CLASSIFICATION TERHADAP DATASET SALJU MENGGUNAKAN MODEL ALGORITMA KNN, RANDOM FOREST, DAN DECISION TREE

Ahmad Farhan A (1301171751)
Husnul Aminindya Maheswari (1301174632)
IF-42-02

1. Teknik Eksplorasi dan Pre-Processing Data
    a. Data Eksplorasi

       Pertama, dataset yang akan digunakan untuk pembangunan modelling ini adalah '**salju_train.csv**'. Pada tahap data explorasi, kami hanya mengeksplorasi banyak baris dan atribut. Dataset tersebut mengandung 109095 baris dan 24 atribut.
    b. Missing Values

       Setelah mengeksplorasi data, tahap selanjutnya ada mencari missing values. Dataset tersebut mengandung missing values yang cukup banyak, baik dari data numerik maupun data kategorik. Untuk itu, akan dilakukannya 2 cara untuk mengatasi 2 tipe data tersebut. Pertama, untuk data bersifat numerik, kami mencari tahu data skewness. Skewness merupakan suatu teknik yang bisa menentukan apakah dataset tersebut harus menggunakan mean atau median. Hasil tersebut bisa ditentukan jika berada di rentang -2 dan 2. Hasil yang kami peroleh adalah karena ada beberapa atribut yang mempunyai rentang diatas 2, maka kami menggunakan median. Teknik selanjutnya ada dengan menggunakan library **SimpleImputer**. Teknik tersebut digunakan untuk menutup missing values data kategorik.
    c. Encoding data numerik dan kategorik

       Supaya semua atribut bisa diproses dengan baik oleh algoritma-algoritma klasifikasi, library **category_encoders** dipakai. Library tersebut digunakan karena **category_encoders** dapat mengubah variabel kategorik menjadi numerik dengan beberapa teknik.
2. Algoritma KNN
    a. Teori

Algoritma k-Nearest Neighbour (kNN) merupakan algoritma supervised learning yang menggunakan metode klasifikasi objek berdasarkan jarak terdekat di antara data pembelajaran (training data).

3. Hasil dan Analisis

```
Akurasi Model: 75.0497 %

Akurasi Traing set: 80.3460 %
```

Pada gambar tersebut, bahwa akurasi model dan akurasi training set memiliki persentase masing-masing 75,09% dan 80,34%. Namun, dengan persentase yang tinggi, kNN belum memuaskan untuk dijadikan sebagai akurasi terbaik. Hal ini dikarenakan pada kNN yang tidak memiliki kinerja yang yang baik terhadap dataset yang berjumlah banyak.

Lalu, berikut adalah hasil confusion matrix bersama dengan precision, recall, f-1 score, dan support.

```
----Confusion Matrix-----
[[24070  1476]
 [ 6706   477]]

----Precision, Recall, F1-Score, Support-----
              precision    recall  f1-score   support

       Tidak       0.78      0.94      0.85     25546
          Ya       0.24      0.07      0.10      7183

    accuracy                           0.75     32729
   macro avg       0.51      0.50      0.48     32729
weighted avg       0.66      0.75      0.69     32729
```

4. Eksperimen

Untuk menutup kekurangan yang ada di algoritma kNN, kami menganalisis 2 algoritma klasifikasi lagi yaitu Random Forest dan Decision Tree.

a. Random Forest

```
Akurasi Model: 81.8448 %
Akurasi Traing set: 99.9961 %
```

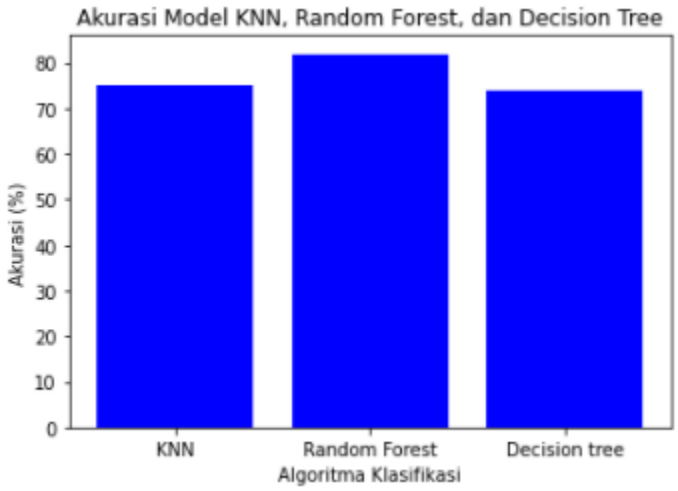
Gambar diatas menunjukkan hasil akurasi model random forest dengan 81.8% dan akurasi training set nya dengan 99.99%. Dapat dilihat bahwa hasil dari algoritma Random Forest lebih baik dari hasil algoritma kNN.

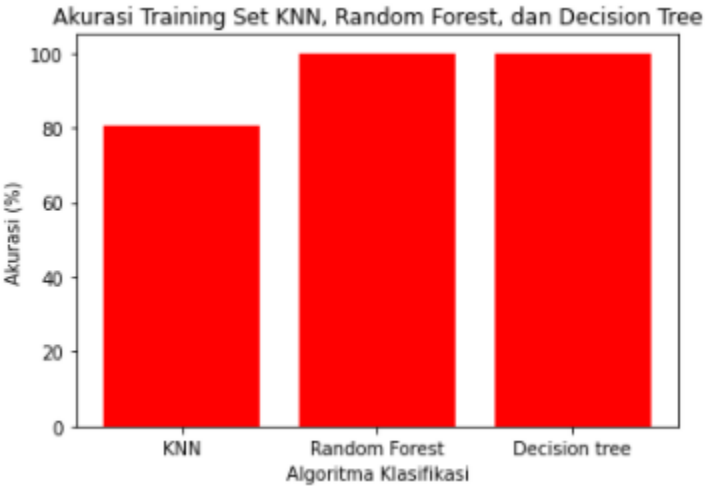
b. Decision Tree

```
Akurasi Model: 73.9405 %
Akurasi Traing set: 100.0000 %
```

Pada gambar diatas, hasil akurasi model dan training set adalah 73,9% dan 100%. Berdasarkan data tersebut, bahwa Decision tree memiliki training set yang lebih baik dari random forest, namun kurang dalam akurasi model.

Berikut adalah graf perbandingan antara ke-3 algoritma tersebut.





Berdasarkan graf diatas, dapat dilihat bahwa algoritma Random Forest tidak mempunyai perubahan yang drastis di keu2 graf tersebut. Ini menunjukkan tingkat akurasi yang dipunyai oleh Random Forest cukup tinggi dan stabil bagi dataset ini.

5. Kesimpulan

    Berdasarkan eksperimen dengan 3 algoritma diatas, bisa disimpulkan bahwa algoritma Decision Tree merupakan algoritma yang mempunyai akurasi yang paling memuaskan.Hal ini dapat dibuktikan dari perbandingan graph di atas. Meskipun akurasi training set dari algoritma Decision Tree adalah 100%, algoritma Random Forest mempunyai kinerja yang tinggi untuk dataset yang berjumlah banyak dan dapat mempertahankan tingkat akurasinya.

    Lalu, meskipun akurasi training set dari algoritma Decision Tree adalah 100%, akurasi model yang dicapai adalah 74.5%. Ini membuktikan bahwa selain KNN, Decision Tree tidak bisa berkinerja tinggi dengan dataset yang berjumlah banyak.

    Yang juga menjadi faktor dalam hasil akurasi adalah penginputan data, berapa kali iterasi, dan penentuan model dan teknik-teknik di tahap pre-processing. Pada library **SimpleImputer**, strategy yang di implementasi adalah ‘most_frequent’ yang berarti mengisi missing value dengan suatu variabel yang paling banyak muncul di dataset. Kemungkinan besar bahwa hasil **SimpleImputer** tersebut menjadi salah satu faktor dalam hasil akurasi.

LINK VIDEO PRESENTASI: Here