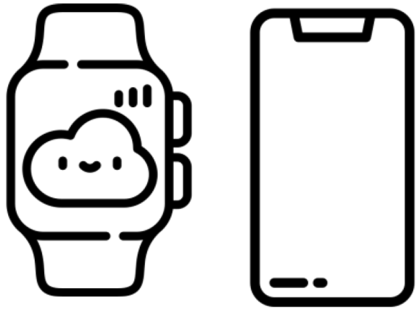
모바일 컴퓨팅과 응용

Team 2

마음의 소리

당신의 스트레스 ,
걱정마세요



Project Strategy

00

Midterm Feedback

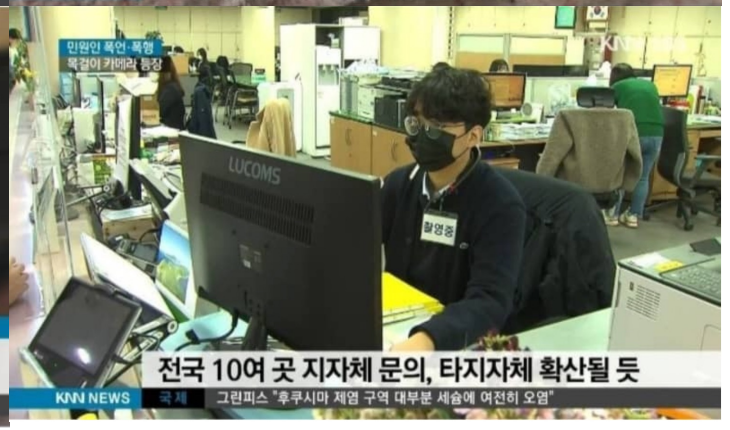
- 좁힌 User-Scenario에 맞추어진 기능 추가
- PPG-to-Stress 모델에 추가 Feature 활용하여 성능 개선
- App의 시각화 개선을 통한 전반적인 UX 개선

Project Strategy

01

Target User & Problem

Target User & Problem



Key Solution idea

Wear OS

Android OS

PPG → Stress

Speech → Text

스트레스가 높았던 구간의
음성 및 텍스트 검토

저장 & 공유를 통한 활용

Evaluation Strategy

Usage Scenario

1. 업무 시작 시점에 버튼을 눌러 기록 시작
2. 대면 업무 처리 간 Watch에서 PPG 및 음성 기록
3. Watch에서 스마트폰으로 주기적으로 데이터 전송
4. Speech-To-Text (Cloud-based), PPG-To-Stress (Edge-based)
5. DB에 음성 / 텍스트 / 스트레스 데이터 저장
6. 업무 종료 시점에 버튼을 눌러 기록 종료
7. Sync된 음성 / 텍스트 / 스트레스 로그 확인
8. 필요 시 폭언, 욕설이 들어간 부분을 저장하여 활용

Evaluation Strategy

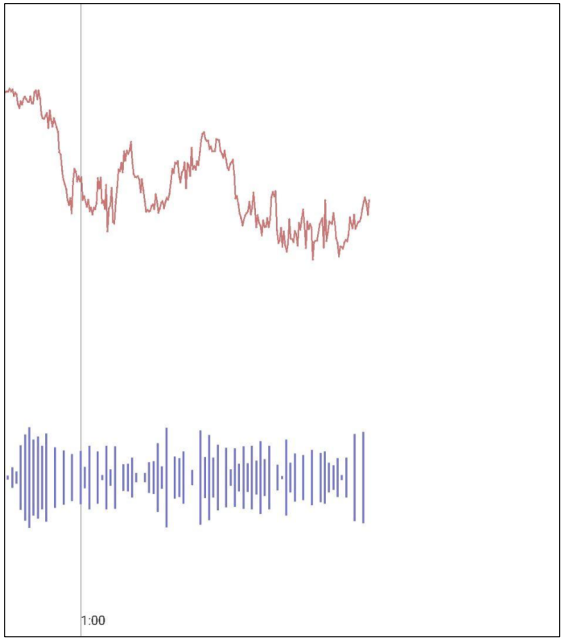
1. 피실험자의 스트레스 지수를 낮추기 위해 20분간 명상한다.

2. 민원 업무 환경을 가정하고 욕설/폭언이 담긴 음성파일을 청취한다

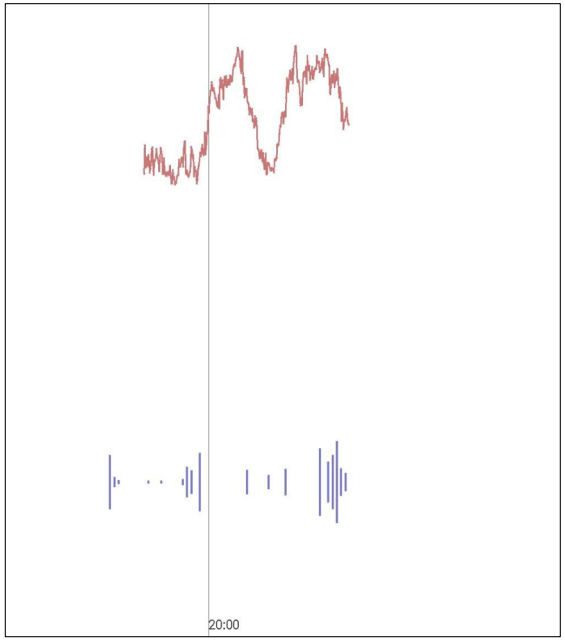
3. 명상을 했을 때와 비교하여 스트레스 지수의 변화를 확인한다.

