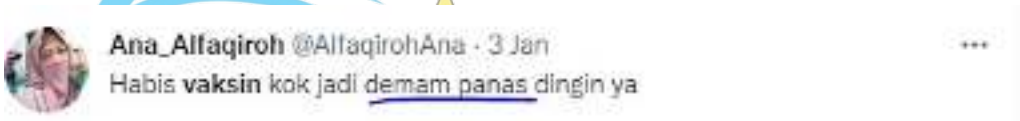
BAB III
METODE PENELITIAN

3.1. Objek Penelitian

Objek penelitian dilakukan pada media sosial twitter dengan mengambil sampel dari postingan yang dicuitkan oleh masyarakat ditunjukkan pada Gambar 3.1 postingan positif dan Gambar 3.2 postingan negatif.

Gambar 3. 1 Postingan Positif



Gambar 3. 2 Postingan Negatif

Berdasarkan yang ditunjukkan oleh Gambar 3.1 terdapat kalimat "Pentingnya Vaksin dan Vaksin Terjamin Aman" mengandung makna positif pada kata "Aman". Gambar 3.2 terdapat sentimen negatif pada kalimat "Habis vaksin kok jadi demam panas dingin ya" pada kata "panas".

3.2. Lokasi dan Waktu Penelitian

Lokasi pada kegiatan penelitian dilaksanakan di Laboratorium Riset Universitas Buana Perjuangan Karawang dilaksanakan sejak bulan November sampai Mei 2022. Detail pelaksanaan penelitian ditunjukkan pada Tabel 3.1.

Tabel 3. 1 Waktu Penelitian

	November				Desember				Januari				Februari			
Kegiatan	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Studi Literatur																
Proposal																
Pengumpulan data																
Text preprocessing																

Model												
Analisis												
Sentimen												

	Maret				April				Mei			
Kegiatan	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
TF-IDF												
Klasifikasi												
Evaluasi												

3.3. Prosedur Penelitian

Tahapan prosedur penelitian dimulai mengumpulkan data yang diambil dari twitter. Kemudian data diolah dengan *Text Preprocessing* agar terstruktur dengan baik dan menghilangkan *noise,* lalu memunculkan Model. Setelah itu, data yang telah diolah akan dilakukan analisis sentimen untuk mendapatkan kelas positif, negatif atau netral, lalu melakukan pembobotan kata dan melakukan klasifikasi menggunakan KNN serta evaluasi. Tahapan prosedur penelitian digambarkan dengan *Flowchart* ditunjukkan Gambar 3.3.

Gambar 3. 3 Prosedur Penelitian

3.3.1. Pengumpulan Data

Pada tahap pengumpulan data dilakukan pada media sosial twitter. Karena media sosial twitter tempat masyarakat untuk mengeluarkan opini yang sedang terjadi. Maka dari itu, data diambil dari twitter dengan cara mengunduh dan mengumpulkan data tweet mengenai vaksinasi dari hasil pencarian mesin pencarian twitter pada bulan maret tahun 2022 dan menyimpannya menggunakan suatu program dengan Rstudio. Berikut Tabel 3.2 yang menjelaskan alur Algoritma Pengumpulan Data.

Tabel 3. 2 Algoritma Pengumpulan Data

Algoritma Pengumpulan Data.
Input: apikey, apisecret, acctoken dan tokensecret.
Proses: membaca token twitter developer
Output: kalimat

3.3.2. Text Preprocessing

Pada tahap *text preprocessing*, data yang telah diperoleh akan dilakukan proses penyeleksian data dengan tujuan menjadikan data tidak terstruktur menjadi data terstruktur. Terdapat beberapa tahapan pada *Text Preprocessing* yaitu *Case Folding*, *Tokenizing*, dan *Filtering* sebagai berikut:

a. *Case Folding*

Case Folding proses mengubah huruf kapital menjadi huruf nonkapital. Proses ini dilakukan untuk menyamaratakan penggunaan huruf yang terdapat di dalam dokumen. Contoh dari *Case Folding* dapat dilihat dari Tabel 3.3.

