

딥러닝 기반 핵심 산업별 빅데이터 분석

<머신러닝&딥러닝 파일럿 프로젝트>

주 제	Airbnb New User Bookings	링 크	https://www.kaggle.com/c/airbnb-recruiting-new-user-bookings/
팀 명	여행조하팀	일 자	2018년 11월 23일
팀 장	유수정 <usj0410@gmail.com>	팀 원	김하준, 김흥기

1. 과제 개요

주제	Airbnb 신규 이용자의 첫 여행지 예측하기(1~5순위예측)
프로젝트의 의미	1) 첫 여행지 예측을 통해 유저별 콘텐츠를 개인화하여 사용자가 처음 숙소를 예약하는 데 드는 시간 비용을 줄여 고객만족을 높이고 신규고객을 유치. 예약률 향상에 도움. 2) 향후 나라/도시별 수요예측을 통해 상품을 기획하고 파트너를 구축하는 데 용이.
팀과제로 선정이유	1) 이용경험이 있는 사이트의 데이터를 활용해 실제 예측모델을 만들어 보는 것에 대한 흥미. 2) 여러 피쳐값과 분류형 결과값을 가진 데이터셋으로 머신러닝 학습내용을 복습, 적용하기에 적합하다고 판단했음. 3) 팀원들의 도전의식을 불러일으킨 정제되지 않은 데이터. 데이터전처리 과정을 경험할 수 있는 기회라고 생각.

2. 데이터 설명

train데이터	기존 유저의 id, 첫 예약일, 성별, 나이, 사용한 브라우저 종류 등의 15개의 feature컬럼과 결과값 여행지 정보. Nan값이 많고 정제되지 않아 전처리 과정이 필요 (id, date account-created, timestamp first active, date first booking, gender, age, signup method, signup flow, language, affiliate channel, affiliate provider, first affiliate tracked, signup app, first device type, first browser, country destination)
Test데이터	Train데이터와 동일한 피쳐값. 결과값 제외.
Session	유저들의 로그데이터. 사이트 내에서 유저들의 action과 action지속시간 등에 대한 데이터 (user id, action, action type, action detail, device type, secs elapsed)
Age gender bucket	여행지별 성별/연령 데이터 (age bucket, country destination, gender, population in thousands, year)
countries	여행지 나라별 정보. 사용언어, 경도, 위도, 미국과의 거리 등의 정보 (country destination, lat destination, lng destination, distance km, destination km2, destination language, language levenshtein distance)

3. 과제 수행 내역

1) 데이터 정제 및 가공

1-1. train/test 데이터 정제 및 가공

Train데이터와 Test데이터 병합 후 데이터타입 변환, nan값 처리

-timeLag칼럼 추가:

$timeLag = timestamp \text{ first active} - date \text{ account-created}$ (계정 생성 후 활성화하기까지의 기간)

1-2. session 데이터 가공

A user_id	A action	A action_type	A action_detail	A device_type	A secs_elapsed
135483 unique values	show 26% index 8% Other (357) 66%	view 34% data 20% Other (8) 46%	view_search_resu... 17% p3 13% Other (153) 70%	Mac Desktop 34% Windows Desktop 25% Other (12) 41%	337661 unique values
d1mm9tcy42	lookup			Windows Desktop	319.0
d1mm9tcy42	search_results	click	view_search_results	Windows Desktop	67753.0
d1mm9tcy42	lookup			Windows Desktop	301.0
d1mm9tcy42	search_results	click	view_search_results	Windows Desktop	22141.0
d1mm9tcy42	lookup			Windows Desktop	435.0
d1mm9tcy42	search_results	click	view_search_results	Windows Desktop	7703.0
d1mm9tcy42	lookup			Windows Desktop	115.0
d1mm9tcy42	personalize	data	wishlist_content_upd ate	Windows Desktop	831.0

Figure 1 session 데이터 형태

가. Secs_elapsed_id 칼럼 추가 :

-pandas pivot table을 사용하여 id값을 multi-indexing, action을 기준으로 지속시간 (secs_elapsed)의 아이디별 총합을 구하여 train/test 데이터의 id값에 병합.

-총 접속시간이 긴 사용자일수록 NDF가 아닐 확률이 높을 것이라고 예상, 여행거리가 멀 것이라고 가정. Sklearn preprocessing 사용하여 데이터 정규화.

