Article Content Classification with NLP

Text Classification
Data Science 4 - Dataloper

Abiyyu Fathin Derian
Alifia C. Harmadi
Dhea Fajriati Anas
Hendri Prabowo
Nikolas Rakryan Widagdo

TABLE OF CONTENTS

01

DATASET

Total Data Per Label

1	Bisnis	40
2	Lifestyle	40
3	Sport	40

Source: https://drive.google.com/drive/folders/13YkQF8A1bD_xNrqBv2lGk7ziM3d240Ld?usp=sharing

Labelling

Create a new column named “label” and label according to the file name

Join Data

- bisnis.csv
- lifestyle.csv
- sport.csv

Drop Column

Drop all column except “label” and “content” column

02

Text Preprocessing

- Case Folding
- Remove Whitespace
- Tokenizing
- Remove Punctuation
- Stemmer
- Remove Stopword

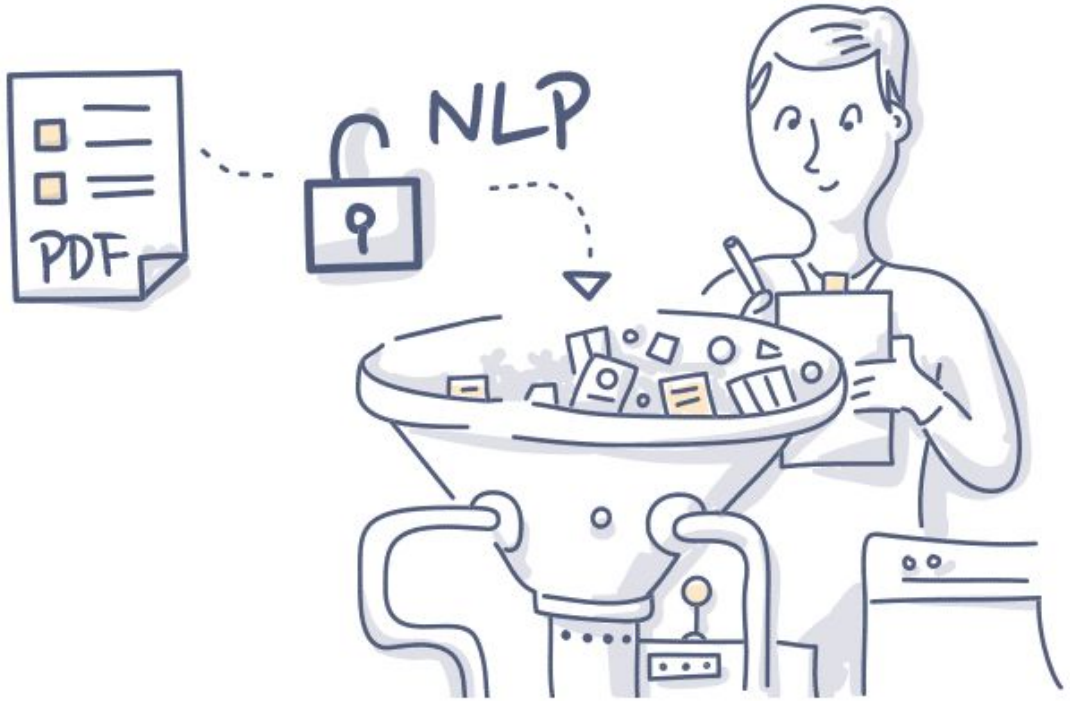
PRE-PROCESSING STEP

```
#preprocessing function
def preprocessing(sentence):
    #lowecasing
    sentence = sentence.lower()
    #remove white spaces
    sentence = sentence.strip()
    #tokenization
    words = sentence.split()
    #remove punctuation/ special character
    remove_table = str.maketrans("", "", punctuation)
    words = [x.translate(remove_table) for x in words]
    #remove nonalphanumeric <=3 chars
    #words = [x for x in words if x.isalnum() and len(x) > 3]
    words = [stemmer.stem(w) for w in words]
    #remove stopwords
    words = [x for x in words if x not in stop_words ]
    #rejoining the words
    sentence = " ".join(words)

    return sentence
```

```
before pre-processing   : 26072
after pre-processing    : 11555
delete                  : 14517
```

03
Feature Extraction



Countvectorizer

transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text

TF-IDF

evaluates how relevant a word is to a document in a collection of documents

04

Models

MODELS

Logistic Regression

Multinomial logistic regression

Support Vector Machine

Using linear kernel

Naive Bayes

Multinomial naive bayes

Random Forest

Random forest with bootstrap

The data is divided into two parts, i.e., 80% training and 20% testing

05

Evaluation

ACCURACY (Countvectorizer)

With Preprocessing

LOGISTIC REGRESSION

100%

SVM

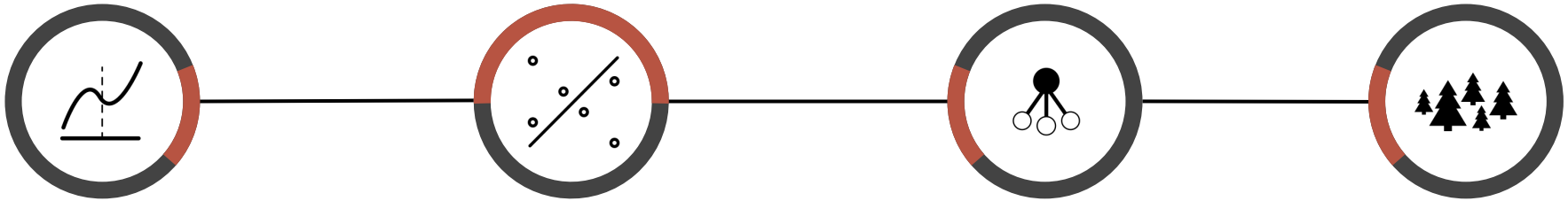
100%

NAIVE BAYES

100%

RANDOM FOREST

95.83%



91.67%

LOGISTIC REGRESSION

91.67%

SVM

100%

NAIVE BAYES

95.83%

RANDOM FOREST

Without Preprocessing

ACCURACY (TF-IDF)

With Preprocessing

LOGISTIC REGRESSION

100%

SVM

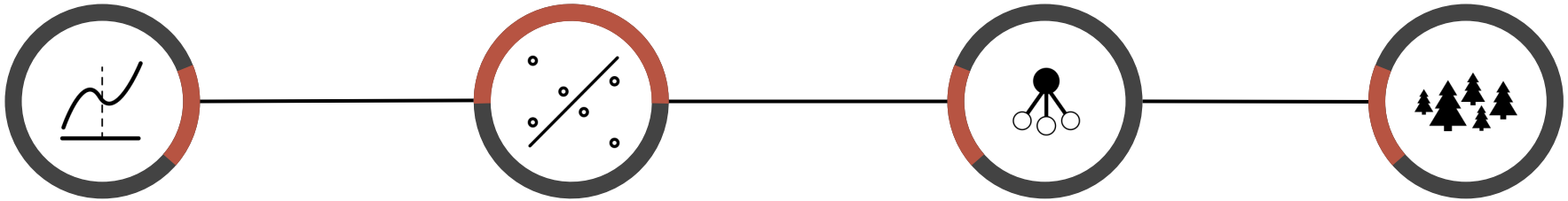
100%

NAIVE BAYES

100%

RANDOM FOREST

95.83%



95.83%

LOGISTIC REGRESSION

100%

SVM

100%

NAIVE BAYES

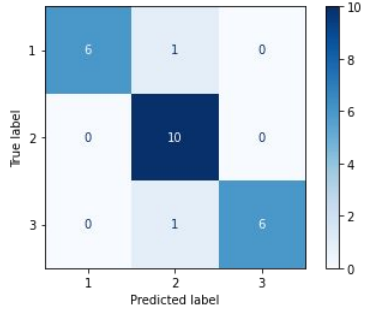
95.83%

RANDOM FOREST

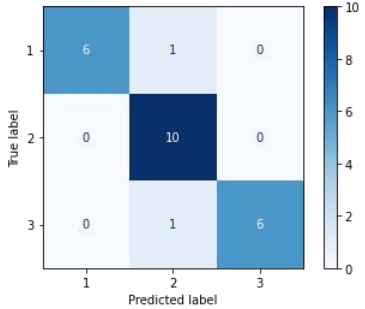
Without Preprocessing

CONFUSION MATRIX

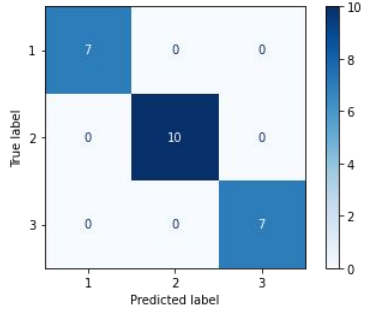
(CountVectorizer w/o Pre-processing)



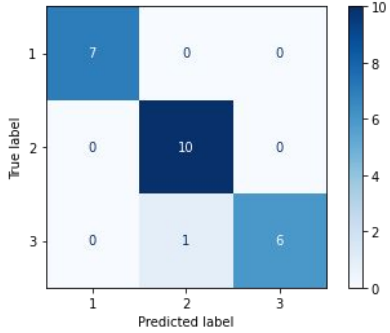
Logistic Regression



SVM



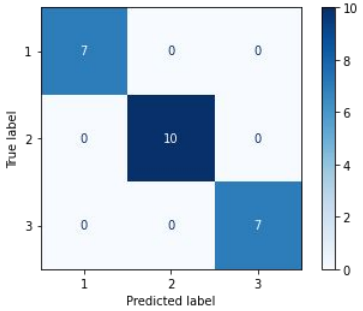
Naive Bayes



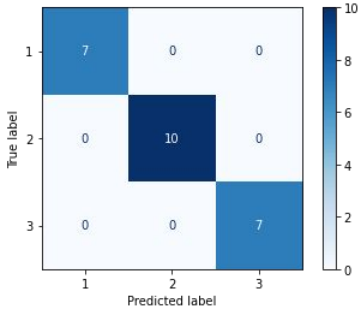
Random Forest

CONFUSION MATRIX

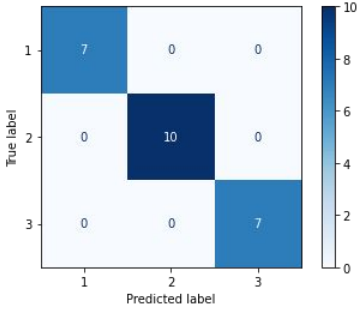
(CountVectorizer with Pre-processing)



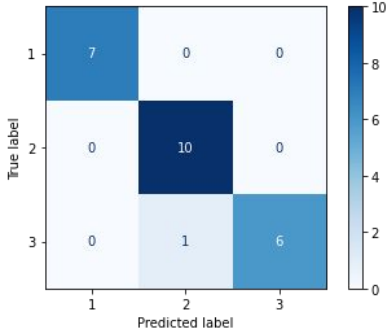
Logistic Regression



SVM



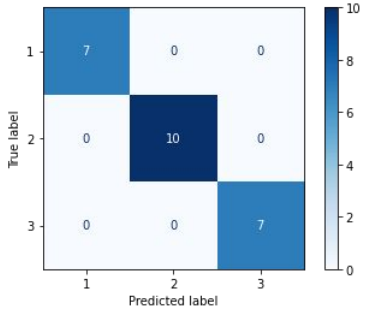
Naive Bayes



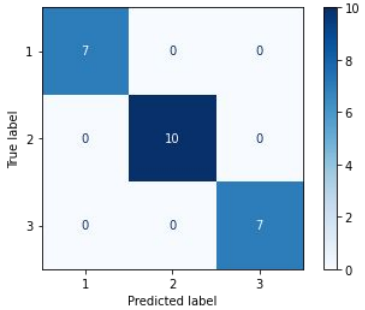
Random Forest

CONFUSION MATRIX

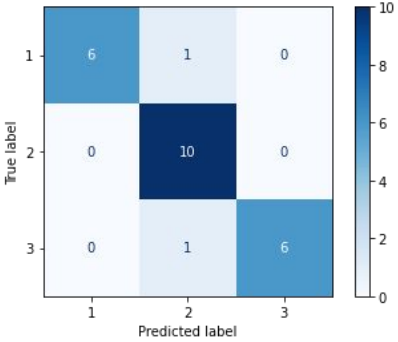
(TF-IDF w/o Pre-processing)



