

Analisis Sentimen Komentar *YouTube TvOne* Tentang Ustadz Abdul Somad Dideportasi Dari Singapura Menggunakan *Algoritma SVM*

Desti Mualfah¹, Ramadhoni², Rahmad Gunawan³, Danang Mulyadipa Suratno⁴

^{1,2,3}Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Muhammadiyah Riau

⁴Program Studi Teknik Industri, Sekolah Tinggi Teknologi Nasional

¹destimualfah@umri.ac.id, ²180401103@student.umri.ac.id, ³goengoen78@umri.ac.id,

⁴danangmulyadipasuratno@stitekna.ac.id

Abstract

Interactions in social media can be seen from comments as feedback from every activity on social media, starting from statuses in the form of text, images or videos. One area of computer technology that can study the meaning of text is text mining. Sentiment analysis or opinion mining is a solution to solving problems to automatically classify opinions into positive and negative. Comments from YouTube video viewers on the TvOne channel about Ustadz Abdul Somad being deported from Singapore. From the various responses in the comment column, information is obtained from unstructured data, so there is a need for a technique to define the value of information that can provide information on the number of positive and negative comments about reviews. The focus in this research is to verify the truth and explore the value of structured information so that it can describe events and topics that are connected from the comments in the YouTube videos which are the object of this research. From the test results above, it can be seen that the performance values from the test results using the Support Vector Machine method get 95.02% Accuracy, 95.02% Recall, 95.18% Precision and 95.01% F1-Score.

Keywords: *YouTube, comments, classification, video, SVM*

Abstrak

Interaksi di dalam sosial media dapat dilihat dari komentar-komentar sebagai umpan balik dari setiap kegiatan yang ada di media sosial, mulai dari status yang berupa teks, gambar maupun video. Salah satu bidang teknologi komputer yang dapat mempelajari makna teks adalah text mining. Analisis sentimen atau *opinion mining* ialah solusi menyelesaikan masalah untuk mengelompokkan opini menjadi positif dan negatif secara otomatis. Komentar – komentar penonton video youtube pada *channel* tvOne tentang Ustadz Abdul Somad di deportasi dari Singapura. Dari berbagai respon pada kolom komentar tersebut diperoleh sebuah informasi dari data yang tidak terstruktur sehingga perlu adanya suatu teknik untuk mendefinisikan nilai informasi yang dapat memberikan informasi banyaknya komentar positif dan negatif tentang ulasan. Fokus dalam penelitian ini adalah untuk memverifikasi kebenaran dan menggali nilai informasi yang terstruktur sehingga dapat menggambarkan kejadian dan topik yang terhubung dari komentar-komentaryang ada di dalam video youtube yang menjadi objek penelitian ini. Dari hasil pengujian di atas dapat dilihat nilai performa dari hasil pengujian menggunakan metode *Support Vector Machine* mendapatkan Akurasi 95.02%, Recall 95.02%, Precision 95.18% dan F1-Score 95.01%

Kata kunci: *YouTube, komentar, klasifikasi, video, SVM*

©This work is licensed under a Creative Commons Attribution - ShareAlike 4.0 International License

1. Pendahuluan

Youtube merupakan sebuah media sosial yang memiliki konten berbentuk video. Dimana, videonya bertujuan sebagai hiburan, pengetahuan dan lain sebagainya. Pengguna youtube juga dapat membuat, melihat, serta berbagi video secara gratis. Di youtube juga disediakan fitur komentar sebagai penyalur *respons viewers* (yang menonton video) terhadap konten video yang di tontonnya. Komentar tersebut bisa bersifat positif atau negatif. Disisi lain situs youtube juga dapat memberikan umpan balik (*feedback*) berupa *like*, *dislike*, dan komentar terhadap video yang ditayangkan di media youtube [1].

Selain itu youtube juga dimanfaatkan sebagai media penyebaran dakwah serta informasi oleh pemuka

agama khususnya yang ada di indonesia pada saat ini, salah satunya ialah Ustadz Abdul Somad. Ustad Abdul Somad adalah salah satu ulama atau pendakwah yang terkenal di Indonesia. Ustad Abdul Somad sering membahas berbagai macam persoalan agama. Khususnya pada bidang kajian ilmu hadis dan juga ilmu fiqih. Selain itu, Ustadz Abdul Somad juga banyak membahas mengenai nasionalisme dan berbagai masalah terkini yang sedang menjadi pembahasan hangat dikalangan masyarakat (Aswir and Misbah 2018). Seperti yang terjadi pada saat ini bahwa Ustadz Abdul Somad mengalami masalah terkait penolakan beliau oleh keimigrasian negara singapura untuk liburan ke negara tersebut [2].

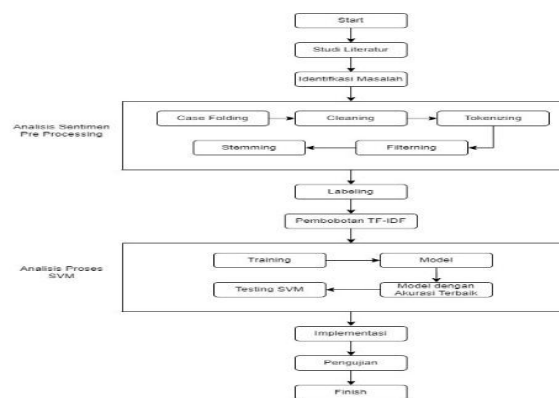
Di sini penulis akan meneliti tentang analisis *sentiment* tentang Ustadz Abdul Somad di deportasi dari singapura melalui *channel* Youtube tvOne news. *Channel* Youtube tvOne news sendiri memiliki jumlah pengikut sebanyak 9,91 juta *subscriber* per tanggal 1 Desember 2022 sejak bergabung pada tanggal 4 desember 2014 dan total *viewers* videonya berjumlah 6,132,720,633 di Youtube. dengan jumlah tersebut menggambarkan bahwa *channel* ini menduduki peringkat ke 8 dengan *subscribe* terbanyak dari 12 stasiun televisi swasta di Indonesia. Banyak mendapatkan respon dari masyarakat yang menontonnya menjadikan *channel* ini sebagai salah satu *channel* terpopuler di Indonesia tentunya dengan tingkat antusiasme yang tinggi, membuat *channel* ini mendapatkan begitu banyak respon dari masyarakat. Hal ini akan berguna bagi Ustadz Abdul Somad untuk mendapatkan informasi yang ada berupa ulasan komentar chanel Youtube tvOne untuk mendapatkan *feedback* berupa komentar positif maupun negatif dari masyarakat secara langsung tentang permasalahan yang terjadi terhadapnya. Hal ini tentunya akan menyulitkan jika harus menganalisis setiap komentar secara manual, karena komentar berisikan ratusan hingga ribuan tanggapan dari setiap penonton di video tersebut [3]. Komentar yang diberikan oleh seseorang tidak selalu dapat dimengerti oleh orang lain. Maka diperlukan bidang teknologi komputer yang bertujuan untuk memahami makna teks tersebut.

Penulis melakukan penelitian mengenai klasifikasi komentar terhadap ustadz abdul somad di deportasi dari singapura pada youtube *channel* tvOne, komentar tersebut perlu dilakukan klasifikasi menggunakan *machine learning*, dengan waktu sebentar saja sudah didapatkan ratusan maupun ribuan komentar pada sosial media youtube, terutama pada video *channel* youtube tvOne tentang video Ustadz Abdul Somad di deportasi dari singapura tersebut. Sehingga membutuhkan waktu lebih lama untuk dilakukan klasifikasi secara manual oleh manusia, dengan begitu penulis menggunakan *machine learning* untuk diterapkannya metode *Support Vector Machine* (SVM). “*Support Vector Machine* (SVM) merupakan metode *supervised learning* yang menganalisa data dan mengenali pola-pola yang digunakan untuk klasifikasi[4].` Digunakannya metode *Support Vector Machine* (SVM) sebab metode ini belum pernah dilakukan penelitian dengan kasus yang sama, ditambah lagi dengan metode ini dikabarkan adalah metode terbaik[5]. Dari berbagai penjelasan latar belakang yang sudah dipaparkan, dirumuskan tugas akhir “Analisis Sentimen Komentar Youtube TvOne Tentang Ustadz Abdul Somad Di Deportasi Dari Singapura Menggunakan Algoritma SVM”. [6].

2. Metode Penelitian

Metodologi disini disebut juga sebagai tahapan pengerjaan yang dilakukan peneliti supaya tersusun

secara baik. Tahap-tahap penelitian yang penulis lakukan dijelaskan pada gambar 1 dibawah ini.



Gambar 1. Tahapan Penelitian

2.1 Studi Literatur

Studi pustaka ialah tahapan setelah dari identifikasi masalah. Pada tahapan ini menjelaskan bagaimana mengumpulkan data dan informasi dari berbagai sumber yang ada. pencarian referensi tersebut meliputi Jurnal, buku, *e-book*, media massa, referensi lainnya yang berkaitan dengan penelitian ini baik berupa teori serta penelitian serupa yang telah dilakukan [10].

2.2 Identifikasi Masalah

Langkah pertama metodologi penelitian agar mempelajari, mengidentifikasi dan menemukan solusi suatu permasalahan serta tujuan dari penelitian ini. Adapun identifikasi permasalahan terhadap penelitian ini bagaimana penerapan metode *Support Vector Machine* untuk mengklasifikasi komentar terhadap ustadz abdul somad di deportasi dari singapura pada video *channel* tvOneNews di youtube.

2.3 Pengumpulan Data

Tahap ini, diperlukan pengumpulan data agar menganalisa dan memperoleh data serta informasi yang berhubungan terhadap penelitian. Data yang didapatkan merupakan komentar video pada media sosial youtube *channel* tvOneNews [11].

Pengumpulan data agar menganalisa dan memperoleh data serta informasi yang berhubungan terhadap penelitian. Data yang didapatkan merupakan komentar video pada media sosial youtube *channel* tvOneNews. Data komentar diambil menggunakan metode *Scraping*. *Tool* yang digunakan dalam proses *Web Scraping* yaitu *google colabraty*. Pengambilan data dengan cara *copy link* video dan data akan *download*, jika sudah berhasil lalu salin ke Ms. Excel karena data tidak akan diolah secara keseluruhan, hanya akan digunakan beberapa data yang dibutuhkan seperti *username* dan *Comment text* [12].

Kategori data yang digunakan sbagai berikut:

1. Kategori Data

Penelitian menggunakan data yang sudah dikelompokkan dalam 2 kategori (*class*) komentar yaitu positif dan negatif. :

a. Kategori positif merupakan komentar yang memiliki arti positif dan mampu mengajak pembaca untuk berfikir atau berpendapat positif terhadap ustadz abdul somad seperti pujian dan dukungan.

b. Kategori negatif merupakan komentar yang memiliki arti negatif dan mampu mengajak pembaca untuk berfikir atau berpendapat negatif terhadap ustadz abdul somad. Biasanya, komentar ini muncul ketika adanya bentuk kekecewaan terhadap ustadz abdul somad. Komentar tersebut seperti sindiran, kata kotor, penolakan, sanggahan dan cacian atau makian.

2. Pembagian Data

Pada penelitian ini, data komentar yang dapat akan dibagi mejadi data latih dan data uji. Dengan memakai perbandingan 70% : 30%, 80 % : 20 % dan 90 % : 10 % [13].

2.4 Labeling

Labelling merupakan tahap dimana hasil dari tahapan sebelumnya akan dilakukan perhitungan terhadap *polarity* dari ulasan yang terambil, sehingga dapat menghasilkan dua kategori yaitu label positif dan negatif.

2.5 Analisa

2.5.1 Analisa Kebutuhan Data

Untuk melakukan tahap ini, berikut beberapa langkah dalam pengerjaannya:

a. Analisis data dilakukannya tahap ini untuk menganalisis data yang telah peneliti dapatkan dari hasil *Scrapping* Data komentar dari youtube *channel* tvOneNews.

b. Pelabelan *Text* dilakukan dengan menggunakan *google colaboraty* 29.122 komentar.

c. Analisis *Text Pre-processing* Secara umum dilakukan pada tahapan penelitian ini yang dipakai ada 5 tahap yaitu *Cleaning*, *Case Folding*, *Tokenizing*, *Filtering* dan *Stemming* [14].

