



3주차_발표

1


주제

숫자 손글씨 이미지에 대한 텍스트 분류

2 데이터

- 데이터셋 선정

5.1.3. MNIST Dataset 소개

 <https://sdc-james.gitbook.io/onebook/4.-and/5.1./5.1.3.-mnist-dataset>

 OneBook(Python & Deep Learning)

5.1.3. MNIST Dataset 소개

Powered By GitBook


⇒ 라벨(클래스) = 이미지가 나타내는 숫자가 어떤 숫자인지 나타냄
⇒ “다중 클래스 분류 문제”

- 데이터셋 가공

1. kaggle 아래 링크에서 숫자 손글씨 이미지를 다운받는다.







MNIST as .jpg

Kaggle Digit Recognizer Competition Dataset as .jpg Image Files











 <https://www.kaggle.com/datasets/scolianni/mnistasjpg>



2. 데이터셋은 아래와 같은 구조로 되어있다.

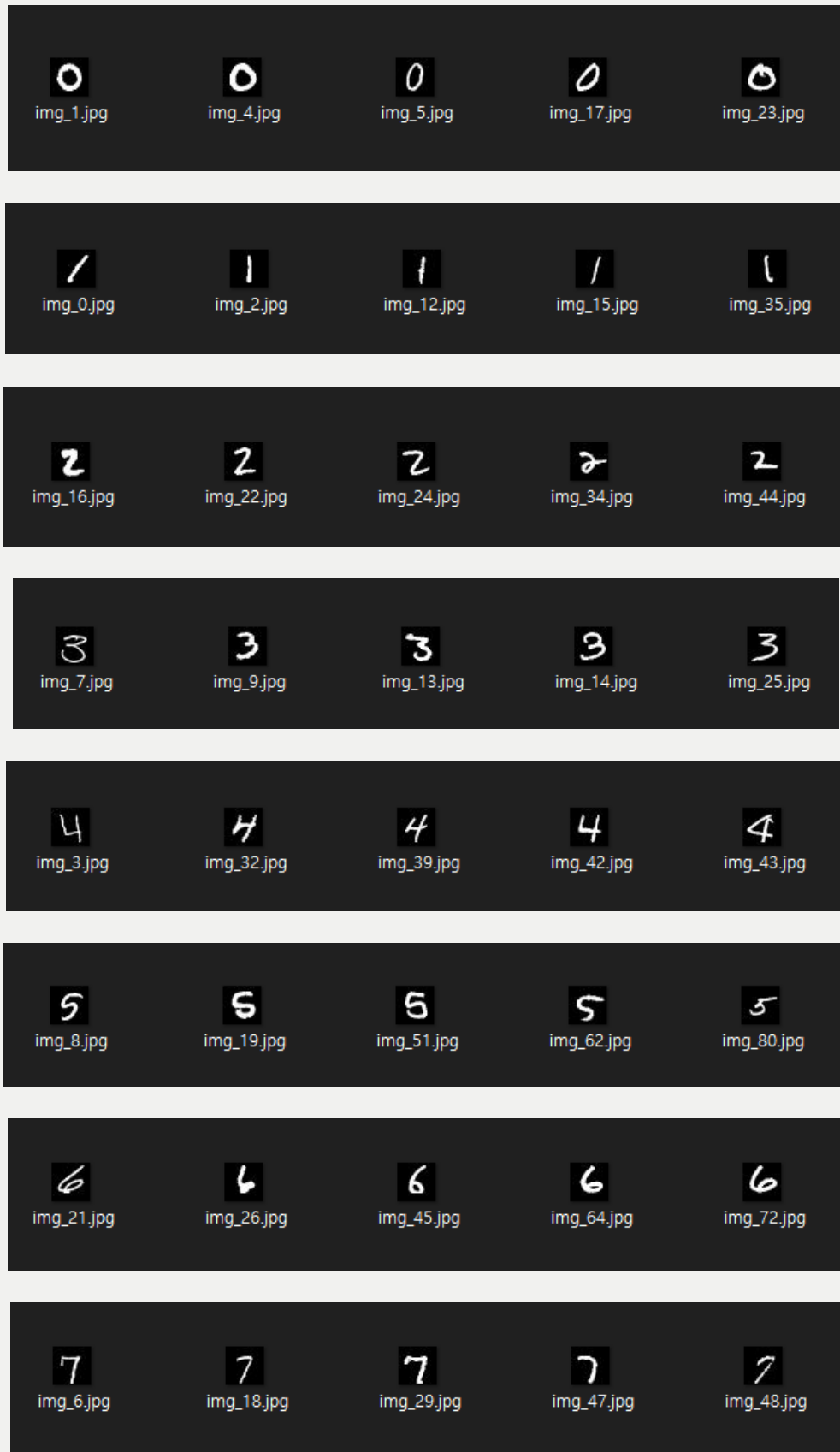
	testSample	2023-02-20 오전 10:38	파일 폴더	
	testSet	2023-02-20 오전 10:38	파일 폴더	
	trainingSample	2023-02-20 오전 10:38	파일 폴더	
	trainingSet	2023-02-20 오전 10:38	파일 폴더	
	testSet.tar.gz	2019-09-20 오후 3:53	압축(GZ) 파일	6,968KB
	trainingSet.tar.gz	2019-09-20 오후 3:54	압축(GZ) 파일	10,495KB

