

SISTEM INFORMASI MONITORING BENCANA ALAM MENGUNAKAN DATA MEDIA SOSIAL DENGAN ALGORITMA NAÏVE BAYES

NATURAL DISASTER MONITORING INFORMATION SYSTEM FROM SOCIAL MEDIA DATA USING NAÏVE BAYES ALGORITHM

Eryesa Ananda Tasya¹, Randy Erfa Saputra², Casie Setianingsih³

^{1,2,3} Universitas Telkom, Bandung

erhyesatasya@student.telkomuniversity.ac.id¹, resaputra@telkomuniversity.ac.id²,
setiacasie@telkomuniversity.ac.id³

Abstrak

Bencana alam yang telah terjadi di Indonesia cukup banyak, diantaranya gempa bumi, tsunami, longsor, banjir, dan lain-lain. Penyebab terjadinya potensi kejadian bencana alam ini karena Indonesia terletak di pertemuan lempeng – lempeng Eurasia, Pasifik, dan Indo-Australia.

Informasi yang di dapat pada social media sangat cepat berkembang dan lebih efektif. Saat terjadinya bencana seperti banjir, social media bekerja untuk menginfokan kepada pengguna dimana terjadinya bencana tersebut. Dalam makalah ini kami menggunakan Twitter sebagai tempat pencarian data. Twitter telah berhasil digunakan sebagai layanan darurat untuk menginformasikan kepada publik tentang perkembangan saat ini. Untuk mendapatkan informasi kita bisa mencari dengan hashtag yang relevan agar bisa mengetahui dimana kejadian tersebut.

Hasil dari pengujian ini akan ditampilkannya peta wilayah Indonesia dan titik terjadinya bencana diambil berdasarkan geolocation pada data tweet. Proses klasifikasi akan menggunakan metode Naïve Bayes. Proses pengelompokan ini dilakukan di masing-masing wilayah yang ada di Indonesia dengan realtime. Pada penelitian ini mendapatkan nilai akurasi sebesar 75% berdasarkan nilai *fold 3* pada pengujian *kfold cross validation*.

Kata Kunci: *Bencana Alam, Twitter, Naïve Bayes.*

Abstract

There are many natural disasters that have occurred in Indonesia, including earthquakes, tsunamis, landslides, floods, and others. The cause of the potential occurrence of this natural disaster is because Indonesia is located at the confluence of the Eurasian, Pacific, and Indo-Australian plates.

The information obtained on social media is growing very quickly and more effectively. When a disaster occurs such as a flood, social media works to inform users where the disaster occurred. In this paper, we use Twitter as a place to search for data. Twitter has been used successfully as an emergency service to inform the public of current developments. To get information we can search with relevant hashtags to find out where the incident occurred.

The results of this test will display a map of the territory of Indonesia and the point of occurrence of the disaster is taken by geolocation on the tweet data. The classification process will use the Naïve Bayes method. This grouping process is carried out in each region in Indonesia in real time. In this study, an accuracy value of 75% was obtained based on the value of fold 3 in the k-fold cross validation test.

Keywords: *Natural Disaster, Twitter, Naïve Bayes.*

1. Pendahuluan

Pada saat ini media sosial seperti Twitter banyak digunakan untuk menyebarkan informasi pada saat darurat. Twitter merupakan salah satu dari microblogging dan memungkinkan penggunaanya berbagi status dalam 280 karakter atau kurang. Dalam makalah ini kami akan fokus pada selebaran yang dibagikan di Twitter atau yang biasa disebut dengan *Tweet* untuk mencari data dimana terjadinya bencana alam. Indonesia merupakan negara yang rawan dilalui bencana alam. Salah satunya adalah banjir. Banjir sering terjadi di kota-kota besar yang ramai penduduk. Salah satu kota yang sering terjadi banjir yaitu ibukota Indonesia, Jakarta. Untuk mendapatkan informasi yang cepat dan akurat, Twitter bisa menjadi tempatnya. Mitigasi adalah suatu upaya untuk mengurangi resiko dari bencana. Salah satunya dengan peningkatan untuk menghadapi ancaman dari bencana tersebut menggunakan data dari Twitter. Untuk mempermudah masyarakat mengakses informasi bencana maka penulis akan menciptakan sistem informasi pemetaan bencana yang dapat memberikan informasi secara *realtime* dengan menggunakan data *tweet*.

Untuk mendapatkan informasi bencana terkini, penulis akan mengambil data dari twitter dan kemudian dikelompokkan ke beberapa kategori seperti gempa, banjir, dan longsor. Informasi yang terkumpul kemudian akan diklasifikasikan menggunakan algoritma *Naïve Bayes* dan ditampilkan di dalam maps berdasarkan *geolocation*.

2. Landasan Teori

2.1 Text Mining

Text Mining merupakan ekstraksi tersembunyi yang mana sebelumnya tidak diketahui. *Text mining* adalah salah satu bidang yang melakukan ekstraksi informasi dari teks alami. Fungsi dari *text mining* ini yaitu sebagai komunikasi informasi yang terpercaya yang nanti akan di ekstrak dan selanjutnya ditarik secara otomatis bagian kata yang penting jika berhasil [1]. Ada bermacam nama *text mining* seperti *text data mining*, *knowledge discovery* dari data base, analisis teks cerdas yang mengambil informasi dari kalimat yang tidak terstruktur [2].

2.1.1 Text Preprocessing

Text preprocessing adalah tahap awal dari *text mining* untuk merubah data menjadi terstruktur. Proses yang dilakukan seperti *casefolding*, *tokenizing*, *filtering*, dan *stemming* [3].

1. Case Folding

Case folding merupakan tahapan untuk mengganti semua huruf dalam dokumen menjadi huruf kecil (*lowercase*). Huruf yang dirubah mulai dari 'a' sampai 'z' [3].

2. Tokenizing

Tokenizing merupakan proses penguraian kalimat menjadi kata-kata dan menghilangkan tanda baca titik (.), koma (,) dan spasi serta karakter yang ada pada kata tersebut [3].

3. Filtering

Filtering merupakan tahap untuk menghilangkan kata yang tidak mempunyai arti seperti "Yang, ke, di, dengan, dan" di tengah kalimat [3].

4. Stemming

Stemming merupakan proses menemukan kata yang sangat dasar dari sebuah kata yang terdiri dari awalan, sisipan, dan akhiran [3].

2.2 Naïve Bayes Classifier

Naïve Bayes Classifier (NBC) adalah algoritma yang paling sederhana dan paling banyak digunakan [4] dengan teknik data *mining* yang menerapkan teori *Bayes* dan klasifikasi[5]. Dalam prosesnya *naïve bayes* tidak membutuhkan data dengan jumlah yang terlalu banyak dalam menentukan parameter. Hal tersebut merupakan salah satu keuntungan dari NBC [6]. Meskipun asumsi dari banyak orang bahwa *naïve bayes* jelas salah dalam dunia aplikasi, *naïve bayes* membuktikan bahwa mampu berkerja dengan baik [4].

Persamaan dari *Naïve Bayes Classifier* adalah

$$P(H|X) = \frac{P(X|H) \times P(H)}{P(X)} \quad (2.1)$$

Keterangan

X: Data dengan *class* yang belum diketahui

H: Hipotesis data X merupakan suatu *class* spesifik

P (H | X): Probalitas hipotesis H berdasar kondisi X

P(H): Probalitas hipotesis H *prior probability*

P (X | H): Probalitas X berdasarkan kondisi pada hipotesis H

P(X): Probalitas X

2.4 POS TF-IDF

TF-IDF (*Term Frequency Inverse Document Frequency*) merupakan teknik ekstraksi yang digunakan dalam text mining dan pencarian informasi. *Term Frequency* untuk mengukur berapa jumlah kata yang tampil di sebuah dokumen. Sedangkan *Inverse Document Frequency* untuk menentukan kata tersebut di perlukan atau tidak [7]. Untuk mengklasifikasikan dalam jumlah kalimat yang banyak, penulis menggunakan TF-IDF. Sistem pertama kali membuat kata kunci yang dimasukkan oleh pengguna, kedua digunakan TF-IDF untuk mengekstrak kata-kata yang diambil dari subjek kalimat berdasarkan kata kunci [8].

Teknik ini merupakan fungsi perhitungan bobot kata fitur yang digunakan pada saat sekarang ini. Terdiri dari dua bagian yaitu TF dan IDF. TF fokus kepada banyaknya jumlah kata yang muncul dalam dataset, sedangkan IDF merupakan ukuran kepentingan umum dari kalimat yang tersedia [9].

Rumus TF-IDF adalah sebagai berikut:

$$f_{ij} = \frac{f_{ij}}{\sum_{j \in D} f_{ij}} \quad (2.2)$$

f_{ij} merupakan jumlah kata yang muncul dalam file, sedangkan $\sum_{j \in D} f_{ij}$ merupakan jumlah kata yang muncul dalam semua file [9].

$$idf_j = \log \frac{|D|}{|D_j|} \quad (2.3)$$

$|D|$ merupakan jumlah total file yang ada di dalam data, $\{w_i \in V\}$ merupakan jumlah dokumen yang terkandung kata. Jika kata tidak ada di dalam data, akan menyebabkan *dividen* menjadi nol.

$$w_{i,j} = \frac{f_{i,j}}{\sqrt{\sum_{d \in D} f_{i,d}^2}} \quad (2.4)$$

$w_{i,j}$ merupakan bobot kata w_i . Dapat dilihat tinggi frekuensi kata dalam file tertentu dan frekuensi file data set dapat menghasilkan bobot tinggi TF-IDF.

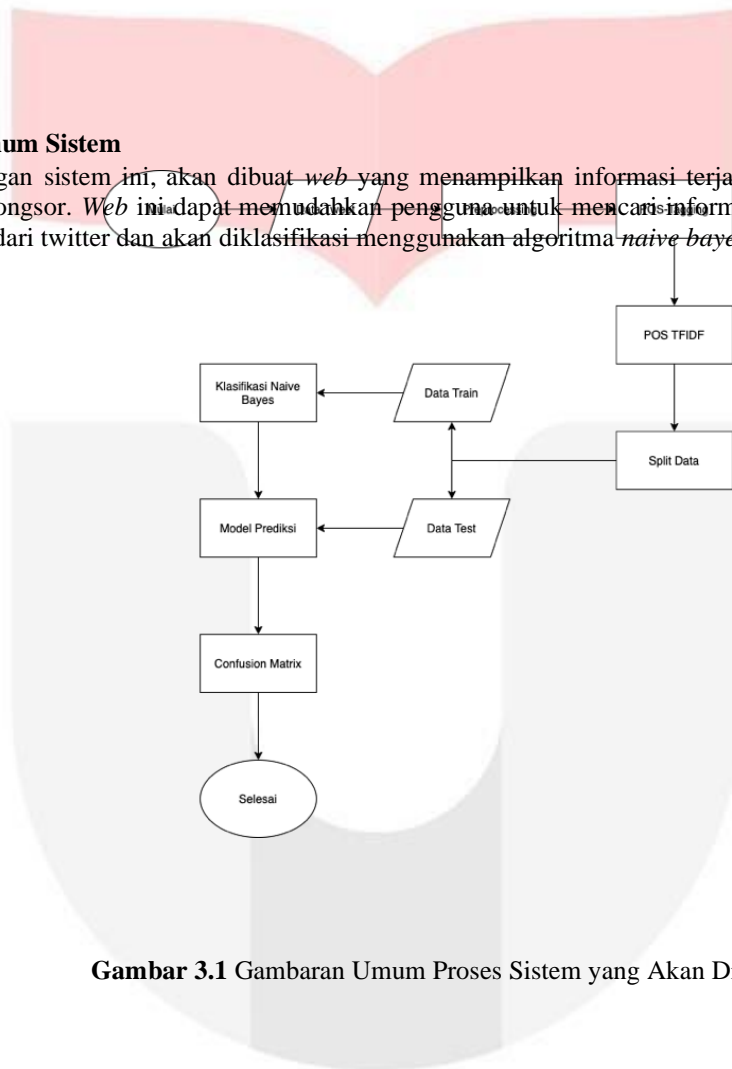
Dimana:

- Tf : jumlah kata/term dalam dokumen.
- Idf : jumlah term pada seluruh dokumen yang ada pada data.
- d : dokumen.

