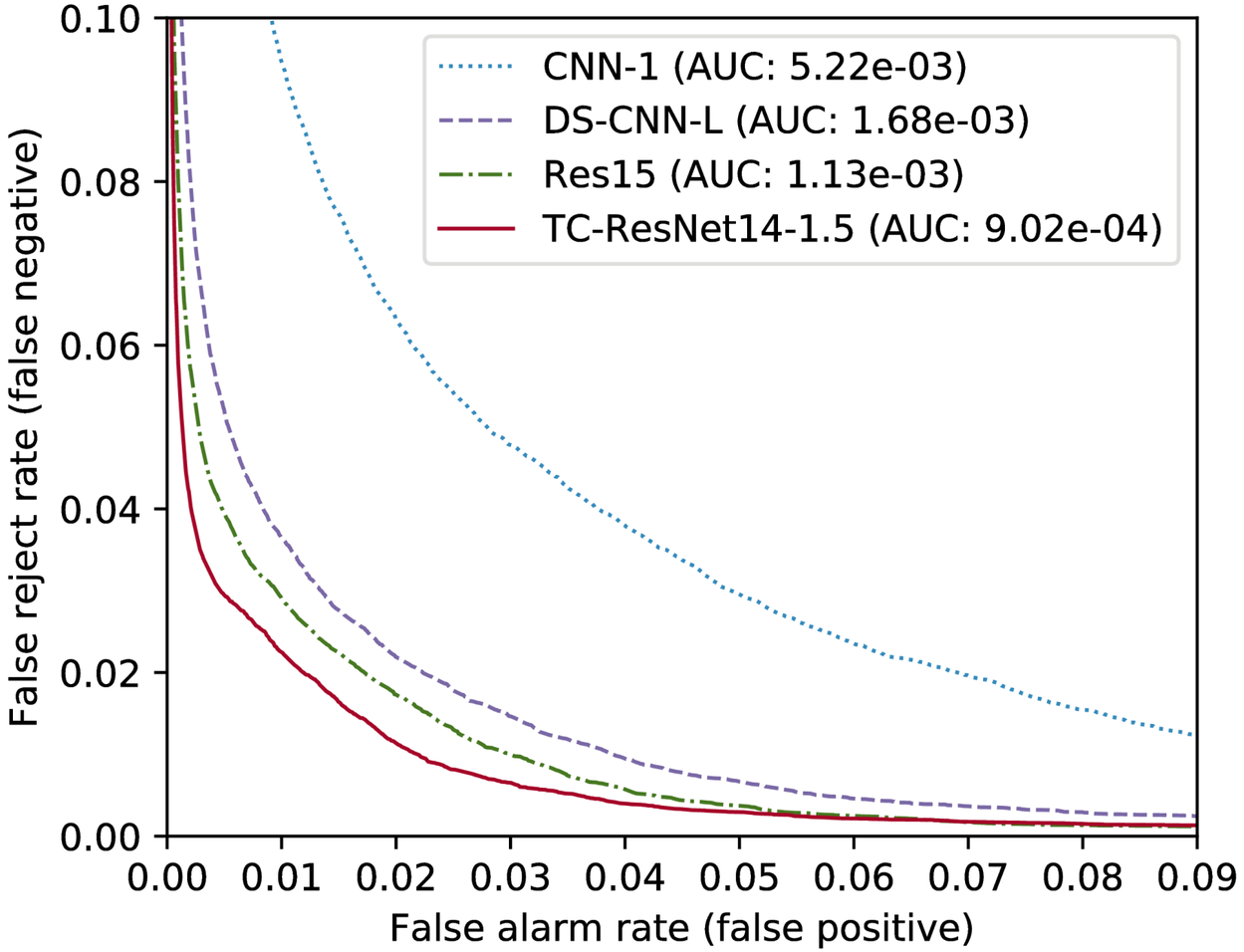
PoC

- 새로운 아이디어 테스트
  - Baseline 모델 개선
- 달성하고 싶은 목표를 설정하고
  - 충분한 정확도 + 모바일 CPU 실시간
- 단계적으로 도착할 수 있는 방법을 설정
  - 충분한 정확도 먼저
  - 모바일 CPU 속도는 그 다음
- 중간 산출물
  - 프로덕션에 활용될 수 있는지 치열하게 고민해야 함
  - 이를 고려한 마일스톤을 잡아야 함

Process

- 모델이 제품에 적용되기 위해 필요한 부분을 모두 만들어 한 바퀴 사이클 돌리기
  - 도메인에 적합한 전처리 기법 구현
  - 학습 및 검증 파이프라인 만들기
  - 디플로이를 위한 고려
    - 단순 연구와 달리 실제 서비스 환경에서의 모델 행동 양식에 대한 고민이 필요
    - TF-Lite 활용 가능성 등

Evaluation



- 여러 모델을 구현/비교할 때에는 공정하게
- 같은 데이터, 같은 종류의 어그멘테이션 등을 활용해야
- 모델 최적화하는 사람의 역량에 따라 다를 수 있음
  - 논문 재현 시 리포팅된 결과보다 좋은 결과가 종종 나옴
- 논문에서는 측정하기 용이한 메트릭을 주로 보지만 프로덕션 환경에서는 다양한 메트릭을 보아야 할 수도 있음
  - 정확도 vs. 시간당 오탐률
  - Flops vs. 실제 latency
  - 모델의 확신에 따른 정밀도와 재현율

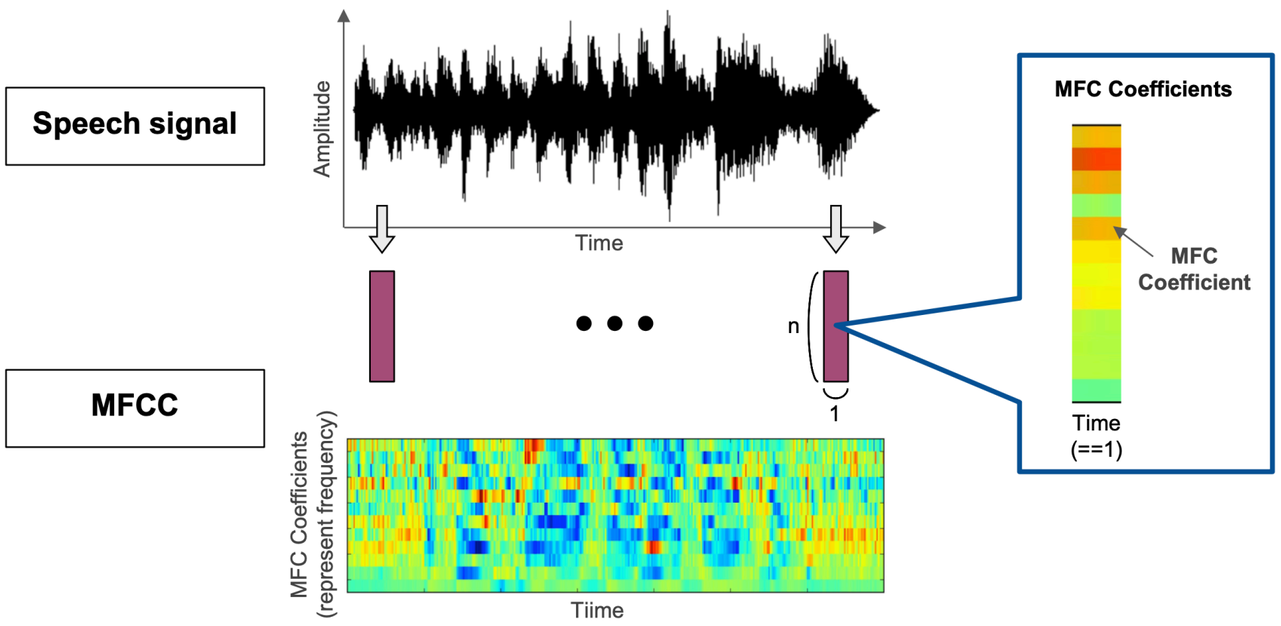
Research

- 원하는 목표에 도달하기까지 계속해서 다양한 시도
- 보통 생각대로 잘 되지 않고 시행착오를 많이 겪게 됨
  - 모두의 인내와 이해가 필요한 시간
- 몇 가지 접근 방법
  - 리터러쳐 서베이에서 유망해보였던 모델 재현하면서 아이디어 얻기
  - 다른 도메인의 아이디어 훔쳐오기
  - 팀원들과의 토론

KWS Research Progress

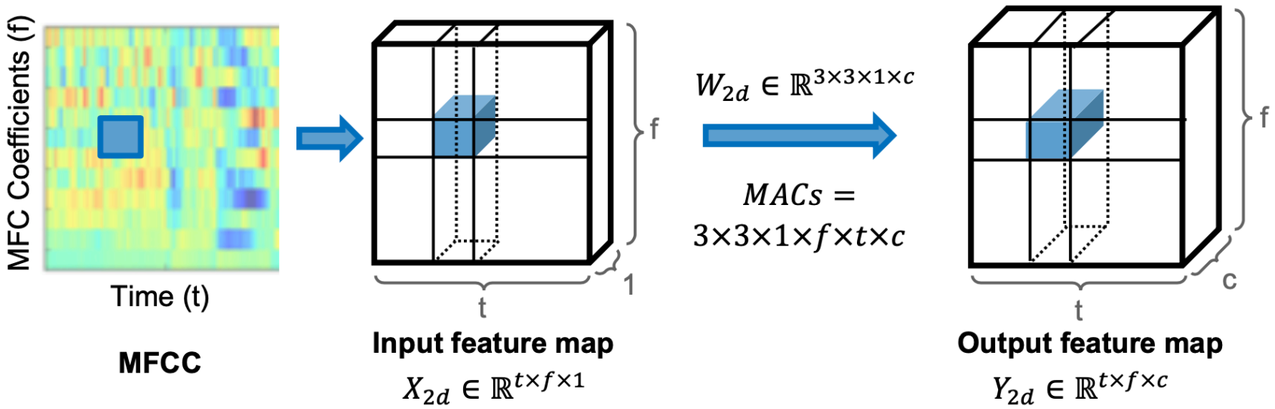
- 기존 연구를 보다보니 ResNet을 활용한 구현들이 미묘하게 오리지널 ResNet과 다른 구조
- 이를 상황에 맞게 최대한 기존 모델과 비슷하게 맞춰보니 결과가 미묘하게 좋아짐
- 다만 속도가 충분히 빠르지 않아서 이를 가속할 방법을 고민
  - 기존에 모바일 컴퓨터 비전 영역에서의 전문성을 최대한 지렛대로 활용

