

[600003] Cloud Computing (F'18)

Term Project (Week 3, Final) Report

팀 정보

No.	이름	학번
1	임승현	20145159
2	정용석	20145169
3	장세영	20165160
4	이은지	20165152
팀 이름: A+		

<목차>

팀 정보	1
팀 정보	2
결과물 제출	2
보고서 작성 가이드	2
Part I: 최종 보고서	3
[Q 0][GitHub URL]	3
[Q 1][프로젝트 제목]	3
[Q 2][프로젝트 개요]	3
[Q 3][프로젝트 개발목표 및 개발/구현 내용]	4
[Q 4][어플리케이션 구성도]	5
[Q 5][기대효과 및 활용방안]	5
Part II: 절차서	6

Part III: 프로젝트 진행중 발생한 문제점 및 해결방안	11
[week2] [Q 2][문제점 및 해결방안]	11
Part IV: 개선방향.....	12

* 보고서 작성이 완료되면 <목차>를 업데이트 해 주세요. (방법: "목차" 클릭>"목차 업데이트">"목차 전체 업데이트")

팀 정보

임승현	20145159
정용석	20145169

-aws 얼굴인식 서비스 구현의 전반적인 부분 담당

장세영	20165160
이은지	20165152

-github 작성과 ppt 제작의 전반적인 부분 담당

결과물 제출

- SmartCampus 에 업로드 해야 하는 제출물 목록은 다음과 같습니다.
 - Report(프로젝트 보고서), **PDF 형식**
 - 프로젝트 발표자료 (ppt, pptx, pdf 등)
 - GitHub 프로젝트를 다운로드 한 zip 파일
- 제출 기한을 넘기면 자동으로 0 점 처리됩니다.
- 팀 과제이며, 팀별로 한 명만 제출하면 됩니다.

* 보고서 및 발표자료도 GitHub 에서 관리해야 합니다!!!

보고서 작성 가이드

- Week 3 시작: 2018. 12. 11. (Tue) 11:00am
- Week 3 마감: 2018. 12. 14. (Tue) 10:59am
- 매 주차별로 반드시 1 회 이상 GitHub 에 commit 한 내역이 있어야 합니다. (팀별로 1 회 이상, 개인별 1 회 아님). GitHub 에 commit 한 내역이 없을 경우, 큰 감점을 받게 됩니다.

본 보고서는 총 3 개의 파트로 구성되어 있습니다.

- Part I: 최종 보고서 [25pts]
 - 1 주차 보고서 내용 전체를 포함하여 작성하고, 그 동안 추가/수정된 내용이 있다면 그에 맞게 보고서를 수정해야 합니다.
- Part II: 절차서 [25pts]
 - 팀 프로젝트로 개발한 결과물을 누구나 손쉽게 따라할 수 있도록, 각 단계별로 어떻게 진행하는지 절차서를 작성하세요. 스크린샷을 첨부하면 더 좋습니다.
- Part III: 프로젝트 진행중 발생한 문제점 및 해결방안 [25pts]
 - 2 주차 보고서 내용중 "프로젝트 진행 중 발생하는 문제점 및 해결방안"에 대한 내용을 포함하는 내용으로 작성하세요.
 - 그 외에 추가로 발생한 문제점이 있었다면 해당 문제점 및 해결방안을 기술하세요..
- Part IV: 개선방향 [25pts]
 - 개발 결과물을 앞으로 어떻게 개선할 수 있을지에 대한 의견을 서술하세요.

Part I: 최종 보고서

[Q 0][GitHub URL]

팀 프로젝트를 위해 사용하는 GitHub 주소는? [10pts]

<https://github.com/seoyo1/Cloud-Computing>

[Q 1][프로젝트 제목]

프로젝트 제목은 무엇인가요? [10pts]

aws 내 인물인식 서비스를 이용하여 인물을 찾고 인물의 감정 읽기

[Q 2][프로젝트 개요]

● 프로젝트 선정 동기? [10pts]

aws 내 어떤 활용할 수 있는 내용이 있을까 살펴보다 인물 인식 서비스가 있다는 것을 알게 되었고 그것을 활용해 사람의 감정을 읽게 된다면 어떨까라는 생각을 해보았습니다 또한

컴퓨터가 사람의 감정을 읽었을 때 사람이 보고 판단하는 것과 비슷하게 인식할 수 있을지에 대한 궁금함이 생겨서 위와 같은 프로젝트를 선정하게 되었습니다.

- 프로젝트 내용? (프로젝트 내용에 대한 간단한 소개) [10pts]

인물 인식 서비스를 이용하여 사람을 찾아내고 찾아낸 사람의 감정이 실제 사람이 인식하는 타인의 감정과 비슷한지 알아보는 프로젝트(위에서 이야기하는 실제 사람은 팀원들에 해당합니다.)

