

LAMPIRAN

Lampiran 1 Import *Library*

```
Import Library

[1] pip install Sastrawi
Python
... Requirement already satisfied: Sastrawi in c:\users\vivobook\AppData\Local\Programs\Python\Python312\Lib\site-packages (1.0.1)
Note: you may need to restart the kernel to use updated packages.

[2] !pip install emoji --upgrade
Python

[26] !pip install nltk
!pip install pandas
!pip install matplotlib
!pip install openpyxl
!pip install lxml
!pip install emoji
import pandas as pd
import re
from bs4 import BeautifulSoup
import emoji
import unicodedata
import re
import string
import unicodedata
import nltk
from bs4 import BeautifulSoup
import pandas as pd
import numpy as np
import emoji
import matplotlib.pyplot as plt
from nltk.tokenize import sent_tokenize
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from nltk.tokenize import word_tokenize
Python
```

Lampiran 2 Import *Dataset*

```
Import Dataset

import pandas as pd

# Membaca data xlsx
review_data = pd.read_excel(r'C:\Users\VivoBook\Desktop\skripsi\data\Data_Segari_gabungan.xlsx')
comments = review_data[['userName', 'text']]
comments.head(5)

Python

...


|   | userName        | text                                               |
|---|-----------------|----------------------------------------------------|
| 0 | Ade Pipin       | sayuran nya fresh banget mantap deh pokoknya       |
| 1 | Kedai Rumpun    | Gambar nya ilang semua                             |
| 2 | Salsabila       | baru pertama kali order dan hasil nya memuaskan... |
| 3 | Dewi kumalasari | amanah                                             |
| 4 | Elis Lisnawati  | trmakasih segari barang fresh kurir ramah          |



len(comments)

Python
```

Dataset yang dipakai dapat diakses di [link](https://docs.google.com/spreadsheets/d/1NH3p4jTiUx6nqSSr7sQYXSygGB17ga) berikut.

<https://docs.google.com/spreadsheets/d/1NH3p4jTiUx6nqSSr7sQYXSygGB17ga/zJ/edit?usp=sharing&ouid=115406227280236311450&rtpof=true&sd=true>

Lampiran 3 Case folding

No	Case folding
1	sayuran nya fresh banget mantap deh pokoknya
2	gambar nya ilang semua
3	baru pertama kali order dan hasil nya memuaskan .smua nya masih fresh..bakal jadi langganan di sini
4	amanah
5	trmakasih segari barang fresh kurir ramah
6	menurut qu bagus, puas belanja di sehari, semoga selalu amanah, dan makin banyak lg promo2 nya
7	fresh produk, harga ok dan pengiriman cepat..mantaf
8	admin nya ramah dan responsif banget. produk nya lumayan lengkap dan fresh bersih. semoga tetap kayak gini
9	selalu segar yg dikirim
10	baru pertama mau coba, pas masukin alamat ternyata belum tersedia
11	seneng banget ada aplikasi ini, sangat memudahkan dalam berbelanja karena lengkap segala macam ada, thank you yaa founder and all team!
12	respon cepat, packing ok. barang juga ok. semoga dipertahankan terus ya. dan makin meningkat. goodluck
13	sayang sekali tidak tersedia di seluruh kota aku di kalimantan barat. kalo udah tersedia di semua kota aku bintang 5 deh
14	harga nya sungguh terjangkau dibanding harga pasar
15	cxxul pp mj uu nny h di loo
16	alhamdulillah sangat membantu. sejauh ini barang yg diberikan kualitasnya
17	pengiriman secepat kilat
18	segari sangat membantu sy, sahyrnya segar, kualitasnya bgs, semoga lbh bgs lg ya segari
19	pelayanan buruk sering di cancle sudah mendekati jam pengiriman..kalau alasannya krna gak ada driver yan pick up kenapa aplikasi pengantar sayuran lain bisa pagi bahkan jam 3 or 4..ko ini gak bisa alasannya gak ada yg pick up..kalau gitu jangan ada pilihan pengantaran pagi dengan jam yg customer bisa pilih mah jam berapa! kalau ujungnya2 gak sanggup dan di cancle sangat kecewa!
20	oke banget lah pokoknya
21	after sales nya bagus.. respon cs nya gercep kalo ada barang rusak atau telur pecah di ganti uang atau koin belanja, bisa juga di kirim ulang barang yang rusak.. bener2 memuaskan belanja di segari
22	baru pertama kali sih belanja di segari. overall pengirimannya cepat, bahan2 belum dibuka karna baru besok mau diolah, tapi sekarang kasih bintang 5 dulu karna walo baru pertama kali, tapi gak ada kendala apa2 .. dan beli di online supermarket juga karna nyari bahan yg dimau di supermarket dekat rumah ternyata gak ada. oke sekian. semoga

	makin makin baik kedepannya. user interface nya juga nyaman diliat dan nyaman buat telusur . thanks!
23	sangat ngebantu
24	semoga di bandung cepat ada layanannya.
25	pengiriman cepat dan barang terjamin bagus
26	puas belanja disini
27	pokoknya mantuull
28	oke
29	adain di bali dong
30	sangat bagus dan membantu dalam belanja berbagai keperluan
31	membantu sekali untuk belanja kebutuhan masak. cuman kurang lengkap di makeup ya.
32	aplikasi sangat membantu
33	aku ksh bintang 4 dulu yaa adain di kota sukabumi dong, kadang suka mager ke pasar. kl ada segari kayanya enak nih buat emak ² yg lg mager
34	sayurannya segar ²
35	sangat mempermudah saya belanja harian.. terimakasih banyak
36	senang sekali belanja di segari . harga nya bersahabat packingnya baik dan pengirimannya cepat .
37	sangat kecewa pakai aplikasi ini. pelayan buruk. aplikasi lemot. jam pengiriman berubah sendiri. setiap mau checkout stock kosong. harus ubah lagi dan lagi sampai akhirnya jam pengiriman berubah sendiri tanpa kita sadari. komplain ke cs pelayanan buruk. jangan pesan di sini deh daripada hati dongkol. pesan dari pagi ngga sampe sampe.
38	ok.. ok
39	semoga waktunya tepat max 2 jam
40	ok...sesuai pesanan
41	alhamdulillah maa syaa allah the best pokoknya
42	pengiriman lamaaaa jauh dari estimasi awal
43	lama belanja disegari tapi kali ini harus extra hati-hati karena seperti main judi, syukur ² buah sayur yang mereka dikirim kualitas baik, tapi jika yang dikirim busuk atau muda tak layak sekarang mereka tidak akan tanggung jawab penuh, dan yang dirugikan konsumen. lebih baik kurangi belanja disini dan harus hati-hati.
44	semoga jd langganan
45	pelayanan bagus, pengiriman cepat, kualitas terjaga. terimakasih.
46	kalau emang tidak beroperasi pada saat malam takbiran,minimal sediakan tombol cancel!!!! saya tidak jadi masak untuk lebaran karna kalian tidak ada kejelasan!!!!
47	pengiriman daging tidak seperti biasa nya tidak menggunakan bubble foil dan therma freeze jdi kualitas daging tidak bagus dan menjadi lembek
48	sehari lagi kenapa sih? ko gabisa buka aplikasinya?
49	mudah digunakan... prosesnya cepat
50	mantap

Untuk melihat data hasil proses *case folding* dapat diakses di *link* berikut.
<https://docs.google.com/spreadsheets/d/1kYYAcjZ4mzetTM0NfJ-BRBTZJcKfjxE8/edit?usp=sharing&ouid=115406227280236311450&rtpof=true&sd=true>

```

Pre-Processing Data

Case Folding

comment = comments

def casefoldingText(text):
    return text.lower()

# Pastikan kolom 'text' berisi string
comment['text'] = comment['text'].astype(str)

# Terapkan fungsi casefoldingText pada kolom 'text'
comment['clean_text'] = comment['text'].apply(casefoldingText)

comment.head(5)

comment.to_csv(r'C:\Users\VivoBook\Desktop\skripsi\preprocess\case_folding_segari.csv', index=False, header=True)