3. Perancangan

3.1 Gambaran Umum Sistem

Pada perancangan sistem ini, akan dibuat *web* yang menampilkan informasi terjadinya bencana alam seperti gempa, banjir dan longsor. *Web* ini dapat memudahkan pengguna untuk mencari informasi terbaru tentang bencana alam. Data diambil dari twitter dan akan diklasifikasi menggunakan algoritma *naive bayes*.



Gambar 3.1 Gambaran Umum Proses Sistem yang Akan Dibuat

3.2 Crawling Data

Proses *crawling* data dari twitter menggunakan *library Twitter Intelligence Tool (TWINT)* pada bahasa pemrograman *python*. Data yang digunakan untuk *training* adalah data *tweet* sejak tahun 2013 hingga tahun 2021, dengan *keyword* banjir, gempa dan longsor. Didapatkan 3841 *tweet* yang dilabelkan menjadi 4 kelas yaitu banjir, gempa, longsor, dan lainnya. Berikut adalah contoh bentuk dataset:

No	Data Tweet
1	Gempa disebut bertitik pusat di 18 kilometer sebelah tenggara Padang Sidempuan. #Gempa #SumateraUtara https://t.co/Ex3X1poMXx
2	Gempa di Padang Lawas Utara dipicu oleh aktivitas Sesar Sumatera https://t.co/GhZJxNUUmT

Tabel 3.1 Hasil Crawling Data

3 | sepertinya dia udah bingung mau ngerjain apa soal banjir...

3.3 Pelabelan Dataset

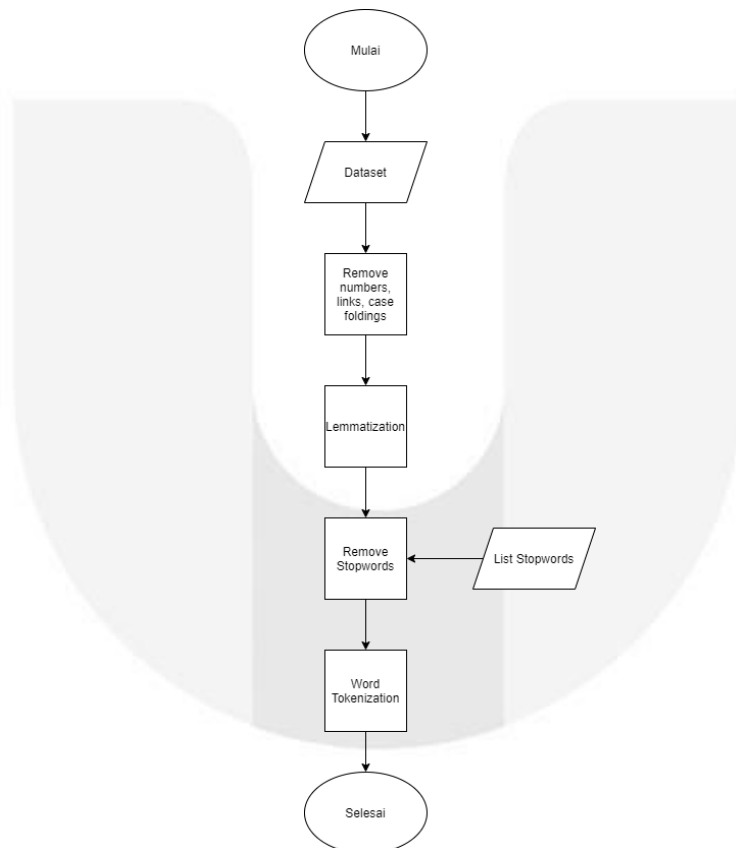
Pelabelan dataset dilakukan secara manual, dataset dibagi menjadi 4 kelas label yakni Banjir, Gempa dan Longsor dengan ketentuan pelabelan data adalah *tweet* pengguna yang memiliki informasi atau pengguna tersebut terkena/terdampak bencana banjir pada saat itu juga. Bencana alam banjir, gempa dan longsor yang sedang terjadi dan pengguna tersebut tidak mengalami/terdampak bencana alam, dengan proporsi data dengan kelas banjir sebanyak 968 data, kelas gempa sebanyak 799 data, kelas longsor sebanyak 843 data, dan kelas lainnya sebanyak 1230 data

Tabel 1.2 Dataset yang telah memiliki label

Tweet	Label
BREAKING NEWS!! Gempa Bumi M 5,3 V MMI Guncang Padang Sumatera Utara, BMKG: Waspada Bangunan Roboh https://t.co/sAhhX0HyvT	Gempa
Dua Kabupaten di Aceh Dilanda Banjir, Ribuan Warga Terdampak: Sementara di Kabupaten Aceh Jaya, katanya, tiga kecamatan yakni Kecamatan Jaya, Darul Hikmah dan Sampoiniet direndam banjir. Ketinggian banjir di Aceh Jaya sekitar 20 sampai 70 sentimeter.... https://t.co/eIkopouOPO	Banjir
Tanah longsor di pegunungan berbatu ya bestie	Longsor
ini Jogja dalam beberapa waktu terakhir lagi rutin gempa kecil yaa? jakal atas nggak kerasa sama sekali. baru tau setelah liat twitter	Lainnya

3.4 Pre-Processing

Pada pembuatan sistem ini sebelum menjadi *web*, data set akan diproses untuk di *filter* menggunakan *text pre-processing*. Tahap-tahap *text pre-processing* tersebut dapat dilihat pada *flowchart* dibawah ini.



