# **BAB 3**

# **ANALISIS DAN PERANCANGAN**

## **Analisis Kebutuhan Fungsional dan Non Fungsional**

### **Analisis Kebutuhan Fungsional**

Kebutuhan Fungsional merupakan proses yang ada didalam sistem serta apa saja yang dibutuhkan oleh sistem agar dapat berjalan baik. Berikut kebutuhan fungsional dari sistem yang akan dirancang:

1. Sistem akan terdiri dari 3 segment yaitu Get Data, Train, Test dengan data Training 90 persen dan Testing 10 persen
2. Halaman Get Data berfungsi untuk mengambil data bersumber dari twitter
3. Data input sistem akan berupa file excel / csv
4. Sistem akan dikembangkan dalam bentuk website
5. Output program akan berupa file hasil prediksi *sentiment analysis*

### **Analisis Kebutuhan Non Fungsional**

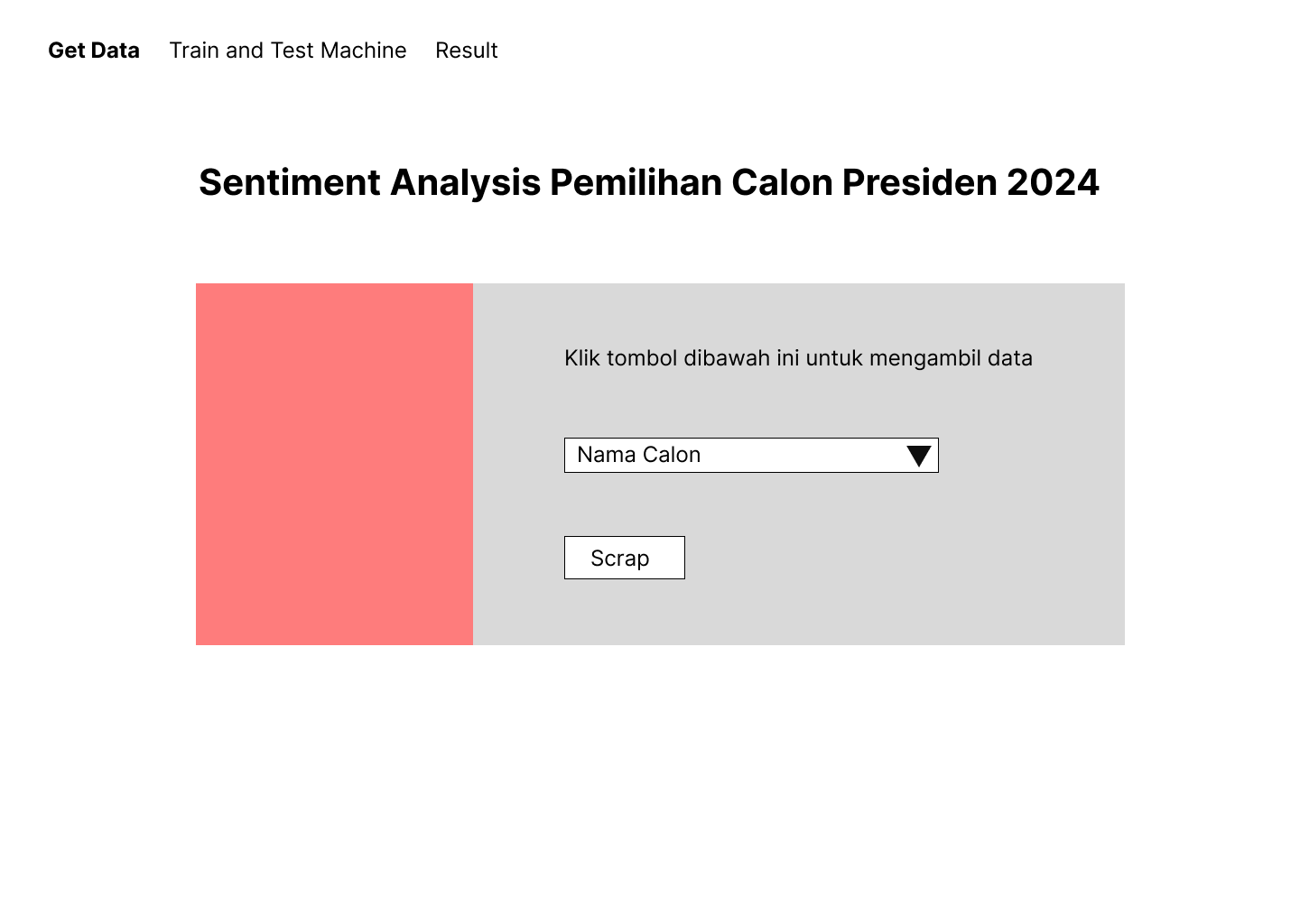
Kebutuhan Non Fungsional merupakan kebutuhan yang harus dipenuhi dalam menunjang berjalannya sistem. Berikut kebutuhan perangkat lunak dan keras dari sistem yang akan dirancang:

1. Ryzen 5 3400G
2. Rx Vega 11 Graphics
3. 16 GB RAM
4. 256 GB SSD
5. 1 TB HDD
6. Windows 10

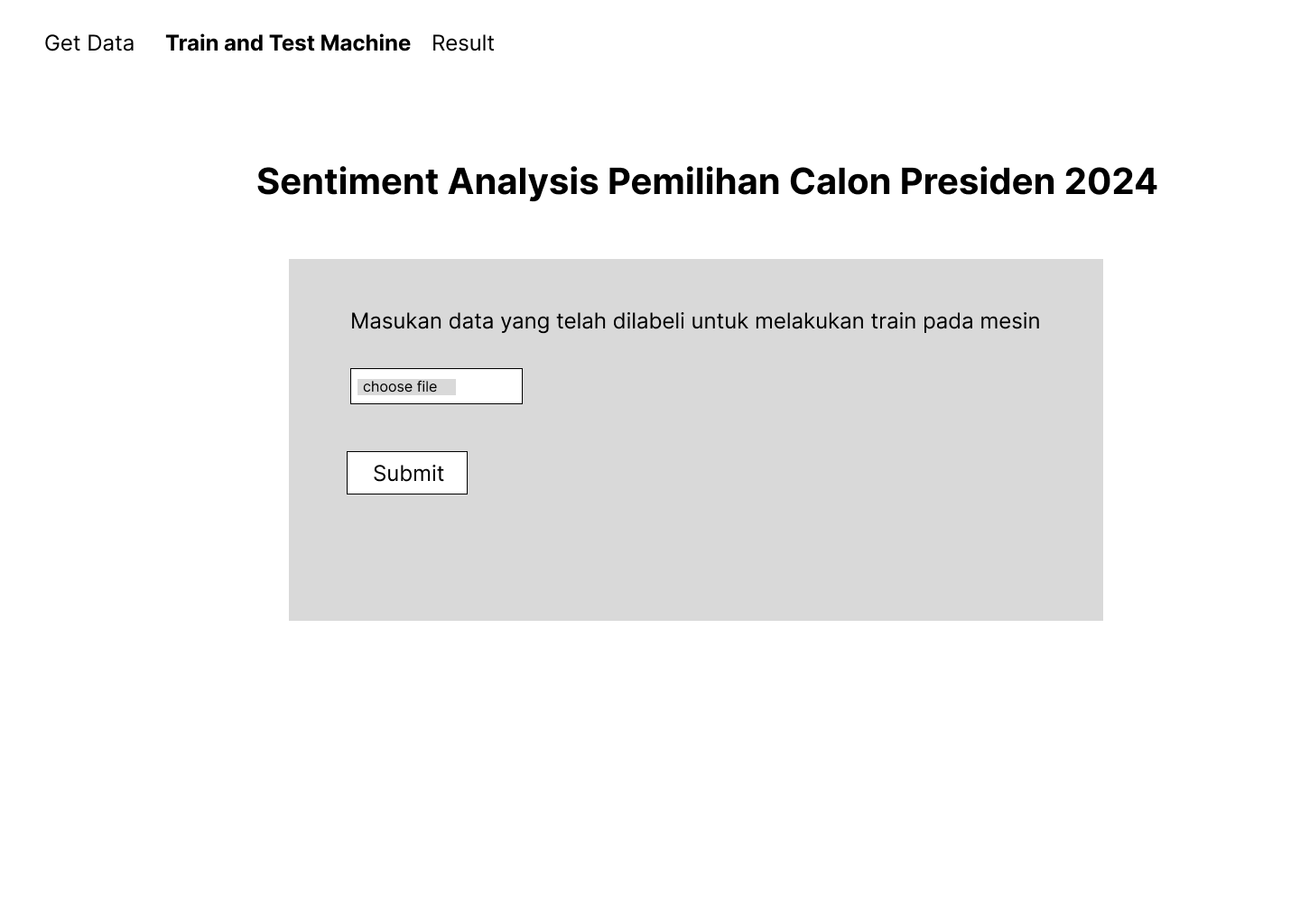
## **Pemilihan Algoritma Pemecahan**

Algoritma yang dipilih adalah Support Vector Machine dikarenakan Support Vector Machine merupakan algoritma text mining yang memiliki akurasi tinggi [14] terlihat pada jurnal ini, algoritma *Support Vector Machine* memiliki akurasi, *precision*, dan *recall* lebih tinggi dibandingkan dengan algoritma *Naïve Bayes*, lalu untuk jurnal [15] diketahui bahwa tingkat akurasi, *precision*, dan *recall* algoritma *Support Vector Machine* memiliki akurasi lebih tinggi daripada algoritma *K-Nearest Neighbor* (K-NN) dan *Naïve Bayes*, sehingga penulis memilih algoritma Support Vector Machine sebagai algoritma untuk memprediksi hasil sentiment pemilihan capres 2024 melalui sosial media twitter. Lalu untuk validasi akan dilakukan dengan *10-Fold Cross Validation*, penulis melakukan 10 iterasi dikarenakan tingkat akurasi pada k = 10 memiliki akurasi tertinggi [16] .

