

## BAB IV. ANALISIS DAN PERANCANGAN SISTEM

### 4.1 Analisis Sistem

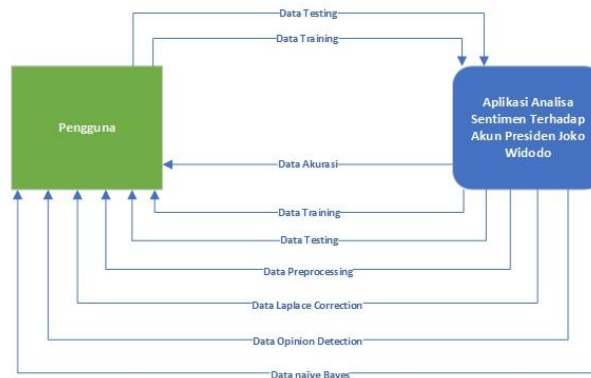
Sistem yang akan dibangun diharapkan dapat mengklasifikasikan *tweet* yang telah didapatkan dari proses *Scraping* menggunakan *scrapy python* ke dalam label positif dan negative pada algoritma naïve bayes dan label opini dan buka opini pada algoritma opinion detection (penggabungan algoritma hmm viterbi dan rule opini). Sistem ini akan memiliki beberapa sub proses yang akan dilakukan untuk menghasilkan satu proses yang utuh.

#### 4.1.1 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan suatu cara atau metode untuk membuat rancangan sebuah sistem yang mana berorientasi pada alur data yang bergerak pada sebuah sistem nantinya. Dalam pembuatan Sistem Informasi, DFD sering digunakan. DFD dibuat oleh para analis untuk membuat sebuah sistem yang baik. Dimana DFD ini nantinya diberikan kepada para programmer untuk melakukan proses coding. Dimana para programmer melakukan sebuah coding sesuai dengan DFD yang dibuat oleh para analis sebelumnya.

##### 4.1.1.1 Diagram Konteks

Sistem Aplikasi Analisa Sentimen Terhadap Akun Presiden Joko Widodo memiliki 1 entitas yaitu entitas pengguna. Entitas pengguna dapat menginputkan data Testing dan data Training yang didapat dari hasil scraping. Entitas pengguna bisa melihat dan memproses data training, data testing, data preprocessing, data laplace correction, data opinion detection dan data naïve bayes.

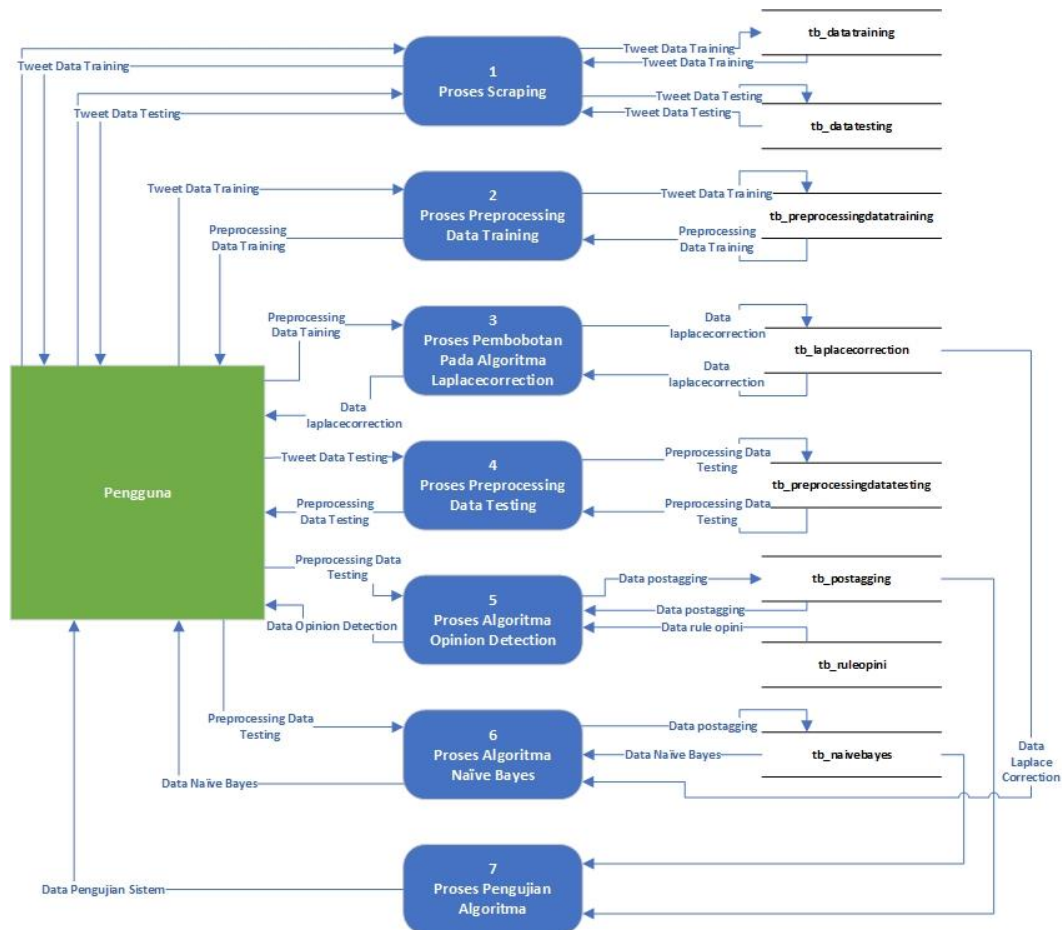


Gambar 4.1. Diagram Konteks Aplikasi Analisa Sentimen Terhadap Akun Presiden Joko Widodo

#### 4.1.1.2 Data Flow Diagram (DFD) Level 1.

DFD Level 1 merupakan penguraian sistem dari Diagram Konteks. Pada gambar 4.1 menjelaskan tentang proses informasi yang terdapat pada sistem, sistem memiliki tujuh proses yaitu:

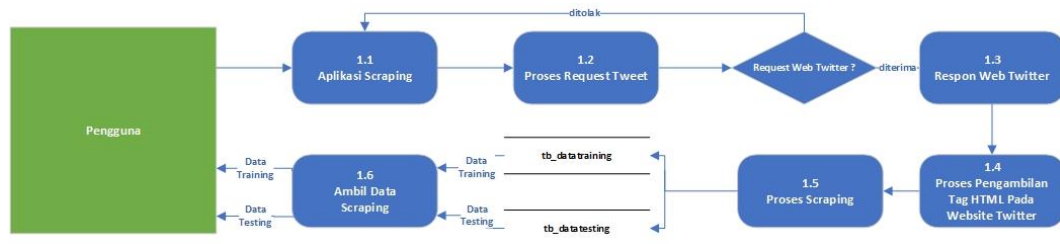
1. Pengguna dapat mendapatkan dataset dari proses scraping.
2. Pengguna dapat melihat dan memproses preprocessing data training.
3. Pengguna dapat melihat dan memproses pembobotan pada algoritma laplacecorrection.
4. Pengguna dapat melihat dan memproses preprocessing data testing.
5. Pengguna dapat melihat dan memproses algoritma opinion detection.
6. Pengguna dapat melihat dan memproses algoritma naïve bayes.
7. Pengguna dapat melihat dan memproses pengujian algoritma.



