

## **BAB III**

### **METODE PENELITIAN**

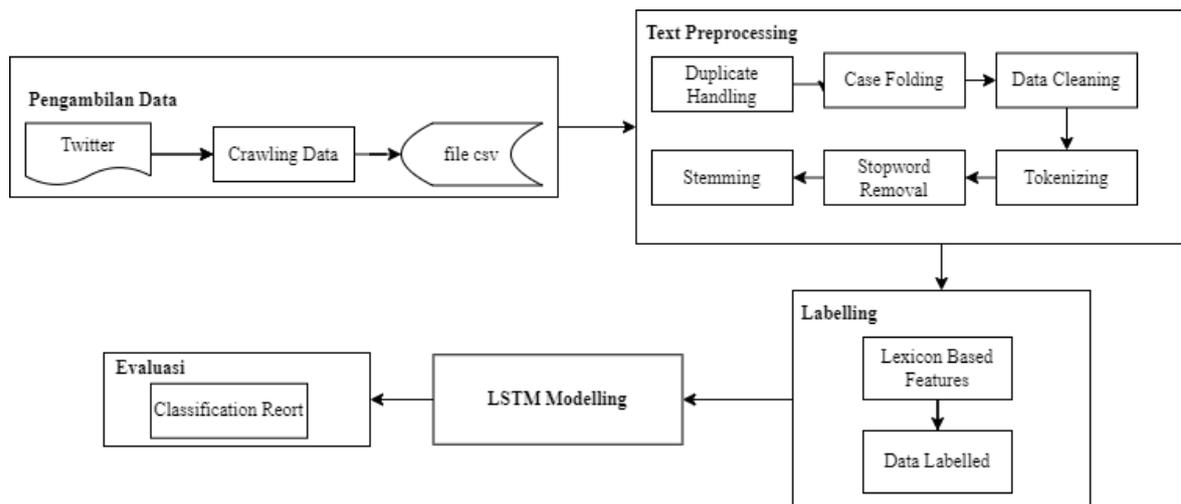
#### **3.1 Jenis Penelitian**

Dalam penelitian ini, digunakan metode penelitian kualitatif deskriptif menurut Sugiyono (2019) metode ini berfokus pada pemahaman objek dalam keadaan alami, dengan peneliti sebagai instrumen utama. Walidin & Tabrani (2015) menyebutkan penelitian kualitatif bertujuan untuk menggambarkan fenomena manusia atau sosial secara komprehensif melalui deskripsi yang terperinci. Penelitian kualitatif cenderung bersifat deskriptif dan data diperoleh melalui pendekatan induktif, dengan penekanan pada perspektif peserta dalam penelitian (Fadil, 2020). Metode penelitian deskriptif, menurut Nazir (2014) digunakan untuk menganalisis status kelompok manusia, objek, kondisi, sistem pemikiran, atau peristiwa pada saat ini. Berdasarkan paparan para ahli diatas penelitian kualitatif deskriptif merupakan kombinasi pendekatan kualitatif dan deskriptif yang bertujuan untuk memperoleh data tanpa mempertimbangkan kondisi tertentu, dengan fokus pada makna.

Pendekatan penelitian kualitatif deskriptif ini dipilih untuk mendapatkan pemahaman dan gambaran tentang realitas sosial terkait pelaksanaan pemilu 2024. Dalam metode deskriptif kualitatif ini, data dianalisis dalam bentuk teks dan bukan berupa angka statistik. Penelitian ini akan memberikan paparan atau penggambaran naratif tentang situasi atau kondisi yang diteliti. Dengan data yang digunakan dalam penelitian ini akan berasal dari kumpulan *tweet* atau pesan para pengguna twitter selama bulan Januari sampai dengan mei 2022 terkait pemilu 2024, setelah data diperoleh akan mencoba untuk melihat kejadian yang menjadi pusat perhatiannya, dan kemudian diilustrasikan sebagaimana apa adanya.