```

Lampiran 4 *Cleaning Data*

No	clean_text
1	sayuran nya fresh banget mantap deh pokoknya
2	gambarnya ilang semua
3	baru pertama kali order dan hasil nya memuaskan semua nya masih fresh bakal jadi langganan di sini
4	amanah
5	trmakasih segari barang fresh kurir ramah
6	menurut qu bagus puas belanja di sehari semoga selalu amanah dan makin banyak lg promo nya
7	fresh produk harga ok dan pengiriman cepat mantaf
8	adminnya ramah dan responsif banget produknya lumayan lengkap dan fresh bersih semoga tetap kayak gini
9	selalu segar yg dikirim
10	baru pertama mau coba pas masukin alamat ternyata belum tersedia

11	seneng banget ada aplikasi ini sangat memudahkan dalam berbelanja karena lengkap segala macam ada thank you yaa founder and all team
12	respon cepat packing ok barang juga ok semoga dipertahankan terus ya dan makin meningkat goodluck
13	sayang sekali tidak tersedia di seluruh kota ðŸ˜¸ aku di kalimantan barat kalo udah tersedia di semua kota aku bintang deh
14	harga nya sungguh terjangkau dibanding harga pasar
15	cxxul pp mj uu nny h di loo
16	alhamdulillah sangat membantu sejauh ini barang yg diberikan kualitasnya ðŸ•• ðŸ•• semua
17	pengiriman secepat kilat
18	segari sangat membantu sy sahyrnya segar kualitasnya bgs semoga lbh bgs lg ya segari
19	pelayanan buruk sering di cancle sudah mendekati jam pengiriman kalau alasannya krna gak ada driver yanh pick up kenapa aplikasi pengantar sayuran lain bisa pagi bahkan jam or ko ini gak bisa alasannya gak ada yg pick up kalau gitu jangan ada pilihan pengantaran pagi dengan jam yg customer bisa pilih mah jam berapa kalau ujungnya gak sanggup dan di cancle sangat kecewa
20	oke banget lah pokoknya
21	after sales nya bagus respon cs nya gercep kalo ada barang rusak atau telur pecah di ganti uang atau koin belanja bisa juga di kirim ulang barang yang rusak bener memuaskan belanja di segari
22	baru pertama kali sih belanja di segari overall pengirimannya cepat bahan belum dibuka karna baru besok mau diolah tapi sekarang kasih bintang dulu karna walo baru pertama kali tapi gak ada kendala apa dan beli di online supermarket juga karna nyari bahan yg dimau di supermarket deket rumah ternyata gak ada oke sekian semoga makin makin baik kedepannya user interface nya juga nyaman diliat dan nyaman buat telusur thanks
23	sangat ngebantu
24	semoga di bandung cepat ada layanannya
25	pengiriman cepat dan barang terjamin bagus
26	puas belanja disini
27	pokoknya mantuull
28	oke
29	adain di bali dong
30	sangat bagus dan membantu dalam belanja berbagai keperluan
31	membantu sekali untuk belanja kebutuhan masak cuman kurang lengkap di makeup ya
32	aplikasi sangat membantu
33	aku ksh bintang dulu yaa adain di kota sukabumi dong kadang suka mager ke pasar kl ada segari kayanya enak nih buat emakâ² yg lg mager
34	sayurannya segar

35	sangat mempermudah saya belanja harian terimakasih banyak
36	senang sekali belanja di segari harga nya bersahabat packingnya baik dan pengirimannya cepat
37	sangat kecewa pakai aplikasi ini pelayan buruk aplikasi lemot jam pengiriman berubah sendiri setiap mau checkout stock kosong harus ubah lagi dan lagi sampai akhirnya jam pengiriman berubah sendiri tanpa kita sadari komplain ke cs pelayanan buruk jangan pesan di sini deh daripada hati dongkol pesan dari pagi ngga sampe sampe
38	ok ok
39	semoga waktunya tepat max jam
40	ok sesuai pesanan
41	alhamdulillah maa syaa allah the best pokoknya
42	pengiriman lamaaaa jauh dari estimasi awal
43	lama belanja disegari tapi kali ini harus extra hati hati karena seperti main judi syukur buah sayur yang mereka dikirim kualitas baik tapi jika yang dikirim busuk atau muda tak layak sekarang mereka tidak akan tanggung jawab penuh dan yang dirugikan konsumen lebih baik kurangi belanja disini dan harus hati hati
44	semoga jd langganan
45	pelayanan bagus pengiriman cepat kualitas terjaga terimakasih
46	kalau emang tidak beroperasi pada saat malam takbiran minimal sediakan tombol cancel saya tidak jadi masak untuk lebaran karna kalian tidak ada kejelasan
47	pengiriman daging tidak seperti biasa nya tidak menggunakan bubble foil dan therma freeze jdi kualitas daging tidak bagus dan menjadi lembek
48	sehari lagi kenapa sih ko gabisa buka aplikasinya
49	mudah digunakan prosesnya cepat
50	mantap

Untuk melihat data hasil proses *data cleaning* dapat diakses di *link* berikut.

<https://docs.google.com/spreadsheets/d/1HCCG7kFhWAgT7WjzbwSZpbPunR7sB9Ub/edit?usp=sharing&oid=115406227280236311450&rtfpof=true&sd=true>


```

def remove_punctuation(text):
    """Remove punctuation"""
    if isinstance(text, str):
        text = re.sub(r'^\w\s', '', text)
    return text

def remove_non_ascii(text):
    """Remove non-ascii character"""
    if isinstance(text, str):
        text = unicodedata.normalize('NFKD', text).encode('ascii', 'ignore').decode('utf-8', 'ignore')
    return text

# Ensure all text values are strings
comments['text'] = comments['text'].astype(str)

# Apply cleaning functions
comments.loc[:, 'clean_text'] = comments['text'].apply(cleaning)
comments.loc[:, 'clean_text'] = comments['clean_text'].apply(remove_email)
comments.loc[:, 'clean_text'] = comments['clean_text'].apply(remove_between_square_brackets)
comments.loc[:, 'clean_text'] = comments['clean_text'].apply(remove_numbers)
comments.loc[:, 'clean_text'] = comments['clean_text'].apply(deleteEmoji)
comments.loc[:, 'clean_text'] = comments['clean_text'].apply(remove_punctuation)
comments.loc[:, 'clean_text'] = comments['clean_text'].apply(remove_non_ascii)

comments.head(5)

```

```

comment.head(5)

```

```

[37] Python
...

```

	userName	text	clean_text
0	Ade Pipin	sayuran nya fresh banget mantap deh pokoknya	sayuran nya fresh banget mantap deh pokoknya
1	Kedai Rumpun	Gambar nya ilang semua	gambar nya ilang semua
2	Salsabila	baru pertama kali order dan hasil nya memuaskan...	baru pertama kali order dan hasil nya memuaskan...
3	Dewi kumalasari	amanah	amanah
4	Elis Lisnawati	trmakasih segari barang fresh kurir ramah	trmakasih segari barang fresh kurir ramah

```

comment.to_excel(r'C:\Users\WivoBook\Desktop\skripsi\preprocess\cleaning_segari.xlsx', index=False, header=True)

```

```

[38] Python

```

Lampiran 5 *Stemming*

No	clean_text
1	sayur nya fresh banget mantap deh pokok
2	gambar ilang semua
3	baru pertama kali order dan hasil nya muas semua nya masih fresh bakal jadi langgan di sini
4	amanah
5	trmakasih segar barang fresh kurir ramah
6	turut qu bagus puas belanja di hari moga selalu amanah dan makin banyak lg promo nya
7	fresh produk harga ok dan kirim cepat mantaf
8	adminnya ramah dan responsif banget produk lumayan lengkap dan fresh bersih moga tetap kayak gin
9	selalu segar yg kirim
10	baru pertama mau coba pas masukin alamat nyata belum sedia
11	neng banget ada aplikasi ini sangat mudah dalam belanja karena lengkap segala macam ada thank you yaa founder and all team

