

## BAB 3 METODOLOGI PENELITIAN

### 3.1 Metodologi Penelitian

Dalam penelitian ini, ada beberapa metode yang digunakan, yaitu :

#### 1. Studi Literatur

Studi Literatur dilakukan untuk mendapatkan informasi dari berbagai sumber tentang cara mengapatkan data ulasan pada aplikasi di Google Play Store, mempelajari *Text Preprocessing*, *TF-IDF*, *K-Fold Cross Validation*, *Naïve Bayes*, dan *Confusion Matrix*.

#### 2. Perancangan Sistem

Perancangan sistem dilakukan untuk merancang alur dari sistem yang dibuat menggunakan *flowchart* berdasarkan studi literatur yang dilakukan.

#### 3. Pemrograman Sistem

Pemrograman sistem dilakukan dalam bentuk implementasi berupa *coding* berdasarkan perancangan sistem yang dibuat.

#### 4. Pengujian Sistem

Pengujian dilakukan untuk mengevaluasi kinerja sistem yang dibangun dan memastikan agar sistem yang diimplementasikan dapat berfungsi sesuai yang diharapkan. Hasil uji coba sistem juga dapat membantu untuk mengidentifikasi masalah dan kekurangan pada sistem sehingga dapat diperbaiki

#### 5. Evaluasi

Evaluasi dilakukan untuk mengukur performa atau tingkat akurasi dari sebuah sistem klasifikasi atau prediksi dengan menghitung metrik-metrik seperti *textitaccuracy*, *precision*, *recall*, *F1-score*, yang berguna untuk mengevaluasi dan memperbaiki sistem agar dapat memberikan hasil yang semakin baik dan akurat.

#### 6. Konsultasi dan Laporan

Konsultasi dilakukan dengan dosen pembimbing untuk menyempurnakan penelitian yang dibuat dan mendokumentasikannya dalam bentuk laporan

supaya penelitian ini dapat menjadi bagian dari kontribusi yang bermanfaat untuk pengembangan kedepannya.

## 3.2 Perancangan Sistem

Rancangan sistem pada penelitian ini mencakup Teknik Pengumpulan Data, *Flowchart Text Preprocessing*, *Data Labeling*, TF-IDF, dan *Flowchart Naive Bayes*.

### 3.2.1 Teknik Pengumpulan Data

Pengumpulan data pada penelitian ini menggunakan teknik *Scraping* dengan bantuan *library* google play scraper yang disediakan oleh Python untuk mengambil data ulasan aplikasi Haji Pintar di Play Store. Aplikasi Haji Pintar memiliki total 5.380 ulasan pengguna namun hanya terdapat 1.864 ulasan yang mengandung komentar teks, yang kemudian Penulis menentukan untuk mengambil 1.000 data ulasan pengguna untuk memudahkan dalam *text preprocessing*. Data yang digunakan merupakan data ulasan per tanggal 5 Juni 2023 sejumlah 1.000 data, yang kemudian disimpan dalam format CSV (*Comma Separated Values*).

### 3.2.2 Flowchart Text Preprocessing

Tahapan ini dilakukan setelah membaca data csv yang telah didapatkan dari *Scraping*, pada Gambar 3.1 berikut tahapan pada proses *Flowchart Text Preprocessing*:

1. *Data Cleaning*

Proses penghapusan data teks dari karakter selain huruf, termasuk data teks yang terdapat duplikasi.

2. *Case Folding*

Mengubah huruf pada data menjadi *lowercase*.

3. *Tokenization*

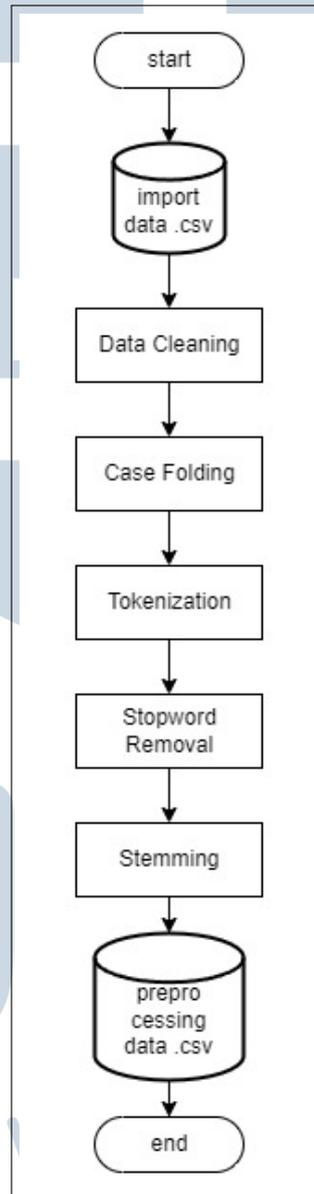
Menggunakan *library* nltk untuk memecah kalimat pada data teks menjadi satuan kata.

4. *Stopword Removal*

Proses ini menggunakan *library* nltk untuk menghapus kata yang tidak memiliki arti khusus seperti "dan", "atau", "yang", "di".

### 5. *Stemming*

Proses ini menggunakan *library* sastrawi yang menjadikan kata beriumbunan menjadi kembali ke kata dasarnya.



