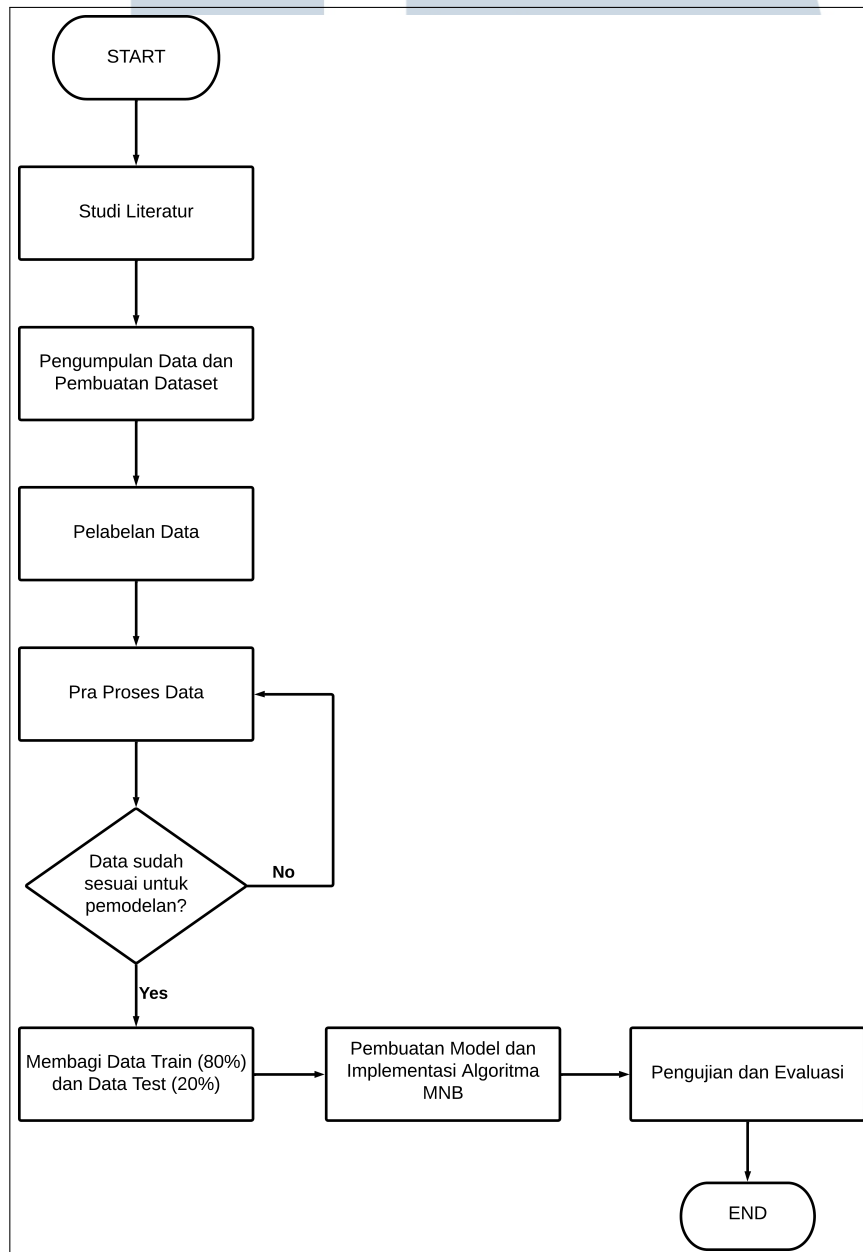


## BAB 3 METODOLOGI PENELITIAN

### 3.1 Metodologi Penelitian

Berikut merupakan *flowchart* yang menjabarkan tahapan yang akan dilakukan pada penelitian ini.



Gambar 3.1. Diagram Alur Metodologi Penelitian

### 3.1.1 Studi Literatur

Tahap pertama dalam penelitian ini adalah studi literatur. Pada tahap ini, pengerjaan penelitian akan berfokus pada pembelajaran teori dan metode yang berkaitan dengan analisis sentimen yang menggunakan algoritma *Multinomial Naïve Bayes*. Pembelajaran ini akan menggunakan sumber dari buku, jurnal ilmiah, serta situs web yang berkaitan dengan teori yang digunakan.