2.5.2 Analisa *Pre-processing*

Pre-processing merupakan sebuah proses dalam klasifikasi yang bertujuan untuk membersihkan dan mempersiapkan data supaya dapat dioalah pada klasifikasi tahap selanjutnya (Haddi, Liu, & Shi, 2013). Pada penelitian ini *text processing* yang dilakukan yaitu *Tokenizing*, *Case Folding*, *Filtering* dan *Stemming*.

2.6 Implementasi

Tahap implementasi disebut tahap kegiatan untuk menyelesaikan rancangan penelitian yang dibuat. Pernagkat yang diperlu ialah:

Sistem Operasi : Windows 8.1 Enterprise
Bahasa Pemograman : Python 3.6 LibrarySVM:nltk, csv, sastrawi, pandas, numpy, re, pickle,matplotlib
Web browser : Mozilla Firefox

2.7 Pengujian

2.7.1 Pengujian Algoritma

Pengujian ini menggunakan pengujian dengan Algoritma/System. Yang bertujuan supaya dapat melihat fungsi aplikasi berjalan lancar atau tidak [15].

2.7.2 Pengujian Akurasi

Pengujian ini dilakukan untuk menentukan tingkat akurasi komentar terhadap ustadz abdul somad di deportasi dari singapura yang didapatkan dengan pengujian parameter menggunakan metode *ConfussionMatrix*.

2.8 Kesimpulan

Tahap berikut berisi kesimpulan akhir dari hasil pengerjaan penelitian tentang klasifikasi komentar terhadap ustadz abdul somad di deportasi dari singapura pada media sosial youtube menggunakan metode SVM. Kemudian, saran berguna diberikan pada penelitian selanjutnya [16].

3. Hasil dan Pembahasan

3.1 Introduction

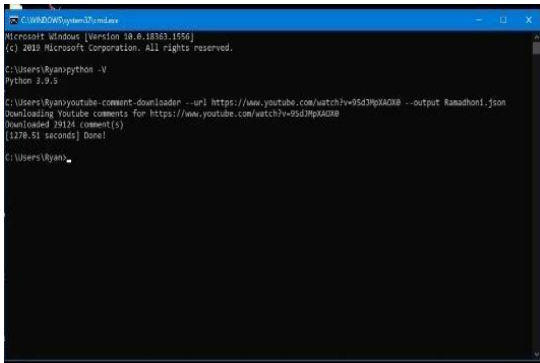
Pada bagian ini akan dipaparkan hasil penelitian berdasarkan tahapan yang telah dijelaskan di metode penelitian pada bab sebelumnya. Tahapan tersebut mulai dari persiapan *dataset*, proses *text-processing*, *labelling*, *training* model klasifikasi menggunakan algoritma *Support Vector Machine* hingga evaluasi performa model dengan beberapa skenario yang telah ditentukan.

3.2 Analisis Sentimen Komentar Video Di Youtube

Pada proses ini akan menjelaskan tahapan analisis sentimen komentar video di *channel* youtube tvone tentang ustadz abdul somad menggunakan algoritma *Support Vector Machine*.

3.2.1 Pengumpulan *Dataset*

Berdasarkan proses yang telah dijelaskan pada Bab 3 diatas, menjelaskan bahwa pengumpulan data yang digunakan pada penelitian ini adalah data komentar video di *channel* youtube tvone tentang ustadz abdul somad. Pengumpulan data dilakukan dengan menggunakan python dengan fungsi *youtube-Comment-downloader*. Berikut sintak yang digunakan dapat dilihat pada gambar 2 di bawah ini:



Gambar 2. Scraping Comment Youtube

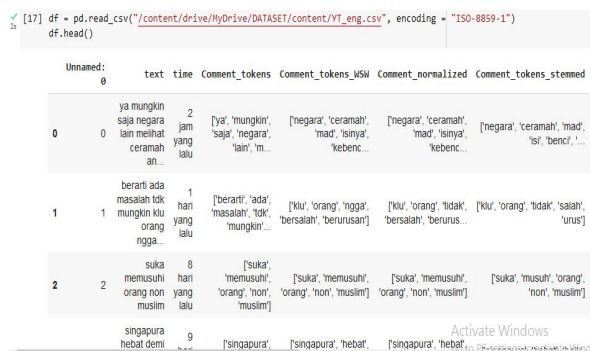
Pengumpulan data dilakukan dengan proses *scraping* dan didapat data sebanyak 29.122 data komentar saat melakukan *scraping* pada video Youtube dengan format file .json. file tersebut kemudian di *convert* menjadi csv [17].

3.2.2 Text Preprocessing

Text preprocessing dilakukan pada kolom “text” berisikan baris data berupa kalimat yang terdiri atas beberapa kata-kata. Agar *dataset* dapat digunakan untuk analisis sentimen, maka harus dilakukan beberapa tahapan *text preprocessing*, Berikut ini merupakan tahapan dari *text preprocessing* yaitu:

3.2.3 Case Cleaning

Sebelum melakukan tahapan *Text preprocessing* lakukan *load dataset* menggunakan *library* panas, data yang akan digunakan hanya data pada kolom *text* yang berisikan data komentar dan data *time* yang berisikan keterangan waktu komentar di terbitkan. Berikut sintak *load dataset* dan hasilnya seperti gambar berikut :



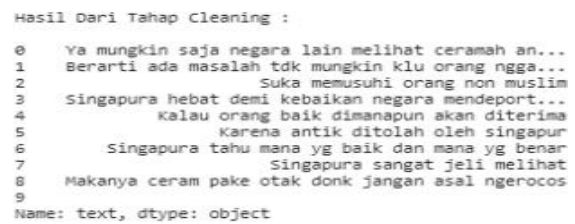
Gambar 3. Load Dataset

Setelah dilakukannya proses *load* data dan menampilkan dataset komentar youtube, kemudian masuk ke tahapan *case cleaning* dengan tujuan membersihkan atau mengubah data teks sehingga lebih mudah diproses oleh model analisis. *Case Cleaning* melibatkan sejumlah teknik seperti penghapusan tanda baca Menghapus tanda baca seperti `(!\"#$%&'()*+,-./:;<=>?@[\\]^_`{|}~)`, penghapusan karakter yang tidak diperlukan seperti angka atau simbol yang diterapkan pada data kolom *text* yang berisi *Comment* youtube. Berikut sintak *case cleaning* pada gambar 4 di bawah ini :



