

# ***RoadScanner***

*Traffic Sign Recognition AI Web Service*

F1(1조)

---

김정민 송세림 오동훈 이주영  
장태근 차지욱 차현경 최유정

## 목차 table of contents

- 1 프로젝트 개요
- 2 프로젝트 구성
- 3 수행 방법 및 절차
- 4 프로젝트 수행 결과
- 5 자체 평가 의견



Part 1

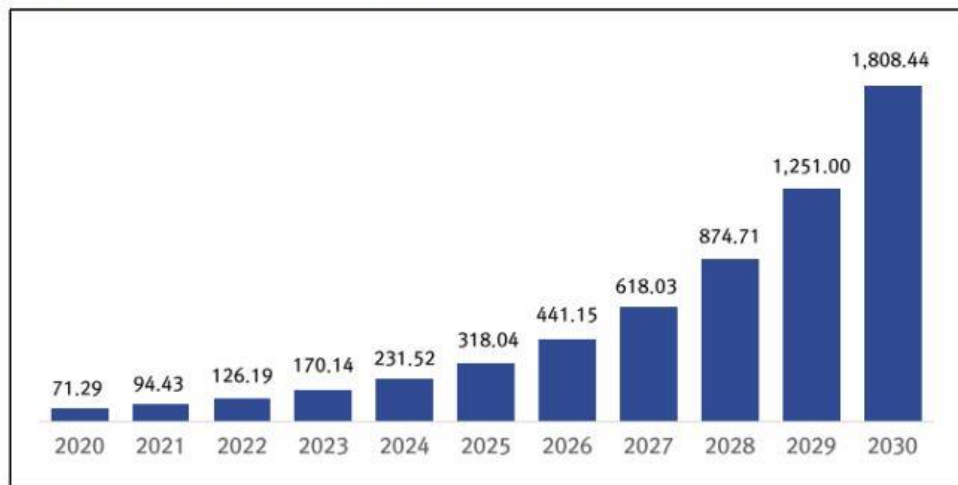
# 프로젝트 개요





[그래프 1] 2020~2030년 자율주행차 시장 규모

(단위 : 십억 달러)



출처 : Precedence research(precedenceresearch.com)



교통 표지판 인식 AI 웹 서비스

1. 딥러닝 알고리즘을 활용하여 교통 표지판을 식별하고 분류하는 모델 개발
2. 사용자의 직관적 이해를 돕기 위해 인식 성능 및 결과 시각화
3. 사용자 피드백을 통한 모델 평가 및 성능 개선

Part 2

# 프로젝트 구성



이름	업무 분류	상세 업무
김정민 (조장)	파일 업로드	Oracle DB구축, 업로드 CRUD, 이미지 관리 기능, 그래프 구현, Flask API 연결
차현경		업로드 CRUD, 업로드 기능, 결과 출력 기능, AWS S3버킷/RDS/EC2 연결 및 배포
이주영	게시판	댓글 CRUD, 게시판 페이징/검색, 게시판 관리 기능
장태근		게시판 구조 설계, 게시판 CRUD 구현, S3 업로드 서비스 기능
최유정		게시판 JSP 연결 및 CSS 디자인
송세림	로그인	탈퇴 기능, 메인 페이지, 공통파일 CSS
오동훈		회원 CRUD, 관리자 기능 및 페이징, 암호화 기능, 로그인 / 회원가입 / ID&PW 찾기 기능
차지욱		마이페이지 CRUD, 메인페이지, 공통파일 제작, 비밀번호 재설정 기능