[Q 3][프로젝트 개발목표 및 개발/구현 내용]

- 프로젝트 개발목표 [10pts]

aws 내의 인물인식 서비스를 이용해 사진속 인물을 인식해 그 사람의 감정을 읽고 사람이 인식했을 때의 타인의 감정과 비슷한지 알아보기

- 프로젝트 개발/구현 내용 [10pts]

사진 속 인물을 잡아냅니다

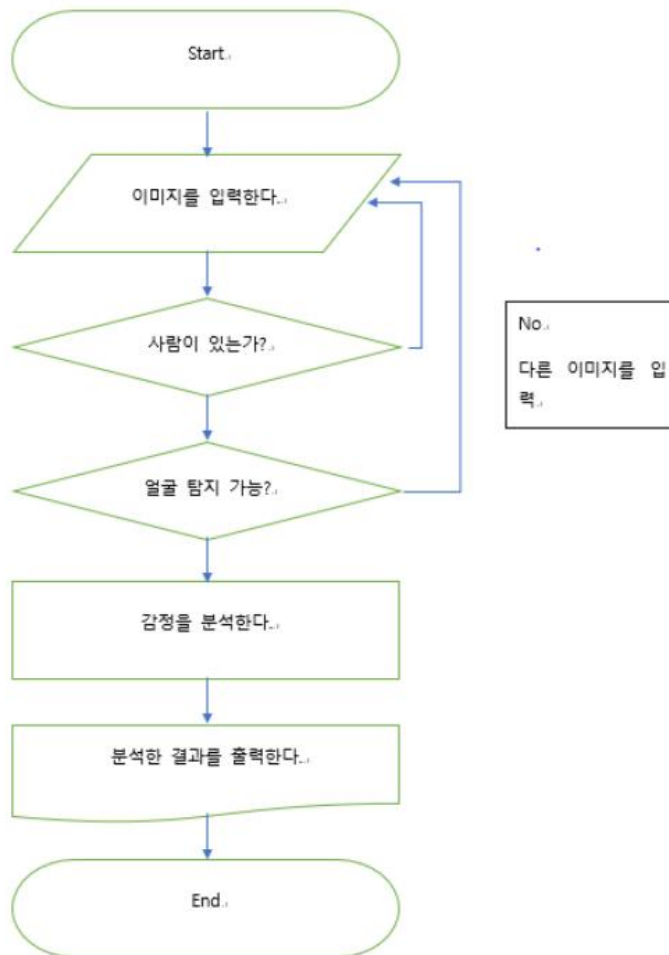
선택한 인물의 감정을 글로 표현합니다

글로 표현된 감정을 읽고 사람의 감정을 파악합니다

컴퓨터가 정해준 감정이 내가 생각하는 타인의 감정과 일치하는지 비교합니다

[Q 4][어플리케이션 구성도]

어플리케이션 구성도를 자세하게 그림으로 나타내세요. [20pts]



[Q 5][기대효과 및 활용방안]

프로젝트 결과물이 어떤 응용분야에서 어떻게 활용될 수 있을지 구체적으로 설명하세요.

[20pts]

컴퓨터 내에서 사람의 감정을 읽을 수 있다면 위와 같은 서비스를 핸드폰 어플리케이션에 연동하거나 여러 기기들에 연동시켜 눈이 보이지 않는 사람들과 안면인식 장애를 지닌 분들의 의사소통이 원활하게 이루어질 수 있을 것 입니다.

Part II: 절차서

<Overview>

Python 에서 AWS 의 Rekognition 을 이용하여 이미지에서 사람의 얼굴을 인식하여 감정을 분석하는 프로그램을 제작합니다.

<IAM 계정 설정>

Amazon Rekognition 과 같은 AWS 의 서비스는 액세스 할 때 자격 증명을 제공해야합니다. 이는 서비스가 해당 서비스가 소유 한 리소스에 액세스 할 수있는 권한이 있는지 여부를 확인할 수 있도록하기위한 것입니다. 콘솔에 암호가 필요합니다. AWS 계정에 대한 액세스 키를 만들어 AWS CLI 또는 API 에 액세스 할 수 있습니다. 그러나 AWS 계정에 대한 자격 증명을 사용하여 AWS 에 액세스하지 않는 것이 좋습니다. 대신 다음과 같이하는 것이 좋습니다.

