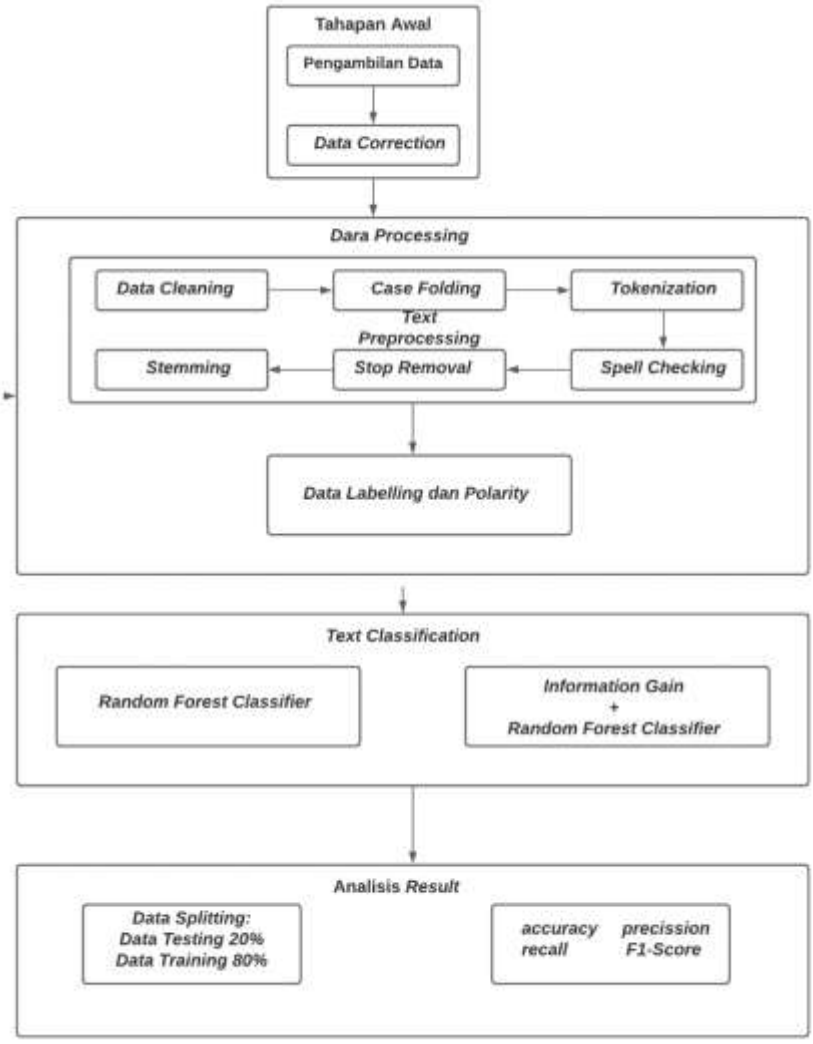
BAB III

METODOLOGI PENELITIAN

Metodologi penelitian merupakan langkah untuk memperoleh data yang diproses menjadi informasi lebih akurat sehingga dapat menjadi pedoman dalam pelaksanaan penelitian agar hasil tidak menyimpang dari tinjauan yang akan dicapai. Berikut metodologi penelitian yang digunakan pada penelitian ini dapat dilihat pada Gambar 3.1 berikut ini :



Gambar 3.1 Metodologi Penelitian

Metodologi penelitian ini bertujuan untuk menguraikan seluruh kegiatan yang akan dilakukan selama penelitian berlangsung. Terdapat empat proses utama dalam penelitian ini, yaitu proses pada tahap awal, *data processing*, *text classification*, dan analisis *result*. Berikut adalah penjelasan dari proses-proses tersebut :

3.1 Tahapan Awal

Pada tahapan awal terbagi menjadi dua proses yang akan dilakukan sebagai berikut :

1. Pengumpulan Data

   Penelitian ini diawali dengan pengumpulan data dengan menggunakan teknik *data crawling* pada *twitter* untuk mengumpulkan data berupa *tweet* yang akan dipakai pada penelitian ini dengan menggunakan akses dari pihak *twitter* bagi pengguna dengan memanfaatkan *Twitter API*, sehingga dapat dengan mudah memperoleh data-data seperti *tweet*, data pengguna, dan lainnya. Dimana pada penelitian ini data *tweet* yang diambil berdasarkan *hashtag* #VaksinasiNasional dan #VaksinCOVID19.

2. *Data Correction*

   Proses koreksi ini dilakukan di *MS.Excel* dengan mengimport *file CSV*, dengan ini format data akan berubah menjadi *xslx* dari format *csv* untuk mempermudah pengoreksian data. Proses koreksi dilakukan dengan membuang atribut yang tidak digunakan dan penghapusan data yang terdeteksi duplikasi. Format data yang diperoleh dari proses *crawling* yang dilakukan pada Bahasa Pemrograman *Python* yaitu berformat *CSV (Comma*

Separated Values), yaitu format data dimana setiap *record* dipisah oleh tanda koma(,), tanda petik dua(“), ataupun pembatasan *(delimiter)* yang dipakai pada umumnya.

3.2 *Data Processing*

Data processing yaitu mengolah data yang di dapat dari *data crawling* yang bertujuan untuk pengontrolan ukuran daftar kata dan diharapkan dapat meningkatkan performa penemuan kembali informasi dengan menggunakan enam tahap *text preprocessing* serta proses *data labelling* dan *polarity* yang akan dijelaskan sebagai berikut :

3.2.1 *Text Preprocessing*

1) *Data Cleaning*

Pada tahap ini data yang telah didapat pada tahapan awal akan dilakukan pembersihan kalimat dan menghilangkan tanda baca dari kalimat.