3. 학습 데이터셋은 trainingSet 폴더를 사용함 (0부터 9까지 클래스별로 분류되어있음)

	0	2023-02-20 오전 10:38	파일 폴더
	1	2023-02-20 오전 10:38	파일 폴더
	2	2023-02-20 오전 10:38	파일 폴더
	3	2023-02-20 오전 10:38	파일 폴더
	4	2023-02-20 오전 10:38	파일 폴더
	5	2023-02-20 오전 10:38	파일 폴더
	6	2023-02-20 오전 10:38	파일 폴더
	7	2023-02-20 오전 10:38	파일 폴더
	8	2023-02-20 오전 10:38	파일 폴더
	9	2023-02-20 오전 10:39	파일 폴더

4. 학습을 위한 각 클래스별 이미지 5개씩 선정

손글씨의 특성은 같은 숫자여도 각기 다른 형태를 가짐



8

img_10.jpg

8

img_20.jpg

8

img_30.jpg

8

img_67.jpg

8

img_82.jpg

9

img_11.jpg

9

img_27.jpg

9

img_28.jpg

9

img_31.jpg

9

img_33.jpg

3 학습

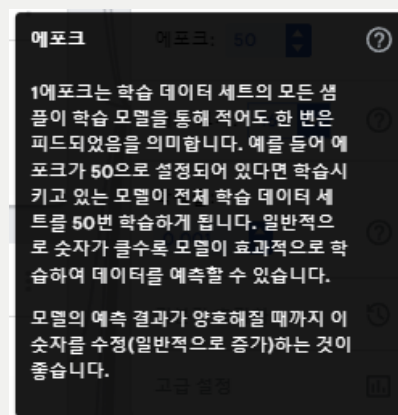
1. 클래스 설정 (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)



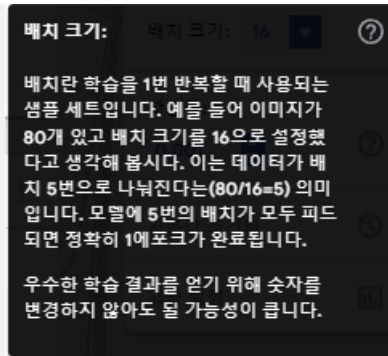
2. 학습 데이터로 선택한 이미지를 각 클래스별로 업로드



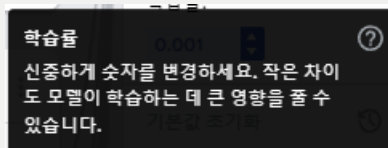
3. 학습하기 전에, “고급”에서 다양한 변수(에포크, 배치크기, 학습률)를 설정
 다양한 변수는 모델의 성능과 직결됨 ⇒ 클래스가 여러개여서, 에포크를 늘림



에포크 = 전체 학습 데이터를 몇 번 볼 것인가



배치크기 = 효율적인 학습을 위해 학습 데이터를 나눠줌 (통째로 넣으면 학습 시간이 오래걸림)



학습률 = 학습률이 클수록 학습속도는 빠르지만, 최소값을 차지 못하거나 값이 발산할 수 있다. 반대로 학습률이 작으면, 학습속도가 느리고 국소값에 머무를 수 있다.

학습

모델 학습시키기

고급

에포크: 100

배치 크기: 16

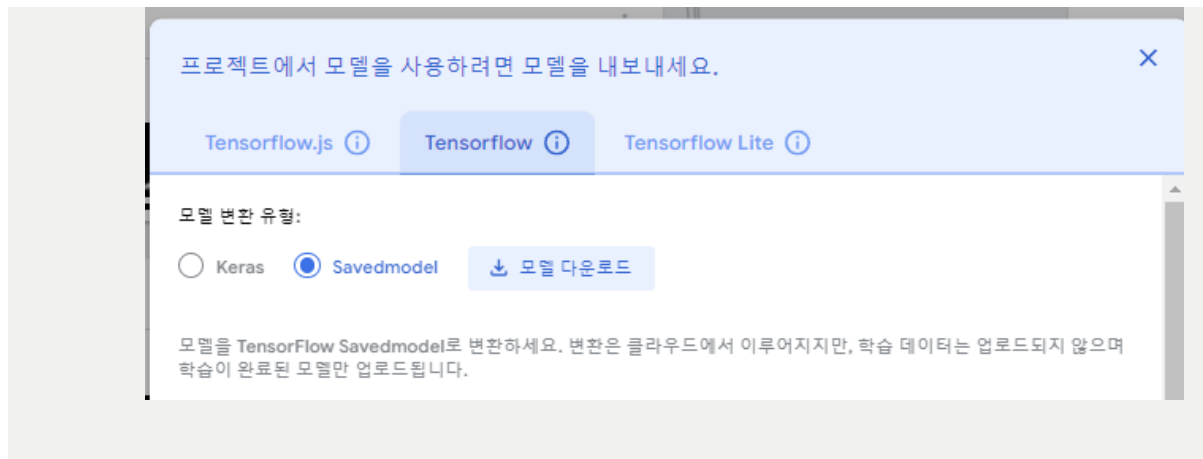
학습률: 0.001

기본값 초기화

고급 설정

4. 모델 내보내기

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/fa4d4183-94a7-42d7-b5d0-cc9ae40c6914/3%EC%A3%BC%EC%B0%A8_%EA%B3%BC%EC%A0%9C_%EA%B0%95%EC%A7%80%EC%9D%80.zip