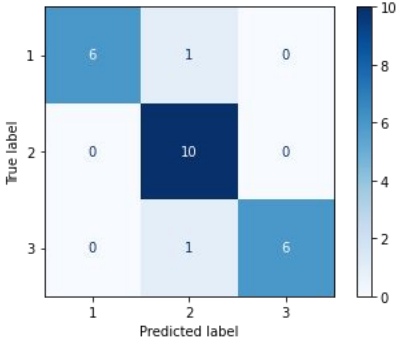
Logistic Regression



SVM



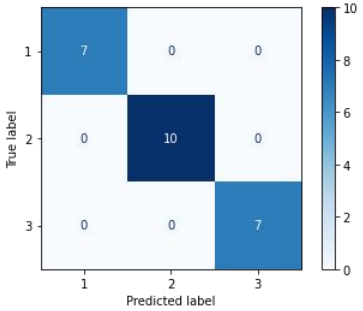
Naive Bayes



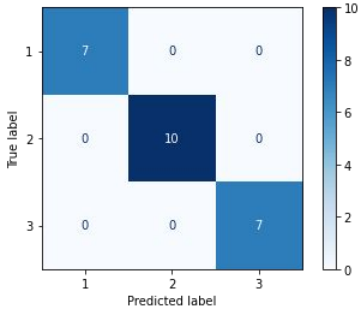
Random Forest

CONFUSION MATRIX

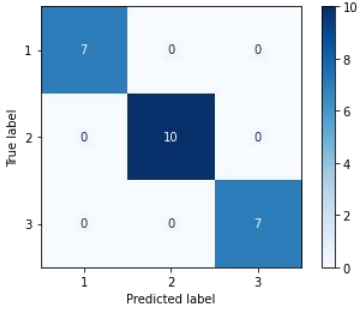
(TF-IDF with Pre-processing)



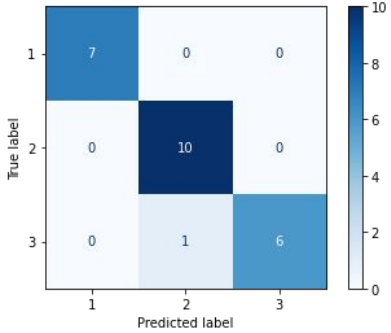
Logistic Regression



SVM



Naive Bayes



Random Forest

CLASSIFICATION REPORT (CountVectorizer)

Class	Precision	Recall	F1 Score
Logistic Regression			
1	1.00	0.86	0.92
2	0.83	1.00	0.91
3	1.00	0.86	0.92
SVM			
1	1.00	0.86	0.92
2	0.83	1.00	0.91
3	1.00	0.86	0.92
Naive Bayes			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
Random Forest			
1	1.00	1.00	1.00
2	0.91	1.00	0.95
3	1.00	0.86	0.92

w/o Pre-processing

Class	Precision	Recall	F1 Score
Logistic Regression			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
SVM			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
Naive Bayes			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
Random Forest			
1	1.00	1.00	1.00
2	0.91	1.00	0.95
3	1.00	0.86	0.92

with Pre-processing

CLASSIFICATION REPORT (TF-IDF)

Class	Precision	Recall	F1 Score
Logistic Regression			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
SVM			
1	1.00	1.00	1.00
2	0.91	1.00	0.95
3	1.00	0.86	0.92
Naive Bayes			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
Random Forest			
1	1.00	1.00	1.00
2	0.91	1.00	0.95
3	1.00	0.86	0.92

w/o Pre-processing

Class	Precision	Recall	F1 Score
Logistic Regression			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
SVM			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
Naive Bayes			
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
Random Forest			
1	1.00	1.00	1.00
2	0.91	1.00	0.95
3	1.00	0.86	0.92

with Pre-processing

06

Conclusion

DATA

Conclusion

- Based on the modelling evaluation, we found that the best model for categorizing articles in this case is **Naive Bayes** with an accuracy score **100%**
- The implementation of **TF-IDF** as feature extraction increases the accuracy of model compared to CountVectorizer from **91.67% to 95.83% for Logistic Regression** and from **91.67% to 100% for SVM**
- **Pre-processing** implementation gains the accuracy score of modelling from **91.67% to 100% with CountVectorizer for Logistic Regression and SVM** and from **95.83% to 100% with TF-IDF for Logistic Regression**


baca
cipta kerja
marc marquez
indonesia
pemerintah
motogp
prancis
valentino rossi
live streaming
listrik gratis
uu cipta
produk
ramalan zodiak
mudah
koperasi
perusahaan
pekerja
le mans
jadwal motogp
pekan minggu
ruu cipta
sirkuit le
pemain
balapan motogp
senin
kerja
halaman
digelar sirkuit
masyarakat
token listrik
dunia
omnibus law
motogp rossi
marc klok
oktober
buruh
salah
harga emas
dikutip laman
pekan
digelar

Thank You