

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan pada data tweet tentang penanggulangan bencana Gunung Merapi meletus tahun 2010 di Yogyakarta terkait kebutuhan khusus dimulai tanggal 26 September 2010 hingga 31 Desember 2010 dapat disimpulkan sebagai berikut

1. Berdasarkan data yang telah melalui proses klasifikasi menggunakan metode *Naïve Bayes Classifier* menunjukkan akurasi tertinggi yaitu 97.22%. Nilai akurasi ini lebih baik dibandingkan dengan metode Support Vector Machine dengan nilai yaitu 88.75%. Selain akurasi, nilai ROC dari rata-rata 0.995 yang menunjukkan metode *Naïve Bayes Classifier* dapat mengklasifikasi data dengan baik.
2. Berdasarkan sentimen masyarakat berdasarkan kategori dapat disimpulkan bahwa kategori yang memiliki sentimen positif tertinggi terdapat pada kebutuhan psikologi kategori pemulihan dengan jumlah 118 tweet yang menginformasikan bahwa kategori pemulihan sebagian telah tertangani dengan baik, dengan kata positif yaitu para relawan memberikan bantuan berupa pemulihan fisik

kepada korban serta upaya dalam melakukan pemulihan pendidikan dan pemulihan tani. Dibandingkan dengan sentimen negatif pada kebutuhan psikologi kategori psikolog memiliki nilai tertinggi dengan jumlah 96 tweet yang menginformasikan bahwa kategori psikolog belum tertangani dengan baik, dengan kata negatif yaitu para korban membutuhkan bimbingan konseling dari para psikolog dikarenakan banyak derita korban yang mengalami gangguan psikologis.

3. Berdasarkan interpretasi data dengan cara pemetaan, dapat disimpulkan bahwa lokasi Kabupaten Yogyakarta dan Kabupaten Klaten telah ditangani dengan baik seperti penanganan pendidikan dan penghiburan untuk para korban, dibuktikan dengan banyaknya jumlah sentimen positif pada kategori tersebut berupa penghimpauan sekolah, serta penghiburan bagi para korban bencana. Pada 3 kabupaten yaitu Kabupaten Sleman, Kabupaten Magelang, dan Kabupaten Boyolali masih belum ditangani dengan baik seperti penanganan kebutuhan balita dan lansia yang masih kekurangan asupan makanan, pakaian, obat-obatan, keperluan mandi dan tidur serta membutuhkan bantuan untuk memenuhi kebutuhan para balita dan lansia korban bencana dibuktikan dengan banyaknya sentimen negatif berupa meminta bantuan, serta laporan kekurangan kebutuhan tersebut.

5.2 Saran

Adapun beberapa saran yang disampaikan untuk pengembangan penelitian tentang analisis sentimen kebutuhan khusus bencana Gunung Merapi Yogyakarta menggunakan metode *Naïve Bayes Classifier* berdasarkan data *Twitter* adalah sebagai berikut :

1. Dari hasil penelitian ini masih ditemukan banyak sentimen netral yang terlihat pada kelas sentimen masyarakat terhadap bencana Gunung Merapi meletus 2010 di Yogyakarta dikarenakan kamus leksikon yang digunakan dalam penelitian masih belum lengkap. Diharapkan untuk penelitian selanjutnya untuk melengkapi lebih banyak kamus leksikon untuk menunjang penetapan kelas sentimennya. Hal ini dapat meningkatkan nilai akurasi yang didapat dalam proses klasifikasi.
2. Dari hasil penelitian ini masih menggunakan query yang cukup sedikit untuk mengumpulkan data. Diharapkan untuk menambahkan query lain agar data semakin lengkap untuk penelitian selanjutnya.

DAFTAR PUSTAKA

- BADAN NASIONAL PENANGGULANGAN BENCANA. (2008). Pedoman Penyusunan Rencana Penanggulangan Bencana.
- Buntoro, G. A. (2017). Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter, *2(1)*, 32-41.
- Bustami. (2014). Penerapan Algoritma Naive Bayes untuk Mengklasifikasi Data Nasabah Asuransi. *Jurnal Penelitian Teknik Informatika Universitas Ahmad Dahlan (TECHSI)*.
- Defiyanti, S. (2015). Integrasi Metode Klasifikasi Dan Clustering dalam Data Mining, (January), 39-44.
- Effendi, Q. (2015). Rancang Bangun Sistem Informasi Klasifikasi Status Gunung Berapi Menggunakan Metode Naive Bayes Classifier.
- Fachrurrozi, M. (2016). Sosial Menggunakan Metode Support, (June).
- Fillah, A. S., & Fedryansyah, M. (n.d.). Program Penanggulangan Bencana Oleh Disaster Management Center (Dmc) Dompot Dhuafa.
- Frank, E., Hall, M. A., & Witten, I. H. (2016). Workbench.
- Gorunescu, F. (2011). *Data Mining: Concepts and Techniques. Elsevier* (Vol. 12).
<https://doi.org/10.1007/978-3-642-19721-5>

- Handayani, F., & Pribadi, S. (2015). Implementasi Algoritma Naive Bayes Classifier dalam Pengklasifikasian Teks Otomatis Pengaduan dan Pelaporan Masyarakat melalui Layanan Call Center 110, 7(1).
- Informasi, J. S., & Informatika, J. T. (2005). 2 2 1, 2, 2-4.
- Jember. (2009). No Title, (1310652051), 1-6.
- Nirmalawati. (2011). Pembentukan konsep diri pada siswa pendidikan dasar dalam memahami mitigasi bencana. *Jurnal SMARTek*, 9(1), 61-69.
- Novantirani, A., Sabariah, M. K., & Effendy, V. (2015). Analisis Sentimen pada Twitter untuk Mengenai Penggunaan Transportasi Umum Darat Dalam Kota dengan Metode Support Vector Machine, 2(1), 1-7.
- Nurrun & Paulus. (2016). STUDI LITERATUR TENTANG PERBANDINGAN METODE UNTUK PROSES, 2016(Sentika), 18-19.
- Pamungkas, D. S., Setiyanto, N. A., & Dolphina, E. (2015). Analisis Sentiment Pada Sosial Media Twitter Menggunakan Naive Bayes Classifier Terhadap Kata Kunci "Kurikulum 2013", 14(4), 299-314.
- Parveen, H., & Pandey, S. (2017). Sentiment analysis on Twitter Data-set using Naive Bayes algorithm. *Proceedings of the 2016 2nd International Conference*

on Applied and Theoretical Computing and Communication
Technology, ICATccT 2016, 416-419.

<https://doi.org/10.1109/ICATCCT.2016.7912034>

Pratama, A. (2010). 1. Pendahuluan, 4(100), 4-20.

Recovery, G. N. on. (2009). No Title.

Rosandy, T. (2016). PERBANDINGAN METODE NAIVE BAYES
CLASSIFIER DENGAN METODE DECISION TREE (C4 . 5)
UNTUK MENGANALISA KELANCARAN PEMBIAYAAN (Study
Kasus : KSPPS / BMT AL-FADHILA). *Jurnal TIM
Darmajaya*, 02(01), 52-62.

Rozi, I. F., Hadi, S., & Achmad, E. (2012). Implementasi
Opinion Mining (Analisis Sentimen) untuk Ekstraksi
Data Opini Publik pada Perguruan Tinggi, 6(1), 37-43.

Sarjana, P. S., Statistika, D., Matematika, F., Ilmu, D.
A. N., & Alam, P. (2017). ANALISIS SENTIMEN PENGGUNA
TWITTER JAKARTA DENGAN METODE NAÏVE BAYESIAN
CLASSIFICATION DAN SUPPORT VECTOR MACHINE JAKARTA
DENGAN METODE NAÏVE BAYESIAN MACHINE.

Sartika, D., Sensuse, D. I., Indo, U., Mandiri, G., &
Komputer, F. I. (2017). Perbandingan Algoritma
Klasifikasi Naive Bayes , Nearest Neighbour , dan
Decision Tree pada Studi Kasus Pengambilan Keputusan
Pemilihan Pola Pakaian. *Jurnal Teknik Informatika Dan
Sistem Informasi (JATISI)*, 1(2), 151-161.

Selvia Lorena Br Ginting, R. P. T. (2014). Penggunaan

Metode Naïve Bayes Classifier Pada Aplikasi
Perpustakaan. *Expert Systems with Applications*, 1-7.

Setyani, N. I., Sri Hastjarjo, & Nora Nailul Amal.
(2013). Penggunaan Media Sosial Sebagai Sarana
Komunikasi bagi Komunitas. *Komunikasi*, 1-17. Retrieved
from
[https://abdulkaharkimia.files.wordpress.com/2013/12/ju
rnal-novia-ika.pdf](https://abdulkaharkimia.files.wordpress.com/2013/12/jurnal-novia-ika.pdf)

Wayan, N., & Saraswati, S. (2013). NAÏVE BAYES CLASSIFIER
DAN SUPPORT VECTOR MACHINES UNTUK SENTIMENT ANALYSIS,
2-4.

Zega, S. A. (2014). Penggunaan Pohon Keputusan untuk
Klasifikasi Tingkat Kualitas Mahasiswa Berdasarkan
Jalur Masuk Kuliah, 7-13.

Zuhri, F. N., Alamsyah, A., Si, S., & Sc, M. (2017).
Menggunakan Naive Bayes Classifier Di Forum Kaskus
Public Sentiment Analysis of Smartfren Brand Using
Naive Bayes Classifier on Kaskus Forum, 4(1), 242-251.

LAMPIRAN

1. Data Set

1.1 Data Mentah

ID	username	Status	tanggal	retweet
1	tribunjogja	Sleman Tidak Tangani Pemulihan Infrastruktur Merapi http://bit.ly/fVfzJK	Thu Dec 30 13:00	
2	berita_rmol	Anak-anak di Lereng Merapi Butuh Recovery Pendidikan: Upaya pemulihan kondisi (recovery) korban erupsi Gunung Merapi	Wed Dec 29 17:00	
3	serunai61	Hey...have you ever tried, really reaching out to the other side... (Bread). Bantu SERUNAI utk pemulihan #Merapi-Mentawai	Tue Dec 28 17:00	
4	tweetDPR	[news] KOMISI IXDPR USULKAN PROGRAM KESEHATAN: Program pemulihan pasca erupsi merapi akan diajukan ke pro...	Mon Dec 27 09:00	
5	KLaproject	I posted 5 photos on Facebook in the album "Bantuan Pemulihan Ekonomi akibat Erupsi Merapi - 22 Des 2010" http://fb.me/vYdaYphb	Sat Dec 25 23:00	
6	serunai61	Compassion is a two way street.-Frank Capra. YUK ulurkan tangan utk pemulihan #Merapi-Mentawai. SERUNAI. http://bit.ly/fbFLqu	Sat Dec 25 10:00	
7	yoeliana	Yup kliatan Gagah&Tenang :) ,tggal pemulihan pasca bencana.. re: #Merapi RT @ratna_pramulia	Fri Dec 24 18:10	
8	kabartimlo	Pemulihan Lereng #Merapi, Pemkab Klaten Berencana Tanam Ribuan Pohon http://bit.ly/eV8v8w	Fri Dec 24 16:00	
9	serunai61	PAMERAN POSTER SERUNAI U PEMULIHAN #MERAPI via Twitter&FB http://serunaiku.blogspot.com/p/galeri-poster.html	Thu Dec 23 12:00	
10	idifvu	OkeZone - Aktual : Pemulihan Wisata Kaliurang Butuh Waktu 6 Bulan: Pascaerupsi Merapi, sektor pariwisata K...	Thu Dec 23 00:00	
11	amat_govind	KOMISI IXDPR USULKAN PROGRAM KESEHATAN: Program pemulihan pasca erupsi merapi akan diajukan ke program Re...	Mon Dec 27 09:00	
12	MerapiNews	RT @kabartimlo: Pemulihan Lereng #Merapi, Pemkab Klaten Berencana Tanam Ribuan Pohon http://bit.ly/eV8v8w	Fri Dec 24 16:50	
13	hardjo_pawiro	Pemulihan Wisata Kaliurang Butuh Waktu 6 Bulan: Pascaerupsi Merapi, sektor pariwisata Kaliurang, Sleman, Yogyakarta...	Wed Dec 22 23:00	
14	herrkusuma	RT @serunai61: DUKUNG TIMNAS CETAK GOL, DUKUNG JUGA PEMULIHAN #MERAPI-MENTAWAI. Support SERUNAI! http://www.serunaiku.blog	Wed Dec 22 21:00	
15	anyahave	via @jalinmerapi Mohon Dukungan Pemulihan Ekonomi, Fasilitas Pendidikan & t4 Ibadah Klik http://bit.ly/fyilQI #merap	Wed Dec 22 18:00	
16	jalinmerapi	Mohon Dukungan: Pemulihan Ekonomi, Fasilitas Pendidikan & Tempat Ibadah Klik http://bit.ly/fyilQI #merapi [Kec Nglu	Wed Dec 22 17:00	
17	serunai61	SUPPORT TIMNAS KITA, DUKUNG JUGA PEMULIHAN #MERAPI-MENTAWAI. Support SERUNAI! http://www.serunaiku.blog	Wed Dec 22 16:00	
18	hardjo61	There is no exercise better for the heart than reaching down and lifting people up."John Andrew Holmes.SERUNAI U PEN	Wed Dec 22 09:00	
19	polindonetwo	Pertamina Peduli, Bantu Pemulihan Korban Merapi http://fb.me/vYdaYphb	Tue Dec 21 16:00	
20	tribunnews	Pertamina Bantu Pemulihan Korban Letusan Merapi http://bit.ly/fbFLqu	Tue Dec 21 10:00	
21	dinadiinnaa	@AbiChino ttg merapi dr mulai erupsi sampai dampak sama pemulihan yg terjadi setelah erupsi ketoke	Tue Dec 21 00:00	
22	VHRmedia	Pemulihan hutan #Merapi butuh waktu 50 tahun. http://bit.ly/ihuk4f	Mon Dec 20 11:00	
23	marilest	Mohon doanya ya.. RT @marilest [RT @jalinmerapi] Tanggap Darurat blm usai. Dukung pemulihan kehidupan warga lere	Mon Dec 20 08:00	
24	jalinmerapi	Tanggap Darurat blm usai. Dukung pemulihan kehidupan warga lereng #Merapi & barantan #Kali. Cek situasi sosial di htt	Mon Dec 20 07:00	
25	maryam_karpo	Body tak mau kompromi, sementara sabtu minggu ini batal ikut team relawan pemulihan Merapi. Buat tmn2 kntor, selan	Sat Dec 18 06:00	

1.2 Data Bersih (Stopword Removal + Pelabelan)

	A	C	D	E	F	G	H	I
1	text	class						
2	sleman tangan pemulihan infrastruktur merapi	positif						
3	anak anak lereng merapi butuh recovery pendidikan upaya pemulihan kondisi recovery korban erupsi gunung	negatif						
4	hey ever tried really reaching side bread bantu serunai utk pemulihan merapi tawa	positif						
5	news komisi ixdpr usul program sehat program pemulihan pasca erupsi merapi aju pro	positif						
6	posted photos facebook album bantu pemulihan ekonomi akibat erupsi merapi des	positif						
7	compassion two way street frank capra yuk ulur tangan utk pemulihan merapi tawa serunai	positif						
8	yup kliatan gagah tenang tggal pemulihan pasca bencana re merapi rt ratna pramulia	positif						
9	pemulihan lereng merapi Pemkab Klaten rencana tanam ribu pohon	netral						
10	pamer poster serunai u pemulihan merapi via twitter fb	positif						
11	okezone aktual pemulihan wisata kaliurang butuh pascaerupsi merapi sektor pariwisata k	netral						
12	komisi ixdpr usul program sehat program pemulihan pasca erupsi merapi aju program re	positif						
13	rt kabartimlo pemulihan lereng merapi Pemkab Klaten rencana tanam ribu pohon	netral						
14	pemulihan wisata kaliurang butuh pascaerupsi merapi sektor pariwisata kaliurang sleman yogyakarta	netral						
15	rt serunai dukung timnas cetak gol dukung pemulihan merapi tawa support serunai	positif						
16	via jalinmerapi mohon dukung pemulihan ekonomi fasilitas pendidikan t ibadah klik merapi kec ngluwar magelang	negatif						
17	mohon dukung pemulihan ekonomi fasilitas pendidikan ibadah klik merapi kec ngluwar magelang	negatif						
18	support timnas dukung pemulihan merapi tawa support serunai fb serunai nusantara	positif						
19	exercise better heart reaching lifting people john andrew holmes serunai u pemulihan merapi tawa	positif						
20	pertamina peduli bantu pemulihan korban merapi	positif						
21	pertamina bantu pemulihan korban letus merapi	positif						
22	abichino ttg merapi dr erupsi dampak pemulihan yg erupsi toke	netral						
23	pemulihan hutan merapi butuh	netral						

2. Stopword Indonesia

ada	dimisalkan	maupun	seharusnya
adalah	dimulai	melainkan	sehingga
adanya	dimulailah	melakukan	seingat
adapun	dimulainya	melalui	sejak
agak	dimungkinkan	melihat	sejauh
agaknya	dini	melihatnya	sejenak
agar	dipastikan	memang	sejumlah
akan	diperbuat	memastikan	sekadar
akankah	diperbuatnya	memberi	sekadarnya
akhir	dipergunakan	memberikan	sekali
akhiri	diperkirakan	membuat	sekali-kali
akhirnya	diperlihatkan	memerlukan	sekalian
aku	diperlukan	memihak	sekaligus
akulah	diperlukannya	meminta	sekalipun
amat	dipersoalkan	memintakan	sekarang
amatlah	dipertanyakan	memisalkan	sekecil
anda	dipunyai	memperbuat	seketika
andalah	diri	mempergunakan	sekiranya
antar	dirinya	memperkirakan	sekitar
antara	disampaikan	memperlihatkan	sekitarnya
antaranya	disebut	mempersiapkan	sekurang-kurangnya
apa	disebutkan	mempersoalkan	sekurangnya
apaan	disebutkannya	mempertanyakan	sela
apabila	disini	mempunyai	selagi
apakah	disinilah	memulai	selain
apalagi	ditambahkan	memungkinkan	selaku
apatah	ditandaskan	menaiki	selalu
artinya	ditanya	menambahkan	selama
asal	ditanyai	menandaskan	selama-lamanya
asalkan	ditanyakan	menanti	selamanya
atas	ditegaskan	menanti-nanti	selanjutnya
atau	ditujukan	menantikan	seluruh
ataukah	ditunjuk	menanya	seluruhnya
ataupun	ditunjuki	menanyai	semacam
awal	ditunjukkan	menanyakan	semakin
awalnya	ditunjukkannya	mendapat	semampu
bagai	ditunjuknya	mendapatkan	semampunya
bagaikan	dituturkan	mendatang	semasa
bagaimana	dituturkannya	mendatangi	semasih
bagaimanakah	diucapkan	mendatangkan	semata

bagaimanapun	diucapkannya	menegaskan	semata-mata
bagi	diungkapkan	mengakhiri	semaunya
bagian	dong	mengapa	sementara
bahkan	dua	mengatakan	semisal
bahwa	dulu	mengatakannya	semisalnya
bahwasanya	empat	mengenai	sempat
baik	enggak	mengerjakan	semua
bakal	enggaknya	mengetahui	semuanya
bakalan	entah	menggunakan	semula
balik	entahlah	menghendaki	sendiri
banyak	guna	mengibaratkan	sendirian
bapak	gunakan	mengibaratkannya	sendirinya
baru	hal	mengingat	seolah
bawah	hampir	mengingatkan	seolah-olah
beberapa	hanya	menginginkan	seorang
begini	hanyalah	mengira	sepanjang
beginian	hari	mengucapkan	sepantasnya
beginikah	harus	mengucapkannya	sepantasnyalah
beginilah	haruslah	mengungkapkan	seperlunya
begitu	harusnya	menjadi	seperti
begitukah	hendak	menjawab	sepertinya
begitulah	hendaklah	menjelaskan	sepihak
begitupun	hendaknya	menuju	sering
bekerja	hingga	menunjuk	seringnya
belakang	ia	menunjuki	serta
belakangan	ialah	menunjukkan	serupa
belum	ibarat	menunjuknya	sesaat
belumkah	ibaratkan	menurut	sesama
benar	ibaratnya	menuturkan	sesampai
benarkah	ibu	menyampaikan	sesegera
benarlah	ikut	menyangkut	sesekali
berada	ingat	menyatakan	seseorang
berakhir	ingat-ingat	menyebutkan	sesuatu
berakhirlah	ingin	menyeluruh	sesuatunya
berakhirnya	inginkah	menyiapkan	sesudah
berapa	inginkan	merasa	sesudahnya
berapakah	ini	mereka	setelah
berapalah	inikah	merekalah	setempat
berapapun	inilah	merupakan	setengah
berarti	itu	meski	seterusnya
berawal	itukah	meskipun	setiap

3. Kutipan Kamus SensiStrength

Kata	Bobot
abadi	5
absen	-3
abu-abu	-1
acuh	4
adil	5
aduh	-2
agresi	-5
aib	-5
ajaib	3
ajar	4
akal	4
alasan	-2
alibi	-4
alien	-1
aman	4
ambigu	-3
ambisi	-2
amoral	-5
anarki	-5
anarkis	-5
ancaman	-5

aneh	-2
anggun	4
angkuh	-5
anjing	-4
anomali	-3
antagoni	-4
anti-sosial	-4
antipati	-5
apes	-4
api	-1

4. Kode Cleansing

```

setwd("D:/")
mydata <- read.csv(file.choose(), header = T)
class(mydata)
library("NLP")
library("tm")
df <- do.call("rbind", lapply(mydata$Status,
as.data.frame))
df <- sapply(df, function(row) iconv(row, "latin1",
"ASCII", sub=""))
#menghapus url
df=gsub("http[[:alnum:]][:punct:]]*", "",df)

#menghapus url
df=gsub("www[[:alnum:]][:punct:]]*", "",df)

```

```
#menghapus angka kombinasi huruf
df=gsub("[^0-9A-Za-z///' ]", " ",df)

#menghapus angka
df=gsub("[0-9]", " ",df)

df=gsub("[^[:alnum:]]", " ",df)

df <- stripWhitespace(df)
df <- tolower(df)

write.table(df,
file="cleansing.csv",sep=" ",row.names=F)
getwd()
```

5. Kode Stopword Removal

```
#Stopwords Custom

#import library text mining
library(NLP)
library(tm)

#read data custom stopword
Custom_stopword <- read.csv(file.choose(), header
= FALSE)

#mengubah menjadi karakter
```

```

stopword_custom <-
as.character(Custom_stopword$V1)

#mengganti stopwords bawaan dengan custom stopwords
stopword_terbaru <- c(stopword_custom,
stopwords())

#read data training
hasil <- read.csv(file.choose(), header = T)

#mengubah menjadi vektor
corpus <- iconv(hasil$X..i.., to = "UTF-8")
corpus <- Corpus(VectorSource(corpus))

#eksekusi stopwords
cleanset <- tm_map(corpus, removeWords,
stopword_terbaru)
cleanset <- tm_map(cleanset, stripWhitespace)

#ubah ke data frame
dataframe <- data.frame(text=sapply(cleanset,
identity),
stringsAsFactors=F)

#save ke csv
write.table(dataframe,
file="stopword1.csv", sep="," , row.names=F)

```

6. Kode SentiStrength

```
import re
from collections import OrderedDict
import numpy as np

class sentistrength:
    def __init__(self, config=dict()):
        self.negasi = [line.replace('\n','') for
line
in
open("negatingword.txt").read().splitlines()]
        self.tanya = [line.replace('\n','') for
line
in
open("questionword.txt").read().splitlines()]
        #create sentiment words dictionary
        self.sentiwords_txt =
[line.replace('\n','').split(":") for line in
open("sentiwords_id.txt").read().splitlines()]
        self.sentiwords_dict = OrderedDict()
        for term in self.sentiwords_txt:
            self.sentiwords_dict[term[0]] =
int(term[1])
        #create emoticon dictionary
        self.emoticon_txt =
[line.replace('\n','').split(" | ") for line in
open("emoticon_id.txt").read().splitlines()]
        self.emoticon_dict = OrderedDict()
```

```

        for term in self.emoticon_txt:
            self.emoticon_dict[term[0]] =
int(term[1])
        #create idioms dictionary
        self.idioms_txt =
[line.replace('\n','').split(":") for line in
open("idioms_id.txt").read().splitlines()]
        self.idioms_dict = OrderedDict()
        for term in self.idioms_txt:
            self.idioms_dict[term[0]] =
int(term[1])
        #create boosterwords dictionary
        self.boosterwords_txt =
[line.replace('\n','').split(":") for line in
open("boosterwords_id.txt").read().splitlines()]
        self.boosterwords_dict = OrderedDict()
        for term in self.boosterwords_txt:
            self.boosterwords_dict[term[0]] =
int(term[1])
        self.negation_conf = config["negation"]
        self.booster_conf = config["booster"]
        self.ungkapan_conf = config["ungkapan"]
        self.consecutive_conf =
config["consecutive"]
        self.repeated_conf = config["repeated"]
        self.emoticon_conf = config["emoticon"]
        self.question_conf = config["question"]
        self.exclamation_conf =
config["exclamation"]

```



```
self.punctuation_conf =
config["punctuation"]
self.mean_conf = False

def senti(self, term):
    try:
        return self.sentiwords_dict[term]
    except:
        return 0

def emosikon(self, term):
    try:
        return self.emoticon_dict[term]
    except:
        return 0

def ungkapan(self, term):
    try:
        return self.idioms_dict[term]
    except:
        return 0

def booster(self, term):
    try:
        return self.boosterwords_dict[term]
    except:
        return 0
```

```

def cek_negationword(self, prev_term,
prev_term2):
    #jika kata sebelumnya (index-1) adalah kata
    negasi, negasikan nilai -+nya
    if prev_term in self.negasi or prev_term2+"
    "+prev_term in self.negasi:
        # print prev_term
        self.score = -abs(self.score) if
self.score>0 else abs(self.score)

    def cek_boosterword(self,term):
        booster_score = self.booster(term)
        if booster_score !=0 and self.score>0:
self.score += booster_score
        if booster_score !=0 and self.score<0:
self.score -= booster_score

    def cek_consecutive_term(self, prev_term):
        if self.prev_score>0 and self.score >=3:
self.score+=1
        if self.prev_score<0 and self.score <=-3:
self.score-=1

    def cek_ungkapan(self, bigram,trigram, i):
        bigram = ' '.join(bigram)
        trigram = ' '.join(trigram)
        ungkapan_score = self.ungkapan(bigram)
        if ungkapan_score==0:

```

```

        ungkapan_score =
self.ungkapan(trigram)
        if ungkapan_score!=0:
            self.score = ungkapan_score
            self.prev_score = 0
            self.pre_max_pos[i-1] = 1
            self.pre_max_neg[i-1] = -1
            self.max_pos = self.pre_max_pos[i-2]
            #if len(self.pre_max_pos)>1 else 1
            self.max_neg = self.pre_max_neg[i-2]
            #if len(self.pre_max_neg)>1 else -1
            self.sentence_score[i-1] =
re.sub(r'\[\d\]', '', self.sentence_score[i-1])

        def cek_repeated_punctuation(self, next_term):
            if re.search(r'!\{2,\}', next_term) and
self.score >=3: self.score+=1
            if re.search(r'!\{2,\}', next_term) and
self.score <=-3: self.score-=1

        def remove_extra_repeated_char(self, term):
            return re.sub(r'([A-Za-
z])\1{2,\}', r'\1', term)
        def plural_to_singular(self, term):
            return re.sub(r'([A-Za-z]+)\-\1',
r'\1', term)
        def classify(self):
            result = "neutral"
            try:

```

```

        if self.mean_conf:
            mean_p = np.mean(self.mean_pos)
            mean_n = np.mean(self.mean_neg)
            print (mean_p, mean_n)
            if mean_p > mean_n:
                result = "positive"
            elif mean_p < mean_n and not
self.is_tanya:
                result = "negative"
            elif mean_p < mean_n and
self.is_tanya:
                result = "neutral"
            else:
                if abs(self.sentences_max_pos) >
abs(self.sentences_max_neg):
                    result = "positive"
                elif abs(self.sentences_max_pos) <
abs(self.sentences_max_neg):
                    result = "negative"
                elif abs(self.sentences_max_pos)
== abs(self.sentences_max_neg):
                    result = "neutral"
        except:
            print("error ",self.sentences_max_pos,
self.sentences_max_neg)
            return result
    def cek_neutral_term(self,terms,i):
        if terms[i-1] in self.neutral_term or
terms[i+1] in self.neutral_term: self.score=1

```

```

def main(self,sentence):
    self.neutral_term = ['jika','kalau']
    sentences = sentence.split('.')
    self.sentences_max_neg = -1
    self.sentences_max_pos = 1
    self.sentences_score = []
    self.sentences_text = []
    for sentence in sentences:
        self.max_neg = -1
        self.max_pos = 1
        self.mean_neg = [1]
        self.mean_pos = [1]
        self.sentence_score=[]
        terms = sentence.split()
        # terms = re.split(r'[\s,.]',sentence)
        terms_length = len(terms)
        self.is_tanya = False
        self.sentence_text = ''
        # print self.max_pos, self.max_neg
        #SEMUA KALIMAT YANG MEMILIKI TANDA SERU
MEMILIKI +ve minimal 2
        if self.exclamation_conf and
re.search('!',sentence): self.max_pos = 2
        self.prev_score = 0
        self.pre_max_pos = []
        self.pre_max_neg = []
        for i,term in enumerate(terms):
            # repeated_term = ''

```

```

        is_extra_char = False
        plural = ''
        self.score = 0
        # if re.search(r'[A-Za-z\-.
        .]+',term):
            # print term
            if re.search(r'([A-Za-
            z])\1{3,}',term):
                is_extra_char = True
                # repeated_term =term
                term =
                self.remove_extra_repeated_char(term)
                if re.search(r'([A-Za-z]+)\-
                \1',term):
                    plural = term
                    term =
                self.plural_to_singular(term)
                #GET SENTI SCORE#
                self.score = self.senti(term)
                # print "senti score",term,
                self.score

                #NEGATION HANDLER#
                if self.negation_conf and
                self.score !=0 and
                i>0:self.cek_negationword(terms[i-1],terms[i-2])
                # print "negation score",term,
                self.score

```

```

#BOOSTERWORD HANDLER#
        if self.booster_conf and
self.score !=0 and i>0 and i<=(terms_length-
1):self.cek_boosterword(terms[i-1])
        if self.booster_conf and
self.score !=0 and i>=0 and i<(terms_length-
1):self.cek_boosterword(terms[i+1])
        # print "booster score",term,
self.score

#IDIOM/UNGKAPAN HANDLER#
        if self.ungkapan_conf and i>0 and
i<=(terms_length-1):self.cek_ungkapan([terms[i-
1],term],[terms[i-2],terms[i-1],term],i)
        # if self.ungkapan_conf and i>=0
and i<(terms_length-
1):self.cek_ungkapan([term,terms[i+1]])
        # print "idiom score",term,
self.score

#CONSECUTIVE SENTIMENT WORD#
        if self.consecutive_conf and i>0
and i<=(terms_length-1) and self.score
!=0:self.cek_consecutive_term(terms[i-1])
        # print "consecutive score",term,
self.score

# +1 SENTI SCORE IF REPEATED CHAR ON
POSITIVE/NEGATIVE +2 IF NEUTRAL TERM

```

```

        if self.repeated_conf and
is_extra_char==True and self.score>0:
self.score+=1
        if self.repeated_conf and
is_extra_char==True and self.score<0: self.score-
=1
        if self.repeated_conf and
is_extra_char==True and self.score==0:
self.score=2
        # print "repeat char score", term,
self.score
        if self.punctuation_conf and i>=0
and i<(terms_length-1):
self.cek_repeated_punctuation(terms[i+1])
        # CEK APAKAH TERDAPAT KATA TANYA
        if self.question_conf and (term in
self.tanya or re.search(r'\?',term)):self.is_tanya
= True
        # CEK neutral term
        if self.score!=0 and i>1 and
i<(terms_length-2): self.cek_neutral_term(terms,i)
        # if self.score!=0 and i>0 and
i<(terms_length-4): self.cek_neutral_term(terms,i)
        if self.emoticon_conf and
self.score==0: self.score = self.emosikon(term)

        self.prev_score = self.score
        if self.mean_conf and
self.score>0: self.mean_pos.append(self.score)

```



```

        if self.mean_conf and
self.score<0:
self.mean_neg.append(abs(self.score))
        #GET MAX SCORE +ve/-ve
        self.max_pos= self.score if
self.score > self.max_pos else self.max_pos
        self.max_neg= self.score if
self.score < self.max_neg else self.max_neg
        #insert score info current term

self.pre_max_pos.append(self.max_pos)

self.pre_max_neg.append(self.max_neg)

        # print self.pre_max_pos,
self.pre_max_neg

        if plural !='': term = plural
self.sentence_text += '
{}'.format(term)

        if self.score != 0:term = "{}
[{}]" .format(term, self.score)
self.sentence_score.append(term)

self.sentences_text.append(self.sentence_text)
self.sentences_score.append("
".join(self.sentence_score))
        if self.is_tanya:
            self.max_neg = -1

```

```

        self.sentences_max_pos = self.max_pos
    if self.max_pos > self.sentences_max_pos else
    self.sentences_max_pos
        self.sentences_max_neg = self.max_neg
    if self.max_neg < self.sentences_max_neg else
    self.sentences_max_neg
        # print self.sentences_max_pos,
    self.sentences_max_neg
        sentence_result = self.classify()
        # print self.sentences_text
        #return {"classified_text": ".
".join(self.sentences_score),"RESULT
":self.sentences_max_pos+self.sentences_max_neg
,"tweet_text": ".
".join(self.sentences_text),"sentence_score":self
.sentences_score,"max_positive":self.sentences_ma
x_pos,"max_negative":self.sentences_max_neg,"kela
s":sentence_result}

        return {"RESULT
":
self.sentences_max_pos + self.sentences_max_neg,
        "max_positive":
self.sentences_max_pos,
        "max_negative":
self.sentences_max_neg,
        "kelas": sentence_result}
        # return {"text
":
".join(self.sentence_score)
,"sentimen":
sentence_result}

```

```

        # return{sentence_result}
config = dict()
config["negation"] = True
config["booster"] = True
config["ungkapan"] = True
config["consecutive"] = True
config["repeated"] = True
config["emoticon"] = True
config["question"] = True
config["exclamation"] = True
config["punctuation"] = True
senti = sentistrength(config)

list1= ['agnezmo22222 malas dan jelek sekali tetapi
lintah darat :)', 'Maafkan aku Ham, aku udah bener-
bener gak bisa sama kamu, aku tuh udah terlanjur
mencintai dia, bahkan lebih dari cinta aku ke kamu,
maaf', 'Aku benar-benar mencintaimu tapi tidak suka
adik dingin Anda'];
list2=[line.strip() for line in
open("D:/testhasil.txt", 'r')]; #E:/00.
SKRIPSIIIIIIIIIIIIIIIIIIIIII
FIIXXXXXXXXXXX/program/DATA/novdsss1.txt
#print senti.main("agnezmo malas dan jelek sekali
tetapi lintah darat :)")
#print senti.main("Maafkan aku Ham, aku udah bener-
bener gak bisa sama kamu, aku tuh udah terlanjur
mencintai dia, bahkan lebih dari cinta aku ke kamu,
maaf ")

```

```

#print senti.main("Aku benar-benar mencintaimu
tapi tidak suka adik dingin Anda.")
for p in list2: print(senti.main(p))

#print senti.main(list1)

```

7. Hasil SentiStrength

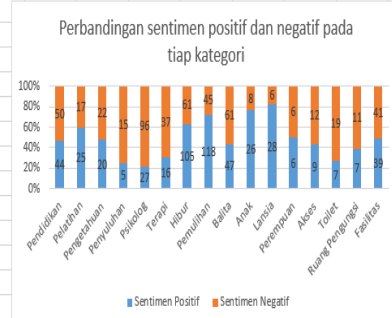
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	Result	Max Posit	Max Nega	Class	Kebutuhan	Kategori															
2	0	1	-1	neutral	psikologi	pemulihan															
3	0	1	-1	neutral	psikologi	pemulihan															
4	-2	1	-3	negative	psikologi	pemulihan															
5	1	2	-1	positive	psikologi	pemulihan															
6	3	4	-1	positive	psikologi	pemulihan															
7	0	1	-1	neutral	psikologi	pemulihan															
8	0	1	-1	neutral	psikologi	pemulihan															
9	3	4	-1	positive	psikologi	pemulihan															
10	0	1	-1	neutral	psikologi	pemulihan															
11	-3	1	-4	negative	psikologi	pemulihan															
12	0	1	-1	neutral	psikologi	pemulihan															
13	3	4	-1	positive	psikologi	pemulihan															
14	0	1	-1	neutral	psikologi	pemulihan															
15	0	1	-1	neutral	psikologi	pemulihan															
16	2	3	-1	positive	psikologi	pemulihan															
17	4	5	-1	positive	psikologi	pemulihan															
18	4	5	-1	positive	psikologi	pemulihan															
19	2	3	-1	positive	psikologi	pemulihan															
20	0	1	-1	neutral	psikologi	pemulihan															
21	1	4	-3	positive	psikologi	pemulihan															
22	-2	1	-3	negative	psikologi	pemulihan															
23	0	1	-1	neutral	psikologi	pemulihan															

