

Nama: Rafael Nicholas Tanaja

NIM: 2540118656

Web Scrapping

```
import requests
from bs4 import BeautifulSoup

def webscrap(url, media, label):
    # Request Website
    response = requests.get(url)

    # Menggunakan BeautifulSoup untuk parsing HTML
    soup = BeautifulSoup(response.text, 'html.parser')

    # Mendefinisikan div class setiap website, karena setiap website berbeda
    if media == 'MediaIndonesia':
        _class_ = 'rows_jap'
    elif media == 'CNN':
        _class_ = 'detail-text text-cnn_black text-sm grow min-w-0'
    elif media == 'Kompas':
        _class_ = 'read_content'

    div = soup.find('div', _class_)

    df = pd.DataFrame(columns=['Teks', 'Media', 'Label'])

    # Mencari <p>
    paragraphs = div.find_all('p')
    content = []

    # Memasukkan setiap <p> kedalam list content
    for paragraph in paragraphs:
        content.append(paragraph.get_text())

    # Menggabungkan content kedalam dataframe
    if content:
        df = pd.concat([pd.DataFrame({'Teks': ['\n'.join(content)], 'Media': media, 'Label': label}, columns=df.columns), df])

    return df
```

✓ 0.3s

Teknik yang digunakan disini adalah BeautifulSoup,

1. Membuat function webscrap dengan parameter url, media, dan label
2. Menggunakan library request untuk mendapatkan webpage.
3. Memanggil BeautifulSoup untuk parsing HTML.
4. Mendefinisikan <div> class tiap berita, untuk mengambil informasi yang diperlukan.
<div> class untuk informasi yang diperlukan berbeda-beda setiap website berita..
5. Find <div> class yang sudah di masukkan kedalam variable _class_.
6. Mencari <p> untuk mendapatkan kalimat berita yang kemudian akan di append kedalam variable content.
7. Join semua kalimat didalam list content yang kemudian dimasukkan ke dalam df.
8. Function me-return df.

```

urls_mediaindonesia = ['https://mediaindonesia.com/politik-dan-hukum/627681/ridwan-kamil-bawa-pengaruh-elektoral-prabowo-gibran-di-jabar',
                        'https://mediaindonesia.com/politik-dan-hukum/627651/jelang-putusan-mkmm-2149-personel-polisi-amankan-gedung-mk',
                        'https://mediaindonesia.com/politik-dan-hukum/627643/anies-percaya-putusan-mkmm-objektif',
                        'https://mediaindonesia.com/politik-dan-hukum/627610/polda-metro-jaya-bisa-ekspose-penetapan-tersangka-kasus-pemerasan-syl-besok',
                        'https://mediaindonesia.com/politik-dan-hukum/627589/nasdem-bersyukur-mendapat-dukungan-dari-din-syamsudin-untuk-pilpres-2024',
                        'https://mediaindonesia.com/politik-dan-hukum/628682/kubu-prabowo-gibran-sebut-baliho-masif-sebagai-bentuk-semangat-relawan',
                        'https://mediaindonesia.com/politik-dan-hukum/628589/wamenkumham-eddy-mengaku-belum-pernah-diperiksa-kpk',
                        'https://mediaindonesia.com/politik-dan-hukum/628574/wamenkumham-jadi-tersangka-gratifikasi-begini-kronologi-versi-pelapor',
                        'https://mediaindonesia.com/politik-dan-hukum/628447/kebocoran-rph-mk-dalam-putusan-usia-cawapres-dilaporkan-ke-bareskrim-polri',
                        'https://mediaindonesia.com/politik-dan-hukum/628446/relawan-gajamada-sebut-polemik-gibran-gerus-suara-prabowo']

