**SKRIPSI**

**KLASTERISASI DAN GEOVISUALISASI *TWEET* PENYEBARAN PENYAKIT MENULAR LANGSUNG (STUDI KASUS COVID-19)**

****

**FAHMIRULLAH ABDILLAH**

**PROGRAM STUDI S1 SISTEM INFORMASI**

**DEPARTEMEN MATEMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS AIRLANGGA**

**2022**

**SKRIPSI**

**KLASTERISASI DAN GEOVISUALISASI *TWEET* PENYEBARAN PENYAKIT MENULAR LANGSUNG (STUDI KASUS COVID-19)**

****

**FAHMIRULLAH ABDILLAH**

**081811633002**

**PROGRAM STUDI S1 SISTEM INFORMASI**

**DEPARTEMEN MATEMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS AIRLANGGA**

**2022**

# KATA PENGANTAR

Puji syukur penulis ucapkan atas kehadirat Tuhan Yang Maha Esa karena telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan penyusunan proposal skripsi yang berjudul **Klasterisasi dan Geovisualisasi *Tweet* Penyebaran Penyakit Menular Langsung (Studi Kasus COVID-19)**.

Ucapan terima kasih dari penulis kepada seluruh pihak yang telah membantu dan mendukung dalam pengerjaan proposal skripsi ini hingga dapat terselesaikan dengan baik. Proposal skripsi ini tidak akan terselesaikan tanpa bantuan dan dukungan dari pihak yang terlibat.

Penulis menyadari bahwa dalam penyusunan proposal skripsi ini terdapat banyak kekurangan, maka dari itu penulis senantiasa terbuka dalam menerima kritik dan saran atas kekurangan dan kesalahan yang ada dalam penelitian ini. Harapan dari penulis, semoga penelitian ini dapat memberikan manfaat dan wawasan yang berguna dan menjadi sumber ilmu yang bermanfaat kedepannya.

|  |
| --- |
| Surabaya, 27 Juli 2022 |
| Penulis |
|  |
| Fahmirullah Abdillah |

# DAFTAR ISI

[KATA PENGANTAR 3](#_Toc115957888)

[DAFTAR ISI i](#_Toc115957889)

[DAFTAR GAMBAR iii](#_Toc115957890)

[DAFTAR TABEL iv](#_Toc115957891)

[BAB 1 1](#_Toc115957892)

[PENDAHULUAN 1](#_Toc115957893)

[1.1 Latar Belakang 1](#_Toc115957894)

[1.2 Rumusan Masalah 4](#_Toc115957895)

[1.3 Tujuan Penelitian 4](#_Toc115957896)

[1.4 Manfaat Penelitian 4](#_Toc115957897)

[1.5 Batasan Masalah 5](#_Toc115957898)

[BAB 2 6](#_Toc115957899)

[TINJAUAN PUSTAKA 6](#_Toc115957900)

[2.1 Twitter API 6](#_Toc115957901)

[2.2 Data Mining 6](#_Toc115957902)

[2.3 Pra-proses Data 7](#_Toc115957903)

[2.3.1](#_Toc115957904) *[Case Folding](#_Toc115957904)* [8](#_Toc115957904)

[2.3.2 Tokenizing 8](#_Toc115957905)

[2.3.3 Penghapusan](#_Toc115957908) *[Stopword](#_Toc115957908)* [8](#_Toc115957908)

[2.3.4](#_Toc115957909) *[Stemming](#_Toc115957909)* [Nazief-Adriani 9](#_Toc115957909)

[2.3.5 Term Document Matrix 11](#_Toc115957910)

[2.3.6 Algoritma TF-IDF 11](#_Toc115957911)

[2.4 Klasterisasi 12](#_Toc115957912)

[2.4.1](#_Toc115957913) *[Cosine Similarity](#_Toc115957913)* [13](#_Toc115957913)

[2.4.2 Algoritma DBSCAN 14](#_Toc115957914)

[2.4.3 Algoritma OPTICS 15](#_Toc115957915)

[2.4.4 Uji Validasi 16](#_Toc115957916)

[2.5 Geovisualisasi 17](#_Toc115957917)

[2.5.1 Sistem Informasi Geografis 17](#_Toc115957918)

[2.5.2](#_Toc115957919) *[Unified Modelling Language](#_Toc115957919)* [(UML) 17](#_Toc115957919)

[2.6 Penelitian Sebelumnya Tentang Penyebaran Informasi Suatu Kejadian Menggunakan Twitter 20](#_Toc115957920)

[BAB 3 23](#_Toc115957921)

[METODE PENELITIAN 23](#_Toc115957922)

[3.1 Waktu Penelitian 23](#_Toc115957923)

[3.2 Objek Penelitian 23](#_Toc115957924)

[3.3 Tahapan Penelitian 24](#_Toc115957925)

[3.3.1 Akuisisi](#_Toc115957926) *[Tweet](#_Toc115957926)* [25](#_Toc115957926)

[3.3.2 Praproses Data 26](#_Toc115957927)

[3.3.3 Klasterisasi dengan DBSCAN 29](#_Toc115957928)

[3.3.4 Klasterisasi dengan OPTICS 31](#_Toc115957929)

[3.3.5 Perancangan Sistem 33](#_Toc115957930)

[3.3.6 Implementasi Sistem 33](#_Toc115957931)

[3.3.7 Geovisualisasi 33](#_Toc115957932)

[3.3.8 Evaluasi Hasil Analisis 34](#_Toc115957933)

[3.3.9 Jadwal Penelitian 35](#_Toc115957934)

[DAFTAR PUSTAKA 0](#_Toc115957935)

# DAFTAR GAMBAR

[Gambar 3.1 flowchart DBSCAN 30](C:\\Users\\fahmy\\OneDrive\\Documents\\#SKRIPSI\\proposal skripsi\\berkas sidang proposal\\Proposal Skripsi_081811633002_Fahmirullah Abdillah.docx" \l "_Toc115792834)

[Gambar 3.2 flowchart OPTICS 32](C:\\Users\\fahmy\\OneDrive\\Documents\\#SKRIPSI\\proposal skripsi\\berkas sidang proposal\\Proposal Skripsi_081811633002_Fahmirullah Abdillah.docx" \l "_Toc115792835)

[Gambar 3.3 Peta Interaktif Plot.ly 34](#_Toc115792836)

# DAFTAR TABEL

[Tabel 2.1 Kombinasi Awalan yang Tidak Diizinkan 10](#_Toc108684906)

[Tabel 2.2 Metode Klasterisasi (Han et al. 2012) 12](#_Toc108684907)

[Tabel 2.3 Komponen use case diagram 18](#_Toc108684908)

[Tabel 3.4 Komponen activity diagram 20](#_Toc108684909)

[Tabel 3.1 Kata Kunci yang digunakan untuk pencarian tweet 24](#_Toc108684910)

[Tabel 3.2 Contoh Tokenizing 27](#_Toc108684911)

[Tabel 3.3 Contoh stopwords removal 28](#_Toc108684912)

[Tabel 3.4 Contoh stemming 28](#_Toc108684913)

# 

# PENDAHULUAN

## Latar Belakang

Layanan media sosial mikroblog seperti Twitter menghasilkan aliran besar dalam penyebaran informasi terhadap suatu kejadian. Sumber informasi *realtime* ini sangat berharga untuk banyak area aplikasi, khususnya untuk deteksi bencana dan skenario respons. Terbukti dengan aliran volume maupun kecepatan *tweet* saat kejadian berlangsung sangat tinggi dan cepat, sehingga masyarakat yang terdampak maupun petugas profesional sedikit mengalami kesulitan saat pemrosesan informasi (Imran et al., 2013).

Sakaki *et al.* 2013 menuturkan bahwa, melalui pemantauan *tweet* dapat dideteksi adanya gempa bumi. Probabilitas yang dihasilkan oleh *Japan Meteorology Agency* cukup tinggi, yaitu 96% untuk gempa bumi dengan *skala richter 3* atau lebih. Situs mikroblog ini dapat digunakan sebagai sistem sensor untuk mendeteksi suatu bencana alam atau kejadian lainnya (Crooks *et al.* 2013).

Beberapa penelitian yang menggunakan data dari media sosial Twitter telah dilakukan sebelumnya. Dwiarni (2019) melakukan penelitian tentang akuisisi dan klasterisasi data teks Twitter untuk memperoleh dasar pengetahuan terhadap profil pengguna Twitter. Penelitian dilakukan dengan ujicoba *keyword* “K-Pop” dan “K-Drama”. Dari hasil ujicoba akuisisi data didapatkan sebanyak 68.393 *tweet.* Hasil tersebut disebar menjadi 3 klaster / k=3, yang mana klaster pertama adalah waktu *tweet* dianggap pada pagi hari, klaster kedua adalah waktu *tweet* dianggap pada siang hari, dan klaster ketiga adalah waktu *tweet* dianggap pada malam hari. Kemudian, hasil klasterisasi didapat jam 21.00 - 01.00 merupakan mayoritas orang-orang melakukan *tweet*. Dari hasil penelitian ini kita dapatkan bahwa penentuan nilai *k* untuk memperkirakan topik suatu klaster didasarkan pada asumsi kebiasaan pengguna dalam menggunakan media sosial Twitter.

Penelitian lainnya tentang kemungkinan analisis secara *realtime* pada media sosial dan otomatis dari pesan Twitter selama terjadinya situasi darurat dikemukakan oleh Terpstra *et al.* (2012). Analisis dilakukan menggunakan *tool* ekstraksi informasi yang berhasil mendapatkan 97.000 *tweet* yang dikirim sebelum, saat, dan setelah kejadian alam (badai) terjadi. Lokasi kejadian adalah di Belgia saat berlangsungnya festival Pukkelpop di tahun 2011. *Tool* ekstraksi dapat menganalisis *tweet* melalui tampilan geografis, jenis isi pesan (kerusakan, korban), dan jenis *tweet* (seperti *retweet*).