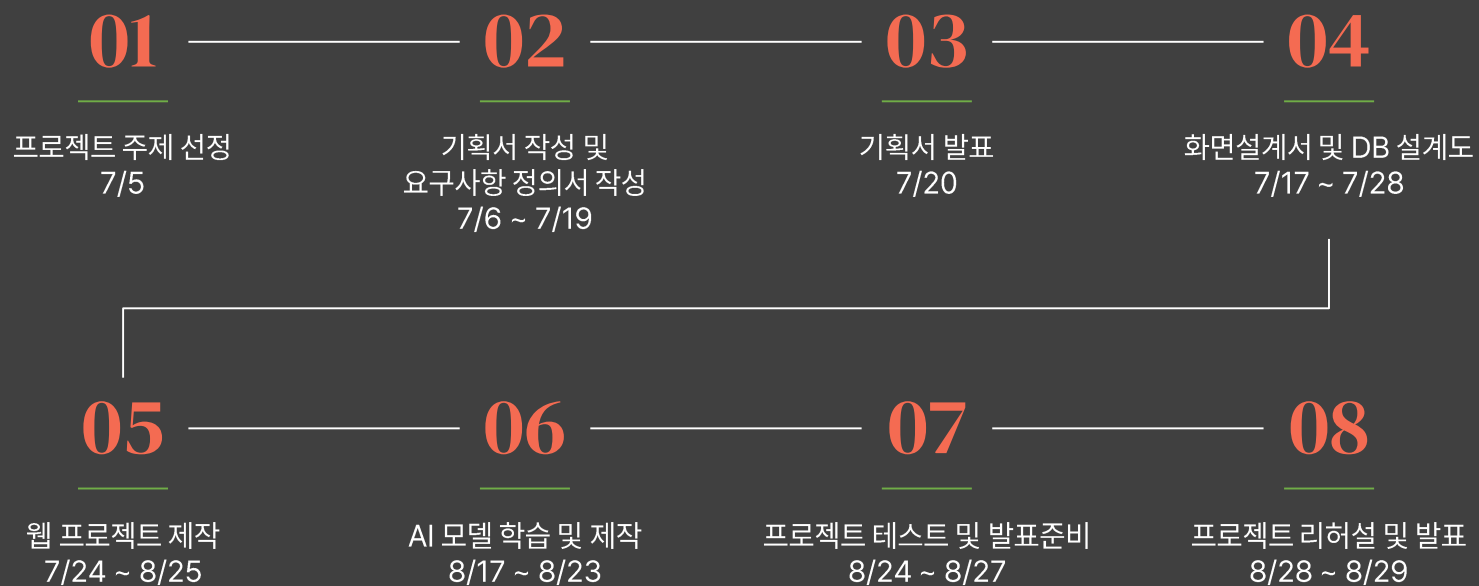




Part 3

# 수행 방법 및 절차







Notion



Excel



Notion

## Part 3

# ERD - 다이어그램

회원 MEMBER

아이디	id	Domain	VARCHAR2(30)	NOT NULL	Default value	Comment
회원번호	no	Domain	NUMBER(10)	NOT NULL	member_seq.NEXTVAL	Comment
비밀번호	password	Domain	VARCHAR2(50)	NOT NULL	Default value	Comment
이메일	email	Domain	VARCHAR2(320)	NOT NULL	Default value	Comment
등급	grade	Domain	NUMBER(1)	NOT NULL	1	관리자는 0
가입일	reg_date	Domain	DATE	NOT NULL	SYSDATE	Comment

문의게시판 QUESTION

게시글번호	no	Domain	NUMBER(10)	NOT NULL	question_seq.NEXTVAL	Comment
게시글분류	category	Domain	NUMBER(2)	NOT NULL	30	공지(10), 답변완료(20), 답변대기(30)
게시글조회수	views	Domain	NUMBER(5)	NOT NULL	0	Comment
게시글작성자	id	Domain	VARCHAR2(30)	NOT NULL	Default value	Comment
게시글제목	title	Domain	VARCHAR2(45)	NOT NULL	Default value	Comment
게시글내용	content	Domain	CLOB	NOT NULL	Default value	Comment
게시글작성일	create_date	Domain	DATE	NULL	SYSDATE	Comment
게시글수정일	update_date	Domain	DATE	NULL	SYSDATE	Comment
원본파일명	original_filename	Domain	VARCHAR2(255)	NULL	Default value	Comment
저장파일명	store_filename	Domain	VARCHAR2(255)	NULL	Default value	Comment
이미지URL	image_url	Domain	VARCHAR2(255)	NULL	Default value	S3 저장 링크

답변 ANSWER

게시글번호	no	Domain	NUMBER(10)	NOT NULL	Default value	Comment
답변작성자	id	Domain	VARCHAR2(20)	NULL	Default value	관리자
답변내용	content	Domain	CLOB	NULL	Default value	Comment
답변작성일	create_date	Domain	DATE	NULL	Default value	Comment
답변수정일	update_date	Domain	DATE	NULL	Default value	Comment

