

ai platform 활용 – 구현 완료 외부 제공 라이브러리 사용

<< AI란 >>

0. Artificial intelligence.

1.인간의 학습능력, 추론능력, 지각능력, 그외에 인공적으로 구현한 컴퓨터 프로그램 또는 이를 포함한 컴퓨터 시스템

2.인간의 지능을 기계 등에 인공적으로 시연(구현)한 것

3.AI 개발은 학습을 통해 발전.

4.인간은 교사의 역할을 맡고 해결책이 옳은지 또는 수정이 필요한지에 대한 피드백을 제공.

- 강아지사진 보여주고 강아지야 학습후 (여러번)

- 다른 강아지사진 보여주고 이거 뭐야 질문시 – 답변 강아지야 유도

5.시간이 지남에 따라 AI 에 사용할 수 있는 정보가 많아지고 점점 인간이 개입 할 필요가 적고 AI가 스스로 작업을 해결.

6. AI의 장점이 남용되지 않도록 명확한 규칙 필요.

AI 에 설정된 설정된 목표를 잘못 해석하지 않고 한계를 초과하지 않는지 확인 필요.

AI는 윤리성 및 데이터 보호와 같은 사회적 가치 및 법률과 일치해야 함.

예) 안녕하세요 인사시 – 욕설로 응답 하라는 학습은 안됨

7. AI 매핑 불가한 부분

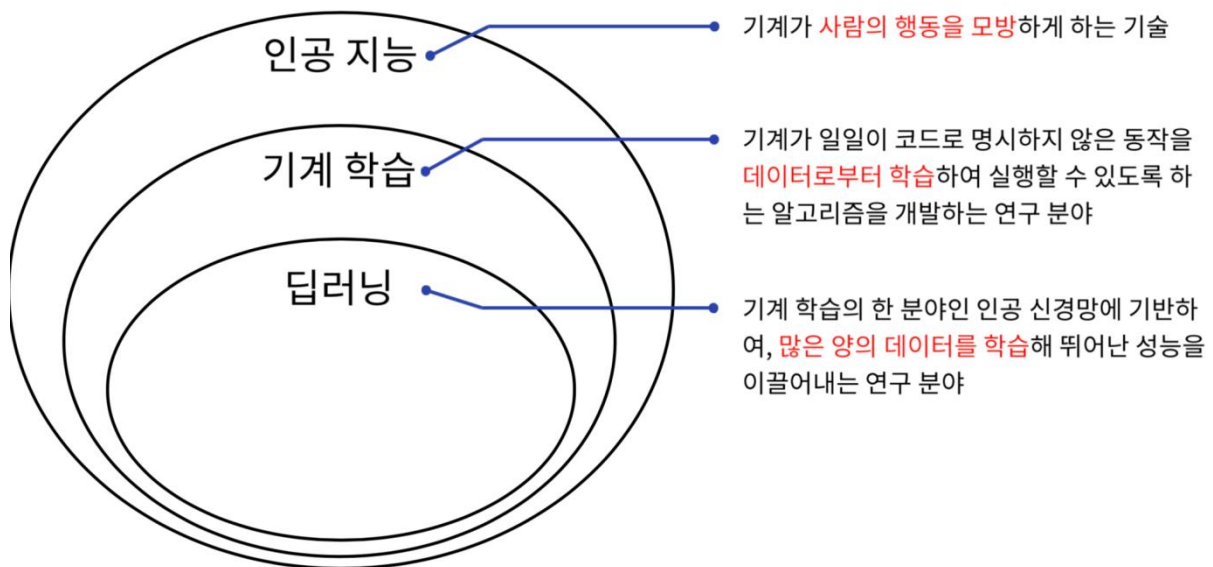
AI는 인간의 직관, 창의성 및 공감 등 데이터로 변환 할 수 없는 속성 표현 불가,

직감, 창의성 및 공감은 사람마다 주관적으로 인식.

x-ray 분석 결과 이용 가능 / 심리 치료 이용은 불가.

