

## 목차

- CIFAR10: Hyper parameter의 영향 파악
  - 기존 세팅
  - Loss function 변경
    - `categorical_crossentropy`
  - Optimizer 변경
    - `adam`
    - `rmsprop`
  - Batch
    - Full batch - batch size == 50000
    - Batch size == 10000
    - Batch size == 1000
    - Batch size == 100
    - Batch size == 20
  - Epoch 변경
    - Epoch == 100

## CIFAR10: Hyper parameter의 영향 파악

---

### 기존 세팅

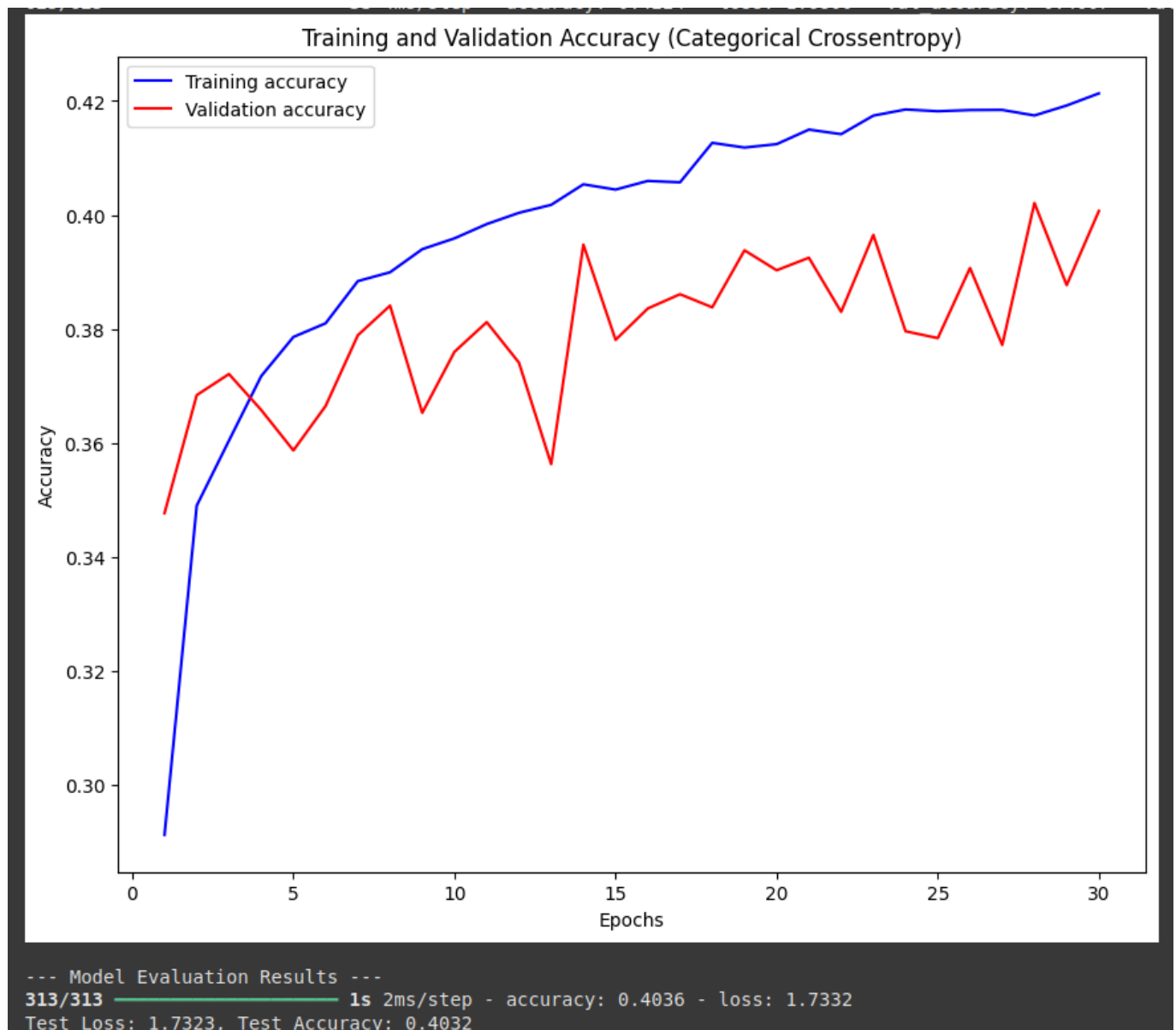
- activation function
  - `softmax`
- Loss function
  - `sparse_categorical_crossentropy`
  - 다중 클래스 분류에 적합한 손실함수
- Optimizer
  - `sgd`: Stochastic Gradient Descent
  - 일반적인 optimizer 사용
- 학습 횟수(epoch)
  - 30회
- batch size
  - 64
  - batch size 크기 별 장단점
    - Size가 크면
      - 경사가 안정적
      - 경사 정확도가 안정적
      - 학습시간 단축 --> 컴퓨터 자원 많이 사용
      - 경사가 완만해 지역 최적점 문제 발생
    - Size가 작으면
      - 경사가 가파름
      - 지역 최적점에 덜 빠짐
      - 전역 최적점을 잘 찾을 가능성이 올라감
      - 전체 데이터셋을 사용하는 것보다 연산 속도가 빠름