2) *Case Folding*

Pada tahap ini akan dilakukan pemeriksaan ukuran setiap karakter dari awal sampai akhir karakter dan jika ditemukan karakter menggunakan huruf kapital (*uppercase*), maka huruf tersebut akan diubah menjadi huruf kecil (*lowercase*).

3) *Tokenization*

Pada tahap ini akan dilakukan pembagian kalimat menjadi perkata (merubah kalimat dan teks menjadi sebuah token-token) untuk menghapus kata-kata yang tidak penting pada tahap *stop removal*.

4) *Spell Checking*

Tahap ini dilakukan untuk menghilangkan kata-kata yang menggangu pada teks dan juga membenahi kata-kata yang berbentuk singkatan ataupun *typo* menjadi kata yang sesuai dengan kamus KBBI.

5) *Stop Removal*

Pada tahap ini dilakukan penghapusan kata-kata yang terlalu umum dan kurang penting yang memiliki frekuensi kemunculan yang jumlahnya cukup banyak dibandingkan dengan kata yang lainnya. Tahap ini juga membuang kata-kata yang tidak deskriptif.

6) *Stemming*

Pada tahap ini dilakukan untuk merubah kata menjadi ke bentuk dasarnya yaitu dengan menghilangkan semua imbuhan kata pada kata turunannya.

3.2.2 *Data Labelling* dan *Polarity*

1) Data Labelling

Proses ini dilakukan menggunakan library python yaitu *Textblob* untuk memberi label secara otomatis dengan memberikan *score* di setiap cuitan yang didapat dari *twitter* (telah melewati proses *text preprocessing*) yang menunjukkan bahwa cuitan tersebut termasuk dalam cuitan positif, negatif dan netral. Dimana *textblob* akan memberi *score* pada setiap label yaitu sebagai berikut:

a) Label positif memiliki *score* > 0
b) Label netral memiliki *score* == 0
c) Label negatif memiliki *score* < 0

2) Polarity

Proses ini dilakukan untuk mempermudah proses selanjutnya yaitu proses *text classification*. Metode yang digunakan pada *text classification* merupakan metode *machine learning* dimana metode tersebut tidak dapat melatih teks secara langsung sehingga data teks harus diubah menjadi numerik. Maka dilakukan proses *polarity* untuk mengkonvert label ke polaritas sebagai berikut:

a) Label positif memiliki nilai *polarity* = 1
b) Label netral memiliki nilai *polarity* = 0
c) Label negatif memiliki nilai *polarity* = -1

3.3 *Text Classification*

Pada tahap ini akan dilakukan proses *data training* dan *predicting sentiment* dengan menggunakan metode *machine learning* yaitu *random forest classifier* dimana hanya dapat melatih ataupun menguji data dalam tipe numerik oleh karena itu diperlukan *converting* data teks menjadi data numerik terlebih dahulu dengan menggunakan *library* ekstrasi fitur yaitu *count vectorizer* dan *tf-idf transformer* setelah itu pendefinisian variabel X dan Y untuk proses *data splitting* dengan membagi *data training* sebesar 80% dan *data testing* sebesar 20% yang akan digunakan pada tahap pemodelan klasifikasi teks yaitu sebagai berikut :

3.3.1 Klasifikasi teks menggunakan algoritma *Random Forest Classifier*

Proses ini dilakukan langkah-langkah bagaimana *Random Forest Classifier* untuk mengklasifikasi *dataset* sehingga dapat mengetahui cara

kerja dari metode *Random Forest Classifier* dan untuk mendapatkan hasil terakhir (pada tahap analisis *result*) dari polaritas sentimen dengan pendukung berupa data latih dan fitur acak yang *independent* dengan fitur yang berbeda-beda.

3.3.2 Klasifikasi teks menggunakan algoritma *Random Forest Classifier* dengan *Information Gain*

Proses *Random Forest Classifier* dengan *Information Gain* dilakukan untuk melihat kinerja dari *feature selection* berupa *Information Gain* pada pengoptimalan tingkat akurasi klasifikasi dari algoritma *Random Forest Classifier*. Dimana *Information Gain* akan menghitung keseluruhan atribut untuk mengetahui kepentingan atribut terhadap klasifikasi yang pada akhirnya setelah diketahui kepentingan dari semua atribut terhadap klasifikasi maka atribut yang kiranya tidak perlu dapat dihilangkan dari *dataset* dan *dataset* diperbaharui.

3.4 Analisis *Result*

Setelah melakukan klasifikasi data, proses selanjutnya yaitu proses evaluasi model dan memprediksi data yang telah diklasifikasi menggunakan metode *Random Forest Classifier*. Pada tahap ini akan dilakukan perbandingan performa dari metode *Random Forest Classifier* dengan *Information Gain* dan metode *Random Forest Classifier* tanpa *Information Gain*. Hasil yang didapatkan akan ditampilkan berupa *confusion matrix*, dari nilai *confusion matrix* ini di hitunglah nilai *accuracy, precission, recall,* dan *F1 score*. Sehingga dari hasil tersebut dapat dianalisis bahwa hasil dari kedua metode tersebut apakah memiliki perbedaan hasil

akurasi yang berbeda ataupun tidak berbeda dan juga dari hasil tersebut dapat memprediksi sentimen opini masyarakat terkait vaksinasi Covid-19.