4 모델 테스트

1. 테스트 데이터 선정

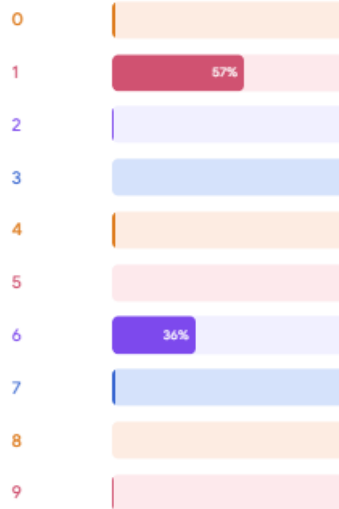


⇒ 결과



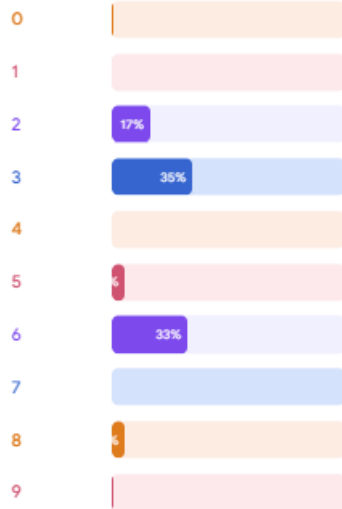


결과



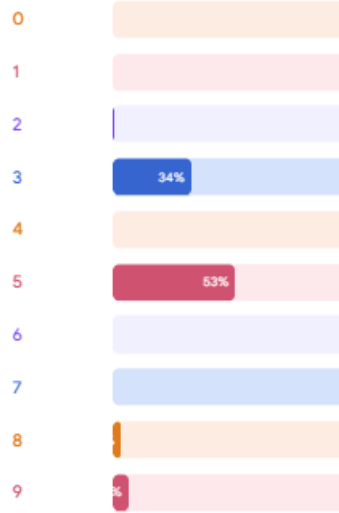


결과



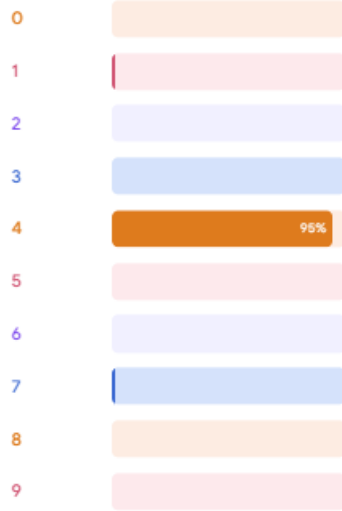


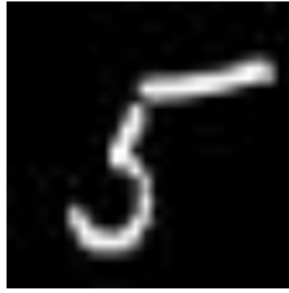
출력



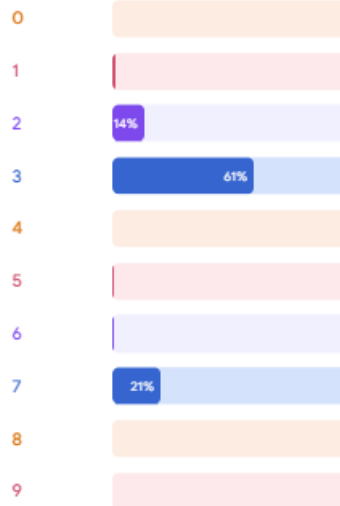


출력



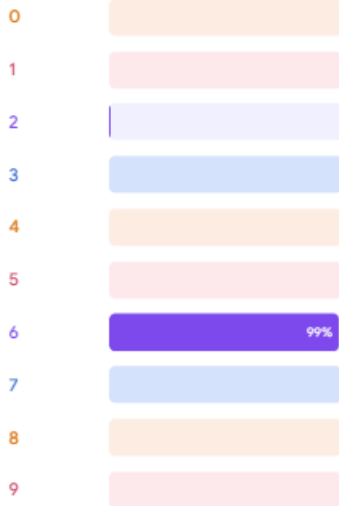


결과



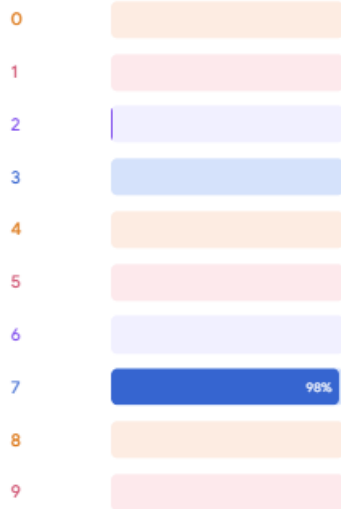


출력





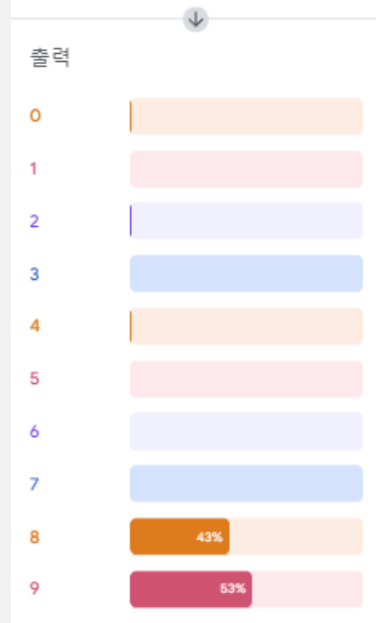
예제





예측

0	
1	
2	
3	
4	
5	
6	
7	
8	99%
9	



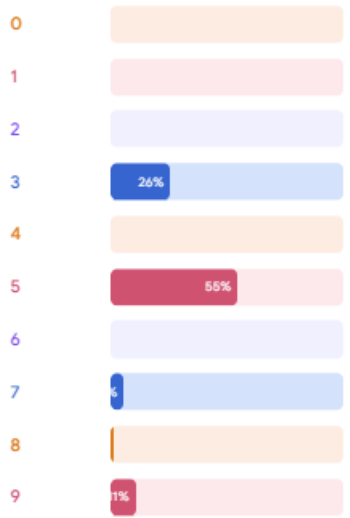
2. 흰색 바탕+검정글씨 이미지 테스트



⇒ 결과

