머신러닝 악성 사이트 탐지 모델링

KT AIVLE 2기

a024138
박다민

01

데이터 처리

중복 데이터 제거

중복된 데이터를
어떻게 제거할까?

변수 중요도 평가

중요도를 파악하여
일부 데이터로만
검사를 시행.

테스트 데이터 처리

Test data의 결측치를
어떻게 처리하는가?

02

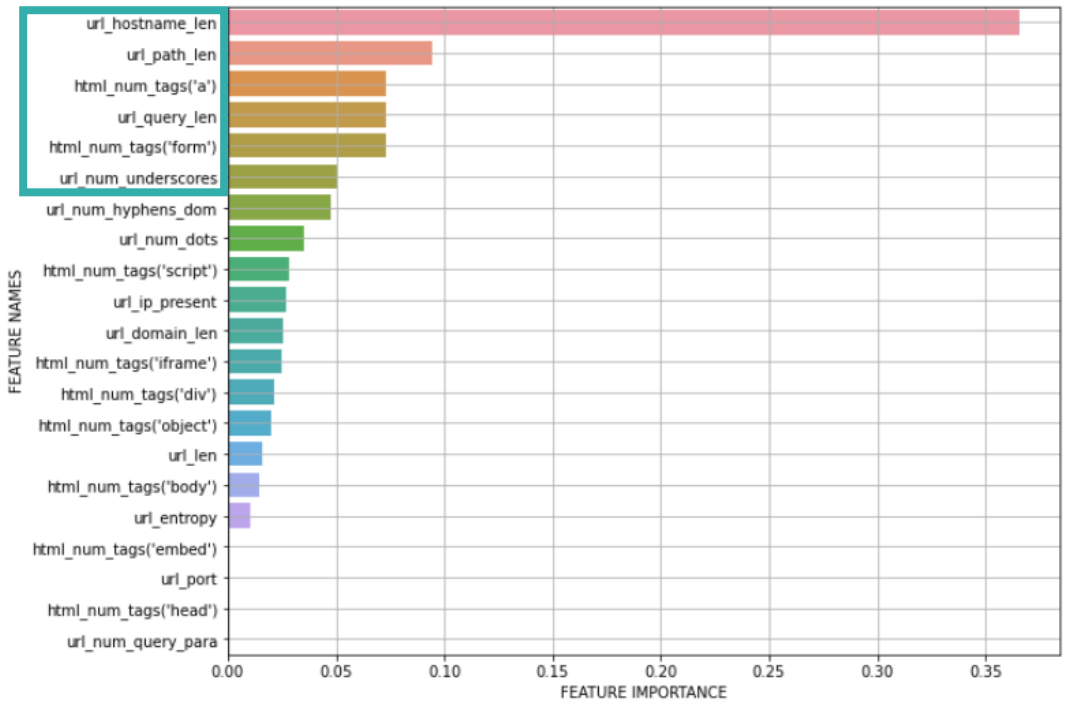
중복 데이터 제거

```
In [46]: data.shape
Out[46]: (3662, 22)

In [60]: columns = ['url_len','url_path_len','url_domain_len', 'url_hostname_len', 'url_num_dots','url_query_len','url_entropy']

In [61]: data.loc[data.duplicated(columns, keep='first')]

...

In [62]: #  columns의 열 데이터들이 중복되지 않은 값을 따로 가지고 온다.
         data2 = data.loc[data.duplicated(columns, keep='first')==False]

In [63]: data2.shape
Out[63]: (3037, 22)
```

로지스틱 회귀 등을 이용하여 **p-value가 0.05 이상**을 갖는 변수(feature)와
모든 값을 0으로 갖는 변수(url_chinese_present, html_num_tags('applet'))를 제거하였습니다.
이 후, **변수 중요도가 높은 변수들 또한 포함**하여 다시 진행하였습니다.

03

변수 중요도 평가




```python
def plot_feature_importance(importance, names):
    feature_importance = np.array(importance)
    feature_names = np.array(names)

    data={'feature_names':feature_names,'feature_importance':feature_importance}
    fi_df = pd.DataFrame(data)

    fi_df.sort_values(by=['feature_importance'], ascending=False,inplace=True)
    fi_df.reset_index(drop=True, inplace = True)

    plt.figure(figsize=(10,8))
    sns.barplot(x='feature_importance', y='feature_names', data = fi_df)

    plt.xlabel('FEATURE IMPORTANCE')
    plt.ylabel('FEATURE NAMES')
    plt.grid()

    return fi_df
```

강사님이 주셨던 코드를
이용하여 변수 중요도를 파악.
0.05 이상을 갖는 변수들을 이용하기로 결정.

04

테스트 데이터 처리

```
In [98]: data = pd.read_csv('test_dataset_v01.csv')
```

결측치 조치

```
In [154]: #data2 = data.interpolate(method='spline', order=2) # 음수 데이터 발생
```

```
In [153]: #data3 = data.interpolate(method='polynomial', order=2) # 음수 데이터 발생
```

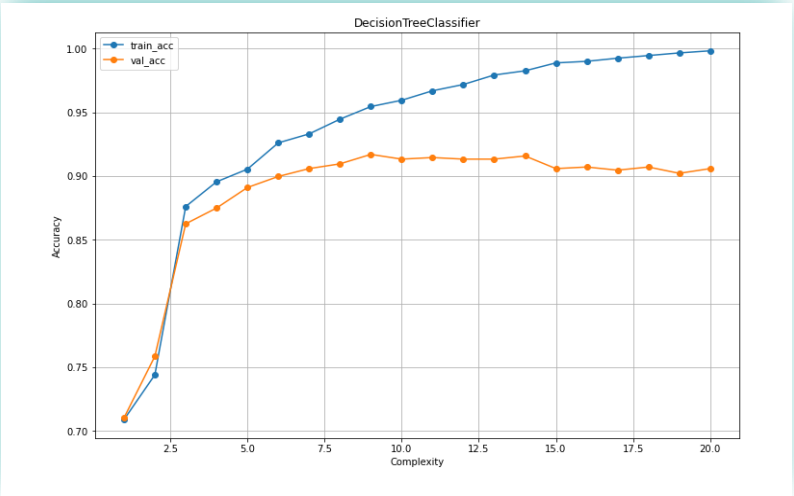
```
In [133]: data2 = data.interpolate(method='linear')
```