AWS ID 및 액세스 관리 (IAM)를 사용하여 IAM 사용자를 생성

Step-by-step instruction

1. AWS 계정을 사용하여 IAM 콘솔에 AWS 계정 루트 사용자로 로그인합니다.
2. 탐색 창에서 사용자를 선택한 다음 사용자 추가를 선택합니다.
3. 사용자의 이름을 입력하고 AWS Management Console 액세스를 체크하고 사용자 지정 비밀번호를 설정한 다음 텍스트 상자에 새 암호를 입력합니다. 다음: 권한을 누르세요.

사용자 추가 1 2 3 4 5

사용자 세부 정보 설정

동일한 액세스 유형 및 권한을 사용하여 한 번에 여러 사용자를 추가할 수 있습니다. [자세히 알아보기](#)

사용자 이름* [다른 사용자 추가](#)

AWS 액세스 유형 선택

해당 사용자가 AWS에 액세스하는 방법을 선택합니다. 마지막 단계에서는 액세스 키와 자동 생성된 비밀번호가 제공됩니다. [자세히 알아보기](#)

액세스 유형* ☐ 프로그래밍 방식 액세스
AWS API, CLI, SDK 및 기타 개발 도구에 대해 액세스 키 ID 및 비밀 액세스 키를(를) 활성화합니다.

☒ AWS Management Console 액세스
사용자가 AWS Management Console에 로그인할 수 있도록 허용하는 비밀번호를(를) 활성화합니다.

콘솔 비밀번호* ☐ 자동 생성된 비밀번호
☒ 사용자 지정 비밀번호
 ☐ 비밀번호 표시

비밀번호 재설정 필요 ☒ 사용자가 다음에 로그인할 때 새 비밀번호 생성 요청
사용자가 비밀번호를 변경할 수 있도록 허용하는 IAMUserChangePassword 정책을 자동으로 가져옵니다.

* 필수 취소 다음: 권한

- 상단에 그룹에 사용자를 추가를 선택하고 그룹 생성을 선택 하십시오.
- 그룹 이름을 설정하고 정책 필터에서 AWS 관리 - 직무로 설정하고 AdministratorAccess를 선택합니다. 그런 다음 그룹 생성을 선택 하십시오.

정책 필터 필터 재설정

정책 유형

- ☐ 고객 관리형 (0)
- ☐ AWS 관리형 (393)
- ☒ AWS 관리 - 직무 (10)

정책 사용

- ☐ 권한에 사용됨 (1)
- ☐ 경계에 사용됨 (0)
- ☐ 사용되지 않음 (402)

정책 이름	유형	사용 용도
<input checked="" type="checkbox"/> AdministratorAccess	직무 기반	Permissions policy (1)

- 그룹 목록으로 돌아가서 새 그룹의 확인란을 선택하십시오. 필요한 경우 새로 고침을 선택 하여 목록에서 그룹을 봅니다.

7. 다음: 검토를 선택 하여 새 사용자에게 추가 할 그룹 구성원 목록을 확인합니다. 계속할 준비가 되면 사용자 만들기를 선택 하십시오.

<AWS CLI 및 AWS SDK 설정>

다음 단계에서는이 설명서의 예제에 사용되는 AWS CLI (Command Line Interface) 및 AWS SDK 를 설치하는 방법을 보여줍니다. AWS SDK 호출을 인증하는 여러 가지 방법이 있습니다. 이 가이드의 예제는 AWS CLI 명령 및 AWS SDK API 작업 호출에 기본 자격 증명 프로필을 사용한다고 가정합니다.

Step-by-step instruction

1. 사용할 AWS CLI 및 AWS SDK를 다운로드 하여 설치하십시오.

Python 용 sdk: <https://boto3.amazonaws.com/v1/documentation/api/latest/index.html>

AWS CLI: <https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-install.html>

2. IAM 사용자 만들기에서 만든 사용자의 액세스 키를 만듭니다.
 - A. IAM 콘솔의 탐색 창에서 사용자를 선택합니다.
 - B. IAM 사용자 만들기에서 만든 사용자 이름을 선택합니다.
 - C. 보안 자격 증명 탭을 선택합니다.
 - D. 액세스 키 생성을 선택 합니다. 그런 다음 .csv 파일 다운로드를 선택하여 액세스 키 ID와 비밀 액세스 키를 컴퓨터의 CSV파일에 저장합니다.

대시보드

그룹

사용자

역할

정책

자격 증명 공급자

계정 설정

자격 증명 보고서

암호화 키

요약

사용자 ARN `arn:aws:iam::476934275291:user/Administrator`

경로 `/`

생성 시간 2018-12-10 21:12 UTC+0900

권한

그룹 (1)

태그

보안 자격 증명

액세스 관리자

로그인 자격 증명

요약 • 콘솔 로그인 링크: <https://476934275291.signin.aws.amazon.com/console>콘솔 비밀번호 활성화 (2 일째 마지막으로 로그인) | [관리](#)할당된 MFA 디바이스 할당되지 않음 | [관리](#)

서명 인증서 없음

액세스 키

AWS 서비스 API에 대한 보안 REST 또는 HTTP Query 프로토콜 요청을 생성하려면 액세스 키를 사용하십시오. 보호를 위해 비밀 키를 다른 사람과 공유하지 마십시오.

