SISTEM INFORMASI MONITORING BENCANA ALAM DARI DATA MEDIA SOSIAL MENGGUNAKAN METODE K-NEAREST NEIGHBOR

NATURAL DISASTER MONITORING INFORMATION SYSTEM FROM SOCIAL MEDIA DATA USING K-NEAREST NEIGHBOR METHOD

Kevin Manfield Anderson Pasaribu[1], Randy Erfa Saputra[2] Casie Setianingsih[3]

[1,2,3] Universitas Telkom, Bandung
[1]kevinmanfield@student.telkomuniversity.ac.id, [2]resaputra@telkomuniversity.ac.id, [3]setiacasie@telkomuniversity.ac.id

Abstrak

Bencana alam adalah hal yang alami dan tidak bisa diperkirakan. Dampak bencana alam tergantung dari besar intensitasnya, yang dapat berupa tanah longsor, banjir, gempa bumi, bahkan korban jiwa sekalipun. Banyak hal yang bisa diminimalisir sebelum bencana tersebut meluas. Untuk itulah pentingnya informasi akan terjadinya bencana.

Sosial media adalah tempat yang dapat menghubungkan satu informasi ke informasi yang lain, sehingga dapat tersebar luas. Perkembangan media sosial saat ini sangatlah cepat terlebih dengan adanya berita yang berkaitan dengan bencana di sekitar. Salah satu media sosial yang sekarang sangat banyak digunakan adalah Twitter. Dengan menggunakan Twitter, masyarakat dapat dengan cepat memberi penyebaran informasi bencana melalui tweet agar dampak penanggulangan dapat dipercepat. Maka dari itu, inilah pentingnya untuk mengetahui informasi secara real-time tentang jumlah kejadian bencana alam agar antisipasi dapat dilakukan sejak dini.

Maka dari penjelasan yang disebutkan diatas dibutuhkan sistem yang dapat memilah dengan sendiri data bencana alam pada Tweet. Hasil pengujian dibuat untuk menampilkan mapping bencana yang terjadi di wilayah Indonesia yang diklasifikasikan berdasarkan area yang memiliki data tweet dalam bentuk visualisasi peta wilayah bencana mana yang lebih tinggi frekuensi serta jenis bencananya. Klasifikasi wilayah dilakukan menggunakan metode algoritma K-Nearest Neighbor. Dalam penelitian ini hasil nilai uji Confusion matrix memiliki akurasi yang terbaik dengan menggunakan metric Jaccard sebesar 86% dan untuk performansi data sharing menggunakan k-fold cross validation 10 Fold, hasil akurasi terbaiknya adalah metric Jaccard yaitu sebesar 83% pada Fold 8.

Kata Kunci: *twitter, bencana alam, klasifikasi, confusion matrix, k-fold cross validation*

Abstract

Natural disasters are natural and unpredictable. The impact of natural disasters depends on their intensity, which can be in the form of landslides, floods, earthquakes, and even fatalities. Many things can be minimized before the disaster spreads. For this reason, it is important to have information about the occurrence of disasters.

Social media is a place that can connect one information to another, so that it can be spread widely. The development of social media today is very fast, especially with the news related to disasters around. One of

the most widely used social media is Twitter. By using Twitter, the public can quickly disseminate disaster information through tweets so that the impact of the response can be accelerated. Therefore, it is important to know real-time information about the number of natural disasters so that anticipation can be done early on.

So from the explanation mentioned above, a system is needed that can sort out natural disaster data by itself on Tweets. The test results are made to display the mapping of disasters that occur in the territory of Indonesia which are classified based on the area that has tweet data in the form of a map visualization of which disaster area has the higher frequency and type of disaster. Regional classification is done using the K-Nearest Neighbor algorithm method. In this study. In this study, the results of the Confusion matrix test have the best accuracy using the Jaccard metric of 86% and for data sharing performance using k-fold cross validation 10 Fold, the best accuracy result is the Jaccard metric, which is 83% at fold 8.