Penyakit menular langsung merupakan suatu infeksi yang disebabkan oleh mikroorganisme, seperti virus, parasit, atau jamur. Infeksi ini dapat berpindah dari orang yang sakit ke orang yang sehat. Bentuk penularannya bisa terjadi secara langsung maupun tidak langsung, penularan secara langsung terjadi ketika benda tak kasat mata di atas pada orang yang sakit berpindah melalui kontak fisik, misalnya lewat sentuhan (Alodokter, 2018).

Saat ini penyakit menular langsung telah menjadi wabah yakni virus Covid-19. Wabah yang terjadi secara mendunia ini diberi nama *Coronavirus Disease* 2019 (Covid-19) yang disebabkan oleh *Severe Acute Respiratory Syndrome Coronavirus-2* (SARS-CoV-2). Penyebaran penyakit menular langsung ini hingga ke seluruh penjuru nusantara dan dunia. Menurut Susilo (2020), virus ini dapat ditularkan dari manusia ke manusia dan telah menyebar secara luas di China (sebagai tempat kemunculan pertama) dan lebih dari 190 negara dan teritori lainnya. Pada 12 Maret 2020, WHO mengumumkan COVID-19 sebagai pandemi. Hingga tanggal 29 Maret 2020, terdapat 634.835 kasus dan 33.106 jumlah kematian di seluruh dunia. Sementara di Indonesia sudah ditetapkan 1.528 kasus dengan positif COVID-19 dan 136 kasus kematian. Per tanggal 20 Desember 2020, Satgas Covid-19 menerbitkan laporan yang berisi informasi kasus terkonfirmasi positif, sembuh, ataupun meninggal. Sebanyak 735.124 kasus terkonfirmasi positif dan 19.880 (2,99%) jumlah kematian di Indonesia, serta jumlah kasus sembuh 541.811 (81,48%).

Metode *density-based spatial cluster of application with noise* (DBSCAN) merupakan salah satu metode *cluster* mengacu pada densitas atau kepadatan. Kepadatan yang dimaksudkan yaitu dalam metode DBSCAN mengelompokkan wilayah dengan jarak yang telah ditentukkan menggunakan nilai parameter Epsilon dan MinPts, sehingga dihasilkan suatu kelompok yang padat dengan jarak antar anggota kelompok yang beragam. Parameter Epsilon merupakan jarak maksimal antar titik pusat dengan titik anggota dalam suatu cluster. Sedangkan MinPts merupakan minimal anggota yang harus terpenuhi dalam sebuah klaster. Apabila kedua parameter tersebut telah terpenuhi, maka akan terbentuklah suatu klaster (Daszykowski & Walczak, 2009).

Analisis *cluster* merupakan teknik multivariat dalam analisis statistik yang dapat mengumpulkan objek-objek dengan karakteristik sama pada suatu kelompok yang lebih kecil. Pada penelitian ini metode klasterisasi *tweet* yang digunakan adalah algoritma DBSCAN dan OPTICS. Metode-metode ini dipilih dan dibandingkan karena keduanya dapat menghasilkan *cluster* tanpa penentuan *centroids* dan juga dapat menemukan titik-titik yang menyimpang. Data hasil klasterisasi divisualisasikan untuk menerapkan geovisualisasi *tweet* untuk kasus penyebaran penyakit menular langsung (studi kasus Covid-19). Proses geovisualisasi digunakan untuk mendapatkan hasil tampilan data *tweet* hasil klasterisasi dan lokasi penyebaran *tweet* terkait penyebaran penyakit menular langsung (studi kasus Covid-19). Pengujian dilakukan dengan mengevaluasi hasil analisis klasterisasi menggunakan analisis *silhouette* yang berdasarkan nilai koefisien *silhouette.*

## Rumusan Masalah

Rumusan masalah pada penelitian ini adalah sebagai berikut:

1. Bagaimana menerapkan algoritma klasterisasi DBSCAN dan OPTICS untuk mengolah data *tweet*?
2. Bagaimana perbandingan algoritma klasterisasi DBSCAN dan OPTICS agar menghasilkan analisis terbaik?
3. Bagaimana menerapkan geovisualisasi hasil klasterisasi data *tweet* untuk kasus penyebaran penyakit menular langsung (studi kasus Covid-19)?

## Tujuan Penelitian

Tujuan penelitian ini adalah:

1. Menerapkan algoritma klasterisasi DBSCAN dan OPTICS untuk mengolah data *tweet.*
2. Mengetahui hasil perbandingan terbaik antara dua algoritma klasterisasi yang digunakan, yaitu DBSCAN dan OPTICS.
3. Menerapkan geovisualisasi hasil klasterisasi data *tweet* untuk kasus penyebaran penyakit menular langsung (studi kasus Covid-19).

## Manfaat Penelitian

Adapun manfaat penelitian ini adalah:

1. Memberikan wawasan kepada masyarakat awam, peneliti, dan pemerintah terkait data penyebaran penyakit menular langsung (Covid-19).
2. Membantu pemerintah dalam memantau keluhan gejala yang dialami masyarakat saat penyebaran penyakit menular langsung dalam waktu tertentu*.*
3. Membantu pemerintah dalam mengambil keputusan untuk menindaklanjuti kebijakan berdasarkan data.
4. Membantu pemerintah dalam memutuskan daerah mana saja yang perlu diantisipasi penanggulangan dan pencegahan dini pada penyakit menular langsung.

## Batasan Masalah

Batasan penelitian ini adalah:

1. Penelitian ini menggunakan data teks dari media sosial Twitter dengan kata kunci tentang penyebaran penyakit menular langsung (studi kasus Covid-19).
2. Data teks yang digunakan yaitu *tweet* berbahasa Indonesia.
3. Data yang digunakan dalam klasterisasi adalah data yang dikumpulkan sejak April 2021 – September 2021 dan Januari 2022 – Maret 2022.
4. Output dari algoritma klasterisasi DBSCAN dan OPTICS yang ditampilkan adalah sistem informasi geografis yang menampilkan sebaran data di wilayah Indonesia.
5. Target pengguna dari penelitian ini adalah masyarakat awam, peneliti, atau pemerintah yang ingin mengetahui sebaran data masyarakat terhadap pandemi Covid-19 saat varian delta terjadi.

# 

# TINJAUAN PUSTAKA

## Twitter API

*Application Programming Interface* merupakan interaksi *online* yang melibatkan komponen perangkat lunak. API sudah banyak digunakan, mulai dari *command-line tool*, aplikasi *enterprise*, hingga aplikasi web. Twitter API merupakan API JavaScript Object Notation (JSON) berbasis web yang dapat digunakan pengembang untuk berinteraksi dengan data Twitter melalui suatu program. Twitter API harus diakses dengan cara membuat request ke layanan yang disediakan oleh Twitter melalui internet. Dengan API berbasis web, seperti Twitter API, aplikasi akan mengirim request HyperText Transfer Protocol (HTTP), sama seperti web browser, namun response tidak ditampilkan sebagai halaman web, melainkan dengan format yang dapat dipisahkan dengan mudah oleh aplikasi. Response memiliki format yang bermacam-macam. Twitter menggunakan format yang terkenal dan mudah digunakan yaitu JSON. Salah satu bagian dasar dari Twitter adalah tweet. Twitter API dapat digunakan untuk melakukan pencarian tweet, membuat tweet, dan menandai tweet yang disukai. Ketika akan melakukan pencarian tweet, diperlukan untuk memasukkan kriteria, seperti kata kunci atau hashtag, geolokasi, bahasa, dan lain-lain (Freeman, 2018).

Twitter API merupakan contoh dari REST API, yaitu API yang menggunakan gaya arsitektur Representational State Transfer (REST). REST adalah gaya dalam mengembangkan sistem yang dapat melakukan komunikasi yang fleksibel dan menampilkan informasi lintas web dengan menyediakan struktur yang diperlukan untuk mengembangkan komponen yang memiliki tujuan umum secara mudah.

## Data Mining

*Data mining* adalah proses penemuan pola dan pengetahuan dari kumpulan data dengan jumlah yang besar. Sumber data meliputi basis data, data warehouse, website, penyimpanan informasi lainnya, atau data streaming yang digunakan oleh suatu sistem secara dinamis (Han *et al.* 2012). Sedangkan menurut Baumgartner *et al.* (1996), *Data mining* adalah langkah analisis terhadap proses penemuan pengetahuan di dalam basis data atau *Knowledge Discovery in Databases* (KDD). Pengetahuan dapat berupa pola data yang valid atau hubungan antar data (tidak diketahui sebelumnya). *Data mining* adalah kombinasi dari banyak disiplin ilmu komputer. Disiplin ini didefinisikan sebagai proses menemukan pola baru dari kumpulan data yang sangat besar, termasuk metode seperti kecerdasan buatan, pembelajaran mesin, statistik, dan sistem basis data*.*

Sedangkan Chakrabarti *et al.* (2006) menjelaskan bahwa, *Data mining* digunakan untuk mengekstrak (mengambil intisari) pengetahuan dari sekumpulan data sehingga didapatkan struktur yang dapat dimengerti manusia serta melibatkan basis data dan manajemen data, prapemrosesan data, pertimbangan model dan inferensi ukuran ketertarikan, pertimbangan kompleksitas, pasca-pemrosesan terhadap struktur yang ditemukan, visualisasi, dan pembaruan secara *online.*

## Pra-proses Data

*Text mining* adalah salah satu teknik yang dapat digunakan untuk melakukan klasifikasi di mana, penambangan teks adalah variasi penambangan data yang mencoba menemukan pola menarik dari banyak koleksi data tekstual. Menurut Liao *et al. (*2012), penambangan teks mirip dengan penambangan data, kecuali untuk teknik penambangan data yang dirancang untuk mengerjakan data terstruktur dalam database, tetapi penambangan teks dapat mengerjakan data yang tidak terstruktur atau semi terstruktur seperti dokumen teks lengkap, halaman web kode/skrip, dan lainnya. Terdapat 5 langkah dalam praproses data yaitu *tokenizing,* normalisasi kata, penghapusan *stopword*, *stemming,* dan pembuatan *Term Document Matrix* (TDM).

Secara umum, tahapan utama dalam penambangan teks terdiri dari tiga bagian utama yaitu pra-pemrosesan teks, pemilihan fitur, dan analitik teks. Pada tahapan praprosesi teks secara umum adalah tokenisasi, pemfilteran, stemming, penandaan, dan analisis. Tokenisasi adalah langkah untuk memisahkan setiap kata (token) dalam dokumen input. Pemfilteran adalah proses pemilihan untuk kata-kata yang dihasilkan dari proses tokenisasi, dapat dilakukan dengan daftar berhenti atau algoritma daftar kata. Algoritma stop list akan membuang kata-kata yang tidak penting seperti kata ganti, kata keterangan, konjungsi, preposisi, dan pakaian. Sebaliknya, algoritma daftar kata akan menyimpan kata-kata penting

### *Case Folding*

*Case Folding* adalah proses mengubah semua karakter huruf pada sebuah kalimat menjadi huruf kecil dan menghilangkan karakter yang dianggap tidak valid seperti angka, tanda baca, dan *Uniform Resource Locator* (URL) (Jumadi et al., 2021). Contoh teks “Pengumuman”, “PENGUMUMAN”, “Pengumuman.com” atau “pengumuman” akan tetap dibaca sama, yaitu “pengumuman”.

### Tokenizing

*Tokenizing* adalah proses pemotongan kumpulan teks dalam dokumen input serta dilakukan pembuangan karakter-karakter tertentu, seperti tanda baca.

*Token* juga dapat disebut sebagai *term* atau kata, namun terkadang perlu dibedakan antara *type/token. Token* adalah kumpulan beberapa karakter pada suatu dokumen, sedangkan *type* merupakan kelas dari semua *token* yang memiliki urutan karakter yang sama. Menurut Manning *et al.* (2009), *Term* merupakan *type* yang termasuk ke dalam kamus sistem temu kembali informasi.

### Penghapusan *Stopword*

Lo *et al.* (2005) menjelaskan bahwa *stopword* adalah kata yang terdapat pada sebuah dokumen yang sering muncul, namun tidak memiliki nilai informasi kata. Beliau juga menjelaskan bahwa banyak yang beranggapan pada *stopword* ini tidak memiliki peran terhadap konteks atau informasi dokumen dan *stopword* harus dihapus sebelum dilakukan proses pada sistem temu kembali informasi, meskipun menggunakan daftar *stopword* tunggal dari berbagai kumpulan dokumen dapat merugikan efektifitas proses pengambilan informasi.

Penghapusan *stopword* secara komprehensif dapat mengurangi jumlah kata yang harus disimpan oleh sistem (Manning *et al.* 2009). Contoh s*topword* dalam BahasaIndonesia diantaranya yaitu dahulu, ada, dalam, adanya, dan lain-lain.

### *Stemming* Nazief-Adriani

*Stemming* adalah proses penghapusan imbuhan kata untuk mendapatkan kata dasar. Teknik ini sering digunakan dalam penelitian *text mining.* Contoh kata tulisnya, tulisannyakah, dan dituliskannya memiliki kata dasar yang sama yaitu tulis. Teknik ini mengurangi kompleksitas teks tanpa memengaruhi nilai informasi (Jumadi et al., 2021)

Algoritme *stemming* salah satunya adalah Nazief-Adriani. Algoritme ini dikembangkan menggunakan pendekatan pencocokan *term* dengan pencarian kamus.

Algoritma Nazief-Adriani memiliki langkah-langkah sebagai berikut:

1. Kata yang akan dilakukan *stemming* dicari dalam kamus. Jika ditemukan, maka akan dianggap kata tersebut adalah kata dasar dan algoritme berhenti. Jika tidak ditemukan maka lanjut ke langkah 2.
2. Menghilangkan imbuhan infleksi/*inflectional suffixes* (“-lah”, “-kah”, “-pun”), kemudian *possessive pronoun* (“-ku”, “-mu”, dan “-nya”). Kata dicari dalam kamus, jika ditemukan, algoritma berhenti. Jika tidak ditemukan, maka lanjut ke langkah 3.
3. Menghilangkan imbuhan derivasi/*derivation suffixes* (“-an”, “-i”, dan “-kan”). Jika akhiran “-an” dihapus dan ditemukan akhiran “-k”, maka akhiran “-k” dihapus. Jika ditemukan, algoritma berhenti. Jika tidak ditemukan, maka lanjut ke langkah 4.
4. Langkah 4 memiliki 3 iterasi.
5. Iterasi berhenti jika:

1. Ditemukannya kombinasi awalan yang tidak diizinkan berdasarkan awalan.

2. Awalan yang dideteksi saat ini sama dengan awalan yang dihilangkan sebelumnya.

3. tiga awalan telah dihilangkan.

Tabel 2.1 Kombinasi Awalan yang Tidak Diizinkan

|  |  |
| --- | --- |
| Awalan | Akhiran yang tidak diijinkan |
| be- | -i |
| di- | -an |
| ke- | -i, -kan |
| me- | -an |
| se- | -i, -kan |

1. Mengidentifikasi tipe awalan dan hilangkan. Awalan terdiri dari dua tipe:

1. Standar (“di-”, “ke-”, dan “se-”) yang dapat langsung dihilangkan dari kata.

2. Kompleks (“me-”, “be-”, “pe-”, “te-”) adalah tipe-tipe awalan yang dapat bermorfologi sesuai kata dasar yang mengikutinya.

1. Mencari kata yang telah dihilangkan awalannya. Jika tidak ditemukan, maka langkah 4 diulang kembali. Jika ditemukan, algoritma berhenti.
2. Apabila setelah langkah 4 kata dasar masih belum ditemukan, maka proses *recoding* dilakukan dengan mengacu pada Tabel 2.2. *Recoding* dilakukan dengan menambahkan karakter di awal kata yang dipenggal. Pada Tabel 2.2, karakter *recoding* adalah huruf kecil setelah tanda hubung (‘-’) dan kadang berada sebelum tanda kurung.
3. Jika semua gagal, maka masukan kata yang diuji pada algoritma ini dianggap sebagai kata dasar.

### Term Document Matrix

*Term Document Matrix* (TDM) adalah matriks 2 dimensi yang memiliki baris yang mewakili dokumen dan kolom yang berisi daftar *term* serta memiliki nilai frekuensi kemunculan suatu *term* pada suatu dokumen. *Term* tersebut didapatkan dari hasil proses *stemming*, kemudian dilakukan pengindeksan. Contohnya, kata “bukunya” dan “bukukan” memiliki indeks nilai yang sama dengan term buku (Manning *et al.* 2009).

### Algoritma TF-IDF

*Term frequency* (TF) dan *Inverse Document Frequency* (IDF) merupakan perhitungan pembobotan dalam frekuensi kemunculan sebuah dokumen tertentu dan *inverse* frekuensi dokumen yang mengandung kata tersebut. Frekuensi kemunculan kata yang menunjukkan seberapa penting kata tersebut dalam kumpulan dokumen (Wahyuni et al., 2017), untuk perhitungannya dapat dilihat pada persamaan 2.1, 2.2, dan 2.3.

*TF(d,t) = f(d,t)* (2.1)

Dimana *f*(*d,t*) adalah frekuensi kemunculan kata *t* pada dokumen *d*.

*IDF(t) =* (2.2)

Dimana *df*(*t*) adalah jumlah dokumen yang memiliki kata *t.*

*TF IDF* = *TF (d,t) × IDF (t)* (2.3)

## Klasterisasi

Klasterisasi adalah proses pengelompokan kumpulan objek data ke klasster yang memiliki kemiripian dan membedakan yang tidak mirip ke dalam klaster lain. Proses ini tidak dilakukan oleh manusia, sedangkan oleh alat algoritma klasterisasi. Maka dari itu, klasterisasi berguna untuk penentuan kelompok data yang tidak diketahui sebelumnya (Han *et al.* 2012).

Metode pengklasteran secara umum dapat dikelompokkan ke dalam tabel sesuai dengan tabel 2.2.

Tabel 2.2 Metode Klasterisasi (Han et al. 2012)

|  |  |  |
| --- | --- | --- |
| Metode | Karakteristik Umum | Contoh Algoritma |
| Metode Partisi | 1. Mencari klaster eksklusif yang mirip 2. Berbasis jarak 3. Menggunakan rata-rata atau medoid, untuk menggambarkan pusat klaster 4. Efektif untuk kumpulan data kecil hingga menengah | 1. K-Means  2. K-Medoids |
| Metode Hierarki | 1. Klasterisasi dekomposisi hierarki 2. Tidak dapat memperbaiki penggabungan dan pemisahan yang salah 3. Menggabungkan teknik lainnya seperti klasterisasi mikro atau keterhubungan objek | 1. Klasterisasi Aglomeratif/Divisif  2. Pengukuran jarak  3. *Balanced Iterative Reducing and Clustering (BIRD)*  4. *Chameleon*  5. Klasterisasi hierarki probabilistik |
| Metode Berbasis Kepadatan | 1. Dapat menentukan klaster yang bentuknya berubah-ubah 2. Klaster merupakan wilayah kepadatan objek pada suatu ruang yang dipisahkan oleh wilayah dengan kepadatan yang lebih rendah 3. Kepadatan klaster; Setiap titik harus memiliki jumlah minimal terdekat | 1. *Density-Based Spatial Clustering of Application with Noise (DBSCAN)*  2. *Ordering Points to Identify the Clustering Structure* (OPTICS) |
| Metode Berbasis Grid | 1. Menggunakan struktur data *grid* multiresolusi 2. Waktu proses yang cepat (tergantung pada jumlah objek data dan ukuran *grid*) | 1. *Statistical Information Grid* (STING)  2. *Clustering in Quest* (CLIQUE) |

### *Cosine Similarity*

*Cosine Similarity* adalah matriks yang digunakan untuk mengukur seberapa mirip dokumen dengan dokumen lain, terlepas dari ukurannya. Metode ini secara umum memiliki perhitungan *similarity* antara dua buah objek (misalkan P1 dan P2) dan dinyatakan dalam dua buah vektor yang menggunakan kata kunci dalam sebuah dokumen sebagai ukuran (Nurdiana et al., 2016).

*CosSim*(*di,qi*) = (2.4)

Dimana *qij* dan *dij* adalah bobot istilah yang dimiliki objek atau vektor pada dokumen.

### Algoritma DBSCAN

DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*) adalah metode *unsupervised-learning* yang populer digunakan dalam pembuatan model dan algoritma pembelajaran mesin. Mengingat bahwa DBSCAN adalah algoritma pengelompokan berbasis kepadatan, metode ini melakukan pekerjaan yang baik untuk mencari area dalam data yang memiliki kepadatan pengamatan yang tinggi, dibandingkan dengan area data yang tidak terlalu padat dengan pengamatan. Keuntungan lainnya adalah penggunaan metode DBSCAN dapat mengelompokkan data ke dalam kelompok dengan berbagai bentuk dan dapat mengidentifikasi *outliers* yang dianggap sebagai *noise*.

Pada DBSCAN, parameter yang digunakan adalah *minPts* (minimum Points) dan *eps / ε* (epsilon) dan kumpulan *dataset* dalam titik *Density-Based* (Capdevila, 2016). Konsep kepadatan pada DBSCAN melahirkan tiga macam status di setiap data, yaitu inti/*core* (titik pusat dalam klaster yang didasarkan pada kepadatan, dimana ada sejumlah titik yang harus berada dalam *Eps*)*,* batas/*border* (titik yang menjadi batasan dalam kawasan titik pusat), dan *noise* (titik yang tidak dapat dijangkau oleh *core* dan bukan merupakan *border*) (Putri et al., 2021).

Metode klasterisasi DBSCAN menemukan klaster-klaster dengan cara (Silitonga, 2016):

1. DBSCAN menelusuri klaster-klaster dengan memeriksa ε-*neighborhood* dari tiap-tiap *point* dalam *database.* Jika ε-*neighborhood* dari point p mengandung lebih dari MinPts, klaster baru dengan p sebagai *core object* diciptakan.
2. Kemudian DBSCAN secara iteratif mengumpulkan secara langsung objek-objek *density reachable* dari *core object* tersebut, dimana mungkin melibatkan penggabungan dari beberapa *cluster-cluster* *density reachable*.

Secara umum komputasi dari algoritma DBSCAN adalah sebagai berikut (Devi et al., 2015):

1. Inisialisasi parameter *minPts* dan *ε.*
2. Tentukan titik awal atau *p* secara acak.
3. Hitung ε atau semua jarak titik pada *density reachable* terhadap p dengan menggunakan *cosine similarity*.
4. Jika titik yang memenuhi ε lebih dari *minPts* maka titik p adalah *core object* dan klaster terbentuk.
5. Ulangi langkah secara iteratif hingga dilakukan proses pada semua titik.

### Algoritma OPTICS

OPTICS (*Ordering Points to Identify the Clustering Structure*) adalah algoritma klastering hierarkis yang bergantung pada kepadatan data. OPTICS mampu mendeteksi klaster yang bermakna dalam data dengan kepadatan yang bervariasi dengan menghasilkan urutan titik-titik yang linier, sehingga titik-titik yang terdekat secara spasial menjadi tetangga dalam urutan tersebut (Patwary, 2013). Berikut langkah-langkah penggunaan algoritma OPTICS (Prabahari, 2014):

1. Temukan jarak inti dari suatu objek. *p* adalah nilai eps/ε terkecil sehingga menjadikannya sebagai *core object*. Jika *p* bukan *core object*, jarak inti *p* tidak terdefinisi.
2. Jarak jangkauan objek *q* dengan objek lain *p* adalah nilai yang lebih besar dari *core object* *p* dan jarak Euclidean antara *p* dan *q*. Jika *p* bukan *core object*, maka jarak jangkauan antara *p* dan *q* tidak terdefinisi.

### Uji Validasi

Kualitas klaster *Ci* dapat diukur dengan menggunakan *silhouette coefficient.* Teknik ini memberikan representasi grafis singkat dari seberapa baik setiap objek terletak pada kelompok. Analisa metode validitas ini dilakukan dengan melihat besar nilai s. Hasil perhitungan nilai indeks validitas silhouette dapat bervariasi antara -1 hingga 1 (Budiman et al., 2016). *Silhouette coefficient* ini dirumuskan pada persamaan 2.5, Persamaan 2.6, dan Persamaan 2.7.

*a(****o****) = ,* ***o* ∈** *Ci* (1 ≤ *i* ≤ *k*) (2.5)

Setiap objek ***o* ∈** dataset, perhitungan dilakukan *a*(***o***)untuk mencari jarak rata-rata antara ***o*** dan semua objek lain pada klaster yang sama.

*b(****o****) =* min *Cj*:1 ≤ j < *k,j ≠ i*  (2.6)

*b*(***o***) adalah jarak rata-rata terkecil dari ***o*** terhadap semua objek lain pada klaster yang lain.

*s*(***o***) = (2.7)

*s*(***o***) merupakan koefisien silhouette. Koefisien memiliki nilai -1 hingga 1. Jika nilai silhouette semakin mendekati nilai 1, maka semakin baik pengelompokan data dalam suatu klaster. Jika sebaliknya, semakin mendekati nilai -1 maka semakin buruk pengelompokan data dalam klaster.

## Geovisualisasi

Geovisualisasi adalah proses analisis data geospasial di mana visualisasi dilakukan melalui suatu alat dengan konvergensi informasi, kartografi, dan metode geografi, menurut Yasobant *et al.* (2015). Fungsi spesifik dari teknik ini adalah digunakan dalam menampilkan data geospasial untuk menjelajahi, menganalisis, dan menyatukan data sehingga dapat menghasilkan hipotesis dan mengembangkan solusi, serta representasi data yang komprehensif. Ada 2 jenis pendekatan geovisualisasi, yaitu pendekatan fenomenologikal dan pendekatan positifistik. Pendekatan fenomenologikal digunakan sebagai interpretasi individu atas ruang dan waktu dalam bentuk abstrak. Berbeda dengan pendekatan positifistik, pendekatan ini menggunakan pemodelan spasial untuk mewakili dunia nyata.

### Sistem Informasi Geografis

Sistem Informasi Geografis merupakan sistem informasi berbasis komputer yang digunakan untuk mengumpulkan, memeriksa, mengintegrasikan, dan menganalisa informasi data yang memiliki dasar penggunaan secara geografis (Koko Mukti Wibowo, Indra Kanedi, 2021). Pada dasarnya, istilah Sistem Informasi Geografis terbagi menjadi tiga kata, yaitu sistem, informasi, dan geografi. Penggunaan kata “geografi” atau “geografis” ini merujuk pada suatu persoalan mengenai bumi; secara permukaan dua dimensi atau tiga dimensi. Istilah “informasi geografis” mengandung pengertian berupa informasi mengenai tempat, pengetahuan posisi suatu objek, dan keterangan-keterangan (atribut) yang terdapat di permukaan bumi yang posisinya diketahui.

### *Unified Modelling Language* (UML)

*Unified Modelling Language* (UML) adalah alat bantu dalam pembuatan sistem aplikasi dengan cara membuat desain terlebih dahulu terhadap sistem yang akan dibangun (Azwanti, 2017). UML menyediakan bahasa pemodelan visual memungkinkan bagi pengembang sistem untuk membuat *blueprint* dari visi yang telah dibuat dalam bentuk yang baku. Keuntungan utama penggunaan UML ini agar sistem lebih mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkomunikasikan rancangan dengan yang lain. UML mempunyai elemen grafis yang dapat dikombinasikan menjadi diagram, diantaranya *Use Case Diagram* dan *Activity Diagram*.

#### *Use Case Diagram*

*Use Case Diagram* merupakan deskripsi fungsi dari sebuah sistem dalam perspektif pengguna. *Use case* menggambarkan peran aktor-aktor dan himpunan *use case* yang terlibat sebagai fungsionalitas dalam penggunaan sistem (Azwanti, 2017).

Tabel 2.3 Komponen use case diagram

|  |  |  |
| --- | --- | --- |
| **Gambar** | **Nama** | **Fungsi** |
|  | *Package* | Menambahkan paket baru dalam diagram |
| Shape, circle  Description automatically generated | *Actor* | Menambah *actor* |
| A picture containing lamp, necklet  Description automatically generated | *Use case* | Menambahkan *use case* |
| A picture containing background pattern  Description automatically generated | *Unidirectional Association* | Menggambarkan relasi antara aktor dengan *use case* |
| Shape, arrow  Description automatically generated | *Dependencies or Instantiates* | Menggambarkan kebergantungan antar item dalam diagram |
|  | *Generalization* | Menggambarkan relasi lanjut antar *use case* atau menggambarkan struktur pewarisan antar aktor |

#### *Activity Diagram*

*Activity Diagram* adalah diagram yang menggambarkan sifat dinamis secara alamiah dalam sebuah sistem yang mengandung model aliran dan kontrol dalam aktivitas lainnya yang akan dibangun (Azwanti, 2017). Diagram ini menggambarkan berbagai alur aktivitas yang terjadi dalam suatu *use case*. Fungsinya sebagai menggambarkan aktivitas aktor yang ada dalam *use case diagram.*

Tabel *3.4* Komponen *activity diagram*

|  |  |  |
| --- | --- | --- |
| **Gambar** | **Nama** | **Fungsi** |
|  | *State* | Menambahkan state untuk suatu objek |
|  | *Activity* | Menambahkan aktivitas baru pada diagram |
|  | *Start state* | Menginisiasi dimana aliran kerja bermula |
|  | *End state* | Menginisiasi dimana aliran kerja berakhir |
| A picture containing background pattern  Description automatically generated | *State transition* | Menambahkan transisi dari suatu aktivitas ke aktivitas yang lainnya |
|  | *Transition to self* | Menambahkan transisi rekursif terhadap diagram |
|  | *Horizontal synchronization* | Menambahkan sinkronisasi horizontal pada diagram |
|  | *Vertical synchronization* | Menambahkan sinkronisasi vertikal pada diagram |

## Penelitian Sebelumnya Tentang Penyebaran Informasi Suatu Kejadian Menggunakan Twitter

Dwiarni (2019) melakukan penelitian tentang akuisisi dan klasterisasi data teks Twitter untuk memperoleh dasar pengetahuan terhadap profil pengguna Twitter. Penelitian dilakukan dengan ujicoba *keyword* “K-Pop” dan “K-Drama”. Dari hasil ujicoba akuisisi data didapatkan sebanyak 68.393 *tweet.* Hasil tersebut disebar menjadi 3 klaster / k=3, yang mana klaster pertama adalah waktu *tweet* dianggap pada pagi hari, klaster kedua adalah waktu *tweet* dianggap pada siang hari, dan klaster ketiga adalah waktu *tweet* dianggap pada malam hari. Kemudian, hasil klasterisasi didapat jam 21.00 - 01.00 merupakan mayoritas orang-orang melakukan *tweet*. Dari hasil penelitian ini kita dapatkan bahwa penentuan nilai *k* untuk memperkirakan topik suatu klaster didasarkan pada asumsi kebiasaan pengguna dalam menggunakan media sosial Twitter.

Terpstra (2012) melakukan penelitian tentang kemungkinan analisis secara *real time* dan otomatis dari pesan Twitter selama terjadinya situasi darurat. Analisis dilakukan dengan menggunakan *tools* ekstraksi informasi yang berhasil mendapatkan 97.000 *tweet* yang dikirim sebelum, saat, dan sesudah badai terjadi pada Festival Pukkelpop 2011 di Belgia. *Tool* ekstraksi dapat menganalisis *tweet* melalui tampilan geografis, jenis isi pesan (kerusakan, korban), dan jenis *tweet* (seperti *retweet*).

Denatari (2015) melakukan penelitian mengenai klasterisasi data teks Twitter untuk kasus pertanian di Indonesia. Data teks Twitter terbagi menjadi 2 jenis, yaitu data *tweet* sejumlah 51 data dan data konten *Uniform Resource Locator* (URL)sejumlah 51 data. Kedua jenis data tersebut dibandingkan dan dikelompokkan dengan algoritma *hierarchial clustering* untuk mendapatkan klaster terbaik.

(Crooks et al., 2013) penelitian ini dilakukan dengan analisis performa *microblogging* sebagai sistem sensor untuk mendeteksi kejadian dengan studi kasus gempa bumi yang ada di daerah East Coast, Amerika Serikat. Peneliti mengambil hasil deteksi yang memiliki karakteristik spasial dan temporal dari penyebaran informasi yang ada di situs *microblogging* (Twitter). Analisis terhadap situs ini juga dilakukan dengan teknik *crowdsourcing*, karena setiap media sosial atau situs *microblogging* juga memiliki informasi geografis ketika seorang pengguna mengomentari suatu kejadian yang dialami terjadi di sekitarnya, atau mengenai pusat lokasi yang menjadi pusat perhatian. Namun, perbedaannya media sosial atau situs *microblogging* tidak menyediakan informasi geografis pengguna secara terang-terangan, berbeda dengan teknik *crowdsourcing* yang sudah ada pada aplikasi Wikimapia atau OpenStreetMap.

Penelitian ini bertujuan untuk menilai kualitas informasi yang telah diambil dari masyarakat dengan mempertimbangkan reaksi pengguna Twitter terhadap gempa bumi yang terjadi di Virginia, Amerika Serikat pada tanggal 23 Agustus 2011. Hasilnya, *tweet* dapat digunakan untuk memberi perkiraan yang cepat dan bagus dari wilayah yang terkena dampak gempa bumi. Perkiraan ini digunakan sebagai informasi yang penting untuk penanganan dan pemulihan dampak bencana. Dengan kemampuannya untuk memperkirakan wilayah yang terkena dampak gempa bumi dengan akurat, hal tersebut mendukung pernyataan bahwa dengan mengambil informasi geospasial di Twitter, peneliti memperoleh informasi yang penting mengenai dampak dari suatu kejadian dengan cepat.

(Rahmanti et al., 2021) melakukan penelitian dengan mengidentifikasi informasi tentang resiko dan respon komunikasi masyarakat Indonesia terhadap pemberlakuan *New Normal* ketika pandemi Covid-19 yang ada pada situs *microblogging* (Twitter) di wilayah Indonesia. Penelitian ini bertujuan untuk menggolongkan *tweet* yang memiliki sentimen positif, negatif, dan netral dengan klasifikator *naïve-bayes* dan memasukkannya ke dalam analisis emosi dasar dari *Plutchik’s Wheel of Emotions* (*joy, fear, anticipation, anger, disgust, sadness, surprise,* dan *trust*). Penelitian ini dilakukan pada tanggal 21 Mei 2020 – 18 Juni 2020, dengan hasil data sebanyak 282.216 *tweet* dari 137.057 pengguna. *Tweet* itu semua mengandung 88.677 *mention*, 31.452 *reply*, 164.087 *retweet*. Hasil tersebut disebar ke dalam *Plutchik’s Wheel of Emotions* dengan persentase; *joy* (9,01%)*, fear* (6,50%)*, anticipation* (14,82%)*, anger* (4,81%)*, disgust* (0,73%)*, sadness* (1,74%)*, surprise* (8,62%)*,* dan *trust* (53,77%). Kemudian, didapatkan hasil penggunaan tiga *hashtag* terbanyak, yaitu *#NewNormal* (17.051 *tweet*), *#TataKehidupanBaru* (10.980 *tweet*), dan *#DisiplinPolaHidupBaru* (5.200 *tweet*). Dari hasil ini peneliti dapat menggolongkan suatu kejadian yang ada di situs *microblogging* (Twitter) ke dalam pemetaan analisis berdasarkan emosi pengguna.

# 

# METODE PENELITIAN

Metode penelitian merupakan suatu prosedur yang digunakan untuk melakukan penelitian, sehingga mampu menjawab rumusan masalah dan tujuan penelitian dengan landasan ilmiah tertentu.

## Waktu Penelitian

Penelitian ini dilaksanakan selama 6 bulan mulai dari 1 Mei 2022 – 30 Oktober 2022.

## Objek Penelitian

Objek penelitian ini adalah *tweet* terkait penyebaran penyakit menular langsung di Indonesia dengan periode waktu tertentu, yaitu khusus virus Covid-19 varian delta dimulai dari 1 April 2021 – 30 September 2021 dan khusus virus Covid-19 Varian Omicron dimulai dari 1 Januari 2022 – 30 Juni 2022, dengan menggunakan kata kunci. Kata kunci dapat dilihat pada Tabel 3.1.

Tabel 3.1 Kata Kunci yang digunakan untuk pencarian tweet

|  |  |  |
| --- | --- | --- |
| No. | Kata Kunci Covid-19 | Keterangan |
|  | Covid-19 | Virus penyakit yang menjadi pandemi dimulai pada tahun 2019 |
|  | *SARS-CoV-2* | Virus penyebab penyakit Covid-19 |
|  | Coronavirus | Jenis virus penyakit yang menginfeksi sistem pernapasan |
|  | Batuk | Bentuk respons tubuh terhadap infeksi atau iritasi yang terjadi di dalam sistem pernapasan |
|  | Batuk Kering | Jenis batuk yang tidak mengandung dahak atau lendir |
|  | Pilek | Kondisi saat hidung mengeluarkan lendir yang berlebihan |
|  | Kelelahan | Bentuk respons tubuh terhadap virus yang telah menginfeksi sistem imun |
|  | Sakit Kepala | Rasa nyeri yang menyerang bagian kepala |
|  | Demam | Bentuk respons tubuh terhadap Penyakit |
|  | Sesak Napas | Kondisi seseorang mengalami kesusahan dalam bernapas |
|  | Varian Omicron Covid-19 | Sebuah varian atau jenis penyakit dari virus Covid-19 (*Variants of Concern*) dengan tingkat fatalitas 1,9% |
|  | Varian Delta Covid-19 | Sebuah varian atau jenis penyakit dari virus Covid-19 (*Variants* of Concern) dengan tingkat fatalitas 3,4% |
|  | Sakit Tenggorokan | Rasa nyeri yang menyerang tenggorokan dan susah menelan makanan |
|  | Anosmia | Kehilangan kemampuan untuk mendeteksi rasa atau bau |
|  | Diare | Buang air besar encer dan berulang |

## Tahapan Penelitian

Penelitian ini memiliki beberapa tahapan yang dapat dilihat pada Gambar di bawah ini.

Mulai

Akuisisi *Tweet*

Praproses Data

Hasil Praproses

Klasterisasi DBSCAN dan OPTICS

Evaluasi Hasil Analisis

Geovisualisasi

Selesai

### Akuisisi *Tweet*

Akuisisi *tweet* menggunakan proses *web crawling* dan *web scraping* dengan menggunakan program Python serta Twitter API. Proses *web crawling* dan *web scraping* menggunakan *library snscrape* yang tersedia pada  [https://github.com/JustAnotherArchivist/snscrape/](https://www.seleniumhq.org/). *Snscrape* merupakan *tools* berupa *package* *library* untuk melakukan interaksi *web browser* secara otomatis yang dikombinasikan dengan program Python. Program Python akan melakukan proses *web crawling* dengan cara membuka halaman pencarian Twitter secara otomatis berdasarkan input tautan yang dimasukkan sebelumnya. Setelah halaman selesai dimuat seluruhnya, dilakukan proses *web scraping* dengan cara mengekstrak data dari halaman hasil pencarian Twitter.

Akuisisi dilakukan dengan memasukkan kata kunci yang telah ditentukan dan rentang waktu selama dua minggu pada tautan pencarian Twitter. Akuisisi dilakukan sebanyak dua kali per bulan mulai bulan April 2021 sampai bulan Juni 2022. Tahap akuisisi *tweet* akan menghasilkan *output* dalam format *Comma Separated* *Value* (CSV). Metode *web crawling* dan *web scraping* menghasilkan *username,* *date, time, content,* dan *tweetID.* Twitter API digunakan untuk mendapatkan *placeID,* daerah, lokasi, *longitude,* dan *latitude.*

### Praproses Data

Praproses data dilakukan untuk pengolahan data yang akan menghasilkan berupa *Term Document Matrix* (TDM) untuk tahap pembobotan *term* dan klasterisasi.

#### Case Folding

*Case Folding* adalah tahapan yang berfungsi mengonversi keseluruhan teks dalam dokumen menjadi huruf kecil, kemudian dilakukan penghapusan *mention*, URL, dan tanda baca. Contoh data *tweet* sebelum dan sesudah proses *case folding*, dapat dilihat pada Tabel 3.1.

Tabel 3.1 Contoh *Case Folding*

|  |  |
| --- | --- |
| **Data awal** | **Data akhir** |
| dr. Erlina menyarankan agar pasien Covid-19 selalu memantau frekuensi napas agar bisa mengetahui tanda sesak napas - #Sains https://t.co/mxpV2xCBBP | “dr. erlina menyarankan agar pasien covid19 selalu memantau frekuensi napas agar bisa mengetahui tanda sesak napas” |
| kyk ga sanggup pegang hp lamaÂ².. liat twitter pd nyari oksigen trs kamar RS, di WA juga tiap hr ada aja grup yg ngabarin positif, nyari obat, donar darah plasma, kritis lah sesak napas lah.. gw yg sehat jd berasa ikut sakit.. ðŸ˜µ | “kyk ga sanggup pegang hp lama².. liat twitter pd nyari oksigen trs kamar rs, di wa juga tiap hr ada aja grup yg ngabarin positif, nyari obat, donar darah plasma, kritis lah sesak napas lah.. gw yg sehat jd berasa ikut sakit.. “ |

#### Tokenizing

Tokenizing dilakukan dengan menggunakan *library* Natural Language Toolkit berbasis Python yang tersedia pada <https://www.nltk.org/>. Data *tweet* diubah menjadi kumpulan data dengan mengubah formatnya menjadi *Comma Separated Values* (CSV). Kumpulan data *tweet* kemudian diubah menjadi *corpus*. *Corpus* merupakan entitas yang secara konseptual mirip dengan basis data dalam penyimpanan dan pengaturan dokumen teks (Feinerer et al., 2008). Semua huruf pada *corpus* telah menjadi huruf kecil. Contoh data *tweet* sebelum dan sesudah proses *tokenizing* dapat dilihat pada Tabel 3.2.

Tabel 3.2 Contoh Tokenizing

|  |  |
| --- | --- |
| **Data awal** | **Data akhir** |
| dr. Erlina menyarankan agar pasien Covid-19 selalu memantau frekuensi napas agar bisa mengetahui tanda sesak napas - #Sains https://t.co/mxpV2xCBBP | “dr”, “erlina”, “menyarankan”, “agar”, “pasien”, “covid19”, “selalu”, “memantau”, “frekuensi”, “napas”, “agar”, “bisa”, “mengetahui”, “tanda”, “sesak”, “napas” |
| kyk ga sanggup pegang hp lamaÂ².. liat twitter pd nyari oksigen trs kamar RS, di WA juga tiap hr ada aja grup yg ngabarin positif, nyari obat, donar darah plasma, kritis lah sesak napas lah.. gw yg sehat jd berasa ikut sakit.. ðŸ˜µ | “kyk”, “ga”, “sanggup”, “pegang”, “hp”, “lama”, “liat”, “twitter”, “pd”, “nyari”, “oksigen”, “trs”, “kamar”, “RS”, “di”, “WA”, “juga”, “tiap”, “hr”, “ada”, “aja”, “grup”, “yg”, “ngabarin”, “positif”, “nyari”, “obat”, “donar”, “darah”, “plasma”, “kritis”, “lah”, “sesak”, “napas”, “lah”, “gw”, “yg”, “sehat”, “jd”, “berasa”, “ikut”, “sakit” |

#### Penghapusan *Stopword*

Nilai informasi yang terdapat dalam *stopword* hampir mendekati nol, dengan kata lain entropi yang dimiliki sangat rendah (Feinerer *et al.* 2008). Sebelum dilakukan analisis lebih lanjut, *stopword* harus dihilangkan. Tahap penghapusan *stopword* dilakukan untuk membuang kata-kata yang termasuk ke dalam daftar *stopword.* Contoh *stopword* dalam Bahasa Indonesia diantaranya yaitu dahulu, ada, dalam, adanya, dan, pada, dan lain-lain. Acuan daftar kata-kata yang termasuk ke dalam *stopword* diperoleh dari *library* Natural Language Toolkit pada Python sebanyak xx kata dalam Bahasa Indonesia. Contoh data *tweet* sebelum dan sesudah penghapusan *stopword* dapat dilihat pada Tabel 3.3.

Tabel 3.3 Contoh stopwords removal

|  |  |
| --- | --- |
| **Data awal** | **Data akhir** |
| “dr”, “erlina”, “menyarankan”, “agar”, “pasien”, “covid19”, “selalu”, “memantau”, “frekuensi”, “napas”, “agar”, “bisa”, “mengetahui”, “tanda”, “sesak”, “napas” | “dr”, “erlina”, “menyarankan”, “pasien”, “covid19”, “memantau”, “frekuensi”, “napas”, “tanda”, “sesak”, “napas” |
| “kyk”, “ga”, “sanggup”, “pegang”, “hp”, “lama”, “liat”, “twitter”, “pd”, “nyari”, “oksigen”, “trs”, “kamar”, “RS”, “di”, “WA”, “juga”, “tiap”, “hr”, “ada”, “aja”, “grup”, “yg”, “ngabarin”, “positif”, “nyari”, “obat”, “donar”, “darah”, “plasma”, “kritis”, “lah”, “sesak”, “napas”, “lah”, “gw”, “yg”, “sehat”, “jd”, “berasa”, “ikut”, “sakit” | “kyk”, “ga”, “sanggup”, “pegang”, “hp”, “liat”, “twitter”, “pd”, “nyari”, “oksigen”, “trs”, “kamar”, “RS”, “WA”, “hr”, “aja”, “grup”, “yg”, “ngabarin”, “positif”, “nyari”, “obat”, “donar”, “darah”, “plasma”, “kritis”, “sesak”, “napas”, “gw”, “yg”, “sehat”, “jd”, “berasa”, “sakit” |

#### Stemming

Proses *stemming* dilakukan untuk menghapus awalan dan akhiran dari suatu kata. Tujuan dari tahap *stemming* adalah untuk mendapatkan kata dasar yang sesuai. Proses *stemming* menggunakan *library* Sastrawi berbasis Python yang tersedia pada <https://github.com/sastrawi/sastrawi>. Algoritma yang terdapat pada *library* Sastrawi adalah Nazief-Adriani yang digunakan untuk menghapus berbagai variasi awalan dan akhiran kata. Contoh data *tweet* sebelum dan sesudah proses *stemming* dapat dilihat pada Tabel 3.4.

Tabel 3.4 Contoh stemming

|  |  |
| --- | --- |
| **Data awal** | **Data akhir** |
| “dr”, “erlina”, “menyarankan”, “pasien”, “covid19”, “memantau”, “frekuensi”, “napas”, “tanda”, “sesak”, “napas” | “dr”, “erlina”, “saran”, “pasien”, “covid19”, “pantau”, “frekuensi”, “napas”, “tanda”, “sesak”, “napas” |
| “kyk”, “ga”, “sanggup”, “pegang”, “hp”, “lama”, “liat”, “twitter”, “pd”, “nyari”, “oksigen”, “trs”, “kamar”, “RS”, “di”, “WA”, “juga”, “tiap”, “hr”, “ada”, “aja”, “grup”, “yg”, “ngabarin”, “positif”, “nyari”, “obat”, “donar”, “darah”, “plasma”, “kritis”, “lah”, “sesak”, “napas”, “lah”, “gw”, “yg”, “sehat”, “jd”, “berasa”, “ikut”, “sakit” | “sanggup”, “pegang”, “oksigen”, “kamar”, “grup”, “positif”, “obat”, “darah”, “plasma”, “kritis”, “sesak”, “napas”, “sehat”, “asa”, “sakit” |

#### Pembuatan Term Document Matrix

Proses pembuatan *Term Document Matrix* (TDM) menggunakan *library* scikit-learn berbasis Python yang tersedia pada <https://scikit-learn.org/>. Tahap pembuatan TDM dilakukan untuk menghasilkan matriks frekuensi kemunculan *term* pada suatu dokumen. Satu *tweet* mewakili 1 dokumen. Pada proses ini dilakukan reduksi *term* untuk memperkecil dimensi matriks. Matriks pada TDM memiliki baris yang berisi indeks dari dokumen pada kumpulan *corpus* dan kolom yang menunjukkan kata yang ada dalam data *tweet*. Hasil pembuatan TDM digunakan pada tahap klasterisasi.

#### Pembobotan TF-IDF

Tahap terakhir yang dilakukan pada praproses data dengan TF-IDF. Pada tahap ini dilakukan perhitungan sesuai dengan persamaan 2.1, 2.2, dan 2.3. Hasil dari tahap ini adalah vektor ukuran kemiripan yang dimiliki tiap kata kunci dokumen.

### Klasterisasi dengan DBSCAN

Pada tahap ini dilakukan klasterisasi dengan algoritme DBSCAN untuk data *tweet* yang telah dilakukan praproses menjadi *Term Document Matrix* (TDM) menggunakan *library* scikit-learn dan pembobotan TF-IDF. Klasterisasi digunakan untuk mendapatkan klaster dari setiap dokumen berdasarkan *term* terkait penyebaran penyakit menular langsung (studi kasus Covid-19). Klaster yang dihasilkan akan digunakan pada proses geovisualisasi. *Flowchart* yang menunjukkan teknik metode DBSCAN dapat dilihat pada gambar 3.1.

Diagram

Description automatically generated

Gambar 3.1 flowchart DBSCAN

Penjelasan *flowchart* algoritma DBSCAN sebagai berikut:

1. Dataset yang berbentuk file .csv dianggap sebagai input
2. Menghitung bobot tiap-tiap term dengan algoritme TF-IDF
3. Epsilon dan MinPts dimasukkan dan dianggap sebagai input
4. Menentukan titik awal atau *p* secara acak
5. Mendapatkan titik *p* sebagai *core object* dan titik *p* tidak memiliki *density reachable*
6. Menghitung epsilon atau semua jarak titik pada *density reachable* terhadap *p* dengan menggunakan *cosine similarity* (sesuai dengan persamaan 2.4).
7. Mengulangi langkah perhitungan hingga tercipta titik yang dianggap sebagai *noise.*
8. Mendapatkan *cluster* terbaik.

### Klasterisasi dengan OPTICS

Pada tahap ini dilakukan klasterisasi dengan algoritme OPTICS untuk data *tweet* yang telah dilakukan praproses menjadi *Term Document Matrix* (TDM) menggunakan *library* scikit-learn dan pembobotan TF-IDF. Klasterisasi digunakan untuk mendapatkan klaster dari setiap dokumen berdasarkan *term* terkait penyebaran penyakit menular langsung (studi kasus Covid-19). Klaster yang dihasilkan akan digunakan pada proses geovisualisasi. *Flowchart* yang menunjukkan teknik metode OPTICS dapat dilihat pada gambar 3.2.

Diagram

Description automatically generated

Gambar 3.2 flowchart OPTICS

Penjelasan *flowchart* algoritma OPTICS sebagai berikut:

1. Dataset yang berbentuk file .csv dianggap sebagai input
2. Menghitung bobot tiap-tiap term dengan algoritme TF-IDF
3. Epsilon dan MinPts dimasukkan dan dianggap sebagai input
4. Menentukan titik awal atau *p* secara acak
5. Menghitung epsilon atau semua jarak titik pada *density reachable* terhadap *p* dengan menggunakan *cosine similarity* (sesuai dengan persamaan 2.4).
6. Mengulangi langkah perhitungan hingga tercipta titik yang dianggap sebagai *noise.*
7. Mendapatkan *cluster* terbaik.

### Perancangan Sistem

Pada tahap ini akan dilakukan perancangan sistem dari hasil model klasterisasi antara metode DBSCAN dan OPTICS, dengan tujuan untuk mengolah dataset yang sudah disiapkan, menampilkan hasil geovisualisasi dan untuk mengelola sistem penggunaan.

Pada perancangan sistem ini digunakan konsep UML (*Unified Modelling Language*), yaitu *Use Case Diagram* dan *Activity Diagram. Use Case Diagram* digunakan sebagai penggambaran interaksi antar aktor dengan sistem, sedangkan *Activity Diagram* digunakan untuk menggambarkan alur penggunaan sistem.

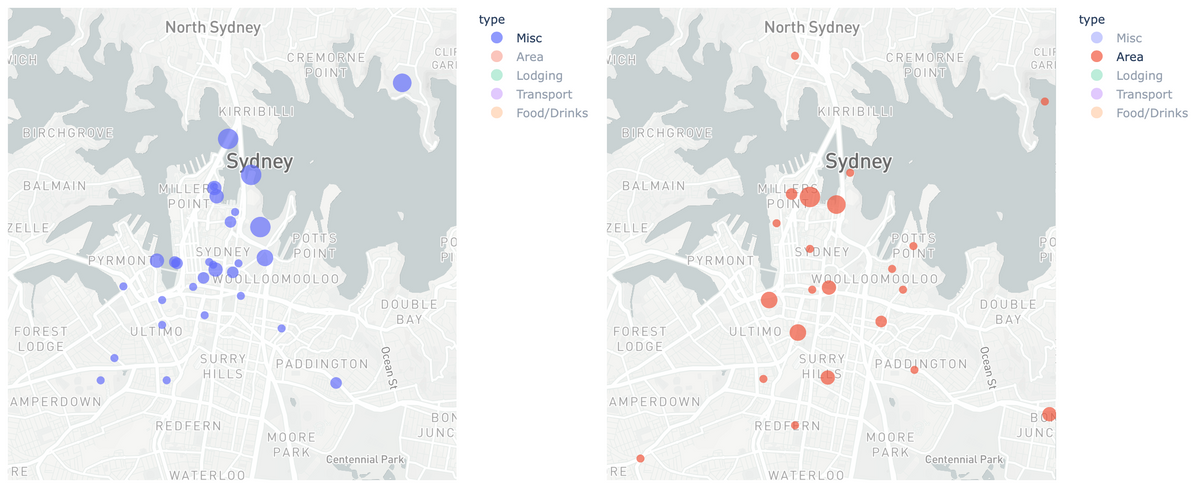
### Implementasi Sistem

Implementasi sistem pada penelitian ini yaitu *website* untuk menampilkan hasil sebaran klasterisasi penyebaran penyakit menular langsung (studi kasus Covid-19). Sistem ini dibangun dengan menggunakan bahasa pemrograman Python dan *framework* Flask untuk menjalankan model klasterisasi perbandingan DBSCAN atau OPTICS dengan memanfaatkan *library* Pythonplotly (<https://plotly.com/python/maps/>) sebagai menampilkan hasil geovisualisasi dan manajemen pengguna.

### Geovisualisasi

Proses geovisualisasi digunakan untuk mendistribusikan hasil klasterisasi yang diambil dari geolokasi setiap data *tweet*. Penerapan geovisualisasi dengan pemilihan warna, pola, dan ukuran agar dapat menambah informasi yang dibutuhkan untuk evaluasi hasil.

Proses geovisualisasi memanfaatkan *tools* plot.ly yang tersedia pada *library* bahasa python dan Chart Studio, merupakan *tools* analitik dan visualisasi secara *online*. Peta interaktif dapat dijalankan melalui *website* atau *web browser* dan dibagikan melalui URL. Berikut contoh gambar peta interaktif dari plot.ly pada Gambar 3.3.



Gambar 3.3 Peta Interaktif Plot.ly

### Evaluasi Hasil Analisis

Pada tahap ini klaster dianalisis dari proses klasterisasi dan lokasi penyebaran penyakit menular langsung dari proses geovisualisasi. Analisis hasil klaster dilakukan dengan melihat *term* yang sering muncul dan sesuai pada setiap klaster untuk dijadikan dasar dalam penentuan label klaster. Sedangkan analisis lokasi penyebaran penyakit menular langsung dilakukan dengan cara membandingkan pola geolokasi dengan data mengenai daerah penyebaran penyakit menular yang diperoleh dari Kementerian Kesehatan Republik Indonesia (2021) melalui pendekatan statistik deskriptif, yaitu koefisien deskriptif yang dapat mewakili suatu *dataset.* Hasil analisis diharapkan dapat menjadi dasar keputusan alternatif dalam penanganan penyebaran penyakit menular langsung (studi kasus Covid19).

# HASIL DAN PEMBAHASAN

## Akuisisi Tweet

Pengumpulan data *tweet* ini memiliki judul akuisisi *tweet*. Hasil dari akuisisi *tweet* diperoleh dari proses *web crawling* dan *web scraping* dengan bahasa pemrograman Python dan Twitter API. Program Python menggunakan snscrapeyang terdapat pada library Python di Github untuk mendapatkan data *tweet* dan TwitterAPI untuk mendapatkan geolokasi pengguna. Data *tweet* yang diambil adalah kumpulan kata kunci yang disebutkan pada Tabel 3.1 sejak April – September 2021 dan Januari – Maret 2022. *Source code* untuk scraping data terdapat pada Lampiran X. Langkah-langkah untuk melakukan akuisisi *tweet* atau *scraping* data adalah sebagai berikut:

1. Mendaftar akun *developer* pada Twitter Developer untuk mendapatkan akses ke setiap *tweet,* termasuk akses geolokasi yang dibutuhkan. Kemudian membuat aplikasi baru pada Twitter API untuk mendapatkan API *key* dan *access token* digunakan untuk autentikasi mendapatkan lokasi pengguna sesuai lokasi yang ada pada profil pengguna. Contoh pembuatan aplikasi baru dapat dilihat pada Gambar xx
2. Melakukan *scraping* berdasarkan *query* pencarian menggunakan *library* snscrape dengan mengirimkan parameter kata kunci yang tertera pada tabel

## Praproses Data

### Case Folding

*Case Folding* adalah tahapan yang berfungsi mengonversi keseluruhan teks dalam dokumen menjadi huruf kecil (*lowercase*). Langkah-langkah melakukan *case folding* yaitu memanggil fungsi *lower()* dalam iterasi setiap data dokumen. Contoh hasil praproses data pada tahap ini dapat dilihat pada Tabel xx

### *Tokenizing*

Setelah dilakukan *case folding*, tahap praproses data selanjutnya adalah *Tokenizing* yang berfungsi untuk melakukan penghapusan *hashtag*, angka, *mention*, URL, dan tanda baca. Langkah – langkah *tokenizing* adalah sebagai berikut:

1. Memanggil *library*, *numpy, re,* dan *string*.
2. Menggunakan *library* pandas untuk membaca data csv menjadi *dataframe.*
3. Mengambil data di kolom *text* dan menyimpan di dalam *list.*
4. Melakukan iterasi pada *list* dan menghapus *hashtag,* angka, *mention,* URL, dan tanda baca.

