AIVLE

kt

KT AIVLE School

4차 미니프로젝트

AI 5반 14조

AIVLE

Let’s make it possible

결측치 제거

```
# 결측치를 제거한 후 확인합니다.
spam.isna().sum()
```

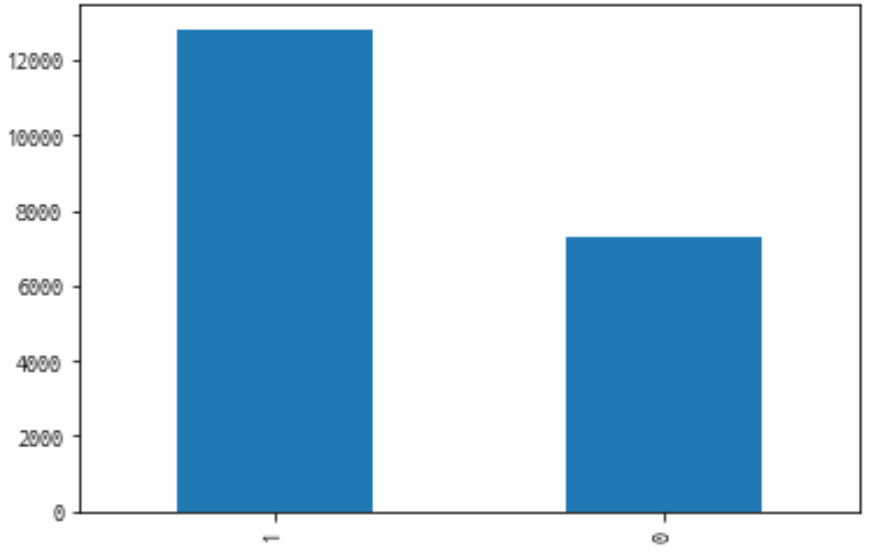
```
text     11
label     0
dtype: int64
```

```
spam.dropna(axis=0, inplace=True)
```

데이터 분석

```
spam['label'].value_counts().plot(kind='bar')
```

```
<AxesSubplot:>
```




```
print('정상 메일과 스팸 메일의 개수')
print(spam.groupby('label').size().reset_index(name='count'))

print(f'정상 메일의 비율 = {round(spam["label"].value_counts()[0]/len(spam) * 100,3)}%')
print(f'스팸 메일의 비율 = {round(spam["label"].value_counts()[1]/len(spam) * 100,3)}%')
```

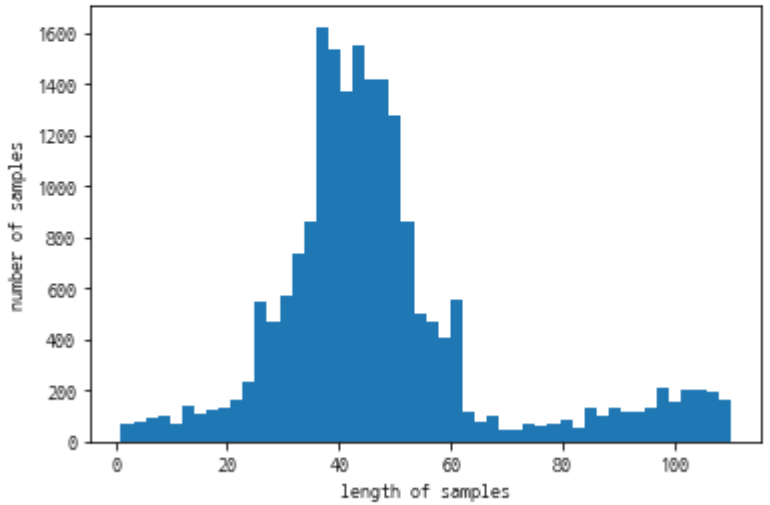
```
정상 메일과 스팸 메일의 개수
   label  count
0      0   7272
1      1  12817
정상 메일의 비율 = 36.199%
스팸 메일의 비율 = 63.801%
```

데이터 전처리

▪ Sequence Vectorize

```
tokenizer = Tokenizer()
tokenizer.fit_on_texts(x_t)
x_train_encodedt = tokenizer.texts_to_sequences(x_t)
print(x_train_encodedt[:5])

word_to_indext = tokenizer.word_index
#print(word_to_index)

threshold = 2
total_cnt = len(word_to_index)
rare_cnt = 0
total_freq = 0
rare_freq = 0

for key, value in tokenizer.word_counts.items():
    total_freq = total_freq+value
    if value<threshold:
        rare_cnt=rare_cnt+1
        rare_freq=rare_freq+value
print('등장 빈도가 %s번 이하인 희귀 단어의 수: %s'%(threshold - 1, rare_cnt))
print("단어 집합(vocabulary)에서 희귀 단어의 비율:", (rare_cnt / total_cnt)*100)
print("전체 등장 빈도에서 희귀 단어 등장 빈도 비율:", (rare_freq / total_freq)*100)

vocab_size = len(word_to_indext)+1
print('단어 집합의 크기: {}'.format((vocab_size)))

print('메일의 최대 길이 : %d' % max(len(sample) for sample in x_train_encodedt))
print('메일의 평균 길이 : %f' % (sum(map(len, x_train_encodedt))/len(x_train_encodedt)))
plt.hist([len(sample) for sample in x_t], bins=50)
plt.xlabel('length of samples')
plt.ylabel('number of samples')
plt.show()

max_len = 23
x_t_padded = pad_sequences(x_train_encodedt, maxlen = max_len)
print("훈련 데이터의 크기(shape):", x_t_padded.shape)
```

```
[[1, 5299, 9, 877], [2, 1271, 145, 5300, 2546, 122], [1, 878,
등장 빈도가 1번 이하인 희귀 단어의 수: 23920
단어 집합(vocabulary)에서 희귀 단어의 비율: 81.86734204942158
전체 등장 빈도에서 희귀 단어 등장 빈도 비율: 26.65151363216009
단어 집합의 크기: 29219
메일의 최대 길이 : 23
메일의 평균 길이 : 5.584656
```




```
훈련 데이터의 크기(shape): (16071, 23)
```

모델링

- **RNN**

```
from tensorflow.keras.layers import SimpleRNN, Embedding, Dense
from tensorflow.keras.models import Sequential

embedding_dim = 32
hidden_units = 32

model = Sequential()
model.add(Embedding(vocab_size, embedding_dim))
model.add(SimpleRNN(hidden_units))
model.add(Dense(1, activation='sigmoid'))

model.compile(optimizer='rmsprop', loss='binary_crossentropy', metrics=['acc'])
model.fit(x_train2, y_train, epochs=4, batch_size=64, validation_split=0.2)
```

```
x_test_encoded2 = tokenizer.texts_to_sequences(x_val)
x_test_padded2 = pad_sequences(x_test_encoded2, maxlen = max_len)
print("\n 테스트 정확도: %.4f" % (model.evaluate(x_test_padded2, y_val)[1]))
```

```
126/126 [==============================] - 3s 20ms/step - loss: 0.2969 - acc: 0.8512

 테스트 정확도: 0.8512
```


kt
AIVLE
AIVLE
Let’s make it possible