와인 리뷰데이터를 활용한 품종 분류 모델

SIMPLE PROJECT PROPOSAL TEMPLATE

2024.12.03

@ 202004100 장주훈

@ 202004015 김두섭

CONTENTS

목차

프로젝트 배경
CONTENTS 01

프로젝트 개요
CONTENTS 02

데이터 확인
CONTENTS 03

코드 내용
CONTENTS 04

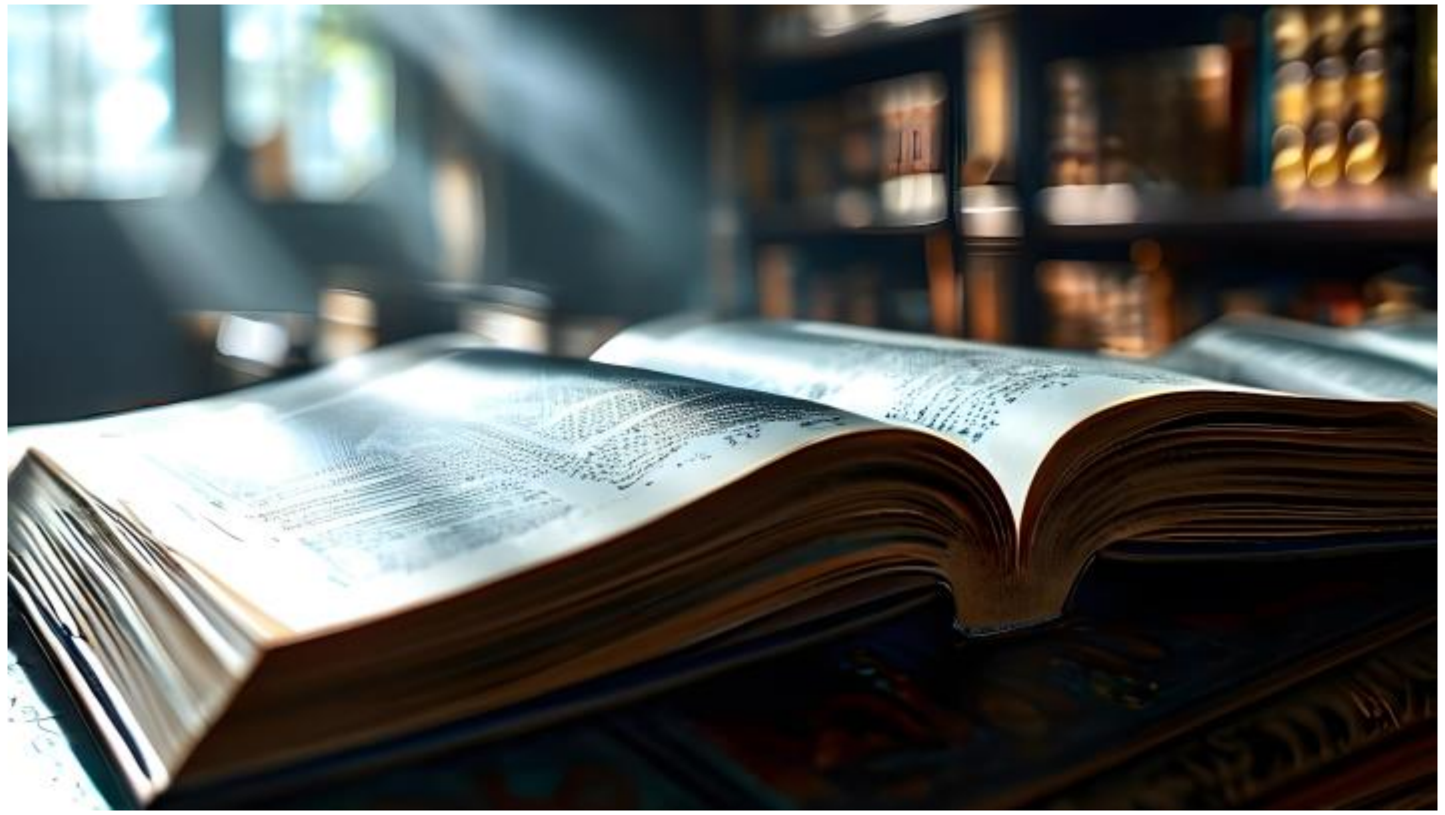
구현 내용
CONTENTS 05

평가지표
CONTENTS 06

프로젝트 배경

프로젝트 배경

- 와인은 오랜 역사와 다양한 품종으로 인해 많은 사람들에게 사랑받고 있습니다. 너무 많은 종류와 품종으로 입문이 어렵다고 느끼는 경우가 많은데 이를 도와주는 자연어처리 모델이 있으면 편리할 것이라고 생각했습니다.



SUMMARY

프로젝트 개요

프로젝트 목표

- 다양한 와인 리뷰 데이터를 기반으로 와인의 종류(RED/WHITE)을 정확하게 분류하는 LSTM 모델을 개발합니다.
- 분류된 품종에 따라 적절한 음식 페어링을 추천하는 시스템을 구축하고자 합니다.