### 3.1.2 Pengumpulan Data dan Pembuatan *Dataset*

Tahap kedua dalam penelitian ini adalah pengumpulan data. Pada tahap ini, data-data akan dikumpulkan dengan memberikan sebuah angket atau kuesioner kepada sejumlah mahasiswa Universitas Multimedia Nusantara program studi Informatika. Data yang didapat akan dikumpulkan dalam bentuk *csv*. Metode kualitatif akan digunakan untuk memahami pendapat subjek penelitian seperti perilaku, persepsi, emosi, dan sebagainya.

### 3.1.3 Pelabelan Data

Tahap ketiga dalam penelitian ini adalah pelabelan data. Data yang telah dikumpulkan berjumlah 86 dan setiap datanya akan diberi sebuah label positif, label negatif, dan label netral. Pemberian label ini dilakukan secara manual. Kemudian seorang pakar Bahasa Indonesia, Ibu Niknik Mediyawati, akan melakukan validasi pada data yang sudah diberi label. Apabila terdapat label yang salah, label tersebut akan diberi perbaikan.

### 3.1.4 Pra Proses Data

Tahap keempat dalam penelitian ini ialah melakukan pra proses data. Data *train* akan dilakukan proses *text preprocessing* untuk mengubah data yang didapat menjadi sebuah data yang siap untuk diolah. Berikut merupakan penjabaran dari pra proses data.

1. *Case Folding*

Pada tahap ini, data yang digunakan dalam pemodelan akan diubah menjadi huruf kecil semua. Berikut merupakan contoh dari tahap *case folding*.

Tabel 3.1. Contoh Proses *Case Folding*

<b>Contoh Data</b>	<b>Hasil <i>Case Folding</i></b>
Terkadang aku TIDAK paham sama kamu.	terkadang aku tidak paham sama kamu.

Pada tabel di atas, dapat dilihat huruf "T" pada kata "Terkadang" diubah menjadi huruf kecil. Dapat dilihat juga pada kata "TIDAK" dengan huruf kapital semua, diubah menjadi huruf kecil semua.

## 2. Cleaning Text

Pada tahap ini, data hasil *case folding* akan dilakukan pembersihan teks. Pembersihan ini bertujuan untuk membersihkan teks menjadi data teks yang lebih bersih sehingga memudahkan tahap pengolahan. Berikut merupakan proses-proses yang ada pada tahap *cleaning text*.

### (a) Menghapus Tanda Baca

Pada bagian ini, tanda baca yang tidak digunakan untuk pengolahan seperti tanda titik (.), tanda koma (,), dan sebagainya akan dihapus. Berikut merupakan contoh penghapusan tanda baca.

Tabel 3.2. Contoh Proses Menghapus Tanda Baca

<b>Contoh Hasil <i>Case Folding</i></b>	<b>Hasil <i>Cleaning Text</i> Menghapus Tanda Baca</b>
Terkadang aku TIDAK paham sama kamu.	terkadang aku tidak paham sama kamu

### (b) Menghapus Angka

Pada bagian ini, angka tidak akan digunakan dalam pengolahan ini. Sehingga angka tersebut akan dihapus dari teks. Berikut merupakan contoh penghapusan angka.

Tabel 3.3. Contoh Proses Menghapus Angka

<b>Contoh Hasil Menghapus Tanda Baca</b>	<b>Hasil <i>Cleaning Text</i> Menghapus Angka</b>
sejak tahun 2020 kita semua harus menetap di rumah	sejak tahun kita semua harus menetap di rumah

(c) Menghapus Emoji.

Pada bagian ini, emoji akan dihapus dari teks. Berikut merupakan contoh penghapusan emoji.

Tabel 3.4. Contoh Proses Menghapus Angka

Contoh Hasil Menghapus Angka	Hasil <i>Cleaning Text</i> Menghapus Emoji
sejak tahun kita semua harus menetap di rumah 😊	sejak tahun kita semua harus menetap di rumah

(d) Menghapus Spasi Lebih

Pada bagian ini, teks pada dokumen yang memiliki spasi lebih akan dihapus. Penghapusan ini dilakukan karena *python* menganggap spasi sebagai karakter. Dalam beberapa kasus, ini juga dapat merusak logika kode. Berikut merupakan contoh penghapusan spasi lebih.