urls_cnn = ['https://www.cnnindonesia.com/olahraga/20231104202115-142-1020057/sty-blak-blakan-soal-peluang-duel-indonesia-vs-korea-di-piala-asia',
            'https://www.cnnindonesia.com/olahraga/20231104154839-142-1020000/psis-lepas-hulk-ke-timnas-indonesia-tunjukkan-kamu-layak',
            'https://www.cnnindonesia.com/olahraga/20231105103652-142-1020140/dicintai-indonesia-sty-merasa-seperti-guus-hiddink-di-korea',
            'https://www.cnnindonesia.com/olahraga/20231107115438-142-1020935/legenda-man-city-darius-vassell-dampingi-inggris-u-17-di-indonesia',
            'https://www.cnnindonesia.com/olahraga/20231107034242-142-1020771/pelatih-ekuator-puji-timnas-indonesia-u-17-setinggi-langit',
            'https://www.cnnindonesia.com/olahraga/20231110163653-142-1022662/saifdine-chlaghmo-pencetak-gol-pertama-piala-dunia-u-17-2023',
            'https://www.cnnindonesia.com/olahraga/20231110163807-142-1022664/piala-dunia-u-17-mali-unggul-1-0-atas-uzbekistan-di-babak-pertama',
            'https://www.cnnindonesia.com/olahraga/20231110151354-156-1022598/hasil-practice-motogp-malaysia-alex-marquez-tercepat-martin-kedua',
            'https://www.cnnindonesia.com/olahraga/20231110170432-170-1022672/hasil-korea-masters-ester-menang-kento-momota-cemerlang',
            'https://www.cnnindonesia.com/olahraga/20231110072550-142-1022312/panama-incar-menang-lawan-tim-bintang-sebelum-jumpa-timnas-indonesia']

urls_kompas = ['https://travel.kompas.com/read/2023/11/04/101300927/obyek-wisata-baru-parapuar-bisa-nikmati-labuan-bajo-dari-ketinggian',
               'https://travel.kompas.com/read/2023/11/03/130050127/menikmati-momijigari-di-tohoku-dan-kanto-saat-musim-gugur',
               'https://travel.kompas.com/read/2023/11/01/143400827/pengalaman-berburu-sunset-di-kuta-bali-saat-akhir-pekan-awas-macet',
               'https://travel.kompas.com/read/2023/11/06/171205927/7-aktivitas-di-cfd-colomadu-bisa-cek-kesehatan',
               'https://travel.kompas.com/read/2023/11/05/210100027/5-wisata-favorit-di-kabupaten-semarang-dusun-semilir-paling-banyak-dikunjungi',
               'https://travel.kompas.com/read/2023/11/10/122815527/healing-murah-meriah-ke-taman-langsar-bisa-ngapain-aja',
               'https://travel.kompas.com/read/2023/11/10/153728527/gunung-padang-disebut-bisa-jadi-piramida-tertua-di-dunia-ketahui-6-faktanya',
               'https://travel.kompas.com/read/2023/11/08/224248727/pantai-pink-bima-surga-di-ujung-timur-pulau-sumbawa',
               'https://travel.kompas.com/read/2023/11/08/204000527/mengapa-kudus-dijuluki-kota-kretek-ketahui-5-alasannya',
               'https://travel.kompas.com/read/2023/11/06/194000327/museum-kretek-kudus-harga-tiket-jam-buka-dan-koleksi']

df1 = pd.DataFrame(columns=['Teks', 'Media', 'Label'])

for i in urls_mediaindonesia:
    df1 = pd.concat([df1, webscrap(i, 'MediaIndonesia', 'Politik')])

for i in urls_cnn:
    df1 = pd.concat([df1, webscrap(i, 'CNN', 'Olahraga')])

for i in urls_kompas:
    df1 = pd.concat([df1, webscrap(i, 'Kompas', 'Hiburan')])

```

1. Terdapat 30 link berita yang dimana setiap media memiliki jenis berita dan jumlah masing-masing 10.
2. Menggunakan for loop untuk iterate setiap berita dan memberikan label secara manual yang nantinya dimasukkan kedalam dataframe 'df1'.