업로드이미지 UPLOAD\_IMAGE

이미지ID	idx	Domain	NUMBER(28)	NOT NULL	upload_seq.NEXTVAL	Comment
사용자ID	id	Domain	VARCHAR2(30)	NOT NULL	Default value	Comment
구분	category	Domain	NUMBER(2)	NOT NULL	10	기본(10), 좋아요(20), 싫어요(30), 문의사항(40)
등록일	upload_date	Domain	DATE	NOT NULL	SYSDATE	Comment
파일명	name	Domain	VARCHAR2(200)	NOT NULL	Default value	YYMMDDHH24MISS_원본파일명
URL	url	Domain	VARCHAR2(500)	NOT NULL	Default value	S3 저장 링크
파일크기	file_size	Domain	NUMBER(8, 2)	NOT NULL	Default value	Comment
검토여부	checked	Domain	NUMBER(1)	NOT NULL	0	검토 완료되면 1
다른 모양	u1	Domain	NUMBER(1)	NULL	0	체크하면 1
다른 색깔	u2	Domain	NUMBER(1)	NULL	0	체크하면 1
다른 그림 혹은 숫자	u3	Domain	NUMBER(1)	NULL	0	체크하면 1

결과이미지 RESULT\_IMAGE

표지판번호	no	Domain	NUMBER(3)	NOT NULL	Default value	Comment
표지판이름	name	Domain	VARCHAR2(100)	NOT NULL	Default value	Comment
표지판내용	content	Domain	VARCHAR2(500)	NOT NULL	Default value	Comment
URL	url	Domain	VARCHAR2(500)	NOT NULL	Default value	S3 저장 링크



## GTSRB (German Traffic Sign Recognition Benchmark)

- 자율주행 차량이나 운전자 보조 시스템에서 중요하게 사용되는 교통 표지판 인식을 위한 데이터
- 데이터 셋은 독일의 공공 도로에 있는 여러 교통 표지판 이미지
- 다양한 각도, 크기 및 형태로 촬영
- 약 39,000개의 훈련 이미지 & 약 12,600개의 테스트 이미지
- 총 43개의 클래스로 분류 (43개 종류의 교통 표지판)



## FRONT



## BACK



## DB &amp; Coop.



## AI &amp; Server



Part 4

# 프로젝트 수행 결과



**Mapper**

Member : 회원과 관련된 Oracle DB SQL Query

Admin : 관리자 기능과 관련된 Oracle DB SQL Query

**DAO / Service**

Member : 회원 CRUD, 아이디 및 비밀번호 찾기

Admin : 일반 회원 / 관리자 / 정지 회원 페이징 출력, 회원 관리 기능(계정 정지 / 삭제)

Mail : 인증번호 전송, 아이디 찾기 성공 시 메일로 아이디 전송

**Controller**

Member : 회원정보를 세션으로 화면 연결

Admin : 유저 정보를 리스트 형태로 화면 연결

**Views**

Member : 로그인, 회원가입, ID 및 PW 찾기, 비밀번호 재설정, 회원정보, 탈퇴 페이지

Admin : 관리자 페이지 (일반회원 / 관리자 / 정지회원 리스트)

## MailSender

MailSenderKey : 랜덤함수를 활용하여 랜덤숫자 6자리 생성

MailSenderService : Controller에서 요청을 받으면 인증번호 및 아이디를 전송하는 메일 작성

---

## BCrypt

UserDaoImpl : 회원가입시 입력한 비밀번호를 암호화하여 DB에 저장하고, 비밀번호를 요청할 때 마다 입력한 값을 암호화하여 저장된 비밀번호와 매칭하여 값이 올바른지 확인

---

## iframe

Admin.jsp : 일반회원 / 관리자 / 정지회원 페이징 페이지를 따로 만든 뒤, Admin.jsp 페이지 한 공간에 모두 띄워 한 눈에 리스트를 확인

---

## fullpage

main.jsp : 프로젝트와 연관성 있는 영상을 메인화면으로 활용하여 사용자가 어떤 서비스인지 쉽게 확인



Mapper / DAO



Service



Controller



Views

Question : 게시글 CRUD, 조회수 증가, 카테고리 변경, 페이징/검색, 공지 및 본인 글 조회

Answer : 답변(댓글) CRUD

---

Question : 글 등록/수정 시 첨부파일 업로드/수정/삭제 (S3) 관련 로직 추가

Answer : 답변(댓글) CRUD

---

Question : 게시글 화면 연결 및 사용자 session 처리, 서버 측 유효성 검사, 첨부파일 업로드

Answer : 답변(댓글) CRUD, 사용자 session 처리

---

Question : 게시글 작성/수정/상세 조회/목록 조회 화면 연결, 클라이언트 측 유효성 검사

My-question: myPage 화면에서 사용자 본인이 작성한 글 조회 페이지 연결

## Custom Annotation

- @ImageFile,  
@NotBlankWithoutHtml 정의
- 각각의 Validator를 통한 구현
- Question save/update 시  
DTO, Controller에서 유효성 검사