Gambar 3. 1 Flowchart Preprocessing

Pada gambar 3.2 adalah proses *Pre-Processing*, dataset dibersihkan dengan menghilangkan angka, tanda baca dan lain-lain yang tidak digunakan dalam penelitian ini, lalu data akan melewati proses *lemmatization* untuk mengembalikan sebuah kata menjadi kata dasar dari kata itu sendiri, setelah itu kata yang tidak merubah arti dari keseluruhan kalimat akan dihilangkan dengan proses *remove stopwords*, yang terakhir adalah proses *word tokenization*.

3.4.1 Data Cleaning

Proses ini adalah untuk menghilangkan teks yang tidak digunakan seperti menghilangkan angka, tanda baca, *hashtag*, *url* karena tidak berpengaruh untuk proses klasifikasi, setelah itu semua kalimat diubah menjadi huruf kecil.

Table 3.3 Data Cleaning

Data (Input)	Data (Output)
telah terjadi gempa sebanyak 13 kali di selatan Yogyakarta dalam kurun 24 Jam terakhir, selalu waspada dan kita semua berharap semua aman. https://t.co/gzWyzXF14k	telah terjadi gempa sebanyak 13 kali di selatan yogyakarta dalam kurun 24 jam terakhir selalu waspada dan kita berharap semua aman
Tim Gabungan, Polsek Limau Gotong Royong Jalan Longsor https://t.co/5CHMDvvpUe	tim gabungan polsek limau gotong royong jalan longsor
Ribuan Rumah di Lebak Banten Terendam Banjir https://t.co/11YNiVQJbh https://t.co/84UdHmMHH6	ribuan rumah di lebak banten terendam banjir

Pada tabel 3.3 menunjukkan hasil dari proses *data cleaning* yang mana untuk menghilangkan url, kata dengan *hashtag*, tanda baca, angka, dan merubah kata dari huruf besar menjadi huruf kecil.

3.4.2 Lemmatizer

Lemmatizer adalah proses mengubah sebuah kata mejadi kata dasar dari kata itu sendiri tanpa merubah makna/arti dari kata tersebut. Berikut adalah contoh proses *lemmatization*

Table 3.4 Lemmatizer

Data (Input)	Data (Output)
telah terjadi gempa sebanyak 13 kali di selatan yogyakarta dalam kurun 24 jam terakhir selalu waspada dan kita berharap semua aman	telah jadi gempa banyak 13 kali di selatan yogyakarta dalam kurun 24 jam akhir selalu waspada dan kita harap semua aman
tim gabungan polsek limau gotong royong jalan longsor	tim gabung polsek limau gotong royong jalan longsor
ribuan rumah di lebak banten terendam banjir	ribu rumah lebak banten rendam banjir

Pada tabel 3.4 adalah proses *lemmatization*, yaitu masing-masing kata dalam sebuah kalimat dikembalikan ke kata dasar dari kata itu sendiri, proses ini bertujuan untuk mengurangi jumlah *unique word*.

3.4.3 Stopword Removal

Stopword Removal adalah proses penghilangan kata penghubung yang tidak memiliki makna/arti dalam suatu kalimat, ada atau tidak nya kata ini tidak akan merubah arti dari suatu kalimat.

Table 3.5 Stopword Removal

Data (Input)	Data (Output)
telah jadi gempa banyak 13 kali di selatan yogyakarta dalam kurun 24 jam akhir selalu waspada dan kita harap semua aman	jadi gempa banyak selatan yogyakarta dalam kurun jam akhir selalu waspada kita harap semua aman
tim gabung polsek limau gotong royong jalan longsor	tim gabung polsek limau gotong royong jalan longsor
ribuan rumah di lebak banten terendam banjir	ribuan rumah lebak banten rendam banjir

Pada tabel 3.5 adalah proses data yang telah melewati proses *stop removal* dan akan dihilangkan kata penghubung yang tidak terlalu berpengaruh.

3.4.3 Word Tokenize

Word Tokenize adalah proses untuk memisahkan kata didalam kalimat yang menghasilkan *list* kata-kata yang membentuk kalimat itu sendiri. Seperti bisa dilihat pada tabel berikut.

Table 3.6 Word Tokenize

Data (Input)	Data (Output)
--------------	---------------

jadi gempa banyak selatan yogyakarta dalam kurun jam akhir selalu waspada kita harap semua aman	[‘jadi’, ‘gempa’, ‘banyak’, ‘selatan’, ‘yogyakarta’, ‘dalam’, ‘kurun’, ‘jam’, ‘akhir’, ‘selalu’, ‘waspada’, ‘kita’, ‘harap’, ‘semua’, ‘aman’]
tim gabung polsek limau gotong royong jalan longsor	[‘tim’, ‘gabungan’, ‘polsek’, ‘limau’, ‘gotong’, ‘royong’, ‘jalan’, ‘longsor’]
ribuan rumah lebak banten rendam banjir	[‘ribu’, ‘rumah’, ‘lebak’, ‘banten’, ‘rendam’, ‘banjir’]

Pada tabel 3.6 menunjukkan hasil dari *word tokenize* yang bertujuan untuk memisahkan kata-kata dalam sebuah kalimat untuk nanti dilanjutkan ke proses *POS tag* agar masing-masing kata dapat diberikan *POS tag*.

3.5 POS-Tagging

Pada proses ini, kata dalam sebuah kalimat akan dikategorikan berdasarkan peran dan fungsinya. Digunakan *Pre-trained POS-Tagger* dataset untuk dapat diimplementasikan kedalam Bahasa Indonesia.

