

BAB III

METODOLOGI DAN PERANCANGAN SISTEM

3.1 Metodologi

Metode dan tahapan penelitian yang digunakan dalam penelitian ini akan dipaparkan sebagai berikut.

1. Telaah Literatur

Bentuk kegiatan pada tahapan ini adalah berupa pengumpulan berbagai bentuk informasi mulai dari buku, referensi, data, fakta ataupun jurnal-jurnal penelitian yang terkait dengan penelitian yang akan dilakukan. Tahapan telaah literatur ini menjadi tahapan yang paling awal adalah proses penelitian yang akan dilakukan.

2. Analisa Kebutuhan

Pada tahap ini, dilakukan analisis terhadap metode, data dan proses apa saja yang akan digunakan pada penelitian ini. Proses analisis pada algoritma *Naive Bayes* dilakukan mulai dari teori dan perhitungan yang akan digunakan ke dalam sistem yang dibuat. Pada tahap ini juga menganalisis hasil output yang akan dikeluarkan nantinya adalah positif, netral atau negatif.

3. Desain Sistem

Setelah mengumpulkan bahan literatur dan menganalisis kebutuhan apa saja yang akan digunakan pada penelitian ini, maka dapat dibuat sebuah rancangan atau gambaran kasar tentang sistem yang akan dibuat. Fungsionalitas sistem ditentukan pada tahap ini sesuai dengan spesifikasi yang sudah diumumkan sebelumnya.

4. Pemrograman Sistem

Pada tahap ini dilakukan proses implementasi dari rancangan yang sudah dilakukan pada tahap-tahap sebelumnya. Proses pembuatan sistem (*coding*) sesuai spesifikasi dan fungsionalitas yang telah dirancang dilaksanakan. Berbagai komponen yang dibutuhkan dalam sistem seperti tampilan antarmuka, modul program dikerjakan pada tahap ini.

5. Uji Coba dan Evaluasi

Sistem yang telah dibuat, diuji pada tahap ini untuk memastikan semua fungsionalitas yang telah ditentukan dapat berjalan dengan baik.

Evaluasi dilakukan dengan cara menghitung nilai *accuracy* dengan menggunakan Persamaan 1, nilai *precision* dengan menggunakan Persamaan 2, nilai *recall* dengan menggunakan Persamaan 3, dan nilai *F1-Score* dengan menggunakan Persamaan 4 berdasarkan data dari uji coba untuk memenuhi tujuan dari penelitian ini.

6. Konsultasi dan Penulisan

Penulisan laporan dilakukan pada tahapan ini dengan tujuan mendokumentasikan segala bentuk proses penelitian serta menyimpulkan hasil akhir yang didapat dari pengerjaan skripsi ini.

3.2 Perancangan Aplikasi

Perancangan sistem implementasi algoritma *Naive Bayes* untuk *feedback user* baik dari segi pelatihan dan pengenalan akan dijabarkan dalam *flowchart* dan rancangan antarmuka berikut ini.

3.2.1 Flowchart

Berikut *flowchart* yang akan digunakan dalam rancangan sistem. Terdapat *flowchart training*, *flowchart testing*, *flowchart preprocessing*, dan *flowchart naïve bayes*.