Gambar 4.2. Data Flow Diagram (DFD) Level 1

#### 4.1.1.3 Data Flow Diagram (DFD) Level 2 Proses Scraping.

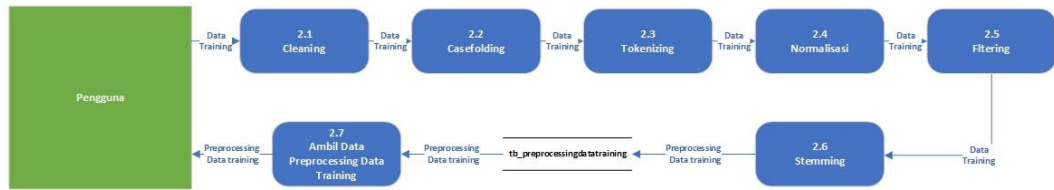
DFD level 2 proses scraping merupakan penguraian proses sistem DFD level 1 pada proses scraping. Pada gambar 4.3 menjelaskan pengguna bisa melakukan proses scraping dengan cara membuka aplikasi sraping python, setelah itu aplikasi melakukan proses request pada web twitter, jika web twtter menolak request dari aplikasi scraping maka akan kembali ke aplikasi scraping itu sendiri, jika web twitter menerima request dari aplikasi scraping maka akan diteruskan untuk mengambil Tag HTML pada website twitter, setelah aplikasi scraping mendapatkan Tag HTML dari website twitter maka proses scraping bisa dilkukan, setelah proses scraping selesai hasil scraping di simpan di tabel tb\_datatraining dan tabel tb\_datatesting, setelah itu pengguna bisa melihat data tersebut melalui proses pengambilan data dari tabel tb\_datatraining dan tabel tb\_datatesting .



Gambar 4.3. Data Flow Diagram (DFD) Level 2 Proses Scraping

#### 4.1.1.4 Data Flow Diagram (DFD) Level 2 Proses Preprocessing Data Training

DFD level 2 proses preprocessing data training merupakan penguraian proses sistem DFD level 1 pada proses preprocessing data training. Pada gambar 4.4 menjelaskan pengguna bisa melakukan pemrosesan preprocessing data training dengan 6 tahap preprocessing, yang pertama yaitu proses cleaning (untuk menghilangkan url, mestion, tag, tanda baca, angka), casefolding (untuk mengubah huruf asli menjadi huruf kecil atau huruf besar semua), tokenizing (untuk memecah kalimat menjadi kata), normalisasi (untuk mengganti kata gaul atau kata tidak baku menjadi kata baku), filtering (untuk menghilangkan kata yang tidak diperlukan (stopword)) dan stemming (untuk mengganti kata berimbuhan menjadi kata dasar), setelah data selesai memproses preprocessing, data dimasukkan ke database pada tabel tb\_preprocessingdatatraining, setelah itu pengguna bisa melihat dan mengambil data preprocessing data training.



Gambar 4.4. Data Flow Diagram (DFD) Level 2 Preprocessing Data Training

#### 4.1.1.5 Data Flow Diagram (DFD) Level 2 Proses Laplace Correction.

DFD level 2 proses laplace correction merupakan penguraian proses sistem DFD level 1 pada proses laplace correction. Pada gambar 4.5 menjelaskan pengguna bisa memproses laplace correction dengan cara menghitung jumlah kata per kata terlebih dahulu, setelah itu proses pembobotan laplace correction bisa di laksanakan, setelah proses laplace correction selesai, data laplace correction dimasukan ke database pada tabel tb\_laplacecorrection, setelah itu pengguna bisa mengambil dan melihat data laplace correction.

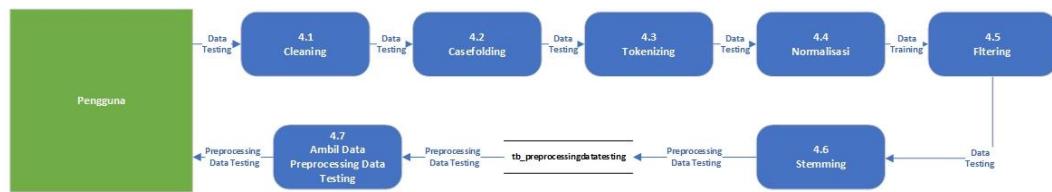


Gambar 4.5. Data Flow Diagram (DFD) Level 2 Proses Laplace Correction

#### 4.1.1.6 Data Flow Diagram (DFD) Level 2 Proses Preprocessing Data Testing

DFD level 2 proses preprocessing data testing merupakan penguraian proses sistem DFD level 1 pada proses preprocessing data testing. Pada gambar 4.6 menjelaskan pengguna bisa melakukan pemrosesan preprocessing data testing dengan 6 tahap preprocessing, yang pertama yaitu proses cleaning (untuk menghilangkan url, mestion, tag, tanda baca, angka), casefolding (untuk mengubah huruf asli menjadi huruf kecil atau huruf besar semua), tokenizing (untuk memecah kalimat menjadi kata), normalisasi (untuk mengganti kata gaul atau kata tidak baku menjadi kata baku), filtering (untuk menghilangkan kata yang tidak diperlukan (stopword)) dan stemming (untuk mengganti kata berimbuhan menjadi kata dasar), setelah data selesai memproses preprocessing, data dimasukan ke database pada

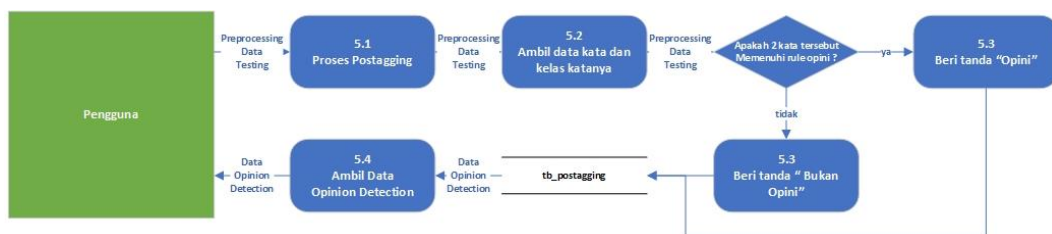
tabel `tb_preprocessingdatatesting`, setelah itu pengguna bisa melihat dan mengambil data preprocessing data testing.



Gambar 4.6. Data Flow Diagram (DFD) Level 2 Preprocessing Data Testing

#### 4.1.1.7 Data Flow Diagram (DFD) Level 2 Proses Opinion Detection

DFD level 2 proses opinion detection merupakan penguraian proses sistem DFD level 1 pada proses opinion detection. Pada gambar 4.7 menjelaskan pengguna bisa melakukan proses opinion detection dengan cara yang pertama adalah proses postagging (memberi tag pada kata) pada tweet data testing, setelah proses postagging selesai selanjutnya ambil 2 data kata dan kelas katanya lalu dilihat apakah kelas kata dari dua kata tersebut mengandung rule opini atau bukan, jika mengandung rule opini maka beri tanda “opini” pada tweet data testing, jika tidak mengandung rule opini maka beri tanda “bukan opini” pada tweet data testing, setelah itu data tersebut di masukan ke dalam database pada tabel `tb_postagging`, setelah itu pengguna bisa mengambil dan melihat data opinion detection.