액세스 키 만들기

액세스 키 ID	생성 완료	마지막 사용
AKIAI2FEI45UQT7K40EQ	2018-12-10 21:26 UTC+0900	ap-northeast-2에서 2018-12-13 17:50 UTC+0900에 s3 사용

AWS CodeCommit에 대한 SSH 키

SSH 퍼블릭 키를 사용하여 AWS CodeCommit 리포지토리에 대한 액세스를 인증합니다. [자세히 알아보기](#)

3. AWS CLI를 설치한 다음 명령 프롬프트에서 `aws configure`를 입력하여 AWS SDK에 대한 자격 증명 및 지역구성을 설정합니다.

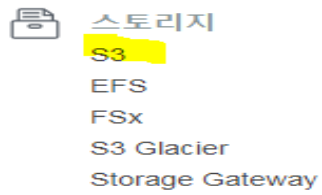
```
C:\Users\Administrator>aws configure
AWS Access Key ID [*****40EQ]:
AWS Secret Access Key [*****Ydry]:
Default region name [ap-northeast-2]:
Default output format [None]:
```

<S3 버킷 생성>

이미지를 인식하기 위해 파일을 업로드 시킬 S3 버킷을 생성합니다.

Step-by-step instruction

1. AWS 서비스에서 스토리지의 S3를 선택합니다.
2. 버킷 만들기를 클릭하여 버킷이름을 설정하고 다음을 눌러 버킷을 생성합니다.



이후엔 Python 을 통해 코드작업을 합니다.

-github 사이트 내 python 코드

<https://github.com/seoyo1/CloudComputing/blob/master/WEEK3/rekog%20pyhton.txt>

Part III: 프로젝트 진행중 발생한 문제점 및 해결방안

[week2] [Q 2][문제점 및 해결방안]

- 프로젝트 진행 중, 어떤 문제점이 발생했나요? [30pts]

얼굴 인식 후 사람의 표정을 이모티콘으로 표현하는 부분과 만일 사람이 다수가 있을 때 모든사람의 얼굴을 이모티콘으로 표현할지 한사람의 얼굴만 표현할지에 대한 문제가 발생하였습니다

- 문제점에 대한 해결책은 무엇인지 설명하세요. 아직 해결방법을 모를 경우, 어떤 식으로 접근해서 문제를 해결할 지에 대한 계획을 설명하세요 [30pts]

현재 이모티콘을 얼굴에 대입시키는부분에 대해 구현을 시키지 못하였는데 인터넷에 이모티콘을 구현시키는 방법이 있기 때문에 그것을 보고 현재 진행된 부분에서 추가적으로 구현을 할 예정이고 이미지에 사람이 많이 있을 경우에는 모든 사람의 얼굴에 표정이모티콘을 넣거나 한사람만 이모티콘을 넣게된다면 좌표를 이용해서 한사람의 얼굴에만 이모티콘을 넣게 할 예정입니다.

- 그 외 다른 문제가 생긴 것이 있다면 무엇이 있나요?

[문제점]

얼굴인식을 하는 부분까지는 순조롭게 잘 진행되었는데 사람의 얼굴이 인식하기 어려운 정도로 작을경우에는 얼굴을 인식하지 못한다는 문제점이 발생했습니다.

[해결방안]

현재 문제가 컴퓨터가 사람의 얼굴을 인식하기 어려울 정도로 작아서 인식이 불가능한것이기 때문에 사진을 얼굴 픽셀이 너무 깨지지않을 정도로 확대시키면 인식이 가능해지는 것을 알 수 있었습니다.

Part IV: 개선방향

Aws 얼굴인식 서비스를 이용해서 사람의 감정을 인식하고 인식한 감정을 토대로 사람의 감정을 누구나 쉽게 알아볼 수 있도록 인물의 얼굴을 (해당 이미지 속 얼굴을) 이모티콘으로 변경시킬 수 있을 것이라고 생각합니다.

위의 방법이 도입되고, Aws 얼굴인식 서비스내의 감정 인식이 잘 이루어진다는 것이 보장된다면 사람의 기분을 이모티콘으로 인식해 쉽게 알아낼 수 있을 것이기 때문에 위와 같이 개선되면 좋을 것이라고 생각합니다.

[끝]