HELLO

LAPORAN PRAKTIKUM TEXT MAINING

Klasifikasi

Dosen Pengampu : Tita Karlita S.Kom., M.Kom.

Bayu Kurniawan / 3322600019

Start

KLASIFIKASI

Klasifikasi dan regresi merupakan dua konsep utama dalam Machine Learning. Klasifikasi digunakan ketika output yang diinginkan adalah dalam bentuk kategori diskrit, sementara regresi digunakan ketika output yang diharapkan adalah nilai kontinu. Dalam klasifikasi, model berupaya mengelompokkan data ke dalam kategori yang telah ditentukan sebelumnya, seperti membedakan email spam dan bukan spam. Sementara itu, regresi berfokus pada memodelkan hubungan antara variabel input dan output secara terus menerus, misalnya, dalam memprediksi harga rumah atau suhu berdasarkan variabel tertentu.

Back to Agenda

-ASSIGNMENT 2 Text-Classification using Logistic Regression

```
import numpy as np
import pandas as pd
import os
import re
import matplotlib.pyplot as plt
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.pipeline import Pipeline
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest, chi2
from sqlite3 import Error
from sklearn.ensemble import RandomForestClassifier
import sqlite3
import pickle
import nltk
```

- numpy untuk operasi numerik.
- pandas untuk manipulasi data terstruktur.
- matplotlib.pyplot untuk visualisasi data.
- nltk untuk pemrosesan bahasa alami, seperti penghapusan kata-kata umum (stopwords) dan stemming.

-ASSIGNMENT 2 Text-Classification using Logistic Regression

```
dataset = pd.read_csv('bbc-text.csv')
dataset.head()
```

✓ 0.1s

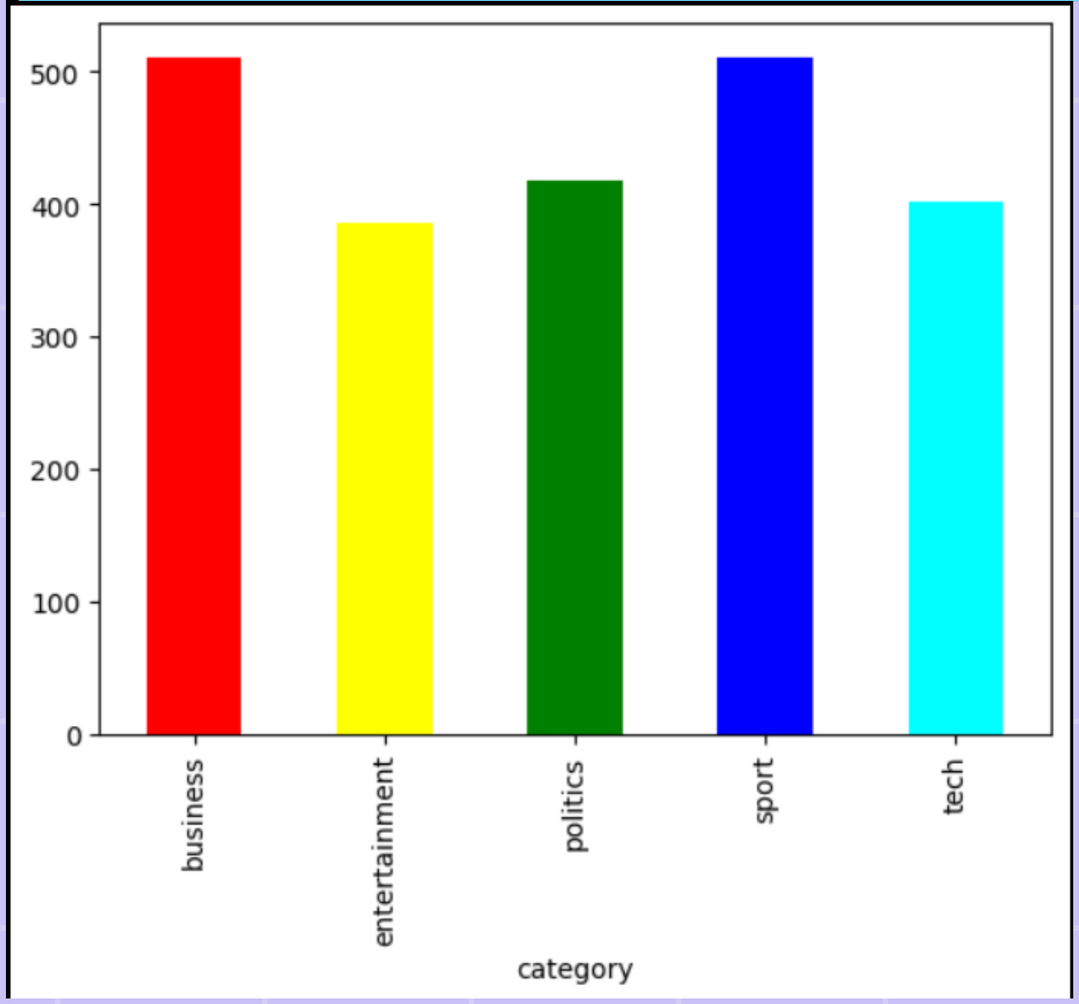
	category	text
0	tech	tv future in the hands of viewers with home th...
1	business	worldcom boss left books alone former worldc...
2	sport	tigers wary of farrell gamble leicester say ...
3	sport	yeading face newcastle in fa cup premiership s...
4	entertainment	ocean s twelve raids box office ocean s twelve...

