

## BAB 4

### HASIL PENELITIAN

#### 4.1 RINGKASAN HASIL PENELITIAN

Terdapat beberapa langkah yang dilakukan untuk melaksanakan penelitian analisis sentimen yang menggunakan data *tweet* dari pengguna media sosial X untuk mendapatkan hasil berupa sentimen positif dan negatif. Berikut adalah langkah yang harus dilalui pada penelitian ini:

##### 4.1.1 Pengambilan Data

Pengambilan data adalah langkah pertama yang dilakukan untuk mengambil data *tweet* dari X yang berkaitan dengan program makan siang gratis menggunakan *tool Google Colaboratory* yang selanjutnya disimpan dengan format *Microsoft Excel*. Total data yang didapatkan sebanyak 400 yang berisi *tweet* dengan kata kunci “Makan Siang Gratis” dari 28 November 2023 sampai 10 Februari 2024.

Pengambilan data pada penelitian ini menggunakan *library tweet-harvest* yang memudahkan untuk mengambil data tanpa API. Langkah pertama yang dilakukan yaitu install Node.js karena *tweet-harvest* dibuat menggunakan Node.js. Pada Gambar 4.1 merupakan kode yang digunakan untuk *import library* dan install Node.js.

```
# Install Node.js (because tweet-harvest built using Node.js)
!sudo apt-get update
!sudo apt-get install -y ca-certificates curl gnupg
!sudo mkdir -p /etc/apt/keyrings
!curl -fsSL https://deb.nodesource.com/gpgkey/nodesource-repo.gpg.key | sudo gpg --dearmor -o /etc/apt/keyrings/nodesource.gpg
!NODE_MAJOR=20 && echo "deb [signed-by=/etc/apt/keyrings/nodesource.gpg] https://deb.nodesource.com/node_${NODE_MAJOR}.x nodistro main" | sudo tee
!sudo apt-get update
!sudo apt-get install nodejs -y
!node -v
```

Gambar 4.1 Import Library Crawling Data

Kode ini berfungsi untuk pengambilan data jika telah mengimport library, selanjutnya kode untuk mengambil data. Diawali dengan memasukkan kode token, diikuti dengan nama file CSV, kata kunci yang diinginkan, dan lama waktu pengambilan data yang dibutuhkan. Dapat dilihat pada Gambar 4.2 yang menunjukkan kode yang digunakan untuk melakukan crawling data:

```
[ ] token = '3b503b54cc948c72576851361dca8c47cfb70a11'
```

```

filename = 'makansiag.csv'
keyword = 'makan siang gratis'
search_keyword = f'{keyword} since:2023-11-28 until:2024-02-10'
limit = 10000

!npx -y tweet-harvest@2.6.1 -o "{filename}" -s "{search_keyword}" --tab "LATEST" -l {limit} --token {token}

```

Gambar 4.2 Crawling Data

Setelah data berhasil diambil selanjutnya disimpan dengan format *Microsoft Excel* atau CSV. Tabel 4.1 menunjukkan data yang telah berhasil dikumpulkan:

Tabel 4. 1 Contoh Data Tweet dan Re-tweet

| No | Tweet                                                                                                                                                                                                                                                                                                                                               |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | @tanyakanrl Angkot gratis itu kayak BLT tp tanpa seleksi semua yg naik dapat. Klo di tingkat nasional kira2 sama dgn program 02 makan siang gratis utk anak sekolah mau kaya/miskin semua dapat.                                                                                                                                                    |
| 2  | @zarryhendrik Program makan siang gratis ya? Ngk ush dikasi makan jg rakyat kecil udh pintar cari makan sendiri yg perlu itu masadepan anak <sup>2</sup> gratis sekolah.                                                                                                                                                                            |
| 3  | @Yurissa_Samosir @gibran_tweet @AdamVelcro @AdeDhani9 @aking_adit @AlbertKusen11 @chaidar_2000 @Kang_Tonieee @NohCeukenohhh @mustafaagus007 @choymarkochoy Selain makan siang gratis saya usul program tidur sehari 8 jam mas biar pertumbuhan kita optimal                                                                                         |
| 4  | @Piyusaja2 Sok2an nolak program makan siang gratis Giiran ada gratisan diserbu juga Kasihan                                                                                                                                                                                                                                                         |
| 5  | Ini emang pro kontra banget sih masalah makan siang gratis ya. Sempet juga kemarin baca tweet tentang anggarannya bisa 1T cuma buat 1 program dan 1 hari which is for me it's a pity only for one program but you spend that much gak efektif buat 5 tahun ke depan.                                                                                |
| 6  | @ganjarpranowo Program Makan Gratis Duwit nya Dari Mana? Menurut Ganjar program makan siang gratis dari Prabowo ini tidak menyelesaikan permasalahan stunting di Indonesia. Ganjar menilai program makan gratis ini tidak tepat sasaran. Kalau ngasih makannya kepada anak-anak untuk <a href="https://t.co/63iG7gaabl">https://t.co/63iG7gaabl</a> |