나. Action 횟수 칼럼 추가 :

-id를 기준으로 action의 횟수를 카운트, train/test데이터에 action_count 칼럼 추가.

-action횟수가 높을수록 실제 예약으로 이어질 가능성이 높을 것이라고 예상.

1-3. age gender bkt 데이터 가공

가. 성별 .연령별 여행지 빈도정보를 train/test데이터에 추가

기존 train/test데이터의 age칼럼을 age gender bkt의 연령 정보에 맞게 범주화 시킨 후 병합

```
from sklearn import preprocessing

min_max_scaler = preprocessing.MinMaxScaler()
np_scaled = min_max_scaler.fit_transform(pd.DataFrame(df['US']))
df['US'] = pd.DataFrame(np_scaled)
df
```

ig	first_affiliate_tracked	first_browser	first_device_type	...	AU	CA	DE	ES	FR	GB	IT	NL	PT	US
-1	untracked	Chrome	Mac Desktop	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
-1	untracked	Chrome	Mac Desktop	...	0.916295	0.842256	0.728631	0.969311	0.990157	0.894468	0.979041	0.895545	0.992857	0.875711
12	untracked	IE	Windows Desktop	...	0.737723	0.815846	0.774550	0.621813	0.924385	0.755745	0.758162	0.806452	0.814286	0.862351
18	untracked	Firefox	Mac Desktop	...	0.886161	0.869379	0.928907	0.874882	0.993736	1.000000	1.000000	0.980031	0.942857	0.918807
18	untracked	Chrome	Mac Desktop	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
12	omg	Chrome	Mac Desktop	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
15	untracked	Safari	Mac Desktop	...	0.891741	0.993576	0.972061	0.818697	1.000000	0.981702	0.956872	0.978495	0.947619	0.983796
13	omg	Safari	Mac Desktop	...	0.891741	0.993576	0.972061	0.818697	1.000000	0.981702	0.956872	0.978495	0.947619	0.983796

Figure 2 age gender bkt 피벗테이블

1-4. 결측값 처리

Age : 결측값이 약 40%정도 있을 정도로 결측치가 많아서 여러가지 방법을 시도했다.

1. drop하기
2. Regression 으로 age 추측. 상관관계가 높은 피쳐값이 나오지 않아 포기
3. 기술통계표를 사용하여 정규분포 형식으로 랜덤으로 지정
4. -1 로 missing value 처리 (xgboost 내부 결측값 처리 방법)

Session(secs_elapsed):

1. drop하기
2. -1 로 missing value 처리

4. 결과 도출 과정

0) 머신머닝 모델 결정

수업시간에 배운 NN기반 딥러닝 Softmax코드에 Lelu, Xavier initializer, Adam optimizer 등을 사용해 보았으나 29.25 퍼센트에서 정확도가 올라가지 않음

Figure 4 softmax 결과정확도 : 29.25%

```
sess = tf.Session()
sess.run(tf.global_variables_initializer())

for step in range(2000):
    feed_dict = {X: x_train, Y: y_train}
    sess.run(optimizer, feed_dict=feed_dict)
    if step % 10 == 0:
        loss, acc = sess.run([cost, accuracy], feed_dict={X: x_train, Y: y_train})
        print("Step: {:5}, Loss: {:.3f}, Acc: {:.2%}".format(step, loss, acc))

Step: 250, Loss: 2.325, Acc: 29.25%
Step: 270, Loss: 2.325, Acc: 29.25%
Step: 290, Loss: 2.325, Acc: 29.25%
Step: 310, Loss: 2.325, Acc: 29.25%
Step: 330, Loss: 2.325, Acc: 29.25%
Step: 350, Loss: 2.325, Acc: 29.25%
Step: 370, Loss: 2.325, Acc: 29.25%
Step: 390, Loss: 2.325, Acc: 29.25%
Step: 410, Loss: 2.325, Acc: 29.27%
Step: 430, Loss: 2.325, Acc: 29.25%
Step: 450, Loss: 2.325, Acc: 29.27%
```