Gambar 3.1. *Flowchart Text Preprocessing*

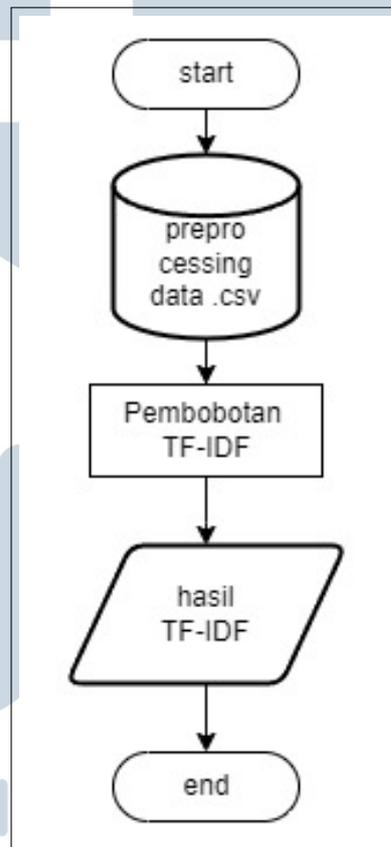
### 3.2.3 Data Labeling

*Data labeling* dilakukan dengan bantuan *library* TextBlob untuk menentukan sentimen Positif, Netral, dan Negatif. Data yang sudah dilakukan

proses *Case Folding*, selanjutnya akan diterjemahkan menjadi Bahasa Inggris secara manual dengan *google spreadsheet* untuk memudahkan TextBlob dalam memberikan label sentimen dikarenakan TextBlob belum bisa memberikan label sentimen pada data yang berbahasa Indonesia.

### 3.2.4 Flowchart Term Frequency - Inverse Document Frequency

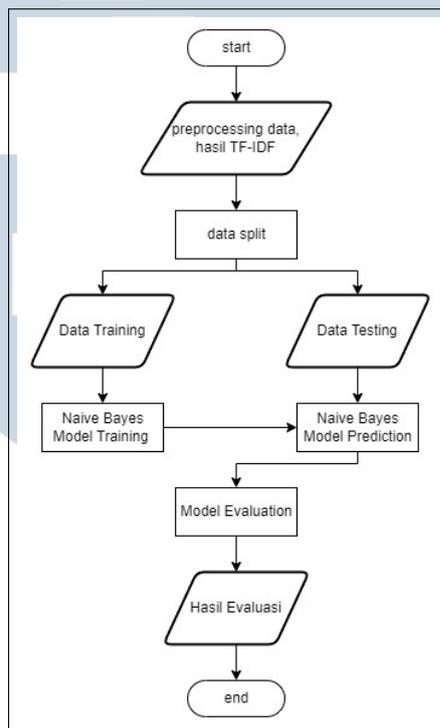
Pembobotan dilakukan dengan TF-IDF pada kata-kata dalam data hasil *Flowchart Text Preprocessing* berdasarkan frekuensi kemunculan kata tersebut dalam data dan seberapa umum kata didalam seluruh koleksi data. Proses ini akan mempelajari data dan menghasilkan matriks berbobot TF-IDF.



Gambar 3.2. *Flowchart* pembobotan *Flowchart* TF-IDF

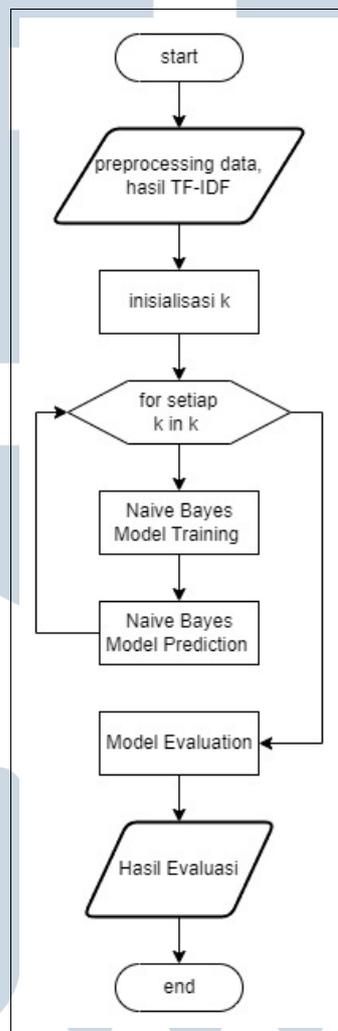
### 3.2.5 Flowchart Naive Bayes

Setelah dilakukan pembobotan TF-IDF, selanjutnya seperti Gambar 3.3 data di-*split* menjadi dua, *data training* digunakan untuk melatih model Naive Bayes dalam pengklasifikasian, kemudian model yang sudah dilatih akan diuji untuk memprediksi *data testing*. Kemudian model akan dievaluasi untuk dinilai keakuratannya dalam melakukan prediksi.



Gambar 3.3. Flowchart Naive Bayes dengan *Data Split*

Selain membaginya dengan *data split*, terdapat *K-Fold Cross Validation* yaitu metode untuk membagi data menjadi  $k$  *subset* yang disebut *fold*. Dalam setiap iterasi, salah satu *fold* menjadi *data testing* dan sisanya akan menjadi *data training*. Iterasi ini diulang sebanyak  $k$  seperti Gambar 3.4.



Gambar 3.4. Flowchart Naive Bayes dengan *K-Fold Cross Validation*

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA