**6조**

201711356천세진

201612066김지효

지도교수: 박소영 교수님

제출일: 2020년 6월 29일

프로젝트 설계서

[개인 프라이버시를 보호하는 협업 학습을 활용한 스마트폰 사용 패턴 분석 및 스트레스 예측]

목차

[1. 개요 2](#_Toc44255413)

[1.1 프로젝트의 주요 기능 2](#_Toc44255414)

[1.2 소프트웨어 Top-Level 구조 5](#_Toc44255415)

[2. 컴포넌트별 기능 정의 5](#_Toc44255416)

[2.1 데이터 수집용 앱 5](#_Toc44255417)

[2.2 최종 구현 앱 7](#_Toc44255418)

[3. 모듈별 설계 9](#_Toc44255419)

[3.1 데이터 수집용 앱 모듈 설계 9](#_Toc44255420)

1. 개요

# 1.1 프로젝트의 주요 기능

우리는 사용자의 개인 프라이버시를 보호하기 위해 여러 기술들을 적용해서 스마트폰 사용 패턴 수집 및 분석을 하고, 이를 통해 그들의 스트레스 수준을 예측하고자 한다.

**1단계 데이터 수집용 앱**

사용자가 자신의 Android 스마트폰에 직접 앱을 설치해 2주 간의 사용을 통해 사용자의 스트레스 척도와 스마트폰 사용 기록을 수집한다.

1. 최초 실행 시
   1. GPS, 알림에 대한 권한 요청을 한다.
   2. 배터리 최적화 기능 제외를 요청한다.
   3. 사용 기록 수집 권한을 요청한다.
   4. 프로젝트의 주제를 소개하고 수행 목적 및 팀원을 명시한다.
   5. 이 앱이 사용자의 어떠한 데이터를 수집하게 되는지 명시한다.
2. 스트레스 설문 데이터 수집
   1. 설문요청 푸쉬알림은 9시에 22시 사이에 2시간 간격으로 전송한다.
   2. 설문 문항은 Perceived Stress Scale (PSS) -14의 축약된 버전인 PSS-4 라는 스트레스 조사 설문지를 기반으로 작성한다.
   3. 푸쉬 알림을 클릭하면 바로 스트레스 설문 창을 띄워 설문을 수행하도록 한다.
   4. 설문 종료 버튼을 누를 때 앱 사용기록을 DB에 저장하고, Motion data를 1초 간격으로 1분 간 측정해 DB에 저장한다.
3. 스마트폰 사용 데이터 수집

* 데이터는 서버로 전송해 DB에 저장한다.
  1. 스마트폰 앱 사용 기록
     1. UsageStatsManager 서비스를 이용해서 15분마다 주기적으로 사용자의 앱 사용 내역과 사용량을 수집한다.
  2. 앱을 유형별로 분류한다.
     1. Utility: 캘린더, 지도, 시계, 날씨, 계산기 등
     2. SNS: Facebook, Instagram 등
     3. Messenger: 카카오톡, 페이스북 메신저 등
     4. Entertainment: 멜론, Youtube, 넷플릭스 등
     5. Browser: Chrome, Internet
     6. Game
  3. 분류 별로 앱 사용시간, 사용 선후관계 등을 분석하기 위한 데이터이다.
     1. GPS 데이터
        1. GPS 센서를 기반으로 한 Location Manager을 이용하여 사용자의 위치 데이터를 수집한다.
        2. 앱 최초 실행 시 집의 위치를 입력하도록 해 추후 집과 집이 아닌 곳을 구분할 수 있도록 한다.
     2. Motion 데이터
        1. Magnetic\_field와 Accelerometer 등의 모션 센서를 감지해서 Rotation vector를 산출한다.