Figure 3 randomforest 결과정확도 : 57.35%

```
#RandomForest

from sklearn.ensemble import RandomForestClassifier
import pandas as pd
import numpy as np
np.random.seed(0)

# Create a random forest Classifier. By convention, clf means 'Classifier'
x_train, x_test, y_train, y_test = train_test_split(X_train, Y_train, test_size=0.33, random_state=7)
clf = RandomForestClassifier(n_jobs=100, random_state=0)
training_start = time.perf_counter()
clf.fit(x_train, y_train)
training_end = time.perf_counter()
prediction_start = time.perf_counter()
preds = clf.predict(x_test)
prediction_end = time.perf_counter()
nd_y_test = y_test.values.reshape(len(y_test.values))
acc_xgb = (preds == nd_y_test).sum().astype(float) / len(preds)*100
xgb_train_time = training_end - training_start
xgb_prediction_time = prediction_end - prediction_start
print("RF's prediction accuracy is: %.2f" % (acc_xgb))
print("Time consumed for training: %.3f" % (xgb_train_time))
print("Time consumed for prediction: %.5f seconds" % (xgb_prediction_time))

C:\Python\Anaconda3-5.2.0\lib\site-packages\ipykernel_launcher.py:5: DataConversionWarning: A column-vector
expected. Please change the shape of y to (n_samples,), for example using ravel().

RF's prediction accuracy is: 57.35
Time consumed for training: 2.051
Time consumed for prediction: 0.25475 seconds
```

➔ Xgboost 사용

-최고 70% 까지 정확도. Xgboost 모델 사용 결정

-값은 타겟 0~9 중에 NDF, US 인 0,1 만 나옴. 피쳐 값 설정에 따라 한 값으로만 수렴.

➔ 새로운 문제 인식. Kaggle Evaluation을 다시 확인한 결과, 결과값 여행지를 5순위까지 구하는 문제였음을 확인한 후 머신러닝 모델을 수정함.

```
import xgboost as xgb
import time
import tensorflow as tf
import random
from sklearn.model_selection import train_test_split
import tensorflow as tf

x_train, x_test, y_train, y_test = train_test_split(X_train, Y_train, test_size=0.33, random_state=7)

dtrain = xgb.DMatrix(x_train, y_train)
dtest = xgb.DMatrix(x_test)

param = {
    'max_depth': 9,
    'learning_rate': 0.01,
    'n_estimators': 5,
    'objective': 'multi:softmax',
    'num_class': 12,
    'gamma': 0,
    'min_child_weight': 6,
    'max_delta_step': 0,
    'subsample': 0.8,
    'colsample_bytree': 0.8,
    'colsample_bylevel': 1,
    'reg_alpha': 0.01,
    'reg_lambda': 1,
    'scale_pos_weight': 1,
    'base_score': 0.5,
    'missing': None,
    'silent': True,
    'nthread': 4,
    'seed': 27}

training_start = time.perf_counter()
bst = xgb.train(param, dtrain)
training_end = time.perf_counter()
# make prediction
prediction_start = time.perf_counter()
preds = bst.predict(dtest)
prediction_end = time.perf_counter()
nd_y_test = y_test.values.reshape(len(y_test.values))
acc_xgb = (preds == nd_y_test).sum().astype(float) / len(preds)*100
xgb_train_time = training_end - training_start
xgb_prediction_time = prediction_end - prediction_start
print("XGBoost's prediction accuracy is: %3.2f" % (acc_xgb))
print("Time consumed for training: %4.3f" % (xgb_train_time))
print("Time consumed for prediction: %5.5f seconds" % (xgb_prediction_time))

XGBoost's prediction accuracy is: 63.42
Time consumed for training: 84.235
Time consumed for prediction: 1.06651 seconds
```

-다양한 조합, Feature Drop, Nan 값 처리(-1)를 하여 63.42%의 validation 정확도를 얻음.

```

import xgboost as xgb
import time
import tensorflow as tf
import random
from sklearn.model_selection import train_test_split
import tensorflow as tf

x_train, x_test, y_train, y_test = train_test_split(X_train, Y_train, test_size=0.33, random_state=7)

dtrain = xgb.DMatrix(x_train, y_train)
dtest = xgb.DMatrix(x_test)

param = {
    'max_depth': 9,
    'learning_rate': 0.01,
    'n_estimators': 5,
    'objective': 'multi:softprob',
    'num_class': 12,
    'gamma': 0,
    'min_child_weight': 6,
    'max_delta_step': 0,
    'subsample': 0.8,
    'colsample_bytree': 0.8,
    'colsample_bylevel': 1,
    'reg_alpha': 0.01,
    'reg_lambda': 1,
    'scale_pos_weight': 1,
    'base_score': 0.5,
    'missing': None,
    'silent': True,
    'nthread': 4,
    'seed': 27}

bst = xgb.train(param, dtrain)