Tabel 3. 3 Contoh *Case Folding*

Sebelum	Sesudah
Vaksin Booster Jadi Syarat Mudik Lebaran 2022	vaksin booster jadi syarat mudik lebaran 2022

Dari contoh *Case Folding* Tabel 3.3 terdapat pada kalimat sebelum dan sesudah, pada kalimat sebelum itu, kalimat yang belum diproses dan kalimat sesudah, kalimat yang sudah diproses pada tahap *case folding*. Terdapat kata

"Vaksin" diawali dengan huruf kapital yaitu huruf "V" yang berubah menjadi huruf nonkapital "v" dan seterusnya. Berikut Tabel 3.4 yang menjelaskan alur Algoritma *Case Folding*.

Tabel 3. 4 Algoritma *Case Folding*

Algoritma *Case Folding*.
Input: Kalimat
Proses: - Baca kalimat - Rubah huruf kapital pada kalimat menjadi huruf nonkapital
Output: Kalimat menjadi huruf nonkapital

b. *Tokenizing*

Proses *tokenizing* yang dilakukan yaitu menghilangkan delimiter yaitu tanda baca dan symbol yang ada pada teks seperti @, $, &,#, tanda titik (.), tanda koma (,), tanda seru (!), tanda tanya (?). Contoh dari *Tokenizing* dapat dilihat dari Tabel 3.5.

Tabel 3. 5 Contoh *Tokenizing*

Sebelum	Sesudah
Vaksinasi COVID-19 Anak 6-11 Tahun Aman, Pakar: Kalaupun Ada KIPI, Itu Ringan, sebelum mudik ingat vaksin Booster	vaksinasi covid19 anak 6 11 tahun aman pakar kalaupun ada kipi itu ringan sebelum mudik ingat vaksin booster

Dari Contoh *Tokenizing* Tabel 3.5 perubahan dari kalimat sebelum dan sesudah, dimana tanda baca atau symbol pada kalimat sebelum menghilang, ketika sesudah diproses. Huruf dari *Tokenizing* semua nonkapital karena lanjutan dari tahapan *Case Folding*. Berikut Tabel 3.6 yang menjelaskan alur algoritma *tokenizing.*

Tabel 3. 6 Algoritma *Tokenizing*

Algoritma *Tokenizing*.
Input: Kalimat
Proses: - Membaca kalimat - Menghilangkan tanda baca atau karakter lainnya.
Output: Kalimat tidak ada tanda baca dan karakter lainnya.

c. Filtering

Filtering tahapan yang dilakukan menggunakan algoritma *stopword* (membuang kata kurang penting) dan *wordlist* (menyimpan kata penting). *Stopword* kata yang termasuk ke dalam kamus *stopword* akan dibuang dan tidak digunakan pada proses selanjutnya. Sedangkan *wordlist* kata yang tidak termasuk kedalam *stopword* akan digunakan pada proses selanjutnya. Contoh dari *Filtering* dapat dilihat dari Tabel 3.7.

Tabel 3. 7 Contoh *Filtering*

Sebelum	Sesudah
Kapolres Pasaman Yang Diwakili Wakapolres Ikuti Zoom Meeting Vaksinasi Serentak Dengan Kabaintelkam Polri	kapolres pasaman diwakili wakapolres ikuti zoom meet vaksinasi serentak kabaintelkam polri

Dari contoh *Filtering* Tabel 3.7 kalimat sebelum dan setelah sesudah diproses, terdapat kata yang hilang yaitu pada kata “Dengan”. Maka dari itu, kata yang hilang memiliki kata yang kurang penting pada kalimat. Berikut Tabel 3.8 yang menjelaskan alur Algoritma *Filtering.*

Tabel 3. 8 Algoritma *Filtering*

Algoritma *Filtering*.
Input: Kalimat
Proses: - Baca kalimat - Menghilangkan kata tidak penting
Output: Kalimat membuang kata tidak penting

3.3.3. Model

Model yang dihasilkan yaitu berbentuk *Wordcloud* dan histogram kata sering muncul. *Wordcloud* sebuah represintasi data yang berbentuk teks dan juga dapat berguna dalam berbagai konteks untuk memvisualisasikan data teks secara menarik. *Wordcloud* biasanya digunakan untuk menampilkan data teks atau kata yang sering muncul. Berikut Tabel 3.9 yang menjelaskan alur Algoritma Model.