|    |                                                                                                                                                                                                                                                                                                                            |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                                                                                                                                                                                                                                                                                                                            |
| 7  | @revanianara ada aja ya org kek gini sama aja kaya anak kesehatan yang setuju sama program makan siang gratis untuk atasi stunting                                                                                                                                                                                         |
| 8  | @zidanpragata @TxdariHI Mau nanya nih buat kaum all in satu putaran yang ekonominya menengah ke bawah. Program makan siang gratis kan tidak untuk semua orang dan harga bahan pokok naik terus contohnya saat ini harga beras naik. Nasib kalian nanti gimana kira2?                                                       |
| 9  | @Kupang458869 @Sgrdamai @ChrisJ_2211 Keluarga yg terbantu menjadi mandiri dan bahkan terentaskan kemiskinanya tidak perlu lagi menadahkan tangan menunggu makan siang dan susu gratis. Bahkan kalau program itu dijalankan dg tepat dan benar saya yakin untuk menuju Indonesia emas dan maju tidak perlu menunggu th 2045 |
| 10 | Saya allin in 02 saya dukung program anak sekolah makan siang gratis kalo perlu tambah jam belajar fullday 5hari efektif 2 hari libur kalo bisa hapus penyeragaman sekolah sekolah pakaian bebas sopan .                                                                                                                   |

#### 4.1.2 Preprocessing Data

Untuk memperbaiki data yang sebelumnya tidak terstruktur agar menjadi lebih terorganisir, atribut yang tidak berpengaruh pada proses analisis sentimen akan dihilangkan dari data saat ini selama tahap *preprocessing data*. Dalam *preprocessing data* membutuhkan beberapa library guna membantu proses tahapan *preprocessing*. Library yang dibutuhkan seperti *library numpy* yang membantu komputasi, *library pandas* berguna untuk memanipulasi data, *nlTK* membantu mengolah data teks menjadi bahasa yang natural, *nlTK.corpus* merupakan standarisasi data teks, serta *nlTK.tokenize* guna membantu tahapan *tokenizing* dan *stemming*. Pada Gambar 4.3 adalah library yang telah diimport sebelum melakukan proses *preprocessing data*.

```
import numpy as np
import pandas as pd
from pandas import DataFrame
import nltk, emoji, re, string
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
```

Gambar 4.3 Library Preprocessing Data

Adapun langkah-langkah yang harus dilalui dalam proses *preprocessing data* sebagai berikut:

#### 4.1.2.1 Cleaning Data

Tahapan *cleaning data* melibatkan penghapusan digit, tanda baca, dan tautan url yang tidak relevan dengan data yang diambil. Kode yang digunakan pada *cleaning data* ditunjukkan pada Gambar 4.4 seperti di bawah ini:

```
def url_remove(text):
    t = re.sub(r'https://\S+', '', text)
    return t

def punc_remove(text):
    t = re.sub(r'^\W\s+', '', text)
    return t

def number_remove(text):
    t = re.sub(r"\d+", "", text)
    return t

def regex_remove(text):
    reg = "b'"
    return re.sub(reg, " ", text)

def emoji_remove(text):
    pattern = r"(?=[:=][oO\.\.]+[:=])|(?=[-;_\*]{2,})"
    newText = re.sub(pattern, '', text) #remove emoji dg re
    return emoji.replace_emoji(newText, replace='') #remove emoji dg lib emoji

def hashtag_remove(text):
    reg = "#(\w+)"
    return re.sub(reg, " ", text)

cleaned = []
```

Gambar 4.4 Cleaning Data

Pada Gambar 4.5 menampilkan hasil dari *cleaning data* sebagai berikut:

| No |    | Text                                                          | Cleaned_Text                                      | Label |
|----|----|---------------------------------------------------------------|---------------------------------------------------|-------|
| 0  | 0  | @tanyakanri Angkot gratis itu kayak BLT tp tan...             | tanyakanri angkot gratis kayak blt tapi seleks... | 1     |
| 1  | 1  | Kampanye pake Ferrari sedang program unggulann...             | kampanye pake ferrari sedang program unggul ma... | 0     |
| 2  | 2  | @Dahnilanzar Kemiskinan ko dirawat lewat makn ...             | dahnilanzar miskin ko rawat lewat makn siang g... | 0     |
| 3  | 3  | @tempodotco Kerja aki aki ini kampanye kok nge...             | tempodotco kerja aki aki kampanye kok ngedumel... | 0     |
| 4  | 4  | Siapa yg Paling diuntungkan dlm Program makan ...             | siapa yang paling untung dlm program makan gra... | 1     |
| 5  | 5  | @Yurissa_Samosir @gibran_tweet @AdamVelcro @Ad...             | yurissa samosir gibrantweet adamvelcro adedha...  | 1     |
| 6  | 6  | Rasionalitas program <sup>2</sup> yg akan dikerjakan pa @g... | rasionalitas program yang kerja pa ganjarprano... | 0     |
| 7  | 7  | @Yurissa_Samosir @gibran_tweet @AdamVelcro @Ad...             | yurissa samosir gibrantweet adamvelcro adedha...  | 1     |
| 8  | 8  | @zarryhendrik Gini gini si caleg itu kan bilan...             | zarryhendrik gin gin si caleg kan bilang gabut... | 1     |
| 9  | 9  | @Piyusaja2 Sok2an nolak program makan siang gr...             | piyusaja sok nolak program makan siang gratis ... | 1     |
| 10 | 10 | Ini emang pro kontra banget sih masalah makan ...             | memang pro kontra banget sih masalah makan sia... | 0     |