***Keywords**: twitter, natural disaster, classification, confusion matrix, k-fold cross validation*

1. Pendahuluan

Twitter adalah media sosial yang saat ini sedang banyak digemari masyarakat luas. Tentu saja ini dapat digunakan ke arah lebih positif, khususnya dalam pemberitahuan informasi tentang bencana alam. Berdasarkan survey We Are Social pada Januari 2021, Twitter di Indonesia di Indonesia menempati peringkat keempat setelah Whatsapp, Facebook, Instagram dan Tiktok [1]. Artinya sosial media di Indonesia khususnya Twitter sudah menjadi sebuah kebutuhan bagi masyarakat luas. Informasi dalam Twitter sangatlah cepat bahkan dalam hitungan menit.

Dikarenakan Indonesia adalah salah satu negara yang sangat rawan dalam bencana alam seperti banjir, angin puting beliung, dan tanah longsor. Maka hampir seluruh kawasan di Indonesia berpotensi menimbulkan bencana alam dengan intensitas dan kekuatan yang berbeda-beda. Bencana yang sering melanda Indonesia adalah Banjir,Gempa dan Longsor. Banjir khususnya sangatlah sering terjadi pada kota-kota besar seperti Jakarta.

Untuk meminimalisir resiko bencana tentu, penting bagi masyarakat untuk lebih mengetahui informasi seputar bencana alam yang sedang terjadi. Salah satunya bisa dengan menggunakan media sosial media Twitter. Karena pada saat ini Twitter sudah dipakai oleh segala kalangan usia, dan informasi tweet dalam Twitter sangatlah cepat untuk diketahui orang banyak. Salah satu informasi yang cepat untuk menyebar adalah tentang bencana alam khususnya di wilayah Indonesia. Informasi tweet tentang bencana alam sangat berguna untuk mengetahui dimana saja, kapan, hingga siapa saja korban jiwa dalam bencana alam tersebut bahkan dapat dimanfaat sebagai antisipasi jikalau bencana akan melanda daerah orang yang belum terkena dampaknya.

Sehubungan dengan permasalahan diatas, penulis mempunyai ide untuk membuat sebuah sistem informasi bencana dari data sosial media terkhusus Twitter dimana sistem informasi ini akan memberi mengklasifikasikan dan memetakan bencana alam sesuai dengan jenisnya dan lokasinya masing-masing

2. Landasan Teori

2.1 Text Preprocessing

Merupakan data yang sudah diambil awalnya akan dilakukan preprocessing. Preprocessing data teks merupakan salah satu metode yang efektif dalam hal pembuatan data yang tidak terstruktur, terstruktur dan bermakna. Diterapkan untuk menghilangkan sifat data yang tidak terstruktur yang diperoleh dari media sosial [2].

2.1.1 POS TF-IDF

TF-IDF adalah algoritma yang berfungsi untuk mengetahui nilai kata dalam suatu dokumen sehingga dapat menyimpulkan kata tersebut dianggap penting atau tidak . Dalam suatu dokumen, Term Frequency adalah banyaknya term yang muncul. Dalam menilai pentingnya suatu kata dibutuhkan Inverse Document Frequency [3].

$$tf(t,d) = c\,(t,d) \quad (1)$$

Dimana dalam dokumen d dihitung frekuensi

kata t dari c(t,d).

$$tf(t,d) = \frac{c(t_i d)}{docLen(d)} \quad (2)$$

docLen(d) = adalah panjang dokumen d, rumusnya adalah untuk normalize panjang dokumen

$$\begin{cases} \frac{C(t,d)*Wpos(t)}{\sum C(t_i,d)*Wpos(t_i)} * \left(1 + \log\left(\frac{N}{k}\right)\right) & if\ c(t,d) \geq 1 \\ 0 & otherwise \end{cases} \quad (3)$$

$$\text{Wpos} = \begin{cases} \text{W1} & if\ t\ is\ a\ verb\ or\ noun \\ \text{W2} & \text{if t is a adverb } or\ adjective \\ \text{W3} & \text{otherwise} \end{cases} \quad (4)$$

tf = is a term frequency
t = is the term
d = is a document
c(d,t) = term frequency t in the document
N = total number of documents
K = number of documents that occur word t

Nilai W1 > W2 > W3 >0, menunjukkan bahwa kata-kata dengan tag kata kerja atau kata benda lebih penting daripada kata-kata dengan tag kata keterangan atau kata sifat karena setiap kata per kata akan diberi bobot tambahan dari part of speech tag [4].