4. 음성 파일을 들으며 Speech-To-Text 변환 정확도를 확인한다.

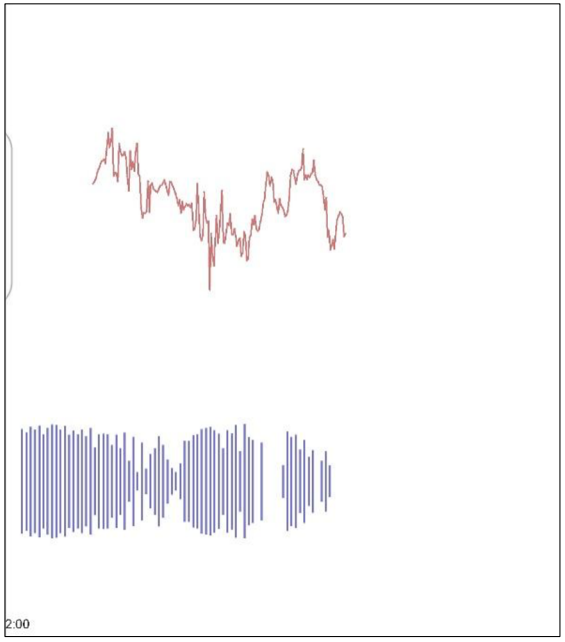
Evaluation Result



표재우



정재윤



박지상

System Architecture

PPG : 10초 단위 전송

Voice : 1분 단위 전송

Voice Recording
PPG Sensors

- PPG
- PPG Raw
- PPG SPO2
- PPG SDNN
- PPG HD
- PPG Static

Feature Extraction

- LF, HF, LF_HF
- HR, TP, VLF
- RMSSD, pNN50

Speech-To-Text

- Cloud-based Approach
- Volume-based Control
- Threading Control

Stress Index Calculation

- M/L Model
- Pre-Trained with Dataset
- Outputs 3 stage & calculate by ratio

Visualization & Management

- Stress & Audio Visualization
- Labelling & Leaving a text
- Export & Sharing

Challenge 1) Voice Data Processing

Audio Segmentation

Speech-To-Text

Challenge 1-1) Audio Segmentation

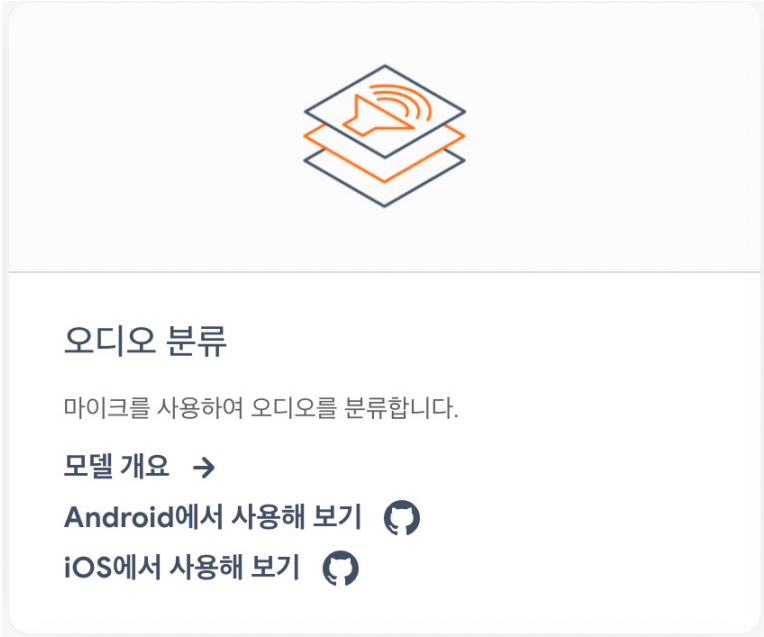
Dear developer,
The identity verification application you submitted on 2021-11-20 was reviewed.