Tabel 3. 9 Algortima Model

Algoritma Model.
Input: kalimat
Proses: - Membaca kalimat - Memisahkan kalimat menjadi kata
Output: Memuncul kata sering muncul

3.3.4. Analisis Sentimen

Pada proses ini digunakan untuk menentukan sentimen lalu mengelompokkan teks ke dalam beberapa kelas, yaitu positif, negative, dan netral. Berikut Tabel 3.4 yang menjelaskan alur Algoritma Analisis Sentimen.

Tabel 3. 10 Algoritma Analisis Sentimen

Algoritma Analisis Sentimen.
Input: kata
Proses: Membaca kata positif, negative, dan neutral
Output: Memuncul kata sentimen

3.3.5. Pembobotan Kata

Tahapan selanjutnya adalah menghitung bobot dari setiap *term* atau kata berdasarkan frekuensi kemunculan *term* tersebut dalam dokumen menggunakan metode TF-IDF. TF-IDF sebuah statistik numerik yang dapat menunjukkan kata kunci dengan kata tertentu. Selain dari itu, TF-IDF dapat mengetahui kata apa yang sering muncul pada suatu dokumen. Berikut Tabel 3.4 yang menjelaskan alur Algoritma TF-IDF.

Tabel 3. 11 Algoritma TF-IDF

Algoritma TF-IDF.
Input: Kata
Proses: Membobotkan kata
Output: Memuncul bobot nilai pada setiap kata

3.3.6. Klasifikasi Menggunakan Algoritma KNN

Tahap klasifikasi melakukan pengelompokkan *itemset* berdasarkan bobot nilai setiap *itemset*. Algoritma KNN berguna untuk melakukan klasifikasi pada suatu data pembelajaran (*train data sets*). Jadi, k diambil dari tetangga terdekat nya (*nearest neighbors*). Ada beberapa tahapan algoritma KNN, sebagai berikut:

a. Menentukan nilai K.
b. Menentukan nilai K yang paling besar.
c. Mengurutkan dari nilai K tinggi ke rendah.
d. Memprediksikan kategori objek dengan menggunakan kategori *nearest neighbors* yang paling mayoritas.

3.3.6. Evaluasi

Pada proses terakhir yaitu evaluasi yang menggunakan *confusion matrix* untuk melakukan tahapan pengujian. Proses evaluasi menggunakan *confusion matrix* banyak digunakan pada penelitian yang menggunakan algoritma K-NN. Proses evaluasi berguna untuk menghitung tingkat akurasi algoritma. Selain tingkat akurasi, dengan *confusion matrix* juga dapat menghitung nilai *precision*, dan *recall*. Pengujian dilakukan dengan menghitung *accuracy*, *precision* dan *recall*.

a. *Accuracy*

Menggambarkan seberapa akurat model dengan mengklasifikasikan dengan benar.

$$\frac{TP}{Jumlah\ Data} \quad (2)$$

b. *Precision*

Menggambarkan akurasi data yang diminta dan hasil prediksi yang diberikan oleh model.

$$\frac{TP}{TP+FP} \quad (3)$$

c. *Recall*

Menggambarkan keberhasilan model dalam menemukan sebuah informasi.

$$\frac{TP}{TP+FN} \quad (4)$$

Keterangan:

TP = *True* Positif (memprediksi positif dan itu benar)

TN = *True* Negatif (memprediksi negatif dan itu benar)

FP = *False* Positif (memprediksi positif dan itu salah)

FN = *False* Negatif (memprediksi negatif dan itu salah)