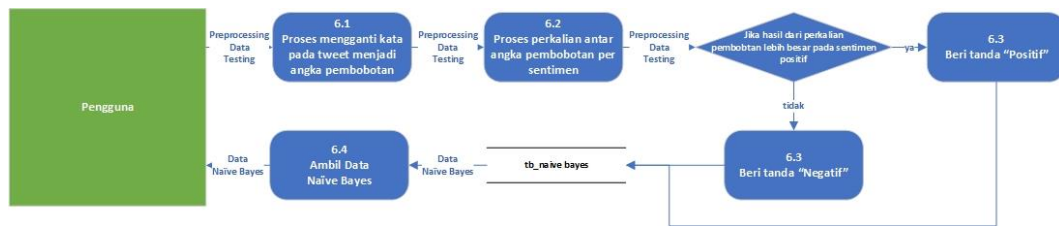


Gambar 4.7. Data Flow Diagram (DFD) Level 2 Proses Opinion Detection

#### 4.1.1.8 Data Flow Diagram (DFD) Level 2 Proses Naïve Bayes

DFD level 2 proses naïve bayes merupakan penguraian proses sistem DFD level 1 pada proses naïve bayes. Pada gambar 4.8 menjelaskan pengguna bisa melakukan proses naïve bayes dengan cara pertama mengganti kata pada tweet menjadi angka pembobotan, setelah itu kalikan antar angka pembobotan per sentimen (sentiment positif dan sentiment negatif), jika hasil dari perkalian pembobotan lebih besar pada sentimen negatif beri tanda “negatif” pada tweet data testing, jika hasil dari perkalian pembobotan lebih besar pada sentimen positif beri tanda “positif” pada tweet data testing, setelah proses memberi tanda selesai,

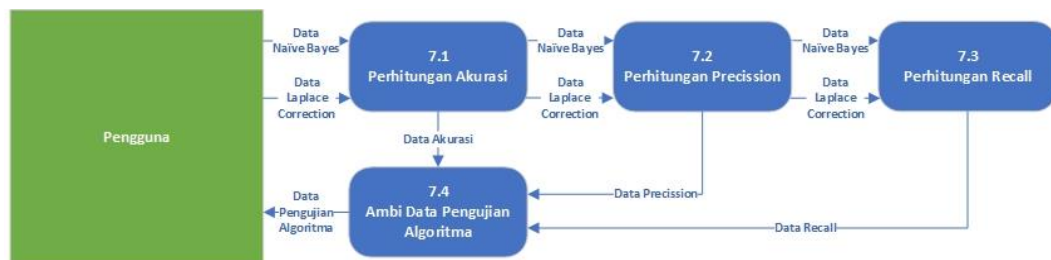
masukan data testing ke database pada tabel tb\_naivebayes, setelah itu pengguna bisa mengambil dan melihat data naïve bayes.



Gambar 4.8. Data Flow Diagram (DFD) Level 2 Proses Naïve Bayes

#### 4.1.1.9 Data Flow Diagram (DFD) Level 2 Proses Pengujian Algoritma

DFD level 2 proses pengujian sistem merupakan penguraian proses sistem DFD level 1 pada proses pengujian sistem. Pada gambar 4.9 menjelaskan pengguna bisa melakukan proses pengujian sistem dengan cara melakukan perhitungan akurasi, precision dan recall dengan membandingkan antara label yang dilabelkan oleh pengguna dan label yang dilabelkan oleh sistem.



Gambar 4.9. Data Flow Diagram (DFD) Level 2 Proses Pengujian Algoritma

## 4.2 Analisis Algoritma

Analisis Algoritma adalah suatu cabang khusus dalam ilmu komputer yang mempelajari karakteristik dan performa dari suatu algoritma dalam menyelesaikan masalah, terlepas dari implementasi algoritma tersebut. Dalam cabang disiplin ini algoritma dipelajari secara abstrak, terlepas dari sistem komputer atau bahasa pemrograman yang digunakan. Ada 4 algoritma pada aplikasi yang di buat penulis, diantaranya yaitu:

### 1. Algoritma preprocessing

Algoritma preprocessing adalah tahapan dimana aplikasi melakukan seleksi data atau text yang akan diproses pada setiap dokumen, penulis menggunakan algoritma preprocessing pada data testing dan data training, ada 6 proses di dalam proses preprocessing yang meliputi :

#### A. Cleaning

Proses Cleaning adalah proses penghapusan url, mention, hastag, tanda baca dan angka di dalam data atau text.

#### B. Casefolding

Proses Casefolding adalah proses mengubah kalimat teks dalam dokumen menjadi konsisten, bisa berupa lowercase atau uppercase.

#### C. Tokenizing

Proses Tokenizing adalah proses mengubah kalimat menjadi pecahan kata.

#### D. Normalisasi

Proses Normalisasi adalah proses mengubah kata gaul atau singkatan menjadi kata yang sesuai Kamus Besar Bahasa Indonesia (KBBI) pada data atau text, contoh kata singkatan dan kata gaul yaitu 'yg', 'dpt', 'otw' dan lain lain.

#### E. Filtering

Proses Filtering adalah proses penghapusan kata stopwords pada suatu data atau text, contoh dari kata stopword adalah kata sambung dan kata yang meliputi 5W1H.

#### F. Stemming

Proses Stemming adalah proses mengubah kata yang berimbuhan menjadi kata dasar, contoh dari proses stemming adalah kata berimbuhan "menyapu" menjadi kata dasar "sapu".

### 2. Algoritma Laplace Correction

Laplace Correction (Laplacian Estimator) atau additive smoothing adalah suatu cara untuk menangani nilai probabilitas 0 (nol). Dari sekian banyak data di training set, pada setiap perhitungan datanya ditambah 1 (satu) dan tidak akan membuat perbedaan yang berarti pada estimasi probabilitas sehingga bisa menghindari kasus nilai probabilitas 0 (nol). Penulis mengimplementasikan Algoritma Laplace Correction pada data training untuk pembobotan kata training, Rumus laplace correction yaitu:

$$\rho_i = \frac{m_i + 1}{n + k}$$

Keterangan :

$P_i$  : laplace correction

$m_i$  : jumlah kata

$n$  : jumlah kata per kata

$k$  : jumlah semua kata

### 3. Algoritma Naïve Bayes

Algoritma Naive Bayes merupakan sebuah metoda klasifikasi menggunakan metode probabilitas dan statisti. Algoritma Naive Bayes memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes. Ciri utama dari algoritma Naive Bayes Classifier ini adalah asumsi yg sangat kuat (naif) akan independensi dari masing-masing kondisi / kejadian. Keuntungan penggunaan adalah bahwa metoda ini hanya membutuhkan jumlah data pelatihan (training data) yang kecil untuk menentukan estimasi parameter yg diperlukan dalam proses pengklasifikasian. Karena yg diasumsikan sebagai variabel independent, maka hanya varians dari suatu variabel dalam sebuah kelas yang dibutuhkan untuk menentukan klasifikasi, bukan keseluruhan dari matriks kovarians. Penulis menggunakan algoritma Naive Bayes untuk mengklasifikasi data tweet yang hasilnya berupa tweet positif dan tweet negatif.