Table 3.7 POS-Tagging

Data (Input)	Data (Output)
jadi gempa banyak selatan yogyakarta dalam kurun jam akhir selalu waspada kita harap semua aman	[(‘jadi’, ‘NN’), (‘gempa’, ‘JJ’), (‘banyak’, ‘NN’), (‘selatan’, ‘NN’), (‘yogyakarta’, ‘NN’), (‘kurun’, ‘NN’), (‘jam’, ‘NN’), (‘akhir’, ‘JJ’), (‘selalu’, ‘RB’), (‘waspada’, ‘JJ’), (‘kita’, ‘NN’), (‘harap’, ‘NN’), (‘semua’, ‘RB’), (‘aman’, ‘JJ’)]
tim gabung polsek limau gotong royong jalan longsor	[(‘tim’, ‘NN’), (‘gabung’, ‘JJ’), (‘polsek’, ‘NN’), (‘limau’, ‘NN’), (‘gotong’, ‘JJ’), (‘royong’, ‘JJ’), (‘jalan’, ‘RB’), (‘longsor’, ‘NN’)]
ribu rumah lebak banten rendam banjir	[(‘ribu’, ‘RB’), (‘rumah’, ‘NN’), (‘lebak’, ‘NN’), (‘banten’, ‘NN’), (‘rendam’, ‘NN’), (‘banjir’, ‘NN’)]

Pada tabel 3.7 adalah proses pemberian *Part Of Speech Tag (POS)* kepada masing-masing kata dalam sebuah kalimat, proses ini bertujuan untuk mengetahui seberapa penting sebuah kata tersebut dibandingkan dengan kata lainnya dalam sebuah kalimat.

3.6 Pembobotan

Pembobotan dilakukan untuk merubah dataset menjadi angka yang dihitung berdasarkan rumus dari *POS-TF*IDF* yang akan memberikan bobot pada masing-masing kata agar bisa lanjutkan ke proses klasifikasi.

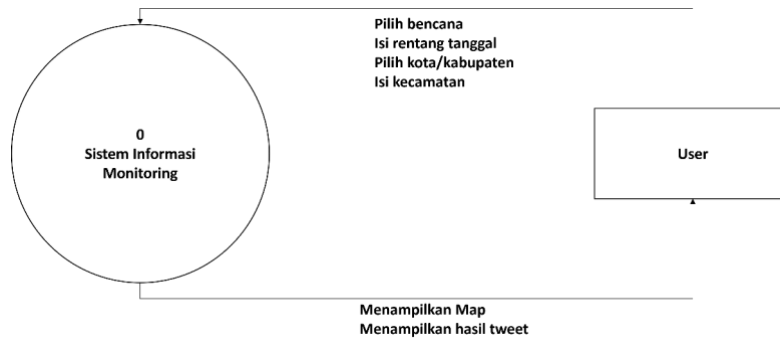
$$\begin{aligned}
 & \frac{1 * 5}{\sum((1 * 5) + (1 * 5) + (1 * 5) + (1 * 3))} \\
 = & \frac{3}{-} \\
 & \text{TF} = 1 + \log \text{IDF} \\
 & \text{TF} * \text{IDF} = 0.4106
 \end{aligned}$$

3.6 Perancangan Proses

3.6.1 Data Flow Diagram (DFD)

Sistem *monitoring* bencana alam berbasis web ini memiliki diagram alir guna untuk mengetahui kegunaan dari input dan output sistem web aplikasi. *Data Flow Diagram* dapat dilihat sebagai berikut

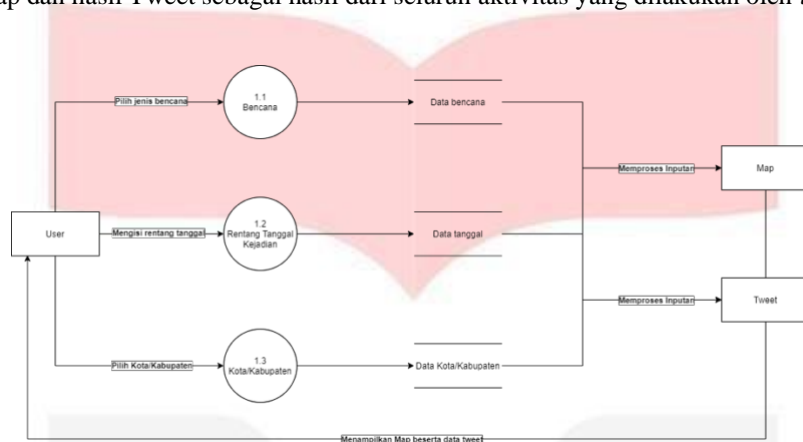
1. *DFD Level 0*



Gambar 3. 2 DFD Level 0

Pada gambar 3.3 menunjukkan *Data Flow Diagram (DFD) Level 0* sebagai awal dari program, *user* dapat melakukan pilih bencana, pilih kota/kabupaten, isi rentang tanggal kejadian serta isi kecamatan (*optional*). Kemudian Sistem akan menampilkan Map dan hasil Tweet sebagai hasil dari seluruh aktivitas yang dilakukan oleh *user*.