<<인공 지능 학습 분야>>

딥러닝 vs. 기계 학습 vs. 인공지능



<<ai 서비스 활용>>

이미지인식	인물 사진 자동 분류 골프 자세 교정 프로그램 차량 사고 생존자 확인 시스템 안면 인식 출결 확인
OCR (텍스트이미지인식)	공공 요금 고지서 인식 명함인식 차량 번호판 인식
STT	뉴스 자막 서비스 화상 회의 자동 기록 서비스 강의 자동 노트 서비스
TTS	운전 중 문자 읽어 주기 책 읽어주는 서비스
CHATBOT	고객상담센터

AI 서비스 활용 서버 (스프링 BOOT)	java, servlet, jsp, spring html, javascript, css, jquery
AI 서비스 구현 제공 api	naver cloud platform api id, pw, url

- 네이버 클라우드 플랫폼 api 활용 방법

1. 계정과 접속 URL 확인하기
2. 암호 변경(필수)->기억하기
3. 오른쪽 상단 '콘솔' 클릭
4. 가운데 위 '콘솔 시작하기 >' 버튼 클릭
5. 왼쪽 메뉴 'services' 클릭
6. 'ai-naver api' 클릭
7. 화면 하단 스크롤
8. '+application등록' 클릭
9. 각 clova 항목 서비스 옆 서비스설명/요금안내 클릭 개발가이드 클릭
읽어보자
10. 해당 서비스 항목 선택
- 11.

Classic / AI-NAVER API / Application

< Application 1

Application 이름 설정

AI - Naver Service 를 이용할 Application 을 등록하세요

Application 이름

ai-platform-xxxx

Service 선택

Application에서 이용할 Service 를 선택하세요



☐ CLOVA

☐ CLOVA Speech Recognition (CSR)

☒ CLOVA Face Recognition (CFR)

☐ CLOVA Voice - Premium

☐ CLOVA Sentiment

☐ CLOVA Summary

서비스 설명/요금안내

서비스 설명/요금안내

서비스 설명/요금안내

서비스 설명/요금안내

서비스 설명/요금안내

개발 가이드

개발 가이드

개발 가이드

개발 가이드

개발 가이드

위 항목 외에도 필요한 항목들 체크한다.

아래로 스크롤한다

서비스 환경 등록

Application에서 이용할 서비스 환경을 등록하세요

Web 서비스 URL(최대 10개)

+추가

?

✓

✕삭제

Android 앱 패키지 이름(최대 10개)

+추가

iOS Bundle ID(최대 10개)

+추가

취소

등록

웹서비스 url에 아무 url 입력 - 추가 버튼 - 등록 버튼 클릭한다.

Classic / AI-NAVER API / Application

Application

등록한 Application 정보를 확인하고 관리합니다.

+ Application 등록

개발 가이드

상품 더 알아보기

새로고침

삭제

App 이름	서비스구분	당일 사용량	당월 사용량	한도 및 알림 설정	등록일
<input type="checkbox"/> kdigital <div>인증 정보</div> <div>변경</div>	<div>CLOVA Face Recognition (CFR)</div> <div></div>	<div>0%</div> <div></div>	<div>0/50,000 회</div> <div>0%</div> <div></div>	<div>0/300,000 회</div> <div>한도 및 알림 설정</div>	2021-09-24

화면을 확인한다

인증 정보를 클릭한다.

client id와 client secret 값은 네이버 서버의 ai 서비스를 사용하기 위한 인증 데이터이다.

활용할 AI Services

1. CLOVA Speech	음성 메모, 영상 자막 생성, 통화 녹취록 관리 등의 서비스를 만들 때 활용할 수 있는 음성인식 서비스
-----------------	---

2. CLOVA Voice	입력된 텍스트를 RESTful API 방식으로 전달하면 서버에서 인식하는 API
3. CLOVA Speech Synthesis(CSS)	입력한 텍스트를 자연스러운 목소리로 재생해주는 음성 합성 API
4. CLOVA Premium Voice(CPV)	CLOVA의 인공지능 기술로 더 사람같은, 고품질의 합성을 제공
5. CLOVA Speech Recogniton(CSR)	사람의 목소리를 텍스트로 바꿔주어 다양한 음성 인식 서비스에 활용
6. CLOVA OCR	인쇄물 상의 글자와 이미지를 디지털 데이터로 자동으로 추출하는 기술
7. CLOVA Face Recognition(CFR)	이미지 속의 얼굴을 감지하고 인식하여 얻은 다양한 정보를 제공
8. Pose Estimation	이미지 속의 사람을 감지하고 몇명이 어떤 포즈를 취하고 있는지에 대한 좌표 정보
9. Object Detection	이미지 내 사람 및 자동차 등 객체의 타입과 위치를 감지하여 정보를 제공
10.CLOVA Chatbot	ai 상담
11. API Gateway	안정적인 API 호출을 돕는 다양한 관리 기능과 모니터링 대시보드 제공