Tabel 3.4. Contoh Proses Menghapus Spasi Lebih

Contoh Hasil Menghapus Emoji	Hasil <i>Cleaning Text</i> Menghapus Spasi Lebih
sejak tahun kita semua harus menetap di rumah	sejak tahun kita semua harus menetap di rumah

3. Tokenizing

Pada tahap ini, data hasil *cleaning text* akan dipecah setiap kata menjadi bentuk *token*. *Token* akan disimpan ke dalam sebuah *list*. Berikut merupakan contoh *tokenizing*.

Tabel 3.5. Contoh Proses *Tokenizing*

Contoh Hasil <i>Cleaning Text</i>	Hasil <i>Tokenizing</i>			
sejak tahun kita semua harus menetap di rumah	sejak	tahun	kita	semua
	harus	menetap	di	rumah

4. Spell Check

Selanjutnya adalah memeriksa kata-kata pada setiap *token* apakah terdapat salah ketik atau tidak. Jika terdapat salah ketik, kata tersebut akan diperbaiki menjadi kata yang sebenarnya. Berikut merupakan contoh *spell check*.

Tabel 3.6. Contoh Proses *Spell Check*

Contoh Hasil <i>Tokenizing</i>				Hasil <i>Spell Check</i>			
sejak	thn	kita	semua	sejak	tahun	kita	semua
hrus	menetap	di	rumah	harus	menetap	di	rumah

#### 5. Stopword

Pada tahap ini, akan dilakukan penghapusan pada kata-kata yang sering muncul dan tidak memiliki arti seperti kata depan, dan sebagainya. Berikut merupakan contoh *stopword*.

Tabel 3.5. Contoh Proses *Stopword*

Contoh Hasil <i>Spell Check</i>	Hasil <i>Stopword</i>
perkuliahan	perkuliahan
berjalan	berjalan
dengan	
lancar	lancar
mengeluarkan	mengeluarkan
tenaga	tenaga
lebih	

#### 6. Stemming

Pada tahap ini, akan dilakukan reduksi pada kata-kata di dokumen menjadi bentuk kata dasar. Proses *stemming* ini akan menggunakan *library* Sastrawi untuk Bahasa Indonesia. Kata-kata yang dapat direduksi oleh Sastrawi bisa berupa imbuhan depan, imbuhan belakang, imbuhan sisipan, serta imbuhan depan belakang. Berikut merupakan contoh hasil *stemming*.

Tabel 3.6. Contoh Proses *Stemming*

Contoh Hasil <i>Stopword</i>	Hasil <i>Stemming</i>
perkuliahan	kuliah
berjalan	jalan
lancar	lancar
mengeluarkan	keluar
tenaga	tenaga

### 3.1.5 Pembuatan Model dan Implementasi Algoritma

Pada tahap ini, model akan dibuat dengan menggunakan algoritma *Multinomial Naïve Bayes*. Data hasil pra proses akan dibagi menjadi data *train* sebesar 80% dan data *test* sebesar 20%. Dalam pembuatan model, akan diimplementasikan algoritma *Multinomial Naïve Bayes* untuk analisis sentimen.

### 3.1.6 Pengujian dan Evaluasi

Tahap terakhir dari metodologi penelitian ini adalah pengujian dan evaluasi. Pada tahap ini, dilakukan pengecekan apakah model yang dibuat sudah berjalan dengan baik atau tidak. Hasil yang sudah diuji akan dihitung dengan menggunakan *confusion matrix*. Sehingga hasil akhir dapat disimpulkan manakah pengujian yang memberikan hasil klasifikasi terbaik.

## 3.2 Perancangan Sistem

Pada proses perancangan sistem, akan dilakukan pengumpulan serta pembuatan dataset ke dalam aplikasi Microsoft Excel untuk dijadikan file *csv*. Sebelum model dibuat, akan dilakukan sebuah perancangan dengan menggunakan *flowchart*. *Flowchart* tersebut terdiri dari *flowchart* utama, *flowchart text preprocessing*, *flowchart TF-IDF*, dan *flowchart* klasifikasi *Multinomial Naïve Bayes*.