Gambar 4.5 Hasil Cleaning Data

#### 4.1.2.2 Case Folding

Mengubah huruf kapital menjadi huruf kecil atau *lowercase* diketahui sebagai *case folding*. Pada pengambilan data awal belum akurat untuk dianalisis sehingga untuk memenuhi standard analisis maka membutuhkan proses *case folding*. Gambar 4.6 di bawah ini menampilkan kode yang digunakan untuk *case folding*.

```
def lowercase():
    lower_word = data['text'].str.lower()
    return lower_word

lower_tweet = lowercase()

print(lower_tweet)
```

Gambar 4.6 Case Folding

Hasil dari tahap *case folding* ditampilkan pada Gambar 4.7 berikut ini:

```
⇒ 4      indonesia
   5      dki jakarta
   21     indonesia
   22     indonesia
   24     indonesia
```

Gambar 4.7 Hasil Case Folding

#### 4.1.2.3 Tokenizing

*Tokenizing* adalah proses yang dilakukan untuk membagi teks menjadi beberapa bagian yang dikenal dengan *token*. Selain itu, *tokenizing* juga berfungsi untuk menyatukan sekumpulan karakter dalam satu teks menjadi satu kata. Kode yang digunakan pada *tokenizing* ditampilkan pada Gambar 4.8 di bawah:

```
def clean_text(text):
    for i in text:
        cleaned.append(emoji_remove(punc_remove(number_remove(regex_remove(hashtag_remove(url_remove(
            re.sub("[\n\r\t\x0a]", " ", i).strip()))))))
    clean_text(data["full_text"])
```

Gambar 4.8 Tokenizing

Hasil dari *tokenizing* ditunjukkan pada Gambar 4.9 sebagai berikut:

|    | full_text                                        | text        |
|----|--------------------------------------------------|-------------|
| 4  | Indonesia;0;0;0;https://x.com/ZoEI__SG/status... | Indonesia   |
| 5  | DKI Jakarta;0;0;0;https://x.com/sfarisin3/sta... | DKI Jakarta |
| 21 | Indonesia;0;0;0;https://x.com/tegarirsyad/sta... | Indonesia   |

Gambar 4.9 Hasil Tokenizing

#### 4.1.2.4 Stopword Removal

Teknik menghilangkan kata-kata umum dari teks yang tidak memiliki informasi penting apa pun dikenal sebagai *stopword removal*. Kata-kata yang mengandung informasi adalah satu-satunya yang akan dipilih dalam tahap ini, karena istilah yang tidak relevan seperti “untuk”, “namun”, “dua”, dan lain sebagainya. Kode *stopword removal* ditampilkan pada Gambar 4.10 di bawah ini:

```
from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory

factory = StopWordRemoverFactory()
more_stopword = ['rt', 'anjing', 'anjir', 'pantek']
stopword = factory.create_stop_word_remover()
stopwords = factory.get_stop_words()+more_stopword
```

Gambar 4.10 Stopword Removal

Kata-kata yang perlu dilakukan penghapusan sesuai dengan yang ada pada *library sastrawi* termasuk dalam tahap *stopword removal*. Daftar kata-kata yang terdapat di *library sastrawi* disajikan dalam Tabel 4.2 berikut:

Tabel 4.2 Daftar Kata Stopword Removal

| Daftar Kata Stopword Removal                                                              |
|-------------------------------------------------------------------------------------------|
| ['yang', 'untuk', 'pada', 'ke', 'para', 'namun', 'menurut', 'antara', 'dia', 'dua', 'ia'] |

Proses penghapusan kata-kata akan dilakukan jika daftar kata *stopword* telah muncul dan disesuaikan dengan yang ada pada daftar kata *stopword*. Pada Gambar 4.11 terdapat kode yang digunakan untuk melakukan penghapusan kata-kata yaitu:

```
def removeStopWords(text):
    clean_word_list = [word for word in text.split() if word not in stopwords]
    return clean_word_list

stopwords_tweet = lower_tweet.apply(removeStopWords)

print(stopwords_tweet)
```

Gambar 4.11 Proses Stopword Removal

Pengecekan kata yang ada pada *term* sesuai atau tidaknya merupakan fungsi dari *removeStopWords*. Jika ternyata sesuai maka penghapusan akan dilakukan dengan variabel *clean\_word\_list*. Gambar 4.12 di bawah ini menampilkan hasil *stopword removal* yang telah dilakukan.

```
0 [tanyakanr1, angkot, gratis, kayak, blt, tp, s...
1 [kampanye, pake, ferrari, sedang, program, ung...
2 [dahnilanzar, kemiskinan, ko, dirawat, lewat, ...
3 [zarryhendrik, program, makan, siang, gratis, ...
4 [program, makan, siang, gratis, kalo, bulan, p...
...
79 [dahnilanzar, gratis, pala, lu, diambil, uang,...
80 [todayis_esa, bowcuteeee, wonyoungiee, kadafi, ...
81 [zarryhendrik, walaupun, programnya, kena, si,...
82 [b_resswo, doktertifa, omong, kosong, rakyat, ...
83 [tanyarlfe, selama, gue, sinis, dgn, program,...
Name: text, Length: 84, dtype: object
```