- 경사 정확도가 떨어짐
- 노이즈 민감
- Validation\_split
  - 0.2
  - 값이 크면
    - 훈련 데이터양이 줄어듦
    - 학습을 덜 하니 최종 성능이 줄어듦
  - 값이 작으면
    - 훈련 데이터 양이 늘어남
    - 최종 성능이 증가
    - 검증 데이터가 줄어 정확도의 변동성이 커질 수 있음



## Loss function 변경

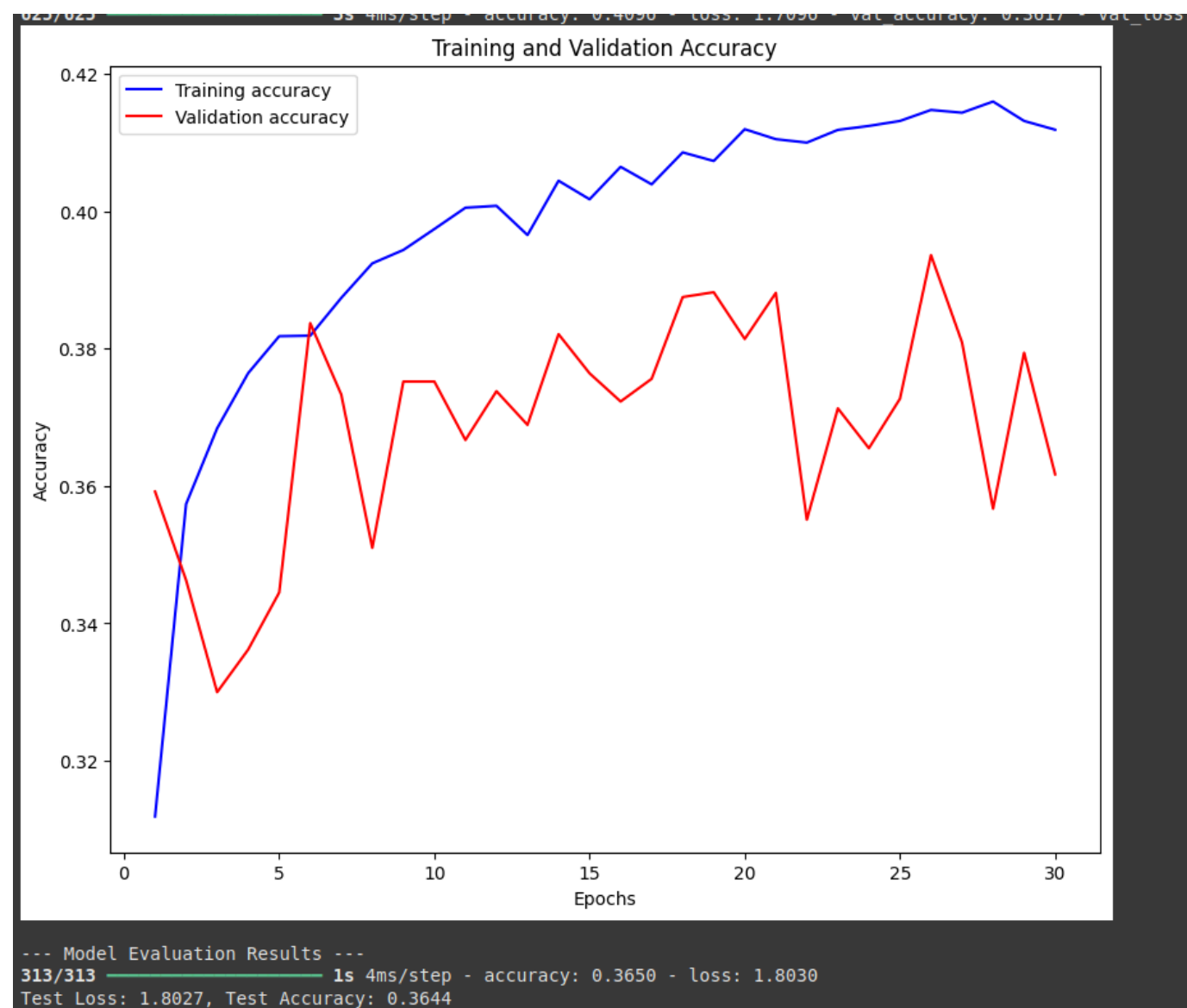
categorical\_crossentropy



- accuracy가 기존 0.389에서 0.4036으로 증가함
- 사실 기존 손실함수와 바꾼 함수는 큰 차이가 없어야함
- 랜덤 값이 영향을 준 듯함

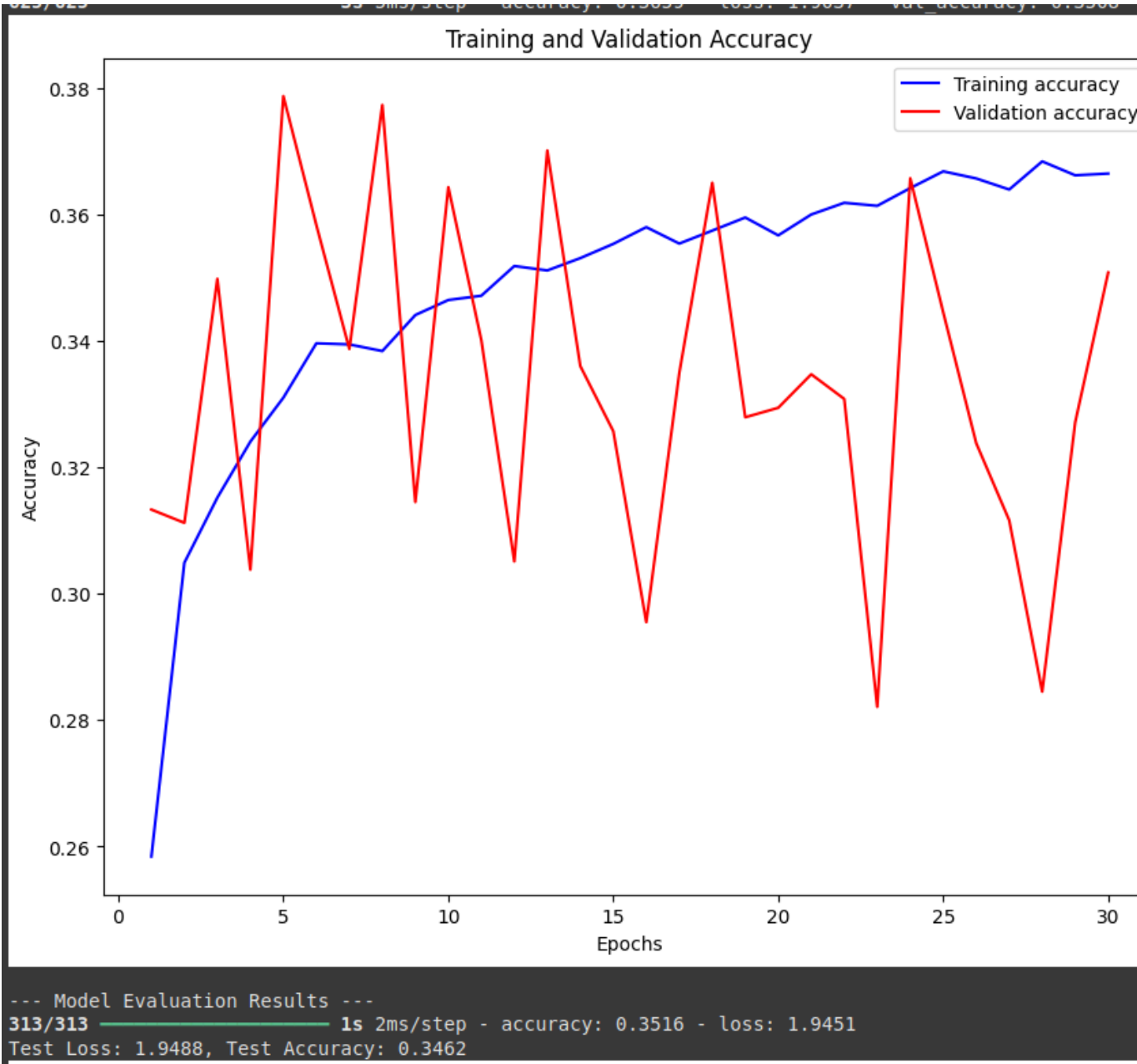
## Optimizer 변경

adam



- 정확도가 0.365로 낮아짐

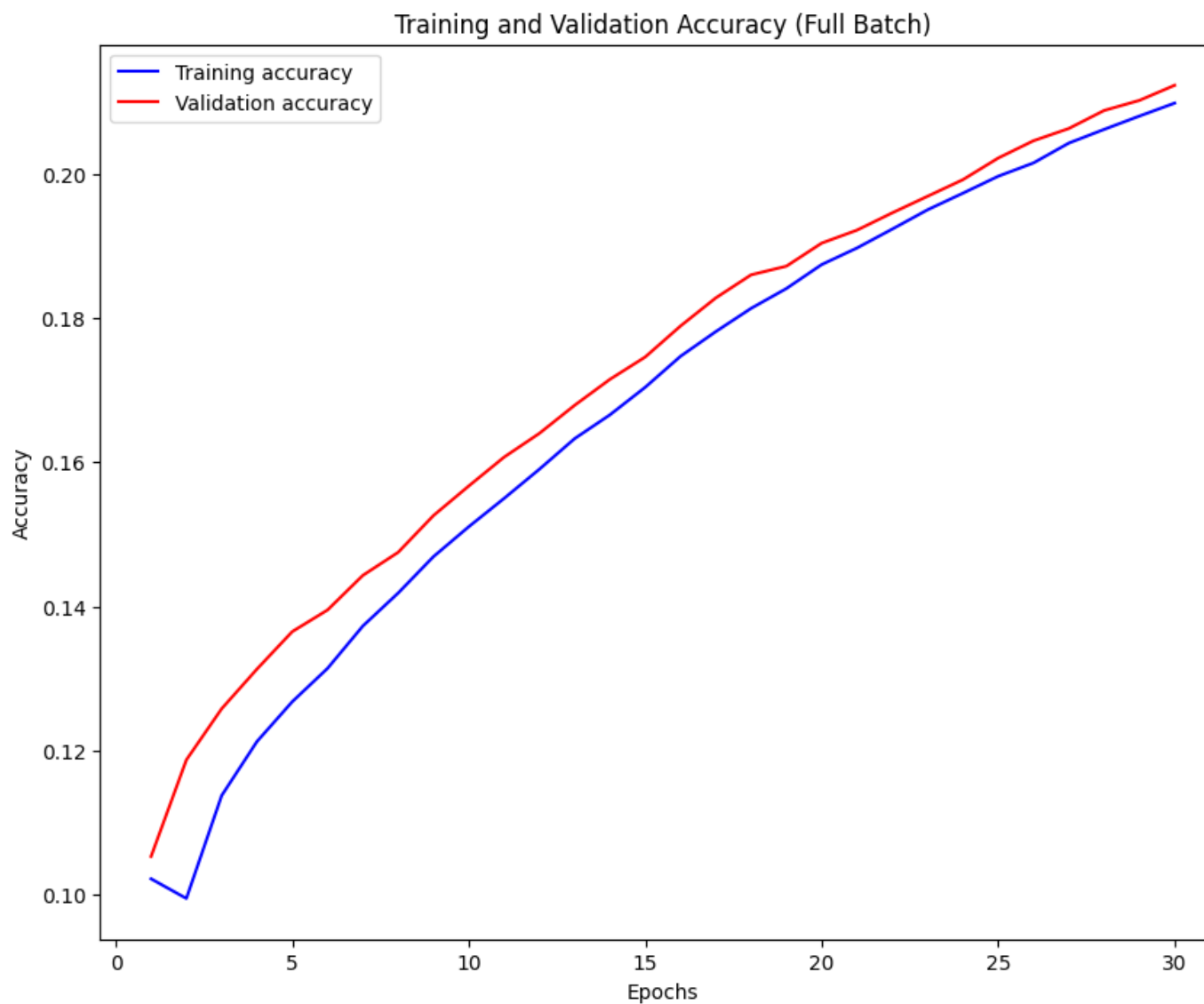
rmsprop



- 0.352정도로 낮아짐

Batch

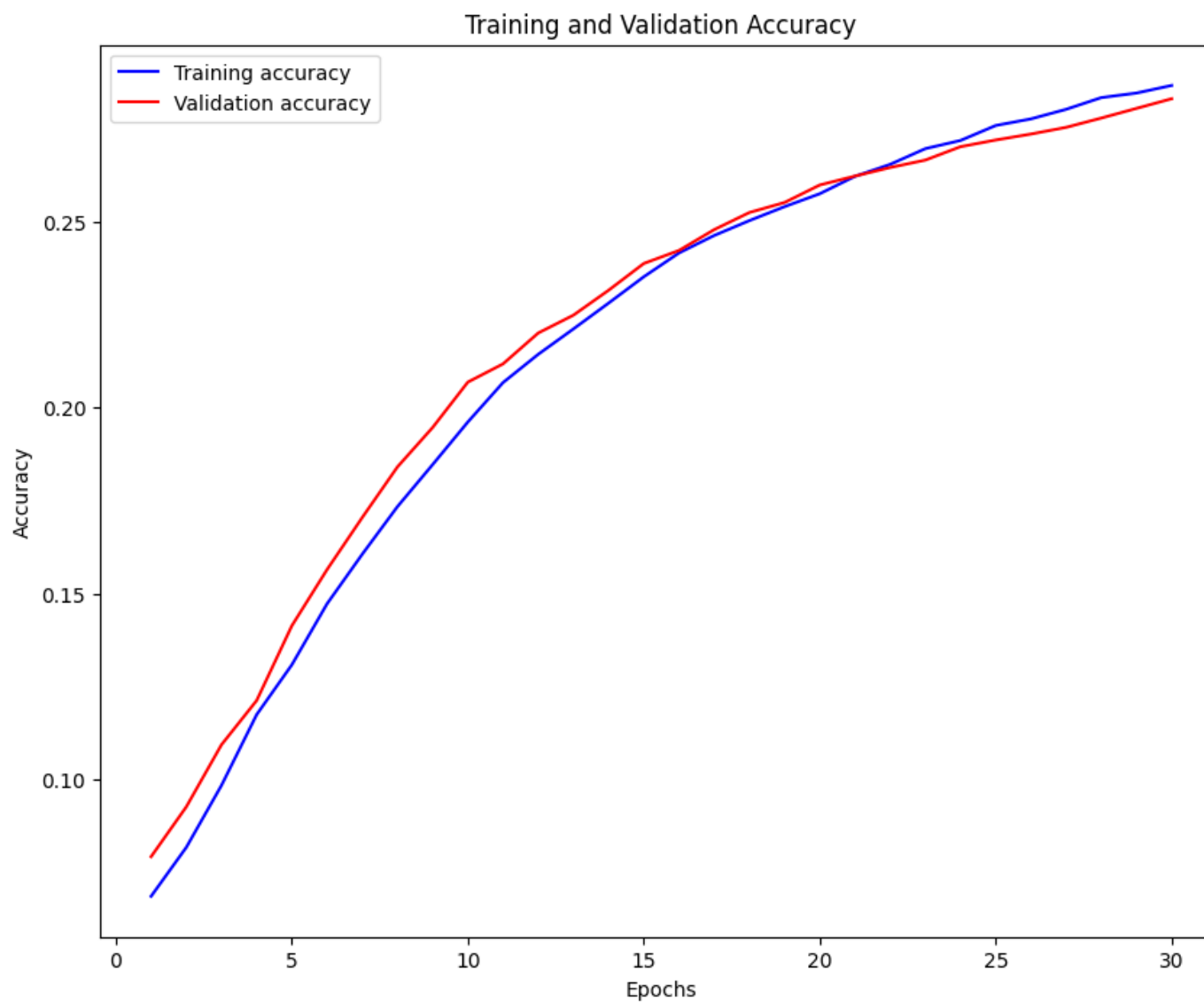
Full batch - batch size == 50000



```
--- Model Evaluation Results ---  
313/313 ----- 1s 2ms/step - accuracy: 0.2176 - loss: 2.1771  
Test Loss: 2.1770, Test Accuracy: 0.2155
```

- 정확도가 0.22 수준으로 떨어짐

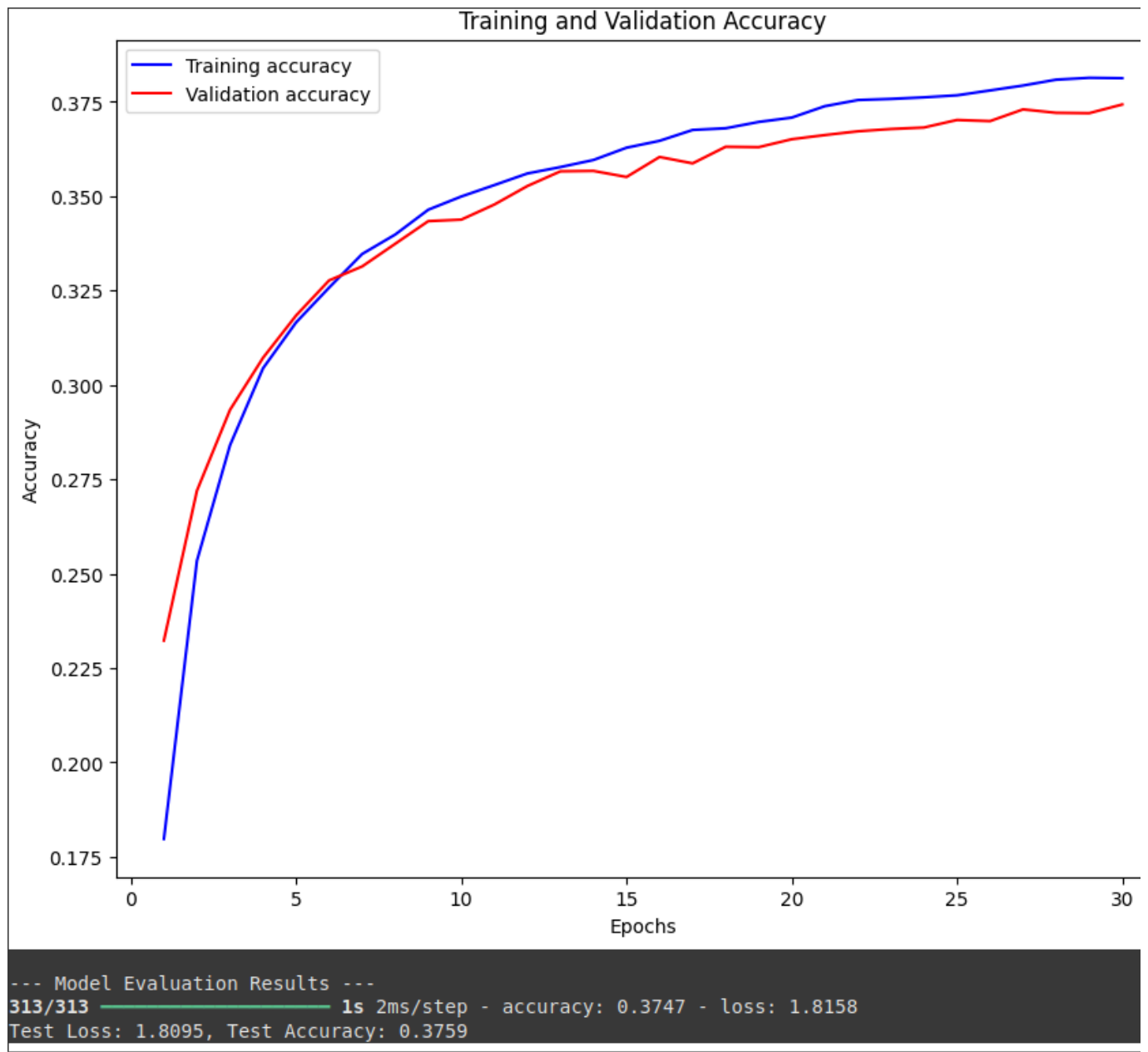
Batch size == 10000



```
--- Model Evaluation Results ---  
313/313 ————— 1s 2ms/step - accuracy: 0.2830 - loss: 2.0227  
Test Loss: 2.0207, Test Accuracy: 0.2870
```