## CKEditor

- CKEditor4 Basic ver. 사용
- toolbar 설정을 통해 원하는 기능만  
포함하여 사용 가능

## Interceptor

- Login, Admin Interceptor 작성
- 로그인하지 않은 사용자의 게시판  
접근 방지
- 관리자만 답변 기능 접근 허용

Mapper / DAO



Service



Controller



Views

Upload : 이미지 업로드 관련 CRUD, 목록 조회(전체/카테고리별), 피드백 그래프(누적/월별)

Result : 결과 이미지 상세조회

Upload : 아마존 S3에 이미지 업로드, 삭제, 버킷이동 로직 추가

Result : 결과 이미지 상세조회

RestRemplate : Flask API로 POST 요청

Upload : 이미지 업로드 처리, Flask 통신, 결과 이미지 출력, 피드백 업데이트 처리

ImgManage : 업로드된 이미지 관리(CRUD), 피드백 그래프 출력

preUpload : 이미지 업로드 기능, 업로드 후 upload.jsp로 이동

upload : 이미지 재업로드 기능, 결과 이미지 출력, 피드백 기능(좋아요/싫어요)

imgManage : 업로드된 이미지 카테고리별 목록 조회, 상세 조회, 삭제, 저장(S3버킷이동)

graph : '싫어요' 피드백 관련 누적 그래프(막대)와 월별 그래프(꺾은선)

## MultipartFile

- 이미지 업로드를 위해 MultipartFile 인터페이스 활용
- 'jpg', 'jpeg', 'png', 'bmp', 'tiff', 'webp', 'ico', 'svg'로 확장자 제한
- 파일 크기 최대 5MB로 제한

## RestTemplate

- TensorFlow 모델을 통해 예측을 수행하고 결과를 반환하는 Flask API 구축
- Flask API와 통신하기 위해 RestTemplate 활용
- 이미지URL을 매개변수로 Flask API의 엔드포인트에 HTTP 요청, JSON 형식으로 응답을 받음
- Integer로 변환 후 해당하는 결과이미지 출력

## Chart.js

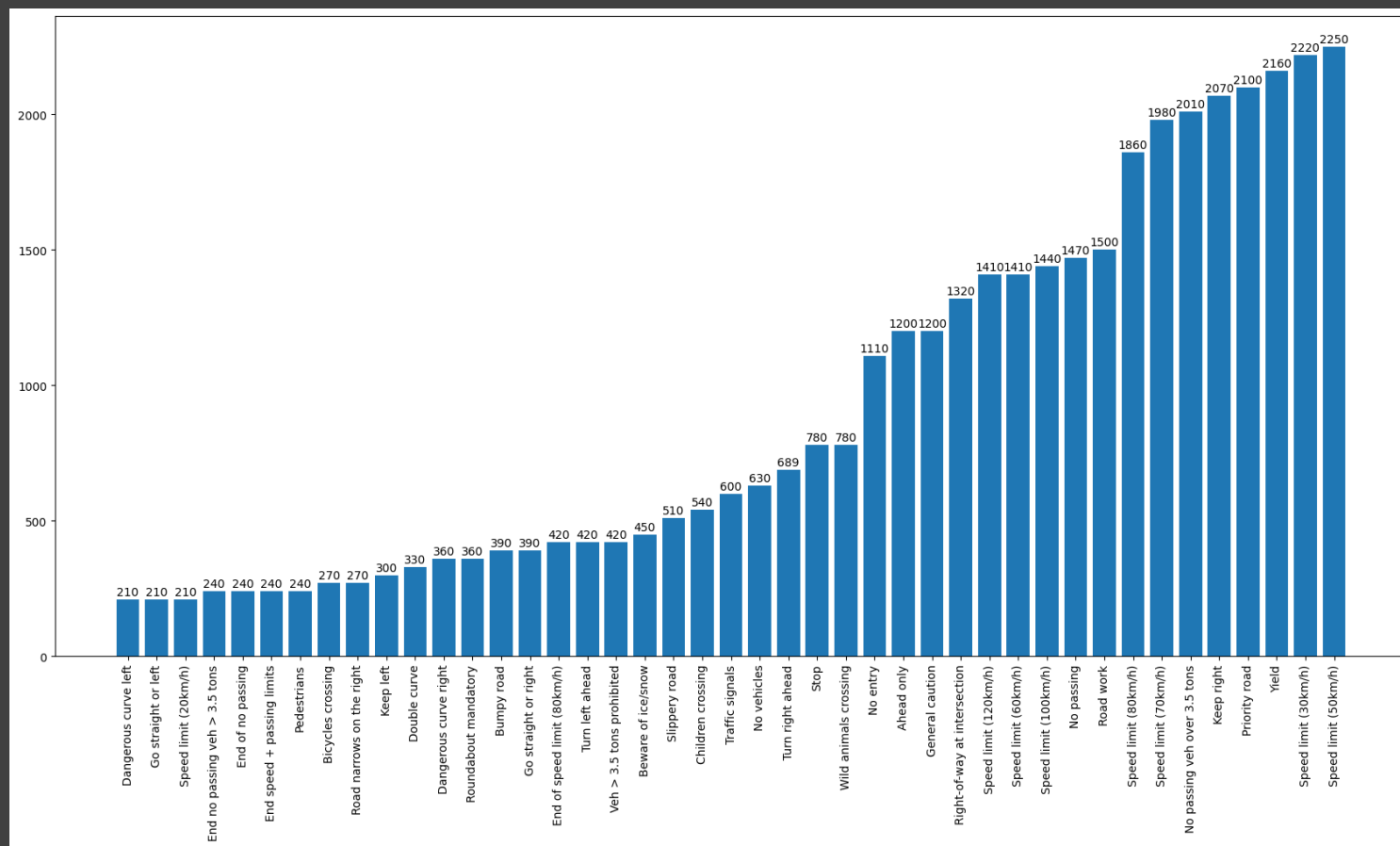
- Chart.js 3.5.1 ver. 사용
- 막대그래프, 꺾은선그래프를 그려 데이터 시각화
- Ajax를 활용하여 데이터베이스와 연동

## Amazon S3 / EC2 / RDS

- S3, EC2, RDS 계정 생성하고 인스턴스끼리 접근 가능하도록 보안 설정
- S3 버킷 2개로 나누어 필요에 따라 관리 가능하도록 함
- EC2 인스턴스에 매크로 만들어 커맨드에 따라 풀&푸시, 톰캣 서버 시작/종료, 로그, SQL관리 자동화
- 스왑 메모리 사용으로 RAM이 1기까지만 톰캣에서 WAR 실행 시 30000ms 미만이 걸리도록 함

## 지도 학습 (supervised learning)

입력 데이터와 그에 상응하는 레이블(label)을 함께 사용하여 모델을 학습하는 방식



\*43개 레이블의 훈련 데이터 : 39209개

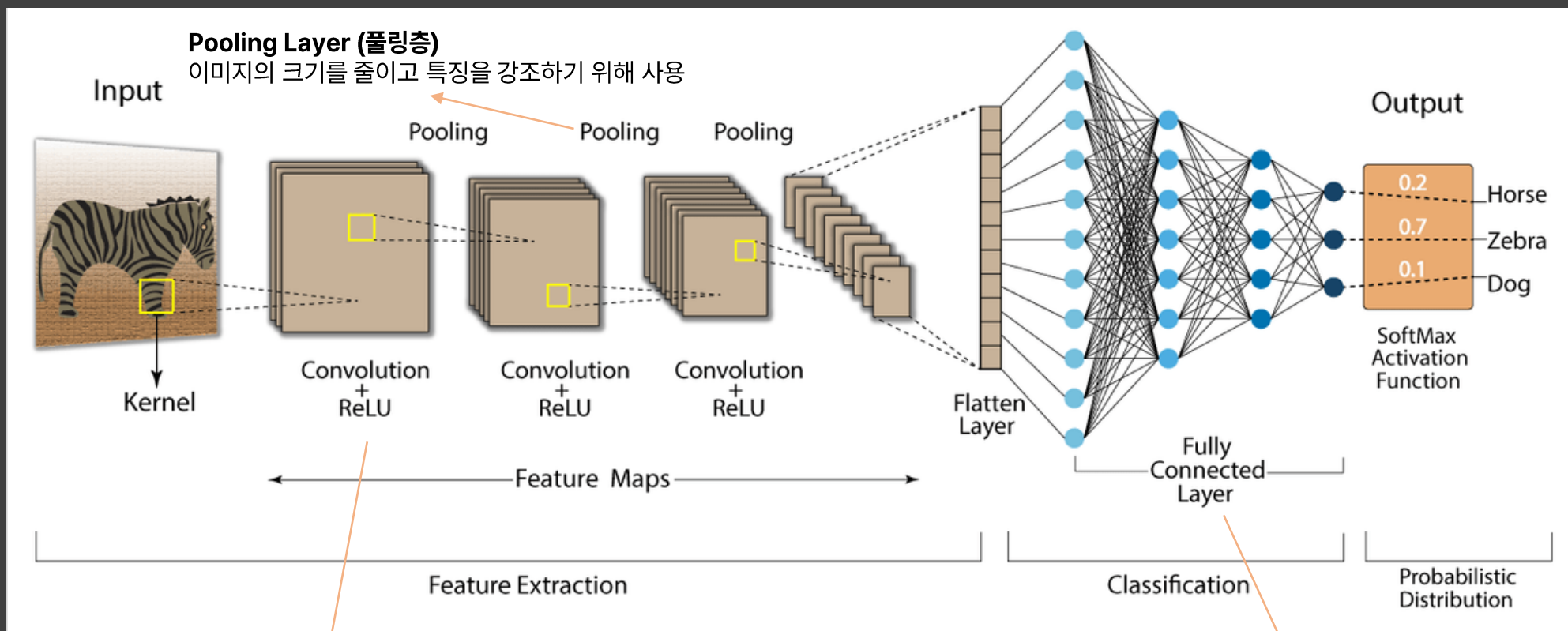


## 학습 데이터 시각화



## 컨볼루션 신경망(Convolutional Neural Network, CNN) 모델

부분의 특징을 추출하여 결합한 뒤 예측하는 방식 (실제 인간의 시각 체계와 비슷함, 이미지 인식에 탁월)



### Convolutional Layer (합성곱층)

입력 이미지에 여러 개의 필터(커널)를 적용하여 추출  
각 필터는 이미지의 특정 패턴이나 특징을 감지하는 파라미터

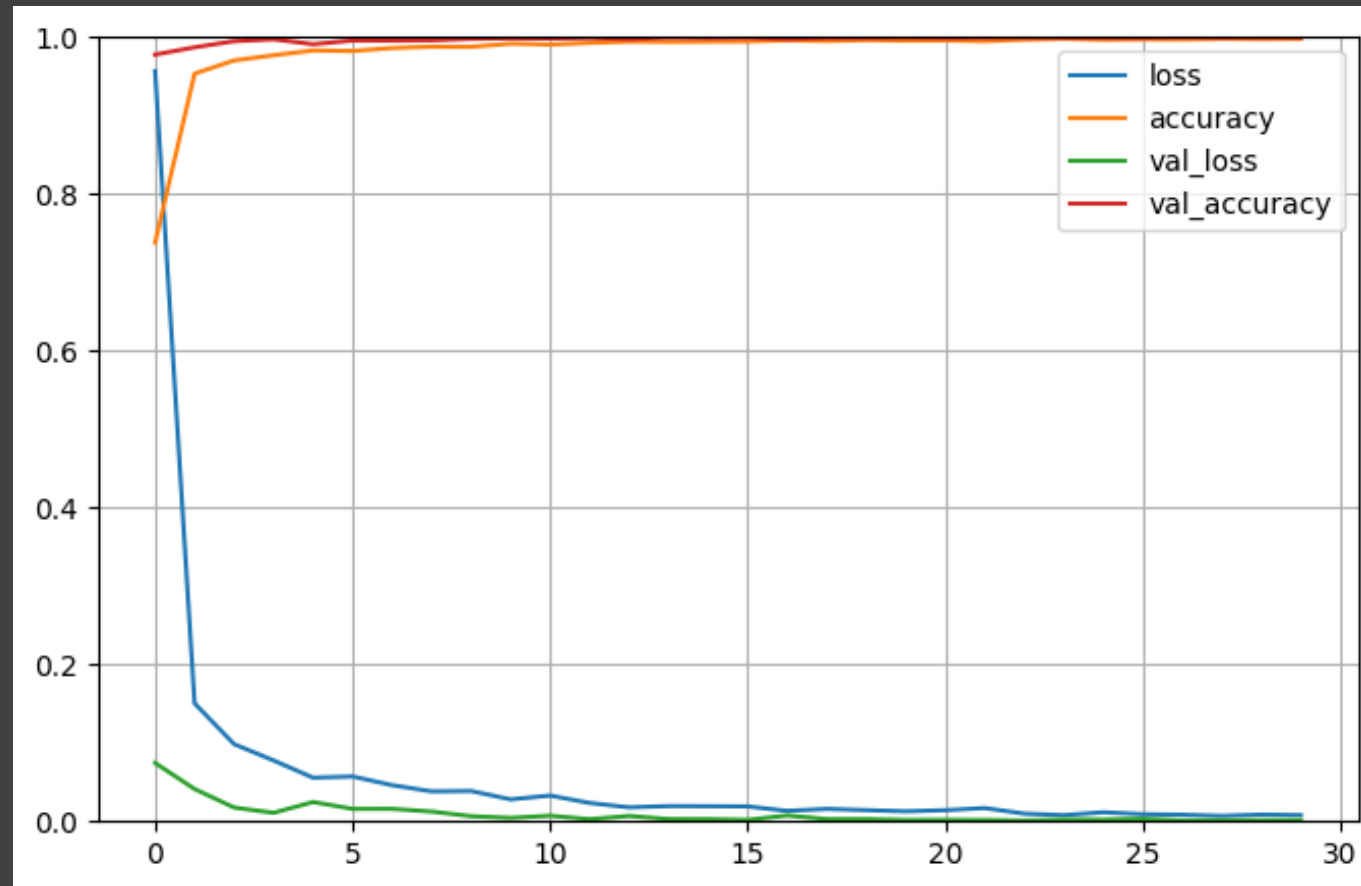
### Fully Connected Layer (완전 연결층)