Text Cleaning

```
import string
import re

df2 = df1.copy()

### Menghapus advertisement text pada setiap media berita
def replace_teks(row):
    text = row['Teks']
    if row['Media'] == 'MediaIndonesia':
        text = text.replace('Baca juga:', ' ')
    elif row['Media'] == 'CNN':
        text = text.replace('CNN', ' ')
        text = text.replace('ADVERTISEMENT', ' ')
        text = text.replace('SCROLL TO CONTINUE WITH CONTENT', ' ')
        text = text.replace('[Gambas:Video CNN]', ' ')
    elif row['Media'] == 'Kompas':
        text = text.replace('JAKARTA, KOMPAS.com -', ' ')
        text = text.replace('LABUAN BAJU, KOMPAS.com -', ' ')
        text = text.replace('KOMPAS.com', ' ')
        text = text.replace('BALI, KOMPAS.com -', ' ')
        text = text.replace('Kompas', ' ')
        text = text.replace('Baca juga:', ' ')
        text = text.replace('\n', ' ')
        text = text.replace('UNGARAN, KOMPAS.com -', ' ')

    return text

df2['Teks'] = df2.apply(replace_teks, axis=1)

## Menghapus HTTPS dan .com
pattern = r"https?://[^\s]+\.\com"

df2['clean_Teks'] = df2['Teks'].str.replace(pattern, "", regex=True)

### Mengubah teks menjadi lowercase
df2['clean_Teks'] = df2['clean_Teks'].str.lower()

# Menghilangkan simbol (tanda baca)
table = str.maketrans('', '', string.punctuation)

df2['clean_Teks'] = df2['clean_Teks'].str.translate(table)
```

```
# Menghilangkan number
def remove_number(text):
    return re.sub(r"\d+", '', text)

df2['clean_Teks'] = df2['clean_Teks'].apply(remove_number)

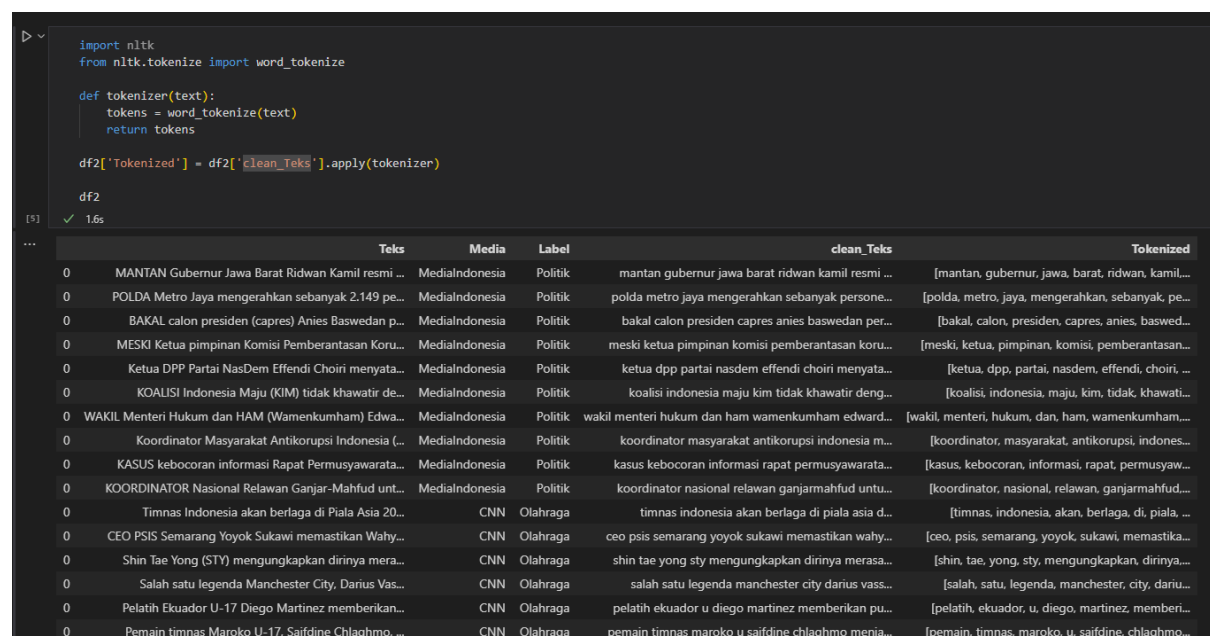
# Menghilangkan enter
df2['clean_Teks'] = df2['clean_Teks'].str.replace('\n', '')

# Menghilangkan white space
df2['clean_Teks'] = df2['clean_Teks'].str.strip()

print(df2)
```

1. Membersihkan data teks berita yang dimulai dengan menghapus Advertisement, Baca Juga:, dan berbagai 'noise' lainnya berdasarkan media berita.
2. Menghapus https dan .com (memastikan untuk tidak tersisa didalam data).
3. Merubah teks menjadi lower case.
4. Menghapus punctuation yaitu tanda baca.
5. Menghapus angka-angka.
6. Menghilangkan enter.
7. Menghilangkan white space pada text.
8. Text yang sudah di clean kemudian dimasukkan kedalam kolom clean_Teks.

Tokenizing, Remove Stop Words, and Stemming



```

import nltk
from nltk.tokenize import word_tokenize

def tokenizer(text):
    tokens = word_tokenize(text)
    return tokens

df2['Tokenized'] = df2['clean_Teks'].apply(tokenizer)

df2

```

	Teks	Media	Label	clean_Teks	Tokenized
0	MANTAN Gubernur Jawa Barat Ridwan Kamil resmi ...	MedialIndonesia	Politik	mantan gubernur jawa barat ridwan kamil resmi ...	[mantan, gubernur, jawa, barat, ridwan, kamil, ...]
0	POLDA Metro Jaya mengerahkan sebanyak 2.149 pe...	MedialIndonesia	Politik	polda metro jaya mengerahkan sebanyak persone...	[polda, metro, jaya, mengerahkan, sebanyak, pe...
0	BAKAL calon presiden (capres) Anies Baswedan p...	MedialIndonesia	Politik	bakal calon presiden capres anies baswedan per...	[bakal, calon, presiden, capres, anies, baswed...
0	MESKI Ketua pimpinan Komisi Pemberantasan Koru...	MedialIndonesia	Politik	meski ketua pimpinan komisi pemberantasan koru...	[meski, ketua, pimpinan, komisi, pemberantasan...
0	Ketua DPP Partai NasDem Effendi Choiri menyata...	MedialIndonesia	Politik	ketua dpp partai nasdem effendi choiri menyata...	[ketua, dpp, partai, nasdem, effendi, choiri, ...]
0	KOALISI Indonesia Maju (KIM) tidak khawatir de...	MedialIndonesia	Politik	koalisi indonesia maju kim tidak khawatir deng...	[koalisi, indonesia, maju, kim, tidak, khawati...
0	WAKIL Menteri Hukum dan HAM (Wamenkumham) Edwa...	MedialIndonesia	Politik	wakil menteri hukum dan ham wamenkumham edward...	[wakil, menteri, hukum, dan, ham, wamenkumham, ...]
0	Koordinator Masyarakat Antikorupsi Indonesia (...)	MedialIndonesia	Politik	koordinator masyarakat antikorupsi indonesia m...	[koordinator, masyarakat, antikorupsi, indones...
0	KASUS kebocoran informasi Rapat Permusyawarata...	MedialIndonesia	Politik	kasus kebocoran informasi rapat permusyawarata...	[kasus, kebocoran, informasi, rapat, permusyaw...
0	KOORDINATOR Nasional Relawan Ganjar-Mahfud unt...	MedialIndonesia	Politik	koordinator nasional relawan ganjarmahfud unt...	[koordinator, nasional, relawan, ganjarmahfud...
0	Timnas Indonesia akan berlaga di Piala Asia 20...	CNN	Olahraga	timnas indonesia akan berlaga di piala asia d...	[timnas, indonesia, akan, berlaga, di, piala, ...]
0	CEO PSIS Semarang Yoyok Sukawi memastikan Wahy...	CNN	Olahraga	ceo psis semarang yoyok sukawi memastikan wahy...	[ceo, psis, semarang, yoyok, sukawi, memastika...
0	Shin Tae Yong (STY) mengungkapkan dirinya mera...	CNN	Olahraga	shin tae yong sty mengungkapkan dirinya merasa...	[shin, tae, yong, sty, mengungkapkan, dirinya...
0	Salah satu legenda Manchester City, Darius Vass...	CNN	Olahraga	salah satu legenda manchester city darius vass...	[salah, satu, legenda, manchester, city, dariu...
0	Pelatih Ekuador U-17 Diego Martinez memberikan...	CNN	Olahraga	pelatih ekuador u diego martinez memberikan pu...	[pelatih, ekuador, u, diego, martinez, memberi...
0	Pemain timnas Maroko U-17, Saifline Chlaghmo, ...	CNN	Olahraga	pemain timnas maroko u saifline chlaghmo menja...	[pemain, timnas, maroko, u, saifline, chlaghmo...

1. Melakukan tokenizing data menggunakan library nltk. Ex: 'aku suka makan ayam' -> ['aku', 'suka', 'makan', 'ayam']
2. Berdasarkan hasil yang sudah ditampilkan, tokenizing berhasil dan dapat dilihat pada kolom Tokenized.

```

!pip install Sastrawi

from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory

### Memanggil library Sastrawi untuk mendapatkan stopwords bahasa indonesia
stopword = ['dengan', 'bahwa', 'oleh']

stop = StopWordRemoverFactory()

data = stop.get_stop_words()+stopword

df2['Tokenized'] = df2['Tokenized'].apply(lambda text: [word for word in text if word not in data])

```

✓ 4.2s

Requirement already satisfied: Sastrawi in f:\anaconda\lib\site-packages (1.0.1)

1. Melakukan remove stop words dengan menggunakan library Sastrawi, hal ini dilakukan karena Sastrawi memiliki database mengenai stop words didalam bahasa Indonesia.
2. Stopword ditambahkan 3 kata tambahan yaitu ‘dengan’, ‘bahwa’, ‘oleh’.
3. Hasil yang didapatkan dimasukkan kedalam kolom Tokenized.

```

from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

def stemmer_func(text):
    factory = StemmerFactory()
    stemmer = factory.create_stemmer()
    stemmed_text_list = [stemmer.stem(text) for text in text]
    return stemmed_text_list

df2['Tokenized'] = df2['Tokenized'].apply(stemmer_func)

```

✓ 3m 49.7s

1. Melakukan stemming juga dengan library Sastrawi karena support bahasa Indonesia.
2. Stemming dimasukkan kedalam function stemmer_func yang nantinya diapply ke kolom df[‘Tokenized’].

Hasil

	Teks	Media	Label	clean_Teks	Tokenized
0	MANTAN Gubernur Jawa Barat Ridwan Kamil resmi ...	MediaIndonesia	Politik	mantan gubernur jawa barat ridwan kamil resmi ...	[mantan, gubernur, jawa, barat, ridwan, kamil, ...]
0	Timnas Indonesia akan berlaga di Piala Asia 20...	CNN	Olahraga	timnas indonesia akan berlaga di piala asia d...	[timnas, indonesia, laga, piala, asia, shin, t...
0	Badan Pelaksana Otorita Labuan Bajo Flores (...)	Kompas	Hiburan	badan pelaksana otorita labuan bajo flores bpo...	[badan, laksana, otorita, labu, bajo, flores, ...]

1. Berikut merupakan hasil tokenized, remove stopwords, dan stemming pada setiap kategori berita.

Modelling

```
# Merubah label menjadi numerik
label_mapping = {'Politik': 0, 'Olahraga': 1, 'Hiburan': 2}

df2['Label'] = df2['Label'].map(label_mapping)
```

✓ 0.0s

1. Mengubah label secara kategori menjadi numerik yaitu Politik:0, Olahraga:1, dan Hiburan:2.

```
from sklearn.model_selection import train_test_split

X = df2['Tokenized']
y = df2['Label']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state = 777)
```

✓ 0.0s

1. Train test split pada kolom Tokenized dan juga Label dengan test size sebesar 0.2

BoW

```
from sklearn.feature_extraction.text import CountVectorizer

# Membuat BoW Vectorizer
vectorizer = CountVectorizer(analyzer=lambda x: x, min_df=3, max_features=50)

# Transform text data menjadi BoW vectors
X_train_bow = vectorizer.fit_transform(X_train)

X_test_bow = vectorizer.transform(X_test)
```

✓ 0.0s

1. Menggunakan Bag of Words (BoW) untuk transform data menjadi BoW vector dengan parameter min_df = 3 dan max_features = 50.
2. Disimpan ke dalam variable X_train_bow dan X_test_bow

TF-IDF

```
from sklearn.feature_extraction.text import TfidfVectorizer

# Membuat TF-IDF Vectorizer
tfidf_vectorizer = TfidfVectorizer(min_df=3, max_features=50)

# Transform text data menjadi TF-IDF vectors
X_train_tfidf = [' '.join(tokens) for tokens in X_train]

X_test_tfidf = [' '.join(tokens) for tokens in X_test]

X_train_tfidf = tfidf_vectorizer.fit_transform(X_train_tfidf)

X_test_tfidf = tfidf_vectorizer.transform(X_test_tfidf)
```

✓ 0.0s

1. Menggunakan TF-IDF untuk transform data menjadi BoW vector dengan parameter `min_df=3` dan `max_features=50`.
2. Melakukan join pada teks yang sudah ter-tokenized karena TF-IDF tidak dapat menerima data yang sudah ter-vektor.
3. Disimpan ke dalam `X_train_tfidf` dan `X_test_tfidf`.

SVM + BoW

SVM + BoW

```
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

svm_bow = SVC(kernel='linear', C=1.0)

svm_bow.fit(X_train_bow, y_train)

### Predict
y_pred_svm_bow = svm_bow.predict(X_test_bow)
```

✓ 0.0s

Menggunakan Support Vector Machine dengan vektor BoW dan dengan test size sebesar 0.3 dan random state = 777

```
from sklearn.metrics import classification_report

svm_bow_report = classification_report(y_test, y_pred_svm_bow)
print("SVM BoW:\n", svm_bow_report)
```

✓ 0.0s

SVM BoW:

	precision	recall	f1-score	support
0	0.50	1.00	0.67	1
1	1.00	1.00	1.00	3
2	1.00	0.50	0.67	2
accuracy			0.83	6
macro avg	0.83	0.83	0.78	6
weighted avg	0.92	0.83	0.83	6

```
from sklearn.metrics import confusion_matrix
```

```
confusion_matrix(y_test, y_pred_svm_bow)
```

✓ 0.0s

```
array([[1, 0, 0],
       [0, 3, 0],
       [1, 0, 1]], dtype=int64)
```

1. Menggunakan Support Vector Machine dengan kernel linear dan $c=1.0$.
2. Menggunakan Train dan Test Set dari BoW.
3. Hasil yang didapat cukup memuaskan dengan accuracy sekitar 0.83 dan precision yang cukup bagus.
4. Hasil dilihat lebih detail lagi menggunakan confusion matrix.
5. Model dapat memprediksi semua jenis berita akan tetapi pada berita Olahraga ada yang terprediksi menjadi berita Politik.

SVM + TF-IDF

```
SVM + TF-IDF

svm_tfidf = SVC(kernel='linear', C=1.0)

svm_tfidf.fit(X_train_tfidf, y_train)

y_pred_svm_tfidf = svm_tfidf.predict(X_test_tfidf)
[16] ✓ 0.0s
```

Menggunakan Support Vector Machine dengan vektor TF-IDF dan test size = 0.3 dan random state = 777

```
svm_tfidf_report = classification_report(y_test, y_pred_svm_tfidf)
print("SVM TF-IDF:\n", svm_tfidf_report)
[17] ✓ 0.0s
```

```
... SVM TF-IDF:
              precision    recall  f1-score   support

     0           1.00        1.00        1.00         1
     1           0.75        1.00        0.86         3
     2           1.00        0.50        0.67         2

 accuracy          0.83
 macro avg          0.92
weighted avg          0.88
```

```
confusion_matrix(y_test, y_pred_svm_tfidf)
[23] ✓ 0.0s
```

```
... array([[1, 0, 0],
          [0, 3, 0],
          [0, 1, 1]], dtype=int64)
```

1. Menggunakan Support Vector Machine dengan parameter yang sama dengan sebelumnya.
2. Menggunakan Train dan Test Set dari TF-IDF.
3. Hasil yang didapat cukup memuaskan dengan accuracy 0.83.
4. Dilihat lebih detail lagi menggunakan confusion matrix, terdapat berita Hiburan yang terprediksi menjadi berita Olahraga.
5. Secara keseluruhan, model dapat memprediksi semua jenis berita.

Random Forest + BoW

Random Forest + BoW

```
from sklearn.ensemble import RandomForestClassifier

rf_bow = RandomForestClassifier(n_estimators=100, random_state=42)

rf_bow.fit(X_train_bow, y_train)

### Predict
y_pred_rfbow = rf_bow.predict(X_test_bow)
```

✓ 0.3s

Menggunakan model Random Forest dengan vektor BoW dengan test size = 0.3 dan random state = 777

```
rfbow_report = classification_report(y_test, y_pred_rfbow)
print("Random Forest BoW:\n", rfbow_report)
```

✓ 0.0s

```
Random Forest BoW:
              precision    recall  f1-score   support

     0       0.50         1.00         0.67         1
     1       0.75         1.00         0.86         3
     2       0.00         0.00         0.00         2

 accuracy          0.67
 macro avg         0.42
 weighted avg       0.46
```

```
confusion_matrix(y_test, y_pred_rfbow)
```

✓ 0.0s

```
array([[1, 0, 0],
       [0, 3, 0],
       [1, 1, 0]], dtype=int64)
```

1. Menggunakan model machine learning Random Forest dengan parameter `n_estimators=100` dan `random_state = 42`.
2. Menggunakan Train dan Test Set dari BoW
3. Berdasarkan `classification_report`, hasil yang didapatkan memiliki akurasi yang lebih buruk daripada model-model sebelumnya yaitu 0.67.
4. Dilihat pada `confusion matrix`, model tidak dapat memprediksi berita Hiburan.

Random Forest + TF-IDF

Random Forest + TF-IDF

```
rf_tfidf = RandomForestClassifier(n_estimators=100, random_state=777)

rf_tfidf.fit(X_train_tfidf, y_train)

### Predict
y_pred_rf_tfidf = rf_tfidf.predict(X_test_tfidf)
```

✓ 0.2s

Menggunakan Random Forest dengan vektor TF-IDF dengan test size = 0.3 dan random state = 777

```
rf_tfidf_report = classification_report(y_test, y_pred_rf_tfidf)
print("Random Forest TF-IDF:\n", rf_tfidf_report)
```

✓ 0.0s

```
Random Forest TF-IDF:
              precision    recall  f1-score   support

     0         1.00        1.00        1.00         1
     1         0.75        1.00        0.86         3
     2         1.00        0.50        0.67         2

 accuracy          0.83
 macro avg         0.92        0.83        0.84         6
 weighted avg      0.88        0.83        0.82         6
```

```
confusion_matrix(y_test, y_pred_rf_tfidf)
```

✓ 0.0s

```
array([[1, 0, 0],
       [0, 3, 0],
       [0, 1, 1]], dtype=int64)
```

1. Menggunakan model Random Forest yang memiliki parameter yang sama dengan sebelumnya.
2. Menggunakan Train dan Test Set dari TF-IDF.
3. Berdasarkan hasil yang didapat, akurasi memiliki nilai yang cukup memuaskan yaitu 0.83.
4. Diamati melalui confusion_matrix, model dapat memprediksi semua jenis berita akan tetapi terdapat berita Hiburan yang terprediksi berita Olahraga.

Kesimpulan

Model terbaik merupakan SVM + TF-IDF dan RF + TF-IDF, hal ini dikarenakan akurasi yang dimiliki keduanya yang baik yaitu 0.83. Teknik text representation terbaik adalah TF-IDF, konsistensi pada kedua model machine learning menjadikan TF-IDF teknik yang lebih baik dibandingkan dengan BoW.

Model dan hasil yang ada mungkin dapat ditingkatkan jika dataset yang digunakan memiliki jumlah/size yang lebih besar daripada dataset yang sekarang yaitu 30 berita dengan 10 berita pada jenis berita masing-masing. Akurasi dan Presisi dari model mungkin akan meningkat jika dapat pool training berita yang lebih besar.