Rumus Algoritma Naive Bayes :

$$P(C|d) = P(C) \prod_{i=1}^n P(W_i | C)$$

$$P(W_i | C) = \frac{\text{count}(w_i, c) + 1}{\text{count}(c) + |V|}$$

Keterangan :

$C$  : Class

$D$  : Dokumen

$W_i$  : Kata ke-i



- (W<sub>i</sub>|C) : Jumlah kata w<sub>i</sub> dalam C
- Count (C) : Jumlah kata di class C
- |V| : Jumlah Vocabulary (Semua Kata)

#### 4. Algoritma POSTagging Viterbi

Algoritma Viterbi adalah algoritma dynamic programming untuk menemukan kemungkinan rangkaian status yang tersembunyi (biasa disebut Viterbi path) yang dihasilkan pada rangkaian pengamatan kejadian, terutama dalam lingkup HMM. Dengan algoritma viterbi maka pengolahan status-status dalam sistem pengenalan suara dapat lebih optimal.

Berikut Contoh Tagset pada POSTagging Viterbi:

Tabel 4.1 Tabel Contoh Tagset POSTagging Viterbi

Tag	Kategori Kata (Desk)	Contoh Kata
CC	Konjungsi	dan, tetapi, atau
IN	Kata Depan	dalam, di, ke, akan, oleh
JJ	Kata Sifat	bersih, bagus, baik
NEG	Negasi	tidak, belum, jangan
NN	Kata Benda	uang, rupiah, meja
NNP	Kata Benda Spesifik	Indonesia, Jokowi
PRP	Kata Ganti Orang	saya, kami, kita, kamu
VB	Kata Kerja	mengatur, melihat, dapat

#### 4.2.1 Contoh Implementasi Algoritma POSTagging Viterbi Manual.

##### A. Proses Training.

##### a. Proses Emission Probabilities.

Proses Emission Probabilities adalah untuk menyatakan seberapa besar kemungkinan munculnya suatu kata diberikan kelas kata tertentu.

Contoh :

1. Data Training Untuk Emmission Probabilities.

Artha	Ilma	dapat	Melihat	Ida
NNP	NNP	VB	VB	NNP

Ahmad	Akan	melihat	Artha
NNP	IN	VB	NNP

Andi	Akan	bertemu	Ida
NNP	IN	VB	NNP

Ida	bertemu	Ahmad
NNP	VB	NNP

2. Training Emmission Probabilities Tahap 1.

	NNP	VB	IN
Artha	2	0	0
Ilma	1	0	0
Dapat	0	1	0
Melihat	0	2	0
Ida	3	0	0
Ahmad	2	0	0
Akan	0	0	2
Andi	1	0	0
Bertemu	0	2	0
Jumlah	9	5	2

3. Training Emmission Probabilities Tahap 2.

	NNP	VB	IN
Artha	2/9	0/5	0/2
Ilma	1/9	0/5	0/2

Dapat	0/9	1/5	0/2
Melihat	0/9	2/5	0/2
Ida	3/9	0/5	0/2
Ahmad	2/9	0/5	0/2
Akan	0/9	0/5	2/2
Andi	1/9	0/5	0/2
Bertemu	0/9	2/5	0/2

b. Proses Transition Probabilities.

Proses Transition Probabilities adalah untuk menyatakan seberapa besar kemungkinan munculnya suatu kelas kata dimana sebelumnya muncul kelas kata tertentu.

1. Data Training Untuk Transition Probabilities.

	Artha	ilma	dapat	melihat	ida	
<S>	NNP	NNP	VB	VB	NNP	<E>

	Ahmad	akan	melihat	Artha	
<S>	NNP	IN	VB	NNP	<E>

	Andi	akan	bertemu	Ida	
<S>	NNP	IN	VB	NNP	<E>

	Ida	bertemu	Ahmad	
<S>	NNP	IN	VB	<E>

2. Training Transition Probabilities Tahap 1.

	NNP	VB	IN	<E>	Jumlah
<S>	4	0	0	0	4
NNP	1	2	2	4	9
VB	4	1	0	0	5
IN	0	2	0	0	2

### 3. Training Transition Probabilities Tahap 2.

	NNP	VB	IN	<E>
<S>	4/4	0/4	0/4	0/4
NNP	1/9	2/9	2/9	4/9
VB	4/5	1/5	0/5	0/5
IN	0/2	2/2	0/2	0/2

#### B. Proses Testing.

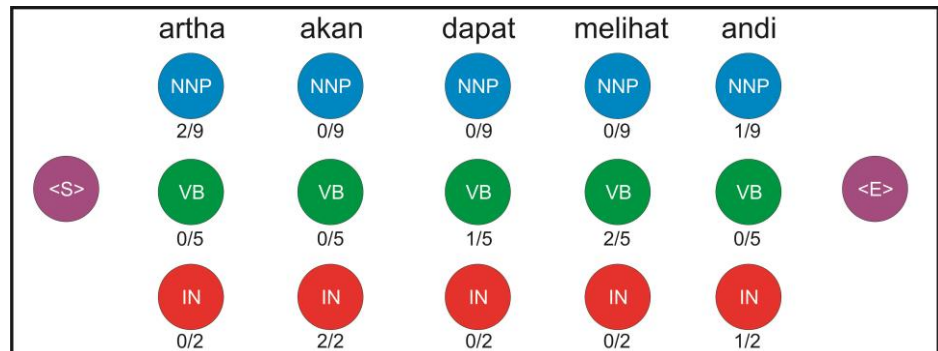
##### a. Contoh Data Testing Viterbi.

Contoh Data Testing Viterbi adalah data yang akan diolah oleh proses testing viterbi.

Artha	Akan	dapat	Melihat	andi
-------	------	-------	---------	------

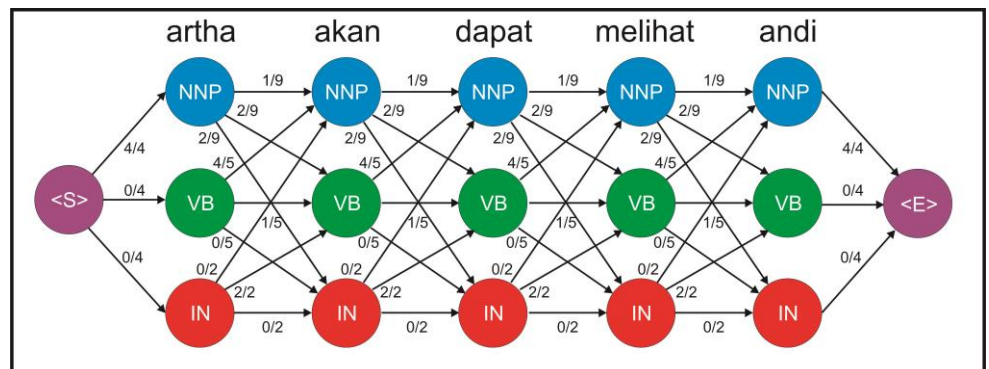
##### b. Proses Testing Emmission Probabilities.

Nilai dari proses Testing Emmission Probabilities diambil dari data Training Emmission Probabilities.