Gambar 4.12 Hasil Stopword Removal

#### 4.1.2.5 Stemming

*Stemming* bertujuan untuk mengurangi kata-kata ke bentuk dasarnya sehingga variasi kata yang terdapat kesamaan dianggap sebagai kata yang sama

dalam analisis teks. *Stemming* menggunakan kode yang ditunjukkan pada Gambar 4.13 seperti berikut:

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

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

def stemmed_wrapper(term):
    return stemmer.stem(term)

term_dict = {}

for document in stopwords_tweet:
    for term in document:
        if term not in term_dict:
            term_dict[term] = " "

print(len(term_dict))
print("-----")

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

print(term_dict)
print("-----")

def get_stemmed_term(document):
    return [term_dict[term] for term in document]

stem_tweet = stopwords_tweet.apply(get_stemmed_term)
print(stem_tweet)
```

Gambar 4.13 Stemming

Berikutnya hasil dari *stemming* ditampilkan pada Gambar 4.14 di bawah ini

```
➡ kang_tonjee : kang tonjee
nohceukenohhh : nohceukenohhh
mustafaagus : mustafaagus
choymarkochoy : choymarkochoy
usul : usul
tidur : tidur
sehari : hari
```

Gambar 4.14 Hasil Stemming

#### 4.1.2.6 Normalization

*Normalization* bertujuan menyeragamkan *term* yang terdapat kesalahan pada penulisan untuk selanjutnya diubah menjadi bentuk standar dan konsisten sehingga dapat dilakukan analisis dengan mudah. Gambar 4.15 mengandung kode yang digunakan pada *normalization* seperti berikut ini:

```
normalizad_word = pd.read_excel("/content/normalization.xlsx")
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 normalized_term(document):
    return [normalizad_word_dict[term] if term in normalizad_word_dict else term for term in document]

normal_tweet = stem_tweet.apply(normalized_term).str.join(" ")