**2단계 신경망 구축**

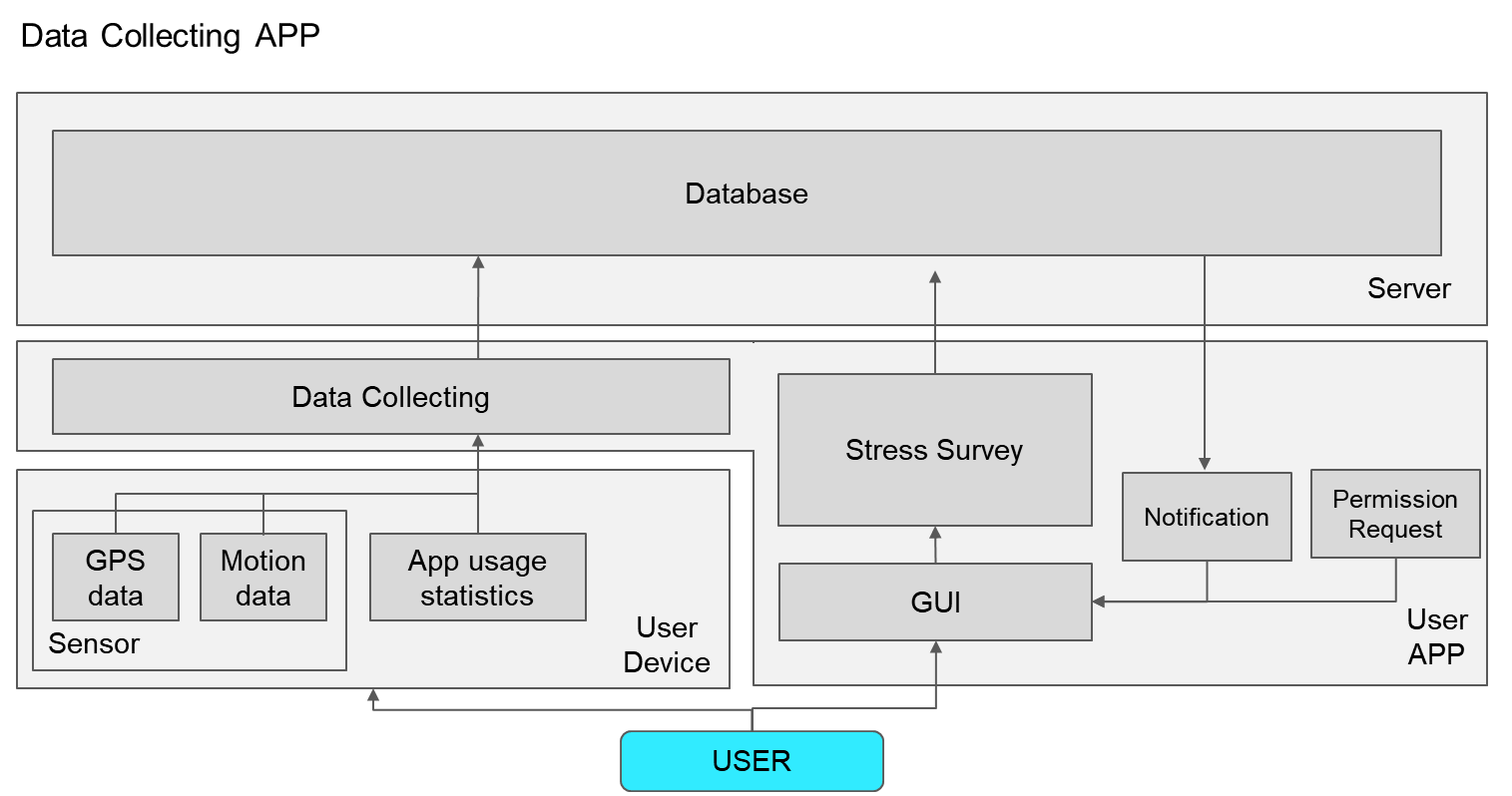
1. 신경망 구축
   1. 사용자들로부터 수집한 스마트폰 앱 사용 기록과 GPS 데이터, Motion 데이터를 기반으로 예측 모델을 구현한다. 사용 앱의 순서 등의 시계열 데이터를 중점적으로 한 모델링을 위해서 Long Short-term Memory model(LSTM) Architecture를 기반으로 구현을 고려하고 있으나 데이터 수집 후 신경망 설계 시 구체화할 예정이다.