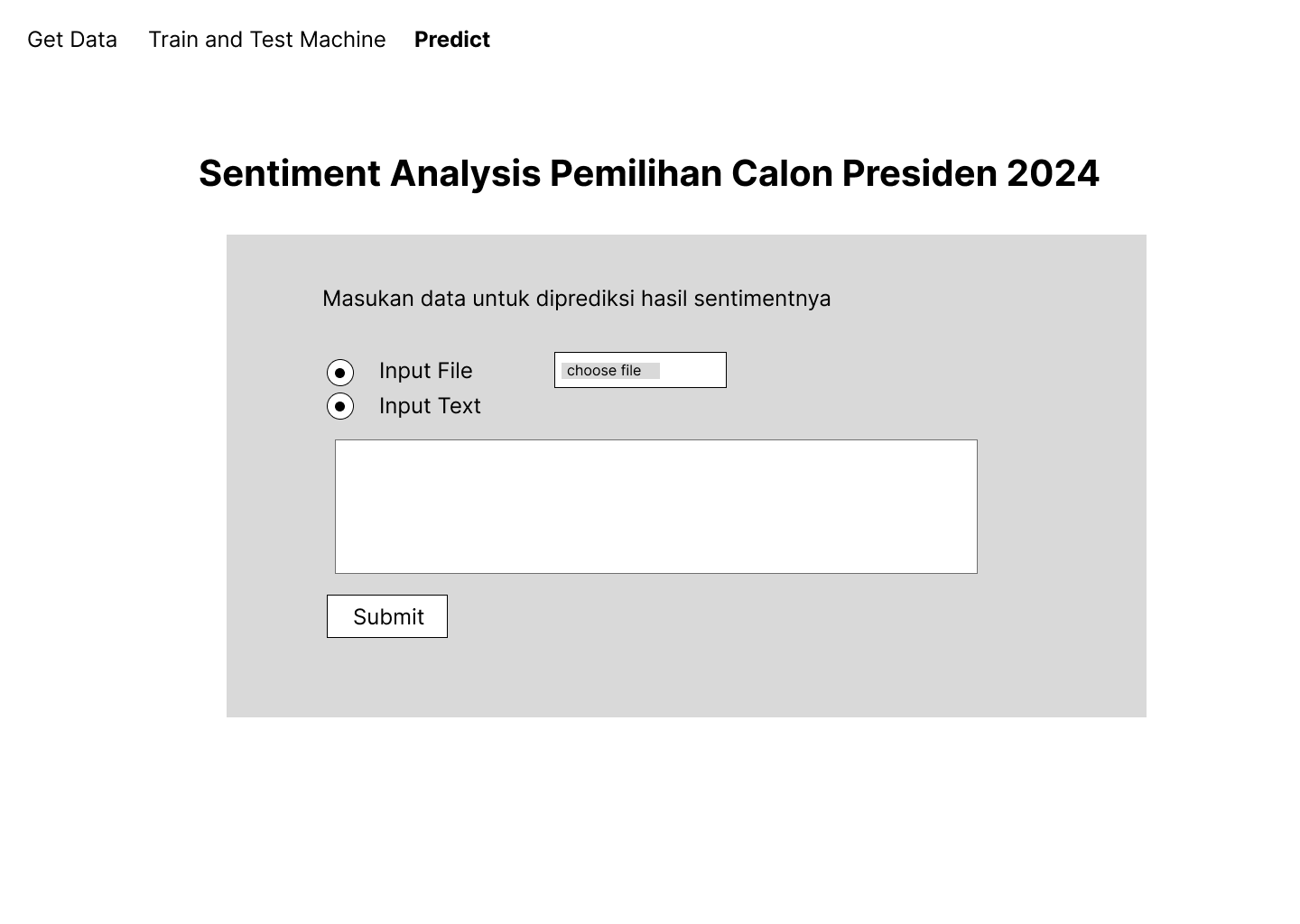
## **Perancangan User Interface**

Berikut Gambar 3.1 sampai 3.3 merupakan rancangan User Interface yang berbentuk wireframe, rancangan ini merupakan gambaran bentuk website akan dibangun

Gambar 3. 1 Rancangan User Interface Halaman Get Data



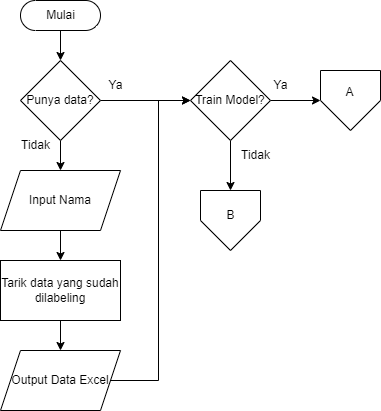
Gambar 3. 2 Rancangan User Interface Halaman Train



Gambar 3. 3 Rancangan User Interface Halaman Predict

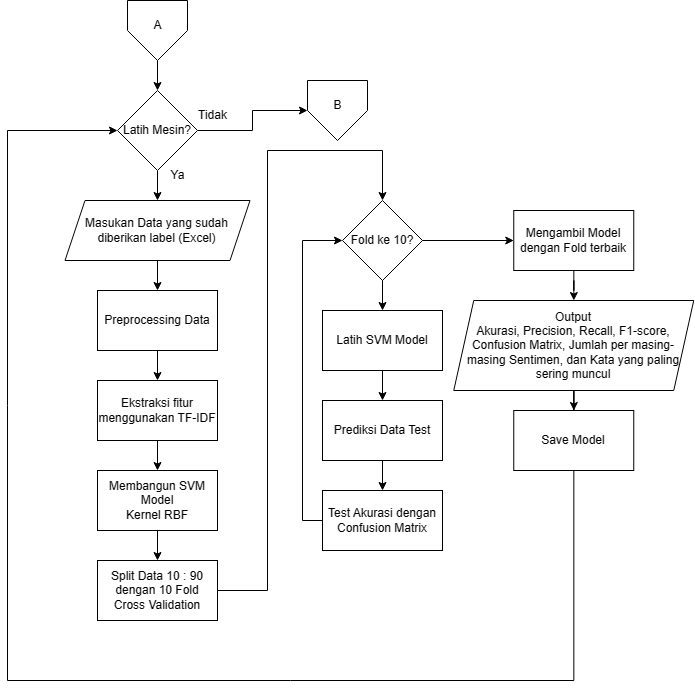
## **Perancangan Proses**

Gambar 3.4 sampai 3.6 memuat gambaran alur proses sistem yang dibagi menjadi 3 bagian inti yaitu “Get Data”, “Train Model” dan “Predict”.

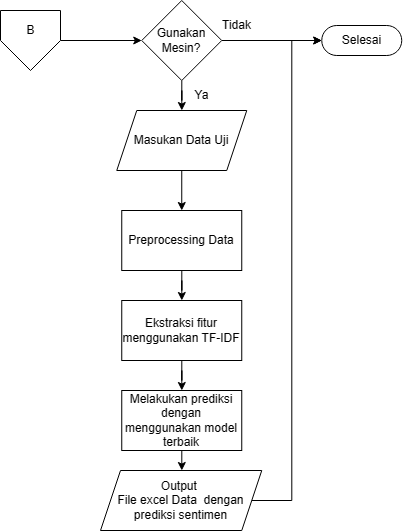
Pada gambar 3.4 dijelaskan pada proses penarikan data yang bersumber dari *Twitter* sehingga user akan diminta memasukan nama calon, yang selanjutnya data untuk calon tersebut yang telah dilabeli akan terdownload untuk user dan dapat dilanjutkan ke proses train model.

Gambar 3. 4 Flowchart Get Data

Selanjutnya pada gambar 3.5 dijelaskan proses train model SVM yang akan dilakukan penyimpanan model dengan nilai *f1-score* terbesar. Proses dimulai dengan memasukan data yang telah diberikan label dari tahap sebelumnya. Lalu dilakukan pre-processing pada setiap baris data. Lalu fitur data hasil pre-processing akan diekstraksi menggunakan TF-IDF. Hasil fitur ekstraksi lalu akan dimasukan kedalam model SVM dengan kernel Gaussian RBF yang dimasukan dengan metode 10 Fold Cross Validation sehingga data dibagi menjadi 10:90 dengan 90 data training dan 10 data testing, lalu dari train tersebut diambil nilai *f1-score* untuk menentukan fold terbaik dengan confusion matrix. Selanjutnya model terbaik akan disave untuk digunakan pada halaman selanjutnya.