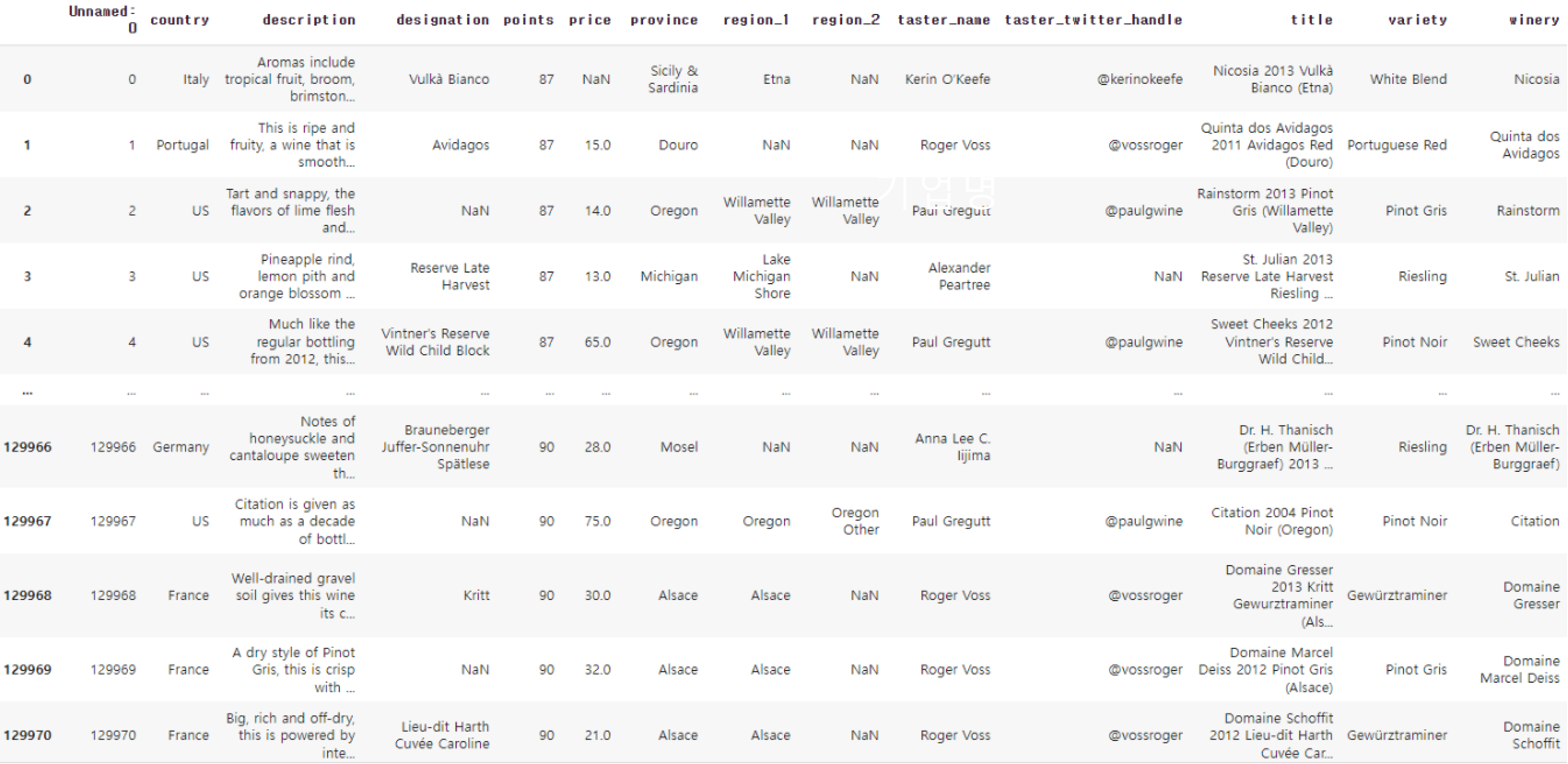
프로젝트 주요 방안

- 와인 리뷰 데이터 전처리
- 모델 선정(LSTM)
- 텍스트 기반 추천 시스템 개발

프로젝트 실행 계획

- 다양한 와인 리뷰 데이터를 기반으로 와인의 품종을 정확하게 분류하는 LSTM 모델을 개발합니다.
- 분류된 품종에 따라 적절한 음식 페어링을 추천하는 시스템을 구축합니다.

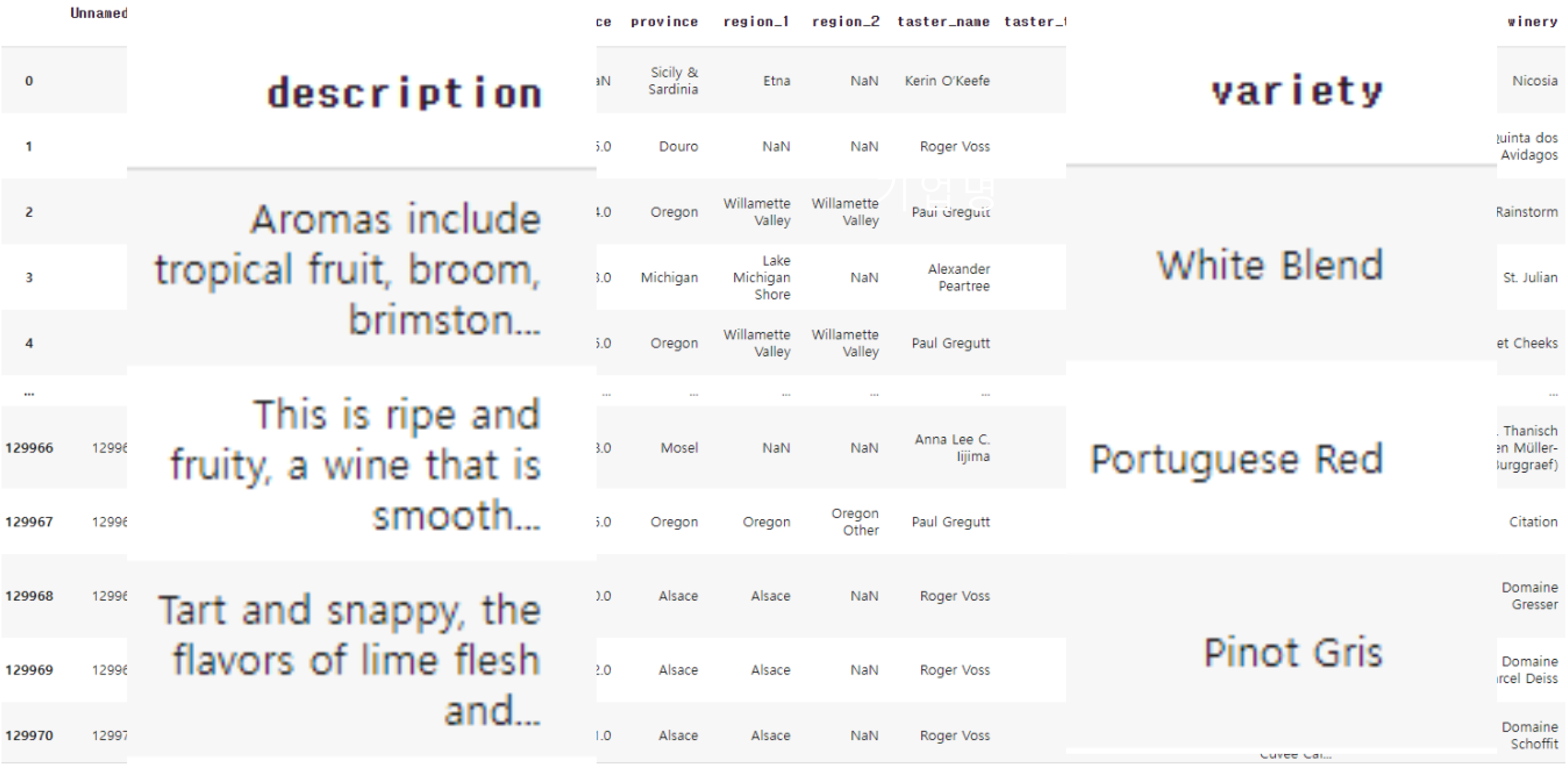
데이터 확인



	Unnamed: 0	country	description	designation	points	price	province	region_1	region_2	taster_name	taster_twitter_handle	title	variety	winery
0	0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Sicily & Sardinia	Etna	NaN	Kerin O'Keefe	@kerinokeefe	Nicosia 2013 Vulkà Bianco (Etna)	White Blend	Nicosia
1	1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Douro	NaN	NaN	Roger Voss	@vossroger	Quinta dos Avidagos 2011 Avidagos Red (Douro)	Portuguese Red	Quinta dos Avidagos
2	2	US	Tart and snappy, the flavors of lime flesh and...	NaN	87	14.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregutt	@paulgwine	Rainstorm 2013 Pinot Gris (Willamette Valley)	Pinot Gris	Rainstorm
3	3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	Michigan	Lake Michigan Shore	NaN	Alexander Peartree	NaN	St. Julian 2013 Reserve Late Harvest Riesling ...	Riesling	St. Julian
4	4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	87	65.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregutt	@paulgwine	Sweet Cheeks 2012 Vintner's Reserve Wild Child...	Pinot Noir	Sweet Cheeks
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
129966	129966	Germany	Notes of honeysuckle and cantaloupe sweeten th...	Brauneberger Juffer-Sonnenuhr Spätlese	90	28.0	Mosel	NaN	NaN	Anna Lee C. Iijima	NaN	Dr. H. Thanisch (Erben Müller-Burggraef) 2013 ...	Riesling	Dr. H. Thanisch (Erben Müller-Burggraef)
129967	129967	US	Citation is given as much as a decade of bottl...	NaN	90	75.0	Oregon	Oregon	Oregon Other	Paul Gregutt	@paulgwine	Citation 2004 Pinot Noir (Oregon)	Pinot Noir	Citation
129968	129968	France	Well-drained gravel soil gives this wine its c...	Kritt	90	30.0	Alsace	Alsace	NaN	Roger Voss	@vossroger	Domaine Gresser 2013 Kritt Gewurztraminer (Als...	Gewürztraminer	Domaine Gresser
129969	129969	France	A dry style of Pinot Gris, this is crisp with ...	NaN	90	32.0	Alsace	Alsace	NaN	Roger Voss	@vossroger	Domaine Marcel Deiss 2012 Pinot Gris (Alsace)	Pinot Gris	Domaine Marcel Deiss
129970	129970	France	Big, rich and off-dry, this is powered by inte...	Lieu-dit Harth Cuvée Caroline	90	21.0	Alsace	Alsace	NaN	Roger Voss	@vossroger	Domaine Schoffit 2012 Lieu-dit Harth Cuvée Car...	Gewürztraminer	Domaine Schoffit