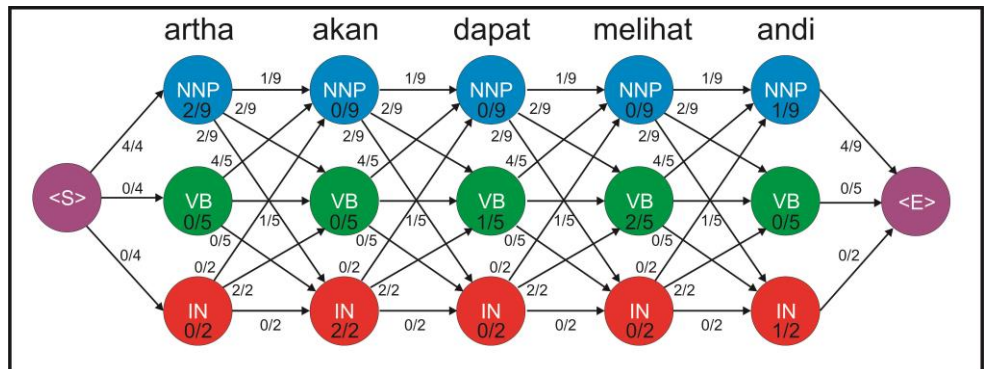


##### c. Proses Testing Transition Probabilities.

Nilai dari proses Testing Transition Probabilities diambil dari data Training Emmission Probabilities.



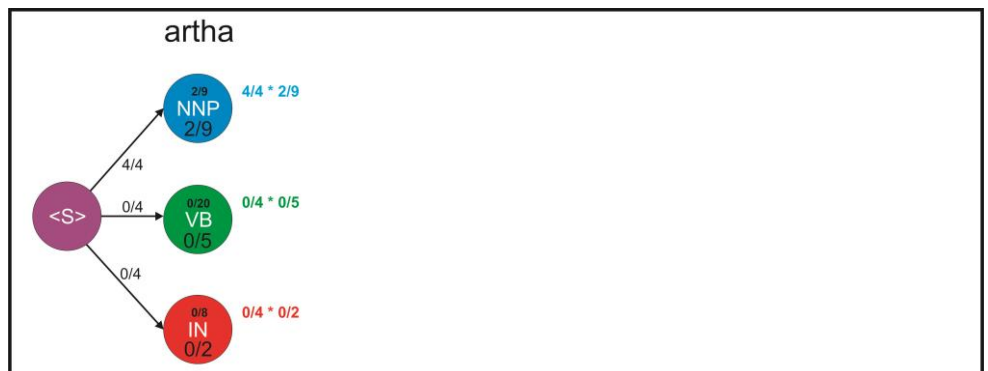
d. Proses Penggabungan Testing Emmission Probabilities dan Testing Training Probabilities.



e. Proses Testing Viterbi.

Diproses ini penulis akan menjelaskan proses testing viterbi per kata pada data testing “artha akan dapat melihat andi”.

1. Tahap algoritma Viterbi pada proses kata “artha”.



	Diketahui	Proses	Hasil
Hasil NNP[1]	Data Emmission <S> dan NNP[1] = 4/4 Data Transition NNP[1] = 2/9	$4/4 * 2/9$	2/9
Hasil VB[1]	Data Emmission <S> dan VB[1] = 0/4 Data Transition VB[1] = 0/5	$0/4 * 0/5$	0/20
Hasil IN[1]	Data Emmission <S> dan IN[1] = 0/4 Data Transition IN[1] = 0/2	$0/4 * 0/2$	0/8

Kesimpulan Tahap algoritma Viterbi pada proses kata “artha”:

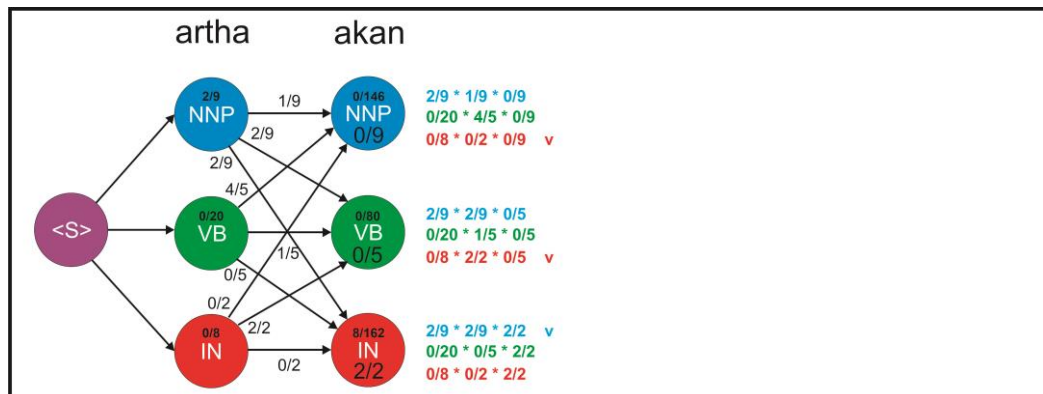
Bandingkan hasil tag dari tag NNP[1], VB[1], IN[1], dan cari hasil tag terbesar dari tag tersebut.

Hasil NNP[1]	Hasil VB[1]	Hasil IN [1]	Hasil Tag terbesar

2/9	0/20	0/8	NNP[1]
-----	------	-----	--------

Hasil perbandingan terbesar dari tag NNP[1], VB[1], IN[1] adalah Tag NNP[1] atau NNP yang bernilai 2/9, dan bisa disimpulkan kata “artha” memiliki tag NNP.

2. Tahap algoritma Viterbi pada proses kata “akan”.



	Diketahui	Proses	Hasil	Status
Hasil NNP[2]	Data Hasil Transition NNP[1] = 2/9 Data Emmission NNP[1] dan NNP[2] = 1/9 Data Transition NNP[2] = 0/9	2/9 * 1/9 * 0/9	0/729	
	Data Hasil Transition VB[1] = 0/20 Data Emmission VB[1] dan NNP[2] = 4/5 Data Transition NNP[2] = 0/9	0/20 * 4/5 * 0/9	0/900	
	Data Hasil Transition IN[1] = 0/8 Data Emmission IN[1] dan NNP[2] = 0/2 Data Transition NNP[2] = 0/9	0/8 * 0/2 * 0/9	0/144	V

\*Cari yang ter besar dari tag NNP[2], hasil yang terbesar dari NNP[2] pada proses algoritma viterbi kata “akan” adalah 0/144.

	Diketahui	Proses	Hasil	Status
Hasil VB[2]	Data Hasil Transition NNP[1] = 2/9 Data Emmission NNP[1] dan VB[2] = 2/9 Data Transition VB[2] = 0/5	2/9 * 1/9 * 0/9	0/405	
	Data Hasil Transition VB[1] = 0/20 Data Emmission VB[1] dan VB[2] = 1/5	0/20 * 1/5 * 0/5	0/500	

	Data Transition VB[2] = 0/5			
	Data Hasil Transition IN[1] = 0/8 Data Emmission IN[1] dan VB[2] = 2/2 Data Transition VB[2] = 0/5	0/8 * 2/2 * 0/5	0/80	V

\*Cari yang terbesar dari tag VB[2], hasil yang terbesar dari VB[2] pada proses algoritma viterbi kata “akan” adalah 0/80.

	Diketahui	Proses	Hasil	Status
Hasil IN[2]	Data Hasil Transition NNP[1] = 2/9 Data Emmission NNP[1] dan IN[2] = 2/9 Data Transition IN[2] = 2/2	2/9 * 2/9 * 2/2	8/162	V
	Data Hasil Transition VB[1] = 0/20 Data Emmission VB[1] dan IN[2] = 0/5 Data Transition IN[2] = 2/2	0/20 * 0/5 * 2/2	0/200	
	Data Hasil Transition IN[1] = 0/8 Data Emmission IN[1] dan IN[2] = 0/2 Data Transition IN[2] = 2/2	0/8 * 0/2 * 2/2	0/32	

\*Cari yang terbesar dari tag IN[2], hasil yang terbesar dari IN[2] pada proses algoritma viterbi kata “akan” adalah 8/162.