Selanjutnya pada gambar 3.6 ditunjukan cara mesin melakukan prediksi dengan model yang sudah di save saat melakukan train mesin. Awalnya, data yang akan diprediksi dimasukan dalam bentuk teks atau file, lalu data tersebut dilakukan pre-processing, dan fiturnya diekstraksi menggunakan TF-IDF. Hasil fitur ekstraksi tersbut lalu diprediksi menggunakan model yang telah disave pada tahap sebelumnya. Output dari predict ini adalah hasil teks pre-procesing dan sentiment untuk inputan teks, sedangkan file excel berisikan *tweet* dan label hasil prediksi mesin.

Gambar 3. 5 Flowchart Train



Gambar 3. 6 Flowchart Predict

### ***Scrap* Data**

Tahapan awal dari penelitian ini adalah melakukan *scraping data* bersumber dari *Twitter* dengan kata kunci “(Nama Calon) Presiden” untuk mempersempit konteks dari *Tweet* pengguna *Twitter*. Proses pengumpulan data dilakukan dengan bantuan library *snscrape* oleh python. Data dikumpulkan mulai dari tanggal 1 Januari 2023 sampai 31 Maret 2023.

Proses *scraping* ini menghasilkan total 18.267 data dengan 7.810 data yang duplikat dan tidak relevan dengan tema “Pemilihan Calon Presiden 2024” sehingga menghasilkan total 10.457 data bersih dari 3 calon presiden.

### **Labeling**

Pada tahap ini, dilakukan proses pemberian label sentiment secara manual oleh penulis terhadap data bersih *tweet*. Pemberian label berdasarkan asumsi penulis terhadap *tweet* sehingga proses ini dapat berbeda pada setiap pengguna.

Pada penelitian ini, penulis menggunakan 3 jenis sentimen yaitu positif dilambangkan dengan angka 1, netral dilambangkan dengan angka 0, dan negatif yang dilambangkan dengan angka -1.

### ***Pre-processing***

Tahapan *pre-processing* berfungsi untuk membersihkan dan mentransformasi data sehingga dapat diproses oleh sistem dengan mudah dan akurat. Beberapa tahapan *pre-processing* pada penelitian ini diantaranya:

#### **Lowercasing**

*Lowercasing* merupakan tahapan untuk merubah semua huruf menjadi huruf kecil sehingga kata yang diproses menjadi seragam. Tabel 3.1 menampilkan contoh proses *lowercasing* dari beberapa *tweet* Anies Baswedan:

Tabel 3. 1 Tabel Hasil Lowercasing

|  |  |  |
| --- | --- | --- |
| No | Kalimat Awal | Hasil Proses |
| 1 | PKS Mendukung Anies Baswedan Menjadi Calon Presiden 2024 | pks mendukung anies baswedan menjadi calon presiden 2024 |
| 2 | Anies Baswedan Presiden Indonesia | anies baswedan presiden indonesia |
| 3 | Walau sekarang didukung Demokrat dan PKS, ada empat alasan Anies Baswedan gak bakal bisa menang jadi presiden !! | walau sekarang didukung demokrat dan pks, ada empat alasan anies baswedan gak bakal bisa menang jadi presiden !! |
| 4 | Anies Baswedan lh yg pantas jadi presiden | anies baswedan lh yg pantas jadi presiden |
| 5 | Pak Anies Baswedan calon presiden 2024 | pak anies baswedan calon presiden 2024 |

#### **Remove Special Character and Number**

Tahapan ini berfungsi menghasilkan kalimat tanpa tanda baca, *emoticon*, dan angka. Tabel 3.2 menampilkan contoh proses *Remove Special Character and Number*:

|  |  |  |
| --- | --- | --- |
| No | Kalimat Awal | Hasil Proses |
| 1 | pks mendukung anies baswedan menjadi calon presiden 2024 | pks mendukung anies baswedan menjadi calon presiden |
| 2 | anies baswedan presiden indonesia | anies baswedan presiden indonesia |
| 3 | walau sekarang didukung demokrat dan pks, ada empat alasan anies baswedan gak bakal bisa menang jadi presiden !! | walau sekarang didukung demokrat dan pks ada empat alasan anies baswedan gak bakal bisa menang jadi presiden |
| 4 | anies baswedan lh yg pantas jadi presiden | anies baswedan lh yg pantas jadi presiden |
| 5 | pak anies baswedan calon presiden 2024 | pak anies baswedan calon presiden |

Tabel 3. 2 Tabel Hasil Remove Special Character and Number

#### **Stemming**

Tahap ini, semua kata berimbuhan akan diproses menjadi kata dasar, seperti ditampilkan pada tabel 3.3 berikut:

Tabel 3. 3 Tabel Hasil Stemming

|  |  |  |
| --- | --- | --- |
| No | Kalimat Awal | Hasil Proses |
| 1 | pks mendukung anies baswedan menjadi calon presiden | pks dukung anies baswedan jadi calon presiden |
| 2 | anies baswedan presiden indonesia | anies baswedan presiden indonesia |
| 3 | walau sekarang didukung demokrat dan pks, ada empat alasan anies baswedan gak bakal bisa menang jadi presiden | walau sekarang dukung demokrat dan pks ada empat alasan anies baswedan gak bakal bisa menang jadi presiden |
| 4 | anies baswedan lh yg pantas jadi presiden | anies baswedan lh yg pantas jadi presiden |
| 5 | pak anies baswedan calon presiden | pak anies baswedan calon presiden |

#### **Tokenize**

Tahap ini berfungsi untuk memecah kalimat menjadi token per kata. Proses ini dilakukan dengan tujuan agar proses selanjutnya dapat diterapkan dalam level kata. Proses ditunjukan pada tabel 3.4 berikut:

|  |  |  |
| --- | --- | --- |
| No | Kalimat Awal | Hasil Proses |
| 1 | pks dukung anies baswedan jadi calon presiden | [[pks] [dukung] [anies] [baswedan] [jadi] [calon] [presiden]] |
| 2 | anies baswedan presiden indonesia | [[anies] [baswedan] [presiden] [Indonesia]] |
| 3 | walau sekarang dukung demokrat dan pks, ada empat alasan anies baswedan gak bakal bisa menang jadi presiden | [[walau] [sekarang] [dukung] [demokrat] [dan] [pks] [ada] [empat] [alasan] [anies] [baswedan] [gak] [bakal] [bisa] [menang] [jadi] [presiden]] |
| 4 | anies baswedan lh yg pantas jadi presiden | [[anies] [baswedan] [lh] [yg] [pantas] [jadi] [presiden]] |
| 5 | pak anies baswedan calon presiden | [[pak] [anies] [baswedan] [calon] [presiden]] |

Tabel 3. 4 Tabel Hasil Tokenize

#### **Remove Stopwords**

*Stopwords* merupakan untuk menghapus kata yang tidak memiliki dampak pada sentimen, namun sering muncul pada kalimat. Tabel 3.5 menunjukan proses *Remove Stopwords* sebagai berikut:

Tabel 3. 5 Tabel Hasil Remove Stopwords

|  |  |  |
| --- | --- | --- |
| No | Kalimat Awal | Hasil Proses |
| 1 | [[pks] [dukung] [anies] [baswedan] [jadi] [calon] [presiden]] | [[pks] [dukung] [anies] [baswedan] [calon] [presiden]] |
| 2 | [[anies] [baswedan] [presiden] [Indonesia]] | [[anies] [baswedan] [presiden] [Indonesia]] |
| 3 | [[walau] [sekarang] [dukung] [demokrat] [dan] [pks] [ada] [empat] [alasan] [anies] [baswedan] [gak] [bakal] [bisa] [menang] [jadi] [presiden]] | [[sekarang] [dukung] [demokrat] [pks] [empat] [alasan] [anies] [baswedan] [gak] [bakal] [menang] [presiden]] |
| 4 | [[anies] [baswedan] [lh] [yg] [pantas] [jadi] [presiden]] | [[anies] [baswedan] [lh] [pantas] [presiden]] |
| 5 | [[pak] [anies] [baswedan] [calon] [presiden]] | [[pak] [anies] [baswedan] [calon] [presiden]] |

#### **Slangwords Filtering**

Tahapan ini melakukan penghilangan kata-kata gaul menjadi kata baku. Kamus gaul diambil dari repository github yang dipost oleh [louisowen6](https://github.com/louisowen6) yang ditunjukan pada Tabel 3.6 sebagai berikut:

Tabel 3. 6 Tabel Hasil Slangwords Filtering

|  |  |  |
| --- | --- | --- |
| No | Kalimat Awal | Hasil Proses |
| 1 | [[pks] [dukung] [anies] [baswedan] [calon] [presiden]] | [[pks] [dukung] [anies] [baswedan] [calon] [presiden]] |
| 2 | [[anies] [baswedan] [presiden] [Indonesia]] | [[anies] [baswedan] [presiden] [Indonesia]] |
| 3 | [[sekarang] [dukung] [demokrat] [pks] [empat] [alasan] [anies] [baswedan] [gak] [bakal] [menang] [presiden]] | [[sekarang] [dukung] [demokrat] [pks] [empat] [alasan] [anies] [baswedan] [tidak] [akan] [menang] [presiden]] |
| 4 | [[anies] [baswedan] [pantas] [presiden]] | [[anies] [baswedan] [pantas] [presiden]] |
| 5 | [[pak] [anies] [baswedan] [calon] [presiden]] | [[pak] [anies] [baswedan] [calon] [presiden]] |

#### **Remove Synonim**

Pada tahapan ini, penulis melakukan penghapusan sinonim dalam kalimat yang sama jika ada. Contoh penerapan proses ini, yaitu:

Tabel 3. 7 Tabel Hasil Remove Synonim

|  |  |  |
| --- | --- | --- |
| No | Kalimat Awal | Hasil Proses |
| 1 | [[pks] [dukung] [anies] [baswedan] [calon] [presiden]] | [[pks] [dukung] [anies] [baswedan] [calon] [presiden]] |
| 2 | [[anies] [baswedan] [presiden] [Indonesia]] | [[anies] [baswedan] [presiden] [Indonesia]] |
| 3 | [[sekarang] [dukung] [demokrat] [pks] [empat] [alasan] [anies] [baswedan] [tidak] [akan] [menang] [presiden]] | [[sekarang] [dukung] [demokrat] [pks] [empat] [alasan] [anies] [baswedan] [tidak] [akan] [menang] [presiden]] |
| 4 | [[alhamdulillah] [mudah] [mudah] [anies] [baswedan] [takdir] [allah] [swt] [presiden] [aamiin] [yra] | [[alhamdulillah] [mudah] [anies] [baswedan] [takdir] [allah] [swt] [presiden] [aamiin] [yra] |
| 5 | [[pak] [anies] [baswedan] [calon] [presiden]] | [[pak] [anies] [baswedan] [calon] [presiden]] |

### ***Support Vector Machine***

#### **Feature Extraction**

Pada tahapan *Feature Extraction*, penulis menggunakan metode TF-IDF (Term Frequency – Inverse Document Frequency) yang melakukan pembobotan data hasil preprocessing berdasarkan banyaknya kemunculan suatu kata pada dokumen. Hasil TF-IDF sendiri berfungsi untuk mengubah data menjadi bentuk numerikal agar fitur-fitur yang telah diproses dapat dimuat kedalam persamaan. Sehingga metode ini dipilih untuk mengekstrak data tersebut. Berikut merupakan gambaran bagaimana ekstraksi fitur menggunakan TF-IDF :

Pertama-tama, memuat corpus hasil *pre-processing* sebagai berikut, tabel sebagai contoh memuat 2 dokumen.

Tabel 3. 8 Tabel Contoh Corpus

|  |  |
| --- | --- |
| Dokumen | Corpus |
| 1 | mari dukung anies baswedan presiden republik indonesia warga negara indonesia kakek pahlawan jasa genius kenal publik prestasi hitung hidup negara satu republik indonesia |
| 2 | anies baswedan bangga rakyat indonesia gubernur dearah khusus ibukota jakarta cocok presiden republik |

Digambarkan pada tabel 3.8 berisi *corpus* untuk mendapatkan nilai TF-IDF dari setiap *term* pada suatu dokumen. Pertama-tama dicari nilai *tf* dari setiap *term* pada suatu dokumen yang digunakan. Pencarian *tf* dilakukan dengan mencari jumlah kemunculan *term* pada suatu dokumen, lalu membaginya dengan jumlah total *term* pada dokumen tersebut. Sedangkan nilai *idf* didapat dengan membagi jumlah dokumen dengan jumlah dokumen yang memuat suatu *term* sehingga nilai idf dapat dicari dengan persamaan berikut ini: [10]

(7)

Dimana:

: Nilai *idf* dari suatu *term*

: Jumlah dokumen

: Jumlah dokumen yang memuat suatu *term*

Selanjutnya setelah nilai *tf* dan *idf* didapatkan, maka nilai TF-IDF dapat dihitung dengan menggunakan persamaan: [10]

(8)

Dimana:

: Bobot suatu *term* pada dokumen

: Nilai *tf* suatu *term* pada suatu dokumen

: Nilai *idf* dari suatu *term*

Berikut ditampilkan hasil perhitungan TF-IDF berdasarkan 2 dokumen diatas.

Tabel 3. 9 Tabel Perhitungan TF-IDF

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Term | *tf* | | *df* | *idf* | tf-idf | |
| Dokumen 1 | Dokumen 2 | Dokumen 1 | Dokumen 2 |
| mari | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| dukung | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| anies | 0,04 | 0,08 | 2 | 1,00 | 0,05 | 0,10 |
| baswedan | 0,04 | 0,08 | 2 | 1,00 | 0,05 | 0,10 |
| presiden | 0,04 | 0,08 | 2 | 1,00 | 0,05 | 0,10 |
| republik | 0,09 | 0,08 | 2 | 1,00 | 0,12 | 0,10 |
| indonesia | 0,13 | 0,08 | 2 | 1,00 | 0,17 | 0,10 |
| warga | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| negara | 0,09 | 0 | 1 | 1,30 | 0,12 | 0 |
| kakek | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| pahlawan | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| jasa | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| genius | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| kenal | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| publik | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| prestasi | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| hitung | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| hidup | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| satu | 0,04 | 0 | 1 | 1,30 | 0,05 | 0 |
| bangga | 0 | 0,08 | 1 | 1,30 | 0 | 0,10 |
| rakyat | 0 | 0,08 | 1 | 1,30 | 0 | 0,10 |
| gubernur | 0 | 0,08 | 1 | 1,30 | 0 | 0,10 |
| daerah | 0 | 0,08 | 1 | 1,30 | 0 | 0,10 |
| khusus | 0 | 0,08 | 1 | 1,30 | 0 | 0,10 |
| ibukota | 0 | 0,08 | 1 | 1,30 | 0 | 0,10 |
| jakarta | 0 | 0,08 | 1 | 1,30 | 0 | 0,10 |
| cocok | 0 | 0,08 | 1 | 1,30 | 0 | 0,10 |

#### **SVM**

Untuk menggunakan algoritma Support Vector Machine, disini perlu dicari terbaik untuk memisahkan tiap kelas. Normalnya dengan data yang tidak terlalu tersebar, hyperplane berupa linear atau polynomial. Namun jika masuk pada kasus penulis, persebaran data yang sangat luas dan dimensi fitur yang tanpa batas, maka diperlukannya penggunaan *kernel trick*, implementasi yang penulis lakukan untuk mengatasi masalah ini adalah menggunakan Gaussian RBF (Radial Basis Function) yang mampu mengatasi masalah diatas.

Jika diberikan dataset dengan 3 label seperti pada tabel 3.10 juga berikut visualisasi dataset yang ditunjukan pada gambar 3.7.

Tabel 3. 10 Contoh Data 2 Dimensi

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **X** | **Y** | **Label** |
| **1** | -1 | -2 | Positif |
| **2** | -2 | -1 | Positif |
| **3** | 4 | 5 | Netral |
| **4** | 5 | 4 | Netral |
| **5** | 7 | 8 | Negatif |
| **6** | 8 | 7 | Negatif |

Gambar 3. 7 Visualisasi Data 2 Dimensi

Dari hasil observasi dataset diatas, dalam penerapan kernel Gaussian RBF (Radial Basis Function) pertama-tama kita perlu menemukan jarak setiap titik dengan titik yang lain. Hal ini dapat dilakukan dengan persamaan *Euclidean Distance*.

(9)

Dimana:

: Jarak *Euclidean Distance*

: Jumlah titik

: Titik vector data

Hasil perhitungan *Euclidean Distance* ditampilkan pada tabel 3.11 sebagai berikut:

Tabel 3. 11 Tabel Jarak Euclidean Distance Per Titik

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | 0,00 | 1,41 | 8,60 | 8,49 | 12,81 | 12,73 |
| **2** | 1,41 | 0,00 | 8,49 | 8,60 | 12,73 | 12,81 |
| **3** | 8,60 | 8,49 | 0,00 | 1,41 | 4,24 | 4,47 |
| **4** | 8,49 | 8,60 | 1,41 | 0,00 | 4,47 | 4,24 |
| **5** | 12,81 | 12,73 | 4,24 | 4,47 | 0,00 | 1,41 |
| **6** | 12,73 | 12,81 | 4,47 | 4,24 | 1,41 | 0,00 |

Selanjutnya, dilakukan perhitungan per data kedalam fungsi kernel Gaussian RBF dengan persamaan sebagai berikut:

(10)

Dimana:

: Nilai Kernel Gaussian RBF

: Nilai Parameter (Gamma)

: Titik vector data

Kita asumsikan data baru yang akan diklasifikasi memiliki vektor (2, 2) sehingga hasil perhitungan setiap data dengan data baru pada tabel 3.12 sebagai berikut:

Tabel 3. 12 Tabel Hasil Nilai Kernel

|  |  |
| --- | --- |
| **Data** | **Kernel Value** |
| 1 | 1,38879E-11 |
| 2 | 1,38879E-11 |
| 3 | 2,26033E-06 |
| 4 | 2,26033E-06 |
| 5 | 3,22134E-27 |
| 6 | 3,22134E-27 |

Lalu semua data dijumlahkan dengan masing-masing labelnya, pada kasus ini perhitungan akan ditampilkan sebagai berikut:

Label Positif = Data 1 + Data 2 = 2,77759E-11

Label Netral = Data 3 + Data 4 = 4,52066E-06

Label Negatif = Data 5 + Data 6 = 6,44268E-27

Dari ketiga label diatas, angka terbesar ada pada Label Netral, sehingga data baru dengan vektor (2,2) masuk kedalam Label Netral.

#### **10-Fold Cross Validation**

Setelah mesin SVM dilatih, untuk menemukan tingkat untuk menemukan tingkat untuk menemukan tingkat untuk menemukan nilai f1-score paling baik dari model SVM, penulis menggunakan metode 10-Fold Cross Validation. Metode ini membagi sebuah dataset menjadi 10 bagian dengan perbandingan data 10% banding 90%. Seperti yang dijelaskan pada subbab 2.7, penggunaan metode ini mengacak posisi data training dan testing berdasarkan pembagian tersebut.

Selanjutnya, akan dilihat dari 10 pembagian data, mana pembagian yang paling baik f1-scorenya untuk diambil model dari *fold* tersebut. Berikut ditampilkan contoh dataset jika menerapkan metode 10-Fold Cross Validation

Tabel 3. 13 Contoh Dataset untuk 10-Fold Cross Validation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Fitur | Label | No | Fitur | Label |
| 1 | Baik | 1 | 11 | Biasa | 0 |
| 2 | Biasa | 0 | 12 | Buruk | -1 |
| 3 | Buruk | -1 | 13 | Sangat Buruk | -1 |
| 4 | Sangat Baik | 1 | 14 | Biasa | 0 |
| 5 | Sangat Buruk | -1 | 15 | Sangat Baik | 1 |
| 6 | Biasa | 0 | 16 | Baik | 1 |
| 7 | Buruk | -1 | 17 | Biasa | 0 |
| 8 | Sangat Baik | 1 | 18 | Buruk | -1 |
| 9 | Sangat Baik | 1 | 19 | Sangat Buruk | -1 |
| 10 | Baik | 1 | 20 | Biasa | 0 |

Misalkan, pada tabel … memuat dataset utuh, lalu pada tabel … memuat dataset yang telah dipecah sebagai data training dan data testing. Dengan data yang dihuruf tebal sebagai data testing, dan data sisanya sebagai data training.

Tabel 3. 14 Tabel Hasil Pembagian Data Training dan Data Testing

|  |  |  |  |
| --- | --- | --- | --- |
| Fold 1 | | | |
| No | Fitur | No | Fitur |
| 1 | Baik | 11 | Biasa |
| **2** | **Biasa** | 12 | Buruk |
| 3 | Buruk | 13 | Sangat Buruk |
| 4 | Sangat Baik | 14 | Biasa |
| 5 | Sangat Buruk | **15** | **Sangat Baik** |
| 6 | Biasa | 16 | Baik |
| 7 | Buruk | 17 | Biasa |
| 8 | Sangat Baik | 18 | Buruk |
| 9 | Sangat Baik | 19 | Sangat Buruk |
| 10 | Baik | 20 | Biasa |

Data tersebut selanjutnya dimasukan kedalam model sebagai data training dan di uji validasinya pada data dengan nomor 2 dan 15 sebagai data testing.

#### **Confusion Matrix**

Confusion matrix merupakan tabel dimana untuk mengevaluasi model klasifikasi, nilai f1-score sebagai nilai yang digunakan untuk menentukan model terbaik [17]. Sehingga penulis menerapkan evaluasi pada penelitian dengan metode confusion matrix. Ilustrasi penulis dalam menentukan f1-score adalah sebagai berikut:

Tabel 3. 15 Tabel Confusion Matrix

|  |  |  |  |
| --- | --- | --- | --- |
|  | Prediksi Positif | Prediksi Netral | Prediksi Negatif |
| Label Positif | 50 | 4 | 0 |
| Label Netral | 20 | 30 | 2 |
| Label Negatif | 13 | 10 | 8 |

Selanjutnya mencari nilai precision, recall, dan f1-score didapat dengan menggunakan persamaan:

(11)

(12)

(13)

Dimana:

*TP* = True Positive

*FP* = False Positive

*FN* = False Negative

F1-score dicari pada setiap labels sehingga hasil f1-score pada setiap label akan ditampilkan sebagai berikut:

Tabel 3. 16 Hasil Perhitungan f1-score

|  |  |  |
| --- | --- | --- |
| **Positif** | Precision | 0,926 |
| Recall | 0,602 |
| F1-score | 1,806 |
| **Netral** | Precision | 0,577 |
| Recall | 0,682 |
| F1-score | 2,046 |
| **Negatif** | Precision | 0,258 |
| Recall | 0,8 |
| F1-score | 2,4 |

Selanjutnya, untuk menemukan nilai F1-score pada model dengan label lebih dari 2, penulis menggunakan macro f1, dimana nilai yang diambil adalah hasil rata-rata nilai f1-score semua kelas, penulis menggunakan nilai macro dikarenakan nilai akan lebih merata untuk dataset yang tidak seimbang [18]. Sehingga nilai f1-score pada confusion matrix diatas adalah 2,084. Lalu dapat dilanjutkan dengan membandingkan nilai f1 pada model dengan nilai f1 pada model yang lain.