DATA

데이터 확인



	Unnamed	ce	province	region_1	region_2	taster_name	taster_	winery
0		aN	Sicily & Sardinia	Etna	NaN	Kerin O'Keefe		Nicosia
1		5.0	Douro	NaN	NaN	Roger Voss		uinta dos Avidagos
2		4.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregutt		Rainstorm
3		3.0	Michigan	Lake Michigan Shore	NaN	Alexander Peartree		St. Julian
4		5.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregutt		et Cheeks
...		...	...	...	...	...		...
129966	12996	3.0	Mosel	NaN	NaN	Anna Lee C. Iijima		. Thanisch en Müller-Burggraef)
129967	12996	5.0	Oregon	Oregon	Oregon Other	Paul Gregutt		Citation
129968	12996	0.0	Alsace	Alsace	NaN	Roger Voss		Domaine Gresser
129969	12996	2.0	Alsace	Alsace	NaN	Roger Voss		Domaine rcel Deiss
129970	12997	1.0	Alsace	Alsace	NaN	Roger Voss		Domaine Schoffit

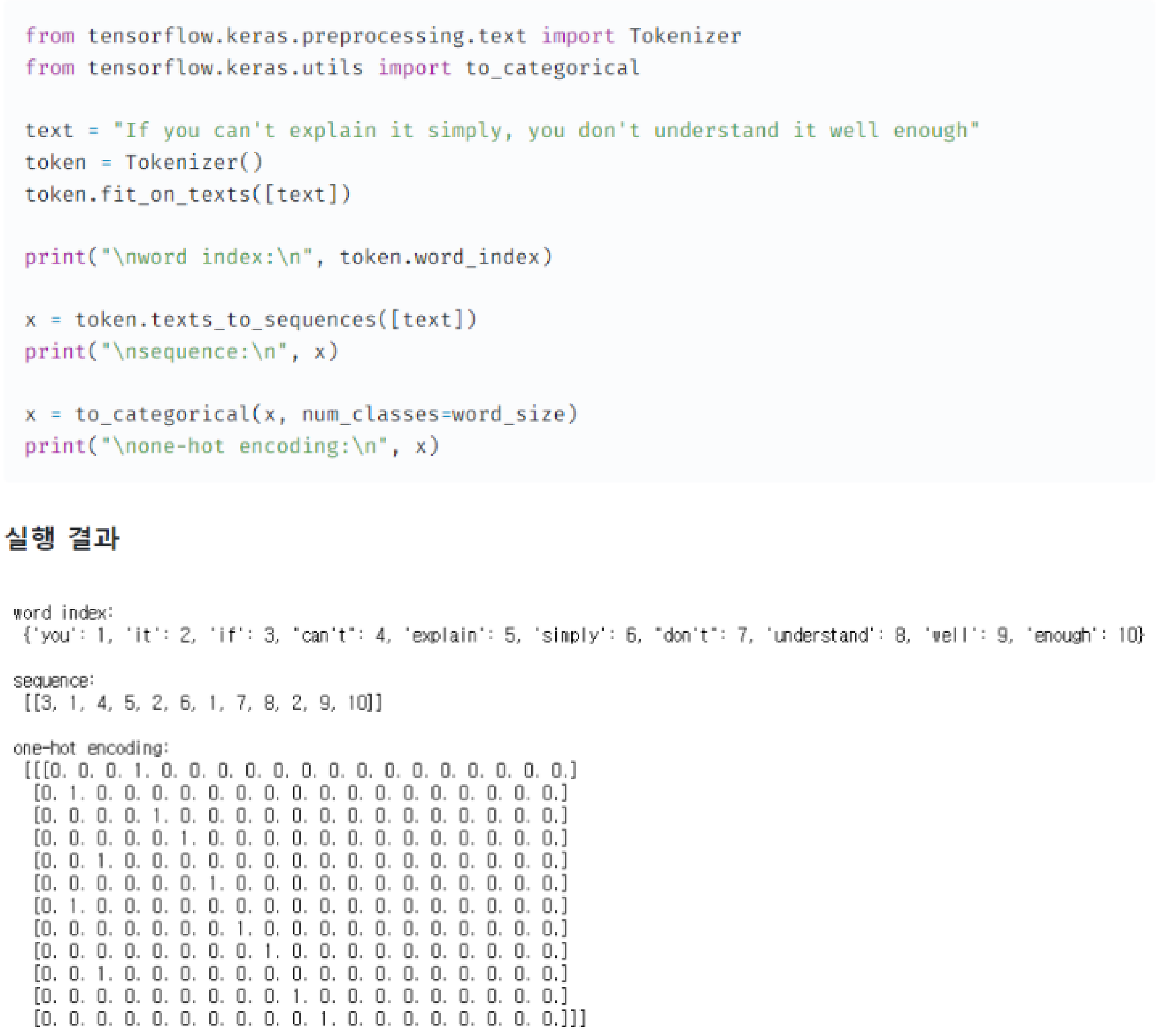
description
Aromas include tropical fruit, broom, brimston...
This is ripe and fruity, a wine that is smooth...
Tart and snappy, the flavors of lime flesh and...

variety
White Blend
Portuguese Red
Pinot Gris

Cuvee Cal...

CODE

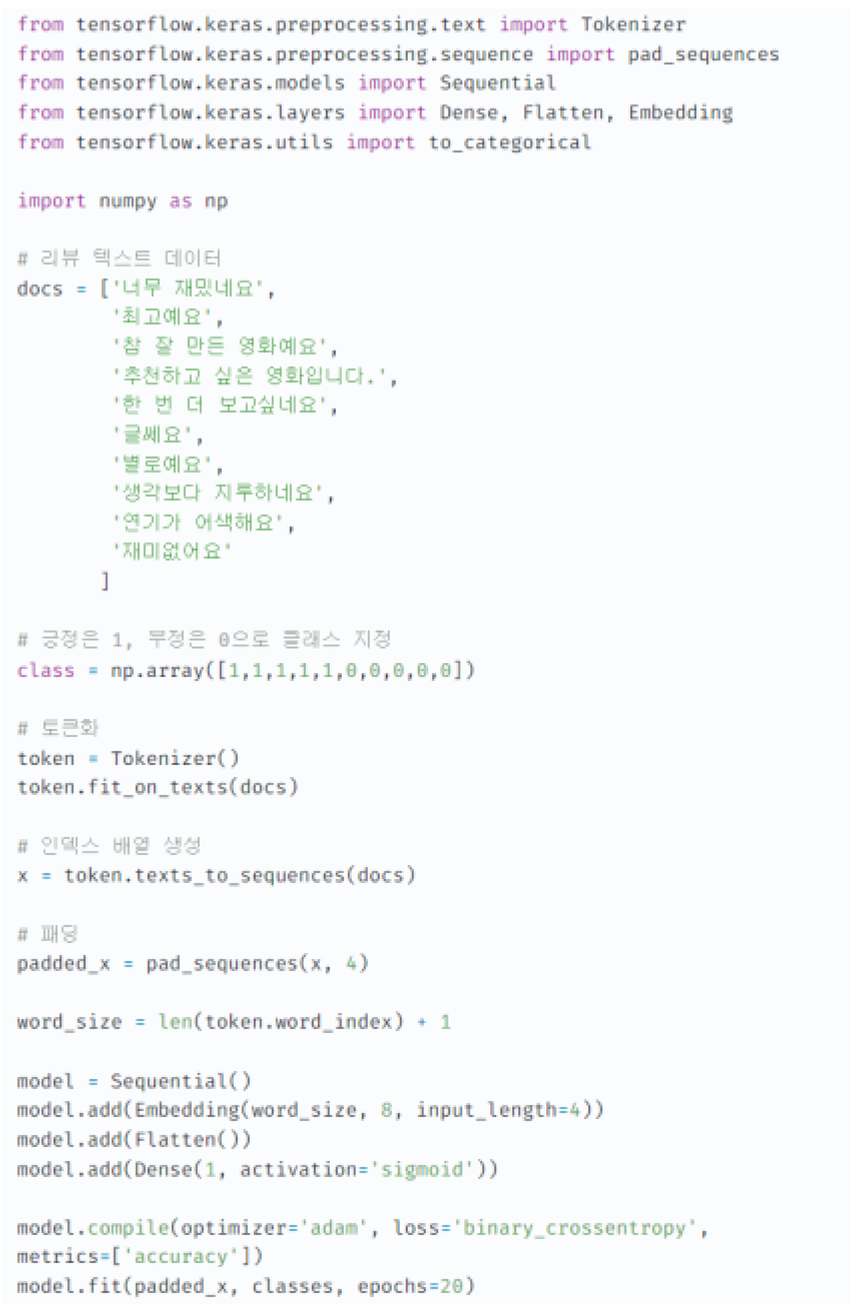
코드 분석



```python
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.utils import to_categorical

text = "If you can't explain it simply, you don't understand it well enough"
token = Tokenizer()
token.fit_on_texts([text])

print("\nword index:\n", token.word_index)

x = token.texts_to_sequences([text])
print("\nsequence:\n", x)

x = to_categorical(x, num_classes=word_size)
print("\none-hot encoding:\n", x)
```

실행 결과

```
word index:
 {'you': 1, 'it': 2, 'if': 3, "can't": 4, 'explain': 5, 'simply': 6, "don't": 7, 'understand': 8, 'well': 9, 'enough': 10}

sequence:
 [[3, 1, 4, 5, 2, 6, 1, 7, 8, 2, 9, 10]]

one-hot encoding:
 [[[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
  [0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]]]
```

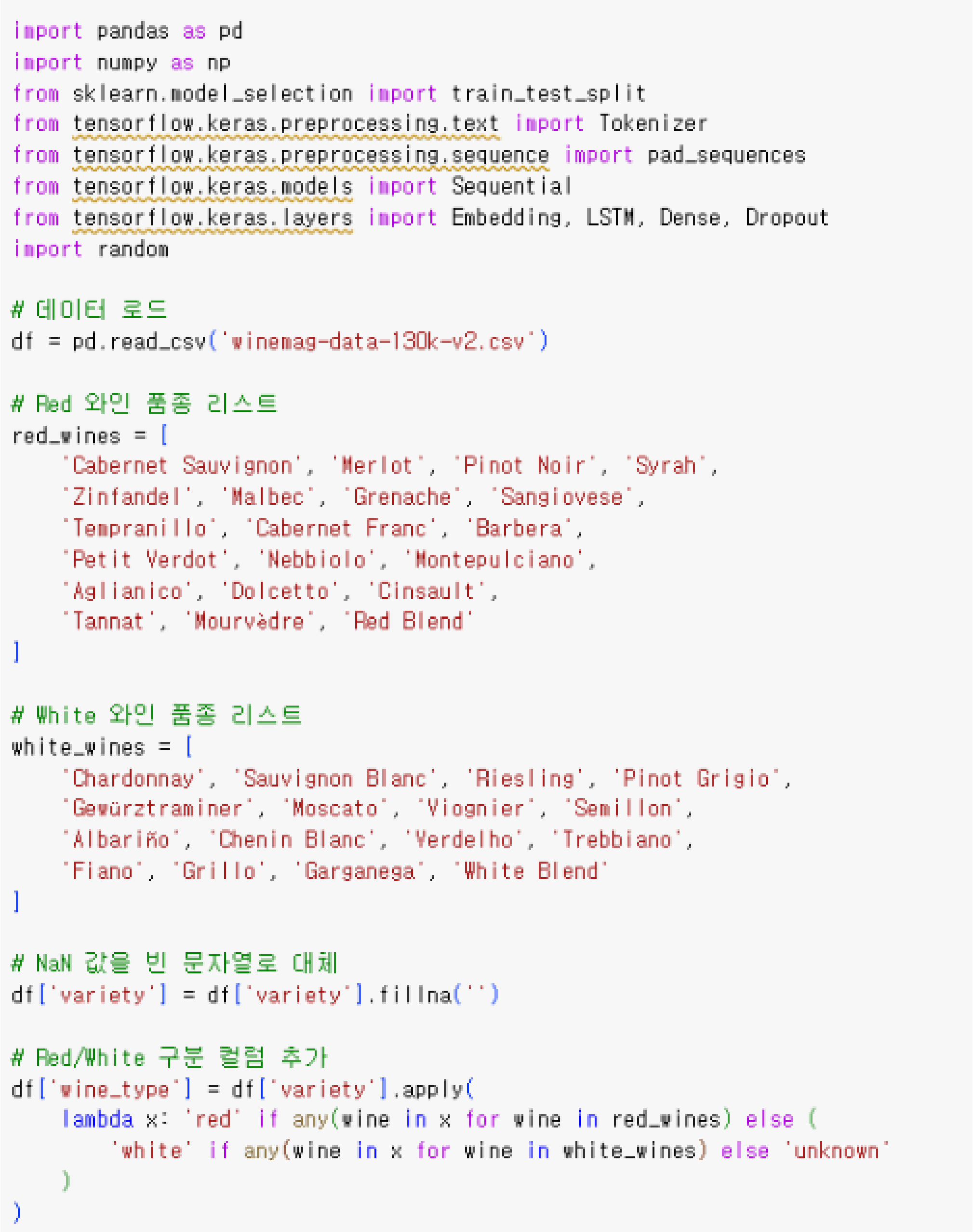


```python
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Flatten, Embedding
from tensorflow.keras.utils import to_categorical

import numpy as np

# 리뷰 텍스트 데이터
docs = ['너무 재밌네요',
        '최고예요',
        '참 잘 만든 영화예요',
        '추천하고 싶은 영화입니다.',
        '한 번 더 보고싶네요',
        '글쎄요',
        '별로예요',
        '생각보다 지루하네요',
        '연기가 어색해요',
        '재미없어요'
       ]

# 긍정은 1, 부정은 0으로 클래스 지정
class = np.array([1,1,1,1,1,0,0,0,0,0])

# 토큰화
token = Tokenizer()
token.fit_on_texts(docs)

# 인덱스 배열 생성
x = token.texts_to_sequences(docs)

# 패딩
padded_x = pad_sequences(x, 4)

word_size = len(token.word_index) + 1

model = Sequential()
model.add(Embedding(word_size, 8, input_length=4))
model.add(Flatten())
model.add(Dense(1, activation='sigmoid'))

model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])
model.fit(padded_x, classes, epochs=20)
```