12	respon cepat packing ok barang juga ok moga tahan terus ya dan makin tingkat goodluck
13	sayang sekali tidak sedia di seluruh kota aku di kalimantan barat kalo udah sedia di semua kota aku bintang deh
14	harga nya sungguh jangkau banding harga pasar
15	cxxul pp mj uu nny h di loo
16	alhamdulillah sangat bantu jauh ini barang yg beri kualitas semua
17	kirin cepat kilat
18	segar sangat bantu sy sahyrnya segar kwalitasnya bgs moga lbh bgs lg ya segar
19	layan buruk sering di cancle sudah dekat jam kirim kalau alas krna gak ada driver yanh pick up kenapa aplikasi antar sayur lain bisa pagi bahkan jam or ko ini gak bisa alasanya gak ada yg pick up kalau gitu jangan ada pilih antar pagi dengan jam yg customer bisa pilih mah jam berapa kalau ujung gak sanggup dan di cancle sangat kecewa
20	oke banget lah pokok
21	after sales nya bagus respon cs nya gercep kalo ada barang rusak atau telur pecah di ganti uang atau koin belanja bisa juga di kirim ulang barang yang rusak bener muas belanja di segar
22	baru pertama kali sih belanja di segar overall kirim cepat bahan belum buka karna baru besok mau olah tapi sekarang kasih bintang dulu karna walo baru pertama kali tapi gak ada kendala apa dan beli di online supermarket juga karna nyari bahan yg mau di supermarket dekat rumah nyata gak ada oke sekian moga makin makin baik depan user interface nya juga nyaman liat dan nyaman buat telusur thanks
23	sangat ngebantu
24	moga di bandung cepat ada layan
25	kirin cepat dan barang jamin bagus
26	puas belanja sini
27	pokok mantuull
28	oke
29	adain di bal dong
30	sangat bagus dan bantu dalam belanja bagai perlu
31	bantu sekali untuk belanja butuh masak cuman kurang lengkap di makeup ya
32	aplikasi sangat bantu
33	aku ksh bintang dulu yaa adain di kota sukabumi dong kadang suka mager ke pasar kl ada segar kaya enak nih buat emak yg lg mager
34	sayur segar
35	sangat mudah saya belanja hari terimakasih banyak
36	senang sekali belanja di segar harga nya sahabat packingnya baik dan kirim cepat
37	sangat kecewa pakai aplikasi ini layan buruk aplikasi lot jam kirim ubah sendiri tiap mau checkout stock kosong harus ubah lagi dan lagi sampai akhir jam kirim ubah sendiri tanpa kita sadar komplain ke cs layan buruk jangan pesan di sini deh daripada hati dongkol pesan dari pagi ngga sampe sampe

38	ok ok
39	moga waktu tepat max jam
40	ok sesuai pesan
41	alhamdulillah maa syaa allah the best pokok
42	kirim lamaaaa jauh dari estimasi awal
43	lama belanja gari tapi kali ini harus extra hati hati karena seperti main judi syukur buah sayur yang mereka kirim kualitas baik tapi jika yang kirim busuk atau muda tak layak sekarang mereka tidak akan tanggung jawab penuh dan yang rugi konsumen lebih baik rangin belanja sini dan harus hati hati
44	moga jd langgan
45	layan bagus kirim cepat kualitas jaga terimakasih
46	kalau emang tidak operasi pada saat malam takbir minimal sedia tombol cancel saya tidak jadi masak untuk lebaran karna kalian tidak ada jelas
47	kirim daging tidak seperti biasa nya tidak guna bubble foil dan therma freeze jdi kualitas daging tidak bagus dan jadi lembek
48	hari lagi kenapa sih ko gabisa buka aplikasi
49	mudah guna proses cepat
50	mantap

Untuk melihat data hasil proses *stemming* dapat diakses di *link* berikut.

<https://docs.google.com/spreadsheets/d/1s6Hb5qJjc0kzmBaFWZE1-Pa07eH2N1IX/edit?usp=sharing&ouid=115406227280236311450&rtpof=true&sd=true>

```
Stemming
review_data=pd.read_excel(r'C:\Users\VivoBook\Desktop\skripsi\preprocess\cleaning_segari.xlsx')
comment = review_data[['userName', 'text', 'clean_text']]
comment.head(5)
```

[25] ✓ 0.4s Python

	userName	text	clean_text
0	Ade Pipin	sayuran nya fresh banget mantap deh pokoknya	sayuran nya fresh banget mantap deh pokoknya
1	Kedai Rumpun	Gambar nya ilang semua	gambar nya ilang semua
2	Salsabila	baru pertama kali order dan hasil nya memuaskan	baru pertama kali order dan hasil nya memuaskan
3	Dewi kumalasari	amanah	amanah
4	Elis Lisnawati	trmakasih segari barang fresh kurir ramah	trmakasih segari barang fresh kurir ramah

```

from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

def stemming(text):
    """Stemming for Bahasa with Sastrawi"""
    factory = StemmerFactory()
    stemmer = factory.create_stemmer()

    # Daftar kata yang tidak ingin distemming
    non_stemming_words = ['Segari'] # Ganti dengan kata-kata Anda

    # Fungsi untuk mengembalikan kata asli jika ada dalam daftar non_stemming_words
    def custom_stemming(word):
        if word in non_stemming_words:
            return word
        else:
            return stemmer.stem(word)

    # Melakukan stemming dengan pengecekan non stemming words
    text = [custom_stemming(word) for word in text.split()]
    return ' '.join(text)

# Contoh penggunaan dengan data frame
comment['clean_text'] = comment['clean_text'].apply(stemming)

```

26] ✓ 20m 5.8s Python

```

comment.head(5)

```

	userName	text	clean_text
0	Ade Pipin	sayuran nya fresh banget mantap deh pokoknya	sayur nya fresh banget mantap deh pokok
1	Kedai Rumpun	Gambar nya hilang semua	gambar hilang semua
2	Salsabila	baru pertama kali order dan hasil nya memuaskan	baru pertama kali order dan hasil nya memuaskan
3	Dewi kumalasari	amanah	amanah
4	Elis Lisnawati	trmakasih segari barang fresh kurir ramah	trmakasih segar barang fresh kurir ramah

```

comment.to_excel(r'C:\Users\VivoBook\Desktop\skripsi\preprocess\stemming2\stemming2.xlsx', index=False, header=True)

```

28] ✓ 0.4s Python

Lampiran 6 Normalisasi

no	data_normalisasi
1	sayur nya segar sekali mantap deh pokok
2	gambar hilang semua
3	baru pertama kali pesanan dan hasil nya puas semua nya masih segar bakal jadi langgan di sini
4	amanah
5	terima kasih segari barang segar kurir ramah
6	turut saya bagus puas belanja di hari semoga selalu amanah dan semakin banyak lagi promo nya
7	segar produk harga oke dan kirim cepat mantap
8	admin nya ramah dan respons sekali produk lumayan lengkap dan segar bersih semoga tetap seperti gin
9	selalu segar yang kirim
10	baru pertama mau coba pas masukan alamat nyata belum sedia
11	neng sekali ada aplikasi ini sangat mudah dalam belanja karena lengkap segala macam ada terimakasih kamu ya founder dan semua team
12	respons cepat pengemasan oke barang juga oke semoga tahan terus ya dan semakin tingkat goodluck

13	sayang sekali tidak sedia di seluruh kota saya di kalimantan barat kalau sudah sedia di semua kota saya bintang deh
14	harga nya sungguh jangkau banding harga pasar
15	cxxul pp mj uu nny h di loh
16	syukur sangat bantu jauh ini barang yang beri kualitas semua
17	irim cepat kilat
18	segari sangat bantu saya sayur segar kualitas bagus semoga lebih bagus lagi ya segari
19	layan buruk sering di cancel sudah dekat jam kirim kalau alas karena tidak ada driver yan pick up kenapa aplikasi antar sayur lain bisa pagi bahkan jam orang kok ini tidak bisa alasan tidak ada yang pick up kalau begitu jangan ada pilih antar pagi dengan jam yang customer bisa pilih sih jam berapa kalau ujung tidak sanggup dan di cancel sangat kecewa
20	oke sekali lah pokok
21	setelah sales nya bagus respons cs nya cepat kalau ada barang rusak atau telur pecah di ganti uang atau koin belanja bisa juga di kirim ulang barang yang rusak benar puas belanja di segari
22	baru pertama kali memang belanja di segari keseluruhan kirim cepat bahan belum buka karena baru besok mau olah tapi sekarang kasih bintang dulu karena walo baru pertama kali tapi tidak ada kendala apa dan beli di online supermarket juga karena mencari bahan yang mau di supermarket dekat rumah nyata tidak ada oke sekian semoga semakin semakin baik depan user interface nya juga nyaman lihat dan nyaman buat telusur terimakasih
23	sangat membantu
24	semoga di bandung cepat ada layan
25	irim cepat dan barang jamin bagus
26	puas belanja sini
27	pokok mantuull
28	oke
29	mengadakan di bal dong
30	sangat bagus dan bantu dalam belanja bagai perlu
31	bantu sekali untuk belanja butuh masak cuma kurang lengkap di makeup ya
32	aplikasi sangat bantu
33	saya kasih bintang dulu ya mengadakan di kota sukabumi dong kadang suka mager ke pasar kalau ada segari kayak enak ini buat emak yang lagi mager
34	sayur segar
35	sangat mudah saya belanja hari terimakasih banyak
36	senang sekali belanja di segari harga nya sahabat pengemasannya baik dan kirim cepat
37	sangat kecewa pakai aplikasi ini layan buruk aplikasi lot jam kirim ubah sendiri tiap mau checkout persediaan kosong harus ubah lagi dan lagi sampai akhir jam kirim ubah sendiri tanpa kita sadar komplain ke cs layan buruk jangan pesan di sini deh daripada hati dongkol pesan dari pagi tidak sampai sampai