#### **3.2 Desain Penelitian**

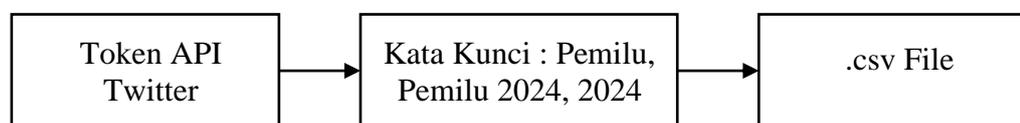
Pada penelitian ini digunakan beberapa tahap-tahap yang meliputi pengumpulan data, *text mining*, pelabelan sentimen, Hasil sentimen, *modeling*, dan evaluasi performa sebagaimana ditunjukkan pada Gambar 3.1.



Gambar 3. 1 Tahap Penelitian

### 3.2.1 Pengumpulan Data

*Crawling* adalah proses pengumpulan data dari situs web dengan menggunakan URL sebagai referensi. URL ini digunakan untuk menemukan semua *hyperlink* yang ada di situs web tersebut. Setelah URL ditemukan, dilakukan pengindeksan untuk mencari kata-kata dalam dokumen pada setiap tautan yang ada. Dalam penelitian ini, data yang diambil adalah tweet yang terdapat pada media sosial Twitter. Untuk mengakses tweet, diperlukan hak akses yang terdiri dari kunci konsumen, rahasia konsumen, token akses, dan rahasia token akses. Kata kunci yang digunakan dalam proses *crawling* adalah "pemilu", "pemilu 2024", dan "2024". Berikut adalah alur proses dari *crawling* data.



Gambar 3. 2 Proses *Crawling*

Dalam proses melakukan *crawling data* langkah pertama yang dilakukan adalah mendapatkan dan memasukan token API kedalam program kemudian masukan kata kunci atau topik yang akan dilakukan penelitian setelah itu program akan melakukan *crawling* pada API twitter kemudian hasilnya akan disimpan kedalam bentuk file .csv

### 3.2.2 Text Preprocessing

Tahap *preprocessing* dilakukan untuk mengolah data mentah menjadi

kumpulan data yang siap digunakan. beberapa tahapan preprocessing yang dilakukan pada penelitian ini, yaitu:

1. *Duplicate Handling*: Tahap *duplicate handling* dalam *preprocessing* teks adalah proses mengidentifikasi dan menangani data duplikat dalam dataset. Hal ini penting untuk menghilangkan redundansi dan mengoptimalkan analisis teks. Kalimat-kalimat yang memiliki nilai sama dan berjumlah lebih satu akan dihapus dan disisakan satu saja.

2. *Case folding*: tahap dalam *preprocessing* teks yang melibatkan perubahan semua huruf dalam teks menjadi huruf kecil atau huruf besar. Tujuan dari *case folding* adalah untuk menghapus perbedaan kasus dalam teks sehingga memudahkan pengolahan lebih lanjut.

Sebagai contohnya adalah kamilmat berikut

Joko Widodo Adalah Presiden ke 7 → joko widodo adalah presiden ke 7

Dengan diubahnya huruf menjadi huruf kecil maka dimensi kata yang dibaca memiliki nilai dimensi yang sama sehingga tidak akan mengalami kekliruan dalam pemrosesan kata nantinya.

3. *Cleaning* : proses menghilangkan komponen-komponen tertentu yang terdapat dalam *tweet* yaitu *Uniform Resource Locator (URL)*, *username*, *RT (Retweet)*, karakter *HTML*, dan *hashtag*.

Sebagai contoh dari proses ini adalah sebagai berikut :

“#2024 pemilu serentak akan diselenggarakan @KPU” setelah dilakukan tahap *cleaning* kata tersebut akan berubah menjadi “2024 pemilu diselenggarakan pemilu”

4. *Tokenizing*: tahap penting dalam *preprocessing* teks yang melibatkan pemisahan teks menjadi unit-unit yang lebih kecil yang disebut token. Tahapan pada proses ini akan menjadi kalimat dalam data akan dipecah menjadi per suku kata yaitu sebagai berikut.