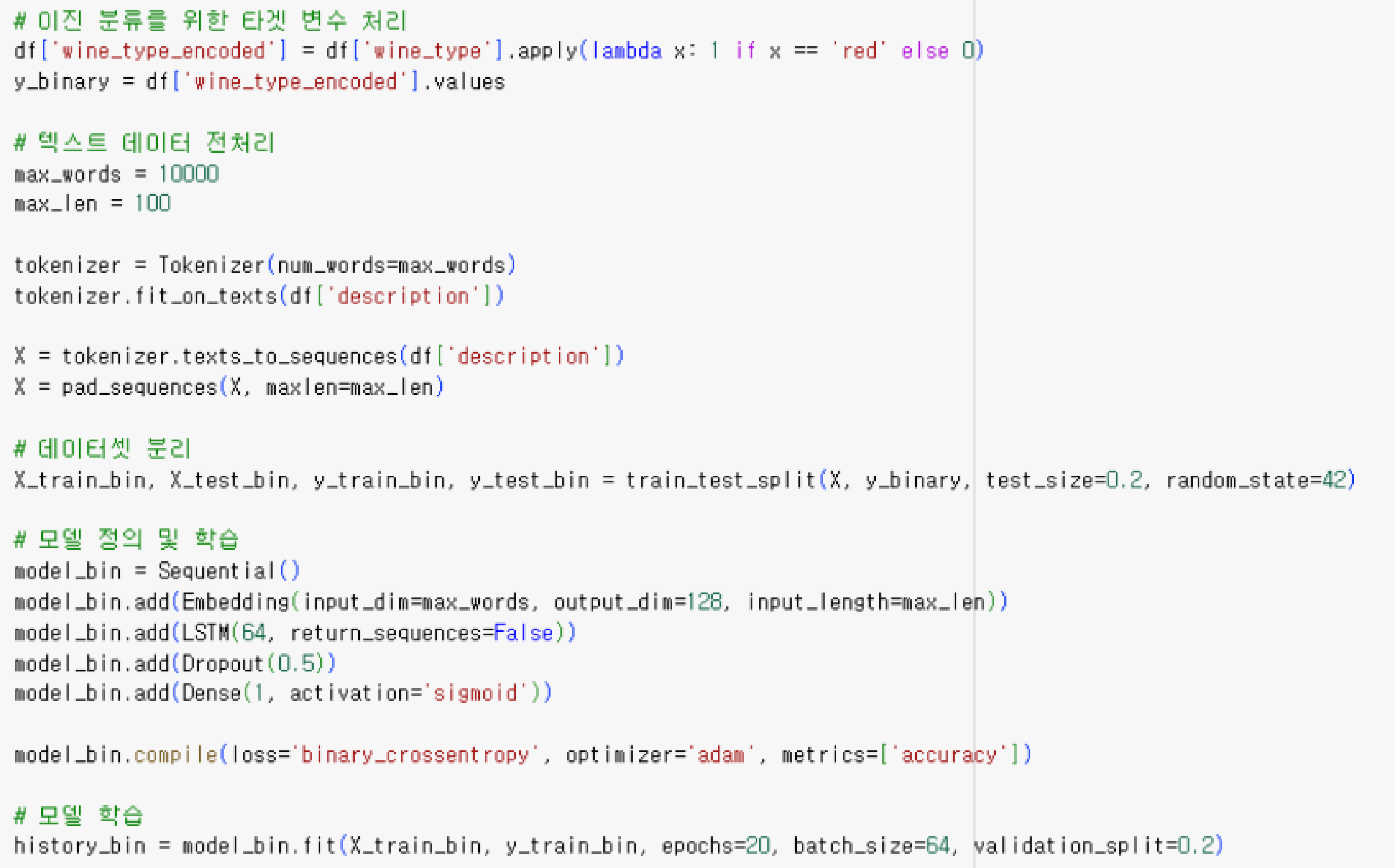
영화 리뷰를 긍정적인지 부정적인지 판단하는 모델

https://velog.io/@oilyhand_01/%EB%94%A5%EB%9F%AC%EB%8B%9D-%EA%B8%B0%EC%B4%88-

1차 구현 내용



```python
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Embedding, LSTM, Dense, Dropout
import random

# 데이터 로드
df = pd.read_csv('winemag-data-130k-v2.csv')

# Red 와인 품종 리스트
red_wines = [
    'Cabernet Sauvignon', 'Merlot', 'Pinot Noir', 'Syrah',
    'Zinfandel', 'Malbec', 'Grenache', 'Sangiovese',
    'Tempranillo', 'Cabernet Franc', 'Barbera',
    'Petit Verdot', 'Nebbiolo', 'Montepulciano',
    'Aglianico', 'Dolcetto', 'Cinsault',
    'Tannat', 'Mourvèdre', 'Red Blend'
]

# White 와인 품종 리스트
white_wines = [
    'Chardonnay', 'Sauvignon Blanc', 'Riesling', 'Pinot Grigio',
    'Gewürztraminer', 'Moscato', 'Viognier', 'Semillon',
    'Albariño', 'Chenin Blanc', 'Verdelho', 'Trebbiano',
    'Fiano', 'Grillo', 'Garganega', 'White Blend'
]

# NaN 값을 빈 문자열로 대체
df['variety'] = df['variety'].fillna('')

# Red/White 구분 컬럼 추가
df['wine_type'] = df['variety'].apply(
    lambda x: 'red' if any(wine in x for wine in red_wines) else (
        'white' if any(wine in x for wine in white_wines) else 'unknown'
    )
)
```



```python
# 이진 분류를 위한 타겟 변수 처리
df['wine_type_encoded'] = df['wine_type'].apply(lambda x: 1 if x == 'red' else 0)
y_binary = df['wine_type_encoded'].values

# 텍스트 데이터 전처리
max_words = 10000
max_len = 100

tokenizer = Tokenizer(num_words=max_words)
tokenizer.fit_on_texts(df['description'])

X = tokenizer.texts_to_sequences(df['description'])
X = pad_sequences(X, maxlen=max_len)

# 데이터셋 분리
X_train_bin, X_test_bin, y_train_bin, y_test_bin = train_test_split(X, y_binary, test_size=0.2, random_state=42)

# 모델 정의 및 학습
model_bin = Sequential()
model_bin.add(Embedding(input_dim=max_words, output_dim=128, input_length=max_len))
model_bin.add(LSTM(64, return_sequences=False))
model_bin.add(Dropout(0.5))
model_bin.add(Dense(1, activation='sigmoid'))

model_bin.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

# 모델 학습
history_bin = model_bin.fit(X_train_bin, y_train_bin, epochs=20, batch_size=64, validation_split=0.2)
```