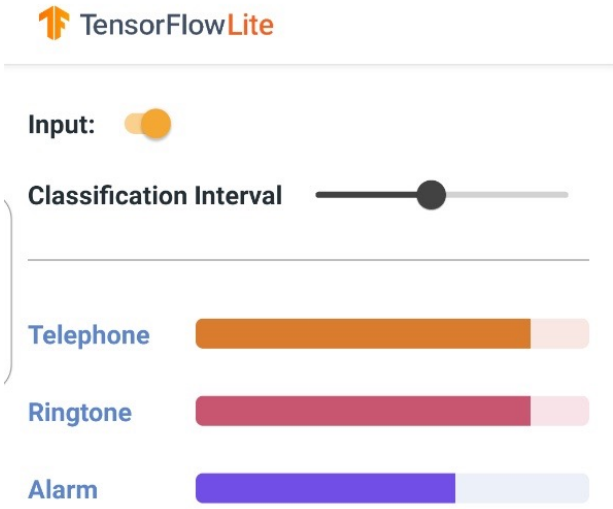
Review result: rejected
Reason for rejection: Please provide your ID information in Roman Character by upload a copy of national passport / national ID

	Sound events
1	Shouting
2	Baby or child crying
3	Mew
4	Barking
5	Running water (such as water taps, streams, and ocean waves)
6	Knocking
7	Doorbell
8	Car horns (such as sedan car and truck)
9	Fire alarm sounds (such as fire alarm and smoke alarm)
10	Alarm sounds (such as fire truck alarm, ambulance alarm, police car alarm, and air defense alarm)
11	Laughter
12	Sneezing
13	Snoring

Challenge 1-1) Audio Segmentation

Tensorflow Officials





Challenge 1-1) Audio Segmentation

```
// Initialize the audio classifier
val classifier = AudioClassifier.createFromFile( context: this, MODEL_FILE)
val audioTensor = classifier.createInputTensorAudio()

// Initialize the audio recorder
val record = classifier.createAudioRecord()
record.startRecording()

// Define the classification runnable
val run = object : Runnable {
  override fun run() {
    val startTime = System.currentTimeMillis()

    // Load the latest audio sample
    audioTensor.load(record)
    val output = classifier.classify(audioTensor)
```

TensorFlow Example

MIC Input을 필요로 함

AudioClassifier Class
- createInputTensorAudio()
- createAudioRecord()
- startRecording()
- audioTensor.load(record)

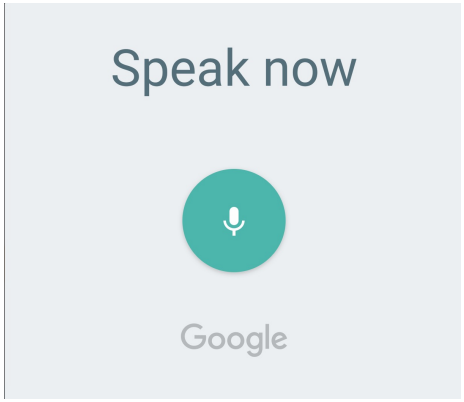
Usage Scenario

Watch에서 받아오는 ByteArray

TensorAudio instance로 만들기 위한 전처리

Challenge 1-2) Speech-To-Text

Google SpeechRecognizer




```
private fun startSTT(){
    Log.d( tag: "STT",  msg: "STT started")
    intent = Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH)
    intent.putExtra(RecognizerIntent.EXTRA_CALLING_PACKAGE,getPackageName())
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE,  value: "ko-KR")
    mRecognizer = SpeechRecognizer.createSpeechRecognizer( context: this)
    mRecognizer?.setRecognitionListener(listener)
    mRecognizer?.startListening(intent)
}
```

- 쉬운 Implementation
- 높은 인식 정확도
- MIC 기반 input
- Utterance 단위 인식

Q) How to record audio and process speech recognition at the same time?
A) POST REQUEST 보내기 직전의 FLAC File을 저장
>>> That works pretty well but has a huge limitation which is **it can't be used with files longer than about 10-15 seconds** (the exact limit is not clear and may depend on file size or perhaps the amount of words). This makes it not suitable for my needs.

Challenge 1-2) Speech-To-Text

CLOVA Speech Recognition(CSR)

사람의 목소리를 텍스트로 바꿔주어 다양한 음성 인식 서비스에 활용할 수 있습니다

이용 방식	지원 플랫폼	인식 가능 언어	인식 가능 시간	인식 가능 음성파일 포맷	CSR 엔진 전달 데이터
모바일 SDK	Android 2.3.3 (API 레벨 10)	한국어, 영어, 일본어, 중국어	60초	-	마이크로 입력된 음성
	iOS 8 이상				
REST API	-			mp3, aac, ac3, ogg, flac, wav	녹음 파일

Challenge 1-2) Speech-To-Text

Naver Clova CSR REST API

```
val outputStream = conn.outputStream
val inputStream = FileInputStream(voiceFile)
val buffer = ByteArray( size: 4096)
var bytesRead = -1
while (inputStream.read(buffer,  off: 0, buffer.size).also { bytesRead = it } > 0) {
    outputStream.write(buffer,  off: 0, bytesRead)
}
outputStream.flush()
inputStream.close()
Log.d( tag: "STT",  msg: "STT Finished transmitting data to server")

var br: BufferedReader? = null
val responseCode = conn.responseCode
br = if (responseCode == 200) { // 정상 호출
    BufferedReader(InputStreamReader(conn.inputStream))
} else {  // 오류 발생
    println("error!!!!!!! responseCode= $responseCode")
    BufferedReader(InputStreamReader(conn.inputStream))
}
var inputLine: String?
if (br != null) {
    response = StringBuffer()
    while (br.readLine().also { inputLine = it } != null) {
        response.append(inputLine)
    }
```