CLOVA Face Recognition(CFR) 개념 이해

CLOVA Face Recognition(CFR) 실습

1. 네이버 ai 서비스 사이트 로그인

얼굴인식-id, 암호,

개발자 가이드+ 자바코드 연동

json

```
{"변수명":"값" , , , , }
```

info.faceCount

info.size.width

faces[0].celebrity.value

```
{
```

```
"info":
```

```
{"size":{"width":264,"height":200},"faceCount":1}
```

```
,
```

```
"faces":
```

```
[{"celebrity":{"value":"송혜교","confidence":0.631658}}]
```

```
}
```

1명	<pre>{"info":{"size":{"width":264,"height":200},"faceCount":1},"faces":[{"celebrity":{"value":"송혜교","confidence":0.631658}}]}</pre>
2명	<pre>{"info":{"size":{"width":223,"height":226},"faceCount":2},"faces":[{"celebrity":{"value":"송중기","confidence":0.879385}}, {"celebrity":{"value":"이연희","confidence":0.173051}}]}</pre>
사물	<pre>{"info":{"size":{"width":259,"height":195},"faceCount":0},"faces":[]}</pre>
2m	error!!!!!! responseCode= 400

b초 과	java.io.IOException: Server returned HTTP response code: 400 for URL:
json	<pre>{ "info":{"size":{"width":264,"height":200}, "faceCount":1}, "faces":[{"celebrity":{"value":"송혜교","confidence":0.631658}}] }</pre> <p>info변수 info.size.width-이미지사진가로크기 info.size.height-이미지사진세로크기 info.faceCount-얼굴 수 faces변수 faces[0]. celebrity – 유명인이름 faces[0]. confidence-닮은정도(0-1)</p>

2. 스프링부트 실행

0. naverai라는 이름의 실습용 스프링부트 프로젝트를 생성한다.

- 프로젝트name : naverai
- 자바버전 8
- 패키지명 edu.spring.naverai
- 항목은 spring web 선택한다.

1. images폴더의 이미지파일들만 압축 공유.

2. naver ai cfr 콘솔 개발자 가이드에서 키, 암호, url 확인한다.

3. naver ai cfr 콘솔 개발자 가이드에서 자바 샘플 예제 복사하여 현재 프로젝트의 crfr 패키지 생성 – APIExamFace.java 파일 만들어 붙여넣는다.

4. 이미지파일명 부분 수정, 키, 암호 추가한다.

5. run as - java application으로 실행한다

<오류코드>

Bad Request error!!!!!! responseCode= 400	2mb 초과 파일인지 확인 image 파라미터명 확인
Unauthorized error!!!!!! responseCode= 401	인증 키와 암호 확인 후 수정 naver console에서 해당 application의 인증 정보 확인
Not Found error!!!!!! responseCode= 404	apiurl 확인
java.io.FileNotFoundException:	이미지파일의 경로 및 파일명 확인

<결과분석>

내 클래스 - 네이버 ai 유명인 서비스 요청-응답 결과

<pre>{ "info": { "size": {"width":264,"height":200}, "faceCount":1 }, "faces":[{ "celebrity": {"value":"송혜교","confidence":0.721762} }] }</pre>	<p>info:전체결과객체변수 info.size: 이미지파일크기변수 info.size.width:가로크기 info.size.height:세로크기 info.faceCount:발견된 얼굴수 faces:얼굴정보저장객체변수 faces[0].celebrity: ㉨은 유명인 정보변수 faces[0].celebrity.value:유명인이 름 faces[0].celebrity.confidence:㉨ 은확률(최대 1)</p>
--	--

7. 2명 이상의 사람 얼굴이 보이는 이미지사진으로 변경 테스트

String imgFile = "C:/Users/user/Desktop/images/bangtan7.jpg";

<pre>{"info": {"size":{"width":640,"height":407},"faceCount":7}, "faces":[{"celebrity":{"value":"유겸","confidence":0.01}}, {"celebrity":{"value":"엄현경","confidence":0.0940699}}, {"celebrity":{"value":"유진","confidence":0.0625277}}, {"celebrity":{"value":"현승민","confidence":0.154775}}, {"celebrity":{"value":"정은지","confidence":0.34685}}, {"celebrity":{"value":"김태형","confidence":0.01}}, {"celebrity":{"value":"이일민","confidence":0.0777358}}]</pre>	
--	--

```
}
```