8. Data Kebutuhan Khusus tiap Kategori

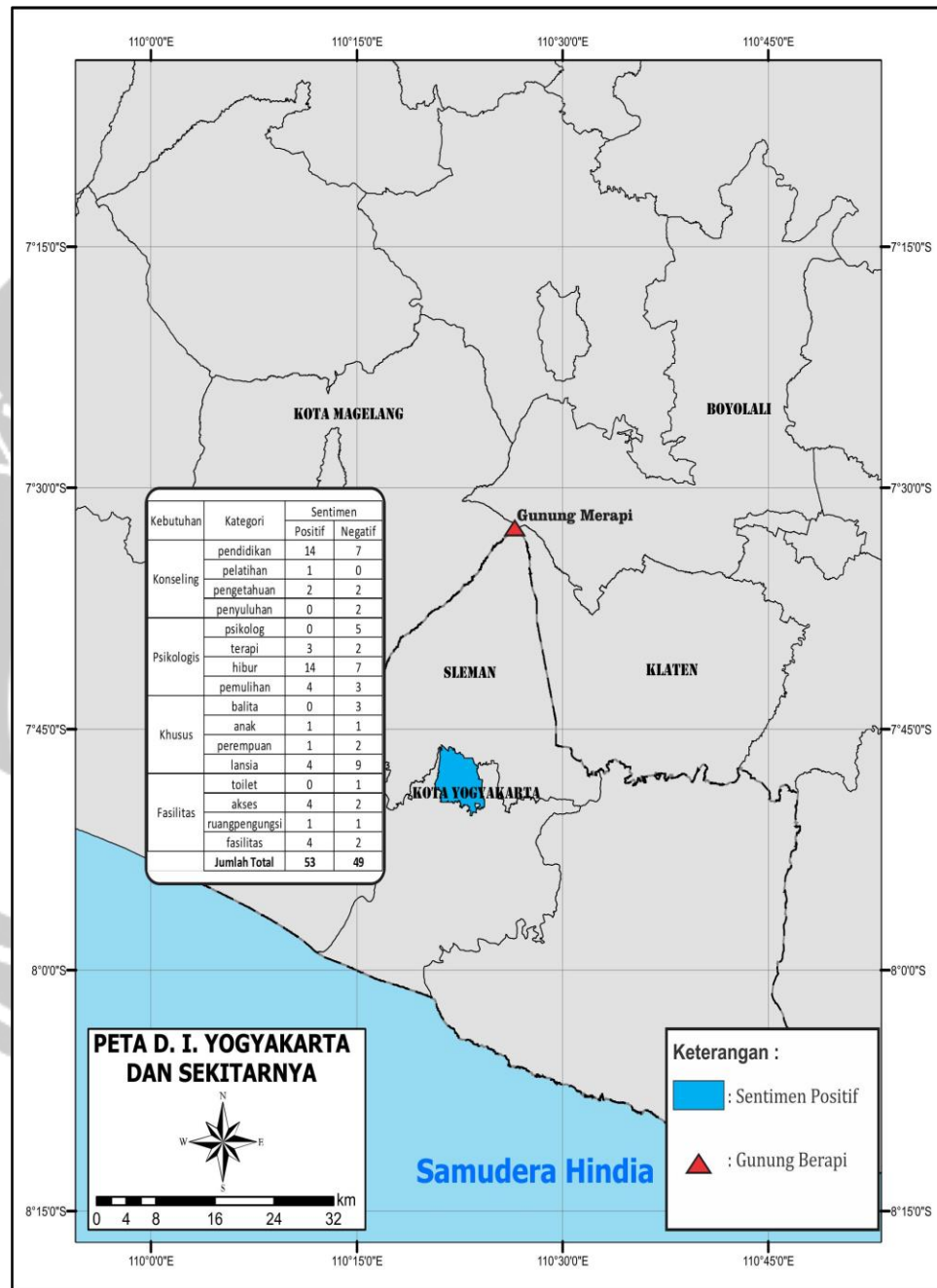
kebutuhan	kategori	Kabupaten					
		Sentimen	Yogyakarta	Bojolali	Magelang	Klaten	Sleman
Konseling	pendidikan	positif	5	0	0	3	2
		negatif	7	1	2	0	3
	pelatihan	positif	1	1	1	0	1
		negatif	0	0	0	0	0
	pengetahuan	positif	1	0	1	0	0
		negatif	0	0	0	0	0
penyuluhan	positif	0	0	0	0	0	
	negatif	2	0	0	0	0	
Psikologis	psikolog	positif	0	0	0	2	0
		negatif	5	0	0	0	1
	terapi	positif	1	0	0	2	1
		negatif	1	0	0	0	0
	hibur	positif	14	0	1	0	1
		negatif	7	0	0	1	0
	pemulihan	positif	4	1	3	3	1
		negatif	2	0	3	0	2
Khusus	balita	positif	0	0	1	0	0
		negatif	1	0	7	2	0
	anak	positif	1	0	0	0	0
		negatif	1	0	0	0	0
	perempuan	positif	1	0	0	1	2
		negatif	1	0	0	0	0

9. Data Kebutuhan Khusus tiap lokasi

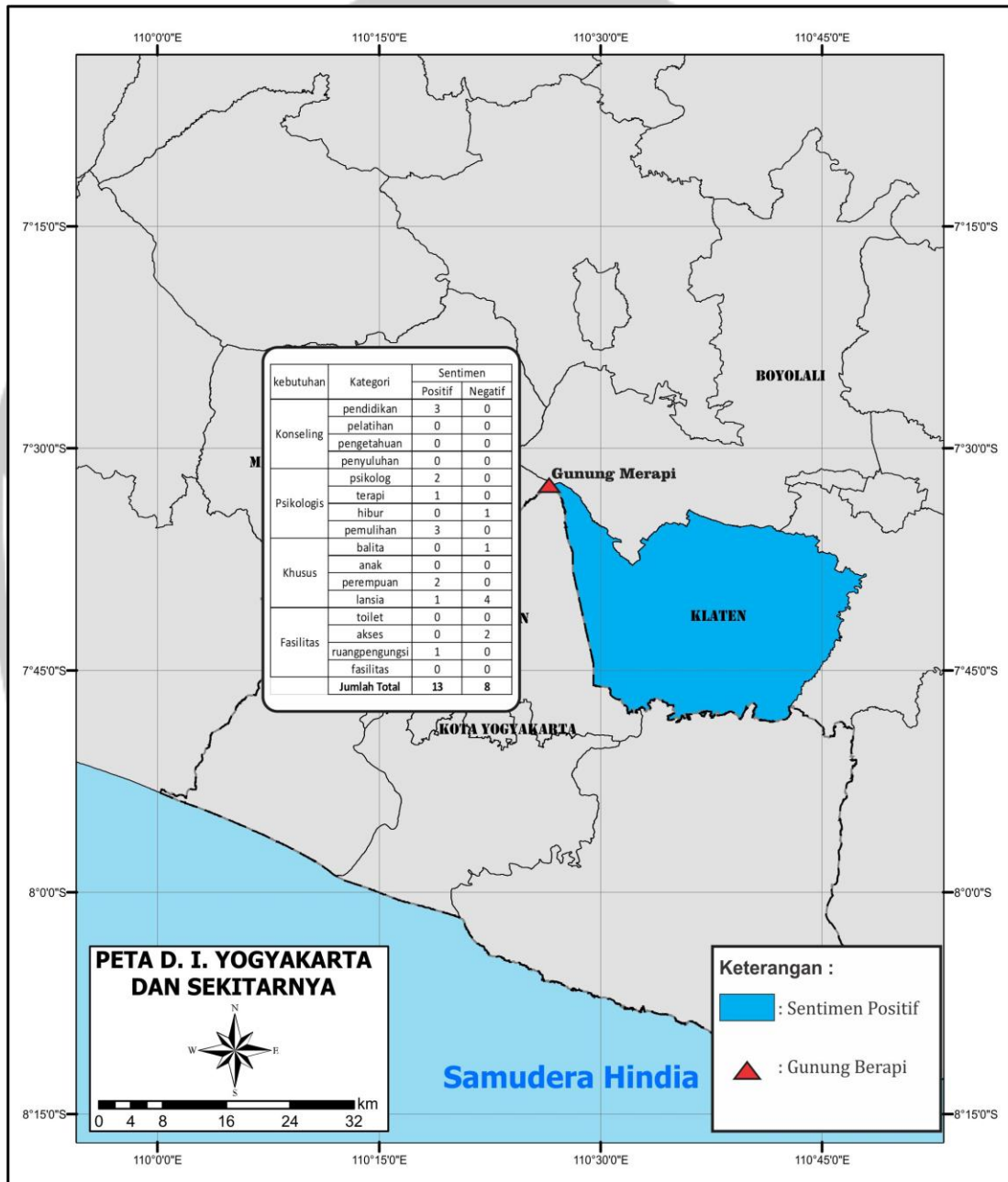
Kategori	Sentimen		Jumlah Tweet	Jumlah Kemunculan	
	Positif	Negatif		Positif	Negatif
Pendidikan	44	50	208	8.678501	9.861933
Pelatihan	25	17	83	4.930966	3.353057
Pengetahuan	20	22	103	3.944773	4.33925
Penyuluhan	5	15	26	0.986193	2.95858
Psikolog	27	96	179	5.325444	18.93491
Terapi	16	37	172	3.155819	7.29783
Hibur	105	61	412	20.71006	12.03156
Pemulihan	118	45	263	23.27416	8.87574
Balita	47	61	155	9.270217	12.03156
Anak	26	8	68	5.128205	1.577909
Lansia	28	6	75	5.522682	1.183432
Perempuan	6	6	24	1.183432	1.183432
Akses	9	12	34	1.775148	2.366864
Toilet	7	19	64	1.380671	3.747535
Ruang Pengungsi	7	11	34	1.380671	2.169625
Fasilitas	39	41	132	7.692308	8.086785
Total		529	507		



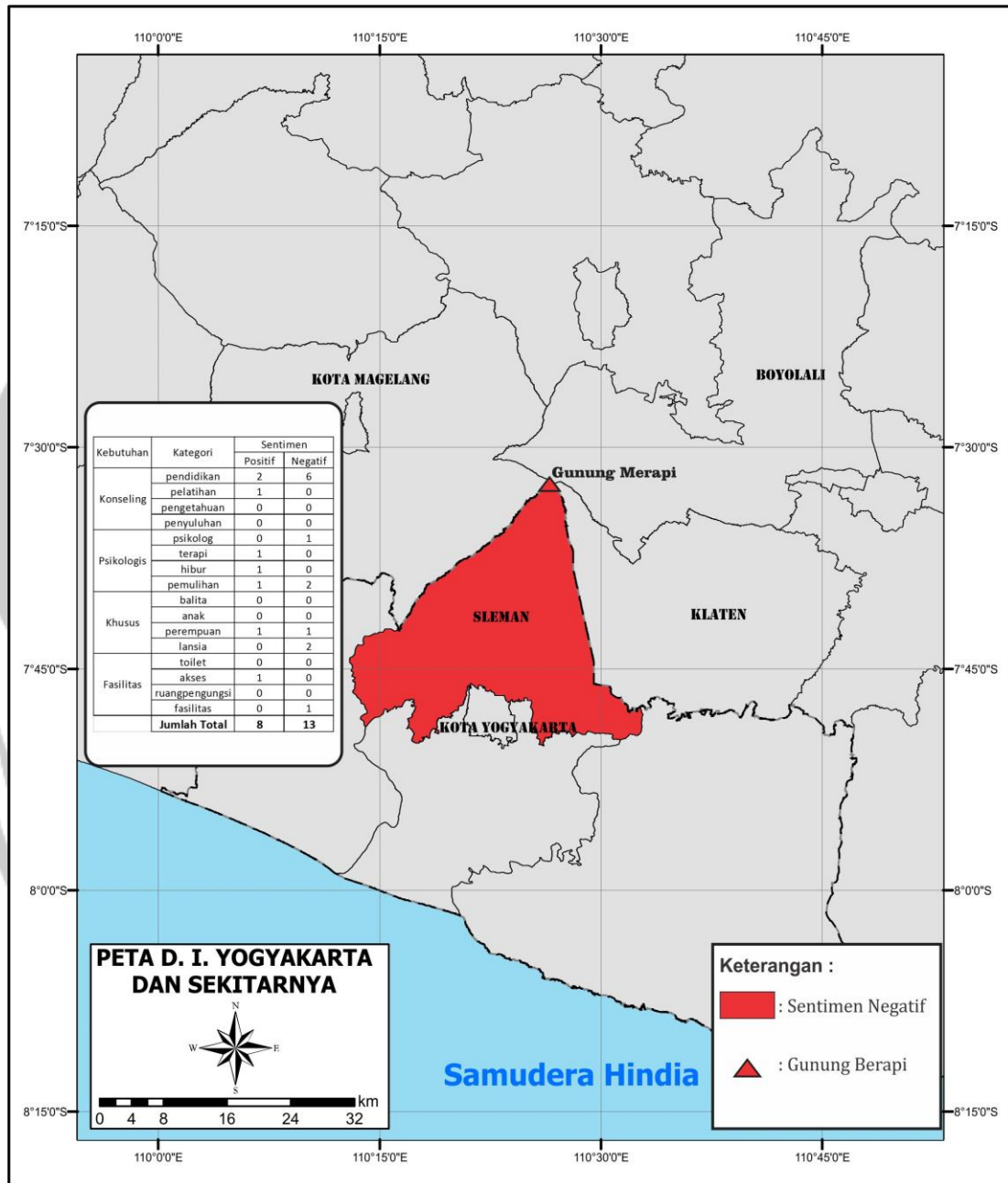
10. Kabupaten Yogyakarta



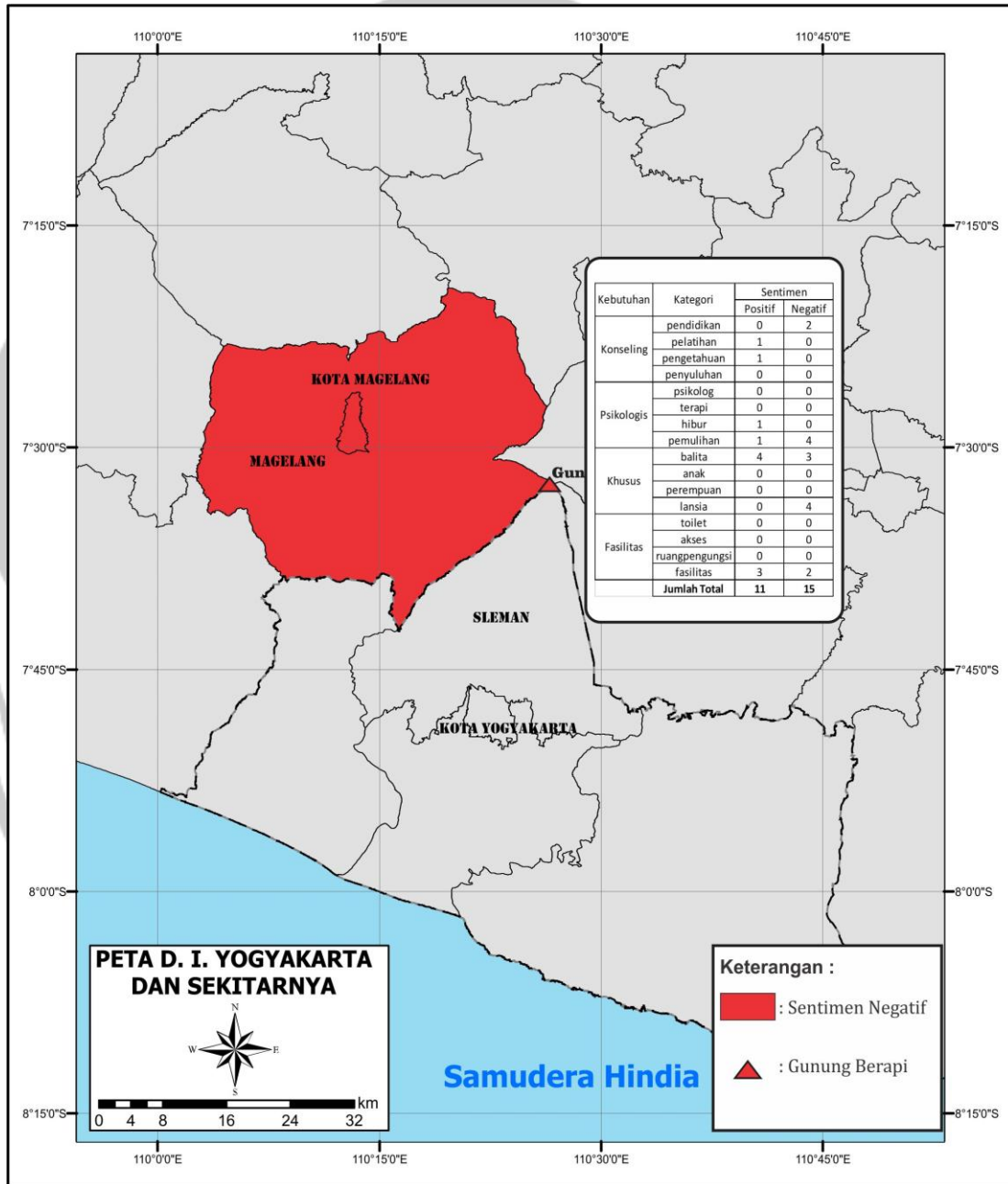
11. Kabupaten Klaten



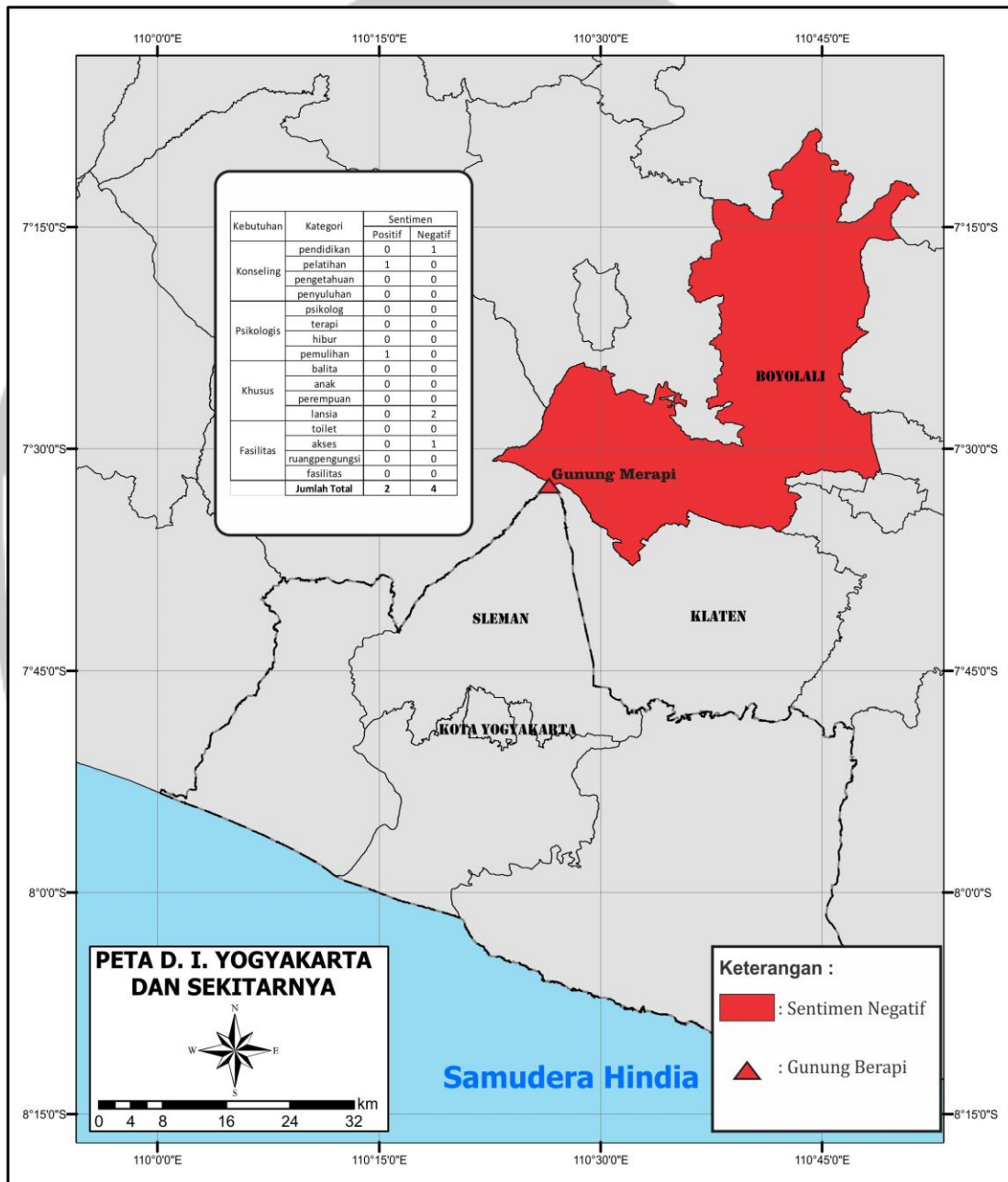
12. Kabupaten Sleman



13. Kabupaten Magelang



14. Kabupaten Boyolali



15. Hasil akurasi metode Naïve Bayes Classifier menggunakan 10 fold cross validation

```

=== Summary ===

Correctly Classified Instances      1401          97.2241 %
Incorrectly Classified Instances    40            2.7759 %
Kappa statistic                    0.9561
Mean absolute error                 0.0212
Root mean squared error            0.1337
Relative absolute error             5.0779 %
Root relative squared error        29.3004 %
Total Number of Instances          1441

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      1.000    0.034    0.868     1.000   0.929     0.916   0.996    0.985    positive
      0.948    0.000    1.000     0.948   0.974     0.952   0.991    0.992    neutral
      0.990    0.000    1.000     0.990   0.995     0.992   0.999    0.999    negative
Weighted Avg.  0.972    0.006    0.976     0.972   0.973     0.959   0.995    0.993

=== Confusion Matrix ===
 a  b  c  <-- classified as
263  0  0 | a = positive
35 644  0 | b = neutral
5   0 494 | c = negative

```

16. Hasil akurasi metode Support Vector Machine menggunakan 10 fold cross validation

```

=== Summary ===

Correctly Classified Instances      1279          88.7578 %
Incorrectly Classified Instances    162          11.2422 %
Kappa statistic                    0.8173
Mean absolute error                 0.2521
Root mean squared error            0.3225
Relative absolute error            60.5287 %
Root relative squared error        70.6654 %
Total Number of Instances          1441

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      0.890    0.019    0.914     0.890   0.902     0.880   0.989    0.910    positive
      0.953    0.171    0.833     0.953   0.889     0.783   0.872    0.816    neutral
      0.798    0.011    0.975     0.798   0.878     0.831   0.921    0.863    negative
Weighted Avg.  0.888    0.087    0.897     0.888   0.887     0.818   0.910    0.849

=== Confusion Matrix ===
 a  b  c  <-- classified as
234  29  0 | a = positive
22 647  10 | b = neutral
0  101 398 | c = negative

```