**3단계 최종 구현 앱**

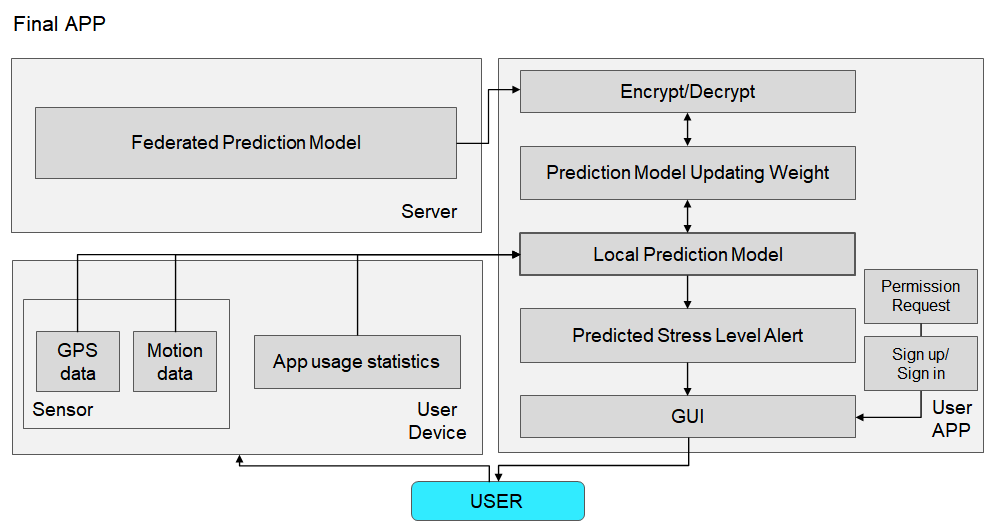
신경망을 적용한 최종 앱은 사용자의 스트레스 설문 조사 없이 스트레스 척도를 예측할 수 있다. 스트레스 척도 예측 모델은 중앙 서버에 연합학습 모델로써 존재한다. 스마트폰 사용 패턴으로 스트레스 수준 예측해 높다고 판단될 시 사용자에게 스트레스 수준 경고 알림을 보낸다.

1. 최초 실행 시
   1. GPS, Motion 센서, 알림에 대한 권한 요청을 한다.
   2. 앱을 사용하면 이 앱이 사용자의 어떠한 데이터를 수집하게 되는지 명시한다.
   3. 회원가입을 한다.
2. 연합학습을 통한 스트레스 척도 예측
   1. 수집하는 데이터는 1단계 -2 스마트폰 사용 데이터 수집 항목의 세부항목과 동일하다.
   2. 사용자 데이터의 Privacy 보존을 위해 앱 내에 존재하는 로컬 신경망 모델에서 수집된 데이터를 학습시킨 후 그 Weight 값만 서버로 전송해 Federated Prediction Model을 업데이트한다.
   3. 서버에서 갱신된 Weight값이 다시 유저의 앱으로 전송되어 로컬 모델을 업데이트 한다.
3. 사용자에게 스트레스 수준 알림
   1. 스트레스 수준이 일정 수준을 넘어 ‘높음’의 단계에 진입하면 사용자에게 경고 알림을 띄워 사용자가 자신의 스트레스에 대한 경각심을 가지도록 한다.
   2. 예측된 스트레스 수준을 그래프로 통계내어 볼 수 있도록 한다.

# 1.2 소프트웨어 Top-Level 구조

1. 데이터 수집용 앱 

2. 최종 구현 앱



2. 컴포넌트별 기능 정의

# 2.1 데이터 수집용 앱

**1. User Device**

1. GPS data
   1. 디바이스 내에 내장되어 있는 GPS 센서를 이용하여 현재 GPS 값을 데이터로 가져온다. 추후에 이 데이터를 스트레스 예측 모델링에 사용한다. Motion 센서에서 significant motion을 인지할 때마다 GPS의 변화를 감지해 저장한다.
2. Motion data
   1. 디바이스 내에 내장되어 있는 자이로 센서 및 가속도기를 이용하여 사용자의 움직임을 측정한다. 추후에 이 데이터를 이용하여 사용자의 자세를 예측한다.
3. App Usage Statistics
   1. 앱 사용 시간 데이터를 안드로이드 내부 API (Usagestatsmanager)를 이용하여 디바이스에서 어떤 앱을 얼마나 사용했는지에 대한 정보를 추출한다.

**2. User App**

1. GUI
   1. 사용자가 앱을 실행하였을 때 보이는 화면으로, 각 동작들의 수행 흐름에 따라서 화면에 표시한다. 디바이스에서 데이터를 가져오기 위한 권한 요청과 데이터 호출, 스트레스 설문조사를 하며 또한 스트레스 설문조사를 시각화하여 진행한다.
2. Stress Survey
   1. 사용자로 하여금 스트레스 설문 조사를 진행한다. 이 설문 결과를 저장하여 추후에 데이터들과 함께 서버로 전송한다.
3. Notification
   1. 사용자가 스트레스 설문 조사를 수행해야 할 시간이 되면 상태 바에 알림을 띄운다. 알림을 누르면 Stress Survey 화면으로 이동한다.
4. Permission Request
   1. 사용자의 디바이스에서 사용할 센서와 API를 이용하기 위하여 통신, GPS, 자이로 센서, 알림, API에 대한 request 권한을 얻는다. 만약 권한이 주어지지 않으면, 프로그램을 종료한다.

**3. Server**

1. Database
   1. 사용자의 디바이스에서 얻어낸 각 정보들과 스트레스 설문 조사 결과를 받아 저장한다. 추후 서버에서는 신경망 모델을 학습시킬 때 이 데이터를 이용한다.

# 2.2 최종 구현 앱

**1. User Device**

1. GPS data
   1. 디바이스 내에 내장되어 있는 GPS 센서를 이용하여 현재 GPS 값을 데이터로 가져온다. 추후에 이 데이터를 이용하여 활동 정도를 예측하기 위해 사용한다.
2. Motion data
   1. 디바이스 내에 내장되어 있는 자이로 센서를 이용하여 현재 자이로스코프 값을 데이터로 가져온다. 추후에 이 데이터를 이용하여 사용자의 자세를 예측하기 위해 사용한다.
3. App Usage Statistics
   1. 앱 사용 시간 데이터를 안드로이드 내부 API (Usagestatsmanager)를 이용하여 디바이스에서 어떤 앱을 얼마나 사용했는지에 대한 정보를 추출한다.