수많은 시행착오

1. NetworkOnMainThreadExcetion()
2. File Storage Permission & Access
3. ByteArray → MP3 Encoding
4. MP3 Encoding Parameter 최적화
5. Threading Process Control
6. Cloud Request Control

Challenge 1-2) Speech-To-Text

Encoding to MP3 & Parameter Setting


```
val encoder = LameEncoder(
    AudioFormat( fSampleRate: 8000.0f, nSampleSizeInBits: 16,
        nChannels: 1, bSigned: true, bBigEndian: false),
    bitRate: 160,
    MPEGMode.MONO,
    Lame.QUALITY_HIGHEST,
    VBR: false
)
```

Mobile : ByteArray → MP3

Transmission period: 60s
Transmission size: 960KB

Encoding : PCM_16BIT
Sample-Rate : 8000Hz
bit-rate : 160 kb/s (Standard)

```
val audioRecord = AudioRecord.Builder()
    .setAudioSource(MediaRecorder.AudioSource.MIC)
    .setAudioFormat(
        AudioFormat.Builder()
            .setSampleRate(RECORDING_RATE)
            .setChannelMask(CHANNEL_IN)
            .setEncoding(FORMAT)
            .build()
    )
    .setBufferSizeInBytes(intSize * 3)
    .build()
```

Watch : Voice Input → ByteArray

Challenge 1-2) Speech-To-Text

Threading Process Control

- REST API based on FileInputStream
- Used to Create temp.mp3 File in internal storage and used it
- Audio File (Temp.mp3) getting overwritten before the thread ends
- Used to control with Thread.Sleep() by the existence of temp.mp3
- Switched from FileInputStream to ByteArrayStream & from File to dump byteArray

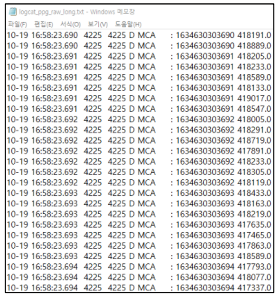
Cloud Request Control

- Sending Request every 1 minute with data transmission of 480KB
- Calculate volume of 1 minute audio file and compare with the threshold (13.5f)
- Request only when the volume of audio file is above threshold

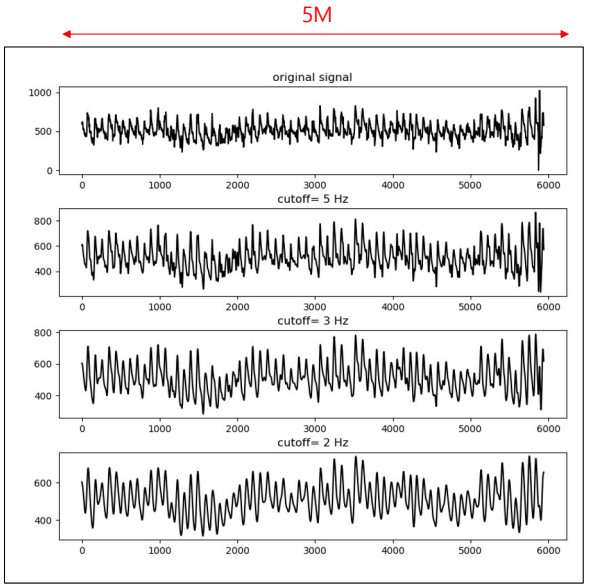
Challenge 2) Stress Data Processing

How to calculate LF and HF?

raw_PPG.txt

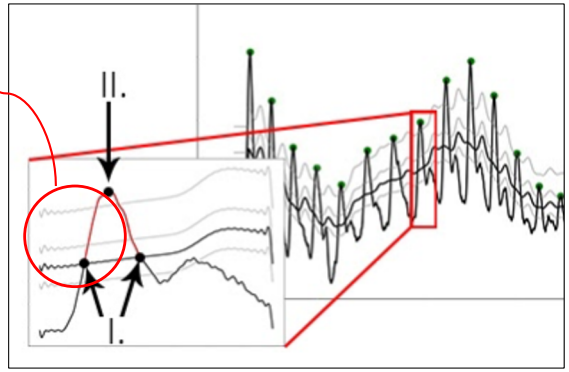


raw PPG data from watch sensor for 5 M



Noise filtering w/ Butterworth Band pass filter

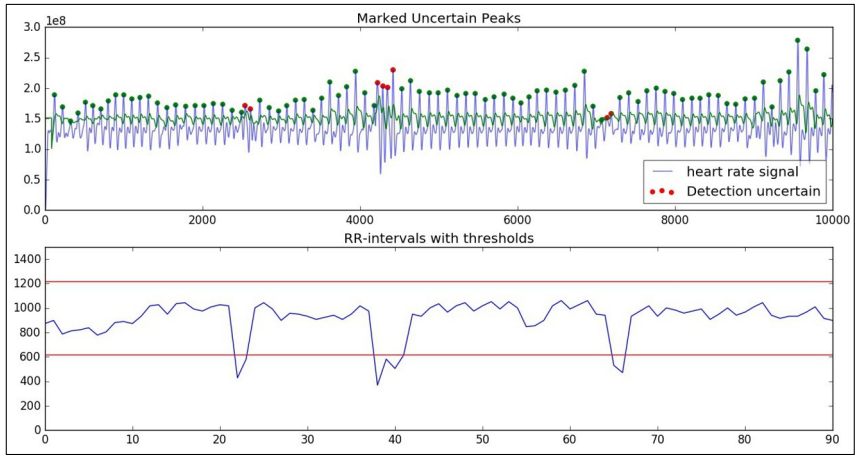
ROI(Region of Interest)



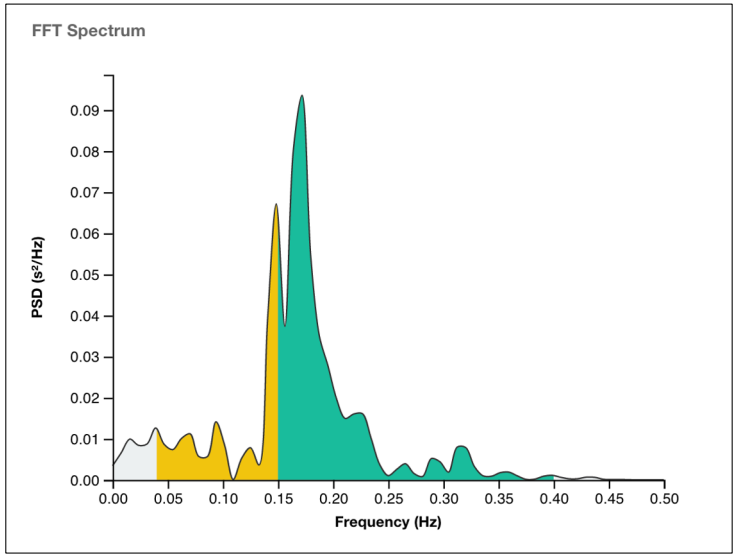
Peak detection from ROI

Challenge 2) Stress Data Processing

How to calculate LF and HF?



Detection uncertain peak point



Get Area of LF & HF for 5M

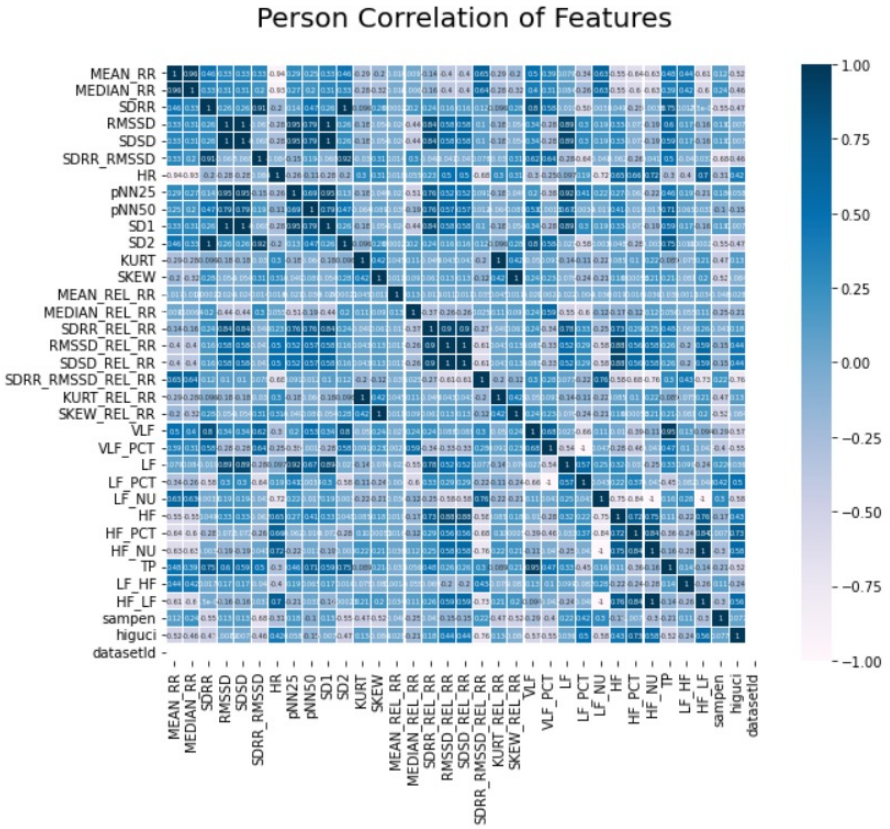
Challenge 2) Stress Data Processing

SWELL Dataset

- Dataset에 parameter가 매우 많다. (30개)
- Pearson correlation 분석을 통해 Features 간의 상관계수를 산출함

>>> 30개 feature중에서 correlation 값이 서로 높은 것들은 제외하고(중복방지), feature selection을 통해 결과 기여에 순위가 높은 feature만 따로 뽑아내었다.