- 정확도가 0.28수준으로 크게 떨어짐

Batch size == 1000



- 정확도가 0.37 정도로 기존과 0.01정도 차이남

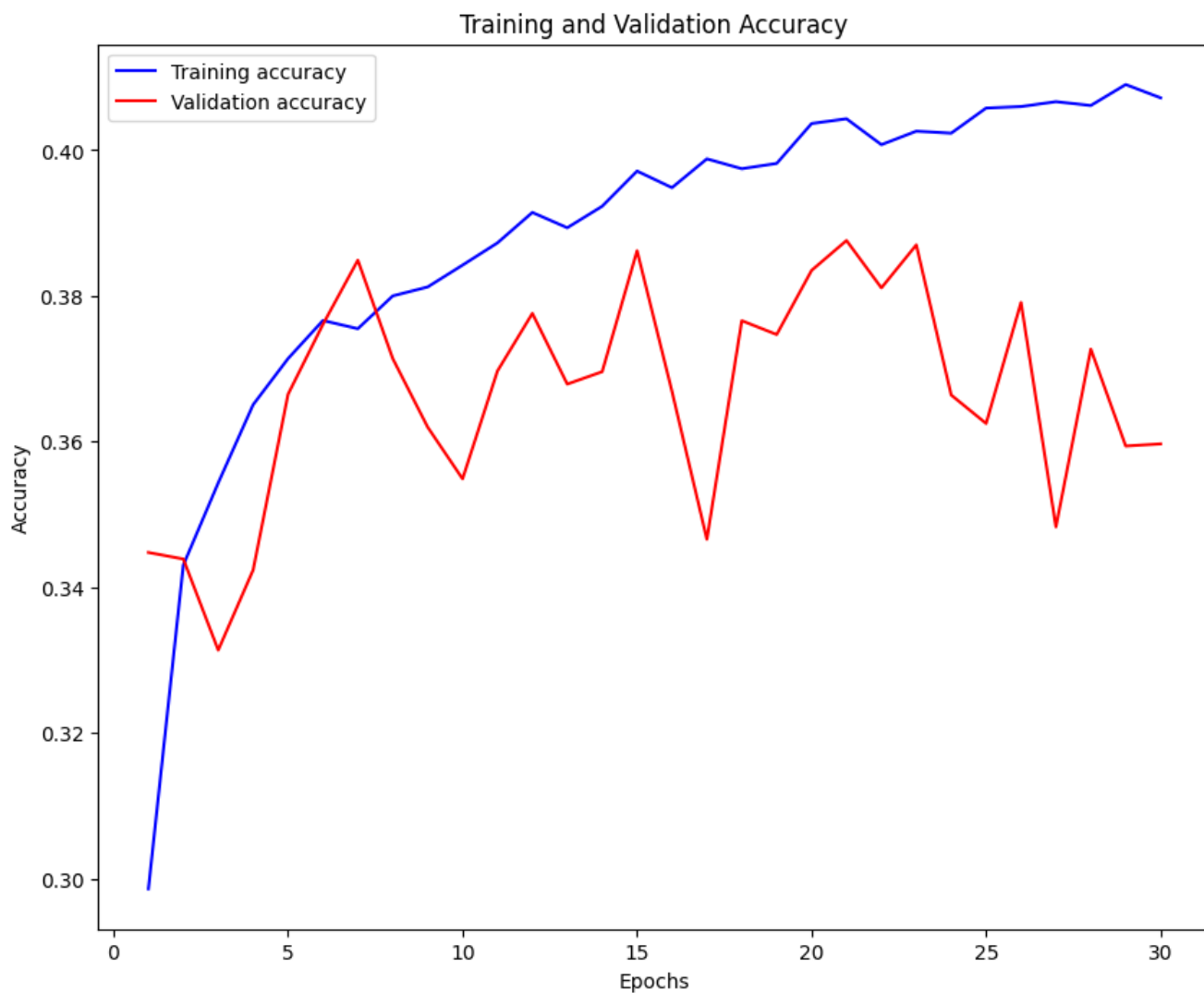
Batch size == 100





- 정확도가 0.40수준으로 향상됨

Batch size == 20



```

--- Model Evaluation Results ---
313/313 1s 2ms/step - accuracy: 0.3595 - loss: 1.8876
Test Loss: 1.8880, Test Accuracy: 0.3540

```

- 정확도 0.35 수준으로 떨어짐
- 그리고 batch size가 줄수록 학습시간이 오래 걸림

## Epoch 변경

지금까지 각 파라미터들 중 가장 정확도가 높았던 것만 뽑아 epoch값만 바꿔줌

Epoch == 100



```
--- Model Evaluation Results ---  
313/313 1s 2ms/step - accuracy: 0.3975 - loss: 1.7302  
Test Loss: 1.7297, Test Accuracy: 0.4020
```

- epoch를 64 --> 100으로 증가시키니 정확도가 0.40정도 향상되었다
- 다만, 기존 batch size = 100일 때 0.40정도의 정확도를 보여줬는데
  - 해당 조건을 유지한 채, 학습횟수를 늘리면 정확도가 상승할 것이라는 예측과 달리 오히려 유지되는 모습을 보인다