TUGAS BESAR PEMBELAJARAN MESIN LANJUT

Laporan

Dibuat untuk memenuhi tugas mata kuliah Pembelajaran Mesin Lanjut

oleh :

Meyzo Naufal R (1301184299)

Alif Ranadian Nadifah (1301184255)

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

UNIVERSITAS TELKOM

BANDUNG

2021

1. Formulasi Masalah

TPOT adalah library python yang merupakaan tools untuk *Automated Machine Learning* yang mengoptimalkan alur pembelajaran mesin menggunakan pemrograman genetik. Dengan menggunakan dataset weatherAus.csv kami akan menyelesaikan masalah yaitu membuat prediksi apakah hari esok akan turun hujan atau tidak dengan menggunakan metode automated machine learning (AutoML). Kami mengharapkan nantinya untuk menghasilkan model yang dapat memprediksi dengan semaksimal mungkin (diatas 80%).

2. Eksplorasi dan Persiapan Data

Eksplorasi dan Persiapan data adalah proses pre-processing terhadap data agar dataset menghasilkan output yang lebih baik. Pre-Processing yang digunakan dalam AutomatedMachine Learning adalah :

A. Drop data duplikat

Data yang memiliki nilai/value sama akan dihapus karena data tersebut tidak akan berdampak ke hasil pemodelan sehingga akan lebih baik jika dihapus untuk mempercepat proses pemodelan.

Code & Output:

```
# Drop data duplikat
df_drop = df.copy()
df_new = df_drop.drop_duplicates()
df_new.reset_index(drop=True, inplace=True)
df_new
```

1 to 25 of 20000 entries Filter

index	Date	Location	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustDir	WindGustSpeed	WindDir9am	WindDir3pm	WindSpeed9am	WindSpeed3pm
0	2008-12-01	Albury	13.4	22.9	0.6	NaN	NaN	W	44.0	W	WNW	20.0	24.
1	2008-12-02	Albury	7.4	25.1	0.0	NaN	NaN	WNW	44.0	NNW	WSW	4.0	22.
2	2008-12-03	Albury	12.9	25.7	0.0	NaN	NaN	WSW	46.0	W	WSW	19.0	26.
3	2008-12-04	Albury	9.2	28.0	0.0	NaN	NaN	NE	24.0	SE	E	11.0	9.
4	2008-12-05	Albury	17.5	32.3	1.0	NaN	NaN	W	41.0	ENE	NW	7.0	20.
5	2008-12-06	Albury	14.6	29.7	0.2	NaN	NaN	WNW	56.0	W	W	19.0	24.
6	2008-12-07	Albury	14.3	25.0	0.0	NaN	NaN	W	50.0	SW	W	20.0	24.
7	2008-12-08	Albury	7.7	26.7	0.0	NaN	NaN	W	35.0	SSE	W	6.0	17.
8	2008-12-09	Albury	9.7	31.9	0.0	NaN	NaN	NNW	80.0	SE	NW	7.0	28.
9	2008-12-10	Albury	13.1	30.1	1.4	NaN	NaN	W	28.0	S	SSE	15.0	11.

B. Drop Data Null

Data yang memiliki nilai null akan dihapus karena data tersebut dapat membuat hasil pemodelan menjadi kurang bagus yaitu penurunan akurasi.

Code & Output:

```
# Drop data yang memiliki nilai null
df_new = df_drop.dropna()
df_new.reset_index(drop=True, inplace=True)
df_new
```

1 to 25 of 4587 entries Filter

index	Date	Location	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustDir	WindGustSpeed	WindDir9am	WindDir3pm	WindSpeed9am	WindSpeed3pm
0	2009-01-01	Cobar	17.9	35.2	0.0	12.0	12.3	SSW	48.0	ENE	SW	6.0	20.
1	2009-01-02	Cobar	18.4	28.9	0.0	14.8	13.0	S	37.0	SSE	SSE	19.0	19.
2	2009-01-04	Cobar	19.4	37.6	0.0	10.8	10.6	NNE	46.0	NNE	NNW	30.0	15.
3	2009-01-05	Cobar	21.9	38.4	0.0	11.4	12.2	WNW	31.0	WNW	WSW	6.0	6.
4	2009-01-06	Cobar	24.2	41.0	0.0	11.2	8.4	WNW	35.0	NW	WNW	17.0	13.
5	2009-01-07	Cobar	27.1	36.1	0.0	13.0	0.0	N	43.0	N	WNW	7.0	20.
6	2009-01-08	Cobar	23.3	34.0	0.0	9.8	12.6	SSW	41.0	S	SSE	17.0	19.
7	2009-01-09	Cobar	16.1	34.2	0.0	14.6	13.2	SE	37.0	SE	S	15.0	6.
8	2009-01-10	Cobar	19.0	35.5	0.0	12.0	12.3	ENE	48.0	ENE	WSW	30.0	9.
9	2009-01-11	Cobar	19.7	35.5	0.0	11.0	12.7	NE	41.0	NNE	WSW	15.0	17.
10	2009-01-12	Cobar	20.9	37.8	0.0	12.8	13.2	E	30.0	SE	ENE	11.0	7.