## **Perancangan Sarana Pendukung**

Sarana pendukung pada penelitian ini berupa kamus data yang berisikan 3 kamus dibawah ini. Kamus data ini berfungsi sebagai pendukung model dalam melakukan *pre-processing* sehingga data yang telah terfilter dapat diproses dengan lebih baik. Kamus ini bersumber dari Kamus Besar Bahasa Indonesia dan Tesaurus Tematis Bahasa Indonesia.

### **Kamus Kata Dasar**

Kamus kata dasar merupakan sekumpulan kata yang bersumber dari Kamus Besar Bahasa Indonesia yang merupakan bentuk awal dari suatu kata, kamus diambil dari website [KBBI](https://kbbi.kemdikbud.go.id/) dengan pengimplementasikan kamus berasal dari:

Tabel 3. 17 Tabel Sumber Kamus Kata Dasar

|  |  |
| --- | --- |
| Library | Sumber |
| Sastrawi | [sastrawi/kata-dasar.txt at master · sastrawi/sastrawi](https://github.com/sastrawi/sastrawi/blob/master/data/kata-dasar.txt) |
| Root Words | [NLP\_bahasa\_resources/combined\_root\_words.txt at master · louisowen6/NLP\_bahasa\_resources](https://github.com/louisowen6/NLP_bahasa_resources/blob/master/combined_root_words.txt) |

### **Kamus Sinonim Kata**

Kamus sinonim merupakan kumpulan kata dengan kata lain dengan arti yang sama, kamus sinonim bersumber dari Tesaurus Tematis Bahasa Indonesia. Kamus diambil dari website [TTBI](https://tesaurus.kemdikbud.go.id/tematis/) dengan bentuk pengimplementasian berasal dari tabel 3.18

Tabel 3. 18 Tabel Sumber Sinonim

|  |  |
| --- | --- |
| Library | Sumber |
| Tesaurus | [tesaurus/dict.json at master · victoriasovereigne/tesaurus](https://github.com/victoriasovereigne/tesaurus/blob/master/dict.json) |

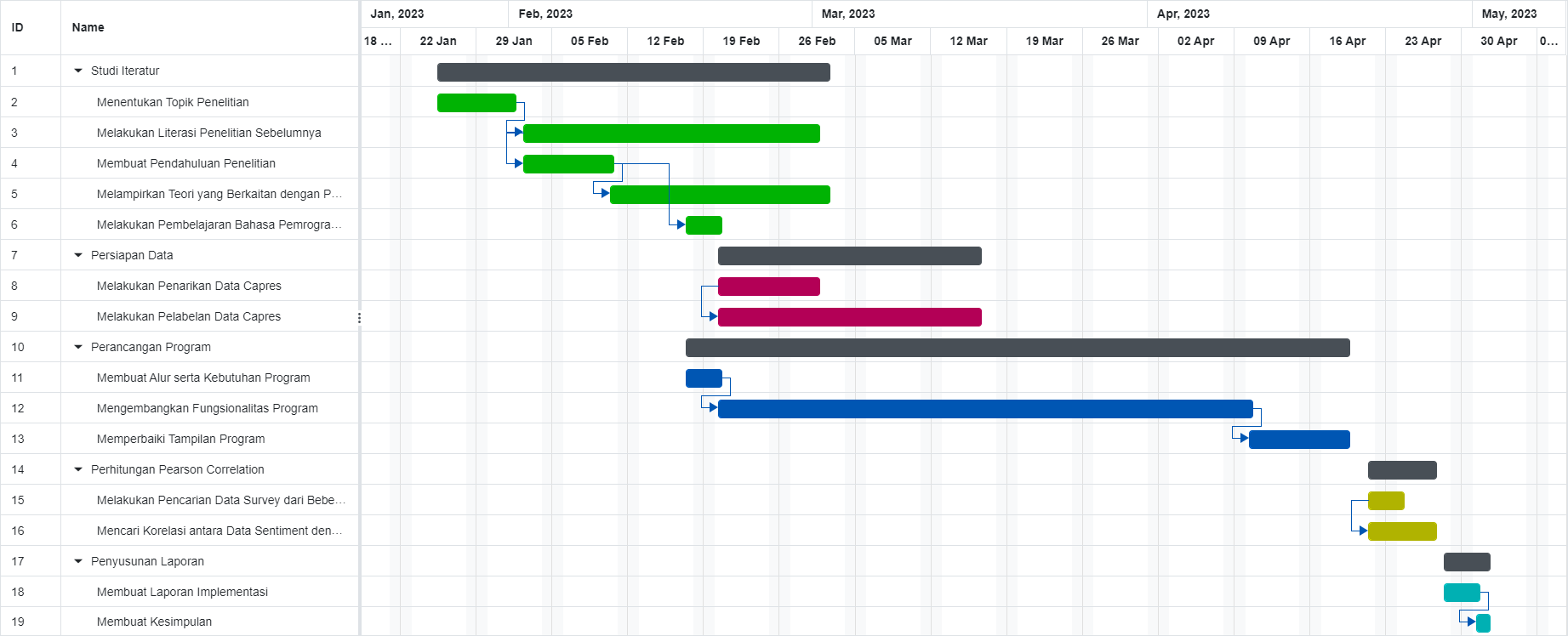
### **Kamus Kata Tidak Baku**

Untuk kamus data kata tidak baku merupakan sekumpulan kata yang sering digunakan namun tidak ada pada Kamus Besar Bahasa Indonesia. Kamus diambil dari repository github [louisowen6](https://github.com/louisowen6) sebagai berikut:

Tabel 3. 19 Tabel Sumber Kata Tidak Baku

|  |  |
| --- | --- |
| Library | Sumber |
| Slang Words | [NLP\_bahasa\_resources/combined\_slang\_words.txt at master · louisowen6/NLP\_bahasa\_resources](https://github.com/louisowen6/NLP_bahasa_resources/blob/master/combined_slang_words.txt) |

## **Jadwal Pengerjaan**

Adapun jadwal pengerjaan pada penelitian ini dijabarkan dalam bentuk Gantt Chart sebagai berikut.

Gambar 3. 8 Gantt Chart Jadwal Pengerjaan