Penyelenggaraan pemilu akan diadakan secara serentak di 2024 → penyelenggaraan, pemilu, akan, diadakan, secara, serentak, di, 2024. Proses pemecahan menjadi bagian unit kecil ini akan menghasilkan pemahaman kata yang dilakukan komputer menjadi jauh lebih baik.

5. *Stopword Removal* : pada tahap ini akan dilakukan penghapusan kata

berdasarkan kata-kata yang terdapat pada stoplist. Dalam konteks ini kata kata penghubung akan dihapus seperti bekerja akan diproses dan diubah menjadi kerja.

6. *Stemming*: Proses memecah kata menjadi tense dasar yaitu kata kata yang memiliki makna yang sama namun diucapkan dengan berbeda akan disetarakan dalam proses ini dan diubah menjadi kata dasarnya misalnya saja memakai dan dipakai memiliki kata dasar yang sama maka komputasi akan memproses kedua kata tersebut untuk diubah menjadi kata dasarnya yaitu pakai.

### 3.2.3 Labelling

Pada penelitian kali ini dilakukan proses pelabelan secara otomatis dengan memanfaatkan kamus sentimen atau *lexicon based features* dimana kamus sentimen yang digunakan pada penelitian ini diambil dari repositori github yaitu melalui <https://github.com/fajri91/InSet> dimana terdapat dua *lexicon inset* yaitu negatif dengan nilai -1 sampai 5 dengan jumlah kata yaitu 6.609 dan positif dengan nilai 1 sampai 5 dengan jumlah kata mencapai 3.609. Setelah dataset dimasukan kedalam fitur leksikon maka hasil dari *labeling* tersebut disimpan dalam bentuk csv yang kemudian akan dilakukan proses selanjutnya yaitu tahap modeling dan melihat jumlah kecenderungan sentimen dalam dataset. Berikut beberapa contoh kata yang terdapat dalam kamus sentimen yang dipakai

mendahului	2
segera	3
lekas	2
akas	2
terampil	4
jeraus	2
pesat	3
sebat	1
kawal	1
berkawal	4
gugah	2

Gambar 3. 3 Kamus sentimen positif

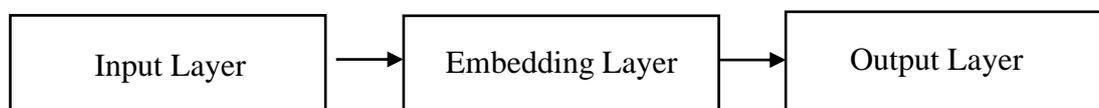
anjir	-3
putus jiwa	-5
berlalu dr dunia	-5
putus nyawa	-5
menarik napas pengha	-5
membuang nyawa	-5
hilang hayat	-4
putus napas	-4
kojor	-3
berputih tulang	-2
buntu	-3

Gambar 3. 4 Kamus Sentimen negatif

Kata-kata yang berada dalam kalimat pada dataset nantinya akan dihitung berdasarkan kamus data di atas seperti misalnya kalimat seorang pria nelangsa lakukan gantung diri berdasarkan perhitungan kamus sentimen yang dipakai maka kalimat tersebut akan memiliki sentimen negatif dengan nilai polaritas sebesar -5 ( gantung diri -2 + nelangsa -3). Begitu juga ketika pada kalimat yang memiliki kata negatif dan positif maka akan diambil selisihnya positif atau negatif.

### 3.2.4 Modelling

Dalam penelitian kali ini akan dilakukan *modelling* dengan menggunakan algoritma LSTM dimana dalam proses modeling yang dilakukan akan digunakan dengan tanpa proses *hyperparameter tuning* sebagaimana yang disebutkan pada latar belakang kemudian . Data yang dimodelkan dibagi ke dalam data test dan data train dengan presentasi 20% sebagai data test kemudian 80% sebagai data testing. Setelah data dipisahkan proses modeling dengan LSTM dilakukan dengan *epochs=50, batch\_size=64, validation\_split=0.1, verbose=2*, dan menggunakan optimasi menggunakan RMSProp. Berikut merupakan skema dari *modelling* yang digunakan secara sederhana.



