# BAB III

# METODOLOGI PENELITIAN

## Data Penelitian

*Dataset* atau data yang digunakan dalam penelitian ini bersumber dari Twitter berupa data teks kicauan (*tweet)* yang diperoleh dari tanggal 1 Desember 2020 hingga 31 Januari 2021. Data tersebut diperoleh menggunakan pustaka Tweepy melalui proses *crawling*. *Dataset* yang diperoleh dikumpulkan berdasarkan beberapa parameter kata kunci yang terkait dengan sistem pembelajaran daring antara lain: X, Y dan Z. Berikut beberapa contoh data *tweet* hasil dari proses *crawling* yang dapat dilihat pada Tabel 3.1 berikut:

**Tabel 13.1 Sampel data tweet**

| *Tweet ID* | *Username* | *Tweet* | Waktu *tweet* |
| --- | --- | --- | --- |
| 1335989354792103936 | LRomdani | Tetap memakai masker meski dirumah sendiri  Tetap semangat belajar dari rumah dimasa pandemi  #DiktiMengajarDariRumah  #DiktiDutaEdukasiPerubahanPrilaku https://t.co/c1WMa5SVSj | 2020-12-07 16:47:38 |
| 1336520460255724032 | kelaskitadotcom | Gunakan hak suara kamu dengan bijak, ya! Selamat memilih! #kelaskita #carabarubelajarseru #belajardirumah #elearning #belajaronline #dirumahaja #quotes https://t.co/1anyTiETlA | 2020-12-09 03:58:03 |
| 1336742494122340096 | fandimas16 | @collegemenfess 1. Jenuh banget di rumah 2. Gw dri dulu suka ama suasana kelas, dan suasana itu mendukung gw untuk belajar dan memahami suatu materi | 2020-12-09 18:40:20 |
| 1338003730587812096 | kumparan | Tanpa smartphone di masa pandemi, bisa berarti putus sekolah, karena kini dilakukan belajar online atau pembelajaran jarak jauh. https://t.co/rVW6xOgrfI | 2020-12-13 06:12:02 |

## Penerapan Metode

Dalam membangun aplikasi analisis sentimen yang dilakukan pada penelitian ini, terdapat beberapa tahapan yang dilakukan. Tahapan tersebut merepresentasikan setiap proses dan rancangan dalam penelitian, dari awal hingga akhir aplikasi berjalan. Tahapan yang dilakukan dapat dilihat pada Gambar 3.1 berikut:



**Gambar 13.1 Tahapan metode**

Pada Gambar 3.1, pengumpulan data dilakukan melalui proses *crawling* untuk mendapatkan *dataset* berupakicauan (*tweet*). Selanjutnya, *tweet* yang telah diperoleh dalam bentuk *excel*, kemudian dimasukkan kedalam basis data (*database*) untuk dilakukan proses *preprocessing*, pada proses *preprocessing* dilakukan penyaringan, pembuangan dan perbaikan kata. Hasil dari proses *preprocessing* menghasilkan kalimat yang lebih terstruktur (*clean text*) yang kemudian digunakan pada tahap selanjutnya. *Clean text* yang diperoleh dari proses *preprocessing* akan diproses dalam tahap *labeling* untuk menentukan kelas (*label*) berupa sentimen positif atau negatif, kemudian *tweet* yang telah berlabel akan dibagi menjadi dua (2) buah bagian antara lain: data uji dan data latih. Data latih merupakan data yang berfungsi sebagai pembangun pengetahuan untuk proses klasifikasi, proses pembangunan pengetahuan tersebut dilakukan melalui proses *modeling* dan menghasilkan sebuah model latih menggunakan data latih yang tersedia. Sementara data uji merupakan data yang disiapkan untuk menguji tingkat keakuratan model latih yang dihasilkan oleh proses *modeling*, proses pengujian tingkat akurasi tersebut dilakukan melalui proses pengujian dan evaluasi. Setelah dilakukan proses pengujiandan evaluasi model latih menggunakan data uji yang tersedia, hasil pengujian tersebut dipaparkan dalam bentuk persentase dan grafik.

### Pengumpulan Data

Pada tahapan pengumpulan data dilakukan melalui proses *crawling*. Proses tersebut meliputi: mendapatkan API *key* Twitter melalui akun *developer* Twitter (<https://developer.twitter.com/>). API *key* Twitter yang diperoleh antara lain: *Customer* API *key*, *Customer* API *Secret key*, *Access Token*, dan *Access Token Secret*. Proses selanjutnya adalah penambangan data yang bersumber pada media sosial Twitter menggunakan pustaka Tweepydengan akses dari API *key* yang telah didapatkan. Data *tweet* yang berhasil di kumpulkan akan disimpan ke dalam sebuah file *excel (.xlsx)*, yang kemudian dimasukkan ke dalam basis data (*database*) MySQL. Ilustrasi tahap pengumpulan data dapat dilihat pada Gambar 3.2 berikut:



**Gambar 23.2 Tahap pengumpulan data**

### *Preprocessing*

Pada tahapan *preprocessing* dilakukan penyaringan, pembuangan dan perbaikan kata melalui beberapa proses. Hal tersebut dimaksudkan untuk menghasilkan data kicauan (*tweet*) yang lebih terstruktur atau disebut dengan *clean text.* Berdasarkan terori yang terlah dijelaskan pada sub bab (2.5), proses yang dilakukan dalam tahap *preprocessing* antara lain*: casefolding, cleansing, mengubah slang word, menghapus stop word, dan stemming.*

#### *Casefolding*

Pada Gambar 3.3 proses *casefolding* dilakukan penyetaraan teks menjadi huruf kecil secara keseluruhan, misalnya: ‘Kalian’ akan diubah menjadi ‘kalian’, ‘PASTI BISA’ akan diubah menjadi ‘pasti bisa’, dan seterusnya.



**Gambar 33.3 Proses casefolding**

#### *Cleansing*

Pada proses *cleansing* dilakukan penyaringan dan pembuangan teks yang untuk proses analisis. Proses *cleansing* terdiri atas beberapa tahapan antara lain: menghapus URL, *mention* (*@mention*), *hastag* (*#hastag*), angka (0-9), tanda baca, dan spasi berlebih.

##### Menghapus URL

Pada Gambar 3.4 proses penghapusan URL akan menghapus semua teks yang diawali dengan ‘http’, karena dianggap kurang memiliki makna namun sering disisipkan dalam sebuah kicauan (*tweet*).

****

**Gambar 43.4 Proses menghapus URL**

##### Meghapus *mention* (*@mention*)

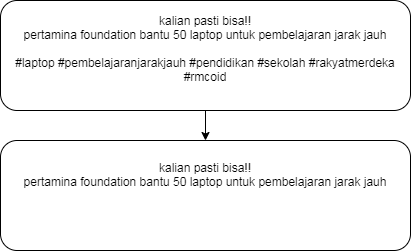
Pada Gambar 3.5 proses penghapusan *mention* (*@mention*) akan menghapus semua teks yang diawali dengan ‘@’.



**Gambar 53.5 Proses menghapus mention**

##### Menghapus *hastag* (*#hastag*)

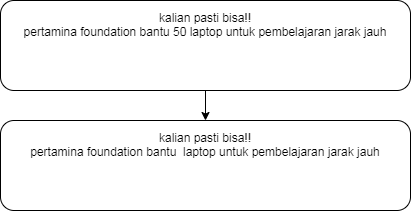
Pada Gambar 3.6 proses penghapusan tagar atau *hastag* (#hastag) akan menghapus semua teks yang diawali dengan ‘#’.



**Gambar 63.6 Proses menghapus hastag**

##### Menghapus angka

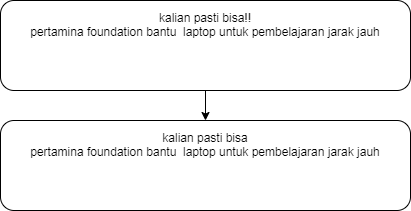
Pada Gambar 3.7 proses penghapusan angka akan menghapus angka (0-9) pada teks, misalnya: ‘bantu 50 laptop’ menjadi ‘bantu laptop’.



**Gambar 73.7 Proses menghapus angka**

##### Menghapus tanda baca

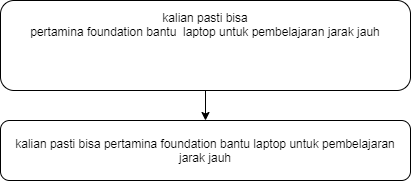
Pada Gambar 3.8 proses penghapusan tanda baca akan menghapus tanda baca pada teks, misalnya: ‘bisa!!’ menjadi ‘bisa’.



**Gambar 83.8 Proses menghapus tanda baca**

##### Menghapus spasi berlebih

Pada Gambar 3.9 proses penghapusan spasi atau (*whitespace)* berlebih akan menghapus baris dan *whitespace* yang lebih dari satu diantara kata, misalnya: ‘bantu laptop’ menjadi ‘bantu laptop’.



**Gambar 93.9 Proses menghapus spasi berlebih**

#### Merubah *slang word*

Pada Gambar 3.10 proses merubah *slang word* akan merubah setiap kata gaul, kata singkatan atau kata tidak baku ke bentuk bakunya, misalnya: ‘utk’ menjadi ‘untuk’, ‘yng’ menjadi ‘yang’ dan seterusnya. Proses pengubahan tersebut melibatkan kamus *slang word* yang terdapat dalam basis data (*database*).



**Gambar 103.10 Proses merubah slang word**

#### Menghapus *stop word*

Pada Gambar 3.11 proses menghapus *stop word* akan menghapus setiap kata yang kurang memiliki makna namun sering dijumpai dalam sebuah teks, misalnya kata: ‘untuk’ ‘yang’, dan seterusnya. Proses penghapusan tersebut melibatkan kamus *stop word* yang terdapat dalam basis data (*database*).



**Gambar 113.11 Proses menghapus stop word**

#### *Stemming*

Pada Gambar 3.12 proses *stemming* akan mengubah kata berimbuhan menjadi kata dasar dengan melibatkan pustaka Sastrawi, misalnya kata: ‘membantu’ menjadi ‘bantu’, ‘pembelajaran’ menjadi ‘ajar’, dan seterusnya.



**Gambar 123.12 Proses stemming**

### *Labeling*

Pada tahapan *labeling* dilakukan pemberian *label* (kelas) berdasarkan ciri atau karakteristik yang terkandung dalam sebuah dokumen atau kalimat. Pada tahap ini, t*weet* yang telah melalui proses *preprocessing* dan menghasilkan *clean text* akan diberikan kelas positif atau negatif. Kelas positif dimaksudkan untuk teks *tweet* tersebut yang mengandung pernyataan yang setuju, mendukung atau menerima proses berjalanya pembelajaran daring. Sedangkan kelas negatif dimaksudkan untuk teks *tweet* yang cenderung menyangkal, menolak atau menampik proses berjalanya pembelajaran daring.

Berdasarkan teori yang telah dijelaskan dalam sub bab (2.6), bahwa proses *labeling* dapat dilakukan dengan dua (2) buah cara, antara lain: pelabelan manual dan pelabelan dengan kamus sentimen. Pelabelan manual merupakan proses pemberian kelas berdasarkan subjektifitas peneliti terhadap sebuah kalimat secara satu per satu. Sedangkan pelabelan dengan kamus sentimen merupakan proses pemberian kelas secara otomatis berdasarkan kamus sentimen, di mana prosesnya melibatkan kamus kata positif dan kamus kata negatif yang terdapat dalam basis data (*database*). Pada Tabel 3.2 berikut berisi proses perhitungan skor *labeling* dengan kamus sentimen:

**Tabel 23.2Proses perhitungan skor**

| ***Dataset (clean text)*** | **Kata Positif** | **Kata Negatif** |
| --- | --- | --- |
| semangat teman teman ikut aktif bantu dalam proses ajar jarak jauh | semangat (1)  teman (2)  ikut (1)  aktif (1)  bantu (1)  proses (1)  ajar (1) | semangat (1)  jauh (1) |
| **Jumlah** | 8 | 2 |

Berdasarkan Tabel 3.2, menggunakan persamaan (2.1) maka dapat diperoleh perhitungan skor untuk *tweet* ‘semangat teman teman ikut aktif bantu dalam proses ajar jarak jauh’ yaitu sebagai berikut:

skor = (jumlah kata positif) – (jumlah kata negatif)

skor = 8 – 2

skor = 6

Setelah diketahui nilai skor, proses selanjutnya adalah pemberian kelas sentimen berdasarkan aturan sebagai berikut:

if skor > 0:

kelas = 'positif'

elif skor < 0:

kelas = 'negatif'

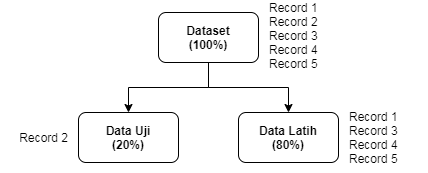
else:

continue

Maka dapat disimpulkan bahwa *tweet* ‘semangat teman teman ikut aktif bantu dalam proses ajar jarak jauh’ akan memiliki kelas positif, karena nilai skor > 0. Dengan demikian proses *labeling* dengan cara pendekatan menggunakan kamus sentimen akan menghasilkan sebuah kelas berdasarkan pada jumlah kata paling dominan (bermuatan positif atau negatif) dalam sebuah kalimat.

### Pemisahan data

Pada tahapan pemisahan data, *tweet* yang telah berlabel akan dibagi menjadi dua (2) buah bagian antara lain: data uji dan data latih. Proses pemisahan data dilakukan dengan membagi *dataset* secara acak menjadi 80% data latih dan 20% menjadi data uji. Ilustrasi tahap pemisahan data dapat dilihat pada Gambar 3.13 berikut:



**Gambar 133.13Tahap pemisahan data**

#### Data Latih

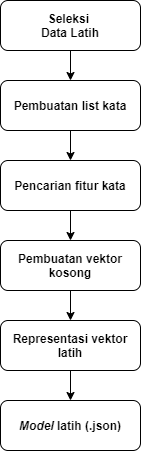
Data latih merupakan data yang berfungsi sebagai pembangun pengetahuan untuk proses klasifikasi.

#### Data Uji

Data uji merupakan data yang disiapkan untuk menguji tingkat keakuratan algoritme klasifikasi berdasarkan pengetahuan dari data latih.

### *Model*ing

Tahap modeling dilakukan untuk menekstraksi *tweet* data latih menjadi representasi vektor menggunakan *CountVectorizer*. Pada tahap ini terdapat lime (5) proses utama yang dilalui untuk menghasilkan sebuah *model* latih, lima proses itu antara lain: seleksi data latih, pembuatan list kata, pencarian fitur kata, pembuatan vektor kosong dan membuat representasi vektor. Ilustrasi proses tersebut dapat dilihat pada Gambar 3.14 berikut:



**Gambar 143.14 Proses modeling**

Pada Gambar 3.14, penyeleksian data latih dilakukan dengan menggunakan teknik sampling kuota (*quota sampling*), dimaksudkan untuk mendapatkan keseimbangan pada data latih. Tahapan seleksi tersebut dilakukan secara acak sebanyak kriteria tertentu. Data latih telah terseleksi akan diproses (ekstraksi fitur) menggunakan *CountVectorizer* sehingga dapat diperoleh daftar fitur dan representasi vektor angka untuk tiap data latih. Daftar fitur dan vektor angka tersebut kemudian akan disimpan dan dijadikan sebagai sebuah *model* pengetahuan (*model* latih) dalam bentuk file JSON (.json).