2.2 K-Nearest Neighbor

KNN merupakan algoritma pengenalan pola klasifikasi yang sudah lama ada. Sangat mudah untuk mengklasifikasikan dokumen berjenis teks. Pertama cari nilai K tetangga yang terdekat sesuai dengan strategi perhitungan similarity, kemudian jumlahkan semua masing-masing kemiripan sesuai kategorinya [5]. Pada algoritma K-Nearest Neighbor juga memiliki beberapa metric yang bisa dipilih dan digunakan pada dataset agar mendapatkan performa terbaik. Beberapa Distance Metric yang bisa digunakan pada algoritma K-Nearest Neighbor adalah Minkowski, Manhattan, Euclidean, Cosine, Jaccard, dan Hamming.

2.2.1 *Minkowski Metric*

Dalam KNN, *metric Minkowski* merupakan *metric* rangkuman dari Euclidean distance dan Manhattan distance, yang hanya dapat dihitung dalam (normed vector space). Minkowski memiliki parameter yang bisa di ubah menjadi Euclidean distance dan Manhattan distance yaitu dengan mengubah parameternya menjadi p = 1 atau p = 2. Berikut adalah rumus dari Metric Minkowski Distance :

$$(X,Y) = \left(\sum_{i=1}^{n} |x_I - y_i|^P\right)^{1/p} \quad (5)$$

d = jarak antara x dan y
x = pusat data cluster
y = atribut data
i = setiap data
n = jumlah data
xi = data di pusat cluster ke i
yi = data pada setiap data ke i
p = kekuatan

2.2.2 Manhattan Distance

Seiring dengan *metric Minkowski*, ada juga *metric Manhattan Distance. Manhattan Distance* merupakan pengukuran *similarity* yang merepresentasikan kasus yang relevant untuk approval project dengan angka kuantitatif yang bersifat natural [6].

$$d(x,y) = \left(\sum_{i=1}^{n} |x_I - y_i\right) \quad (6)$$

d = jarak antara x dan y
x = data pusat cluster
y = data pada atribut

2.2.3 Euclidean Distance

Euclidean Distance adalah ukuran satu titik dan yang lainnya satu garis lurus. Metode jarak ini adalah yang paling sering digunakan pada Machine learning. Berikut adalah Rumus dari Euclidean Distance [7].

$$d(x,y) = |x - y| = \sqrt{\sum_{i=1}^{n} (x_{i-y_i})^2} \quad (7)$$

d = jarak antara x dan y
x = pusat data cluster
y = atribut data
i = setiap data
n = nomor data
xi = data di pusat cluster ke i
yi = data setiap data ke i

2.2.4 Hamming Distance

Hamming distance is used to calculate the difference between two binaries that have the same length. Hamming distance is an approximate string matching algorithm invented by Richar Hamming, in 1950.

$$D_n = \frac{\sum_i^K |I_i - W_i|}{\sum_i^K |I_i - W_i|} \quad (8)$$

K = attributes number in each case
I = new case
W = the value of the proximity of the case in storage

2.2.5 Jaccard Distance

Jaccard Distance adalah algoritma berfungsi untuk membandingkan dua sample yaitu dokumen yang satu dengan yang lainnya dan menghitung seberapa besar kemiripan dengan dokumen lainnya [8]. Rumusnya adalah :