```

```

X_test=xgb.DMatrix(X_test) #Dmatrix 변환 오류시 주석처리

y_pred = bst.predict(X_test)
#Taking the 5 classes with highest probabilities
ids = [] #list of ids
cts = [] #list of countries
for i in range(len(id_test)):
    idx = id_test[i]
    ids += [idx] * 5
    cts += np.argsort(y_pred[i])[::-1][:5].tolist()


#generate submission
country_label = pd.DataFrame(np.column_stack((ids, cts)), columns=['id', 'country'])
result=pd.DataFrame(country_label['country'])
country_label.drop(['country'], axis=1, inplace=True)
result=result['country'].astype(int)
inv_mapping= {0:'NDF', 1:'US', 2:'other', 3:'FR', 4:'CA', 5:'GB', 6:'ES', 7:'IT', 8:'PT', 9:'NL', 10:'DE', 11:'AU'}
result=result.map(inv_mapping)
country_label['country']=result
country_label.to_csv('Submission.csv', encoding='utf-8')

```

Figure 5 제출파일 생성

다양한 피쳐값 조합을 거쳐 가장 점수가 높았던 0.86694 score의 파일을 최종 결과물로 도출.

Figure 6 Kaggle에 제출한 submission파일

All	Successful	Selected		
Submission and Description		Private Score	Public Score	Use for Final Score
Submission.csv 2 minutes ago by Honggi Kim XGBoost 61.53		0.79188	0.79263	<input type="checkbox"/>
Submission.csv 17 minutes ago by Honggi Kim XGBoost 63.42		0.86694	0.86363	<input type="checkbox"/>
Submission.csv 40 minutes ago by Honggi Kim XGBoost 0.7723		0.85497	0.85172	<input type="checkbox"/>
Submission.csv 41 minutes ago by Honggi Kim XGBoost 0.7723			Error 	<input type="checkbox"/>
sub2.csv 2 hours ago by Honggi Kim Top 5		0.86747	0.86354	<input type="checkbox"/>

<최종 사용한 피쳐값>

```
Int64Index: 275547 entries, 0 to 213450
Data columns (total 31 columns):
affiliate_channel      275547 non-null object
affiliate_provider     275547 non-null object
age                   275547 non-null float64
country_destination   275547 non-null object
first_affiliate_tracked 275547 non-null object
first_browser         275547 non-null object
first_device_type     275547 non-null object
gender               275547 non-null object
id                   275547 non-null object
language              275547 non-null object
signup_app            275547 non-null object
signup_flow           275547 non-null int64
signup_method         275547 non-null object
timelag_nu            275547 non-null int64
secs_elapsed          275547 non-null float64
AU                   275547 non-null float64
CA                   275547 non-null float64
DE                   275547 non-null float64
ES                   275547 non-null float64
FR                   275547 non-null float64
GB                   275547 non-null float64
IT                   275547 non-null float64
NL                   275547 non-null float64
PT                   275547 non-null float64
US                   275547 non-null float64
dac_year              275547 non-null object
dac_month              275547 non-null object
dac_day               275547 non-null object
tfa_year              275547 non-null object
tfa_month              275547 non-null object
tfa_day               275547 non-null object
```

Figure 7 최종 submission file

	A	B
1	id	country
2	5uwns89z	NDF
3	5uwns89z	US
4	5uwns89z	FR
5	5uwns89z	other
6	5uwns89z	GB
7	jtl0dijy2j	NDF
8	jtl0dijy2j	US
9	jtl0dijy2j	other
10	jtl0dijy2j	FR
11	jtl0dijy2j	AU
12	xx0ulgorjt	NDF
13	xx0ulgorjt	US
14	xx0ulgorjt	FR
15	xx0ulgorjt	other
16	xx0ulgorjt	GB
17	6c6puo6ix	NDF

5. 프로젝트를 진행하며 아쉬웠던 점

- 프로젝트에서 정확히 요구하는 바를 철저하게 확인하지 못한 점.
- 가공된 데이터 전달 과정에서 효율적으로 데이터 관리가 이루어지지 않은 점.
- 여행사 사이트에 대한 도메인 지식 부족으로 유의미한 Feature 값들을 예상하고 이에 맞게 가공하는 작업에서 어려움을 겪음.

감사합니다