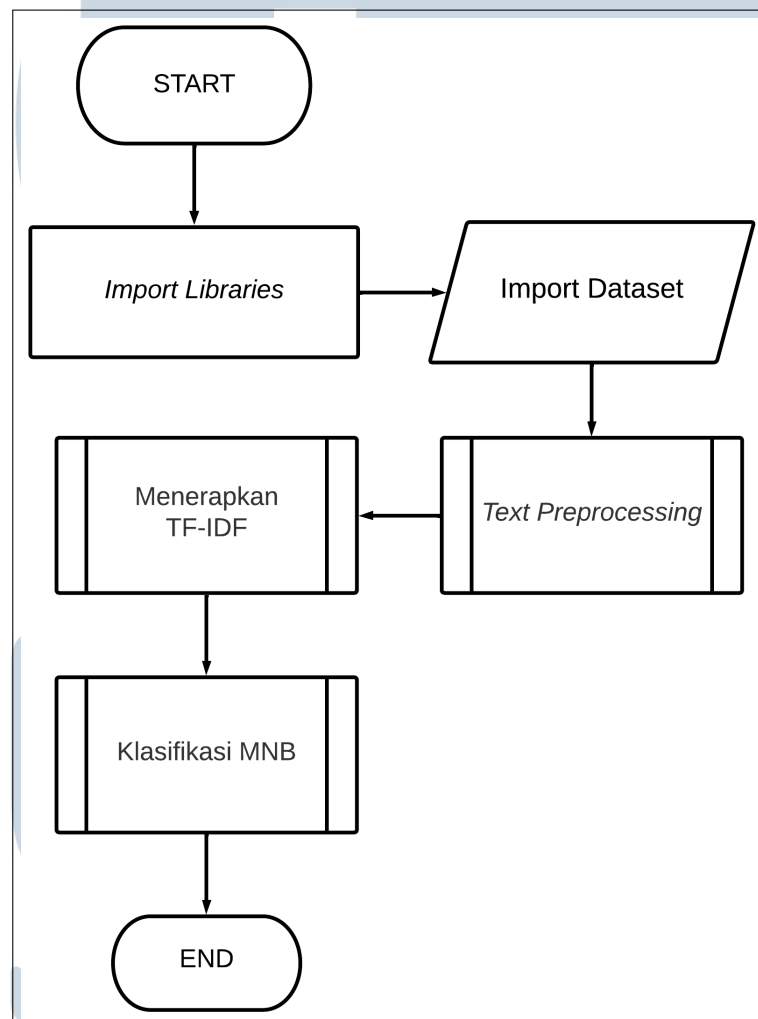
### 3.2.1 Flowchart Utama

Sistem yang dibuat dalam penelitian ini adalah mengimplementasikan algoritma *Multinomial Naïve Bayes* untuk melakukan analisa sentimen pendapat mahasiswa mengenai pengalaman kuliah *hybrid*. Proses pertama adalah melakukan *import library* yang akan digunakan dalam pembuatan model. Selanjutnya adalah mengambil *dataset* yang sebelumnya sudah dikumpulkan. *Dataset* didapatkan dengan membuat sebuah *google form* berupa pertanyaan terkait pengalaman kuliah *hybrid* dan kemudian disebarikan kepada mahasiswa program studi Informatika.

Setelah data terkumpul, akan diberikan sebuah label untuk setiap responden. Pemberian label ini akan dibantu oleh seorang pakar ahli Bahasa Indonesia untuk menentukan manakah kalimat positif, negatif, dan netral. Proses selanjutnya adalah melakukan *text preprocessing*. Proses ini bertujuan untuk melakukan olah data text

supaya menjadi data yang terstruktur. *Text preprocessing* terdiri dari tahap, *case folding*, *cleaning text*, *tokenization*, *spell check*, *stopword*, dan *stemming*.