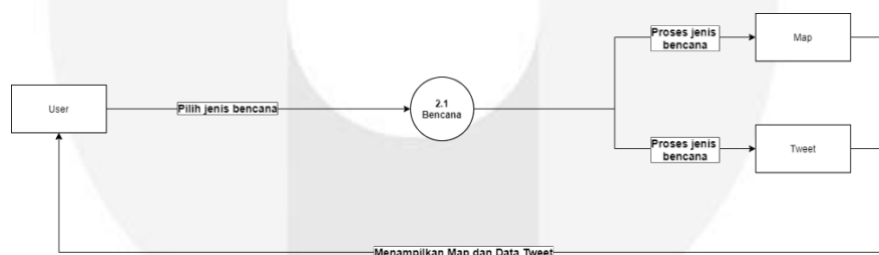
2. *DFD Level 1*



Gambar 3. 3 DFD Level 1

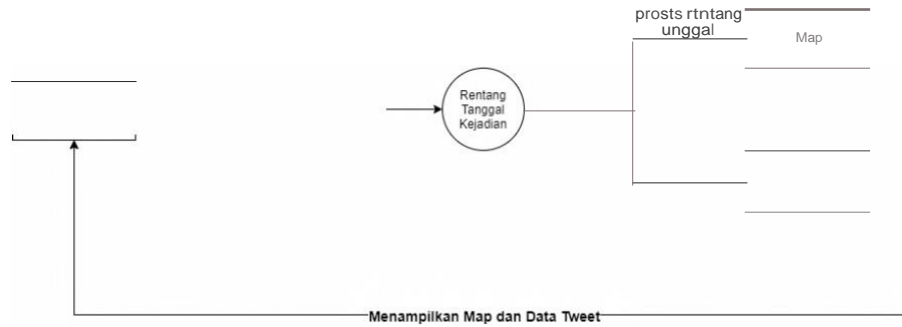
Pada gambar 3.4 menunjukkan *Data Flow Diagram (DFD) Level 1*. *User* akan melakukan pengisian parameter yakni jenis bencana, rentang tanggal kejadian, serta kota/kabupaten. Kemudian sistem akan menampilkan *Map* dan data *Tweet* sebagai *output* sebagai keluaran dari parameter yang di-*input* kan oleh *user*.

3. *DFD Level 2*



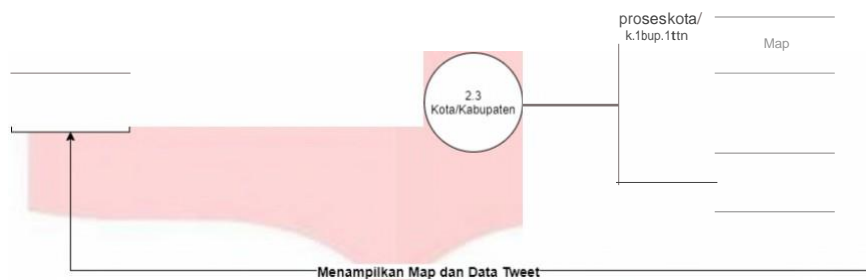
Gambar 3. 4 DFD Level 2 Proses 1

Pada gambar 3.5 menunjukkan *Data Flow Diagram (DFD) Level 2* proses 1. Berbeda dengan DFD level 0, DFD level 2 memiliki 3 proses dari masing-masing parameter yang berikan oleh *user*, pada proses 1, *user* akan memilih jenis bencana yakni banjir/gempa/longsor. Sistem akan merespon dengan menampilkan *Map* serta *Tweet* yang berhubungan dengan bencana yang dipilih.



Gambar 3. 5 DFD Level 2 Proses 2

Pada gambar 3.6 menunjukkan *Data Flow Diagram (DFD) Level 2* proses 2. Berbeda dengan *DFD level 0*, *DFD level 2* memiliki 3 proses dari masing-masing parameter yang berikan oleh *user*, pada proses 2, *user* akan mengisikan rentang tanggal kejadian. Sistem akan merespon dengan menampilkan *Map* serta *Tweet* yang berhubungan dengan bencana yang dipilih.



Gambar 3. 6 DFD Level 2 Proses 3

Pada gambar 3.7 menunjukkan *Data Flow Diagram (DFD) Level 2* proses 3. Berbeda dengan *DFD level 0*, *DFD level 2* memiliki 3 proses dari masing-masing parameter yang berikan oleh *user*, pada proses 2, *user* akan memilih kota/kabupaten. Sistem akan merespon dengan menampilkan *Map* serta *Tweet* yang berhubungan dengan bencana yang dipilih.

4. Implementasi dan Pengujian

4.1 Dataset

Dataset yang digunakan dari tweet user dan dibagi dalam 4 kategori yaitu gempa, banjir, longsor, dan lainnya. Data dengan kategori gempa, banjir, longsor merupakan tweet yang terdapat informasi user sedang terdampak, sedangkan kelas lainnya merupakan tweet user tidak terdampak dan hanya mengeluarkan opini mereka tentang bencana tersebut. Keseluruhan data tweet diambil dari twitter dan dikelompokkan satu per satu oleh penulis.

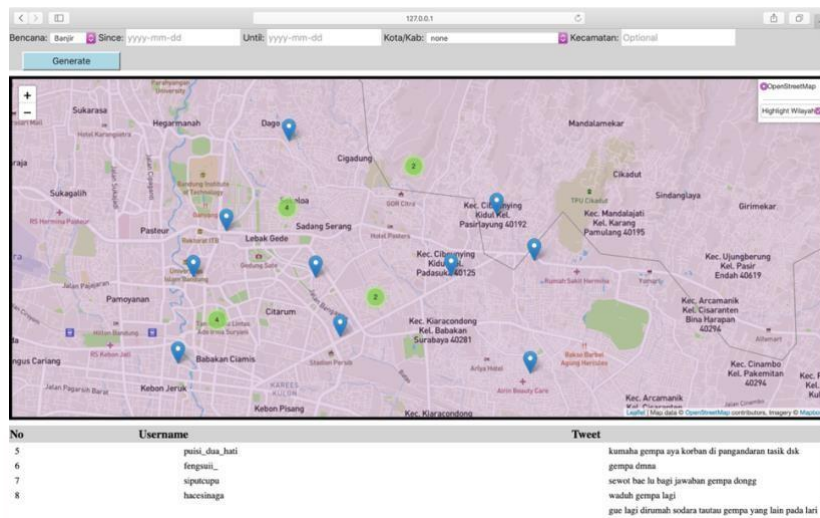
4.2 Implementasi Desain Antarmuka

Implementasi desain antarmuka dari sistem informasi monitoring bencana alam dari data media sosial terdapat pada gambar berikut.