$$Similarity\ (X,Y) = |X \cap Y| |X \cup Y| \quad (9)$$

X = Dokumen 1
Y = Dokumen 2

2.2.6 Cosine Similarity

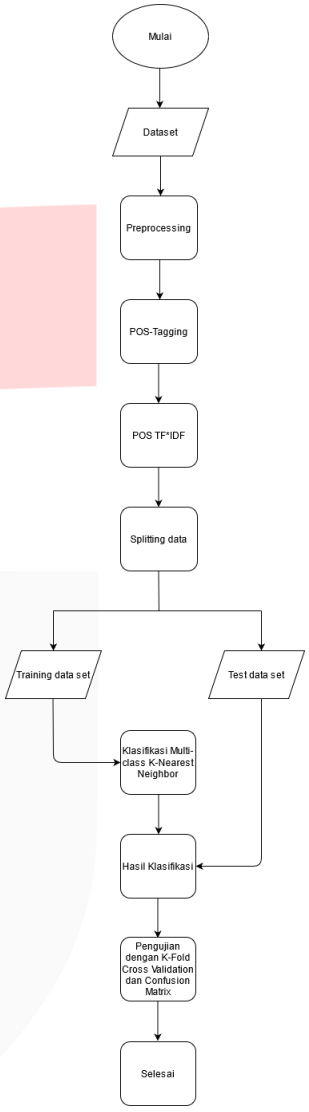
Metrik terakhir adalah Cosine Similarity, yang memiliki ruang dimensi di mana dua vektor adalah sama. Jika nilai cosine similarity adalah 1, maka vektor tersebut dapat dikatakan mirip [9].

$$Similarity = \cos(\theta) = \frac{A.B}{||A||\,||B||} \frac{\sum_{i-1}^n A_i x B_i}{\sqrt{\sum_{i-1}^n (A_i)^2}\, x \sqrt{\sum_{i-1}^n (B_i)^2}} \quad (10)$$

3. Perancangan dan Implementasi

3.1 Desain Umum Sistem

Dalam perancangan sistem, sistem ini berfungsi untuk mengetahui informasi akan bencana bencana alam khususnya banjir,gempa, dan longsor yang terjadi di wilayah Indonesia. Berikut alur sistem mulai dari proses awal sampai dengan proses akhir.



Gambar 3.1 Gambaran sistem yang akan dibuat

3.2 Data Crawling

Dalam implementasi crawling data dalam penelitian ini, menggunakan library TWINT (Twitter Intelligence Tool). Twint adalah library untuk python yang dapat mengambil data Tweet tanpa harus menggunakan Twitter API. Data tweet dengan kata kunci banjir,gempa dan longsor akan diambil untuk di training dari rentang waktu tahun 2013 sampai dengan tahun 2021. Contoh dataset

adalah sebagai berikut :

Tabel 3.1 Hasil Crawling Data

No	Data Tweet
1	Semoga cepat surut air banjirnya #jatibeningdua #jatibeningbaru #pondokgede #kotabekasi #banjir
2	Krasa bergoyang wilayah Purworejo #Gempa
3	Longsor tebing di Piket Nol lereng gunung Semeru bagian tenggara di tiga lokasi yaitu Km 56 Km 57 dan Km 58 menyebabkan arus kendaraan dari Malang ke Jember terganggu, Jumat (6/8).

Tabel 3.1 menjelaskan contoh hasil data crawling dataset menggunakan twint yang menampilkan data tweet bencana banjir, gempa, dan longsor. Data yang akan digunakan nantinya dibagi menjadi 4 kelas label. Banyak data yang dibagi per 4 kelas label tadi yaitu data banjir, data gempa, longsor, dan untuk kelas terakhir ‘’lainnya’’. Untuk data yang pertama yaitu kelas banjir berisi tweet yang usernya terkena dampak langsung bencana tersebut, begitu juga dengan kelas gempa dan longsor. Kelas yang terakhir yaitu ‘lainnya’ hanya berisi berita ataupun pendapat user tweet tersebut terhadap kejadian bencana dari ketiga kelas bencana sebelumnya.

3.3 Part of Speech - Tagging (Post-Tagging)

Merupakan proses di mana kata-kata dalam kalimat diurutkan berdasarkan pelabelan menurut kelas kata, seperti kata benda, kata kerja, kata sifat, dan lain-lain [10].

Tabel 3.2 Contoh data setelah proses Part Of Speech-Tagging

Input Data	Output Data
tahan lewat banjir jalan	[(‘tahan’, ‘NN’), (‘lewat’, ‘VB’), (‘banjir’, ‘NN’), (‘jalan’, ‘JJ’)]
camat serta satpol jambi saling korban gempa	[(‘camat’,’NN’), [(‘serta’, ‘JJ’), [(‘satpol’, ‘NN’), [(‘saling’, ‘JJ’), [(‘korban’, ‘NN’), [(‘gempa’, ‘NN’)
baru ini ada jatuh longsor desa	[(‘baru’,’OD’)), (‘ini’, ‘NND’), (‘ada’, ‘NN’), (‘jatuh’, ‘NN’), (‘longsor’, ‘NN’), (‘desa’, ‘JJ’)]

Pada Tabel 3.2 yaitu proses pemberian Part Of Speech –Tagging pada setiap kata dalam suatu kalimat bertujuan untuk mengetahui penting atau tidaknya sebuah kata dengan kata lainnya dalam suatu kalimat.

3.4 Klasifikasi

Dalam Klasifikasi Multi-Class K-Nearest Neighbor menggunakan parameter n_neighbors = 5, metric = 'jaccard'. Jika pada KNN memiliki nilai parameter n_neighbors yang digunakan besar maka hasil klasifikasi akan terasa kabur, tidak dapat dipastikan berapa nilai n_neighbors sehingga harus dicoba untuk mencari hasil yang maksimal.

3.5 Evaluasi

Dalam proses ini klasifikasi yang sudah ada dilihat performanya dengan menggunakan K-Fold Cross Validation dan Confusion Matrix. Merupakan evaluasi untuk mendapatkan keakurat suatu model dengan membagi data yang ada. K = 10. Kemudian menghitung confusion matrix dengan nilai True Positive, True Negative, False Positive, False Negative.

4. Implementasi dan Pengujian

4.1 Dataset

Dataset yang akan digunakan berasal dari tweet pengguna dan dibagi menjadi 4 kategori kelas yaitu banjir, gempa bumi, tanah longsor, dan lain-lain. Data kelas banjir, gempa bumi dan longsor merupakan data yang memiliki tweet berupa dampak langsung dari bencana yang sedang

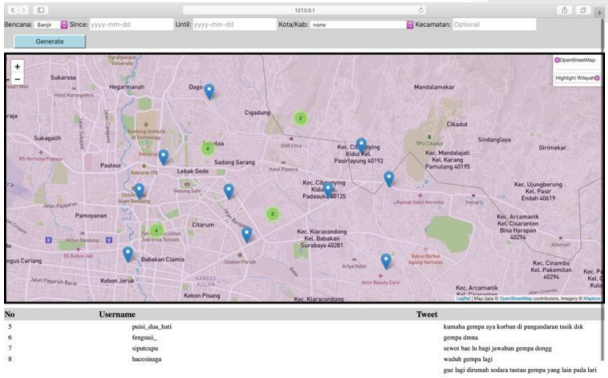
berlangsung pada pengguna, sedangkan untuk data lainnya merupakan data yang hanya berisi tanggapan atau komentar dari pengguna yang tidak terkena dampak langsung bencana tersebut kepada pengguna. tiga kelas bencana sebelumnya. Satu per satu data tweet dikategorikan oleh penulis dan seluruh data diambil dari Twitter. Proporsi datanya adalah 968 data banjir, 799 data gempa bumi dan 843 data longsor, untuk kelas terakhir "lain-lain" yaitu 1230 data.

4.2 Implementasi Desain Antarmuka

Berikut ini adalah gambar tampilan dari implementasi desain antarmuka sistem informasi bencana alam dari data media sosial.

Gambar 4. 1 Tampilan halaman utama Sistem Web Aplikasi Bencana alam

Pada Gambar 4.1 pengguna harus memilih bencana yang ingin diketahui, yaitu diantara salah satu antara tiga pilihan bencana alam. Kemudian user dapat menginput jangka waktu tanggal bencana yang diinginkan. Setelah itu user memilih kota atau kabupaten yang ingin diketahui. Input kecamatan bersifat opsional atau tidak wajib diisi. Setelah melakukan semua langkah-langkah tersebut, kemudian user memilih generate untuk melihat hasil data tweet bencana beserta lokasinya berdasarkan acuan jenis bencana, jangka waktu, dan kota yang sudah dipilih sebelumnya.



Gambar 4. 2 Tampilan hasil pemetaan geolokasi data tweet pengguna

Pada Gambar 4.2 menunjukkan tampilan hasil pemetaan geolokasi data tweet pengguna, aplikasi web akan memetakan data tweet pengguna sesuai dengan geolokasi pengguna tersebut dan hanya jika pengguna mengaktifkan geolokasinya, tweet akan muncul ke wilayah yang dipilih ketika pengguna tidak mengaktifkan geolokasinya.

4.3 Pengujian Alpha

Pengujian Alpha untuk mengetahui bahwa sistem informasi monitoring bencana alam dari data media sosial dapat berjalan dengan baik, pengujian alpha dilakukan agar dapat mengetahui fitur web aplikasi berjalan sesuai dengan seharusnya atau tidak. Berikut merupakan skenario pengujian alpha pada tabel pada tabel 4.1.

Table 4.1 Tabel Skenario Pengujian Alpha

Pada tabel 4.1 ada 4 fitur yang pengujian alpha dilakukan secara black box merupakan pengujian untuk melihat berjalannya suatu fitur atau sarana dari suatu web aplikasi.

Hasil pengujian alpha dari web aplikasi sistem monitoring bencana alam dengan data media sosial terdapat pada tabel di bawah ini :

Tabel 4. 2 Hasil pengujian membuka aplikasi

Data masukan	Hasil yang diharapkan	Hasil Pengamatan	Kesimpulan
Membuka Web Aplikasi	Menampilkan halaman utama	Halaman utama berhasil tampil	Berhasil

Pada Tabel 4.2 Menunjukkan pengujian web aplikasi yang menyatakan berhasil untuk menampilkan halaman utama serta dapat berjalan sesuai ekspetasi.

Tabel 4.3 Hasil uji Menu pilihan bencana alam

Data masukan	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Klik menu pilihan “bencana”	Menampilkan 3 bencana alam banjir, gempa, longsor	3 bencana berhasil ditampilkan yaitu banjir,gempa,longsor	Berhasil

Tabel 4.3 menunjukan bahwa pengujian tombol pilihan bencana dan menyatakan berhasil sesuai dengan hasil yang diharapkan.

Tabel 4. 4 Hasil pengujian pilihan kota atau kabupaten

Data masukan	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Klik menu pilihan “Kota atau Kabupaten”	Menampilkan pilihan nama-nama kota/kab di Indonesia	Nama-nama Kota atau Kabupaten di Indonesia berhasil ditampilkan	Berhasil

Tabel 4.4 dapat menunjukkan untuk

No	Fitur yang Diuji	Detail Pengujian	Jenis Pengujian
1	Buka Web Aplikasi	Untuk dapat memuat halaman utama	*Black Box*
2	Menu pilihan bencana alam	Menampilkan 3 bencana alam yaitu bencana alam yaitu banjir, gempa, longsor	*Black Box*
3	Menu pilihan kota atau kabupaten	Menampilkan pilihan kota/kabupaten di indonesia	*Black Box*
4	Kasifikasi	Menampilkan hasil tweet yang sudah diklasifikasikan dan memetakan data tweet	*Black Box*

pengujian tombol menu pilihan Kota atau

Kabupaten pada web aplikasi berhasil dan dapat menampilkan nama-nama kota atau kabupaten di Indonesia.

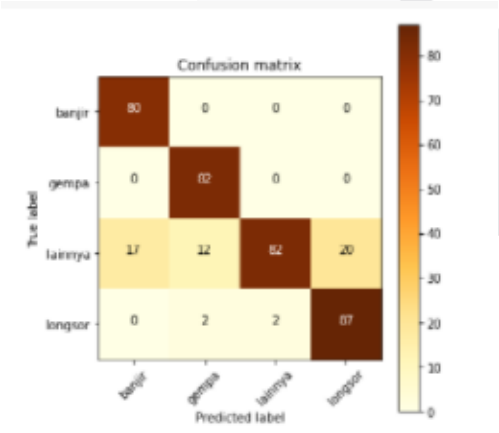
Tabel 4. 5 hasil pengujian tombol klasifikasi

Pada Tabel 4.5 pengujian fungsi tombol klasifikasi menunjukkan keberhasilan karena dapat mengklasifikasikan data tweet dari twitter, memetakan di map, dan memunculkan tweet di tabel.

Data masukan	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Klik Tombol "Generate"	Mengklasifikasikan data tweet twitter serta memetakan di map lalu memunculkan tweet di tabel	Berhasil Mengklasifikasikan data tweet dari twitter dan memetakan di map lalu memunculkan tweet di tabel	Berhasil

4.4 Pengujian Confusion Matrix dan K-Fold Cross Validation

Dalam algoritma K-Nearest Neighbor, matriks konfusi diuji menggunakan parameter n_neighbors = 5 untuk setiap metrik. Dimulai dari metric Manhattan, kemudian Minkowski, Euclidean, Cosine, Hamming, dan yang terakhir adalah Jaccard. Setelah itu dilakukan uji K-fold Cross Validation untuk setiap metrik untuk mengetahui mana yang memiliki akurasi terbaik dan hasilnya sebagai berikut :



Gambar 4.3 Kinerja KNN metric Jaccard dengan pengujian confusion matrix

	precision	recall	f1-score	suppor
banjir	0.82	1.00	0.90	8
gempa	0.85	1.00	0.92	8
lainnya	0.98	0.63	0.76	13
longsor	0.81	0.96	0.88	9
accuracy			0.86	38
macro avg	0.87	0.90	0.87	38
weighted avg	0.88	0.86	0.85	38

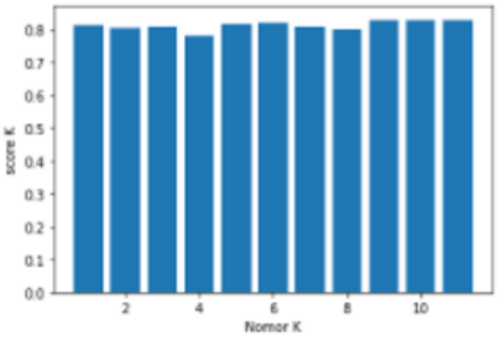
Gambar 4.4 Hasil Akurasi agloritma knn metric Jaccard dengan confusion matrix

Didapat hasil akurasi confusion matrix dengan nilai akurasi sebesar 86% pada metric Jaccard.

Pengujian Cross validation dilakukan dengan 10 fold. Langkahnya adalah fold 1 dibuat menjadi data test, kemudian sisa yang lainnya menjadi data training. Kemudian data training adalah fold 2 yang sebelumnya menjadi data test dan pindah ke tahap kedua, seterusnya hingga mencapai fold 10.

```
Scores for each fold 1: 0.8142857142857143
Scores for each fold 2: 0.8051575931232091
Scores for each fold 3: 0.8080229226361032
Scores for each fold 4: 0.7822349570200573
Scores for each fold 5: 0.8223495702005731
Scores for each fold 6: 0.8080229226361032
Scores for each fold 7: 0.8022922636103151
Scores for each fold 8: 0.830945558739255
Scores for each fold 9: 0.828080229226361
Scores for each fold 10: 0.828080229226361
```

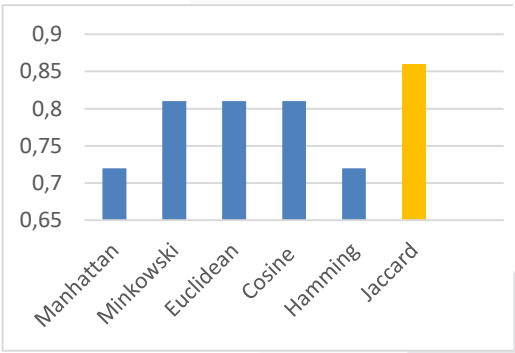
Gambar 4.5 Hasil score setiap fold metric jaccard



Gambar 4.6 Grafik score hasil K-Fold Cross Validation metric Jaccard

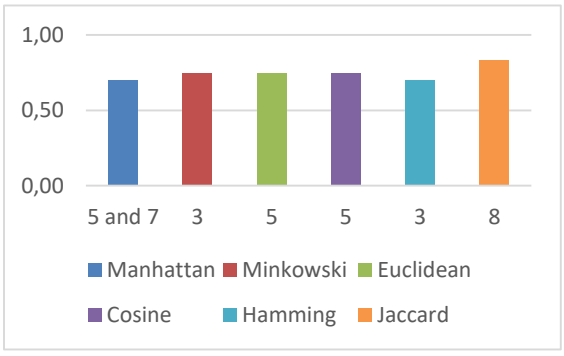
Hasil Grafik K-Fold Cross Validation Jaccard menunjukkan nilai fold pada fold 8 memliki nilai fold yang tertinggi yaitu 83 %.

Tabel 4.6 Perbandingan Hasil Confusion Matrix terhadap setiap metric



Dapat dikatakan pada hasil keseluruhan yang sudah ada pada Pengujian Confusion Matrix menyatakan bahwa penilaian performa terbaik diraih oleh metric Jaccard sebesar 86%.

Tabel 4.8 Perbandingan Hasil K-Fold Cross Validation setiap metric



Untuk hasil pengujian untuk uji K-Fold Cross alidation menyatakan bahwa penilaian kinerja dan curasi untuk sampel data dicapai oleh metrik Jaccard besar 83% di fold 8.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan seluruh pengujian aplikasi ini, fitur-fitur yang ada pada aplikasi dapat berjalan sesuai harapan, dan dapat ditarik kesimpulan yaitu :

1. Web Aplikasi Sistem informasi monitoring bencana alam menggunakan data media sosial dengan metode K-Nearest Neighbor berhasil mengklasifikasikan data tweet yang terbagi menjadi 4 kelas yaitu banjir, gempa, dan longsor.
2. Web Aplikasi dapat menampilkan data hasil klasifikasi dan menampilkan geolocation user tweet.
3. Nilai pengujian Confusion matrix menyatakan akurasi yang paling baik dengan adalah dengan metric Jaccard sebesar 86%.
4. Untuk Performa pembagian data dilakukan menggunakan k-fold cross validation 10 Fold didapatkan akurasi terbaik adalah metric Jaccard yaitu sebesar 83% pada fold 8

5.2 Saran

Ada beberapa saran berdasarkan hasil dari penelitan yang dapat dipertimbangkan untuk pengembangan penelitian selanjutnya yaitu sebagai berikut:

1. Aplikasi dapat dicoba menggunakan algoritma lain untuk mengetahui dan meningkatkan performanya.
2. Membuat dataset tweet bencana selain yang ada dalam penelitian ini

3. Menambahkan fitur tracking bencana secara live agar lebih akurat
4. Menambah menu pilihan pada halaman web aplikasi

REFERENSI

[1] Datareportal.com,[Online].Available:(https://datareportal.com/reports/digital-2021-indonesia, [Diakses pada 8 Agustus 2021].

[2] Saurav Pradha, Malka N. Halgamuge, Nguyen Tran Quoc Vinh, "Effective Text Data Preprocessing Technique for Sentiment Analysis in Social Media Data", 2019 11th International Conference on Knowledge and Systems Engineering (KSE).

[3] D. D. Mehare, "Introduction to TF-IDF: To Represent Importance of Keyword within whole Dataset," Int. J. Res. Appl. Sci. Eng. Technol., vol. 6, no. 3, pp. 2321–2323, 2018.

[4] S. B. B. Choi, "The real-time monitoring system of social big data for disaster management," dalam Computer science and its applications, Berlin, 2015.

[5] Zonghu Wang, Zhijing Liu, 2010, "Graph-based KNN text classification" , 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery.

[6] Muhammad K. Farooq, Malik Jahan Khan, Shafay Shamail and Mian M. Awais, Intelligent Project Approval Cycle for Local Government – Case- Based Reasoning Approach, CEGOV2009, November 10-13, Bogota, Colombia, ACM 2009.

[7] M. Nishom, "Euclidean Distance Accuracy Comparison, Minkowski Distance, and Manhattan Distance on the KMeans . Algorithm Chi-Square .based Clustering", 2019.

[8] Komang R. Simple Query Suggestion untuk Pencarian Artikel menggunakan Jaccard Similarity. Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi. 2017; 3(1): 30-34.

[9] Dewa Ayu Rai Ariantini, Arie S. M. Lumenta, Agustinus Jacobus, "Pengukuran Kemiripan Dokumen Teks Bahasa Indonesia Menggunakan Metode Cosine Similarity", 2016

[10] Agus Mulyanto, Yeni Agus Nurhuda, Nova Wiyanto, "Solving Ambiguous Words in the POS Tagging Process Using the Hidden Markov Model (HMM) Algorithm", 2017.