Audio Processing



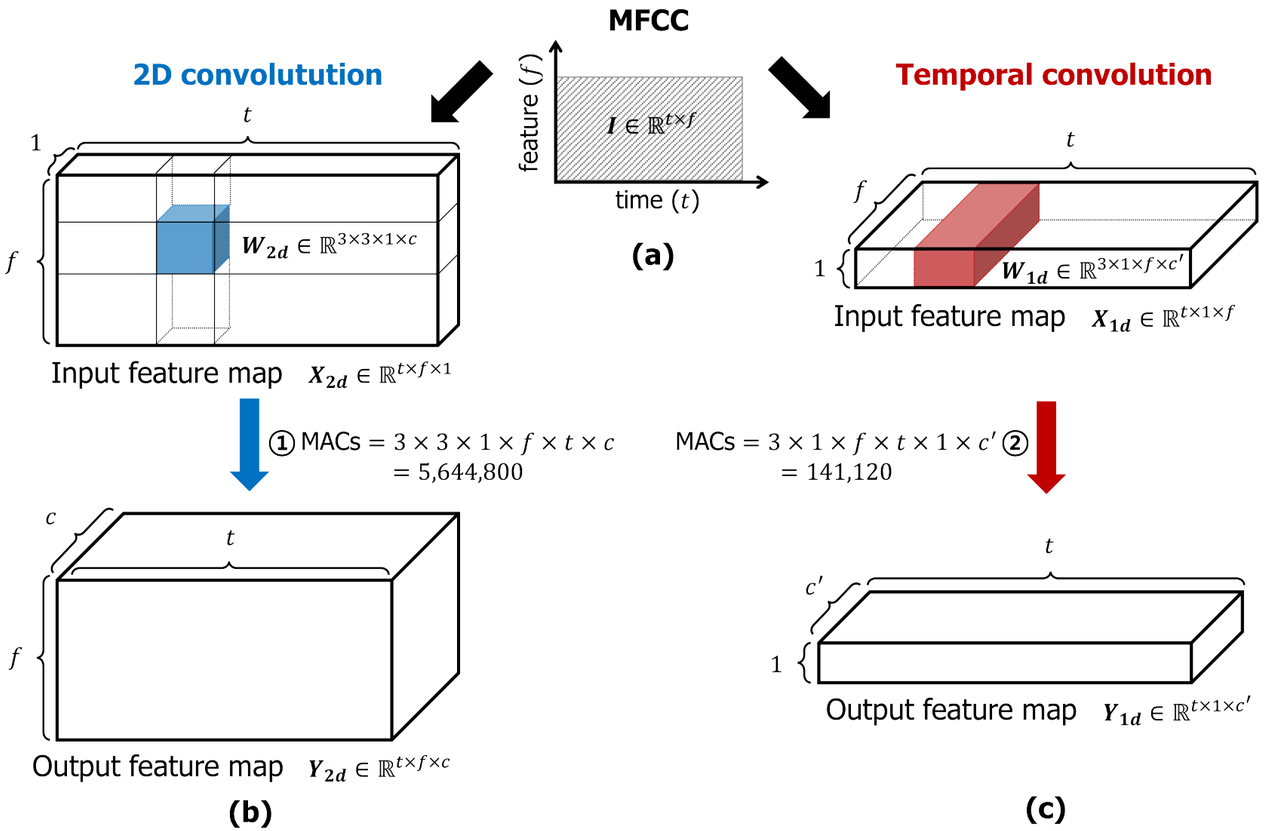
- 주로 MFCC를 활용
  - 사람이 인지하는 방식을 고려한 푸리에 변환 정도로 이해해도 무방
- 시간-계수의 꼴로 데이터가 변환됨