추출된 특징을 기반으로 분류 및 예측을 수행

## TensorFlow 설계한 딥러닝 신경망 모델

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 29, 29, 16)	448
conv2d_1 (Conv2D)	(None, 27, 27, 32)	4640
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)	0
batch_normalization (BatchNormalization)	(None, 13, 13, 32)	128
conv2d_2 (Conv2D)	(None, 11, 11, 64)	18496
conv2d_3 (Conv2D)	(None, 9, 9, 128)	73856
max_pooling2d_1 (MaxPooling2D)	(None, 4, 4, 128)	0
batch_normalization_1 (BatchNormalization)	(None, 4, 4, 128)	512
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 512)	1049088
batch_normalization_2 (BatchNormalization)	(None, 512)	2048
dropout (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 43)	22059

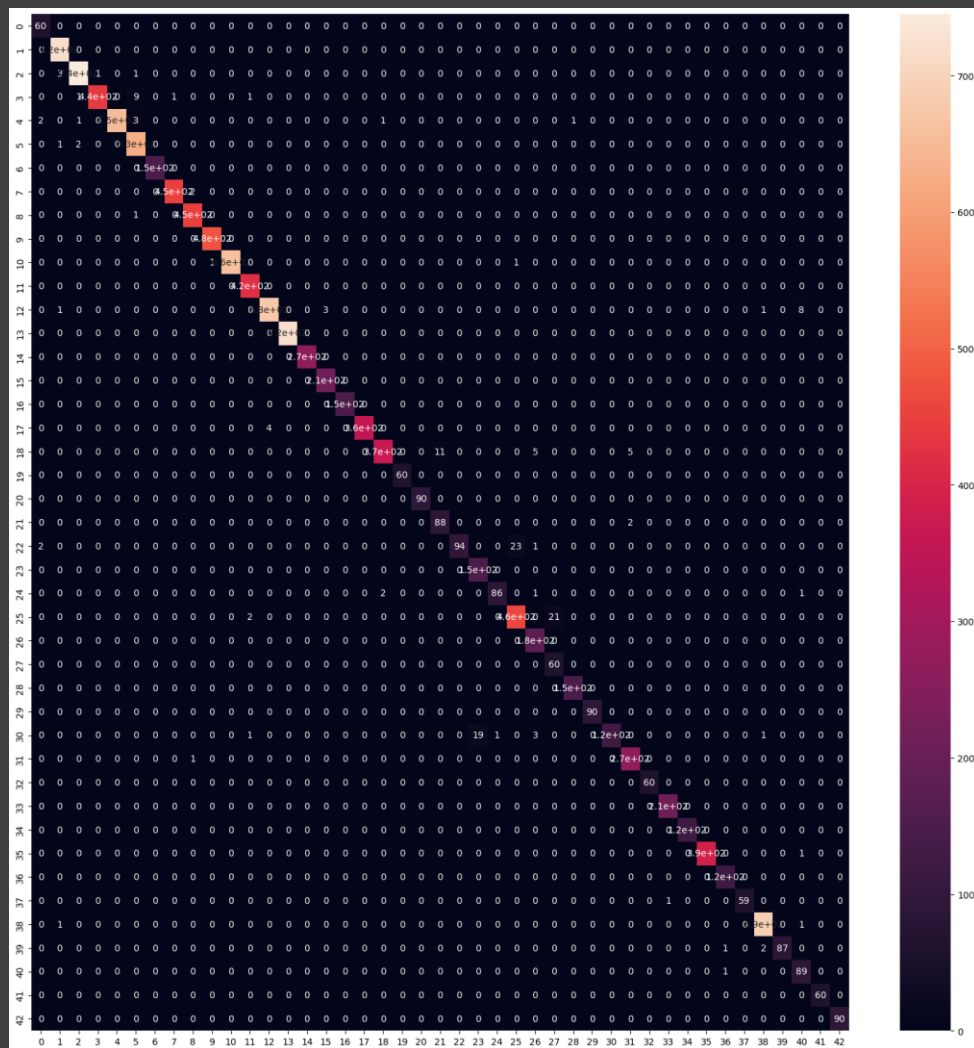
Matplotlib 모델 정확도 그래프화



훈련 데이터 정확도	99.78%
------------	--------

검증 데이터 정확도	99.95%
------------	--------

## Seaborn 모델 정확도 그래프화



테스트 데이터 정확도 98.73%

\* 대각선 값 : 정확한 예측의 수



## 테스트 데이터 중 랜덤 25개 이미지 예측 결과



Part 5

# 자체 평가 의견



**김정민**

머신러닝에 대해 깊이 있게 배우지 못한 상태에서 프로젝트에 적용하는 데에 어려움이 있었지만, 직접 모델을 학습시키고 웹에 연동시키는 경험이 의미 있었고, 암호화, S3, EC2 등 다양한 기능을 프로젝트에 접목시킬 수 있어서 뜻깊었다. 조원들이 모두 자기 역할을 충실히 하고 열심히 참여해줘서 너무 고맙다는 말을 꼭 하고싶다!

**송세림**

모두 열심히 참여하는 분위기였다. 경험이 많은 조원들이 조원들에게 도움을 많이 주었고 자신의 파트가 아닌 것에도 함께 해결하며 좋은 성과가 나왔다. 처음엔 막막했지만 어느새 프로젝트가 잘 마무리 되는 것을 보고 성취감이 생겼다. 좋은 분위기 속에서 프로젝트를 할 수 있는 값진 경험이 되었다. 다들 소중한 인재가 될 것 같다.