**2. User App**

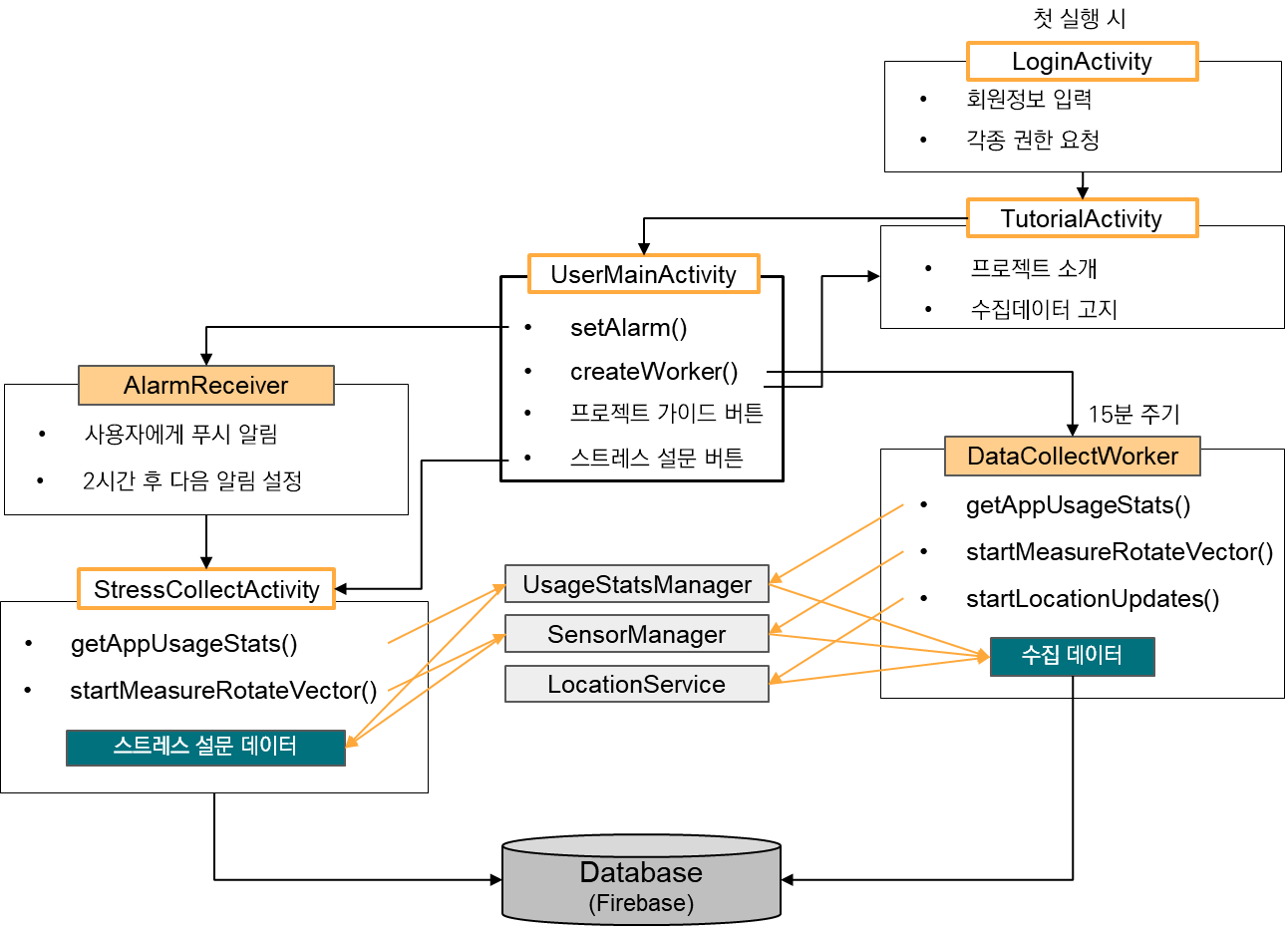
1. GUI
   1. 사용자가 앱을 실행하였을 때 보이는 화면으로, 각 동작들의 수행 흐름에 따라서 화면에 표시한다. 디바이스에서 데이터를 가져오기 위한 권한 요청과 데이터 호출, 스트레스 지수 예측 상세 정보 등을 시각화하여 보여준다.
2. Local Prediction Model
   1. 중앙 서버의 모델을 기반으로 한 Local Prediction Model이다. 이는 연합 학습에 이용하며, 유저의 사용 패턴에 따라서 weight 값이 조정된다. 학습 및 결과 도출은 사용자의 데이터 값으로 진행되며, 추후에 이 변경된 weight 값을 암호화하여 중앙 서버로 보낸다.
3. Predicted Stress Level Alert
   1. 사용자의 예측 스트레스 값이 계산되는 시점에, 사용자로 하여금 결과값을 앱 내부에서 확인할 수 있도록 상태 바에 알림을 띄운다. 이것을 클릭하면 상세 정보 페이지로 이동한다.
4. Permission Request
   1. 앱 최초 실행 시, 사용자의 디바이스에서 사용할 센서와 API를 이용하기 위하여 통신, GPS, 자이로 센서, 알림, API에 대한 request 권한을 요청한다. 만약 권한이 주어지지 않으면, 프로그램을 종료한다.
5. Prediction Model Updating Weight
   1. Local Prediction Model에 사용자 개인의 데이터를 학습시켜 사용자 특성에 맞게 weight 값을 조정한다.
6. Encrypt/Decrypt
   1. Local Model에서 변화된 weight 값을 서버로 전송하기 위하여 동형 암호로 암호화한다. 이후 서버로 전송한다.
   2. Server에서 받은 weight 값을 앱의 개인 키로 복호화하여 Local Model을 Update한다.

**3. Server**

1. Federated Prediction Model
   1. 데이터 수집 단에서 수집된 데이터를 이용하여 머신 러닝을 진행한 결과의 Model이다. 이는 연합 학습을 진행하는 각 디바이스의 update값으로 인하여 주기적으로 변한다.