38	oke oke
39	semoga waktu tepat maksimal jam
40	oke sesuai pesan
41	syukur maa syaa allah the terbaik pokok
42	kirim lama jauh dari estimasi awal
43	lama belanja gari tapi kali ini harus extra hati hati karena seperti main judi syukur buah sayur yang mereka kirim kualitas baik tapi jika yang kirim busuk atau muda tidak layak sekarang mereka tidak akan tanggung jawab penuh dan yang rugi konsumen lebih baik rangin belanja sini dan harus hati hati
44	semoga jadi langgan
45	layan bagus kirim cepat kualitas jaga terimakasih
46	kalau memang tidak operasi pada saat malam takbir minimal sedia tombol batal saya tidak jadi masak untuk lebar karena kalian tidak ada jelas
47	kirim daging tidak seperti biasa nya tidak guna bubble foil dan therma freeze jadi kualitas daging tidak bagus dan jadi lembek
48	hari lagi kenapa memang kok tidak bisa buka aplikasi
49	mudah guna proses cepat
50	mantap

Untuk melihat data hasil proses normalisasi dapat diakses di *link* berikut.

<https://docs.google.com/spreadsheets/d/1Jzld7iJFo0htum5nPZsQ2nYPIUofcvvK/edit?usp=sharing&ouid=115406227280236311450&rtpof=true&sd=true>

```

Normalisasi
import pandas as pd

# Membaca data xlsx
hasil_normalize=pd.read_excel(r'C:\Users\WivoBook\Desktop\skripsi\preprocess\stemming2\stemming2.xlsx')
table_normalize = hasil_normalize[['text', 'clean_text']]
table_normalize.head(5)
✓ 0.5s Python

      text                                     clean_text
0  sayuran nya fresh banget mantap deh pokoknya  sayur nya fresh banget mantap deh pokok
1          Gambarnya ilang semua                    gambar ilang semua
2  baru pertama kali order dan hasil nya memuaska...  baru pertama kali order dan hasil nya muas smu...
3          amanah                                    amanah
4  trmakasih segari barang fresh kurir ramah      trmakasih segar barang fresh kurir ramah

comment = table_normalize
✓ 0.0s Python

table_normalize.head(20)
✓ 0.0s Python

```

```

import pandas as pd

def normalize_word(text):
    """Normalize slang words"""
    normal_word_path = pd.read_excel(r'C:\Users\VivoBook\Desktop\skripsi\katabaku.xlsx')

    # Convert all columns to strings to avoid comparison issues
    normal_word_path['singkat'] = normal_word_path['singkat'].astype(str)
    normal_word_path['hasil'] = normal_word_path['hasil'].astype(str)

    # Ensure text is a string
    text = str(text)

    # Normalize words
    text = ' '.join([normal_word_path[normal_word_path['singkat'] == word][['hasil']].values[0]
                    if (normal_word_path['singkat'] == word).any() else word for word in text.split()])
    return text

# Ensure all values in 'data_stemming' are strings
table_normalize['data_stemming'] = table_normalize['data_stemming'].fillna('').astype(str)

# Apply normalization
comment['data_normalisasi'] = comment['data_stemming'].apply(lambda x: normalize_word(x))

```

Lampiran 7 Stopword

no	data_stopword
1	sayur segar mantap pokok
2	gambar hilang
3	kali pesanan hasil puas segar langgan
4	amanah
5	terima kasih segari barang segar kurir ramah
6	bagus puas belanja semoga amanah promo
7	segar produk harga oke kirim cepat mantap
8	ramah respons produk lumayan lengkap segar bersih semoga gin
9	segar kirim
10	coba masukan alamat nyata sedia
11	aplikasi mudah belanja lengkap terimakasih founder team
12	respons cepat pengemasan oke barang oke semoga tahan tingkat goodluck
13	sayang tidak sedia kota kalimantan barat sedia kota bintang
14	harga sungguh jangkau banding harga pasar
15	
16	syukur bantu barang kualitas
17	kirim cepat kilat
18	segari bantu sayur segar kualitas bagus semoga bagus segari
19	layan buruk cancel jam kirim alas tidak driver yan pick aplikasi sayur pagi jam orang tidak alasan tidak pick pilih pagi jam customer pilih jam ujung tidak sanggup cancel kecewa
20	oke pokok

21	sales bagus respons cs cepat barang rusak telur pecah ganti uang koin belanja kirim ulang barang rusak puas belanja segari
22	kali belanja segari kirim cepat bahan buka besok olah kasih bintang walo kali tidak kendala beli online supermarket mencari bahan supermarket rumah nyata tidak oke sekian semoga user interface nyaman lihat nyaman telusur terimakasih
23	membantu
24	semoga bandung cepat layan
25	kirim cepat barang jamin bagus
26	puas belanja
27	pokok mantap
28	oke
29	mengadakan
30	bagus bantu belanja
31	bantu belanja butuh masak lengkap makeup
32	aplikasi bantu
33	kasih bintang mengadakan kota sukabumi kadang suka mager pasar segari kayak enak enak mager
34	sayur segar
35	mudah belanja terimakasih
36	senang belanja segari harga sahabat pengemasannya kirim cepat
37	kecewa pakai aplikasi layan buruk aplikasi lot jam kirim ubah checkout persediaan kosong ubah jam kirim ubah sadar komplain cs layan buruk pesan hati dongkol pesan pagi tidak
38	oke oke
39	semoga maksimal jam
40	oke sesuai pesan
41	syukur allah terbaik pokok
42	kirim estimasi
43	belanja gari kali extra hati hati main judi syukur buah sayur kirim kualitas kirim busuk muda tidak layak tidak tanggung penuh rugi konsumen rangin belanja hati hati
44	semoga langgan
45	layan bagus kirim cepat kualitas jaga terimakasih
46	tidak operasi malam takbir minimal sedia tombol batal tidak masak lebar tidak
47	kirim daging tidak tidak bubble foil therma freeze kualitas daging tidak bagus lembek
48	tidak buka aplikasi
49	mudah proses cepat
50	mantap

Untuk melihat data hasil proses *stopword* dapat diakses di *link* berikut.

<https://docs.google.com/spreadsheets/d/1OciUwteoutNp65Gf29igdDCQISspKdh8/edit?usp=sharing&ouid=115406227280236311450&rtpof=true&sd=true>

```

import pandas as pd
import nltk
from nltk.corpus import stopwords

# Download stopwords for Indonesian language
nltk.download('stopwords')

def stopwords_removal(text):
    """Stopword removal"""
    stopword = stopwords.words('indonesian')
    more_stopword = [
        'kalau', 'yang', 'pd', 'kl', 'hahaha', 'nya', 'pke', 'dehh', 'wkwk', 'dpt', 'jg', 'yg', 'km', 'ia', 'msih',
        'sih', 'ddpn', 'dsruh', 'dlu', 'trhbung', 'lah', 'sik', 'sak', 'the', 'wef', 'klaus', 'nih', 'gaes', 'guys',
        'gas', 'ct', 'dek', 'ive', 'kak', 'ka', 'aj', 'oiya', 'loh', 'kok', 'gek', 'wes', 'ada', 'bu', 'langsung',
        'knphehehe', 'hej', 'hem', 'teh', 'mah', 'da', 'atuh', 'jd', 'ak', 'lg', 'ya', 'rfr', 'gda', 'gc', 'gb',
        'tp', 'kalo', 'pake', 'liat', 'udh', 'aja', 'gt', 'gais', 'blm', 'tau', 'tahu', 'udah', 'utk', 'rb', 'rp',
        'dgn', 'ayo', 'isi', 'bian', 'yah', 'dn', 'bawa', 'gitu', 'eh', 'oh', 'noh', 'pas', 'td', 'sm', 'pengen', 'pgn',
        'sd', 'byr', 'min', 'ydh', 'yee', 'l', 'dan', 'di', 'l', 'ini', 'x008d', 'dy', 'and', 'neng', 'yaa',
        'oya', 'paaakkkk', 'pacing', 'l', '2', 'ooo', 'nii', 'pakkk', 'pat', 'nny', 'nm', 'ny', 'ngelo', 'nii', 'ngg',
        'nihh', 'nich', 'ni', 'nh', 'nyaa', 'nyaaaa', 'nyak', 'nye', 'of', 'oia', 'ocha', 'obe', 'oas', 'o', 'nyq', 'ptg',
        'rang', 're', 'rbu', 'zzzzzzzzzz', 'puts', 'putri', 'q', 'qu', 'qt', 'qda', 'ppp', 'pp', 'mang', 'mak2', 'ma',
        'm', 'm2', 'maa', 'll', 'mahh', 'maesaroh', 'lu', 'ltr', 'lt', 'lma', 'lm', 'loh', 'looh', 'looo', 'loo', 'naaahhhh',
        'nahhh', 'na', 'n', 'myk', 'mya', 'my', 'deh', 'mba', 'mu', 'ndasamu', 'nda', 'nd', 'mi', 'moms', 'mom', 'mo', 'mj', 'we',
        'waww', 'waww', 'waw', 'wkwkwkwk', 'wkwkwkwk', 'wkwkwk', 'wkwkwk', 'wooooww', 'wooooww', 'yng', 'yha', 'yh',
        'yg', 'yeah', 'youuuu', 'yra', 'zzk', 'zz', 'zeezzz', 'za', 'z', 'ys', 'wth', 'woyyyy', 'woy', 'woowwww', 'wow', 'x',
        'yaaah', 'yahhhh', 'yahhh', 'yahh', 'yag', 'yach', 'yaaaaaaa', 'yaaa', 'yaaa', 'y', 'xxss', 'tx', 'tul', 'u', 'ui',
        'tuk', 'uga', 'tu', 'tpp', 'ttg', 'tsb', 'trz', 'uu', 'v', 'uyk', 'uy', 'ux', 'up', 'rek', 'rt', 's', 's', 'ref',
        'rfr085132764501', 'rfr082126386297', 'rfr082123636614', 'rfr082111229263', 'rfr081991011904', 'rfr081389353244',
        'rfr081347719617', 'rfr081325422359', 'rfr081298875353', 'rfr081290346626', 'rfr081283030337', 'rfr081239655085',
        'rfr081214878725', 'rfr081213310574', 'rfr08099512709688', 'rfr08517479953', 'rfr085782213026', 'rfr085782678308',
        'rfr0809617785047', 'rfr081223618345', 'rfr08099741259', 'rfr080998277949', 'rfr0809689859380', 'rfr0809652016242',
        'rfr087880159255', 'rfr085939960526', 'rfr087784189644', 'rfr087782974932', 'rfr087782097107', 'rfr087761515641',
        'rfr087724069648', 'rfr087709050606', 'so', 'skr', 'sk', 'sj', 'siy', 'sllu', 'sll', 'si', 'lang', 'zzzzzzzzzz',
        'gg', 'gan', 'g', 'ge', 'aa', 'aduh', 'aaaahh', 'aaahhh', 'aaahhh', 'aaahhh', 'daaan', 'daahh', 'ab', 'hx', 'abai',
        'bal', 'abang', 'bapel', 'bestie', 'besty', 'boskuh', 'bosss', 'boskuhh', 'buibu', 'buk', 'abc', 'byeeeee', 'byee',
        'gaess', 'hehe', 'hehehe', 'heiiiii', 'hellow', 'heeee', 'hhhh', 'dah', 'dagh', 'im', 'imeldapf', 'abcde', 'did', 'hufft',
        'hufftt', 'huh', 'huhu', 'huhuuh', 'dam', 'if', 'ig', 'en', 'farhan', 'guys', 'dlli', 'heee', 'heehh', 'du', 'eliitha',
        'elizabethmart', 'emayramart', 'acr', 'diskaz', 'dih', 'hah', 'haha', 'hahaaaa', 'didnt', 'doonnnkk', 'dor', 'deuh', 'dinda',
        'deeh', 'dikab', 'adaaa', 'gr', 'adek', 'adding', 'accordingly', 'coz', 'adminnya', 'admint', 'agam', 'agst', 'agustus',
        'ah', 'ahir', 'ahza', 'ain', 'ajaa', 'animah', 'aju', 'ajuin', 'ajwa', 'ajwanya', 'alamt', 'alit', 'an', 'aulia', 'az', 'ba',
        'babang', 'bas', 'bb', 'be', 'belim', 'ber', 'bi', 'bl', 'bo', 'hiihihi', 'bon', 'boro', 'eeh', 'eehh', 'eeuyyyy', 'by', 'hk',
        'hm', 'ce', 'cihuy', 'dar', 'dllnya', 'dm', 'dnga', 'doanggg', 'doenk', 'doraemon', 'xixixixiii', 'yuk', 'you', 'your', 'wuhuu',
        'woooiii', 'byeeeee', 'byee', 'dsyur', 'duh', 'uhhhh', 'uwow', 'uwahhhh', 'ulalah', 'uhhhh', 'ude', 'tqu', 'tlt', 'te', 'ta',
        'yaa', 'ssh', 'ss', 'yaa' ] # add more stopword to default corpus
    stop_factory = stopword + more_stopword

    clean_words = []
    for word in str(text).split(): # Memastikan pemisahan per kata dan pastikan text adalah string
        if word not in stop_factory:
            clean_words.append(word)
    text = clean_words
    return text

def join_words(text):
    """Join all words"""
    text = ' '.join(text)
    return text

# Ensure all values in 'data_stemming' are strings
table_stopword['data_normalisasi'] = table_stopword['data_normalisasi'].fillna('').astype(str)

# Apply stopwords removal and join words using lambda functions
table_stopword['data_stopword'] = table_stopword['data_normalisasi'].apply(lambda x: stopwords_removal(x))
table_stopword['data_stopword'] = table_stopword['data_stopword'].apply(lambda x: join_words(x))

# Display the result
print(table_stopword.head())

```

Lampiran 8 *Inset Lexicon*

text	sentiment	kategori_sentiment
sayur segar mantap pokok	12	Positif
gambar hilang	-2	Negatif
kali pesanan hasil puas segar langgan	14	Positif
amanah	5	Positif
terima kasih segari barang segar kurir ramah	17	Positif
bagus puas belanja semoga amanah promo	16	Positif

segar produk harga oke kirim cepat mantap	22	Positif
ramah respons produk lumayan lengkap segar bersih semoga gin	24	Positif
segar kirim	6	Positif
coba masukan alamat nyata sedia	5	Positif
aplikasi mudah belanja lengkap terimakasih founder team	10	Positif
respons cepat pengemasan oke barang oke semoga tahan tingkat goodluck	17	Positif
sayang tidak sedia kota kalimantan barat sedia kota bintang	-1	Negatif
harga sungguh jangkau banding harga pasar	9	Positif
syukur bantu barang kualitas	12	Positif
kirim cepat kilat	8	Positif
segari bantu sayur segar kualitas bagus semoga bagus segari	24	Positif
layan buruk cancel jam kirim alas tidak driver yanh pick aplikasi sayur pagi jam orang tidak alasan tidak pick pilih pagi jam customer pilih jam ujung tidak sanggup cancel kecewa	-13	Negatif
oke pokok	5	Positif
sales bagus respons cs cepat barang rusak telur pecah ganti uang koin belanja kirim ulang barang rusak puas belanja segari	28	Positif
kali belanja segari kirim cepat bahan buka besok olah kasih bintang walo kali tidak kendala beli online supermarket mencari bahan supermarket rumah nyata tidak oke sekian semoga user interface nyaman lihat nyaman telusur terimakasih	35	Positif
membantu	4	Positif
semoga bandung cepat layan	6	Positif
kirim cepat barang jamin bagus	2	Positif
puas belanja	5	Positif
pokok mantap	8	Positif
oke	2	Positif
mengadakan	4	Positif
bagus bantu belanja	7	Positif
bantu belanja butuh masak lengkap makeup	4	Positif
aplikasi bantu	-1	Negatif