**오동훈**

이번 프로젝트를 통해 암호화와 같이 수업 외의 부분들을 배우는 등 백엔드에 대해 좀 더 깊이 공부할 수 있는 기회가 되었습니다. 또한 혼자서는 어려웠던 부분들은 팀원들과 함께 해결하며 팀워크의 중요성을 다시 한번 느끼는 시간이었습니다.

**이주영**

처음 프로젝트를 시작할 때 스스로 부족한 부분뿐이어서 잘 해낼 수 있을지 걱정했는데, 좋은 팀원들을 만나서 끝까지 완주 가능했다. 각자의 자리에서 열심히 해준 팀원들에게 감사하고, 덕분에 많이 배우고 성장할 수 있어 의미 있는 시간이었다.

### 장태근

우여곡절 많았지만 어떻게든 끝이 나서 다행이다.

### 차지욱

조원들과 소통이 잘 되고, 업무분담이 효율적으로 이루어져서 프로젝트를 끝낼 수 있었다고 생각한다. 프로젝트를 수업 이외의 부분을 스스로 공부해서 진행하는 힘든 과정이었지만 모두가 끝까지 포기하지 않고 참여해주어서 고맙다는 말을 전하고, 짧은 기간이었지만 다른 곳에서도 부디 잘 지내고 다음에 만날 때는 더 성장한 모습으로 만나길 바란다

### 차현경

문제를 해결해가며 내 마음껏 만들어가는 게 재밌었다. 클라이언트로부터 DB까지 복잡한 요청/반환을 다루며 흐름을 이해하게 되었고, 알고리즘에서 순서가 얼마나 중요한지 알게 되었다. 처음이지만 무턱대고 시작한 AWS 툴 연결 과정에서 며칠 끙끙대며 앓았던 문제를 함께 고민하고 해결해준 팀원 모두에게 고생 많았고 감사하다고 말하고 싶다.

### 최유정

각자 역할 배분에 충실하게 맡아서 작업했다. 시행착오가 있어도 서로 소통하며 원활히 해결해가는 등 전체적으로 분위기나 참여도가 좋았다.



A nighttime photograph of a city street. The street is wet, reflecting the lights from street lamps and buildings. In the background, there are several tall buildings, some with lit windows. The street is illuminated by street lamps, and there are long, horizontal light trails from cars, suggesting a long exposure. The overall scene is dark, with the primary light sources being the artificial lights of the city.

**Thank you for watching!**