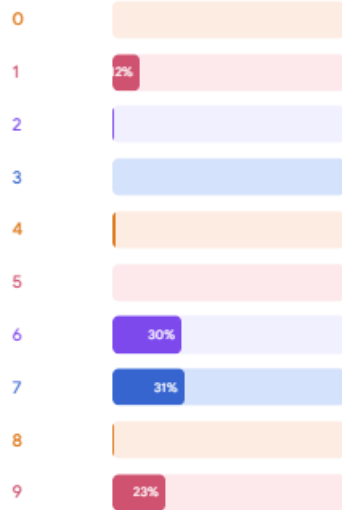


점수



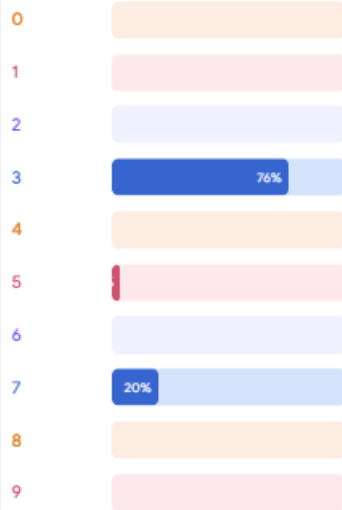


음표





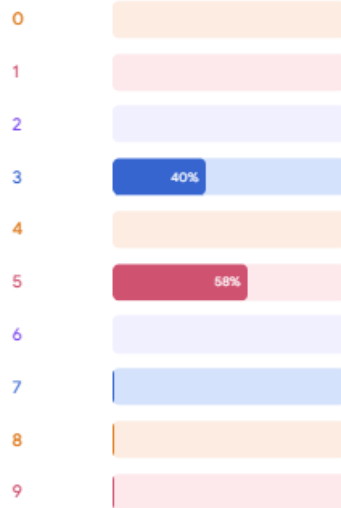
음표



3



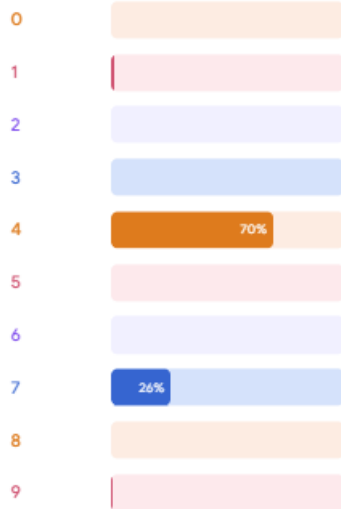
출
발



4



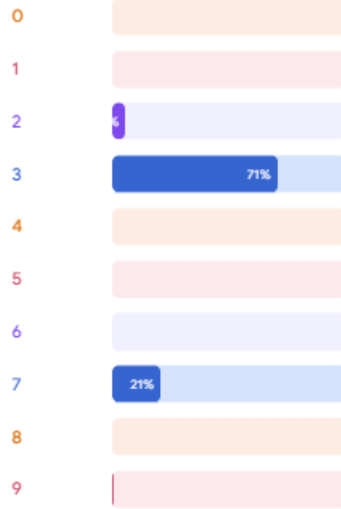
순위



5

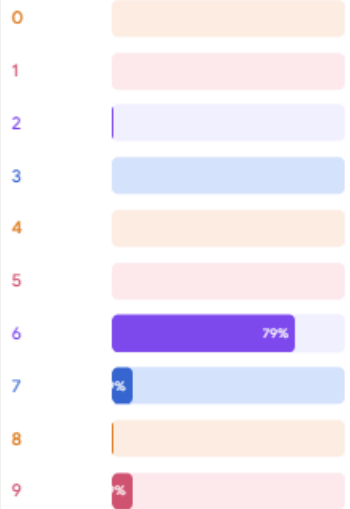