LF/HF (상위권), pnn 50 (상위권), HR (중위권), RMSSD (하위권)



Challenge 2) Stress Data Processing

SWELL Dataset

Dataset 정보 : Train = 369,290 개의 datapoint / Test = 41,000 개의 datatpoint

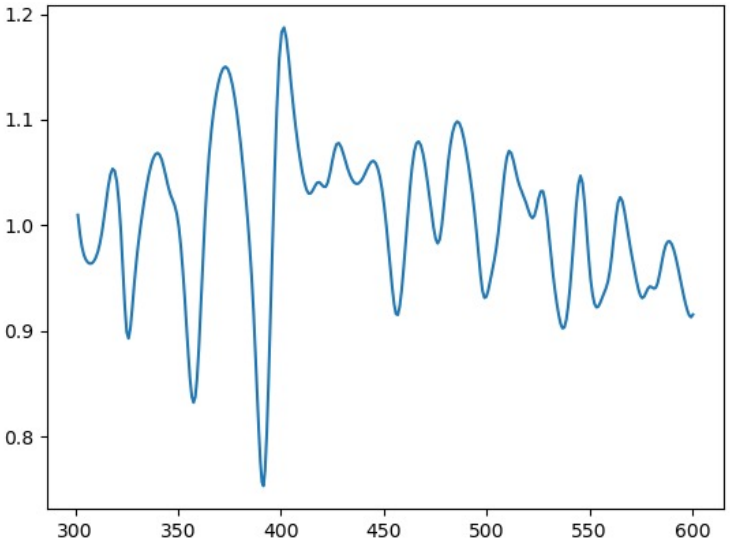
4가지 Feature를 가지고 다양한 ML Classifier (3단계 - non stress (0), interrupt (1), timepressure(2)

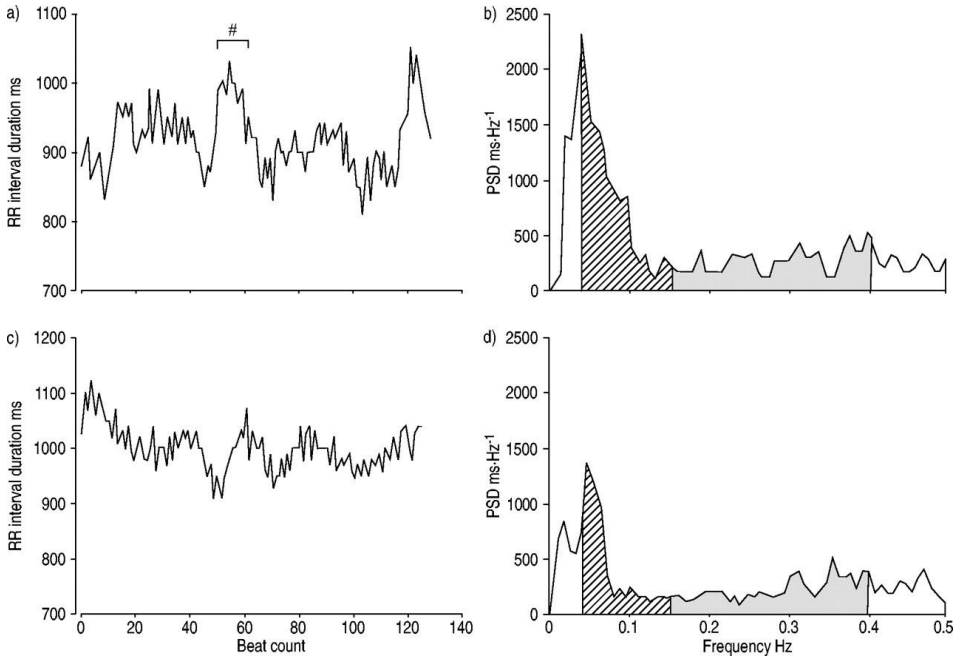
- MLP classifier - 0.86 (hyperparameter / random_state=1, max_iter=300)
- KNN class- 0.98 (hyperparameter / n_neighbors=3, Metric = minkowsi, weight = default)
- DecisionTree - 0.99 (hyperparameter / max_depth=30, random_state=2)

최종적으로 DecisionTree가 충분히 좋은 성능을 보였고,
이를 sklearn-porter를 통해 자바로 변환하여 탑재하였다.

Challenge 2) Stress Data Processing

SWELL Dataset Limitation – RR Interval





Challenge 2) Stress Data Processing

WESAD Dataset

15명의 약 2시간 가량의 BVP데이터

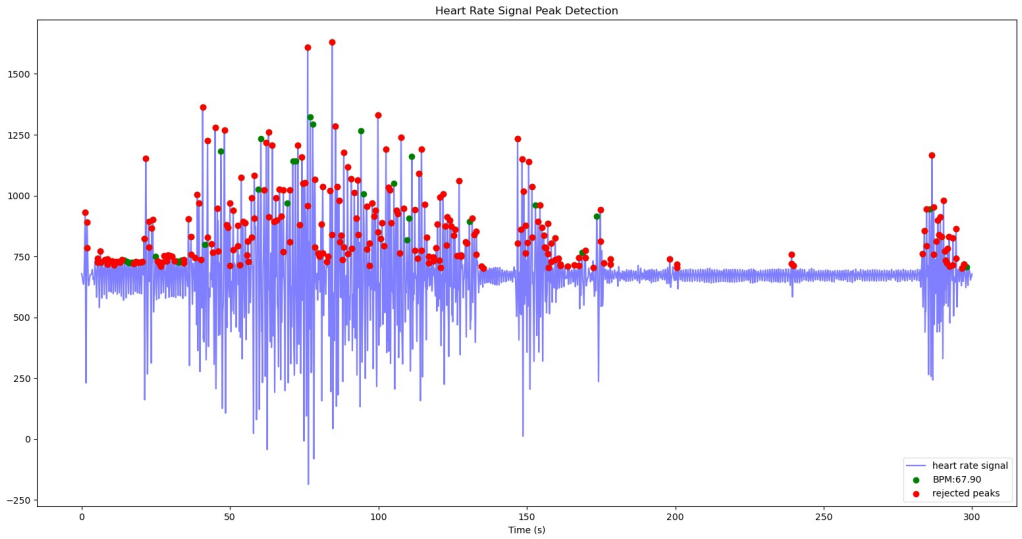
4단계 Labeling

- baseline, non stress, stress, amusement

Converted to Binary Classification

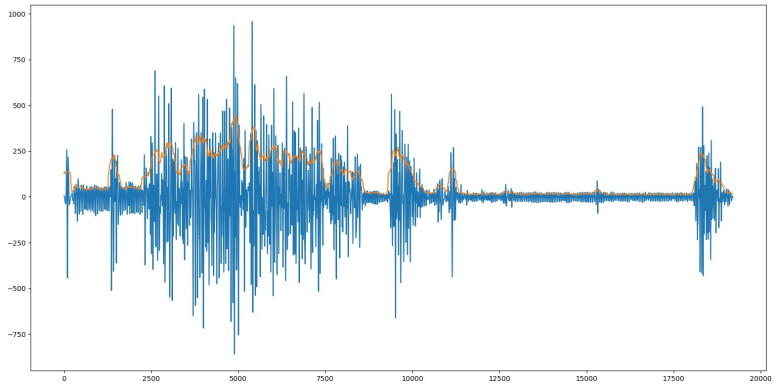
1 - Stress

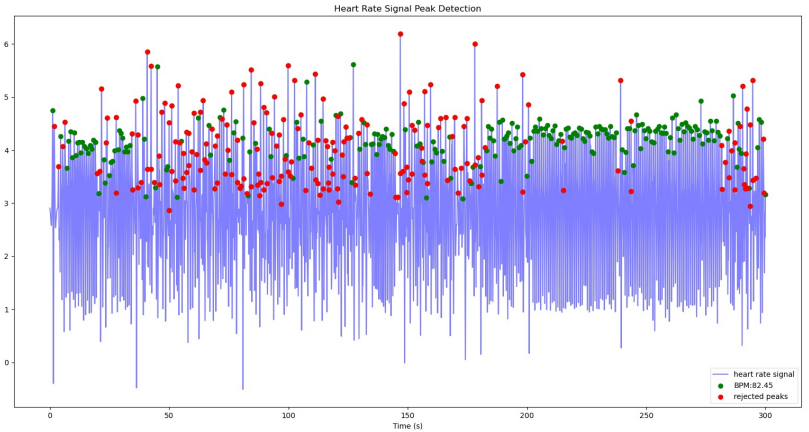
0 - Baseline, non stress, amusement



Challenge 2) Stress Data Processing

WESAD Dataset Challenge – Peak Detection





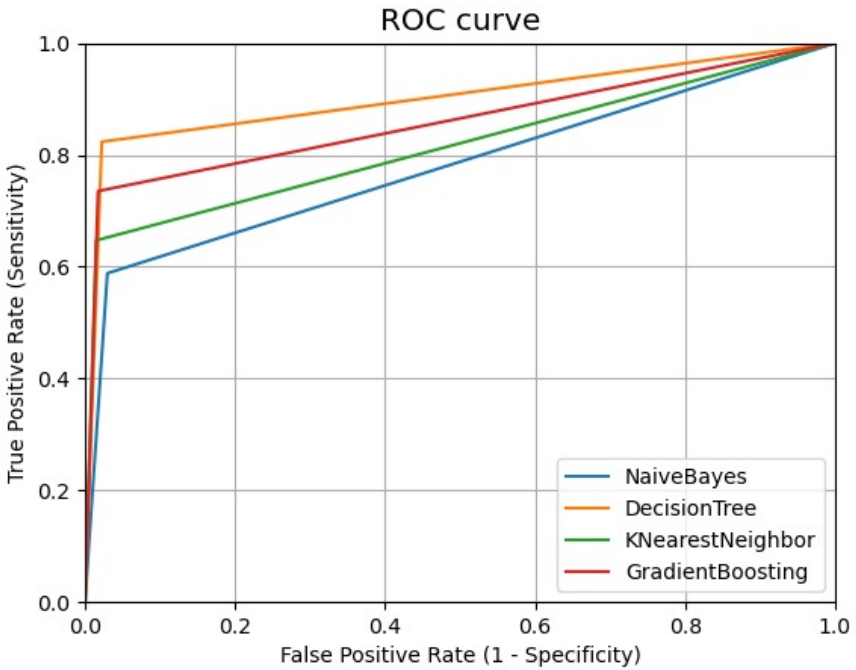
Challenge 2) Stress Data Processing

Classification Based on WESAD

- 전처리한 BVP에서 RR Interval 산출
- RR Interval에서 4가지 Major Features 산출
  - HR RMSSD pnn50 LF/HF