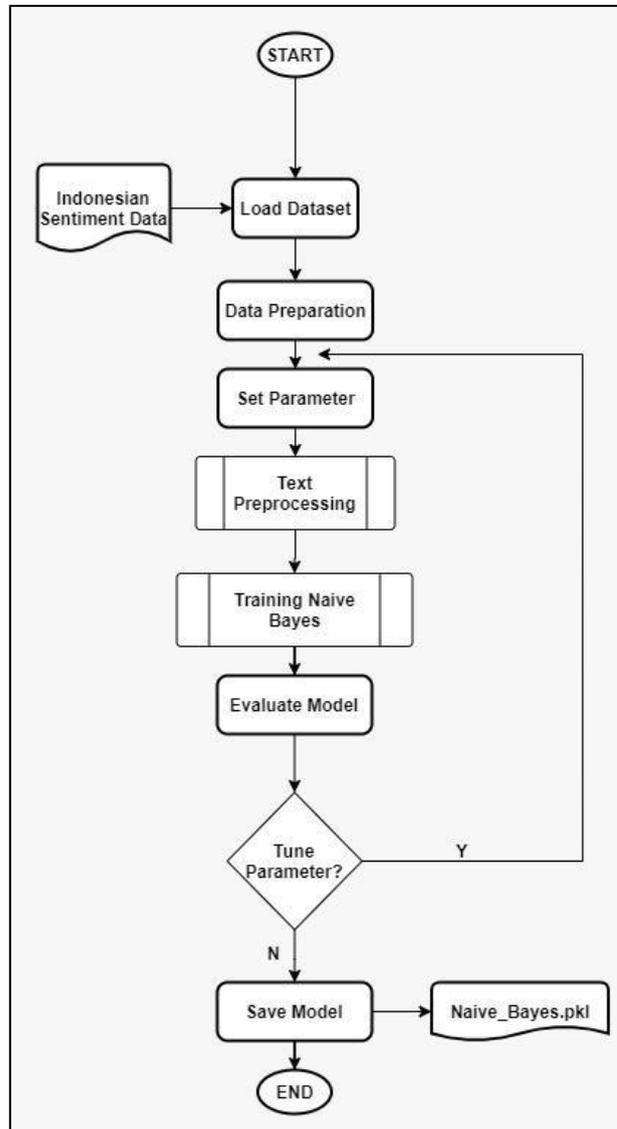
A. Flowchart Training

Pada proses training dalam implementasi *Naive Bayes* memiliki delapan tahap, yaitu *load* dataset berbahasa Indonesia. Lalu setelah tahap itu masuk pada tahap selanjutnya, yaitu *data preparation* berguna untuk merubah dataset ke bentuk dataframe rapi yang diinginkan. Tahap selanjutnya yaitu, *set parameter* untuk melakukan *tuning* yang berfungsi untuk meningkatkan akurasi dari algoritma *Naive Bayes*.

Kemudian masuk ke tahap *text preprocessing*, di tahap ini *input* akan disederhanakan menjadi kata yang baku dan membuang kata-kata yang tidak memiliki arti dan mengubah *input* ke dalam bentuk *vector* (angka). Lalu setelah tahap itu masuk ke tahap *training Naive Bayes*, untuk mencari *prior probability* yang berguna untuk memprediksi *probability* setiap klasifikasi.

Tahap evaluate model untuk melihat performa model dan performa dari implementasi algoritma *Naive Bayes*, jika belum memuaskan dapat dilakukan

tuning ulang pada tahap selanjutnya yaitu, *tune parameter*. Jika performa dan akurasi sudah cukup maka model tersebut akan di simpan pada tahap *save model*.

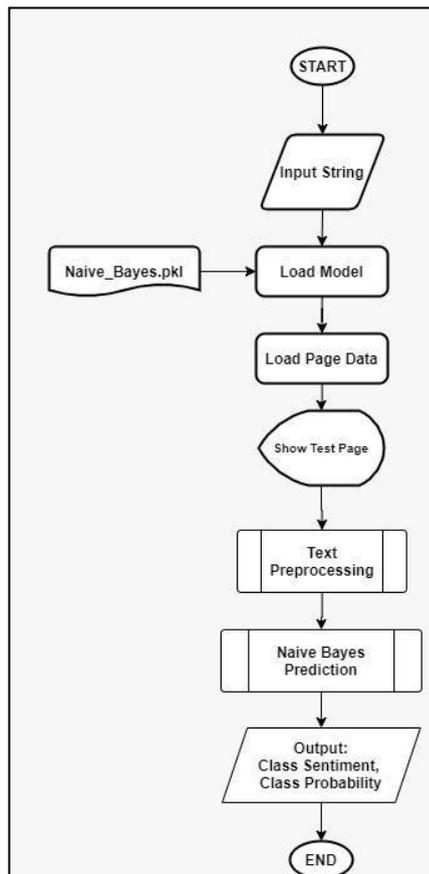


Gambar 3.1 Flowchart Training

B. Flowchart Testing

Proses *testing* memiliki enam tahap sebelum output dihasilkan, yaitu memasukkan kalimat pada tahap pertama, kemudian tahap selanjutnya adalah *load model*, *load page data* dan *show text page* yang mengambil data dari data *training* yang sebelumnya. Tahap selanjutnya adalah *text preprocessing* yang berguna untuk

menyederhanakan dan menyisakan kata-kata yang memiliki arti dari *inputan*, juga mengubah *input* ke dalam bentuk *vector* (angka). Lalu akan masuk ke tahap *Naive Bayes prediction*, memprediksi hasil dari *input*. Pada proses testing akan mengeluarkan hasil dari klasifikasi kalimat yang dimasukkan dan persentase setiap klasifikasi.



