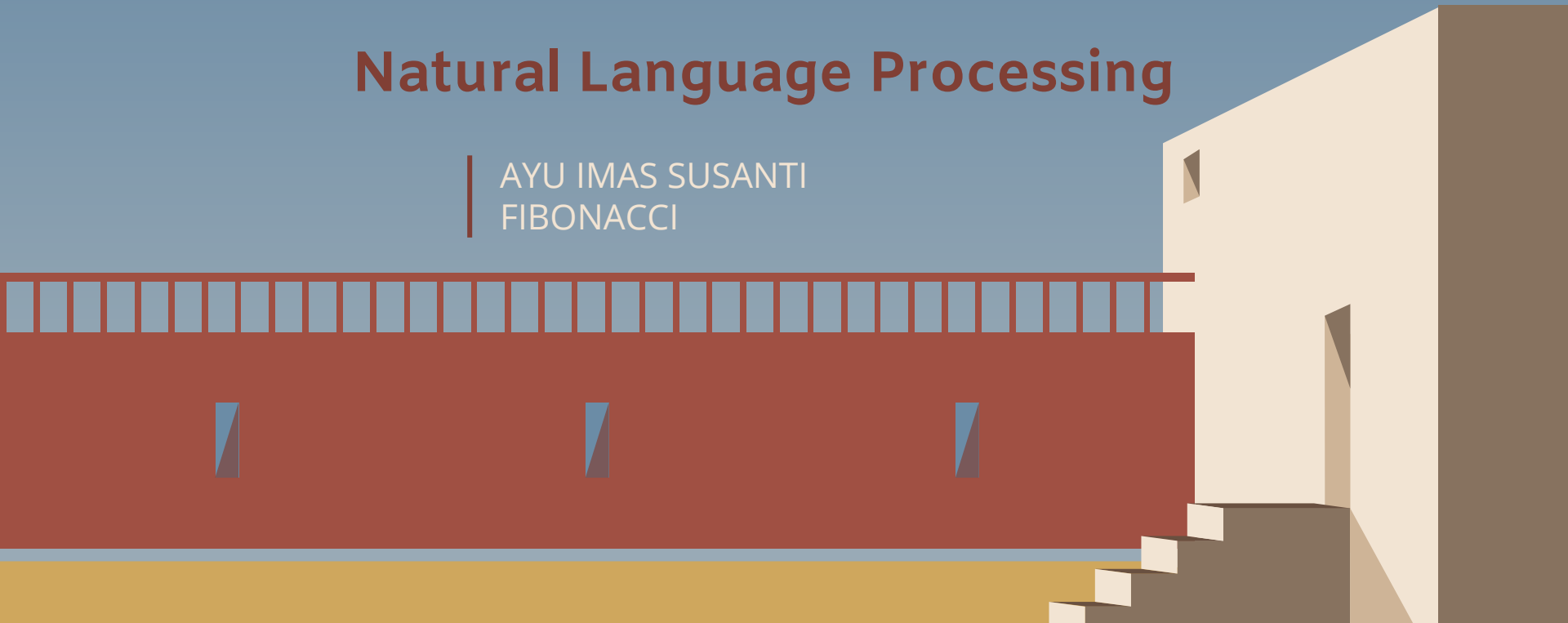
Analisis Sentimen Twitter PILKADA DKI Jakarta Tahun 2017 Menggunakan Algoritma Naive Bayes dengan Machine Learning

Natural Language Processing

AYU IMAS SUSANTI
FIBONACCI



Latar Belakang Masalah

Sentimen adalah pendapat atau pandangan yang didasarkan pada perasaan yang berlebih-lebihan. Sentimen analysis adalah proses penggunaan text analytics untuk mendapatkan berbagai sumber data dari internet dan beragam platform media social. Tujuannya adalah untuk memperoleh opini dari pengguna yang terdapat pada platform tersebut. Salah satu algoritma dalam analisis sentiment adalah metode klasifikasi Naïve Bayes, Algoritma Naïve Bayes merupakan suatu algoritma yang mempelajari probabilitas suatu objek dengan ciri-ciri tertentu yang termasuk dalam kelompok/kelas tertentu.

Sentiment tentang opini Pilkada DKI 2017 merupakan suatu opini publik pada aplikasi twitter yang dimana di dalamnya terdapat dua sentiment yaitu sentiment negative dan sentiment positive, pada analisis ini digunakan untuk mengetahui seberapa besar sentiment yang ada di masyarakat, dan kita akan mencari sentiment manakah yang lebih besar dan seberapa besar akurasi dari pengujian yang dilakukan dengan cara pengklasifikasian.

Rumusan Masalah

- **Bagaimana hasil akurasi yang didapatkan pada evaluasi modelling**
- **Apa saja parameter yang digunakan untuk pengolahan data pada tweet di twitter.**

DATASET YANG DIGUNAKAN

```
In [5]: data = pd.read_csv('dataset_tweet_sentiment_pilkada_DKI_2017.csv')
        data.head()
```

Out[5]:

	Id	Sentiment	Pasangan Calon	Text Tweet
0	1	negative	Agus-Sylvi	Banyak akun kloning seolah2 pendukung #agussil...
1	2	negative	Agus-Sylvi	#agussilvy bicara apa kasihan yaa...lap itu ai...
2	3	negative	Agus-Sylvi	Kalau aku sih gak nunggu hasil akhir QC tp lag...
3	4	negative	Agus-Sylvi	Kasian oh kasian dengan peluru 1milyar untuk t...
4	5	negative	Agus-Sylvi	Maaf ya pendukung #AgusSilvy..hayo dukung #Ani...

```
In [6]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 900 entries, 0 to 899
Data columns (total 4 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   Id              900 non-null    int64 
 1   Sentiment       900 non-null    object
 2   Pasangan Calon  900 non-null    object
 3   Text Tweet      900 non-null    object
dtypes: int64(1), object(3)
memory usage: 28.2+ KB
```

Sentiment Tweet Twitter Pilkada DKI Jakarta pada tahun 2017

PREPROCESSING DATA

```
In [21]: # Buat fungsi untuk menggabungkan seluruh langkah text preprocessing
         def text_preprocessing_process(text):
           text = casefolding(text)
           text = text_normalize(text)
           text = remove_stop_words(text)
           text = stemming(text)
           return text

In [22]: %%time
         data['clean_teks'] = data['Text Tweet'].apply(text_preprocessing_process)

         # Perhatikan waktu komputasi ketika proses text preprocessing

         CPU times: total: 4min 4s
         Wall time: 4min 9s

In [23]: data

Out[23]:

In [11]: # Download corpus kumpulan slangwords
         !wget https://raw.githubusercontent.com/ksnugroho/klasifikasi-spam-sms/master/data/key_norm.csv

         'wget' is not recognized as an internal or external command,
         operable program or batch file.

In [12]: key_norm = pd.read_csv('key_norm.csv')
         print(key_norm.head())

         key_norm.shape

In [14]: from nltk.tokenize import sent_tokenize, word_tokenize
         from nltk.corpus import stopwords

         stopwords_ind = stopwords.words('indonesian')

In [15]: len(stopwords_ind)

Out[15]: 758

In [16]: # Lihat daftar stopword yang disediakan NLTK
         stopwords_ind[:20]

In [19]: from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

         factory = StemmerFactory()
         stemmer = factory.create_stemmer()

         # Buat fungsi untuk langkah stemming bahasa Indonesia
         def stemming(text):
           text = stemmer.stem(text)
           return text
```

- Case Folding

Proses menyeragamkan karakter pada data, yaitu mengubah huruf menjadi huruf kecil

- Word Normalization

Proses menormalisasi kata yang ada di dataset.

- Filtering (Stopword Removal)

Stopwords merupakan kata yang diabaikan dalam pemprosesan dan biasanya disimpan dalam stop lists. Stopword removal adalah proses filtering, pemilihan kata-kata penting dari hasil token yang akan digunakan dalam data.

- Stemming

Proses menghilangkan imbuhan pada suatu kata

EKSTRASI DATA & SELECTION DATA

TF-IDF(Term Frequency Inverse Document Frequency) merupakan metode pembobotan untuk mengubah teks menjadi vector, TF-IDF digunakan untuk memilih fitur sebagai hasil ringkasan.

Feature Extraction (TF-IDF & N-Gram)

Proses mengubah teks menjadi vector menggunakan metode TF-IDF

Feature Selection (Chi Square)

Chi Square adalah uji komparatif nonparametric yang dilakukan pada dua variable, dimana skala data dari kedua variable tersebut adalah nominal.

MODEL Yang DIGUNAKAN

05 Modelling (Machine Learning)

```
'''
Supervised learning in Sklearn
https://scikit-learn.org/stable/supervised_learning.html
'''
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from joblib import dump
```

Naïve Bayes adalah algoritma pada Machine Learning yang digunakan untuk masalah klasifikasi. Naïve bayes merupakan metode yang sangat sesuai untuk klasifikasi, dengan menerapkan teknik supervised klasifikasi objek di masa depan dengan menetapkan label kelas ke instance / catatan menggunakan probabilitas bersyarat.

PERFORMA MODEL

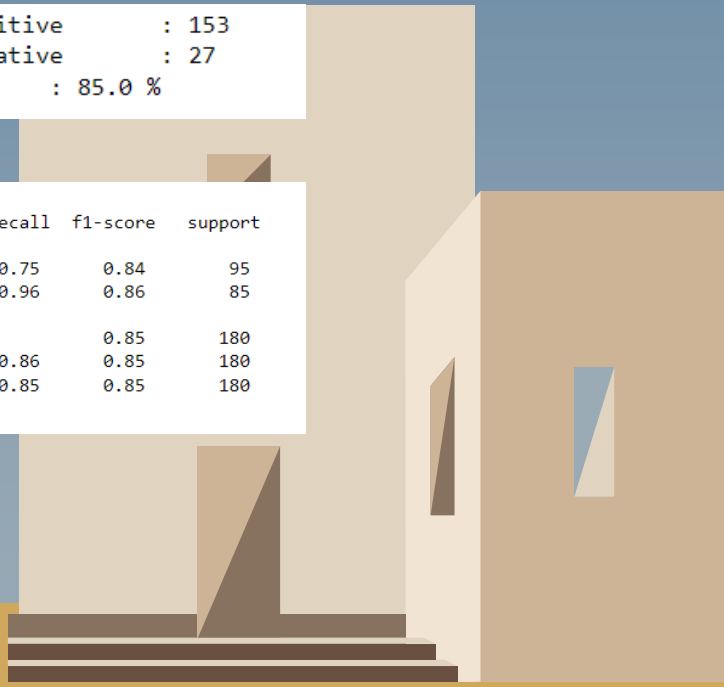
Setelah diketahui nilai x dan y yang disudah dilakukan proses train dan test kemudian diketahui nilai akurasinya.

```
Jumlah sentiment positive       : 153
Jumlah sentiment negative       : 27
Akurasi pengujian       : 85.0 %
```

```
Classification report:
              precision    recall  f1-score   support

    negative       0.96      0.75      0.84        95
    positive       0.77      0.96      0.86        85

    accuracy                           0.85       180
   macro avg       0.87      0.86      0.85       180
weighted avg       0.87      0.85      0.85       180
```



KESIMPULAN

Diketahui akurasi pengujian dari proses classification adalah sebesar 85% dengan jumlah sentiment positive lebih besar dibandingkan sentiment negative.
Classification Report dari proses ini juga diketahui seperti gambar disamping.
Nilai rata-rata akurasi pada cross validation adalah sebesar 82%.

```
Jumlah sentiment positive       : 153
Jumlah sentiment negative       : 27
Akurasi pengujian       : 85.0 %
```

```
Classification report:
              precision    recall  f1-score   support

    negative       0.96      0.75      0.84        95
    positive       0.77      0.96      0.86        85

    accuracy                           0.85       180
   macro avg       0.87      0.86      0.85       180
weighted avg       0.87      0.85      0.85       180
```

```
Akurasi setiap split: [0.77222222 0.86666667 0.81666667 0.85555556 0.83888889 0.81666667
 0.85555556 0.85       0.81666667 0.81111111]

Rata-rata akurasi pada cross validation: 0.8299999999999998
```

AYU IMAS SUSANTI
FIBONACCI

TERIMA KASIH