Gambar 4. Case Cleaning

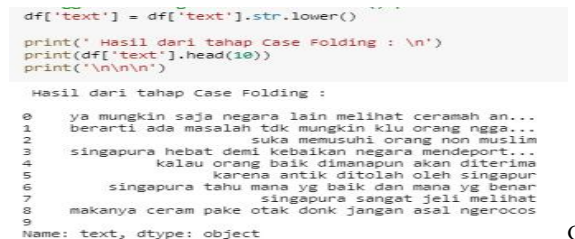
Dapat dilihat pada gambar dibawah hasil dari *case cleaning* data sudah bersih dari simbol, karakter, tanda baca yang tidak diperlukan, berikut hasilnya pada gambar 5 di bawah :



Gambar 5. Hasil Case Cleaning

3.2.4 Case Folding

Dapat dilihat pada gambar 6 berikut merupakan sintak *case folding* dan hasil dari tahapan *case folding* terlihat semua huruf sudah menjadi huruf kecil [18].



Gambar 6. Case folding

3.2.5 Tokenizing

Tahapan ini menggunakan *library Natural Language Toolkit* atau disingkat NLTK, merupakan *library* python untuk bekerja dengan memproses *tokenizing* pemodelan pada teks Berikut pada gambar 7 merupakan sintak *tokenizing*.



Gambar 7. Tokenizing

3.2.6 Filtering Stopword

Pada tahap ini dilakukan proses *filtering stopwords*, yang mana *stopword* merupakan kata-kata umum yang sering muncul dalam sebuah teks dan biasanya tidak memiliki makna atau kontribusi penting dalam analisis teks. Contoh *stopword* dalam bahasa Indonesia adalah "yang", "dalam", "pada", "di", dan sebagainya. Berikut hasil dari Filtering dilihat pada gambar 8 terlihat kata-kata yang termasuk dalam kategori *stopword* sudah dihapus contohnya yang, akan, dgn dan lain-lain.

```

from nltk.corpus import stopwords
import nltk
nltk.download('stopwords')

# ----- GET STOPWORD FROM NLTK STOPWORD -----
# get stopwords Indonesia
list_stopwords = stopwords.words('indonesian')
print(list(list_stopwords))
# ----- manually add stopword -----
# append additional stopword
list_stopwords.extend(['yg', 'dgn', 'pda', 'di', 'ng', 'd', 'klu',
                       'klu', 'ngga', 'bersalah', 'berurusan',
                       'suka', 'memusuhi', 'orang', 'non', 'muslim',
                       'singapura', 'hebat', 'kebaikan', 'negara', 'mendepor...',
                       'orang', 'dimanapun', 'diterima',
                       'antik', 'ditolah', 'singapur',
                       'singapura',
                       'ceram', 'pake', 'otak', 'donk', 'ngerosos',
                       'jeli'])

len(list_stopwords)
# ----- add stopword from txt file -----
# read txt stopword using pandas
txt_stopword = pd.read_csv('stopwords.csv', names= ['stopwords'], header = None)
# convert stopword string to list & append additional stopword
list_stopwords.extend(txt_stopword['stopwords'][0].split(' '))
len(list_stopwords)
# convert list to dictionary
list_stopwords = set(list_stopwords)
# remove stopword pada list token
def stopwords_removal(words):
    return [word for word in words if word not in list_stopwords]
df['Comment_tokens_NSW'] = df['Comment_tokens'].apply(stopwords_removal)
    
```

Gambar 8. Filtering Stopword

Berikut hasil teks setelah dilakukannya proses *filtering stopwords*. Dapat dilihat pada gambar 9.

```

758
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data] Unzipping corpora/stopwords.zip.
Hasil dari Tahap Filtering :

0 [negara, ceramah, mad, isinya, kebencian, meng...
1 [klu, orang, ngga, bersalah, berurusan]
2 [suka, memusuhi, orang, non, muslim]
3 [singapura, hebat, kebaikan, negara, mendepor...
4 [orang, dimanapun, diterima]
5 [antik, ditolah, singapur]
6 [singapura]
7 [singapura, jeli]
8 [ceram, pake, otak, donk, ngerocos]
9 []
Name: Comment_tokens_NSW, dtype: object
    
```

Gambar 9. Hasil Filtering Stopword

3.2.7 Normalisasi

Fungsi normalisasi adalah untuk meningkatkan kualitas representasi kata dalam dokumen teks sehingga analisis dapat dilakukan dengan lebih akurat dan efisien. Berikut tahapan normalisasi dapat dilihat pada gambar 10 pada tahapan ini melakukan normalisasi pada dataset menggunakan file normalisasi yang nantinya akan diterapkan pada *dataset*.

```

[11] #normalisasi
#untuk mengubah teks kalimat menjadi teks yang secara lengkap memperlihatkan cara pengucapannya
normalizad_word = pd.read_csv("../content/drive/MyDrive/DATASET/content/normalisasi.csv")
normalizad_word_dict
for index, row in normalizad_word.iterrows():
    if row[0] not in normalizad_word_dict:
        normalizad_word_dict[row[0]] = row[1]

def normalizad_term(document):
    return [normalizad_word_dict[term] if term in normalizad_word_dict else term for term in document]
df['Comment_normalized'] = df['Comment_tokens_NSW'].apply(normalizad_term)

print('Hasil dari Tahap Normalization : \n')
print(df['Comment_normalized'].head(10))
print('\n\n')

Hasil dari Tahap Normalization :

0 [negara, ceramah, mad, isinya, kebencian, meng...
1 [klu, orang, tidak, bersalah, berurusan]
2 [suka, memusuhi, orang, non, muslim]
3 [singapura, hebat, kebaikan, negara, mendepor...
4 [orang, dimanapun, diterima]
5 [antik, ditolah, singapur]
6 [singapura]
    
```