print(normal_tweet)
```

Gambar 4.15 Normalization

Contoh data yang ada pada *normalization.xlsx* dengan tujuan memperbaiki kata yang salah ditampilkan di Tabel 4.3 berikut ini:

Tabel 4.3 Dataset Normalization

| Sebelum | Sesudah   |
|---------|-----------|
| tar     | nanti     |
| kmren   | kemarin   |
| moga    | semoga    |
| sdh     | sudah     |
| bpk     | bapak     |
| bs      | bisa      |
| gmn     | bagaimana |
| mo      | mau       |
| hape    | handphone |
| ttg     | tentang   |

Karena komponen *tweet* lebih tersusun selama tahap *preprocessing data* maka dapat digunakan untuk perhitungan selanjutnya. Setelah melalui tahap *preprocessing data*, data *tweet* yang ada akan ditampilkan pada Tabel 4.4 berikut ini:

Tabel 4.2 Data Tweet Hasil Preprocess

| No | Data Tweet Hasil Preprocess                                                                                                                                                                                                                                             |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Ini emang pro kontra banget sih masalah makan siang gratis ya. Sempet juga kemarin baca tweet tentang anggarannya bisa 1T cuma buat 1 program dan 1 hari which is for me it's a pity only for one program but you spend that much gak efektif buat 5 tahun ke depan.    |
| 2  | sebuah gambaran eksekusi program makan siang gratis                                                                                                                                                                                                                     |
| 3  | prof din tanya sumber dana program makan siang gratis usung prabowo                                                                                                                                                                                                     |
| 4  | calon presiden nomor urut prabowo subianto singgung soal orang aku nelayan sebut tak perlu program makan siang gratis singgung prabowo hadir kampanye buka stadion gelora bandung laut api kamis prabowo                                                                |
| 5  | Kalau ditanya solutif apa enggak ya bisa dibilang solutif tapi bukan solusi terbaik. I think there are so many solution instead of doing makan siang gratis. Moreover we know budget of this program isn't only cost 100M but 1 trillion of rupiah which is cost a lot. |
| 6  | dody gambar depan bagaimana eksekusi program makan siang susu gratis                                                                                                                                                                                                    |
| 7  | arifmiqbal goal id tuju kalau saja diknas program push up sit up tiap pagi set terus rep nya tidak usah nyakbanyak yang penting last set sampe fail rep kaya dah cakep banget kalau tambah makan siang gratis high protein kaya dada ayam tiap hari jadi dah            |
| 8  | bospurwa efek program makan siang gratis                                                                                                                                                                                                                                |
| 9  | program makan siang minum susu gratis sekolah program school meals wfp sangat manfaat cegah jadi stunting gizi buruk meski wilayah mu mungkin dapat derita stunting gizi buruk                                                                                          |
| 10 | benar banyak anak yang tidak butuh program makan siang gratis butuh program keluarga sejahtera yang makan seluruh keluarga bukan cuman anak doang                                                                                                                       |

#### 4.1.3 Pelabelan Manual

Pelabelan manual merupakan proses yang dilakukan dengan cara menandai atau memberikan label secara manual terhadap kata yang ada pada suatu dokumen untuk kemudian melakukan analisis terkait sentimen positif atau negatif. Hasil pelabelan manual yang sudah dilakukan dapat dilihat pada Gambar 4.16 di bawah ini:

| No |    | Text                                              | Cleaned_Text                                      | Label |
|----|----|---------------------------------------------------|---------------------------------------------------|-------|
| 0  | 0  | @tanyakanrl Angkot gratis itu kayak BLT tp tan... | tanyakanrl angkot gratis kayak blt tapi seleks... | 1     |
| 1  | 1  | Kampanye pake Ferrari sedang program unggulann... | kampanye pake ferrari sedang program unggul ma... | 0     |
| 2  | 2  | @Dahnilarz Kemiskinan ko dirawat lewat makn ...   | dahnilarz miskin ko rawat lewat makn siang g...   | 0     |
| 3  | 3  | @tempodotco Kerja aki aki ini kampanye kok nge... | tempodotco kerja aki aki kampanye kok ngedumel... | 0     |
| 4  | 4  | Siapa yg Paling diuntungkan dlm Program makan ... | siapa yang paling untung dlm program makan gra... | 1     |
| 5  | 5  | @Yurissa_Samosir @gibran_tweet @AdamVelcro @Ad... | yurissa samosir gibran tweet adamvelcro adedha... | 1     |
| 6  | 6  | Rasionalitas program* yg akan dikerjakan pa @g... | rasionalitas program yang kerja pa ganjarprano... | 0     |
| 7  | 7  | @Yurissa_Samosir @gibran_tweet @AdamVelcro @Ad... | yurissa samosir gibran tweet adamvelcro adedha... | 1     |
| 8  | 8  | @zarryhendrik Gini gini si caleg itu kan bilan... | zarryhendrik gin gin si caleg kan bilang gabut... | 1     |
| 9  | 9  | @Piyusaja2 Sok2an nolak program makan siang gr... | piyusaja sok nolak program makan siang gratis ... | 1     |
| 10 | 10 | Ini emang pro kontra banget sih masalah makan ... | memang pro kontra banget sih masalah makan sia... | 0     |

Gambar 4.16 Data Pelabelan Manual

Berikut ini adalah data yang telah melalui proses pelabelan manual yang dapat ditampilkan pada Tabel 4.5 di bawah ini:

Tabel 4.5 Pelabelan Manual

| No | Kelas   | Label | Cleaned_Text                                                                                                                                                                                                                                                 |
|----|---------|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Positif | 1     | tanyakanrl angkot gratis kayak blt tapi seleksi semua yang naik kalau tingkat nasional kira sama dgn program makan siang gratis untuk anak sekolah mau kayamiskin semua                                                                                      |
| 2  | Positif | 1     | siapa yang paling untung dlm program makan gratis sekali semua bisnis istilah makan siang gratis bisnis                                                                                                                                                      |
| 3  | Positif | 1     | yurissa samosir gibran tweet adamvelcro adedhani aking adit albertkusen chaidar kang toniee nohceukenohhh mustafaagus choymarkochoy makan siang gratis usul program tidur hari jam mas biar tumbuh optimal                                                   |
| 4  | positif | 1     | piyusaja sok nolak program makan siang gratis giiran gratis serbu kasihan                                                                                                                                                                                    |
| 5  | positif | 1     | arifmiqbal goal id tuju kalau saja diknas program push up sit up tiap pagi set terus rep nya tidak usah nyakbanyak yang penting last set sampe fail rep kaya dah cakep banget kalau tambah makan siang gratis high protein kaya dada ayam tiap hari jadi dah |
| 6  | Negatif | 0     | kampanye pake ferrari sedang program unggul makan siang gratis mending ikut junat berkah saja wibawa dikit kali malah pamer warga susah                                                                                                                      |

|    |         |   |                                                                                                                                                                                                                                      |
|----|---------|---|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7  | Negatif | 0 | dahnilanzar miskin ko rawat lewat makan siang gratis sama bansos cari solusituh yang buat orang tua kasih makan sendiri anak sesuai butuh gizi buat program tuh yang kembang sendiri kalau bodoh miskin rawat buat langsung oligarki |
| 8  | Negatif | 0 | tempodotco kerja aki aki kampanye kok ngedumelllll saja tanya program loe apa aki jawab cuma makan siang gratis dobolllll                                                                                                            |
| 9  | Negatif | 0 | rasionalitas program yang kerja pa ganjarpranowo amp mohmahfudmd pilih jgn hambur uang negara jadi tai dgn program makan siang gratis yang manja mafia impor ujung sengsara tani lokal makin                                         |
| 10 | Negatif | 0 | ilhamkusno nakula dedynurpalakka kurang gizi lu makanya junjung kamu program nya makan siang gratis otak kamu kaya tai isi                                                                                                           |

Pada Tabel 4.5 menunjukkan bahwa kelas positif diberi label 1 sedangkan kelas negatif diberi label 0. Melakukan pelabelan manual bertujuan untuk menghitung akurasi yang telah diberi nilai positif dan negatif.

#### 4.1.4 Training Data

Klasifikasi studi ini menggunakan fitur ekstraksi menghasilkan bobot kata yang secara otomatis dibuat berdasarkan istilah atau dokumen yang ditemukan dalam *training data*. Library *sklearn.feature\_extraction.text* serta *TfidfVectorizer* digunakan dalam perhitungan TF-IDF untuk melakukan perhitungan otomatis. Metode perhitungan TF-IDF, yang mencoba mengklasifikasikan teks dalam *training data*, dibantu oleh Library *Multinomial Naïve Bayes*. Hasil perhitungan otomatis TF-IDF- ditampilkan dalam Gambar 4.17 dan 4.18 sebagai berikut:

|          |                     |
|----------|---------------------|
| (0, 158) | 0.2388438942742163  |
| (0, 203) | 0.1953155521425483  |
| (0, 292) | 0.1953155521425483  |
| (0, 12)  | 0.12129823031702579 |
| (0, 362) | 0.2388438942742163  |
| (0, 308) | 0.06279812333147552 |
| (0, 191) | 0.06279812333147552 |
| (0, 268) | 0.06279812333147552 |
| (0, 79)  | 0.18130255588649513 |
| (0, 279) | 0.1443906564113752  |
| (0, 168) | 0.2388438942742163  |
| (0, 224) | 0.2388438942742163  |
| (0, 345) | 0.2388438942742163  |
| (0, 145) | 0.1443906564113752  |
| (0, 220) | 0.2388438942742163  |
| (0, 377) | 0.13178891332873008 |
| (0, 298) | 0.33069797642966464 |
| (0, 294) | 0.2388438942742163  |
| (0, 333) | 0.2388438942742163  |
| (0, 45)  | 0.2388438942742163  |
| (0, 157) | 0.2388438942742163  |
| (0, 108) | 0.1063264654631435  |
| (0, 14)  | 0.2388438942742163  |
| (0, 332) | 0.2388438942742163  |
| (1, 321) | 0.2244350010326135  |

Gambar 4.17 Hasil Perhitungan TF-IDF Awal

|           |                     |  |
|-----------|---------------------|--|
| :         | :                   |  |
| (31, 26)  | 0.20031209209365625 |  |
| (31, 25)  | 0.20031209209365625 |  |
| (31, 234) | 0.20031209209365625 |  |
| (31, 219) | 0.20031209209365625 |  |
| (31, 50)  | 0.20031209209365625 |  |
| (31, 237) | 0.20031209209365625 |  |
| (31, 118) | 0.16380601642335968 |  |
| (31, 208) | 0.1789574067780438  |  |
| (31, 324) | 0.16380601642335968 |  |
| (31, 19)  | 0.1789574067780438  |  |
| (31, 320) | 0.24119106963209347 |  |
| (31, 206) | 0.1789574067780438  |  |
| (31, 49)  | 0.15205368503100722 |  |
| (31, 150) | 0.16380601642335968 |  |
| (31, 352) | 0.1789574067780438  |  |
| (31, 53)  | 0.14245133110774724 |  |
| (31, 48)  | 0.1343326565459384  |  |
| (31, 196) | 0.16380601642335968 |  |
| (31, 308) | 0.11052789531776312 |  |
| (31, 191) | 0.11052789531776312 |  |
| (31, 268) | 0.0526671343318541  |  |
| (31, 279) | 0.2050344443982194  |  |
| (31, 145) | 0.12109664579213479 |  |
| (31, 298) | 0.27734769486596755 |  |
| (31, 108) | 0.08917321000215067 |  |

Gambar 4.18 Hasil Perhitungan TF-IDF Akhir

Gambar 4.19 di bawah ini menunjukkan hasil metode pengklasifikasi *Naïve Bayes*, yang digunakan untuk menyediakan *training data* untuk hasil TF-IDF pada Gambar 4.17 dan 4.18 sebelumnya.

```
auc (train data): 1.0
top 10 scores: [0.727688  0.28781012 0.32004429 0.30402163 0.71529318 0.83975498
0.30470871 0.83975498 0.77141026 0.71151569]
None
```

Gambar 4.19 Hasil Perhitungan Training Data

Langkah selanjutnya adalah membangun model klasifikasi dengan menggunakan variabel *x* dan *y* dalam *training data* yang tersedia setelah menghitung TF-IDF. Fungsi dari pemodelan yang dibuat agar memudahkan untuk melakukan pemanggilan serta eksekusi dalam tahapan berikutnya. Pembuatan model klasifikasi menggunakan kode yang ditunjukkan pada Gambar 4.20 sebagai berikut:

```
X_train,X_test,y_train,y_test = train_test_split(data['Cleaned_Text'],data['polarity'],test_size=0.1,random_state=100)

pipeline = Pipeline([
    ('bow',CountVectorizer()),
    ('tfidf',TfidfTransformer()),
    ('classifier',MultinomialNB())
])

X_train = np.asarray(X)
pipeline = pipeline.fit(X_train, np.asarray(y))
```

Gambar 4.20 Pembuatan Model Klasifikasi

Untuk memudahkan membuka dan menggunakan ulang, model dengan nama pipeline selanjutnya disimpan dalam format *.pickle*. Kode yang digunakan untuk menyimpan dan membuka file *pickle* dijelaskan pada Gambar 4.21 dan Gambar 4.22.