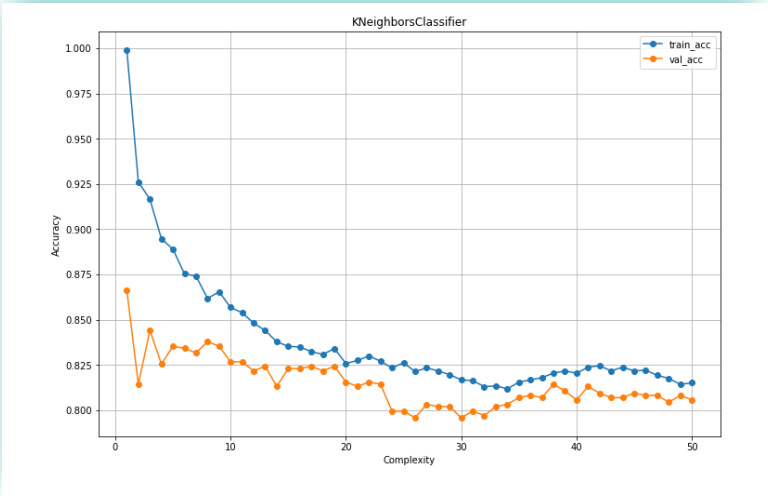
```
In [102]: data3 = data.fillna(data.median())
```

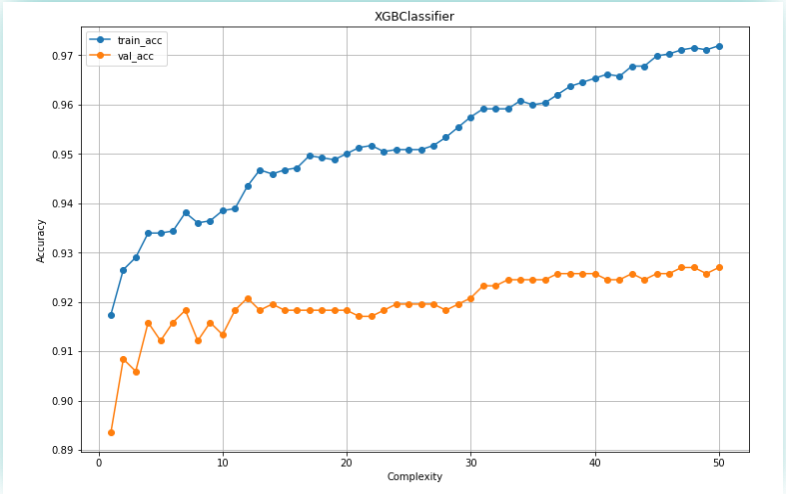
Method='linear'를 갖는 interpolate와
중앙값으로 결측치를 조치하는 fillna,
총 2가지로 나누어서 테스트를 진행하였습니다.

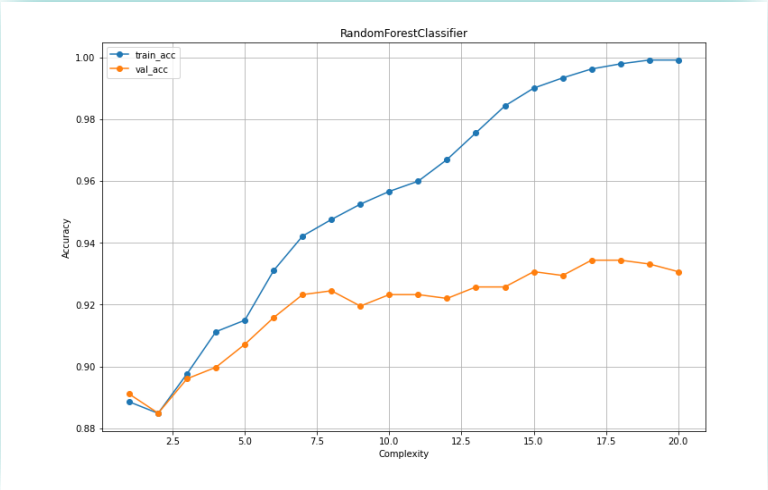
05

머신러닝 모델 선택


DecisionTreeClassifier
train_acc
val_acc
Accuracy
Complexity


KNeighborsClassifier
train_acc
val_acc
Accuracy
Complexity


XGBClassifier
train_acc
val_acc
Accuracy
Complexity


RandomForestClassifier
train_acc
val_acc
Accuracy
Complexity

05

머신러닝 모델 선택

```
: # train 및 val 데이터 정확도 확인
print(accuracy_score(y_val,pred1_1)) # 로지스틱
print(accuracy_score(y_val,pred2_1)) # RandomForest
print(accuracy_score(y_val,pred3_1)) # KNN
print(accuracy_score(y_val,pred4_1)) # DecisionTree
y_val.replace({-1:0}, inplace=True)
print(accuracy_score(y_val,pred5_2)) # XGB
y_val.replace({0:-1}, inplace=True)
```

```
0.8155940594059405
0.9257425742574258
0.6794554455445545
0.9146039603960396
0.9183168316831684
```

Thank You

감 사 합 니 다