와인 리뷰 데이터를 활용해 품종을 통해 RED와 WHITE로 라벨링 후, 리뷰 데이터만으로 RED/WHITE 구분

추가한 내용

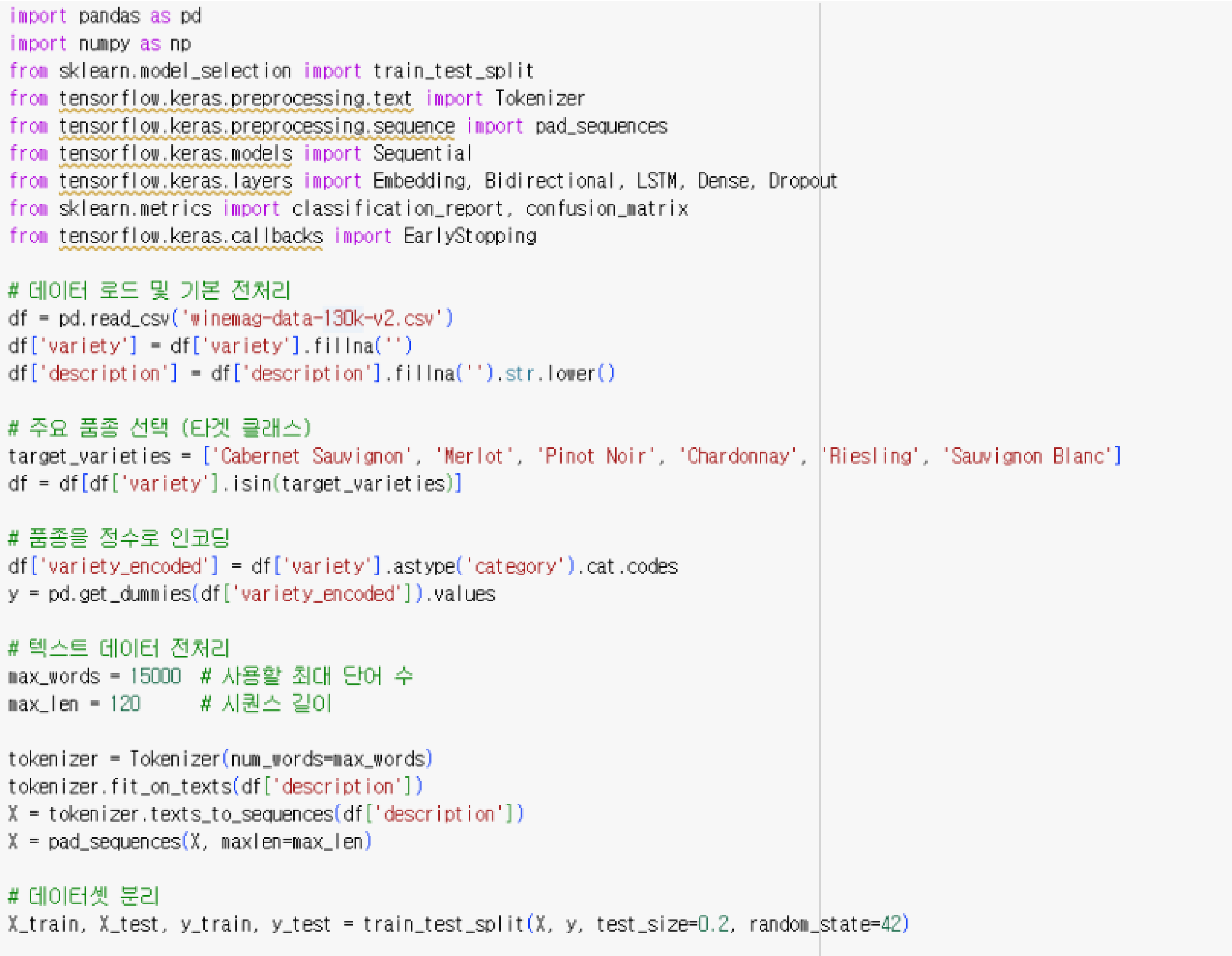
1. 사용자가 와인 설명을 입력하면, 모델이 입력된 설명을 기반으로 어떤 와인 품종인지 예측
2. 예측된 와인 품종에 따라 어울리는 음식 추천

gpt에게 뉴질랜드 배비치 블랙 소비뇽 블랑의 영어로 된 와인 리뷰를 한개만 가져다 줘 질문

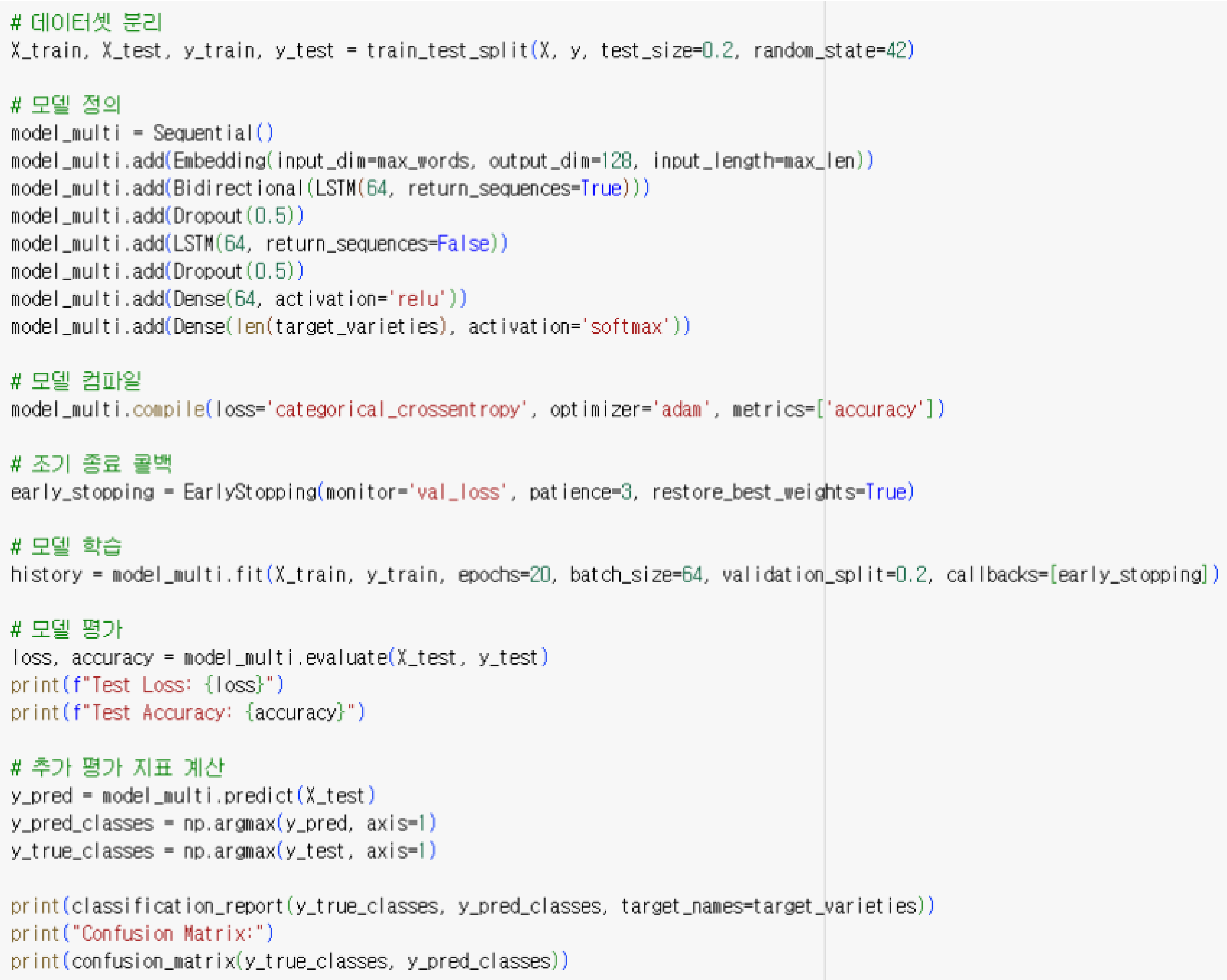
Here is an English wine review for the Babich Black Label Sauvignon Blanc 2023:

The wine is described as inviting and musky, with aromas of gooseberry, mandarin, and a hint of fennel on the nose. On the palate, it is fresh and lively with flavors of lemon, lime, gooseberry, and blackcurrant. The mid-palate develops with additional flavors of grapefruit and honeydew melon, while pineapple is noticeable on the finish. The wine is characterized as soft and layered, with a lingering spicy finish

최종 구현 내용



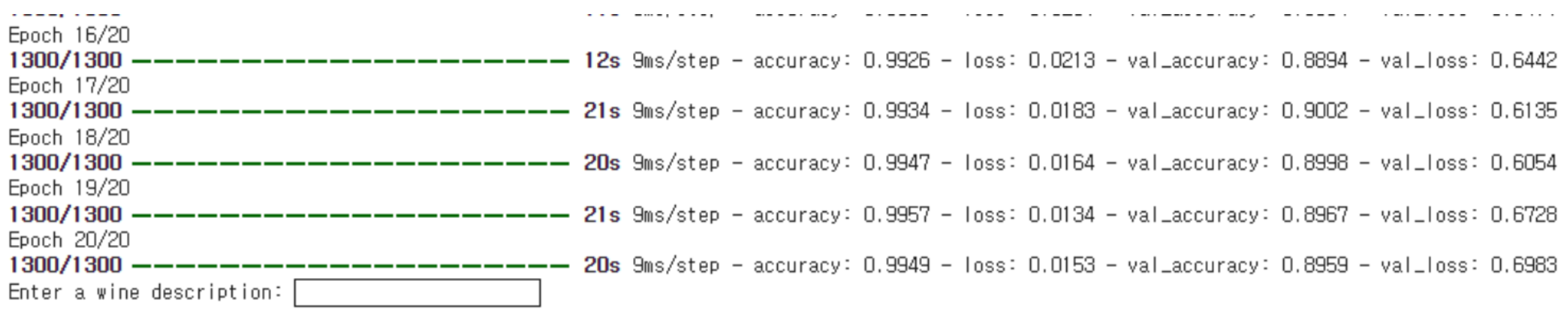
```python
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Embedding, Bidirectional, LSTM, Dense, Dropout
from sklearn.metrics import classification_report, confusion_matrix
from tensorflow.keras.callbacks import EarlyStopping

# 데이터 로드 및 기본 전처리
df = pd.read_csv('winemag-data-130k-v2.csv')
df['variety'] = df['variety'].fillna('')
df['description'] = df['description'].fillna('').str.lower()

# 주요 품종 선택 (타겟 클래스)
target_varieties = ['Cabernet Sauvignon', 'Merlot', 'Pinot Noir', 'Chardonnay', 'Riesling', 'Sauvignon Blanc']
df = df[df['variety'].isin(target_varieties)]

# 품종을 정수로 인코딩
df['variety_encoded'] = df['variety'].astype('category').cat.codes
y = pd.get_dummies(df['variety_encoded']).values

# 텍스트 데이터 전처리
max_words = 15000  # 사용할 최대 단어 수
max_len = 120      # 시퀀스 길이

tokenizer = Tokenizer(num_words=max_words)
tokenizer.fit_on_texts(df['description'])
X = tokenizer.texts_to_sequences(df['description'])
X = pad_sequences(X, maxlen=max_len)

# 데이터셋 분리
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```



```python
# 데이터셋 분리
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# 모델 정의
model_multi = Sequential()
model_multi.add(Embedding(input_dim=max_words, output_dim=128, input_length=max_len))
model_multi.add(Bidirectional(LSTM(64, return_sequences=True)))
model_multi.add(Dropout(0.5))
model_multi.add(LSTM(64, return_sequences=False))
model_multi.add(Dropout(0.5))
model_multi.add(Dense(64, activation='relu'))
model_multi.add(Dense(len(target_varieties), activation='softmax'))

# 모델 컴파일
model_multi.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

# 조기 종료 콜백
early_stopping = EarlyStopping(monitor='val_loss', patience=3, restore_best_weights=True)

# 모델 학습
history = model_multi.fit(X_train, y_train, epochs=20, batch_size=64, validation_split=0.2, callbacks=[early_stopping])

# 모델 평가
loss, accuracy = model_multi.evaluate(X_test, y_test)
print(f"Test Loss: {loss}")
print(f"Test Accuracy: {accuracy}")

# 추가 평가 지표 계산
y_pred = model_multi.predict(X_test)
y_pred_classes = np.argmax(y_pred, axis=1)
y_true_classes = np.argmax(y_test, axis=1)

print(classification_report(y_true_classes, y_pred_classes, target_names=target_varieties))
print("Confusion Matrix:")
print(confusion_matrix(y_true_classes, y_pred_classes))
```

RESULT

결과 및 평가지표

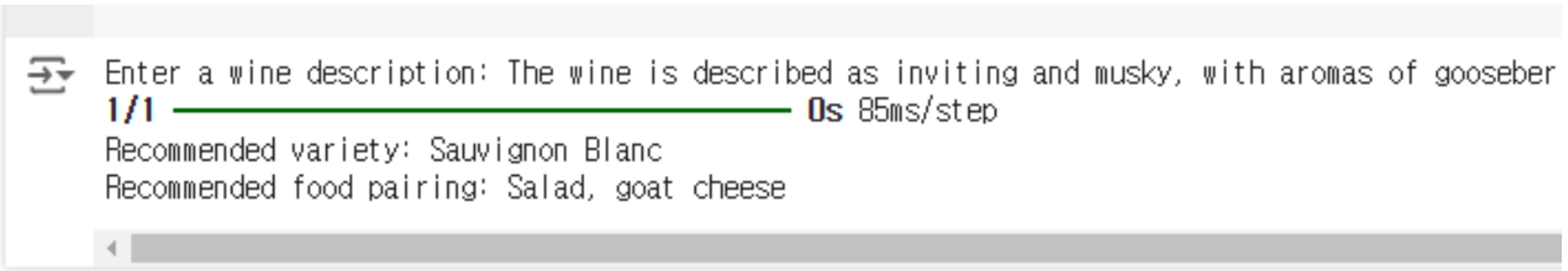