Gambar 4. 1 Halaman Depan Web Aplikasi

Pada gambar 4.1 user diharuskan untuk input parameter yang tersedia pada *web*. Dapat dilihat pada gambar terdapat pilihan bencana yaitu banjir, gempa, longsor. *User* diharuskan memilih salah satu dari bencana yang ada, lalu *user* diharuskan memilih lokasi kota/kabupaten untuk melihat lokasi dimana terjadinya bencana, terdapat kolom kecamatan yang bersifat optional.



Gambar 4. 2 Hasil Pemetaan Tweet

Pada gambar 4.2 diatas merupakan hasil dari mapping tweet data berdasarkan geolokasi yang ada pada tweet. Hasil dari klasifikasi juga bisa dilihat pada tabel di bawah *map*.

4.3 Pengujian Alpha

4.3.1 Tujuan Pengujian

Pengujian *alpha* dilakukan untuk mengetahui fungsi dari sistem informasi monitoring bencana alam dari data media sosial dapat berjalan dengan baik. Pengujian ini dilakukan dengan mencoba semua fitur yang sudah ada di *web*.

4.3.2 Skenario Pengujian

Skenario pengujian alpha yang penulis lakukan dapat dilihat pada tabel 4.1 berikut.

Table 4.1 Skenario Pengujian Alpha

No	Fitur yang diuji	Detail pengujian	Jenis pengujian
1	Membuka web	Memampilkan halaman	<i>Black box</i>
2	Menu pilihan bencana alam	Menampilkan 3 jenis bencana alam: gempa, banjir, longsor	<i>Black box</i>
3	Menu pilihan kota/kabupaten	Menampilkan kota/kabupaten di Indonesia	<i>Black box</i>
4	Klasifikasi	Menampilkan hasil tweet klasifikasi dan mapping dari data tweet	<i>Black box</i>

Pada skenario pengujian *alpha* pada tabel 4.1 terdapat 4 fitur yang diuji. Pengujian *alpha* dilakukan dengan *black box* dimana hanya mengamati hasil dari eksekusi dengan memeriksa apakah fitur bekerja dengan benar.

4.3.3 Hasil Pengujian Alpha

Hasil pengujian *alpha* sistem *monitoring* bencana alam dengan data media sosial dapat dilihat pada tabel di bawah ini.

Table 4.2 Hasil Pengujian Membuka Web

Data masukan	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Membuka <i>web</i>	Menampilkan halaman utama	Halaman utama ditampilkan	Berhasil

Pada tabel 4.2 di atas menunjukkan pengujian membuka web dapat menampilkan halaman utama dengan hasil yang diharapkan.

Table 4.3 Hasil Pengujian pilihan bencana alam

Data masukan	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Pilih bencana yang terdapat pada kolom	Menampilkan 3 jenis bencana alam yaitu banjir, gempa, longsor	Menampilkan 3 bencana alam	Berhasil

Pada tabel 4.3 menunjukkan pengujian pilihan bencana alam berhasil dengan yang hasil yang diharapkan.

Table 4.4 Hasil Pengujian Pilihan Kota/Kabupaten

Data masukan	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Pilih kolom kota/kabupaten	Menampilkan nama kota/kabupaten di Indonesia	Dapat menampilkan nama kota/kabupaten di Indonesia	Berhasil

Pada tabel 4.4 menunjukkan bahwa pengujian pilihan kota/kabupaten berhasil sesuai yang diharapkan.

Table 4.5 Hasil pengujian tombol klasifikasi

Data masukan	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Pilih tombol "Generate"	Mengklasifikasikan tweet data dan memunculkan tweet pada tabel <i>mapping</i>	Dapat mengklasifikasikan tweet data dan memetakan di map serta memunculkan tabel berisi tweet dan <i>username</i>	Berhasil

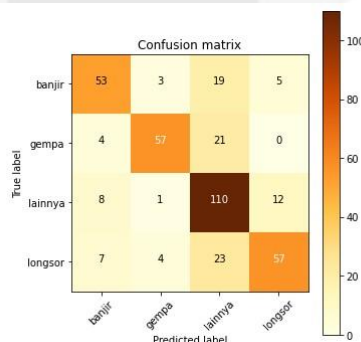
Pada tabel 4.5 menunjukkan pengujian tombol klasifikasi berhasil sesuai yang diharapkan.

4.4 Pengujian Akurasi

Table 4.9 Akurasi Dataset

Testing	Data Training	Data Testing	Results			
			Accuracy	Precision	Recall	F1-Score
1	90%	10%	72%	75%	71%	72%
2	80%	20%	70%	73%	69%	70%
3	70%	30%	70%	74%	69%	71%
4	60%	40%	71%	74%	70%	71%
5	50%	50%	70%	73%	69%	70%
6	45%	55%	69%	72%	68%	69%
7	35%	65%	69%	72%	68%	69%
8	25%	75%	66%	69%	65%	66%
9	15%	85%	64%	67%	62%	64%
10	5%	95%	61%	64%	59%	61%

Untuk mengukur kinerja algoritma *Naive Bayes*, maka dilakukan *confusion matrix* sebagai berikut:



Gambar 4.3 Confusion Matrix

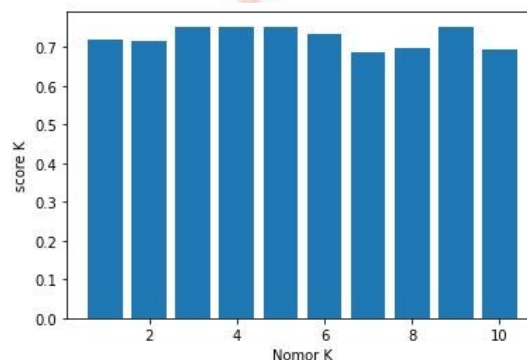
Gambar 4.4 memperlihatkan kinerja klasifikasi dengan metode *Naïve Bayes*. Didapat hasil *confusion matrix* dengan mendapatkan nilai akurasi sebesar 0.72 atau sebesar 72%.

