MODUL DATA MINING

TEXT CLASSIFICATION

Pada modul ini dijelaskan mengenai proses klasifikasi dengan data teks dan menerapkannya dalam bahasa pemrograman python.
Diharapkan setelah mempelajari modul ini, mahasiswa mampu memahami tujuan klasifikasi teks pada suatu kasus.

EPS 8

TEXT CLASSIFICATION

Text classification atau text categorization merupakan salah satu bentuk implementasi text mining. Pada bagian awal akan digunakan pemrosesan NLP dan membentuknya ke dalam bentuk terstruktur (dalam bentuk vectorized) sehingga dapat diolah dengan menggunakan teknik klasifikasi pada data mining. Proses klasifikasi sama seperti proses klasifikasi yang telah dipelajari pada modul klasifikasi.

LOAD LIBRARY

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

plt.style.use('ggplot')
sns.set_style('whitegrid')
```

```
## library untuk preprocessing teks
import csv
import re
import nltk
import string
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize, sent_tokenize
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

## library untuk vectorized
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import TfidfTransformer

## library untuk pemodelan klasifikasi
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.naive_bayes import BernoulliNB
from sklearn.naive_bayes import GaussianNB

## library untuk evaluasi
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score

pd.set_option('max_colwidth',180)
```

LOAD DATA

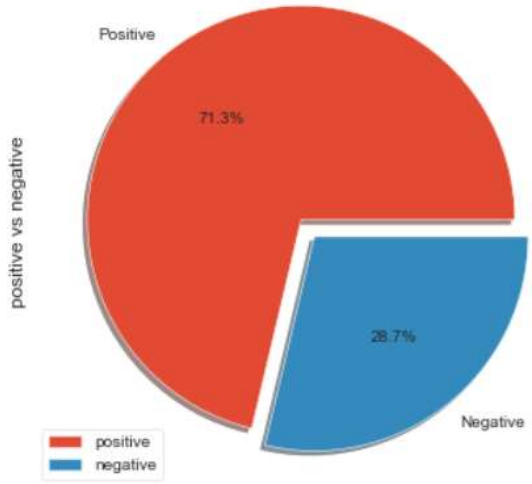
```
df = pd.read_csv('D:[redacted]xiaomi2019.csv', sep=';', encoding='ISO-8859-1')
df.head()
```

Catatan: digunakan encoding karena pada data terdapat karakter non UTF-8. Bisa digunakan cara seperti ini, atau digunakan cara lain.

EXPLORE DATA

```
df.info()
```

```
df["sentiment"].value_counts().plot(kind = 'pie', explode = [0, 0.1], figsize = (6, 6), autopct = '%1.1f%%', shadow = True)
plt.ylabel("positive vs negative")
plt.legend(["positive", "negative"])
plt.show()
```




```
df['length'] = df['tweet'].apply(len)
df.head()
```

	tweet	sentiment	length
0	pake hp xiaomi bisa kan	Positive	23
1	Xiaomi yi action kamera bagus juga buat video bisa di jadikan pertimbangan nihhh	Positive	80
2	Ya Allah jauhkanlah aku dari godaan clickbait line today dan berita-berita di browser xiaomi	Positive	92
3	hpmu opo se? kok sawangane jernih koyok iph â□□ Xiaomi euy ' https://lap78.ask.fm/igoto/45DKECPW7B667HQMHN2IG6NM7SOD5OAUS7QPPAN4D7ROV45V2Q24OJAMGBFM2RRQK2272FYJJNWDWXQVYYU5Y25C	Positive	309
4	numpang nanya itu hapenya xiaomi bukan ya? Kalo iya tipe apa? Jernih kameranya mirip iphone	Positive	91

Catatan: len disini adalah fungsi untuk menghitung panjang karakter (karakter spasi juga dihitung). Silahkan cari cara untuk menghitung panjang/frekuensi per kata, bukan per karakter.

```
df["tweetLength"] = df["tweet"].apply(len)
df["tweetLength"].describe()
```

```
count    101.000000
mean      76.643564
std       56.517534
min        7.000000
25%       36.000000
50%       60.000000
75%       99.000000
max      301.000000
Name: tweetLength, dtype: float64
```

CLEANING DATA

```
def clean_text(text):
    #mengubah semua karakter huruf menjadi huruf kecil
    text = text.lower()
    # menghilangkan Nama Akun
    text = re.sub('@[^\s]+', '', text)
    # menghilangkan punctuation
    # text = re.sub('[%s]' % re.escape(string.punctuation),'',text)
    # menghilangkan angka
    text = re.sub('\w*\d\w*','',text)
    # menghilangkan URL
    text = re.sub(r'\w+:\/{2}[\d\w-]+(\.[\d\w-]+)*(?:(?:\/[^\s/]*))*', '', text)
    text = re.sub(r'(https?:\/\/)?([\da-z\.-]+)\.([a-z\.]{2,6})([\/\w\.-]*)*\/?\S', '', text)
    # menghilangkan Hastag
    text = re.sub('#[^\s]+', '', text)
    # menghilangkan Huruf Tunggal
    text = re.sub(r'\b[a-zA-Z]\b', '', text)
    return text

clean = lambda x: clean_text(x)
```

```
dfx = pd.DataFrame(df.tweet.apply(clean))
dfx
```

Catatan: karena yang akan di praproses adalah kolom tweet yang berisi text, maka kita buat dataframe baru yaitu dfx yang khusus berisi tweet yang akan kita praproses saja.