CNN-Based KWS



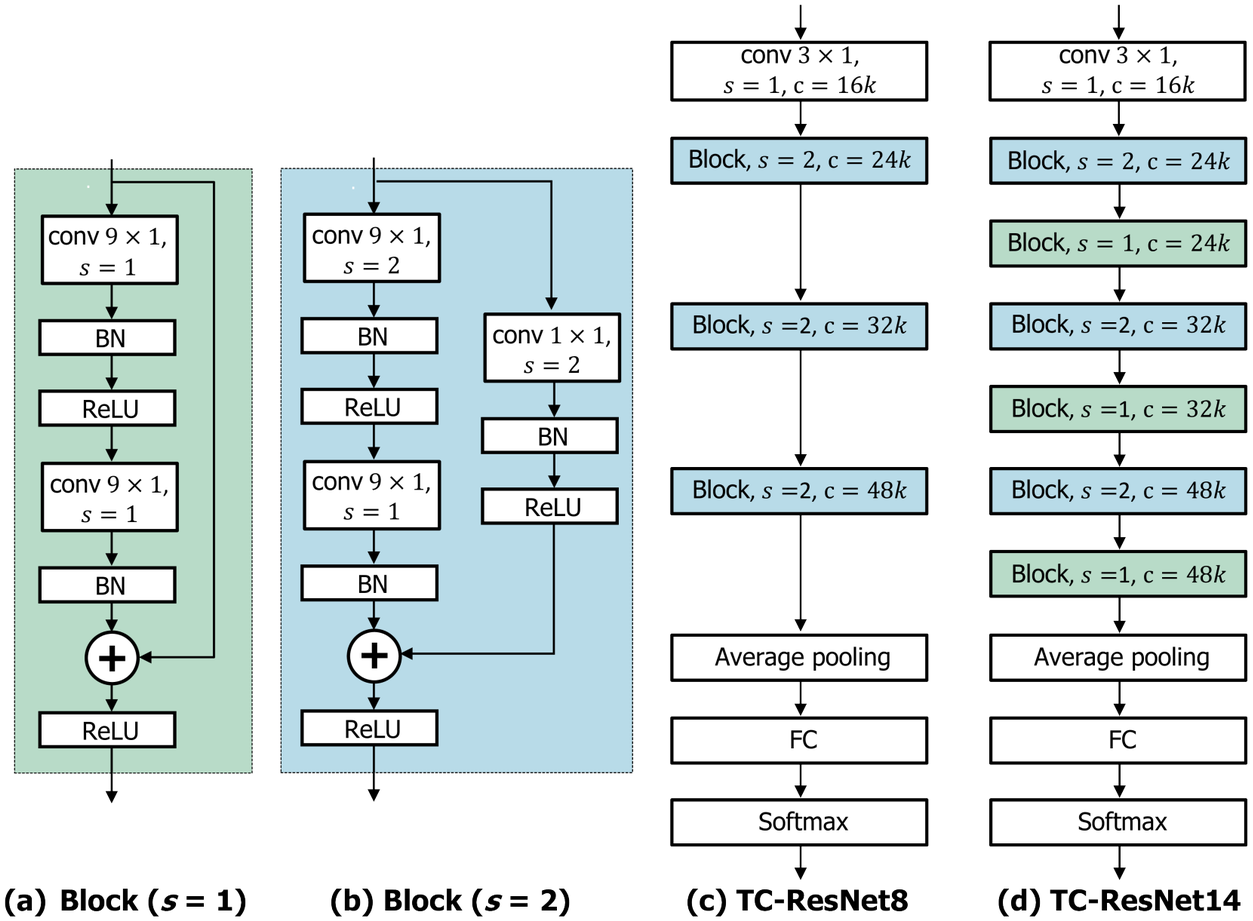
- MFCC를 1-채널 2D-이미지 취급
- 기존의 비전 영역의 CNN을 그대로 적용
- 문제는 충분히 넓은 receptive field를 갖추기 위해서는 깊은 네트워크를 쌓아야 함

Temporal Convolution



- 의미를 생각해보면 동시에 낮은 주파수와 높은 주파수를 함께 봐야할 것 같음
  - MFCC를 다중채널 1D-이미지 취급
- 한 번의 컨볼루션으로도 모든 피쳐가 receptive field에 들어옴
- 연산량의 측면에서도 이득
  - 캐시 친화적인 구조

TC-ResNet



(a) Block (*s* = 1) **(b) Block (*s* = 2)** **(c) TC-ResNet8** **(d) TC-ResNet14**

- 가벼운 *ResNet* 모델에
- *Temporal convolution* 적용
- 아주 간단한 아이디어
  - 팀에서 모바일 *CPU* 파이프라인을 뜯어보고 *ARM* 어셈블리 프로그래밍을 해보며 모바일에서 빠른 코드에 대한 이해가 충분히 깊었기에 쉽게 떠올릴 수 있었음

