Text Mining Case

Knowledge Discovery

Hanif Izzudin Rahman

1. Read Data

```
import nltk
import re
import string
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from nltk.tokenize import word_tokenize
from nltk.probability import FreqDist
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer

from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory
```

```
data=[]
for i in range(1,51):
    openfile='news_dataset/data%d.txt'%i
    f = open(openfile, "r", encoding="Latin-1")
    data.append(f.read())
    f.close()
print("\nJumlah Text Data:", len(data))
```

```
Jumlah Text Data: 50
```

2. Keywords (Preprocessing)

```
for i in range(len(data)):
    data[i] = data[i].lower()
    data[i] = re.sub(r"\d+", "", data[i])
    data[i] = data[i].translate(str.maketrans("","",string.punctuation))
    data[i] = data[i].strip()
```

```
data
```

```
['harga emas batangan bersertifikat antam keluaran logam mulia pt aneka tambang tbk antm naik pada hari selasa \n\nmengutip s
itus logam mulia harga pecahan satu gram emas antam berada di rp  harga emas antam ini naik rp  dari harga jumat  lalu di rp
\n\nsementara harga pembelian kembali atau buyback emas antam juga turun rp  dan berada di rp \n\nberikut harga emas batangan
antam dalam pecahan lainnya per hari ini dan belum termasuk pajak\n\nharga emas  gram rp \n\nharga emas  gram rp \n\nharga em
as  gram rp \n\nharga emas  gram rp \n\nharga emas  gram rp \n\nharga emas  gram rp \n\nharga emas  gram rp \n\nharga emas  g
ram rp \n\nharga emas  gram rp \n\nharga emas  gram rp \n\nketerangan\n\nlogam mulia antam menjual emas dan perak batangan da
lam beberapa ukuran berat misalnya  gram  gram dan  gram biasanya harga per gram emas antam akan berbeda tergantung berat bat
angnya perbedaan ini terjadi karena ada biaya tambahan untuk pencetakan sehingga harga per gram emas antam batang kecil lebih
mahal dari batang yang lebih besar harga yang ada di sini adalah harga per gram emas batang  kilogram yang biasa dijadikan pa
tokan pelaku bisnis emas',
 'saat perdagangan kamis  lalu indeks harga saham gabungan ihsg melemah  ke level  sebanyak  saham menguat  saham melemah dan
saham diam di tempat\n\npasca libur lebaran analis menilai ihsg akan kembali bergerak di zona merah\n\nanalis sucor sekuritas
hendriko gani mengatakan indikator teknikal menunjukkan adanya sinyal pelemahan pada ihsg\n\nselain itu kabar mengenai potens
i meningkatnya kasus covid usai idul fitri juga akan menekan pergerakan ihsg pada selasa \n\nsementara dari luar negeri ihsg
bakal diperberat dengan adanya demonstrasi yang terjadi di hong kong\n\nasal tahu warga hong kong menggelar aksi unjuk rasa t
erkait rencana pemerintah china yang akan menerapkan undangundang keamanan nasional\n\nâ\x80\x9coleh karena itu ihsg berpoten
si bergerak mixed dengan kecenderungan melemahâ\x80\x9d ujar hendriko saat dihubungi kontancoid senin \n\nadapun hendriko men
erawang ihsg akan bergerak di rentang  â\x80\x93 \n\ndirektur indosurya bersinar sekuritas william surya wijaya mengatakan po
la gerak ihsg pasca libur lebaran masih berada dalam fase konsolidasi wajar dengan potensi tekanan yang terlihat masih belum
```