Gambar 3. 5 Skema Modelling LSTM

Pada LSTM, input awal adalah urutan atau rangkaian data yang diberikan sebagai input ke jaringan LSTM. Embedding melibatkan representasi vektor yang diberikan kepada setiap elemen dalam urutan *input*. Ini dilakukan untuk mengubah representasi diskret seperti kata menjadi representasi kontinu dalam bentuk vektor numerik. *Embedding* membantu LSTM dalam mempelajari hubungan semantik antara elemen-elemen dalam urutan. Dan *Output* LSTM adalah prediksi atau representasi yang dihasilkan oleh jaringan LSTM setelah memproses urutan *input*. Dalam kasus klasifikasi teks, *output* dapat berupa probabilitas kelas atau label yang ditugaskan kepada urutan *input*.

### 3.2.5 Evaluasi

Dalam penelitian kali digunakan evaluasi performa dengan memanfaatkan fungsi dari klasifikasi dimana dalam melakukan evaluasi pada penelitian ini akan dilihat dari segi akurasi yang dihasilkan dari sebuah model apakah sudah berjalan cukup baik atau tidak kemudian selain itu akan dilihat dari segi *confusion matrix*, dan *training validation loss*. Pada tahap evaluasi adalah metode untuk mendapatkan kendala yang dialami dari keluaran yang sudah dihasilkan, kemudian ketika model yang dihasilkan memiliki performa kurang baik bisa dilakukan optimasi seperti *hyperparameter tuning* tapi karena pada penelitian ini tidak akan melakukan *hyperparameter tuning* maka sebagai gantinya akan dilakukan percobaan dengan mengganti rasio data *test*, data latih, dan komponen arsitektur LSTM, yang lebih lanjut dan melihat apakah model tersebut mengalami *overfitting* dan *underfitting*.

Dalam penelitian ini akan digunakan fungsi *classification report* dalam melakukan evaluasi performa dimana nantinya akan dihasilkan nilai dari akurasi, presisi, *recall*, *F1-Score*. Untuk menghitung akurasi, presisi, *recall*, dan *F1-Score*, dibutuhkan informasi tentang jumlah prediksi yang benar (*True Positive*, TP), jumlah prediksi negatif yang benar (*True Negative*, TN), jumlah prediksi positif yang salah (*False Positive*, FP), dan jumlah prediksi negatif yang salah (*False Negative*, FN). Dengan menggunakan informasi ini, berikut adalah rumus untuk menghitung metrik evaluasi yang disebutkan:

1. Akurasi (*Accuracy*):  
$$\text{Akurasi} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})$$
2. Presisi (*Precision*):

$$\text{Presisi} = \text{TP} / (\text{TP} + \text{FP})$$

3. *Recall (Sensitivity atau True Positive Rate):*

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

4. *F1-Score:*

$$\text{F1-Score} = 2 * (\text{Presisi} * \text{Recall}) / (\text{Presisi} + \text{Recall})$$

### 3.3 Lingkungan Komputasi

Penelitian yang dilakukan memiliki beberapa alat pendukung baik dari segi perangkat keras maupun perangkat lunak berikut merupakan lingkungan komputasi yang digunakan pada penelitian ini.

Perangkat keras yang digunakan:

1. Lenovo Ideapad 330 14IKB
2. Ram 8GB
3. SSD 256 GB + HDD 1 TB
4. Intel i5 Generasi ke 8
5. VGA Radeon 530
6. Mouse + Keyboard Eksternal

Perangkat Lunak yang digunakan :

1. Google Collaboratory
2. Twitter Developer API
3. Bahasa Pemrograman Python
4. Windows 11