Mengimport dataset “bb-text.csv” yang berisi category dan text

-ASSIGNMENT 2 Text-Classification using Logistic Regression

```
dataset.groupby('category').text.count().plot.bar(ylim=0, color=['red', 'yellow', 'green', 'blue', 'cyan'])
plt.show()
```

✓ 0.4s



Grafik batang ini menunjukkan bahwa dataset yang digunakan memiliki ketimpangan distribusi teks antara kategori. Category teknologi memiliki jumlah teks yang jauh lebih banyak daripada kategori lainnya, sedangkan kategori bisnis memiliki jumlah teks yang paling sedikit

-ASSIGNMENT 2 Text-Classification using Logistic Regression

```
stemmer = PorterStemmer()
words = stopwords.words("english")
dataset['cleaned'] = dataset['text'].apply(lambda x: " ".join([stemmer.stem(i) for i in re.sub("[^a-zA-Z]", " ", x).split() if i not in words]).lower())
```

✓ 8.6s

```
dataset.head()
```

✓ 0.0s

	category	text	cleaned
0	tech	tv future in the hands of viewers with home th...	tv futur hand viewer home theatr system plasma...
1	business	worldcom boss left books alone former worldc...	worldcom boss left book alon former worldcom b...
2	sport	tigers wary of farrell gamble leicester say ...	tiger wari farrel gambl leicest say rush make ...
3	sport	yeading face newcastle in fa cup premiership s...	yead face newcastl fa cup premiership side new...
4	entertainment	ocean s twelve raids box office ocean s twelve...	ocean twelv raid box offic ocean twelv crime c...

Melakukan preprocessing pada teks dalam kolom 'text' dari dataset. Prosesnya meliputi stemming kata-kata menggunakan PorterStemmer() dari NLTK, menghilangkan karakter non-alfanumerik, mengonversi teks menjadi huruf kecil, dan menghapus kata-kata umum (stopwords) dalam bahasa Inggris. Hasilnya disimpan dalam kolom baru 'cleaned' dalam dataset, menyediakan teks yang telah dimodifikasi untuk analisis lebih lanjut.

-ASSIGNMENT 2 Text-Classification using Logistic Regression

```
vectorizer = TfidfVectorizer(min_df= 3, stop_words="english", sublinear_tf=True, norm='l2', ngram_range=(1, 2))
final_features = vectorizer.fit_transform(dataset['cleaned']).toarray()
final_features.shape
```

✓ 1.1s

```
(2225, 29637)
```

menggunakan TfidfVectorizer() dari sklearn untuk mengubah teks yang telah dibersihkan dalam kolom 'cleaned' dataset menjadi representasi numerik dengan skor TF-IDF. Parameter-parameter seperti pembatasan frekuensi kata, penghapusan stopwords, penggunaan skala sublinear untuk frekuensi kata, normalisasi dengan norma L2, dan penggunaan unigram dan bigram sebagai fitur diterapkan pada teks. Hasilnya, disimpan dalam variabel final_features, adalah representasi numerik dari teks yang akan digunakan sebagai input untuk melatih model machine learning. final_features.shape menghasilkan dimensi dari representasi fitur terakhir yang dihasilkan dari teks yang telah diproses.

-ASSIGNMENT 2 Text-Classification using Logistic Regression

```
from sklearn.linear_model import LogisticRegression
X = dataset['cleaned']
Y = dataset['category']
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.25)

pipeline = Pipeline([('vect', vectorizer),
                     ('chi',  SelectKBest(chi2, k=1200)),
                     ('clf', LogisticRegression(random_state=0))])

model = pipeline.fit(X_train, y_train)
with open('LogisticRegression.pickle', 'wb') as f:
    pickle.dump(model, f)

ytest = np.array(y_test)

# confusion matrix and classification report(precision, recall, F1-score)
print(classification_report(ytest, model.predict(X_test)))
print(confusion_matrix(ytest, model.predict(X_test)))
```

✓ 1.3s

mempersiapkan dan melatih model klasifikasi Logistic Regression pada data teks. ertama, menggunakan Train-Test split, data dibagi menjadi set pelatihan dan pengujian. Selanjutnya, dibuat sebuah pipeline yang terdiri dari TfidfVectorizer untuk mengubah teks menjadi fitur numerik dengan skor TF-IDF, terakhir hasil data pengujian digunakan untuk melihat kinerja model terhadap kategori yang diprediks

-ASSIGNMENT 2 Text-Classification using Logistic Regression

	precision	recall	f1-score	support
business	0.98	0.95	0.97	131
entertainment	1.00	0.98	0.99	106
politics	0.98	0.98	0.98	104
sport	0.99	1.00	1.00	121
tech	0.95	1.00	0.97	95
accuracy			0.98	557
macro avg	0.98	0.98	0.98	557
weighted avg	0.98	0.98	0.98	557

```
[[125   0   1   1   4]
 [  0 104   1   0   1]
 [  2   0 102   0   0]
 [  0   0   0 121   0]
 [  0   0   0   0  95]]
```

Hasil evaluasi model menggunakan regresi logistic menunjukkan kinerja yang sangat baik dalam melakukan klasifikasi pada data teks. Dalam hal precision, recall, dan f1-score, model ini secara konsisten memberikan performa yang tinggi untuk setiap kategori. Nilai akurasi mencapai 98%, menunjukkan kemampuan model dalam memprediksi kategori yang benar dari total sampel uji. Confusion matrix juga menunjukkan bahwa sebagian besar prediksi model benar, dengan jumlah kecil dari kesalahan klasifikasi yang terlihat pada sejumlah kecil sampel. Hal ini menandakan bahwa model regresi logistic secara efektif dapat membedakan dan mengklasifikasikan berita pada kategori bisnis, hiburan, politik, olahraga, dan teknologi dengan akurasi yang tinggi.

THANK YOU FOR LISTENING!