Result

Model	Acc. (%)	Time (*ms*)	FLOPs	Params
CNN-1	90.7*	32	76.1M	524K
CNN-2	84.6*	**1.2**	1.5M	148K
DS-CNN-S	94.4*	1.6	5.4M	24K
DS-CNN-M	94.9*	5.2	19.8M	140K
DS-CNN-L	95.4*	16.8	56.9M	420K
Res8-Narrow	90.1*	47	143.2M	20K
Res8	94.1*	174	795.3M	111K
Res15-Narrow	94.0*	107	348.7M	43K
Res15	**95.8***	424	1950.0M	239K
TC-ResNet8	96.1	**1.1**	3.0M	66K
TC-ResNet8-1.5	96.2	2.8	6.6M	145K
TC-ResNet14	96.2	2.5	6.1M	137K
TC-ResNet14-1.5	**96.6**	5.7	13.4M	305K

- 속도 불문 정확도 개선
- 기존 정확도 SotA 모델 대비 *385x* 빠름
- 기존 속도 SotA 모델 대비 *11.5%p* 정확도 개선
- 벤치마크 환경 및 코드 공개
  - https://github.com/hyperconnect/TC-ResNet/
  - https://arxiv.org/abs/1904.03814

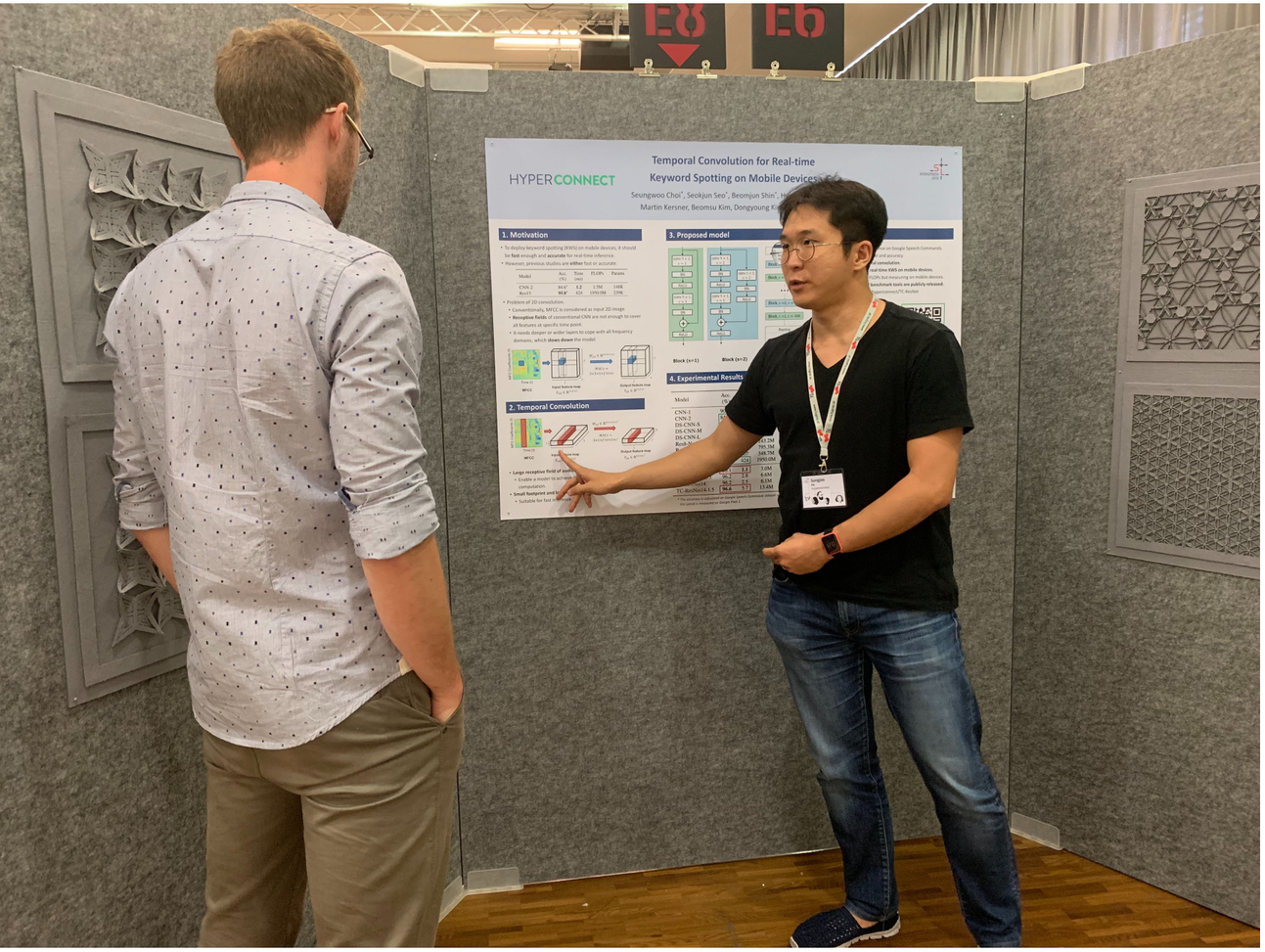