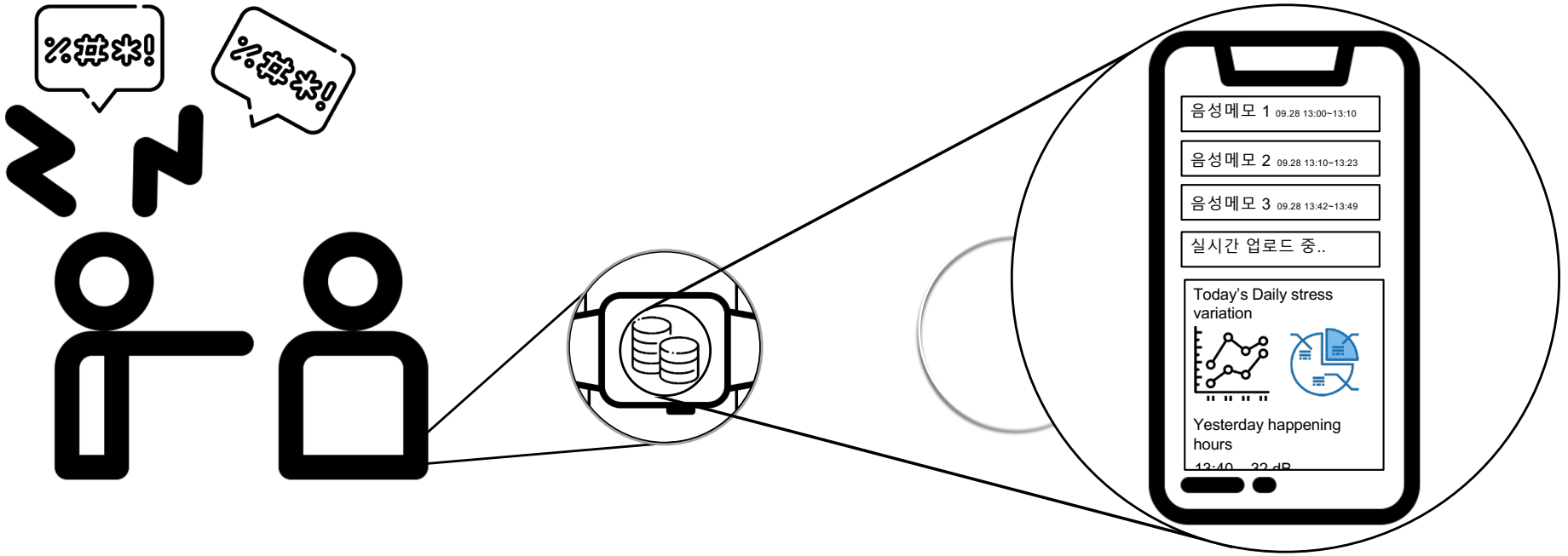
4가지 ML Classifier

- NB - acc : 0.940, f1 : 0.61
- DCT - acc : 0.965, f1 : 0.79
- KNN - acc : 0.958, f1 : 0.71
- GBC - acc : 0.963, f1 : 0.76



넓이가 제일 넓은 것은 Decision Tree이기에 이를 자바로 옮겼다. sklearn-porter 이용함

Final Delivery



Success Criteria

Data Collection

- 음성을 녹음할 수 있는가?
- PPG 데이터를 측정할 수 있는가?

Sensing Control

- 데이터 수집을 키고 끌 수 있는가?
- Power Optimization을 위한 Sensing Control?

Feature Extraction

- PPG Raw Data에서 LF/HF 산출이 가능한가?
- PPG Raw Data에서 기타 Feature 산출이 가능한가?

Stress Index Calculation

- ML 모델을 바탕으로 스트레스 단계 산출이 가능한가?
- Output으로 스트레스 지수를 산출할 수 있는가?

Speech-To-Text

- 음성을 인식하여 텍스트로 변환할 수 있는가?
- Cloud 기반의 작업 구조에 따른 처리와 최적화를 하였는가?

Visualization/Management

- 스트레스 지수 / 음성 / 텍스트를 시각화할 수 있는가?
- 저장된 데이터를 Export하고 공유할 수 있는가?

Project Role

구분	~ Midterm	~ Final
표재우	App Architecture / UI 구현 Watch → Mobile 데이터 전송 구현	Visualization 개선 Export & Share Feature
이동욱	HRV → Stress Study 및 Data 확보 PPG → Stress Data 모델 구현	PPG-To-Stress Model 개선
박지상	PPG 데이터 센싱 / 음성 녹음 구현 Stress 지수 및 음성 Visualization	Speech-To-Text 구현 Audio Segmentation
정재윤	PPG Raw Data LF/HF Processing 구현 PPG → Stress Data 모델 구현	PPG Raw Data Processing 개선

Q&A