Gambar 3.2 Flowchart Testing

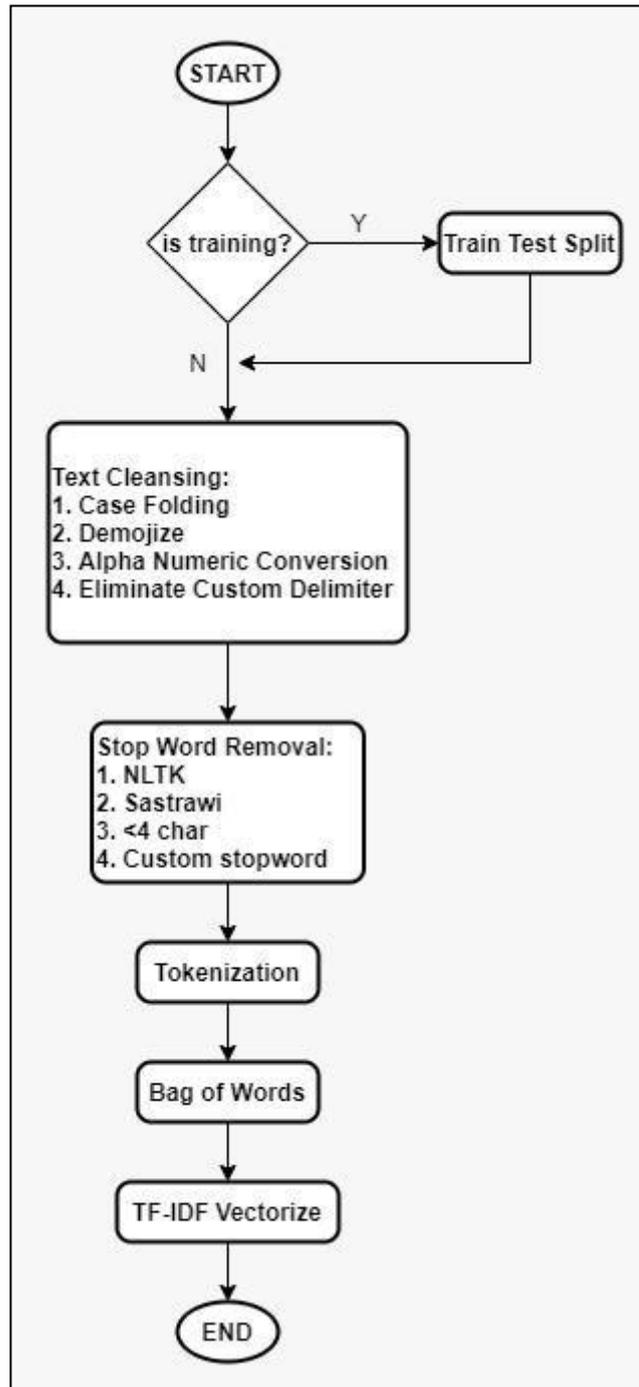
C. Flowchart Preprocessing

Text preprocessing yang digunakan baik pada proses *training* maupun *testing* memiliki proses yang sama, yaitu mengecek data tersebut merupakan *data testing* atau *data training* dahulu. Lalu masuk ke tahap *text cleansing* untuk menyederhanakan kalimat, *case folding* berfungsi untuk merubah semua kata yang

ada di kalimat tersebut menjadi *lowercase*, *demojize* untuk mengubah emoji menjadi *string* agar dapat dikenali oleh sistem, *alpha numeric conversion* menyisakan string dengan menghilangkan angka dan tanda baca, *eliminate custom delimiter* adalah tambahan untuk menghilangkan tanda baca yang tidak terbaca oleh *library*.

Tahap *stop word removal* menyisakan kata-kata yang memiliki arti, NLTK dan sastrawi merupakan *library* untuk melihat kata-kata apa saja yang mengandung arti atau memiliki nilai, kata-kata dibawah empat karakter akan dihilangkan karena tidak memiliki arti, *custom stop word* adalah tambahan untuk menghilangkan kata-kata yang tidak masuk ke dalam library seperti singkatan dan kata-kata ambigu.

Tahap *tokenization* untuk menjadikan kata-kata yang telah diproses sebelumnya ke dalam bentuk *array*. Tahap *bag of words* menggabungkan kata yang sama pada kalimat tersebut dan disimpan dalam bentuk angka berapa banyak jumlah kata tersebut pada kalimat yang dimasukkan. Tahap *TF-IDF Vectorize* mengubah kata yang telah di proses pada tahapan sebelumnya ke dalam bentuk *vector*.