Gambar 10. Normalisasi

3.2.8 Stemming

Fungsi *stemming* adalah untuk mengurangi variasi kata dalam teks, sehingga mengurangi dimensi atau jumlah kata yang harus diproses dalam analisis teks. *Stemming* ini menggunakan *library* sastrawi yang mana teknik ini didasarkan pada aturan morfologi Bahasa Indonesia, dan dirancang khusus untuk mengatasi permasalahan *stemming* pada bahasa Indonesia. Berikut sintak *stemming* yang digunakan :

```

from sastrawi.Stemmer.StemmerFactory import StemmerFactory
import swifter
# create stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

def stemmed_wrapper(term):
    return stemmer.stem(term)
term_dict = {}
for document in df['Comment_normalized']:
    for term in document:
        if term not in term_dict:
            term_dict[term] = 1
            print("-----")

for term in term_dict:
    term_dict[term] = stemmed_wrapper(term)
    print(term, ":", term_dict[term])
    print("-----")

# apply stemmed term to dataframe
def get_stemmed_term(document):
    return [term_dict[term] for term in document]
df['Comment_tokens_stemmed'] = df['Comment_normalized'].swifter.apply(get_stemmed_term)
    
```

Gambar 11. Stemming Sastrawi

Berikut hasil stemming pada gambar 12 teks yang telah di proses oleh library sastrawi :

```

758
[nltk_data] Downloading package stopwords to /root/nltk_data...
Hasil dari Tahap Filtering :

0 [negara, ceramah, mad, isinya, kebencian, meng...
1 [klu, orang, ngga, bersalah, berurusan]
2 [suka, memusuhi, orang, non, muslim]
3 [singapura, hebat, kebaikan, negara, mendepor...
4 [orang, dimanapun, diterima]
5 [antik, ditolah, singapur]
6 [singapura]
7 [singapura, jeli]
8 [ceram, pake, otak, donk, ngerocos]
9 []
Name: Comment_tokens_NSW, dtype: object
print('Hasil dari Tahap Stemming : \n')
print(df['Comment_tokens_stemmed'])

mnumbalkn : mnumbalkn
guran : guran
bolongkarna : bolongkarna
merica : merica
penyedap : sedap
rasahanya : rasahanya
pirman : pirman
korupsitu : korupsitu
oersi : oersi
haf : haf
dikittulisan : dikittulisan
ceker : ceker
dilawanlah : lawan
    
```

Gambar 12. Hasil Stemming Sastrawi

3.2.9 Labeling

Pada tahap ini dilakukan untuk memberikan kategori sentimen pada data, dimana pada penelitian ini data ulasan dibagi menjadi dua kategori kelas yaitu positif dan negatif. Sebelum melakukan pelabelan data teks sentimen diubah terlebih dahulu menjadi bahasa Inggris dikarekan pada proses pelabelan menggunakan *library TextBlob*. Berikut sintak *import TextBlob* dapat dilihat pada gambar 13 [19]:

```

import preprocessor as p
from textblob import TextBlob
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize
    
```

Gambar 13. Import Library Labeling

Kemudian membuat fungsi untuk mendapatkan *subjectivity*, *fungsiion* untuk mendapatkan *polarity* dan *fungsiion* untuk mengkonversi hasil dari *score*. Jika *score* kurang dari 0 maka negative, apabila lebih dari nol maka positif [20].

```
def getSubjectivity(review):
    return TextBlob(review).sentiment.subjectivity

def getPolarity(review):
    return TextBlob(review).sentiment.polarity

def analyze(score):
    if score < 0:
        return 'Negatif'
    else:
        return 'Positif'

df['subjectivity'] = df['comment_eng'].apply(getSubjectivity)
df['polarity'] = df['comment_eng'].apply(getPolarity)
df['textblob'] = df['polarity'].apply(analyze)
```

Gambar 14. Penerapan Fungsi Labeling

Untuk mendapatkan hasil *labeling* yang di ambil hanya kolom *Comment_eng* saja, kemudian masukkan kodingan pelabelan *df subjectivity* untuk *score* kolom *subjectivity*, *df polarity* untuk *score* kolom *polarity* dan *df textblob* untuk analisa *score* kolom *TextBlob/sentiment* dimana akan mendapatkan hasil *negative* dan positif dapat dilihat pada gambar 15.

id	text	type	comment_tokens	comment_tokens_ju	comment_normaliz	comment_tokens_stemmed	comment_ok	comment_eng	subjectivity	polarity	textblob
0	ternyata ini bagus saya sudah pernah di...	Posif	['nyata', 'ini', 'bagus', 'saya', 'sudah', 'pernah', 'di...']	['nyata', 'ini', 'bagus', 'saya', 'sudah', 'pernah', 'di...']	['nyata', 'ini', 'bagus', 'saya', 'sudah', 'pernah', 'di...']	['nyata', 'ini', 'bagus', 'saya', 'sudah', 'pernah', 'di...']	1	ternyata ini bagus saya sudah pernah di...	0.99	0.7124	Negatif
1	barangnya bagus saya suka banget saya...	Posif	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	1	barangnya bagus saya suka banget saya...	1.00	0.2800	Positif
2	barangnya bagus saya suka banget saya...	Posif	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	1	barangnya bagus saya suka banget saya...	1.00	0.0000	Positif
3	barangnya bagus saya suka banget saya...	Posif	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	1	barangnya bagus saya suka banget saya...	0.95	0.7900	Positif
4	barangnya bagus saya suka banget saya...	Posif	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	['barangnya', 'bagus', 'saya', 'suka', 'banget', 'saya...']	1	barangnya bagus saya suka banget saya...	1.00	0.0000	Positif

Gambar 15. Hasil Labeling

Dari pelabelan menggunakan *TextBlob* mendapatkan hasil *sentiment* positif 23568 komentar dan negatif 5554 komentar Berikut total hasil *labeling* sentimen positif dan negatif pada gambar 16 di bawah yang telah diproses oleh *TextBlob* :

```
print (df.textblob.value_counts())

Positif      23568
Negatif      5554
Name: textblob, dtype: int64
```

Gambar 16. Total Hasil Labeling

3.2.10 Pembobotan TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF) adalah pembobotan yang dilakukan untuk mendapatkan nilai dari suatu kata yang berhasil di ekstrak. Pada tahap TF-IDF ini, berikut dilihat pada gambar 17 perhitungan dilakukan menggunakan modul *scikit-learn* yaitu *TfidfVectorizer*. Berikut sintak melakukan proses TF-IDF dan hasil dari pembobotan TF-IDF.

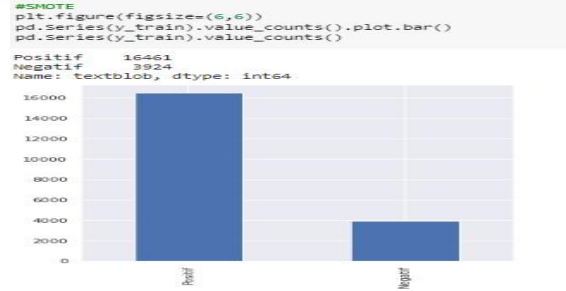
```
#Tahap TF-IDF
from sklearn.feature_extraction.text import TfidfVectorizer
X = APP_data['comment_ok'].values.astype('U')
label = APP_data['textblob']
tfidf_vectorizer = TfidfVectorizer()
tfidf_vector = tfidf_vectorizer.fit_transform(X)
print("Bentuk Data Hasil dari Tahap TF-IDF :")
print(tfidf_vector.shape)

Bentuk Data Hasil dari Tahap TF-IDF :
(29122, 32161)
```

Gambar 17. Pembobotan TF-IDF

3.2.11 SMOTE

Pada Tahapan ini mengatasi masalah *imbalance* pada *dataset* menggunakan *over sampling* SMOTE dari *imblearn library*. Berikut tampilan data sebelum di SMOTE :



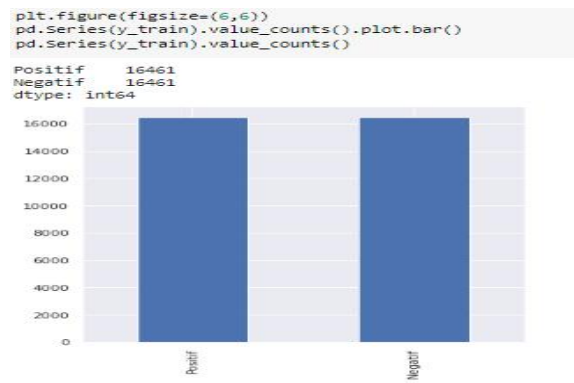
Gambar 18. Sebelum SMOTE

Pada gambar 18 menunjukkan *dataset* yang memiliki *class label* yang tidak seimbang, label positif jauh lebih dominan daripada label negatif. Menggunakan *imbalance dataset* terkadang membuat model yang dibangun akan bias pada satu *class* saja yaitu mayoritas *class*, untuk itu pada penelitian ini terlebih dahulu sebelum membangun model melakukan penyeimbangan *class* untuk setiap label dengan menggunakan teknik *OverSampling* [21].

```
from imblearn.over_sampling import SMOTE
sm = SMOTE(random_state = 33)
X_train, y_train = sm.fit_resample(X_train, y_train.ravel())
```

Gambar 19. SMOTE

Pada gambar 19 melakukan *oversample* dengan teknik SMOTE pada *dataset* dengan menduplikat kelas minoritas sehingga total sama dengan kelas mayoritas, berikut hasil tiap label setelah dilakukannya *oversample* dengan teknik SMOTE :



Gambar 20. Hasil SMOTE

Setelah dilakukannya SMOTE pada *dataset* dilihat pada gambar 20 terjadi perubahan dimensi dalam pada label negatif sebagai minoritas yang sebelumnya 3924 menjadi 16461 dikarenakan SMOTE membuat instance data baru dalam kelas minoritas[22].

3.2.12 Splitting Data

Tahapan ini melakukan *splitting* data atau pembagian data pada *X_train* dan *y_train* dengan rasio 70/30 yaitu 70% untuk data *training* dan 30% untuk data *testing*, berikut sintak :

```
#Split Dataset
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, shuffle=True, random_state=42)
print("Hasil Nilai X Train : ", X_train.shape)
print("Hasil Nilai X Test : ", X_test.shape)
print("Hasil Nilai Y Train : ", y_train.shape)
print("Hasil Nilai Y Test : ", y_test.shape)

Hasil Nilai X Train : (20385, 32161)
Hasil Nilai X Test : (8737, 32161)
Hasil Nilai Y Train : (20385,)
Hasil Nilai Y Test : (8737,)
```

Gambar 21. Splitting Data

Tahapan *splitting* data menghasilkan data X_{train} dengan dimensi 20385/32161, X_{test} dengan dimensi 8737/32161, y_{train} dengan dimensi 20385, dan y_{test} dengan dimensi 8737.

3.2.13 Data Modeling

Tahapan *modeling* akan melakukan pemodelan *dataset* dengan algoritma, *Support Vector Machine* dengan klasifikasi *dataset* menjadi 2 kategori yaitu positif dan negatif kemudian data dilatih dengan rasio data train 70% dan data test 30%. Pemodelan ini menggunakan *library* *sklearn.svm* untuk *import* *SVC* (*Support Vector Classification*) yang kemudian akan difit pada data X_{train} dan y_{train} . Berikut sintak *modeling* menggunakan algoritma *Support Vector Machine* [23].

```
from sklearn.svm import SVC
model_svm = SVC().fit(X_train, y_train)
y_predictions = model_svm.predict(X_test)
print("Hasil Dari SVM Score Data Training : ", model_svm.score(X_train, y_train))
print("Hasil Dari SVM Score Data Testing : ", model_svm.score(X_test, y_test))

Hasil Dari SVM Score Data Training : 0.998178898580968
Hasil Dari SVM Score Data Testing : 0.9581898233680082

accuracy = accuracy_score(y_test, y_predictions)
recall = recall_score(y_test, y_predictions, average='weighted')
precision = precision_score(y_test, y_predictions, average='weighted')
f1 = f1_score(y_test, y_predictions, average='weighted')

print("Accuracy: "+ "{:.2%}".format(accuracy))
print("Recall: "+ "{:.2%}".format(recall))
print("Precision: "+ "{:.2%}".format(precision))
print("F1-Score: "+ "{:.2%}".format(f1))
model_performance_log['SVM'] = [accuracy, recall, precision, f1]

Accuracy: 95.02%
Recall: 95.02%
Precision: 95.18%
F1-Score: 95.01%
```

Gambar 22. Model SVM

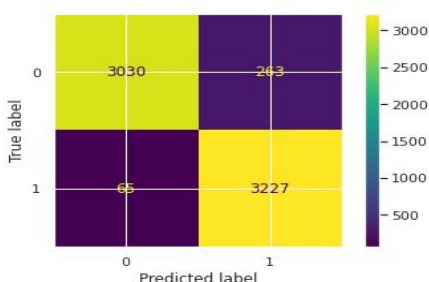
Dari pemodelan *dataset* menggunakan algoritma *Support Vector Machine* menghasilkan nilai *accuracy* 95.02%, *recall* 95.02%, *precision* 95.18%, *F1-Score* 95.01%.