Publishing

- 의미있는 결과는 최대한 논문화
- 논문을 쓰면서 얻을 수 있는 것
  - 우리가 풀고자 하는 문제가 무엇인지 명확하게 정의하는 것
  - 앞으로 진행해야 하는 실험이 무엇인지 알게 되는 것
  - *Ablation* 테스트 등을 통해 불필요한 컴포넌트를 이해하는 것
- 어차피 숨기고 있어봐야 몇 달 내로 더 좋은 기술이 나옴

Ablation Test

- 프로덕션 연구 개발 과정에서는 많은 것들이 점진적으로 이루어짐
- 최종적인 모델에서 예전에는 의미가 있었으나 더 이상 의미 없는 부분이 있을 수 있음
- Ablation 테스트로 제거
  - 열심히 만들었던 컴포넌트가 사실 별로 쓸모 없었다는 결과는 무척 흔하게 나옴
  - 결과적으로는 프로덕션 환경에서 불필요한 부분을 제거하므로 이득

Retrospection



- 프로젝트 시작 3개월 후 SotA 기술 개발 완료 및 프로덕션 진행
  - 기계학습 전문성 > 도메인 전문성이었던 예
  - 기존의 전문성을 잘 활용한 예시
    - 모바일 CPU 실시간-모델 최적화

Production + Research

- 제작을 중심으로 하는 회사/팀에서 성공적인 기계학습 조직 운영
  - 서로의 기대를 맞춰야 하고
  - 서로 윈-윈할 수 있는 *positive-sum* 게임을 만들어야 함

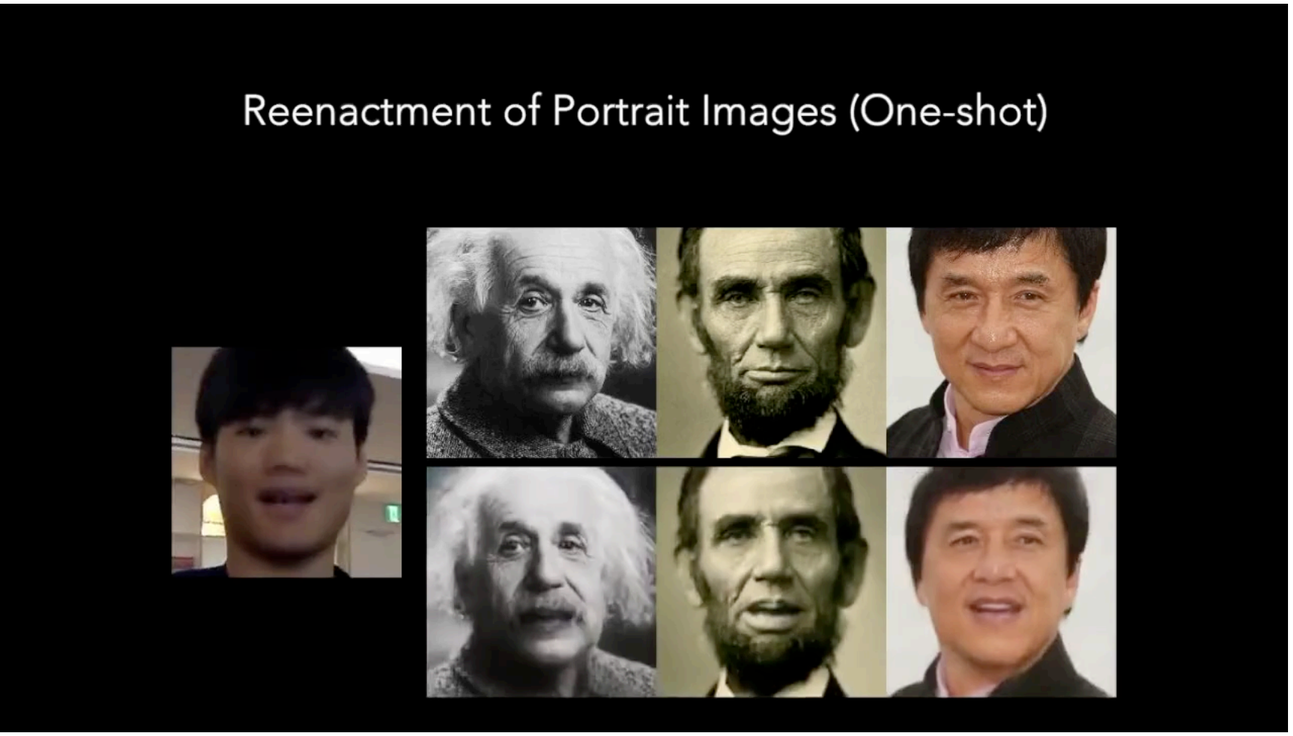
Expectation Management

- ML ≠ Magic
  - 시도하고 실패할 수 있음을 회사에서는 인지해야 함
  - 팀에서는 리스크를 스스로 판단하고 움직일 수 있어야 함
  - 하지만 전문성이 잘 맞는 분야에서는 놀라운 결과를 단시간에 낼 수도 있음
- 팀은 제품에 기여 해야함
  - 팀에서 해당 고민을 꼭 해줘야 함
    - 기계학습 기술은 약간의 변형을 통해 다방면으로 사용될 가능성이 있음
    - 능동적으로 다른 팀과 기술의 활용에 대한 이야기를 해야함
  - 소프트웨어 개발력 + 기계학습 연구력

Positive-Sum Game

- 제로섬 게임이나 네거티브섬 게임보다는 포지티브섬 게임이 낫다
  - 회사와 팀 리더가 특히 고민해줘야 함
- 회사도 팀도 팀원도 연구의 성공 실패와 무관하게 득을 볼 수 있는 방법을 고민해야 함
  - 제품에 들어갈 수 있는 연구
  - 팀원의 성장과 커리어 디벨롭먼트에 대한 고민

Ownership



- 프로젝트의 결정 및 방향 설정에 팀원들이 함께 정함
  - 이 기술이 회사에 쓸모 있을까?
    - 연구를 위한 연구는 대부분의 회사에서는 빛을 발하기 힘듦
  - 내가 이 연구를 하면 재미있을까?
    - 연구에서 막히는 경우 인내심을 발휘할 수 있는 이유
- 기술을 가장 잘 이해하고 있는 것은 연구자 본인
  - 사내 다른 팀들과 지속적으로 이야기 해야함

We Are Hiring![4]

- Mobile Deep Learning
- ML Platform Software Engineering
- Computer Vision
- Speech Recognition
- Natural Language
- Generative Modeling
- Recommender Systems

[4] https://hyperconnect.com/career/
career@hpcnt.com