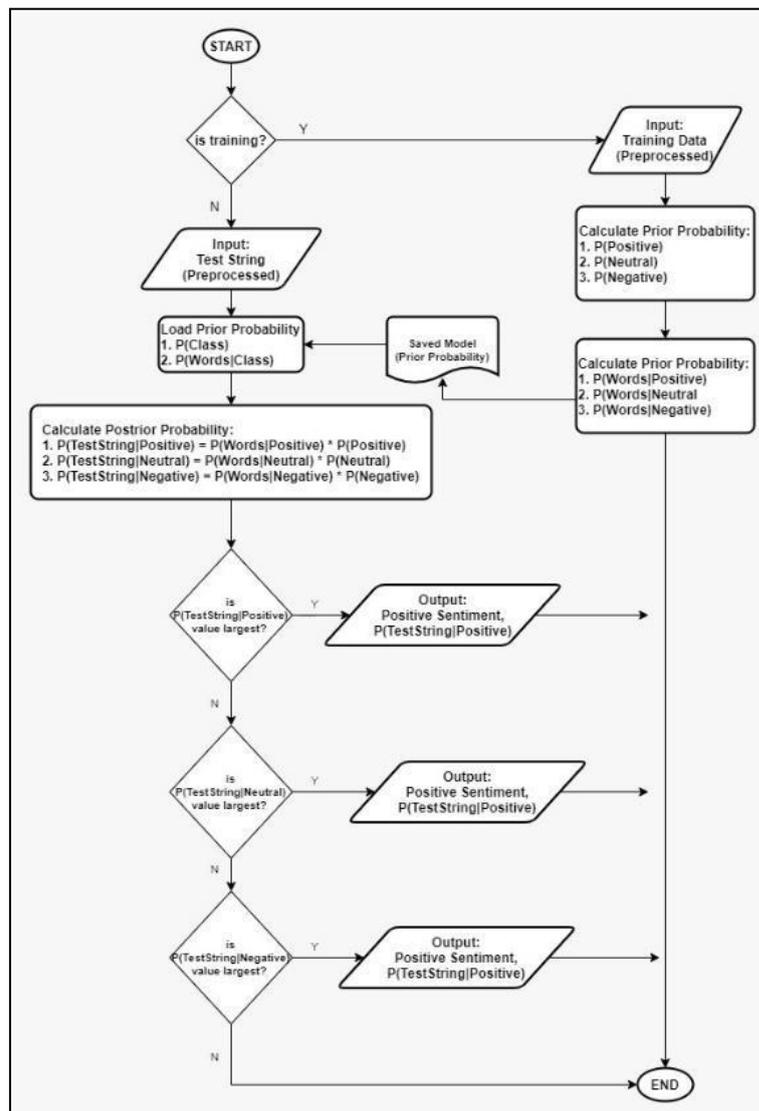
Gambar 3.3 Flowchart Preprocessing

D. Flowchart Naive Bayes

Flowchart Naive Bayes menjabarkan alur kerja dari algoritma *Naive Bayes*.

Pada algoritma *Naive Bayes* akan memproses pengkategorian klasifikasi ke dalam kategori positif, netral dan negatif. *Input* yang dimasukkan ke dalam sistem untuk

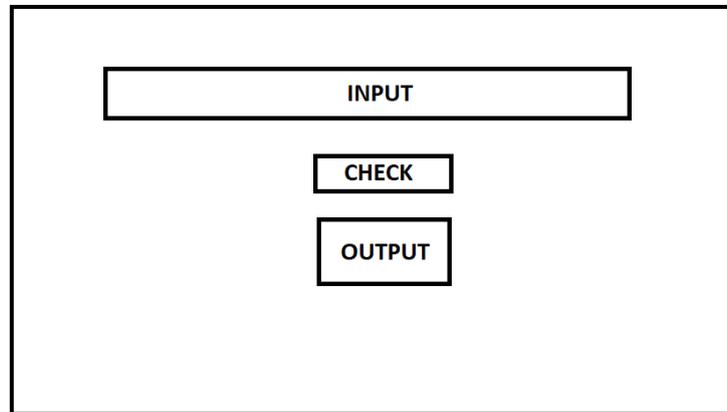
data *training*, maka kalimat tersebut akan dihitung probabilitasnya untuk setiap kategori yang ada dan menghitung probabilitas untuk setiap kata pada kategori yang ada. Tapi jika *input* yang dimasukkan merupakan data *testing*, maka akan dicari probabilitasnya berdasarkan probabilitas yang disimpan pada tahap training, kemudian akan dihitung persentase dari setiap kategori yang ada. Kategori dengan probabilitas terbesar akan menjadi hasil dari pengklasifikasian *Naive Bayes*.



Gambar 3.4 Flowchart Naive Bayes

3.2.2 Rancangan Antarmuka

Rancangan antarmuka dari implementasi algoritma *Naive Bayes* untuk analisis sentimen *feedback user* terdapat satu halaman utama.



Gambar 3.5 Rancangan Antarmuka

Pada Gambar 3.5 merupakan halaman utama dari implementasi algoritma *Naive Bayes*. Pada halaman ini terdapat tombol *check* untuk melihat hasil dari klasifikasi kalimat yang telah di *input*.