Klasifikasi Teks Berita

Deskripsi singkat

- Sumber data: https://archive.ics.uci.edu/ml/datasets/News+Aggregator
- Deskripsi data: Teks berita bahasa inggris
- Goal: Mengklasifikasikan teks berita berdasarkan empat topik yaitu entertainment (e), business (b), technology (t) dan health (m)
- Algoritma yang digunakan: Support Vector Machine
- Hasil: Setelah dilakukan EDA, preprocessing teks, feature extraction dan modeling menggunakan Algoritma SVM didapat hasil akurasi, presisi dan recall yaitu 0.96, 0.96, 0.96. Waw, ga nyangka aing :v

```
from google.colab import drive
drive.mount('/content/drive',force_remount=True)
```

```
Mounted at /content/drive
```

```
import pandas as pd
from collections import Counter
import nltk
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.svm import LinearSVC
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
True
```

```
path="/content/drive/MyDrive/Tugas Akhir DSU Ceunah/"
```

EDA

```
df=pd.read_csv(path+"uci-news-aggregator.csv")
df.columns
```

```
Index(['ID', 'TITLE', 'URL', 'PUBLISHER', 'CATEGORY', 'STORY', 'HOSTNAME',
       'TIMESTAMP'],
      dtype='object')
```

```
df=df[["TITLE","CATEGORY"]]
df.columns=["judul","kategori"]
```

Mengecek jumlah masing-masing label

```
df['kategori']=df['kategori'].replace("b","business").replace("m","health").replac
```

```
df['kategori'].value_counts()
```

```
entertainment    152469
business         115967
technology       108344
health            45639
Name: kategori, dtype: int64
```

Mengecek duplikat dari dataset

- Jika max lebih dari satu maka ada duplikat

```
data=Counter(df.judul)
print(max(data.values()))
print(min(data.values()))
```

```
145
1
```

```
df=df.drop_duplicates(subset=['judul'])
```

Text Preprocessing

Lowercasing

```
df['judul']=df['judul'].str.lower()
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:1: SettingWithCo
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/s
  """Entry point for launching an IPython kernel.
```

Punctuation removal

```
puncs = '''!-():[]{};"\,<>./?@#$%^&*_~'''
def rm_punc(text):
  for el in text:
    if el in puncs:
      text=text.replace(el , '')
  return text
```

```
df['judul']=[*map(lambda word:rm_punc(word) , df['judul'].values)]
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:1: SettingWithCo
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/s
  """Entry point for launching an IPython kernel.
```

Stopword removal

```
sw = stopwords.words('english')
# sw.append('coronavirus')
def stop_word(text):
  new_text = []
  for word in text.split():
    if word not in sw:
      new_text.append(word)
  return(' '.join(new_text))
```

```
df['judul'] = df['judul'].apply(stop_word)
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:1: SettingWithCo
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/s
  """Entry point for launching an IPython kernel.
```

Lemmatization

```
lemmatizer = WordNetLemmatizer()
df['judul']=[*map(lambda word:lemmatizer.lemmatize(word) , df['judul'].values)]
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:2: SettingWithCo
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/s
```

Feature Extraction

```
Tfidf = TfidfVectorizer(ngram_range=(1, 2))
tfidf_features = Tfidf.fit_transform(df.judul)
tfidf_features.shape
```

```
(406455, 1081165)
```

Modeling

```
svc = LinearSVC()
X_train, X_test, y_train, y_test = train_test_split(tfidf_features, df['kategori']
svc.fit(X_train,y_train)
prediction = svc.predict(X_test)
```

Evaluation

```
print("accuracy score:")
print(accuracy_score(y_test, prediction))
print(classification_report(prediction,y_test))
```

```
accuracy score:
0.9594912105891181
              precision    recall  f1-score   support

    business       0.95      0.94      0.94     22425
entertainment       0.99      0.98      0.98     29626
      health       0.94      0.97      0.96      8495
  technology       0.95      0.95      0.95     20745

    accuracy                           0.96     81291
   macro avg       0.95      0.96      0.96     81291
weighted avg       0.96      0.96      0.96     81291
```