kasih bintang mengadakan kota sukabumi kadang suka mager pasar segari kayak enak emak mager	10	Positif
sayur segar	4	Positif
mudah belanja terimakasih	11	Positif
senang belanja segari harga sahabat pengemasannya kirim cepat	27	Positif
kecewa pakai aplikasi layan buruk aplikasi lot jam kirim ubah checkout persediaan kosong ubah jam kirim ubah sadar komplain cs layan buruk pesan hati dongkol pesan pagi tidak	-8	Negatif
oke oke	4	Positif
semoga maksimal jam	4	Positif
oke sesuai pesan	7	Positif
syukur allah terbaik pokok	18	Positif
kirim estimasi	2	Positif
belanja gari kali extra hati hati main judi syukur buah sayur kirim kualitas kirim busuk muda tidak layak tidak tanggung penuh rugi konsumen rangin belanja hati hati	22	Positif
semoga langgan	3	Positif
layan bagus kirim cepat kualitas jaga terimakasih	20	Positif
tidak operasi malam takbir minimal sedia tombol batal tidak masak lebar tidak	-7	Negatif
kirim daging tidak tidak bubble foil therma freeze kualitas daging tidak bagus lembek	9	Positif
tidak buka aplikasi	-7	Negatif
mudah proses cepat	8	Positif
mantap	5	Positif
manfaat bahan segar	6	Positif

Untuk melihat data hasil proses *labelling Inset Lexicon* dapat diakses di *link* berikut.

<https://docs.google.com/spreadsheets/d/19unkMiiijhcjDa6vTMZFO4sern9JhtqH/e/dit?usp=sharing&ouid=115406227280236311450&rtpof=true&sd=true>

```

negasi = ['bukan', 'tidak', 'ga', 'gk']
lexicon = pd.read_excel(r'C:\Users\VivoBook\Desktop\skripsi\inset-lexicon\inset_lexicon.xlsx')
lexicon = lexicon.drop(lexicon[(lexicon['word'] == 'bukan')
                               |(lexicon['word'] == 'tidak')
                               |(lexicon['word'] == 'ga')
                               |(lexicon['word'] == 'gk') ].index,axis=0)
lexicon = lexicon.reset_index(drop=True)

```

154] Python

```

lexicon.head(20)

```

155] Python

	word	weight	number_of_words
0	hai	3	1
1	merekam	2	1
2	ekstensif	3	1
3	paripurna	1	1
4	detail	2	1
5	pernik	3	1
6	belas	2	1
7	welas	4	1
8	kabung	1	1
9	rahayu	4	1
10	maaf	2	1
11	hello	2	1
12	promo	3	1

```

lexicon_word = lexicon['word'].to_list()
lexicon_num_words = lexicon['number_of_words']

```

156] Python

```

import nltk
from nltk.tokenize import word_tokenize

# Pastikan nltk tokenizers sudah diunduh
nltk.download('punkt')

# Pastikan semua nilai di kolom 'clean_text' adalah string
comment['clean_text'] = comment['clean_text'].fillna('').astype(str)

word_dict = {}
for i in range(len(comment['clean_text'])):
    sentence = comment['clean_text'][i]
    word_token = word_tokenize(sentence)
    for j in word_token:
        if j not in word_dict:
            word_dict[j] = 1
        else:
            word_dict[j] += 1

# Display the word dictionary
word_dict

```

157] Python

```

#Checking if there is words in dictionary that does not included in the lexicon
ns_words = []
factory = StemmerFactory()
stemmer = factory.create_stemmer()
for word in word_dict.keys():
    if word not in lexicon_word:
        kata_dasar = stemmer.stem(word)
        if kata_dasar not in lexicon_word:
            ns_words.append(word)
len(ns_words)

```

2279

```

len({k:v for (k,v) in word_dict.items() if ((k in ns_words)&(v>3)) })

```

358

```

#It turns out that the words that is not included in lexicon, are the one that not have signigance sentiment
ns_words_list = {k:v for (k,v) in word_dict.items() if ((k in ns_words)&(v>3))}

```

Python

```

#It turns out that the words that is not included in lexicon, are the one that not have signigance sentiment
ns_words_list = {k:v for (k,v) in word_dict.items() if ((k in ns_words)&(v>3))}

sort_orders = sorted(ns_words_list.items(), key=lambda x: x[1], reverse=True)
sort_orders=sort_orders[0:20]
for i in sort_orders:
    print(i[0], i[1])

barang 987
sayur 669
kali 304
jam 267
daging 225
kurir 223
ongkos 183
pakai 178
cs 146
pasar 135
minyak 133
refund 131
jangkau 122
ikan 84
ribu 81
pesenan 73
bintang 72
online 66
orang 64
uang 63

```

```

#calculating the sentiment of words by mathing them to the lexicon while also creating the bag of words matrix

import numpy as np

sencol = []
senrow = np.array([])
nsen = 0
factory = StemmerFactory()
stemmer = factory.create_stemmer()
sentiment_list = []
# function to write the word's sentiment if it is founded
def found_word(ind,words,word,sen,sencol,sentiment,add):
    # if it is already included in the bag of words matrix, then just increase the value
    if word in sencol:
        sen[sencol.index(word)] += 1
    else:
        #if not, than add new word
        sencol.append(word)
        sen.append(1)
        add += 1
    #if there is a negation word before it, the sentiment would be the negation of it's sentiment
    if (words[ind-1] in negasi):
        sentiment += -lexicon['weight'][lexicon_word.index(word)]
    else:
        sentiment += lexicon['weight'][lexicon_word.index(word)]
    return sen,sencol,sentiment,add

```

```

# checking every words, if they are appear in the lexicon, and then calculate their sentiment if they do
for i in range(len(comment)):
    nsen = senrow.shape[0]
    words = word_tokenize(comment['clean_text'][i])
    sentiment = 0
    add = 0
    prev = [0 for ii in range(len(words))]
    n_words = len(words)
    if len(sencol)>0:
        sen =[0 for j in range(len(sencol))]
    else:
        sen =[]

    for word in words:
        ind = words.index(word)
        # check whether they are included in the lexicon
        if word in lexicon_word :
            sen,sencol,sentiment,add= found_word(ind,words,word,sen,sencol,sentiment,add)
        else:
            # if not, then check the root word
            kata_dasar = stemmer.stem(word)
            if kata_dasar in lexicon_word:
                sen,sencol,sentiment,add= found_word(ind,words,kata_dasar,sen,sencol,sentiment,add)
            # if still negative, try to match the combination of words with the adjacent words
            elif(n_words>1):
                if ind-1>-1:
                    back_1 = words[ind-1]+' '+word
                    if (back_1 in lexicon_word):
                        sen,sencol,sentiment,add= found_word(ind,words,back_1,sen,sencol,sentiment,add)
                elif(ind-2>-1):
                    back_2 = words[ind-2]+' '+back_1
                    if back_2 in lexicon_word:
                        sen,sencol,sentiment,add= found_word(ind,words,back_2,sen,sencol,sentiment,add)

```

```

# if there is new word founded, then expand the matrix
if add>0:
    if i>0:
        if (nsen==0):
            senrow = np.zeros([i,add],dtype=int)
        elif(i!=nsen):
            padding_h = np.zeros([nsen,add],dtype=int)
            senrow = np.hstack((senrow,padding_h))
            padding_v = np.zeros([(i-nsen),senrow.shape[1]],dtype=int)
            senrow = np.vstack((senrow,padding_v))
        else:
            padding =np.zeros([nsen,add],dtype=int)
            senrow = np.hstack((senrow,padding))
            senrow = np.vstack((senrow,sen))
    if i==0:
        senrow = np.array(sen).reshape(1,len(sen))
# if there isn't then just update the old matrix
elif(nsen>0):
    senrow = np.vstack((senrow,sen))

sentiment_list.append(sentiment)

```

Python

```
len(sentiment_list)
```

Python

```
[164] len(sentiment_list)
```

Python

```
... 5632
```

```
print(senrow.shape[0])
```

```
[165]
```

Python

```
... 5632
```

```

#constructing the data frame that contain bag of words and the sentiments that have been calculated before

sencol.append('sentiment')
sentiment_array = np.array(sentiment_list).reshape(senrow.shape[0],1)
sentiment_data = np.hstack((senrow,sentiment_array))
df_sen = pd.DataFrame(sentiment_data,columns = sencol)

```

```
[166]
```

Python

```
df_sen.head(10)
```

```
[167]
```

Python

```

...
   segar mantap pokok gambar hilang pesan hasil puas langgan amanah ... asam bain darah berani sila bangkai tawaran jaman stabil sentime
0      1      1      1      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0
1      0      0      0      1      1      0      0      0      0      0      0      0      0      0      0      0      0      0      0
2      1      0      0      0      0      1      1      1      1      0      0      0      0      0      0      0      0      0      0
3      0      0      0      0      0      0      0      0      0      0      1      0      0      0      0      0      0      0      0
4      2      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0
5      0      0      0      0      0      0      0      0      1      0      1      0      0      0      0      0      0      0      0
6      1      1      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0
7      1      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0
8      1      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0
9      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0      0

```

```
10 rows x 848 columns
```

```

#lets see if the sentiment is correct by looking at the original text

cek_label = pd.DataFrame([])
cek_label['text'] = comment['clean_text'].copy()
cek_label['sentiment'] = df_sen['sentiment'].copy()

```

```
[168]
```

Python

```

cek_label.head(10)

```

	text	sentiment
0	sayur segar mantap pokok	12
1	gambar hilang	-2
2	kali pesanan hasil puas segar langgan	14
3	amanah	5
4	terima kasih segari barang segar kurir ramah	17
5	bagus puas belanja semoga amanah promo	16
6	segar produk harga oke kirim cepat mantap	22
7	adminnya ramah respons produk lumayan lengkap ...	24
8	segar kirim	6
9	coba masukan alamat nyata sedia	5

```

cek_label.to_csv('/content/drive/MyDrive/data-segari-gabungan/hasil_label_inset.csv', index=False, header=True)

```

```

#Kategori Kelas Sentimen dari hasil Pelabelan Data

def kategorikan_kelas(polarity_score):
    if polarity_score > 0:
        return "Positif"
    elif polarity_score < 0:
        return "Negatif"
    else:
        return "Netral"

cek_label['kategori_sentiment'] = cek_label['sentiment'].apply(kategorikan_kelas)

cek_label.head(10)

```

	text	sentiment	kategori_sentiment
0	sayur segar mantap pokok	12	Positif
1	gambar hilang	-2	Negatif
2	kali pesanan hasil puas segar langgan	14	Positif
3	amanah	5	Positif
4	terima kasih segari barang segar kurir ramah	17	Positif
5	bagus puas belanja semoga amanah promo	16	Positif
6	segar produk harga oke kirim cepat mantap	22	Positif
7	adminnya ramah respons produk lumayan lengkap ...	24	Positif
8	segar kirim	6	Positif
9	coba masukan alamat nyata sedia	5	Positif

```

import matplotlib.pyplot as plt

# Data jumlah setiap kategori kelas
jumlah_kategori = cek_label["kategori_sentiment"].value_counts()

# Plot grafik bar
plt.figure(figsize=(8, 6))
jumlah_kategori.plot(kind='bar', color='skyblue')
plt.title('Jumlah Setiap Kategori Kelas Sentimen')
plt.xlabel('Kategori Sentimen')
plt.ylabel('Jumlah Komentar')
plt.xticks(rotation=45)

# Menambahkan keterangan angka/jumlah di (variable) jumlah_kategori: Series[int]
for i in range(len(jumlah_kategori)):
    plt.text(i, jumlah_kategori[i], str(jumlah_kategori[i]), ha='center', va='bottom')

plt.tight_layout()

# Tampilkan grafik
plt.show()

```

```

import matplotlib.pyplot as plt

# Data jumlah setiap kategori kelas
jumlah_kategori = cek_label['kategori_sentiment'].value_counts()

# Plot diagram lingkaran
plt.figure(figsize=(8, 6))
grafik = jumlah_kategori.plot(kind='pie', autopct='%1.1f%%', colors=['lightgreen', 'lightcoral', 'lightskyblue'])
plt.title('Jumlah Setiap Kategori Kelas Sentimen')

# Menambahkan keterangan untuk setiap slice
plt.legend(labels=jumlah_kategori.index, loc="best")

plt.tight_layout()

# Tampilkan grafik
plt.show()

```

Lampiran 9 Pembobotan dengan TF-IDF

TF-IDF untuk teks 'sayur segar mantap pokok':

sayur: 0.44594467118906206
 segar: 0.3255565479607618
 mantap: 0.5357944863511211
 pokok: 0.6388040017984311

Kategori sentimen untuk teks 'sayur segar mantap pokok': Positif

=====

TF-IDF untuk teks 'gambar hilang':

gambar: 0.7071067811865476
 hilang: 0.7071067811865476

Kategori sentimen untuk teks 'gambar hilang': Negatif

=====

TF-IDF untuk teks 'kali pesanan hasil puas segar langgan':

kali: 0.36276195736166955
 pesanan: 0.41118553153691983
 hasil: 0.6202521570873196
 puas: 0.25117120613682237
 segar: 0.20664965667581392
 langgan: 0.45697530259074187

Kategori sentimen untuk teks 'kali pesanan hasil puas segar langgan': Positif

=====

TF-IDF untuk teks 'amanah':

amanah: 1.0

Kategori sentimen untuk teks 'amanah': Positif

=====

TF-IDF untuk teks 'terima kasih segari barang segar kurir ramah':

terima: 0.44340327964478704
kasih: 0.41731301810268073
segari: 0.26335993874729036
barang: 0.29426765182059356
segar: 0.23727381484222382
kurir: 0.44340327964478704
ramah: 0.4694530185947904

Kategori sentimen untuk teks 'terima kasih segari barang segar kurir ramah':
Positif

TF-IDF untuk teks 'bagus puas belanja semoga amanah promo':

bagus: 0.29285900520365704
puas: 0.3121156202982979
belanja: 0.2768856697027655
semoga: 0.4728993614433775
amanah: 0.617552527644122
promo: 0.36762322845133943

Kategori sentimen untuk teks 'bagus puas belanja semoga amanah promo': Positif

TF-IDF untuk teks 'segar produk harga oke kirim cepat mantap':

segar: 0.2645689116383528
produk: 0.40787202234235886
harga: 0.4084118523478196
oke: 0.41395395344456243
kirim: 0.32731313577226034
cepat: 0.35883027236627624
mantap: 0.4354222484655157

Kategori sentimen untuk teks 'segar produk harga oke kirim cepat mantap': Positif

TF-IDF untuk teks 'ramah respons produk lumayan lengkap segar bersih semoga
gin':

ramah: 0.308811695775293
respons: 0.33681510815903876
produk: 0.2406226595253949
lumayan: 0.3513367358366486
lengkap: 0.3687722495048612
segar: 0.1560814953194406
bersih: 0.38394453662358174
semoga: 0.287435101901942

gin: 0.46847348169162456

Kategori sentimen untuk teks 'ramah respons produk lumayan lengkap segar bersih semoga gin': Positif

TF-IDF untuk teks 'segar kirim':

segar: 0.6286255325171978

kirim: 0.7777081328284216

Kategori sentimen untuk teks 'segar kirim': Positif

TF-IDF untuk teks 'coba masukan alamat nyata sedia':

coba: 0.4066489619467804

masukan: 0.5246952219587635

alamat: 0.4366597623424034

nyata: 0.41249672199501936

sedia: 0.44554040456022537

Kategori sentimen untuk teks 'coba masukan alamat nyata sedia': Positif

Untuk melihat data hasil proses pembobotan dengan TF-IDF dapat diakses di *link* berikut.

