Laporan Tugas Besar Machine Learning

1301174064

Muhammad Irvan Tantowi

Program Studi S1 Informatika

Fakultas Informatika

Universitas Telkom

Bandung

2020

A. Formulasi Masalah

- Artikulasi Masalah

  Terdapat sebuah dataset yang harus diselesaikan menggunakan metode Classification dan Clustering.

- Identifikasi Sumber Data

  Dataset yang digunakan adalah air_bnb.csv yang didalamnya terdapat 22552 record dan 14 atibut.

- Identifikasi Potensi Masalah

  Di dalam dataset tersebut masih banyak data yang rusak (ada nilai yang kosong) , sehingga dataset ini masih harus dilakukan pembersihan agar mempermudah dalam proses Classification dan Clusterng.

- Potensi Bias dan Etika

  Dalam dataset ini tidak mengandung unsur sara yang dapat melanggar etika atau mengganggu suatu golongan

B. Eksplorasi Data

- Pada tahap pertama saya mengimpor datset lebih dahulu menggunakan libary pandas yaitu pd.read_csv("air_bnb.csv").
- Langkah selanjutnya saya melakukan cek missing value pada data, apakah terdapat data yang hilang pada data tersebut, sehingga tidak menyulitkan dalam proses selanjutnya.

Gambar 1.Cek missing value

	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month
count	22552.000000	22552.000000	22552.000000	22552.000000	22552.000000	18638.000000
mean	52.509824	13.406107	67.143668	7.157059	17.840679	1.135525
std	0.030825	0.057964	220.266210	40.665073	36.769624	1.507082
min	52.345803	13.103557	0.000000	1.000000	0.000000	0.010000
25%	52.489065	13.375411	30.000000	2.000000	1.000000	0.180000
50%	52.509079	13.416779	45.000000	2.000000	5.000000	0.540000
75%	52.532669	13.439259	70.000000	4.000000	16.000000	1.500000
max	52.651670	13.757642	9000.000000	5000.000000	498.000000	36.670000

Gambar 2. Hasil cek missing value

- Setelah di cek ternyata terdapat misiing value, lalu kita mencari dimana letek missing value tersebut

```
np.where(np.isnan(data["reviews_per_month"]))
```

Gambar 3. Posisi missing value

- Setelah mengetahui lokasi missing value selanjutnya mengisi missing value tersebut dengan nilai 0

```
data["reviews_per_month"] = data["reviews_per_month"].fillna(0)
```

Gambar 4. Missing value diganti dengan nilai 0

- Lalu saya melihat korelasi antar data, data mana saja yang memiliki korelasi yang bagus yang dapat membantu dalam proses klasifikasi dan clustering.

```
corr = data.select_dtypes(include = ['float64', 'int64']).iloc[:, :].corr()
plt.figure(figsize=(10,10))
ax = sns.heatmap(corr, vmax=1, square=True)
plt.xticks(rotation=45)
plt.yticks(rotation=45)
```

Gambar 5. Melihat korelasi

- Lalu memilih fitur yang memiliki korelasi yang bagus

```
list_feature = ['number_of_reviews','reviews_per_month']
data_feature = data[list_feature]
```

Gambar 6. Memilih korelasi

- Lalu dilakukan split data

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(data_feature, data["room_type"], test_size=0.20,
random_state=100)
```

Gambar 7. Split data

C. Classification

- Pemodelan

  Dalam proses classification ini terdapat 2 model yang digunakan yaitu Naïve Bayes dan Decision Tree, saya memilih kedua metode tersebut karenan mudah diimplementasikan.

```
from sklearn.naive_bayes import MultinomialNB
clf = MultinomialNB()
clf.fit(X_train, y_train)
```

Gambar 8. Naive Bayes

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

Gambar 9. Decision Tree

- Eksperimen

  Dalam proses klasifikasi dilakukan percobaan dengan feature "number_of_reviews" dan "reviews_per_month"

```
result = clf.predict(X_test)
```

Gambar 10. Eksperimen naive bayes

```
hasil2 = clf.predict(X_test)
```

Gambar 11. Eksperimen decision tree

- Evaluasi

  Perhitungan akurasi menggunakan library sklearn.matrics

```
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
print (classification_report(y_test, result))
```

```
                  precision    recall  f1-score   support

Entire home/apt       0.55      0.20      0.29      2094
   Private room       0.54      0.86      0.66      2353
    Shared room       0.00      0.00      0.00        64

       accuracy                           0.54      4511
      macro avg       0.36      0.35      0.32      4511
   weighted avg       0.53      0.54      0.48      4511
```

Gambar 12. Akurasi naive bayes

```
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
print (classification_report(y_test, hasil2))
```

	precision	recall	f1-score	support
Entire home/apt	0.48	0.58	0.52	2094
Private room	0.53	0.44	0.48	2353
Shared room	0.00	0.00	0.00	64
accuracy			0.50	4511
macro avg	0.34	0.34	0.34	4511
weighted avg	0.50	0.50	0.49	4511

Gambar 13. Akurasi decision tree

- Kesimpulan

  Dari proses klasifikasi yang menggunakan dua buah metode, didapatkan akurasi 54% dengan metode Naïve Bayes dan 50% menggunakan metode Decision Tree.

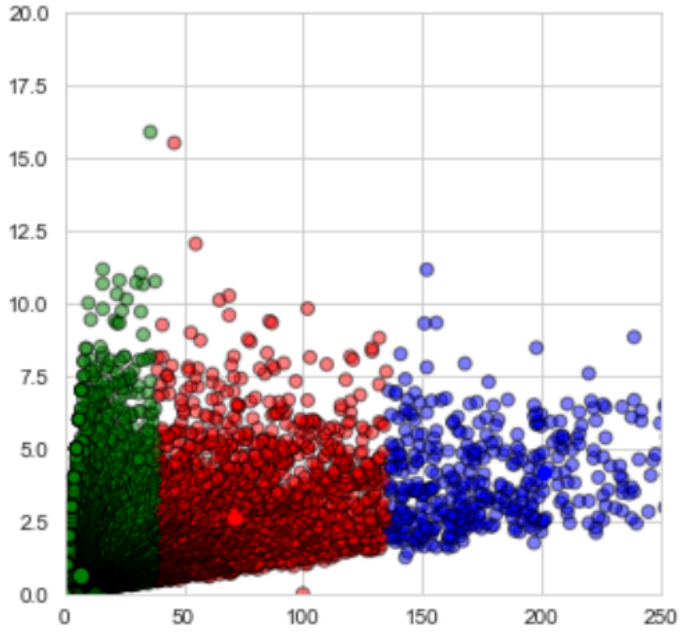
D. Clustering

- Pemodelan

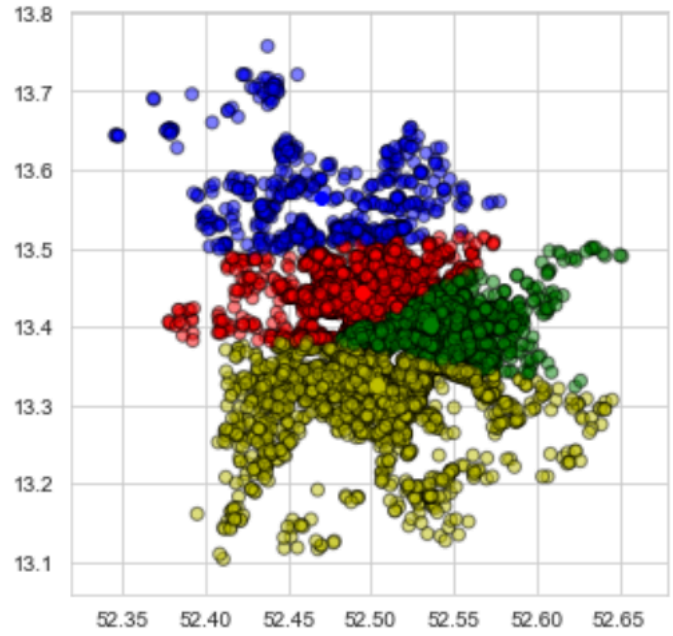
  Metode clustering yang digunakan adalah K-means. Alasan menggunakan metode ini karena setelah saya membaca beberapa sumber, K-means merupakan metode yang relatif mudah untuk di implementasikan dan dijalankan. Selain itu waktu yang digunakan untuk mengeksekusi program pun bisa dibilang cepat.

- Eksperimen

  Pada clustering kali ini digunakan metode K-means dan kali ini saya mencoba menggunakan dua buah feature yang berbeda antara lain “number_of_reviews, reviews_per_month” dan “latitude, longitude”. Dari proses clustering tadi didapatkan dua hasil visualisasi yang berbeda di bawah ini merupakan hasil visualisasi tersebut.



Gambar 14. Visualisasi "number_of_reviews & reviews_per_month"



Gambar 15. Visualisasi "latitude & longitude"

- Evaluasi

  Pada clustering kali ini saya belum bisa melakukan evaluasi pada proses clustering saya karenakan mungkin kekurangan saya. Mohon maaf.

- Kesimpulan

  Dari semua proses yang dijalankan, dapat dikatakan bahwa kamen relative lebih mudah untuk di main bersikan. Waktu yang digunakan untuk menjalankan algoritma tersebut tidak lama. Namun karena titik sentral merupakan random, terkadang hasil berbeda dengan program berikutnya. Untuk improv man ke depannya, lebih emang persiapkan lagi pada proses expiration karena data tersebut berpengaruh pada hasil clustering dan tingkat keaakuratan clustering.