### Klasifikasi *K-nearest neighbors*

Tahap klasifikasi menggunakan *K-Nearest Neighbors* (KNN) merupakan tahapan yang dapat dilakukan setelah terdapat satu atau lebih *model* latih. *Model* latih tersebut merupakan data latih yang telah melalui tahap *modeling* yang dijelaskan pada sub-sub bab (3. 2. 5). Untuk menerapkan *model* klasifikasi menggunakan KNN, terdapat tiga (4) buah proses utama yaitu: Membuat representasi vektor uji, menghitung jarak antar data, mencari tetangga terdekat berdasarkan nilai K, dan menghitung nilai probabilitas *label* sentimen. Ilustrasi tahapan klasifikasi dapat dilihat pada Gambar 3.15 berikut:



**Gambar 153.15 Proses klasifikasi**

Pada Gambar 3.15, pembuatan representasi vektor uji dilakukan menggunakan *model* latih yang dipilih, sehingga terbentuk representasi vektor uji yang sesuai dengan pengetahuan *model* latih. Hasil vektor uji tersebut akan dihitung tingkat kedekatanya (jarak) dengan vektor pada *model* latih, proses tersebut melibatkan perhitungan *euclidean distance*. Hasil perhitungan *euclidean distance* akan menghasilkan nilai jarak, yang kemudian akan disaring berdasarkan K tetangga terdekanya. Selanjutnya dilakukan *voting* untuk menentukan *label* prediksi (positif atau negatif) bedasarkan dominasi *label* pada *K* tetangga terdekatnya.

## Rancangan Pengujian

Pengujian dilakukan untuk mengetahui nilai atau tingkat akurasi, presisi, dan *recall* dari *model* latih menggunakan algoritme yang diusulkan. Pada penelitian ini, pengujian dilakukan dengan cara membandingkan beberapa data hasil prediksi (data hasil tahap klasifikasi) dengan sekumpulan data aktual (data hasil tahap *labeling*). Adapun dimaksud dengan beberapa data hasil prediksi merupakan sekumpulan data yang telah diproses melalui algoritme *K-Nearest Neighbors* (KNN) dengan variasi nilai K, yaitu: K=3, K=5, K=7, K=9, dan K=11.

### Akurasi

Akurasi merupakan tingkat kedekatan antara nilai prediksi dengan nilai aktual persamaan (3. 1).

### Presisi

Presisi merupakan tingkat ketepatan antara informasi yang diminta dengan jawaban yang diberikan oleh sistem persamaan (3. 2).

### *Recall*

*Recall* merupakan tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi persamaan (3. 3).

Berdasarkan pada sub bab (2. 10), pengukuran tingkat akurasi, presisi, dan *recall* dapat diketahui melalui *confusion matrix* dengan persamaan sebagai berikut:

|  |  |
| --- | --- |
|  | Rumus 1(3. 1) |
|  | Rumus 2(3. 2) |
|  | Rumus 3(3. 3) |

dengan,

1. *True Positive* (TP) merupakan data positif yang diprediksi benar. Misalnya: *tweet* 1 ber*label* positif dan dari *model* latih yang dibuat memprediksi *tweet* 1 bernilai positif juga.
2. *True Negative* (TN) merupakan data negatif yang diprediksi benar. Misalnya: *tweet* 1 ber*label* negatif dan dari *model* latih yang dibuat memprediksi *tweet* 1 bernilai negatif juga.
3. *False Postive* (FP) merupakan data negatif namun diprediksi sebagai data positif. Misalnya: *tweet* 1 ber*label* negatif namun dari *model* latih yang dibuat memprediksi *tweet* 1 bernilai positif.
4. *False Negative* (FN) merupakan data positif namun diprediksi sebagai data negatif. Misalnya: *tweet* 1 ber*label* positif namun dari *model* latih yang dibuat memprediksi *tweet* 1 bernilai negatif.