https://drive.google.com/file/d/1F1Uwr84Ccbz9NFGuBh0L8Pd4or0Kx_G/view?usp=sharing

```

import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer

# Memeriksa apakah kolom 'text' ada dalam DataFrame
if 'text' in review.columns:
    # Menghapus baris dengan nilai NaN atau kosong di kolom 'text'
    review = review.dropna(subset=['text'])
    review = review[review['text'].str.strip() != '']

# Memeriksa apakah DataFrame tidak kosong setelah pembersihan
if not review.empty:
    # Inisialisasi TfidfVectorizer
    tfidf_vectorizer = TfidfVectorizer()

    # Fit dan transformasi data teks di kolom 'text'
    tfidf_matrix = tfidf_vectorizer.fit_transform(review['text'])

    # Menyimpan hasil TF-IDF dan kategori sentimen ke dalam list of dictionaries
    results = []
    feature_names = tfidf_vectorizer.get_feature_names_out()
    for i, row in review.iterrows():
        tfidf_dict = {word: tfidf_matrix[i, tfidf_vectorizer.vocabulary_[word]] for word in row['text'].split() if word in tfidf_vectorizer.vocabulary_}
        result = {
            'teks': row['text'],
            'tf_idf': tfidf_dict,
            'kategori_sentimen': row['kategori_sentimen']
        }
        results.append(result)

    # Membuat DataFrame dari hasil tersebut
    kategori_sentimen = pd.DataFrame(results)

    # Menyimpan output ke file teks
    with open(r'C:\Users\WivoBook\Desktop\skripsi\tfidf\hasil_tfidf_dokumen.txt', 'w') as f:
        for result in results:
            f.write(f"TF-IDF untuk teks '{result['teks']}':\n")
            for word, tfidf_value in result['tf_idf'].items():
                f.write(f"  {word}: {tfidf_value}\n")
            f.write(f"\nKategori sentimen untuk teks '{result['teks']}': {result['kategori_sentimen']}\n")
            f.write("-"*50 + "\n")

    print("Data berhasil disimpan ke file teks.")
else:
    print("DataFrame kosong setelah pembersihan.")
else:
    print("Kolom 'text' tidak ditemukan dalam DataFrame review.")

```

Lampiran 10 hasil *Labelling* Dengan Model SVM yang Telah Dibuat

sentiment	text	predicted_sentiment
1	sayur segar mantap pokok	1
0	gambar hilang	0
1	kali pesanan hasil puas segar langgan	1
1	amanah	1
1	terima kasih segari barang segar kurir ramah	1
1	bagus puas belanja semoga amanah promo	1
1	segar produk harga oke kirim cepat mantap	1
1	ramah respons produk lumayan lengkap segar bersih semoga gin	1
1	segar kirim	1
1	coba masukan alamat nyata sedia	1
1	aplikasi mudah belanja lengkap terimakasih founder team	1
1	respons cepat pengemasan oke barang oke semoga tahan tingkat goodluck	1

0	sayang tidak sedia kota kalimantan barat sedia kota bintang	1
1	harga sungguh jangkau banding harga pasar	1
1	syukur bantu barang kualitas	1
1	kirim cepat kilat	1
1	segari bantu sayur segar kualitas bagus semoga bagus segari	1
0	layan buruk cancel jam kirim alas tidak driver yanh pick aplikasi sayur pagi jam orang tidak alasan tidak pick pilih pagi jam customer pilih jam ujung tidak sanggup cancel kecewa	0
1	oke pokok	1
1	sales bagus respons cs cepat barang rusak telur pecah ganti uang koin belanja kirim ulang barang rusak puas belanja segari	1
1	kali belanja segari kirim cepat bahan buka besok olah kasih bintang walo kali tidak kendala beli online supermarket mencari bahan supermarket rumah nyata tidak oke sekian semoga user interface nyaman lihat nyaman telusur terimakasih	1
1	membantu	1
1	semoga bandung cepat layan	1
1	kirim cepat barang jamin bagus	1
1	puas belanja	1
1	pokok mantap	1
1	oke	1
1	mengadakan	1
1	bagus bantu belanja	1
1	bantu belanja butuh masak lengkap makeup	1
0	aplikasi bantu	0
1	kasih bintang mengadakan kota sukabumi kadang suka mager pasar segari kayak enak emak mager	1
1	sayur segar	1
1	mudah belanja terimakasih	1
1	senang belanja segari harga sahabat pengemasannya kirim cepat	1
0	kecewa pakai aplikasi layan buruk aplikasi lot jam kirim ubah checkout persediaan kosong ubah jam kirim ubah sadar komplain cs layan buruk pesan hati dongkol pesan pagi tidak	0
1	oke oke	1

1	semoga maksimal jam	1
1	oke sesuai pesan	1
1	syukur allah terbaik pokok	1
1	kirim estimasi	1
1	belanja gari kali extra hati hati main judi syukur buah sayur kirim kualitas kirim busuk muda tidak layak tidak tanggung penuh rugi konsumen rangin belanja hati hati	1
1	semoga langgan	1
1	layan bagus kirim cepat kualitas jaga terimakasih	1
0	tidak operasi malam takbir minimal sedia tombol batal tidak masak lebar tidak	0
1	kirim daging tidak tidak bubble foil therma freeze kualitas daging tidak bagus lembek	1
0	tidak buka aplikasi	0
1	mudah proses cepat	1
1	mantap	1
1	manfaat bahan segar	1

Untuk melihat data hasil proses *Labelling* dengan model SVM yang telah dibuat sebelumnya dapat diakses di *link* berikut.

<https://docs.google.com/spreadsheets/d/1OMJO1pwJ2o6FN7HuE8q5-h5vNAHKjN4s/edit?usp=sharing&oid=115406227280236311450&rtfpof=true&d=true>

```
import pickle
import pandas as pd

# Misalkan df_tfidf dan df_labels sudah ada dan diisi dengan data yang sesuai

# Memuat model yang telah disimpan
with open(r'C:\Users\VivoBook\Desktop\skripsi\svm\svm_model_linear.pkl', 'rb') as model_file:
    svm_model_linear = pickle.load(model_file)

# Memastikan df_tfidf adalah representasi TF-IDF dari teks
X_new = df_tfidf

# Membuat prediksi pada data baru
predictions = svm_model_linear.predict(X_new)

# Menambahkan hasil prediksi sebagai kolom baru di DataFrame asli (df_labels)
df_labels['predicted_sentiment'] = predictions

# Menampilkan DataFrame dengan label yang diprediksi
print(df_labels)

# Menghitung jumlah prediksi positif dan negatif
pred_count = df_labels['predicted_sentiment'].value_counts()

# Menampilkan jumlah prediksi positif dan negatif
print("Jumlah prediksi positif:", pred_count.get(1))
print("Jumlah prediksi negatif:", pred_count.get(0))
```

Lampiran 11 Korpus katabaku untuk proses normalisasi

_id	singkat	hasil
1	abis	habis
2	accent	tekanan
3	accept	terima
4	accident	kecelakaan
5	achievement	prestasi
6	acra	acara
7	acrany	acaranya
8	acrnya	acaranya
9	action	aksi
10	active	aktif
11	activity	aktivitas
12	actually	sebenarnya
13	actualy	sebenarnya
14	ad	ada
15	ade	ada
16	adult	dewasa
17	adventure	petualangan
18	adventurer	petualang
19	advice	nasehat
20	after	setelah
21	afternun	sore
22	again	lagi
23	agency	perwakilan
24	agent	agen
25	agk	agak
26	agktn	angkatan
27	agree	setuju
28	agreement	persetujuan
29	aing	saya
30	aj	saja
31	aja	saja
32	ajah	saja
33	aje	saja
34	ajeh	saja
35	ajk	ajak
36	ak	saya
37	akeh	banyak
38	akhire	akhirnya

39	aktifkn	aktifkan
40	aku	saya
41	alhamdlh	alhamdulillah
42	alhamdulillah	alhamdulillah
43	almost	hampir
44	almt	alamat
45	alone	sendiri
46	alsn	alasan
47	also	juga
48	always	selalu
49	ama	sama
50	amazing	Mengagumkan

Untuk melihat korpus katabaku yang dipakai untuk proses normalisasi dapat diakses di *link* berikut.

<https://docs.google.com/spreadsheets/d/1A17HpKutT0sfVH6O1zDt4FYWp-2H-hRQ/edit?usp=sharing&oid=115406227280236311450&rtpof=true&sd=true>