```
def Punctuation(string):

    # punctuation marks
    punctuations = '''!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~'''

    # traverse the given string and if any punctuation
    # marks occur replace it with null
    for x in string.lower():
        if x in punctuations:
            string = string.replace(x, "")

    # Print string without punctuation
    return(string)

cleanPunc = lambda x: Punctuation(x)
```

```
dfx = pd.DataFrame(dfx.tweet.apply(cleanPunc))
dfx.head()
```

Catatan: setiap selesai di proses, cek hasilnya, apakah ada yang berbeda?

STOPWORD REMOVAL

```
def get_stopword(stopwordsfile):
    stopwords=[]
    file_stopwords = open(stopwordsfile,'r')
    row = file_stopwords.readline()
    while row:
        word = row.strip()
        stopwords.append(word)
        row = file_stopwords.readline()
    file_stopwords.close()
    return stopwords
```

```
stop_words_indo = get_stopword('D:/KULIAH/Data Mining/Modul/stopwordsindo.txt')
```

Catatan: code diatas dilakukan jika daftar stopword disimpan dalam file terpisah (kamus daftar stopword). Jika menggunakan library stopword, maka tidak dilakukan seperti contoh tersebut.

```
def stopwords(text):

    tokens = word_tokenize(text)
    filtered = []

    for w in tokens:
        if w not in stop_words_indo:
            filtered.append(w)

    hasil = ' '.join(filtered)
    return hasil

st = lambda x: stopwords(x)
```

```
dfx = pd.DataFrame(dfx.tweet.apply(st))
dfx.head()
```

	tweet
0	pake hp xiaomi
1	xiaomi yi action kamera bagus video jadikan pertimbangan nihhh
2	ya allah jauhkanlah godaan clickbait line today beritaberita browser xiaomi
3	hpmu opo sawangane jernih koyok iph â□□ xiaomi euy â â□¦
4	numpang nanya hapenya xiaomi ya kalo iya tipe jernih kameranya iphone

Catatan: apakah ada yang berubah dengan luaran sebelumnya? Apa yang berubah?

STEMMING

```
def stemming(text):

    factory_stem = StemmerFactory()
    stemmer = factory_stem.create_stemmer()
    text = stemmer.stem(text)
    return text

stem = lambda x: stemming(x)
```

```
dfx = pd.DataFrame(dfx.tweet.apply(stem))
dfx.head()
```

	tweet
0	pake hp xiaomi
1	xiaomi yi action kamera bagus video jadi timbang nihhh
2	ya allah jauh goda clickbait line today beritaberita browser xiaomi
3	hpmu opo sawangane jernih koyok iph xiaomi euy
4	numpang nanya hapenya xiaomi ya kalo iya tipe jernih kamera iphone

Catatan: apakah ada yang berubah dengan luaran sebelumnya? Apa yang berubah?

GABUNG DENGAN ATRIBUT KELAS

```
dfx["sentiment"] = df["sentiment"]
dfx
```

	tweet	sentiment
0	pake hp xiaomi	Positive
1	xiaomi yi action kamera bagus video jadi timbang nihhh	Positive
2	ya allah jauh goda clickbait line today beritaberita browser xiaomi	Positive
3	hpmu opo sawangane jernih koyok iph xiaomi euy	Positive
4	numpang nanya hapenya xiaomi ya kalo iya tipe jernih kamera iphone	Positive
...	...	...
96	xiaomi bagus ka hehe	Positive
97	kalo xiaomi screen record wkwkwk	Positive
98	bgst nanya yg xiaomi mi susah cari sinyal ga semenjak update patch juni	Negative
99	hp gitu xiaomi ga sampe konter ku cari google alhamdulillah tau bootlop nder kalo bootlop gausah konter sayang uang	Negative
100	kalo xiaomi note yg ga finger print jarang bgt	Positive

101 rows × 2 columns

VECTORIZED

Mengubah data menjadi TDM sehingga berubah menjadi bentuk vector. Pada code ini menggunakan TFIDF.

```
vectorizer = TfidfVectorizer(use_idf=True, strip_accents='ascii')
```

PEMBUATAN MODEL KLASIFIKASI

```
y = dfx.sentiment
X = dfx.tweet
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state= 33)
```

```
# Fit vectorizer and transform X train, then transform X test
X_train_vect = vectorizer.fit_transform(X_train)
X_test_vect = vectorizer.transform(X_test)

mnb = MultinomialNB()

mnb.fit(X_train_vect, y_train)
y_pred = mnb.predict(X_test_vect)

accuracy_score(y_test, y_pred)
```

```
0.6774193548387096
```

LATIHAN:

1. Jelaskan perubahan setiap proses pada praproses (tunjukkan dalam bentuk screenshot)
2. Evaluasi model pada contoh hanya akurasi, silahkan buat confusion matrixnya dan hitung dengan metric yang lain misalnya precision, recall, ROC/AUC, dsb.
3. Silahkan dicoba jika menggunakan cross-validation, dan hitung evaluasinya.