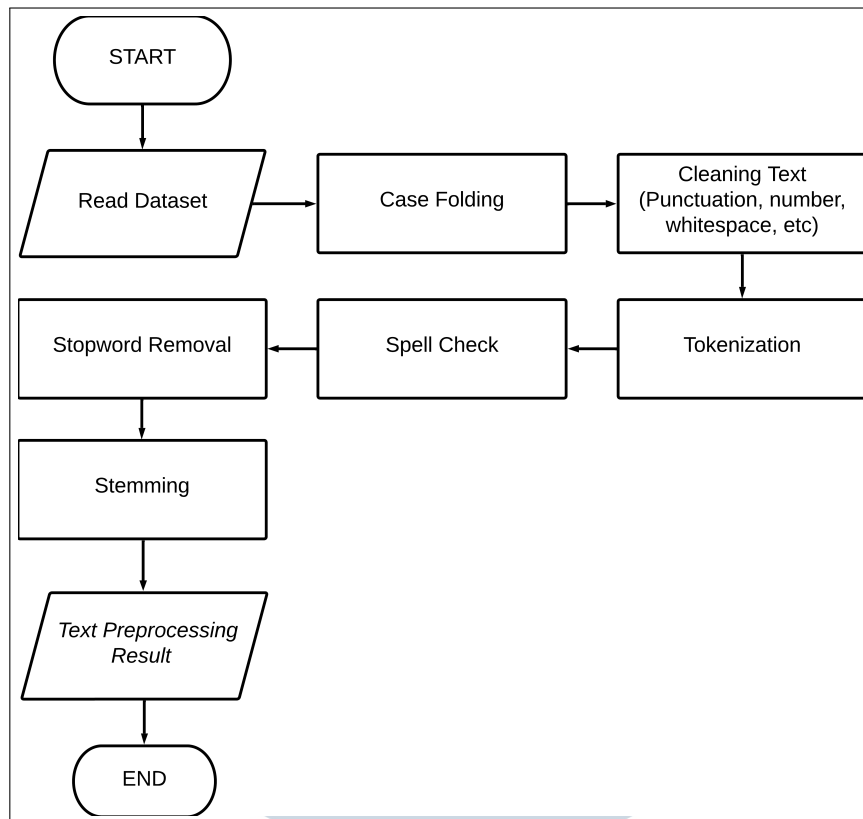
Proses selanjutnya adalah melakukan pembobotan pada setiap kata dengan menggunakan TF-IDF. Proses selanjutnya adalah melakukan klasifikasi *Multinomial Naïve Bayes*. Klasifikasi ini digunakan untuk mencari nilai *probability* dari tiap klasifikasinya. Gambar 3.2 merupakan *flowchart* utama pada penelitian ini.



Gambar 3.2. *Flowchart* Utama

### 3.2.2 **Flowchart Text Preprocessing**

Berikut merupakan *flowchart* yang menjelaskan tahapan dalam *text preprocessing*.



Gambar 3.3. Flowchart Text Preprocessing

Tahap *text preprocessing* ini bertujuan untuk mengolah teks data menjadi lebih terstruktur. Berikut merupakan langkah-langkah melakukan *text preprocessing*.

1. *Cleaning Text*

Pada tahap ini, data teks akan dilakukan pembersihan dengan menghapus tanda baca, angka, spasi lebih, dan emoji yang tidak diperlukan. Pada tahap ini juga diterapkan pemeriksaan kata atau *spell check* yang bertujuan untuk memeriksa kata yang salah ketik.

2. *Case Folding*

Pada tahap ini, huruf pada kalimat akan diubah menjadi huruf kecil semua.

3. *Tokenization*

Tahap ini merupakan proses untuk mengubah kalimat menjadi kata-kata yang disimpan dalam bentuk *array*.

4. *Spelli Check*

Tahap ini merupakan tahap pengecekan kata-kata apabila terdapat kesalahan



ketik. Kata-kata yang salah ketik akan diperbaiki.

#### 5. *Stopword Removal*

Pada tahap ini, akan dilakukan proses menghapus kata-kata yang tidak memiliki makna.