3.2.14 Evaluasi Model

Setelah dilakukannya pemodelan pada *dataset* menggunakan algoritma *Support Vector Machine* tahap selanjutnya mengevaluasi performa model menggunakan *confusion matrix*, berikut hasil *confusion matrix* dari model [24]:

```
cm = confusion_matrix(y_test, y_predictions)
disp = ConfusionMatrixDisplay(confusion_matrix=cm)
disp.plot()

plt.show()
```



Gambar 23. Confusion Matrix Model SVM

Dapat dilihat pada gambar 23 hasil *confusion matrix* pada algoritma *Support Vector Machine* yang terdiri dari 2 nilai klasifikasi yaitu positif dan negatif dijelaskan sebagai berikut :

1. Sentimen positif diprediksi sebagai sentimen positif ada 3227 ternyata benar ada 3227 sentimen positif,
2. Diprediksikan ada 263 sentimen positif ternyata prediksi tersebut salah karena sentimen tersebut merupakan sentimen negatif.
3. Diprediksikan ada 65 sentimen negatif ternyata salah karena 65 sentimen tersebut adalah sentimen positif.
4. Diprediksikan ada 3030 sentimen negatif dan ternyata prediksi tersebut benar .

Setelah mengetahui hasil evaluasi model dengan *confusion matrix* selanjut mengevaluasi model menggunakan *classification report* untuk melihat *Score Precision*, *Recall*, dan *F1-score*.

	precision	recall	f1-score	support
Negatif	0.98	0.92	0.95	3293
Positif	0.92	0.98	0.95	3292
accuracy			0.95	6585
macro avg	0.95	0.95	0.95	6585
weighted avg	0.95	0.95	0.95	6585

Gambar 24. Classification Report Model SVM

Dapat dilihat pada gambar 24. algoritma *Support Vector Machine* pada sentimen negatif mendapat *score precision* 0.98, *recall* 0.92, dan *F1-score* 0.95. Pada sentimen positif mendapat *score precision* 0.92, *recall* 0.98, dan *f1-score* 0.95 dari hasil *confusion matrix*.

Tabel 1. Hasil Kesimpulan Evaluasi

No	Nama Algoritma	Accuracy	Precision		Recall		F1 Score		Jumlah Sentimen	
			Positif	Negatif	Positif	Negatif	Positif	Negatif	Positif	Negatif
1.	Support Vector Machine	0,95	0,92	0,98	0,98	0,92	0,95	0,95	23.568	5.554

3.2.15 Visualisasi Data

Tahapan terakhir dalam penelitian ini adalah menyesuaikan hasil klasifikasi analisis sentimen komentar video Youtube di *channel* youtube tvone tentang ustadz abdul somad untuk kelas positif dan negatif. Visualisasi dalam bentuk *wordcloud* menunjukkan kata-kata yang sering muncul dalam komentar video di Youtube [19].



Gambar 25. Gambar Visual Wordcloud 1



Gambar 26. Gambar Visual Wordcloud 2

Dari hasil visualisasi *wordcloud* ditemukan kata-kata yang paling banyak muncul pada data positif yaitu “singapura”, “tolak”, “negara”, dan “orang”. Sedangkan pada data negatif yaitu “orang”, “negara”, “kafir”, “tolak” dan “uas”.

4. Kesimpulan

Berdasarkan hasil yang diperoleh setelah melalui tahap-tahap evaluasi, penelitian menghasilkan *score accuracy* dan hasil prediksi *confusion matrix* dari algoritma *Support Vector Machine*. Penelitian ini telah mencapai hasil akhir dan di dapatkan kesimpulan sebagaiberikut :

1. Penerapan algoritma SVM (*Support Vector Machine*) untuk mencari penggunaan kata yang paling sering muncul pada ulasan komentar youtube pada *channel* tvone tentang Ustad Abdul Somad di deportasi di Singapura mendapati hasil yang baik didapati beberapa kata yang sering muncul pada sentimen positif maupun negatif diantaranya adalah orang, negara, singapura, uas, tolak, kafir dan indonesia.

2. Klasifikasi komentar video Youtube pada *channel* tvone tentang Ustad Abdul Somad di deportasi di Singapura dengan algoritma *Support Vector Machine* mendapati hasil *Accuracy* 95.02%, *Recall* 95.02%, *Precision* 95.18% dan *F1 - Score* 95.01%. Untuk itu, penerapan *Support Vector Machine* pada peneltian ini memiliki akurasi yang tinggi sehingga dapat digunakan untuk memberikan solusi terhadap permasalahan analisis sentimen komentar video di Youtube.

DAFTAR RUJUKAN

[1] C. A. Fuchs, N. D. Mermin, and R. Schack, “An introduction to QBism with an application to the locality of quantum mechanics,” *Am. J. Phys.*, vol. 82, no. 8, pp. 749–754, 2014, doi: 10.1119/1.4874855.

[2] F. Immermann and Y. Huang, *An introduction to cluster analysis*. 2003.

[3] A. S. H. Basari, B. Hussin, I. G. P. Ananta, and J. Zeniarja, “Opinion mining of movie review using hybrid method of support vector machine and particle swarm optimization,” *Procedia Eng.*, vol. 53, pp. 453–462, 2013, doi: 10.1016/j.proeng.2013.02.059.

[4] J. Eska, “Penerapan Data Mining Untuk Prekdiksi Penjualan Wallpaper Menggunakan Algoritma C4.5 STMIK Royal Ksiaran,” *JURTEKSI (Jurnal Teknol. dan Sist. Informasi)*, vol. 2, pp. 9–13, 2016.