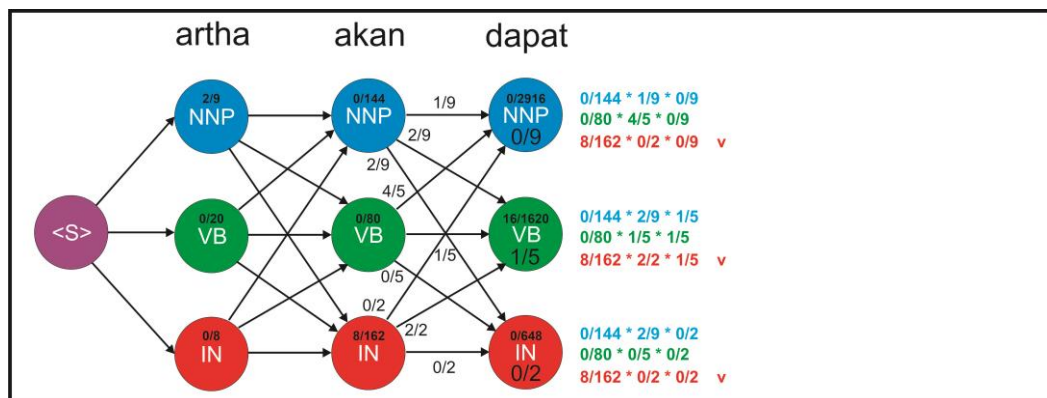
Kesimpulan Tahap algoritma Viterbi pada proses kata “akan”:

Bandingkan hasil tag terbesar dari tag NNP[2], VB[2], IN[2], dan cari hasil tag terbesar dari yang terbesar.

Hasil NNP[2] (terbesar)	Hasil VB[2] (terbesar)	Hasil IN [2] (terbesar)	Hasil Tag terbesar
0/146	0/80	8/16	IN[2]

Hasil perbandingan antara Tag terbesar dari yang terbesar adalah Tag IN[2] atau IN yang bernilai 8/16, dan bisa disimpulkan kata “akan” memiliki tag IN.

3. Tahap algoritma Viterbi pada proses kata “dapat”.



	Diketahui	Proses	Hasil	Status
Hasil NNP[3]	Data Hasil Transition NNP[2] (terbesar) = 0/144 Data Emmission NNP[2] dan NNP[3] = 1/9 Data Transition NNP[2] = 0/9	0/144 * 1/9 * 0/9	0/11664	
	Data Hasil Transition VB[1] (terbesar) = 0/80 Data Emmission VB[1] dan NNP[2] = 4/5 Data Transition NNP[2] = 0/9	0/80 * 4/5 * 0/9	0/3600	
	Data Hasil Transition IN[1] (terbesar) = 8/162 Data Emmission IN[1] dan NNP[2] = 0/2 Data Transition NNP[2] = 0/9	8/162 * 0/2 * 0/9	0/2916	V

\*Cari yang ter besar dari tag NNP[3], hasil yang terbesar dari NNP[3] pada proses algoritma viterbi kata “dapat” adalah 0/2916.

	Diketahui	Proses	Hasil	Status
Hasil VB[3]	Data Hasil Transition NNP[2] (terbesar) = 0/144	0/144 * 2/9 * 1/5	0/6480	



	Data Emmission NNP[2] dan VB[3] = 2/9 Data Transition VB[3] = 1/5			
	Data Hasil Transition VB[2] (terbesar) = 0/80 Data Emmission VB[2] dan VB[3] = 0/80 Data Transition VB[3] = 1/5	0/80 * 1/5 * 1/5	0/2000	
	Data Hasil Transition IN[2] (terbesar) = 8/162 Data Emmission IN[2] dan VB[3] = 2/2 Data Transition VB[3] = 1/5	8/162 * 2/2 * 1/5	16/1620	V

\*Cari yang ter besar dari tag VB[3], hasil yang terbesar dari VB[3] pada proses algoritma viterbi kata “dapat” adalah 16/1620.

	Diketahui	Proses	Hasil	Status
	Data Hasil Transition NNP[2] (terbesar) = 0/144 Data Emmission NNP[2] dan IN[3] = 2/9 Data Transition IN[3] = 0/2	0/144 * 2/9 * 0/2	0/2592	
Hasil IN[3]	Data Hasil Transition VB[2] (terbesar) = 0/80 Data Emmission VB[2] dan IN[3] = 0/5 Data Transition IN[2] = 0/2	0/80 * 0/5 * 0/2	0/800	
	Data Hasil Transition IN[2] (terbesar) = 8/162 Data Emmission IN[2] dan IN[3] = 0/2 Data Transition IN[3] = 0/2	8/162 * 0/2 * 0/2	0/648	V

\*Cari yang ter besar dari tag IN[3], hasil yang terbesar dari IN[3] pada proses algoritma viterbi kata “dapat” adalah 0/648.

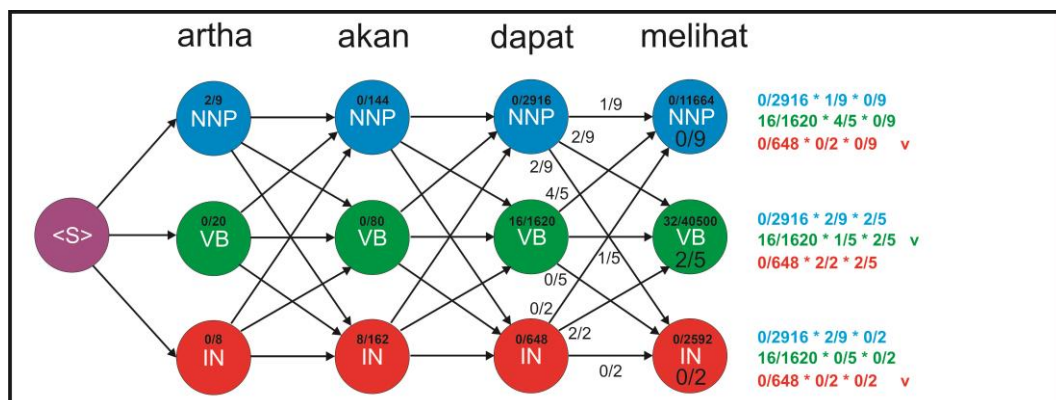
Kesimpulan Tahap algoritma Viterbi pada proses kata “dapat”:

Bandingkan hasil tag terbesar dari tag NNP[3], VB[3], IN[3], dan cari hasil tag terbesar dari yang terbesar.

Hasil NNP[3] (terbesar)	Hasil VB[3] (terbesar)	Hasil IN [3] (terbesar)	Hasil Tag terbesar
0/2916	16/1620	0/648	VB[3]

Hasil perbandingan antara Tag terbesar dari yang terbesar adalah Tag VB[3] atau VB yang bernilai 16/1620.