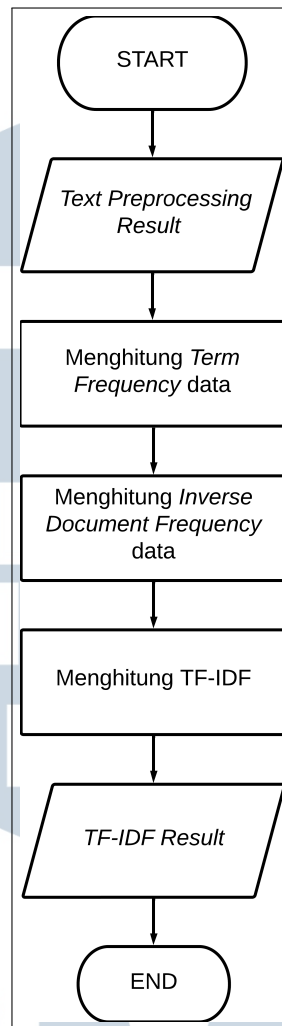
#### 6. *Stemming*

Tahap ini merupakan tahap untuk mereduksi bentuk kata menjadi kata dasar.

### 3.2.3 Flowchart TF-IDF

TF-IDF bertujuan untuk memberikan nilai bobot pada *term* yang dihitung berdasarkan kemunculan *term* dari data hasil *text preprocessing*. Langkah awal yang akan dilakukan dalam pembobotan dimulai dengan menghitung *term frequency* (TF) untuk menentukan banyaknya jumlah kemunculan kata dan kemudian dilakukan perhitungan *inverse document frequency* (IDF). Setelah mengetahui nilai TF dan IDF, langkah selanjutnya adalah menentukan nilai TF-IDF itu sendiri dengan menggabungkan nilai TF dengan nilai IDF. Gambar 3.4 merupakan *flowchart* dalam melakukan pembobotan kata menggunakan metode TF-IDF.



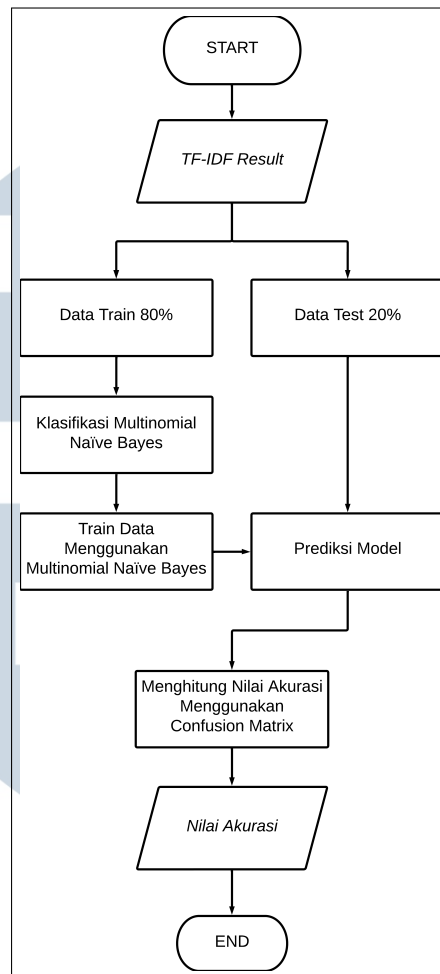


Gambar 3.4. *Flowchart* Perhitungan TF-IDF

### 3.2.4 Menerapkan Algoritma *Multinomial Naïve Bayes*

Berikut merupakan *flowchart* dari penerapan algoritma *Multinomial Naïve Bayes* pada nilai TF-IDF yang sudah didapatkan.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.5. Flowchart Klasifikasi *Multinomial Naïve Bayes*

Setelah mendapatkan nilai TF-IDF, tahap selanjutnya adalah melakukan *train* data menggunakan algoritma *Multinomial Naïve Bayes*. Data akan dipisah menjadi data *train* dan data *test* dengan rasio perbandingan 80:20. Data *train* akan diisi dengan nilai TF-IDF yang diperoleh pada proses sebelumnya dan akan dilatih dengan data *test* yang diisi dengan label data. Algoritma *Multinomial Naïve Bayes* akan melakukan prediksi pada data *test*. Evaluasi akan dilakukan dengan menghitung nilai akurasi menggunakan *confusion matrix* untuk mendapatkan nilai *accuracy*, *precision*, *recall*, *f1-score*.