```
file_data = 'latihan.pickle'
f = open(file_data, 'wb')
pickle.dump(pipeline, f)
f.close()
```

Gambar 4.21 Menyimpan File Pickle

```

file_dataset = 'latihan.pickle'
f = open(file_dataset, 'rb')
training = pickle.load(f)
f.close()
print(training)

Pipeline(steps=[('bow', CountVectorizer()), ('tfidf', TfIdfTransformer()),
                 ('classifier', MultinomialNB())])

```

Gambar 4.22 Membuka File Pickle

File model klasifikasi yang dibuat sebelumnya digunakan untuk data training sejumlah 31 data yang belum diberi label dengan 369 data. Sebanyak 400 data akan dibandingkan dengan pengklasifikasi yang ada untuk memungkinkan prediksi sifat positif dan negatif. Hasil prediksi data *tweet* dapat dilihat pada Gambar 4.23, yaitu:

|     | Cleaned_Text                                       | Label | Kelas   |
|-----|----------------------------------------------------|-------|---------|
| 0   | tanyakanrl angkot gratis kayak blt tapi seleks...  | 1     | Positif |
| 1   | kampanye pake ferrari sedang program unggul ma...  | 0     | Negatif |
| 2   | dahnilanzar miskin ko rawat lewat makan siang g... | 0     | Negatif |
| 3   | zarryhendrik program makan siang gratis ngk us...  | 1     | Positif |
| 4   | program makan siang gratis kalau bulan puasa g...  | 0     | Negatif |
| ... | ...                                                | ...   | ...     |
| 79  | dahnilanzar gratis pala lu ambil uang pajak ko...  | 0     | Negatif |
| 80  | todayis esa bowcuteeee wonyoungiee kadafi choco... | 0     | Negatif |
| 81  | zarryhendrik walaupun program kena si kaya pal...  | 0     | Negatif |
| 82  | b resswa doktertifa omong kosong rakyat tahu s...  | 0     | Negatif |
| 83  | tanyarlifes lama saya sinis dgn program prabowo... | 1     | Positif |

84 rows x 3 columns

Gambar 4.23 Hasil Data Prediksi

Pada Gambar 4.24 menampilkan distribusi data positif dan negatif sebagai berikut:

|         | Cleaned_Text | Label |
|---------|--------------|-------|
| Kelas   |              |       |
| Negatif | 15           | 15    |
| Positif | 17           | 17    |

Gambar 4.24 Distribusi Data Positif dan Negatif

#### 4.1.5 Testing

*Testing* adalah proses memvalidasi keakuratan model sebelum menggunakannya untuk mengklasifikasikan label atau data uji yang telah tersedia. Adapun dua tahapan yang terdapat pada *testing* yaitu:

##### 4.1.5.1 Cross Validation

Proses penghitungan hasil akurasi beberapa kali (*k-fold*) menggunakan pengaturan yang sama disebut *cross validation*. Cara ini untuk mengetahui seberapa akurat tingkat model yang digunakan. Proses ini menggunakan kode yang ditunjukkan pada Gambar 4.25 serta menampilkan hasil dari *cross validation* seperti berikut ini:

```
[ ] from sklearn.model_selection import cross_val_score
    accuracies = cross_val_score(estimator = nbmodel, X = X_train_dtm, y = Y_train, cv = 10)
    print('Rata-rata akurasi yaitu: ', accuracies.mean())
```

Rata-rata akurasi yaitu: 0.65

Gambar 4.25 Perhitungan Cross Validation

Akurasi rata-rata ditentukan dengan memanfaatkan kode *library sklearn* yang dapat mengimport *cross\_val\_score* untuk melakukan tahap *cross validation* sepuluh kali (*cv=10*).

#### 4.1.5.2 Hasil Klasifikasi

Selama tahap *training* dan juga *testing* pengujian dilakukan pada data *tweet* yang telah menjalani pelabelan manual diuji untuk memperoleh label positif dan negatif pada data yang tersedia. Setelah menjalani proses prediksi, data *tweet* yang diuji, menghasilkan seperti Gambar 4.26 di bawah ini:

```
nbmodel = MultinomialNB(alpha=0.1)
nbmodel = nbmodel.fit(X_train_dtm,Y_train)
Y_pred = nbmodel.predict(X_test_dtm)
print(accuracy_score(Y_test,Y_pred))

0.5714285714285714
```

Gambar 4.26 Hasil Perhitungan Akurasi Data Training dan Testing

Dengan menggunakan data yang sama maka hasil yang didapat pada perhitungan akurasi data *training* dan data *testing* juga memiliki hasil yang sama.

Model yang telah dihitung menggunakan metode *confusion matrix* harus dilakukan pengujian guna mengetahui persentasenya. Kode yang digunakan untuk memunculkan *confusion matrix* terdapat pada Gambar 4.27 seperti yang tertera di bawah ini:

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(Y_test, Y_pred)
print(cm)
```

Gambar 4.27 Perhitungan Confusion Matrix

*Confusion matrix* dapat diimpor menggunakan library *sklearn* untuk memfasilitasi *testing* dan *prediction*.

Metode yang digunakan diantaranya:

1. *Accuracy* berupaya menentukan fraksi klasifikasi berdasarkan jumlah total sampel yang diperiksa
2. *Precision* berupaya menentukan persentase semua sampel klasifikasi kategori positif yang tepat
3. *Recall* menghitung persentase sampel dalam kategori positif dibandingkan dengan semua sampel dalam *testing* kategori positif
4. *F1-score* yang menentukan rata-rata *recall* dan *precision*

Untuk menghitung *accuracy*, *precision*, *recall* dan *f1-score* gunakan kode yang ditampilkan pada Gambar 4.28 di bawah ini:

```
from sklearn.metrics import classification_report
report = classification_report(Y_test, Y_pred)
print(report)
```

Gambar 4.28 Perhitungan Classification Report

#### 4.2 HASIL EVALUASI MODEL KLASIFIKASI

Sebanyak 31 data *tweet* 17 label data positif dan 15 label negatif. Pengujian pada *confusion matrix* untuk memastikan nilai data yang sebenarnya dan teraktual untuk menunjukkan keakuratan data yang ada. Hasil perhitungan *confusion matrix* pada data *training* dan *testing* ditampilkan pada Gambar 4.6 di bawah ini:

Tabel 4.6 Hasil Confusion Matrix Data Training

| Kelas Aktual | Kelas Prediksi |         |
|--------------|----------------|---------|
|              | Positif        | Negatif |
| Positif      | 1              | 1       |
| Negatif      | 2              | 3       |

*Confusion matrix* dalam Tabel 4.1 mendapatkan hasil seperti berikut: TP = 1, TN = 1, FP = 2 dan FN = 3. Jika sudah memperoleh nilai dari *confusion matrix* dilanjut menghitung akurasi model menggunakan *cross validation*. Tujuan pengujian dengan *k-fold cross validation* bertujuan agar memperoleh hasil yang sesuai dengan sepuluh kali pengulangan. Setiap perhitungan *fold* pada *k-fold cross validation* mempunyai nilai akurasi yang unik untuk mendapatkan hasil yang berbeda dari yang lain. Hasil perhitungan dari 10 *fold* pada *cross validation* di bawah ini terdapat pada Tabel 4.7:

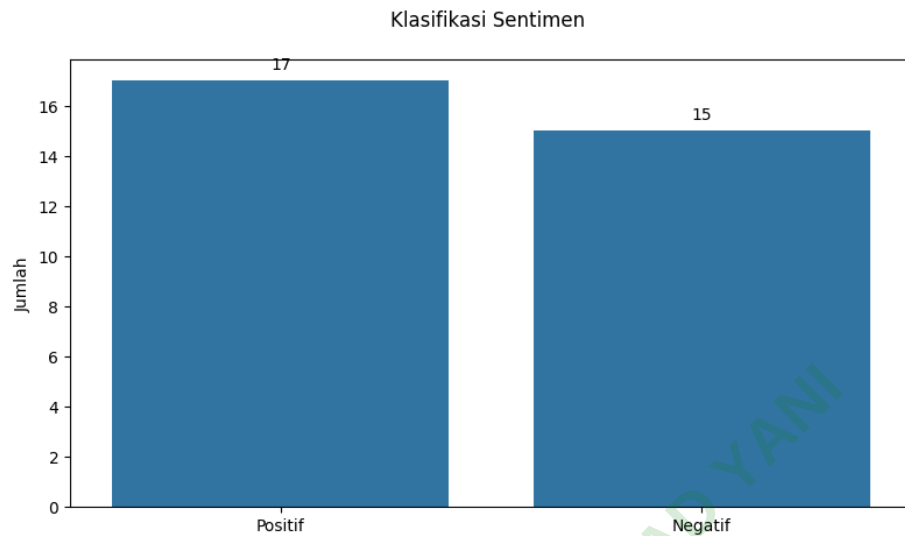
Tabel 4.7 Hasil Cross Validation Data Training dan Testing

| <b>Fold</b> | <b>Accuracy</b> |
|-------------|-----------------|
| Fold 1      | 66,7%           |
| Fold 2      | 66,7%           |
| Fold 3      | 100,0%          |
| Fold 4      | 66,7%           |
| Fold 5      | 100,0%          |
| Fold 6      | 50,0%           |
| Fold 7      | 50,0%           |
| Fold 8      | 0,0%            |
| Fold 9      | 100,0%          |
| Fold 10     | 50,0%           |

Jika *cross validation* telah dilakukan sebanyak 10 kali maka akan menghasilkan rata-rata akurasi sebesar 0,57.

#### 4.3 HASIL ANALISIS

Sebanyak 400 data dikumpulkan, dimana 369 di antaranya diberi label menggunakan prediksi dan 31 diberi label secara manual. Setelah memeriksa data *tweet* yang diprediksi menggunakan 400 data, akurasi rata-rata yang didapatkan sebesar 0,65 yang menunjukkan bahwa hasil yang didapat berhasil memberikan sentimen data *tweet* yang ada. Seperti yang ditunjukkan pada Gambar 4.29, hasil prediksi yang tidak berlabel diperoleh dari data *training* yang dilabeli menghasilkan label positif dan negatif.



Gambar 4.29 Histogram Data Positif dan Negatif

Pada klasifikasi yang ada didapatkan 17 data berlabel positif dan 15 data berlabel negatif. Sentimen positif pada data *tweet* membahas tentang ada beberapa keluarga yang tidak berkecukupan akan terbantu dengan adanya program makan siang gratis ini, program makan siang gratis pada sekolah sangat bermanfaat untuk cegah jadi stunting gizi buruk. Sedangkan sentimen negatif pada data *tweet* membahas tentang bagaimana anggaran yang digunakan, dikhawatirkan makan siang gratis ini tidak tepat sasaran dan hanya membuang uang negara. Berdasarkan hasil klasifikasi sentimen menghasilkan banyak yang bersifat positif dan mendukung adanya program makan siang gratis. Sentimen positif memiliki nilai yang baik karena dengan hasil yang ada dapat mengetahui komentar dari pengguna media sosial X tentang program makan siang gratis.