C. Drop Outliers

Data outlier adalah data yang memiliki persebaran data yang sangat jauh dari rata-rata data yang ada namun jumlahnya hanya sedikit. Data-data tersebut akan dihapus agar hasil pemodelan menjadi lebih baik.

Code & output :

```
[25] # Menampilkan jumlah data dengan outliers
     print("jumlah data dengan outliers:", df_drop.shape)

     # Drop outliers
     df_new = df_new[(np.abs(stats.zscore(df_new.select_dtypes(include=np.number))) < 3).all(axis=1)]
     print("jumlah data tanpa outliers:", df_new.shape)
     df_new.reset_index(drop=True, inplace=True)
     df_new
```

```
jumlah data dengan outliers: (0, 24)
jumlah data tanpa outliers: (4354, 24)
```

1 to 25 of 4354 entries Filter

index	Date	Location	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustDir	WindGustSpeed	WindDir9am	WindDir3pm	WindSpeed9am	WindSpeed3pm
0	2009-01-01	Cobar	17.9	35.2	0.0	12.0	12.3	SSW	48.0	ENE	SW	6.0	20.
1	2009-01-02	Cobar	18.4	28.9	0.0	14.8	13.0	S	37.0	SSE	SSE	19.0	19.
2	2009-01-04	Cobar	19.4	37.6	0.0	10.8	10.6	NNE	46.0	NNE	NNW	30.0	15.
3	2009-01-05	Cobar	21.9	38.4	0.0	11.4	12.2	WNW	31.0	WNW	WSW	6.0	6.
4	2009-01-06	Cobar	24.2	41.0	0.0	11.2	8.4	WNW	35.0	NW	WNW	17.0	13.
5	2009-01-07	Cobar	27.1	36.1	0.0	13.0	0.0	N	43.0	N	WNW	7.0	20.

D. Split Data & Normalisasi

Pada tahap ini data akan dilakukan normalisasi menggunakan MinMaxScaler setelah itu data dibagi menjadi dua yaitu data X dan y, dimana data X mewakili sekumpulan data non-label dan data y mewakili data label. Kedua data ini kemudian akan dibagi lagi menjadi dua, yaitu berupa data train dan data test. Besar masing-masing pada data train dan test sendiri adalah 70% dan 30%.

Code & Output:

```
[38] # dilakukan normalisasi dengan menggunakan minmaxscaler
     df['RainTomorrow_label'] = df['RainTomorrow'].apply(lambda x: 1 if x=='Yes' else 0)
     X = df[fitur_numerikal]
     y = df.iloc[:,-1:]
     mm = MinMaxScaler()
     X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30)

     print('Data X_train:')
     display(X_train)

     print('\nData X_test:')
     display(X_test)
```

Data X_train:

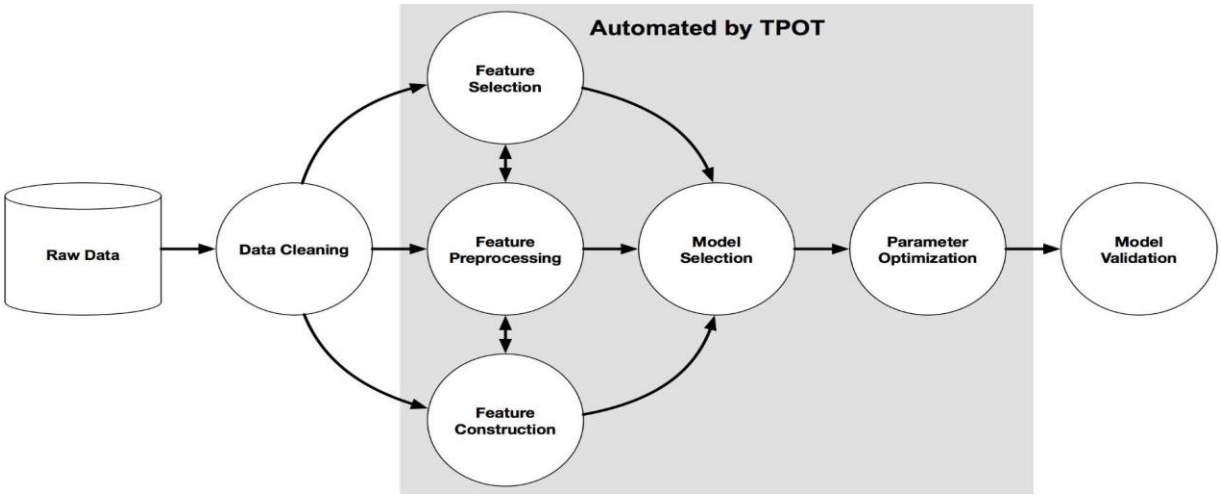
	Sunshine	Humidity	Pressure	Cloud
1360	NaN	72.5	1015.00	8.0
6289	1.4	30.0	1004.90	7.0
9940	9.2	68.0	1023.90	2.5
3804	NaN	53.5	1012.45	NaN
1	NaN	34.5	1009.20	NaN

Data X_test:

	Sunshine	Humidity	Pressure	Cloud
8986	NaN	22.0	1018.80	NaN
18950	NaN	60.5	1028.15	NaN
20217	NaN	79.5	1020.20	NaN
5395	NaN	55.0	NaN	NaN
13159	0.0	90.0	1010.00	8.0

3. Pemodelan

Dalam Automated Machine Learning, terdapat banyak metode yang dapat digunakana untuk pemodelan. Salah satunya adalah menggunakan TPOT. Gambar di bawah adalah carakerja dari Automated Machine Learning by TPOT.



Code & Output:

```
# dilatih menggunakan algoritma Tpot
pipeline_optimizer = TPOTClassifier(generations=5, population_size=20, cv=5,
                                    random_state=42, verbosity=2)
pipeline_optimizer.fit(mm.fit_transform(X_train), y_train)
```

```
Imputing missing values in feature set
/usr/local/lib/python3.7/dist-packages/sklearn/utils/validation.py:985: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Generation 1 - Current best internal CV score: 0.8339204174820611

Generation 2 - Current best internal CV score: 0.8339204174820611

Generation 3 - Current best internal CV score: 0.8339204174820611

Generation 4 - Current best internal CV score: 0.8339204174820611

Generation 5 - Current best internal CV score: 0.8339204174820611

Best pipeline: XGBClassifier(input_matrix, learning_rate=0.001, max_depth=9, min_child_weight=7, n_estimators=100, n_jobs=1, subsample=0.45, ve
TPOTClassifier(generations=5, population_size=20, random_state=42, verbosity=2)
```

4. Evaluasi

Code & Output :

```
[31] print('Classification Report Pipeline')
     print(classification_report(y_test, y_pred))

     print('Confusion Matrix')
     print(confusion_matrix(y_test, y_pred))
```

```
Classification Report Pipeline
              precision    recall  f1-score   support

           0       0.85      0.95      0.90      5142
           1       0.69      0.39      0.50      1428

    accuracy                           0.83      6570
   macro avg       0.77      0.67      0.70      6570
weighted avg       0.81      0.83      0.81      6570

Confusion Matrix
[[4892  250]
 [ 870  558]]
```

5. Kesimpulan

Ada banyak macam tools yang dapat digunakan untuk menjalankan Auto Machine Learning. Salah satunya adalah TPOT. Dengan menggunakan TPOT, user sangat dimanjakan dalam membuat suatu machine learning. Ini dikarenakan user hanya perlu melakukan tahap “data cleaning” dan tahap “evaluation” sedangkan TPOT menjalankan sisa tahap yang diperlukan secara otomatis. Hasil yang diperoleh dari pemodelan menggunakan TPOT sudah lumayan bagus dengan memiliki nilai rata-rata F1-Score sebesar 70% dan accuracy sebesar 83%.