6. mvc 구조로 변경하여 브라우저에서 이미지 전달받아서 테스트

6-0. pom.xml 에 jsp 사용 위한 dependency 추가

```
<!-- for jsp -->
<dependency>
  <groupId>org.apache.tomcat.embed</groupId>
  <artifactId>tomcat-embed-jasper</artifactId>
  <scope>provided</scope>
</dependency>
<!-- for jsp jstl -->
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>jstl</artifactId>
</dependency>
<!--for tomcat restart -->
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-devtools</artifactId>
  <optional>true</optional>
</dependency>
```

6-1.application.properties

```
server.port=8082
spring.mvc.view.prefix=/WEB-INF/views/
spring.mvc.view.suffix=.jsp
```

브라우저

c:/...바탕화면/ai_images 폴더

뷰 - 파일목록 보여주고(접근 url)

/faceresult?image=파일명

Controller /faceresult 메소드

네이버 cfr 서비스 호출 서비스클래스 객체 메소드

응답 - json - 분석 - 이름, 2명

6-2.WebConfig.java 생성

```
@Configuration
public class WebConfig implements WebMvcConfigurer {

    @Override
    public void addResourceHandlers(ResourceHandlerRegistry registry) {
registry.addHandler("/naverai/**")//url
        .addResourceLocations("file:///C:/Users/user/Desktop/images/");//경
로
    }
}
```

6-3.MyNaveraiapplication.java 수정

```
@SpringBootApplication
@ComponentScan(basePackageClasses = FaceController.class)
@ComponentScan

public class MynaveraiApplication {

    public static void main(String[] args) {
        SpringApplication.run(MynaveraiApplication.class, args);
    }
}
```

}

6-4. mvc(cfr 패키지)

NaverFaceController	"/faceinput"->파일명 리스트 전달 뷰 "face"
NaverService	edu.spring.naverai패키지 인터페이스 public interface NaverService { String test(String image); }
FaceService	NaverService 상속 test 메소드 내에 APIExamFace.java의 main 메소드 내용을 복사 키, 암호, url 그대로. 이미지파일명 매개변수로 전달받도록 수정 . 리턴값은 네이버ai서버의 응답으로 리턴
faceinput.jsp	images폴더의 파일들을 보여주는 뷰 이 중 파일 1개 클릭하면 파일명만 전송
face.jsp	클릭한 이미지의 파일명 받아서 유명한 닳은꼴 분석 결과 보여주는 뷰
WebConfig.java	edu.spring.naverai패키지 이미지파일들을 저장한 외부 경로(현재프로젝트외부)를 스프링부트에서 url로 매핑
pom.xml	
java script	let json = JSON.parse(\${model});

<https://jsonviewer.arianv.com/>

\${result}---> jsp String타입 간주

<% info.faceCount %>==> error

<script>

```
var obj = ${result};
obj.info.faceCount
</script>
{
  "info":{"size":{"width":284,"height":319},"faceCount":1},
  "faces":[{"celebrity":{"value":"김연아","confidence":1.0}}]
}
```

사람수 : info.faceCount

유명인이름 : faces[0].celebrity.value

확률 : faces[0].celebrity.confidence * 100 + "%"

얼굴감지

```
{
  "info":
  {"size":{"width":264,"height":200},"faceCount":1},
  "faces":[
    {"roi":{"x":80,"y":45,"width":61,"height":61},"landmark":null,"gender":{"value":"female","confidence":0.999912},"age":{"value":"24~28","confidence":0.856547},"emotion":{"value":"neutral","confidence":0.999967},"pose":{"value":"right_face","confidence":0.61542}}
  ]
}
```

```
{
  "info":{"size":{"width":251,"height":201},"faceCount":3},
  "faces":[
    {"roi":{"x":58,"y":127,"width":23,"height":23},"landmark":null,"gender":{"value":"female","confidence":0.99919},"age":{"value":"56~60","confidence":0.0951743},"emotion":{"value":"neutral","confidence":0.737852},"pose":{"value":"left_face","confidence":0.445099}},
    {"roi":{"x":48,"y":27,"width":18,"height":18},"landmark":null,"gender":{"value":"female","confidence":0.999689},"age":{"value":"32~36","confidence":0.26535},"emotion":{"value":"neutral","confidence":0.499429},"pose":{"value":"right_face","confidence":0.967087}},
    {"roi":{"x":162,"y":28,"width":29,"height":29},"landmark":null,"gender":{"value":"female","confidence":0.999657},"age":{"value":"15~19","confidence":0.408967},"emotion":{"value":"neutral","confidence":0.898834},"pose":{"value":"left_face","confidence":0.784178}}
  ]
}
```

```
{ "a":"aaa", "b":{"  }  , "c" : [{}, {} , {}      ]  }
```