2. Keywords (Filtering Sastrawi, Stemming Sastrawi, Tokenizing)

```
data_sastra=[]
tokens=[]
tf=[]
for i in range(len(data)):
    # Filtering dengan Sastrawi -------------------
    factory = StopWordRemoverFactory()
    stopword = factory.create_stop_word_remover()
    data_sastra.append(stopword.remove(data[i]))

    # Stemming dengan Sastrawi -----------------
    factory = StemmerFactory()
    stemmer = factory.create_stemmer()
    data_sastra[i] = stemmer.stem(data_sastra[i])

    tokens.append(word_tokenize(data_sastra[i]))
    tf.append(FreqDist(tokens[i]))
    word, frequency=tf[i].most_common()[0]
```

```
data_sastra
```

```
['harga emas batang sertifikat antam keluar logam mulia pt aneka tambang tbk antm naik hari selasa kutip situs logam mulia ha
rga pecah satu gram emas antam ada rp harga emas antam naik rp harga jumat lalu rp sementara harga beli atau buyback emas ant
am turun rp ada rp ikut harga emas batang antam pecah lain per hari masuk pajak harga emas gram rp harga emas gram rp harga e
mas gram rp harga emas gram rp harga emas gram rp harga emas gram rp harga emas gram rp harga emas gram rp harga emas gram rp
harga emas gram rp terang logam mulia antam jual emas perak batang beberapa ukur berat misal gram gram dan gram biasa harga p
er gram emas antam beda gantung berat batang beda jadi ada biaya tambah cetak harga per gram emas antam batang kecil lebih ma
hal batang lebih besar harga ada sini harga per gram emas batang kilogram biasa jadi patok laku bisnis emas',
 'dagang kamis lalu indeks harga saham gabung ihsg lemah level banyak saham kuat saham lemah saham diam tempat pasca libur le
baran analis nilai ihsg kembali gerak zona merah analis sucor sekuritas hendriko gani kata indikator teknikal tunjuk ada siny
```

3. Scores

```
1 tf
```

```
[FreqDist({'harga': 20, 'emas': 20, 'gram': 17, 'rp': 15, 'antam': 8, 'batang': 7, 'ada': 4, 'per': 4, 'logam': 3, 'mulia': 3,
...}),
 FreqDist({'saham': 9, 'ihsg': 9, 'gerak': 5, 'dagang': 4, 'lemah': 4, 'william': 4, 'pasca': 3, 'libur': 3, 'lebaran': 3, 'hen
driko': 3, ...}),
 FreqDist({'harga': 22, 'rp': 14, 'bawang': 12, 'putih': 12, 'per': 10, 'naik': 7, 'relaksasi': 7, 'impor': 5, 'perintah': 5,
'kilo': 5, ...}),
 FreqDist({'bank': 13, 'layan': 13, 'kcu': 11, 'banking': 10, 'bca': 10, 'transaksi': 9, 'mei': 9, 'kantor': 9, 'bri': 8, 'oper
asional': 7, ...}),
 FreqDist({'rupiah': 5, 'pasar': 4, 'kuat': 4, 'jadi': 4, 'gera': 3, 'dagang': 3, 'belum': 3, 'ibrahim': 3, 'kata': 3, 'rp': 3,
...}),
 FreqDist({'rp': 18, 'harga': 15, 'emas': 12, 'gram': 11, 'juta': 10, 'ukur': 8, 'batang': 7, 'cetak': 6, 'galeri': 5, 'ubs':
5, ...}),
 FreqDist({'as': 12, 'dolar': 7, 'rp': 6, 'per': 6, 'ariston': 6, 'kuat': 5, 'jadi': 5, 'rupiah': 4, 'lockdown': 4, 'pasar': 4,
...}),
 FreqDist({'saham': 5, 'ihsg': 5, 'ada': 3, 'zona': 3, 'hingga': 3, 'gerak': 3, 'hijau': 2, 'belum': 2, 'william': 2, 'kata':
2, ...}),
 FreqDist({'minyak': 14, 'harga': 8, 'amerika': 7, 'serikat': 7, 'persen': 5, 'cina': 4, 'hari': 3, 'jatuh': 3, 'lalu': 3, 'seb
ut': 3, ...}),
 FreqDist({'rupiah': 9, 'kuat': 7, 'indonesia': 6, 'level': 6, 'bank': 5, 'persen': 5, 'rp': 5, 'per': 5, 'dolar': 5, 'as': 5,
...}),
 FreqDist({'latih': 14, 'inggris': 12, 'tahap': 11, 'dua': 9, 'covid': 9, 'liga': 8, 'main': 8, 'nyata': 6, 'perintah': 5, 'klu
b': 5, ...}),
 FreqDist({'madrid': 9, 'real': 7, 'dortmund': 6, 'sebut': 5, 'haaland': 5, 'siap': 3, 'lepas': 3, 'serang': 3, 'erling': 3, 'd
atang': 3, ...}),
 FreqDist({'liga': 20, 'inggris': 10, 'lanjut': 8, 'latih': 8, 'juni': 7, 'jadwal': 6, 'italia': 6, 'dua': 6, 'nyata': 6, 'mai
n': 5, ...}),
 FreqDist({'italia': 4, 'spadafora': 4, 'kata': 4, 'liga': 4, 'lanjut': 4, 'jadi': 4, 'serie': 3, 'a': 3, 'pandemi': 3, 'akan':
```