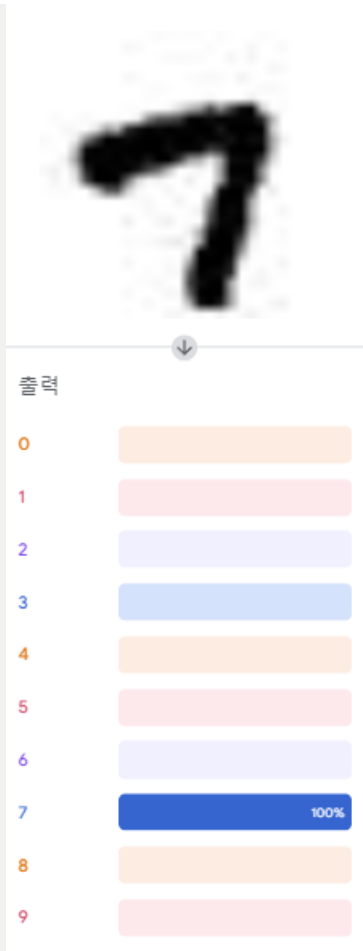
출력





출력

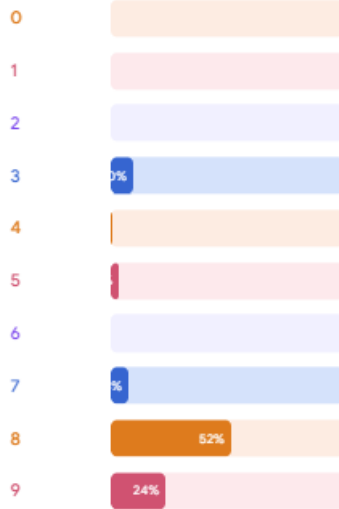


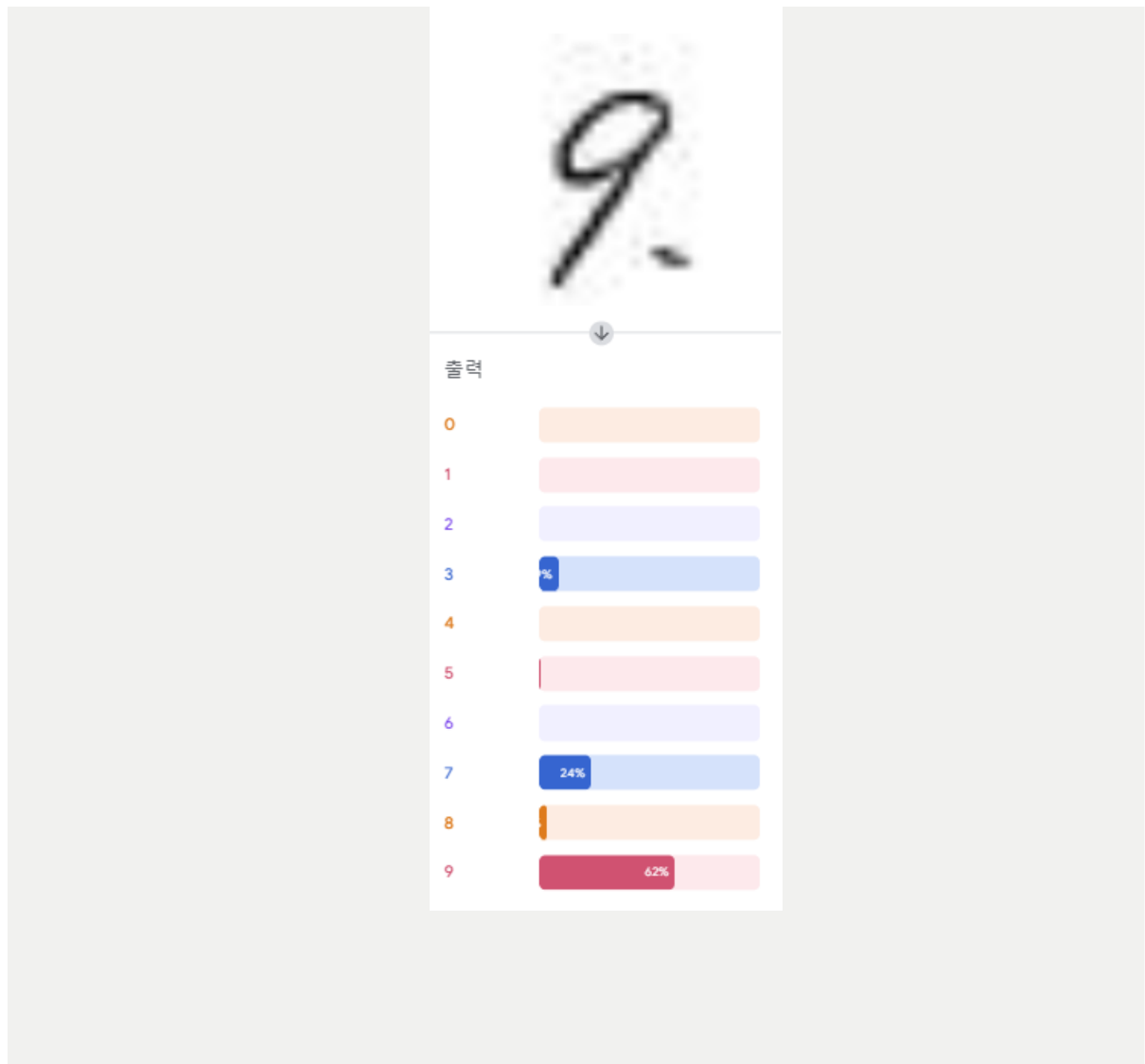


8



결과





! 결과 정리

- 학습에 관련된 파라미터 값에 대한 파악이 필요하며, 값에 대한 여러 번의 실험이 필요함
- 학습한 이미지와 다른 형태의 이미지를 테스트하면 정확도가 낮아지고, 혼돈하게 됨
- 일관된 형태보다 다양한 데이터를 준비하여 학습에 활용하는 것이 필요함