Contoh hasil praproses data tahap *tokenizing* dapat dilihat pada Tabel xx

### Penghapusan *Stopword*

Pada tahap ini dilakukan penghapusan *stopword.* Tahap ini dilakukan untuk membuang kata-kata yang tidak terlalu berpengaruh pada pemrosesan *text mining*, seperti kata hubung dan termasuk ke dalam daftar *stopword.* Daftar kata yang akan dilakukan penghapusan adalah daftar kata yang termasuk pada *library Natural Language Tool Kit* dan daftar kata tambahan. Langkah – langkah penghapusan *stopword* tertera pada Tabel xx

### Stemming Nazief – Adriani

Tahap praproses data berikutnya adalah *stemming*, yaitu mengembalikan kata-kata yang menjadi imbuhan dari kata dasar dan menghapus awalan dan akhiran dari suatu kata. Tujuan dari tahap *stemming* adalah untuk mendapatkan kata dasar yang sesuai. Proses *stemming* menggunakan *library* Sastrawi berbasis Python yang tersedia pada <https://github.com/sastrawi/sastrawi>. Algoritma yang terdapat pada *library* Sastrawi adalah Nazief-Adriani yang digunakan untuk menghapus berbagai variasi awalan dan akhiran kata. Langkah – langkah tahapan *stemming* adalah sebagai berikut:

1. Memanggil *library* Sastrawi.
2. Melakukan iterasi dokumen, kemudian memanggil fungsi *stem()* untuk mengubah kata dalam dokumen tersebut menjadi kata dasar.

Contoh hasil praproses data tahap *stemming* dapat dilihat pada Tabel xx

### *Term Document Matrix*

Tahap selanjutnya adalah pembuatan *Term Document Matrix* dilakukan untuk menghasilkan matriks frekuensi kemunculan term pada suatu dokumen. Pada tahap ini menghasilkan xxx dokumen dengan xxx *term*. Banyaknya *term* yang dihasilkan membuat dimensi matriks menjadi terlalu besar sehingga perlu diperkecil dengan cara mereduksi *term*

### Algoritma TF-IDF

## Klasterisasi

## Geovisualisasi

### Persiapan Data

### Pembuatan Rancangan Sistem

## Evaluasi Hasil Analisis

**KESIMPULAN DAN SARAN**

## Kesimpulan

## Saran

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Metode** | **Juni**  **2022** | | | | **Juli**  **2022** | | | | **Agustus**  **2022** | | | | **September**  **2022** | | | | **Oktober**  **2022** | | | | **November**  **2022** | | | | |
| **1** | **Penyusunan Proposal** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2** | **Akuisisi *tweet*** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3** | **Praproses Data** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4** | **Klasterisasi Metode** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5** | **Evaluasi Hasil Metode** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6** | **Geovisualisasi** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

### Jadwal Penelitian

# DAFTAR PUSTAKA

Baumgartner, C., & Graber, A. (2007). Data mining and knowledge discovery in metabolomics. *Successes and New Directions in Data Mining*, *39*(11), 141–166. https://doi.org/10.4018/978-1-59904-645-7.ch007

Budiman, S., Safitri, D., & Ispriyanti, D. (2016). Perbandingan Metode K-Means Dan Metode Dbscan Pada Pengelompokan Rumah Kost Mahasiswa Di Kelurahan Tembalang Semarang. *Jurnal Gaussian*, *5*(4), 757–762.

Chakrabarti, S., Ester, M., Fayyad, U., & Gehrke, J. (2006). Data mining curriculum: a proposal. *Acm Sigkdd*, 1–10. http://pdf.aminer.org/000/303/279/decision\_tree\_construction\_from\_multidimensional\_structured\_data.pdf%5Cnhttp://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Data+mining+curriculum:+A+proposal+(Version+1.0)#4%5Cnhttp://scholar.google.com/scholar

Crooks, A., Croitoru, A., Stefanidis, A., & Radzikowski, J. (2013). #Earthquake: Twitter as a Distributed Sensor System. *Transactions in GIS*, *17*(1), 124–147. https://doi.org/10.1111/j.1467-9671.2012.01359.x

Devi, A. S., Putra, I. K. G. D., & Sukarsa, I. M. (2015). Implementasi Metode Clustering DBSCAN pada Proses Pengambilan Keputusan. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, *6*(3), 185. https://doi.org/10.24843/lkjiti.2015.v06.i03.p05

Dwiarni, B. A., & Setiyono, B. (2019). Akuisisi dan Clustering Data Sosial Media Menggunakan Algoritma K-Means sebagai Dasar untuk Mengetahui Profil Pengguna. *Jurnal Sains Dan Seni*, *8*(2), 2337–3520. https://apps.twitter.com/

Fay, D. L. (1967). 済無No Title No Title No Title. *Angewandte Chemie International Edition, 6(11), 951–952.*

Feinerer, I., Hornik, K., & Meyer, D. (2008). Text mining infrastructure in R. *Journal of Statistical Software*, *25*(5), 1–54. https://doi.org/10.18637/jss.v025.i05

Freeman, J. (2019). What is an API? Application programming interfaces explained. In *InfoWorld* (pp. 1–9). https://www.infoworld.com/article/3269878/what-is-an-api-application-programming-interfaces-explained.html

Han, J., Kamber, M., & Pei, J. (Eds.). (2012). About the Authors. In *Data Mining (Third Edition)* (Third Edit, p. xxxv). Morgan Kaufmann. https://doi.org/https://doi.org/10.1016/B978-0-12-381479-1.00027-7

Koko Mukti Wibowo, Indra Kanedi, J. J. (2021). Sistem Informasi Geografis (Sig) Menentukan Lokasi Pertambangan Batu Bara Di Provinsi Bengkulu Berbasis Website. *Jurnal Media Infotama*, *11*(1), 223–260.

Liao, S. H., Chu, P. H., & Hsiao, P. Y. (2012). Data mining techniques and applications - A decade review from 2000 to 2011. *Expert Systems with Applications*, *39*(12), 11303–11311. https://doi.org/10.1016/j.eswa.2012.02.063

Melcer, E. F., & Isbister, K. (2018). Bots & (main)frames: Exploring the impact of tangible blocks and collaborative play in an educational programming game. *Conference on Human Factors in Computing Systems - Proceedings*, *2018*-*April*(April). https://doi.org/10.1145/3173574.3173840

Nurdiana, O., Jumadi, J., & Nursantika, D. (2016). Perbandingan Metode Cosine Similarity Dengan Metode Jaccard Similarity Pada Aplikasi Pencarian Terjemah Al-Qur’an Dalam Bahasa Indonesia. *Jurnal Online Informatika*, *1*(1), 59. https://doi.org/10.15575/join.v1i1.12

Prabahari, R. . T. (2014). *A Comparative Analysis of Density Based Clustering Techniques for Outlier Mining*. *3*(11), 132–136.

Putri, M. M., Dewi, C., Permata Siam, E., Asri Wijayanti, G., Aulia, N., & Nooraeni, R. (2021). *Komparasi DBSCAN dan K-Means Clustering pada Pengelompokan Status Desa di Jawa Tengah Tahun 2020*. *17*(3), 394–404. https://doi.org/10.20956/j.v17i3.11704

Rahmanti, A. R., Ningrum, D. N. A., Lazuardi, L., Yang, H. C., & Li, Y. C. (2021). Social Media Data Analytics for Outbreak Risk Communication: Public Attention on the “New Normal” During the COVID-19 Pandemic in Indonesia. *Computer Methods and Programs in Biomedicine*, *205*, 106083. https://doi.org/10.1016/j.cmpb.2021.106083

Sakaki, T., Okazaki, M., & Matsuo, Y. (2013). Tweet analysis for real-time event detection and earthquake reporting system development. *IEEE Transactions on Knowledge and Data Engineering*, *25*(4), 919–931. https://doi.org/10.1109/TKDE.2012.29

Santoso, A. M. . (2022). Covid-19 : Varian Dan Mutasi. *Jurnal Medika Hutama*, *3*(02), 1980–1986. https://jurnalmedikahutama.com/index.php/JMH/article/view/396/271

Silitonga, P. (2016). ANALISIS POLA PENYEBARAN PENYAKIT PASIEN PENGGUNA BADAN PENYELENGGARA JAMINAN SOSIAL (BPJS) KESEHATAN DENGAN MENGGUNAKAN METODE DBSCAN CLUSTERING ( Studi Kasus Rumah Sakit Umum Pusat Haji Adam Malik Medan ). *Jurnal TIMES*, *Vol. V No*(ISSN : 2337-3601), 11–40. http://etd.lib.metu.edu.tr/upload/12620012/index.pdf

Susanto, H., Sumpeno, S., & Rachmadi, R. F. (2014). Visualisasi Data Teks TwitterBerbasis Bahasa Indonesia Menggunakan Teknik Pengklasteran. *Jurnal Teknik Elektro Institut Teknologi Sepuluh Nopember*, 6. http://digilib.its.ac.id/ITS-paper-22121150006831/35629

Susilo, A., Rumende, C. M., Pitoyo, C. W., Santoso, W. D., Yulianti, M., Herikurniawan, H., Sinto, R., Singh, G., Nainggolan, L., Nelwan, E. J., Chen, L. K., Widhani, A., Wijaya, E., Wicaksana, B., Maksum, M., Annisa, F., Jasirwan, C. O. M., & Yunihastuti, E. (2020). Coronavirus Disease 2019: Tinjauan Literatur Terkini. *Jurnal Penyakit Dalam Indonesia*, *7*(1), 45. https://doi.org/10.7454/jpdi.v7i1.415

Wahyuni, R. T., Prastiyanto, D., & Supraptono, E. (2017). Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi. *Jurnal Teknik Elektro Universitas Negeri Semarang*, *9*(1), 18–23. https://journal.unnes.ac.id/nju/index.php/jte/article/download/10955/6659