4. Tahap algoritma Viterbi pada proses kata “melihat”.



	Diketahui	Proses	Hasil	Status
Hasil NNP[4]	Data Hasil Transition NNP[3] (terbesar) = 0/2916 Data Emmission NNP[3] dan NNP[4] = 1/9 Data Transition NNP[4] = 0/9	0/2916 * 1/9 * 0/9	0/236129	
	Data Hasil Transition VB[3] (terbesar) = 0/1620 Data Emmission VB[3] dan NNP[4] = 4/5 Data Transition NNP[4] = 0/9	0/1620 * 4/5 * 0/9	0/72900	
	Data Hasil Transition IN[3] (terbesar) = 0/648 Data Emmission IN[3] dan NNP[4] = 0/2 Data Transition NNP[4] = 0/9	0/648 * 0/2 * 0/9	0/11664	V

\*Cari yang ter besar dari tag NNP[4], hasil yang terbesar dari NNP[4] pada proses algoritma viterbi kata “melihat” adalah 0/11664.

	Diketahui	Proses	Hasil	Status
Hasil VB[4]	Data Hasil Transition NNP[3] (terbesar) = 0/2916 Data Emmision NNP[3] dan VB[4] = 2/9 Data Transition VB[4] = 2/5	0/2916 * 2/9 * 2/5	0/131220	
	Data Hasil Transition VB[3] (terbesar) = 16/1620 Data Emmision VB[3] dan VB[4] = 1/5 Data Transition VB[4] = 2/5	16/1620 * 1/5 * 2/5	32/40500	V
	Data Hasil Transition IN[3] (terbesar) = 0/648 Data Emmision IN[3] dan VB[4] = 2/2 Data Transition VB[4] = 2/5	0/648 * 2/2 * 2/5	0/6480	

\*Cari yang ter besar dari tag VB[4], hasil yang terbesar dari VB[4] pada proses algoritma viterbi kata “melihat” adalah 32/40500.

	Diketahui	Proses	Hasil	Status
Hasil IN[4]	Data Hasil Transition NNP[3] (terbesar) = 0/2916 Data Emmision NNP[3] dan IN[4] = 2/9 Data Transition IN[3] = 0/2	0/2916 * 2/9 * 0/2	0/52488	
	Data Hasil Transition VB[3] (terbesar) = 16/1620 Data Emmision VB[3] dan IN[4] = 0/5 Data Transition IN[4] = 0/2	16/1620 * 0/5 * 0/2	0/16200	
	Data Hasil Transition IN[3] (terbesar) = 0/648 Data Emmision IN[3] dan IN[4] = 0/2 Data Transition IN[4] = 0/2	0/648 * 0/2 * 0/2	0/2592	V

\*Cari yang terbesar dari tag IN[4], hasil yang terbesar dari IN[4] pada proses algoritma viterbi kata “melihat” adalah 0/2592.

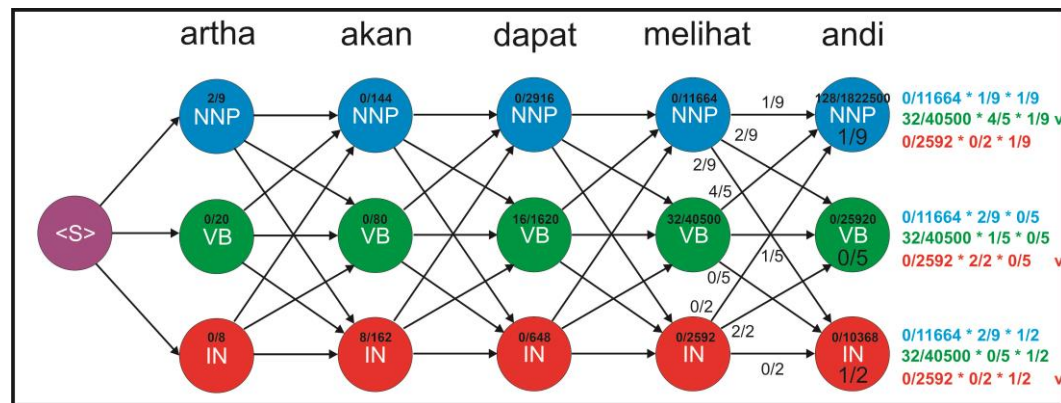
Kesimpulan Tahap Algoritma Viterbi pada proses kata “melihat”:

Bandingkan hasil tag terbesar dari tag NNP[4], VB[4], IN[4], dan cari hasil tag terbesar dari yang terbesar.

Hasil NNP[4] (terbesar)	Hasil VB[4] (terbesar)	Hasil IN [4] (terbesar)	Hasil Tag terbesar
0/11664	32/40500	0/2592	VB[4]

Hasil perbandingan antara Tag terbesar dari yang terbesar adalah Tag VB[4] atau VB yang bernilai 32/40500.

5. Tahap algoritma Viterbi pada proses kata “andi”.



	Diketahui	Proses	Hasil	Status
Hasil NNP[5]	Data Hasil Transition NNP[4] (terbesar) = 0/11664 Data Emmission NNP[4] dan NNP[5] = 1/9 Data Transition NNP[5] = 1/9	0/11664 * 1/9 *	0/944784	
	Data Hasil Transition VB[4] (terbesar) = 32/40500 Data Emmission VB[4] dan NNP[5] = 4/5 Data Transition NNP[5] = 1/9	32/40500 * 4/5 *	128/1822500	V
	Data Hasil Transition IN[4] (terbesar) = 0/2592	0/2592 * 0/2 * 1/9	0/11664	

	Data Emmission IN[4] dan NNP[5] = 0/2 Data Transition NNP[5] = 1/9			
--	--	--	--	--

\*Cari yang ter besar dari tag NNP[5], hasil yang terbesar dari NNP[5] pada proses algoritma viterbi kata “andi” adalah 128/1822500.

	Diketahui	Proses	Hasil	Status
Hasil VB[5]	Data Hasil Transition NNP[4] (terbesar) = 0/11664 Data Emmission NNP[4] dan VB[5] = 2/9 Data Transition VB[5] = 0/5	0/11664 * 2/9 *	0/524880	
	Data Hasil Transition VB[4] (terbesar) = 32/40500 Data Emmission VB[4] dan VB[5] = 1/5 Data Transition VB[5] = 0/5	32/40500 * 1/5 *	0/1012500	
	Data Hasil Transition IN[4] (terbesar) = 0/2592 Data Emmission IN[4] dan VB[5] = 2/2 Data Transition VB[5] = 0/5	0/2592 * 2/2 * 0/5	0/25920	V

\*Cari yang ter besar dari tag VB[5], hasil yang terbesar dari VB[5] pada proses algoritma viterbi kata “andi” adalah 0/25920.

	Diketahui	Proses	Hasil	Status
Hasil IN[5]	Data Hasil Transition NNP[4] (terbesar) = 0/11664 Data Emmission NNP[4] dan IN[5] = 2/9 Data Transition IN[5] = 1/2	0/11664 * 2/9 * 1/2	0/209952	

Data Hasil Transition VB[4] (terbesar) = 32/40500 Data Emmission VB[4] dan IN[5] = $1/2$ Data Transition IN[5] = 0/2	$32/40500$ $* 1/2 *$ $0/2$	$0/16200$	
Data Hasil Transition IN[4] (terbesar) = 0/2592 Data Emmission IN[4] dan IN[5] = 0/2 Data Transition IN[4] = $1/2$	$0/2592 *$ $0/2 * 1/2$	$0/10368$	V

\*Cari yang ter besar dari tag IN[5], hasil yang terbesar dari IN[5] pada proses algoritma viterbi kata “andi” adalah 0/10368.