[5] G. B. A. Irnawati, O., Ilistianto, “Jurnal Evolusi Volume 6 Nomor 2 - 2018 | evolusi.web.id,” *Evolusi*, vol. 6, pp. 12–18, 2018.

[6] F. Dona, I. Maulida, and B. Nugraha, “Klasifikasi Dan Analisis Dataset Komentar Video Youtube Menggunakan Supervised Learning,” *Seminastika*, vol. 3, no. 1, pp. 86–90, 2021, doi: 10.47002/seminastika.v3i1.232.

[7] M. H. Syafa’at, E. R. Setyaningsih, and Y. Kristian, “Svm Untuk Sentiment Analysis Calon Kepala Daerah Berdasar Data Komentar Video Debat Pilkada Di Youtube,” *ANTIVIRUS J. Ilm. Tek. Inform.*, vol. 15, no. 2, pp. 262–276, 2021.

[8] J. A. Septian, T. M. Fachrudin, and A. Nugroho, “Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor,” *J. Intell. Syst. Comput.*, vol. 1, no. 1, pp. 43–49, 2019, doi: 10.52985/insyst.v1i1.36.

[9] H. Setiawan, E. Utami, and S. Sudarmawan, “Analisis Sentimen Twitter Kuliah Online Pasca Covid-19 Menggunakan Algoritma Support Vector Machine dan Naive Bayes,” *J. Komtika (Komputasi dan Inform.)*, vol. 5, no. 1, pp. 43–51, 2021, doi: 10.31603/komtika.v5i1.5189.

[10] D. Mualfah, W. Fadila, and R. Firdaus, “Teknik SMOTE untuk Mengatasi Imbalance Data pada Deteksi Penyakit Stroke Menggunakan Algoritma Random Forest,” *J. CoSciTech (Computer Sci. Inf. Technol.)*, vol. 3, no. 2, pp. 107–113, 2022, doi: 10.37859/coscitech.v3i2.3912.

[11] S. Nanda, D. Mualfah, and D. A. Fitri, “Analisis Sentimen Kepuasan Pengguna Terhadap Layanan Streaming Mola Menggunakan Algoritma Random Forest,” no. x, pp. 210–219, 2022.

[12] K. Di, Y. Tentang, I. Afdhal, R. Kurniawan, I. Iskandar, and R. Salambue, “Penerapan Algoritma Random Forest Untuk Analisis Sentimen,” vol. 5, no. 1, pp. 122–130, 2022.

[13] F. Fauzi *et al.*, “ANALISIS SENTIMEN PENGGUNA YOUTUBE TERHADAP PROGRAM VAKSIN COVID-19,” pp. 141–148, 2021.

[14] H. Setiawan and E. Utami, “Analisis Sentimen Twitter Kuliah Online Pasca Covid-19 Menggunakan Algoritma Support Vector Machine dan Naive Bayes,” vol. 5, no. 1, pp. 43–51, 2021.

[15] A. Sentimen, S. Pekerja, and P. Tolak, “PADA MEDIA SOSIAL YOUTUBE MENGGUNAKAN ALGORITMA,” no. 02, pp. 99–105.

[16] M. Svm and D. A. N. Nbc, “PERBANDINGAN HASIL KLASIFIKASI CURAH HUJAN MENGGUNAKAN METODE SVM DAN NBC 1 Marthin Luter Laia, 2 Yudi Setyawan,” vol. 05, no. 2, pp. 51–61, 2020.

[17] F. D. Marleny *et al.*, “KLASIFIKASI DAN ANALISIS DATASET KOMENTAR VIDEO YOUTUBE MENGGUNAKAN SUPERVISED LEARNING CLASSIFICATION AND ANALYSIS OF YOUTUBE VIDEO COMMENT DATASET USING SUPERVISED LEARNING,” pp. 2–6, 2021, doi: 10.47002/seminastika.v3i1.232.

[18] J. I. Polinema *et al.*, “IMPLEMENTASI SENTIMEN ANALISIS KOMENTAR CHANNEL VIDEO

-
- PELAYANAN PEMERINTAH DI YOUTUBE MENGGUNAKAN,” pp. 209–213, 2012.
- [19] P. Morgan, B. Show, and N. Media, “ANALISIS SENTIMEN PENGGUNA YOUTUBE TERHADAP TAYANGAN # MATANAJWAMENANTITERAWAN DENGAN METODE NAÏVE BAYES SENTIMENT ANALYSIS OF USER COMMENTS ON YOUTUBE VIDEO # MATANAJWAMENANTITERAWAN WITH NAIVE BAYES CLASSIFIER METHOD,” vol. 5, no. 1, pp. 1–6, 2022, doi: 10.33387/jiko.
- [20] N. Herlinawati, Y. Yuliani, S. Faizah, and W. Gata, “ANALISIS SENTIMEN ZOOM CLOUD MEETINGS DI PLAY STORE MENGGUNAKAN NAÏVE BAYES DAN SUPPORT VECTOR MACHINE,” vol. 5, no. 2, pp. 293–298, 2020.
- [21] F. S. Pattiha, “Perbandingan Metode K-NN , Naïve Bayes , Decision Tree untuk Analisis Sentimen Tweet Twitter Terkait Opini Terhadap PT PAL Indonesia,” vol. 9, no. 2, pp. 506–514, 2022, doi: 10.30865/jurikom.v9i2.4016.
- [22] C. Churn and M. Kombinasi, “Penanganan Ketidakseimbangan Data pada Prediksi,” vol. 6, no. 1, pp. 31–37, 2021.
- [23] M. H. Syafa’at, E. R. Setyaningsih, and Y. Kristian, “Svm Untuk Sentiment Analysis Calon Kepala Daerah Berdasar Data Komentar Video Debat Pilkada Di Youtube,” *ANTIVIRUS J. Ilm. Tek. Inform.*, vol. 15, no. 2, pp. 262–276, 2021.
- [24] C. C. Aggarwal, *Data Classification*. 2015. Data Clustering Algorithms and Applications. International Standard Book Number-13: 978-1-4665-5821-2 (Hardback).