```
Epoch 16/20
1300/1300 ━━━━━━━━━━━━━━━━━━━━ 12s 9ms/step - accuracy: 0.9926 - loss: 0.0213 - val_accuracy: 0.8894 - val_loss: 0.6442
Epoch 17/20
1300/1300 ━━━━━━━━━━━━━━━━━━━━ 21s 9ms/step - accuracy: 0.9934 - loss: 0.0183 - val_accuracy: 0.9002 - val_loss: 0.6135
Epoch 18/20
1300/1300 ━━━━━━━━━━━━━━━━━━━━ 20s 9ms/step - accuracy: 0.9947 - loss: 0.0164 - val_accuracy: 0.8998 - val_loss: 0.6054
Epoch 19/20
1300/1300 ━━━━━━━━━━━━━━━━━━━━ 21s 9ms/step - accuracy: 0.9957 - loss: 0.0134 - val_accuracy: 0.8967 - val_loss: 0.6728
Epoch 20/20
1300/1300 ━━━━━━━━━━━━━━━━━━━━ 20s 9ms/step - accuracy: 0.9949 - loss: 0.0153 - val_accuracy: 0.8959 - val_loss: 0.6983
Enter a wine description: 
```

gpt에게 뉴질랜드 배비치 블랙 소비뇽 블랑의 영어로 된 와인 리뷰를 한개만 가져다 줘 질문

Here is an English wine review for the Babich Black Label Sauvignon Blanc 2023:

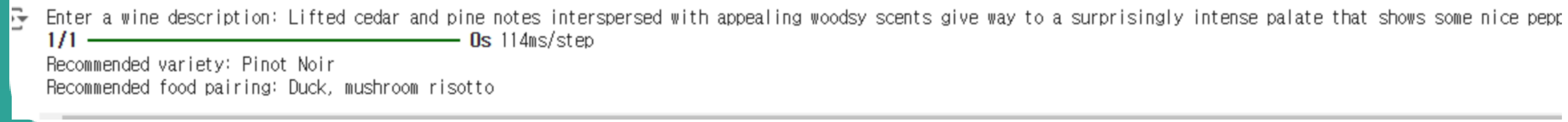
The wine is described as inviting and musky, with aromas of gooseberry, mandarin, and a hint of fennel on the nose. On the palate, it is fresh and lively with flavors of lemon, lime, gooseberry, and blackcurrant. The mid-palate develops with additional flavors of grapefruit and honeydew melon, while pineapple is noticeable on the finish. The wine is characterized as soft and layered, with a lingering spicy finish



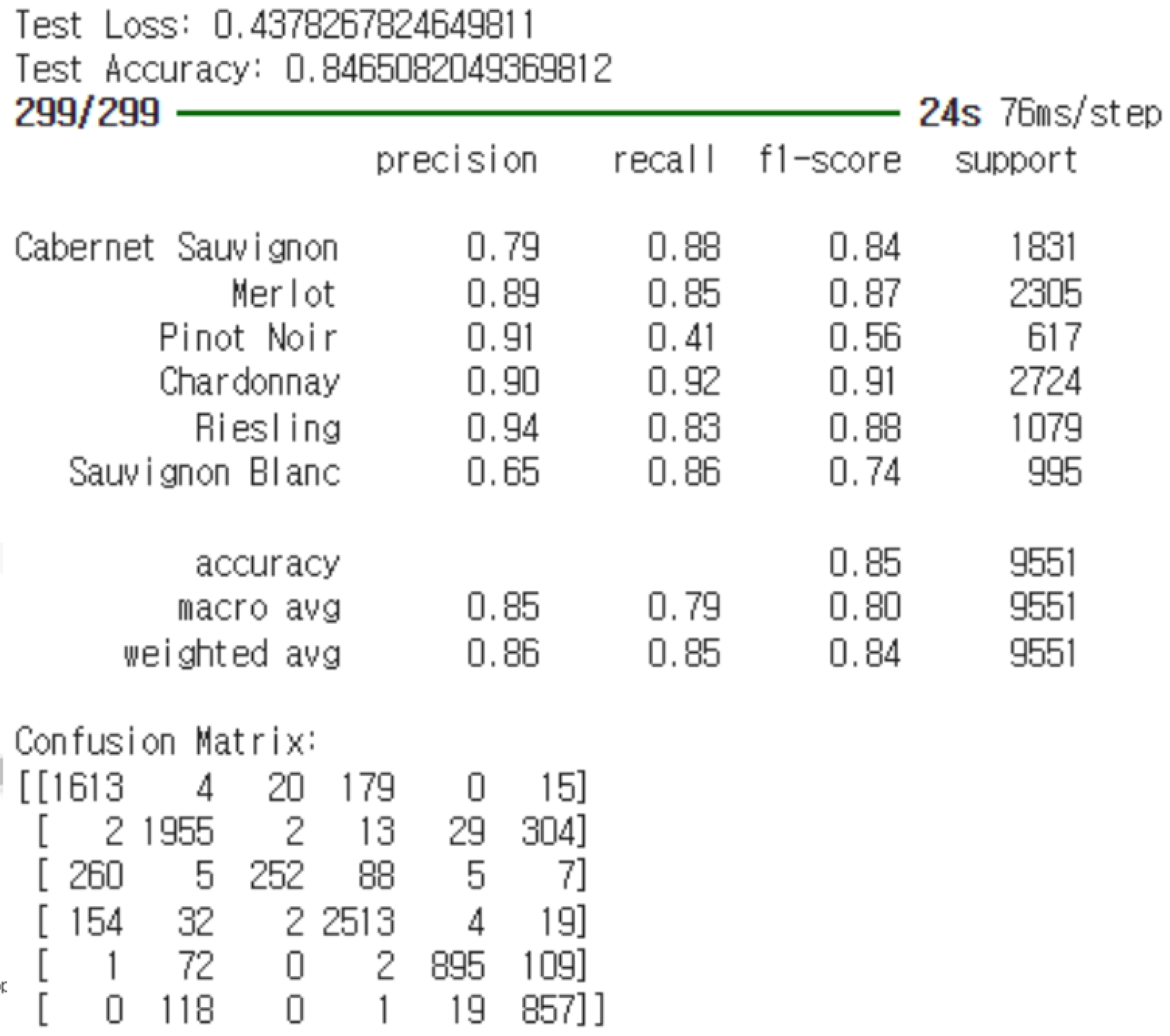
```
Enter a wine description: The wine is described as inviting and musky, with aromas of gooseber
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 85ms/step
Recommended variety: Sauvignon Blanc
Recommended food pairing: Salad, goat cheese
```

gpt에게 피노누아의 영어로 된 와인 리뷰를 한개만 가져다 줘 질문

Lifted cedar and pine notes interspersed with appealing woodsy scents give way to a surprisingly intense palate that shows some nice peppery notes but ultimately falls short on the finish. Nearly imperceptible tannins suggest early consumption.



```
Enter a wine description: Lifted cedar and pine notes interspersed with appealing woodsy scents give way to a surprisingly intense palate that shows some nice pepp
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 114ms/step
Recommended variety: Pinot Noir
Recommended food pairing: Duck, mushroom risotto
```



```
Test Loss: 0.4378267824649811
Test Accuracy: 0.8465082049369812
299/299 ━━━━━━━━━━━━━━━━━━━━ 24s 76ms/step
                    precision    recall  f1-score   support

Cabernet Sauvignon       0.79      0.88      0.84      1831
            Merlot       0.89      0.85      0.87      2305
        Pinot Noir       0.91      0.41      0.56       617
        Chardonnay       0.90      0.92      0.91      2724
          Riesling       0.94      0.83      0.88      1079
   Sauvignon Blanc       0.65      0.86      0.74       995

          accuracy                           0.85      9551
         macro avg       0.85      0.79      0.80      9551
      weighted avg       0.86      0.85      0.84      9551

Confusion Matrix:
[[1613    4   20  179    0   15]
 [   2 1955    2   13   29  304]
 [ 260    5  252   88    5    7]
 [ 154   32    2 2513    4   19]
 [   1   72    0    2  895  109]
 [   0  118    0    1   19  857]]
```