4.4.1 Pengujian Kfold Cross Validation

Pada proses pengujian ini kali ini data yang digunakan berjumlah 3840 yang berisi bencana dan tweet. Jumlah nilai split yang digunakan pada *cross validation* yaitu berjumlah 10 kemudian dibagi dengan data *train* dan data *test*. Setelah data dipartisi tahap selanjutnya yaitu masuk ke percobaan *cross validation*. Percobaan pertama yaitu mengambil *fold* 1 pada partisi pertama akan dijadikan sebagai data *test*, sisanya menjadi data *training*. Kemudian untuk *fold* 2 pada partisi yang sebelumnya merupakan data *test* untuk di *fold* 2 ini menjadi data *training* dan data *test*nya pindah ke bagian partisi ke dua, dan seterusnya hingga *fold* 10. Untuk pengujian *cross validation* untuk percobaan sampai *fold* 10 bisa dilihat pada tabel 24.

Table 4.10 Score Partisi

FOLD	SCORE
FOLD 1	0.71875 \approx 71%
FOLD 2	0.7135416666666666 \approx 71%
FOLD 3	0.7526041666666666 \approx 75%
FOLD 4	0.7526041666666666 \approx 75%
FOLD 5	0.7526041666666666 \approx 75%
FOLD 6	0.7317708333333334 \approx 73%
FOLD 7	0.6875 \approx 68%
FOLD 8	0.6953125 \approx 69%
FOLD 9	0.7526041666666666 \approx 75%
FOLD 10	0.6927083333333334 \approx 69%



Gambar 4. 4 Grafik Kfold Cross Validation

Berdasarkan hasil tabel 24, didapat hasil pada *fold* 3, 4, 5 dan 9 memiliki hasil yang sama besar, yakni 75% sekaligus menjadi nilai persentase terbesar dalam pengujian *Kfold Cross Validation*.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil dari penelitian dan pengujian yang telah dilakukan oleh penulis untuk kepentingan tugas akhir ini dapat ditarik kesimpulan bahwa:

1. *Web* dapat memetakan hasil dari data tweet yang telah diklasifikasikan berdasarkan *geolocation* dari masing-masing data tweet.
2. Berdasarkan perhitungan *Confusion Matrix*, didapatkan nilai akurasi sebesar 72%, dengan pembagian data *train* sebesar 90% dan data *test* sebesar 10% untuk tiap dataset.
3. Dengan data *train* sebesar 90% dan data *test* sebesar 10%, maka didapat *precision* untuk dataset banjir sebesar 74%, *recall* sebesar 66% dan *f1-score* sebesar 70%. Untuk *precision* dataset gempa sebesar 88%, *recall* sebesar 70% dan *f1-score* sebesar 78%. Sedangkan untuk *precision* dataset longsor mendapat nilai sebesar 77%, *recall* sebesar 63% dan *f1-score* sebesar 69%.

5.2 Saran

Berdasarkan hasil dari penelitian dan pengujian yang telah dilakukan oleh penulis, saran yang diusulkan adalah:

1. Meningkatkan akurasi dan performa dari sistem dengan cara menambahkan data latih yang berkualitas
2. Menambahkan data tweet bencana alam atau bencana seperti musibah lainnya.
3. Dapat menemukan *geolocation* dari data tweet dengan metode lain.

Referensi

- [1] K. L.Sumathy and M. Chidambaram, "Text Mining: Concepts, Applications, Tools and Issues An Overview", *International Journal of Computer Applications*, vol. 80, no. 4, pp. 29-32, 2013. Available: 10.5120/13851-1685 [Accessed 23 August 2021].
- [2] S. Dang, "Text Mining : Techniques and its Application", *IJETI International Journal of Engineering & Technology Innovations*, vol. 1, p. 22, 2014. Available: https://www.researchgate.net/publication/273038150_Text_Mining_Techniques_and_its_Application. [Accessed 23 August 2021].
- [3] Kurniawan, B., Fauzi, M., & Widodo, A. Klasifikasi Berita Twitter Menggunakan Metode Improved Naïve Bayes. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 1, no. 10, p. 1193-1200, juli 2017. ISSN 2548-964X. Tersedia pada: <<https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/361>>. Tanggal Akses: 09 agu. 2021
- [4] M. Allahyari et al., "A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques", *arXiv.org*, 2021. [Online]. Available: <https://arxiv.org/abs/1707.02919v2>. [Accessed: 23- Aug- 2021].
- [5] A.Indriani.Klasifikasidataforumdenganmenggunakanmetodenaa Naïvebayesclassifier.InSeminarNasional Aplikasi Teknologi Informasi (SNATI), volume 1, 2014. [6] Reddy, T. A. J dan P. S. Reddy. "Outcome Based Education—Some Initiatives". *Open Journal of Social Sciences*. 02(10):7-11. 2014.
- [6] QAULI ADZKIA, Klasifikasi Status Kemacetan Menggunakan Naive Bayes Classification (Studi Kasus di Persimpangan Buah Batu Kota Bandung), 2020
- [7] SRIVIDYA, Kotagiri; SOWJANYA, A. Mary. Aspect based sentiment analysis using pos tagging and tfidf. *International Journal of Engineering and Advanced Technology (IJEAT)*, 2019, 8.
- [8] S. Kim and J. Gil, "Research paper classification systems based on TF-IDF and LDA schemes", *Human-centric Computing and Information Sciences*, vol. 9, no. 1, 2019. Available: 10.1186/s13673-019-0192-7 [Accessed 23 August 2021].
- [9] C. Liu, Y. Sheng, Z. Wei and Y. Yang, "Research of Text Classification Based on Improved TF-IDF Algorithm," 2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE), 2018, pp. 218-222, doi: 10.1109/IRCE.2018.8492945