3. Scores (50% TF)

```
tf_remove=[]
rankdocs=np.arange(2).reshape(1,2)
for i in range(len(tf)):
    dfdata=pd.DataFrame(np.array(tf[i].most_common()),columns=['word','frequency'])
    dfdata['frequency']=dfdata['frequency'].astype('int')
    dfdata_remove=dfdata.loc[dfdata['frequency']>=dfdata['frequency'].max()/2]
    tf_remove.append(dfdata_remove.values)
    if(dfdata_remove[dfdata_remove.word.isin(list(querytf.keys()))].shape[0]>0):
        rank=np.array([dfdata_remove[dfdata_remove.word.isin(list(querytf.keys())+['gerak'])]['frequency'].sum(),i])
        rankdocs=np.append(rankdocs,rank.reshape(1,2),axis=0)
rankdocs=np.delete(rankdocs,0,axis=0)
rankdocs=pd.DataFrame(rankdocs,columns=['sum','index']).sort_values(['sum'],ascending=[False])
```

```
tf_remove
```

```
[array([['harga', 20],
        ['emas', 20],
        ['gram', 17],
        ['rp', 15]], dtype=object),
 array([['saham', 9],
        ['ihsg', 9],
        ['gerak', 5]], dtype=object),
 array([['harga', 22],
        ['rp', 14],
        ['bawang', 12],
        ['putih', 12]], dtype=object),
 array([['bank', 13],
        ['layan', 13],
        ['kcu', 11],
        ['banking', 10],
        ['bca', 10],
        ['transaksi', 9],
        ['mei', 9],
        ['kantor', 9],
```

4. Query List

```
query='pertumbuhan ekonomi, perkembangan pasar dan pergerakan harga saham'
query = query.lower()
query = re.sub(r"\d+", "", query)
query = query.translate(str.maketrans("","",string.punctuation))
query = query.strip()
```

```
query
```

```
'tumbuh ekonomi kembang pasar gera harga saham'
```

```
# Filtering dengan Sastrawi -------------------
query = stopword.remove(query)
print("\nSetelah filtering:\n----------------\n", query)

# Stemming dengan Sastrawi -----------------
query = stemmer.stem(query)
print("\nOutput stemming:\n----------------\n", query)

querytokens = word_tokenize(query)
print("\nTokenizing:\n----------------\n", querytokens)

querytf = FreqDist(querytokens)
print("\nTerm Frequency:\n----------------\n", querytf.most_common())

word, frequency=querytf.most_common()[0]
print("\nKeyword yang paling banyak muncul:\n----------------\n", word, "=", frequency , "\n")
print("\nKeseluruhan keywords:\n----------------\n")

for word, frequency in querytf.most_common():
    print(word, ":", frequency)
```

4. Query List (Filtering, Stemming, Tokenizing, Term Frequency)

```
Setelah filtering:
---------------
 pertumbuhan ekonomi perkembangan pasar pergerakan harga saham

Output stemming:
---------------
 tumbuh ekonomi kembang pasar gera harga saham

Tokenizing:
---------------
 ['tumbuh', 'ekonomi', 'kembang', 'pasar', 'gera', 'harga', 'saham']

Term Frequency:
---------------
 [('tumbuh', 1), ('ekonomi', 1), ('kembang', 1), ('pasar', 1), ('gera', 1), ('harga', 1), ('saham', 1)]

Keyword yang paling banyak muncul:
---------------
 tumbuh = 1


Keseluruhan keywords:
---------------

tumbuh : 1
ekonomi : 1
kembang : 1
pasar : 1
gera : 1
harga : 1
saham : 1
```

5. Rank Docs

```
label=pd.read_csv('label.csv',names=['Data','Category'])
rankdocs=label.loc[rankdocs['index'].values]
rankdocs.iloc[0:10].reset_index(drop=True)
```

	Data	Category
0	data3	economy
1	data1	economy
2	data6	economy
3	data2	economy
4	data47	tourism
5	data42	tourism
6	data44	tourism
7	data8	economy
8	data9	economy
9	data5	economy

6. Read Label

```
1 label
```

	Data	Category
0	data1	economy
1	data2	economy
2	data3	economy
3	data4	economy
4	data5	economy
5	data6	economy
6	data7	economy
7	data8	economy
8	data9	economy
9	data10	economy
10	data11	soccer
11	data12	soccer
12	data13	soccer
13	data14	soccer
14	data15	soccer
15	data16	soccer
16	data17	soccer
17	data18	soccer

7. Precision - Recall

belum

8. Graph

belum