3. 모듈별 설계

# 3.1 데이터 수집용 앱 모듈 설계



1. LoginActivity
   1. 앱 설치 시 처음에 보여지는 화면이다. 회원정보를 입력 받아 Database에 저장하는 작업을 수행한다.
   2. 데이터를 수집하기 위하여 필요한 권한들을 사용자에게 요청한다. 요청하는 권한들의 목록은 다음과 같다.
      1. Background, Fine, Coarce Location 허용
      2. Battery Optimized 제외
      3. Usagestats 접근
      4. Network State 접근
      5. Internet 접근
   3. 회원 가입 후, 사용자에게 유저 고유 키를 발급하여 SharedPreference를 이용하여 앱 내부에 저장한다. 이 키를 이용하여 Database에 접근 및 저장을 진행한다.
2. TutorialActivity
   1. 사용자에게 프로젝트 소개 및 데이터 수집의 목적, 사용 방법을 고지한다.
   2. 총 3개의 페이지로 구성되어 있으며, 버튼을 클릭 시 다음 페이지로 넘어간다.
3. UserMainActivity
   1. 사용자 정보 등록 (LoginActivity) 및 사용자 가이드 확인 (TutorialActivity) 후 실행되는 Activity이다.
   2. SetAlarm
      1. AlarmManager를 이용하여 사용자에게 2시간의 한번씩 푸시 알람을 생성한다.
      2. 사용자가 푸시 알람을 선택 시, AlarmReceiver 클래스를 이용하여 Stress 설문 조사 화면으로 이동한다. (StressCollectActivity)
   3. createWorker
      1. WorkManager를 이용하여 백그라운드에서 15분에 한번씩 실행되도록 설계한 데이터 수집 작업 (DataCollectWorker)를 Worker Queue에 Peroidic하게 추가한다.
      2. 이는 백그라운드에서 실행되며, 앱이 실행중이지 않아도 WorkManager가 Queue에 존재하여 15분에 한번씩 실행된다.
   4. 프로젝트 가이드 다시보기
      1. 사용자가 가이드를 다시 볼 필요가 있을 때, “프로젝트 가이드 다시보기” 버튼을 누르게 되면 TutorialActivity를 실행한다.
      2. TutorialActivity를 실행하고 나면, 다시 UserMainActivity로 돌아온다.
   5. 스트레스 설문 버튼
      1. 스트레스 설문을 수집하는 Activity (StressCollectActivity)로 이동한다.
4. AlarmReceiver
   1. 사용자에게 푸시 알람을 생성하여 띄운다. 그리고, 2시간 뒤에 다시 알람이 설정되게 한다.
   2. 사용자가 푸시 알람을 눌렀을 때, Stress 측정 Activity (StressCollectActivity)로 이동하게 한다.
5. StressCollectActivity
   1. 스트레스 설문
      1. 사용자에게 스트레스 설문을 보여준다.
      2. 사용자가 스트레스 설문을 완료하고 버튼을 누를 시, Database에 해당 스트레스 설문을 통한 스트레스 지수 값을 저장한다.
   2. getAppUsageStats
      1. 사용자가 스트레스 설문을 제출한 시점에, 그 전의 스트레스 설문조사를 진행한 시간부터 현 시점까지의 App Usage Stats를 가져와 Database에 저장한다.
      2. 이 때, 저장 순서는 최근에 사용한 앱 순서로 저장된다.
   3. StartMeasureRotateVector
      1. 사용자가 스트레스 설문을 제출한 시점에, 1분 동안 RotateVector 값을 측정한다.
      2. 측정이 끝난 후, 그 값을 Database에 저장한다.
6. DataCollectWorker
   1. 백그라운드에서 데이터를 수집하는 용도로 설계된 Job으로 이루어진 Worker 클래스이다.
   2. getAppUsageStats
      1. 15분 동안의 App Usage Stats를 가져와 Database에 저장한다.
      2. 이 때, 저장 순서는 최근에 사용한 앱 순서로 저장된다.
   3. startMeasureRotateVector
      1. 1분 동안 RotateVector 값을 측정한다.
      2. 측정이 끝난 후, 그 값을 Database에 저장한다.
   4. StartLocationUpdates
      1. 현재의 위치를 포함한 Location 정보를 측정한다.
      2. 측정이 끝난 후, 그 값을 Database에 저장한다.

# 3.2 신경망 모델 및 최종 구현 앱

현재 데이터 수집 경과를 지켜보고, 신경망 설계 준비 단계 중임.

- Google에서 진행하는 “G-Board Federated Learning” 논문 등 3개의 논문을 중점적으로 읽으며 Federated Learning Architecture에 대해서 공부하고 있으며, 추후 데이터 수집이 끝날 시 바로 신경망 설계를 진행할 예정임.