Kesimpulan Tahap Algoritma Viterbi pada proses kata “andi”:

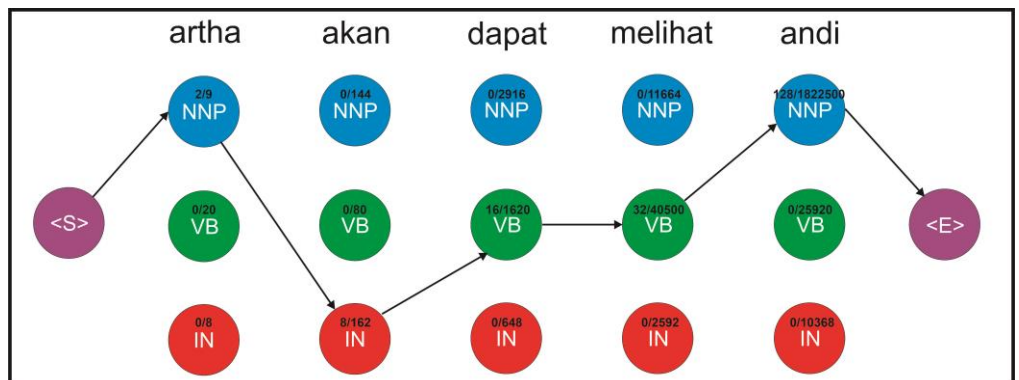
Bandingkan hasil tag terbesar dari tag NNP[5], VB[5], IN[5], dan cari hasil tag terbesar dari yang terbesar.

Hasil NNP[5] (terbesar)	Hasil VB[5] (terbesar)	Hasil IN [5] (terbesar)	Hasil Tag terbesar
128/1822500	0/25920	0/10368	NNP[5]

Hasil perbandingan antara Tag terbesar dari yang terbesar adalah Tag NNP[5] atau NNP yang bernilai 128/1822500.

f. Kesimpulan dan Hasil Akhir Proses POSTagging Viterbi.

Tahap ini adalah hasil akhir viterbi pada kalimat “artha akan dapat melihat andi” dapat melihat andi”.



Kata :	Artha	akan	dapat	melihat	andi
Tag	NNP	IN	VB	VB	NNP
Viterbi :					
Nilai	2/9	8/162	16/1620	32/40500	128/1822500
Viterbi :					

Kesimpulan dari proses contoh implementasi algoritma POSTagging Viterbi adalah kata berhasil di tag berdasarkan tagset yang telah ditentukan pada table tagset viterbi (tagset bisa di lihat pada bab 4.2).

#### 4.2.2 Contoh Implementasi Algoritma Klasifikasi Naïve Bayes dan Laplace Correction Manual.

##### A. Proses Training

##### 1. Daftar Data training

##### a. Data training sentiment positif



arif fikri  
@fikri\_arif

Membalas @BullshitGnfp212 dan @detikcom

Jadilah pendukung yang cerdas.saya yakin pak @jokowi pak @aniesbaswedan pak @ridwankamil pak @ganjarpranowo dan gubernur lainnya berharap kebijakan PSBB bisa diterapkan oleh semua pihak.lebih bijak salurkan langsung bantuan ke rumah2 dan ke kontrakan2 biar adil&merata.

3.53 PM · 12 Apr 2020 · Twitter Web App



Mas Adji  
@adjibaya

Membalas @1ruk4\_dai\_suk1 @jokowi dan 4 lainnya

iya Neng... dan syukurnya secara individu maupun lembaga banyak yg tergerak untuk menyalurkan bantuan kepada saudara2 kita yg membutuhkan... hikmah dibalik Corona.

#SemuaPeduliCovid19 👍👍

7.05 PM · 13 Apr 2020 · Twitter for Android

b. Data training sentiment negatif



2. Tahap Prerprocessing

a. Cleaning

Label +	Label -
Jadilah pendukung yang cerdas saya yakin pak pak pak pak dan gubernur lainnya berharap kebijakan PSBB bisa diterapkan oleh semua pihak lebih bijak salurkan langsung bantuan ke dan ke biar adil merata	Pak presiden lebih baik ga usah ada BLT atau BANSOS dll klo selalu tdk tepat sasaran tdk adil Cuma bikin sakit hati apalagi klo penyaluran bantuannya melalui RT dlu
iya Neng dan syukurnya secara individu maupun lembaga banyak yg tergerak untuk menyalurkan bantuan kepada kita yg membutuhkan hikmah dibalik Corona	jokowi Saya merasa proses penyaluran bantuan ditengah pandemi virus corona ini tidak merata terutama yg berada di daerah bapak masih banyak org yg mbutuhkan kcl



b. Casefolding

Label +	Label -
jadilah pendukung yang cerdas saya yakin pak pak pak pak dan gubernur lainnya berharap kebijakan psbb bisa diterapkan oleh semua pihak lebih bijak salurkan langsung bantuan ke dan ke biar adil merata	pak presiden lebih baik ga usah ada blt atau bansos dll klo selalu tdk tepat sasaran tdk adil cuma bikin sakit hati apalagi klo penyaluran bantuannya melalui rt dlu
iya neng dan syukurnya secara individu maupun lembaga banyak yg tergerak untuk menyalurkan bantuan kepada kita yg membutuhkan hikmah dibalik corona	jokowi saya merasa proses penyaluran bantuan ditengah pandemi virus corona ini tidak merata terutama yg berada di daerah bapak masih banyak org yg mbutuhkan kcl

c. Tokonizing

Label +	Label -
Jadilah, pendukung, yang, cerdas, saya, yakin, pak, pak, pak, pak, dan, gubernur, lainnya, berharap, kebijakan, psbb, bisa, diterapkan, oleh, semua, pihak, lebih, bijak, salurkan, langsung, bantuan, ke, dan, ke, biar, adil, merata	Pak, presiden, lebih, baik, ga usah, ada, blt, atau, bansos, dll, klo, selalu, tdk, tepat, sasaran, tdk, adil, cuma, bikin, sakit, hati, apalagi, klo, penyaluran, bantuannya, melalui, rt, dlu
iya, neng, dan, syukurnya, secara, individu, maupun, lembaga, banyak, yg, tergerak, untuk, menyalurkan, bantuan, kepada, kita, yg,	jokowi, saya, merasa, proses, penyaluran, bantuan, ditengah, pandemik, virus, corona, ini, tidak, merata, terutama, yg, berada, di,

membutuhkan, hikmah, dibalik, corona	daerah, bapak, masih, banyak, org, yg, mbutuhkan, kcl,
---	---

d. Normalisasi

Label +	Label -
jadilah, pendukung, yang, cerdas, saya, yakin, pak, pak, pak, pak, dan, gubernur, lainya, berharap, kebijakan, psbb, bisa, diterapkan, oleh, semua, pihak, lebih, bijak, salurkan, langsung, bantuan, ke, dan, ke, biar, adil, merata	pak, presiden, lebih, baik, tidak, perlu, ada, bantuan, langsung, tunai, atau, bantuan, sosial, dan, lain, lain, kalua, selalu, tidak, tepat, sasaran, tidak, adil, Cuma, bikin, sakit, hati, apalagi, kalua, penyaluran, bantuannya, melalui, rt, dulu
iya, neng, dan, syukurnya, secara, individu, maupun, lembaga, banyak, yang, tergerak, untuk, menyalurkan, bantuan, kepada, kita, yang, membutuhkan, hikmah, dibalik, corona	jokowi, saya, merasa, proses, penyaluran, bantuan, ditengah, pandemik, virus, corona, ini, tidak, merata, terutama, yang, berada, di, daerah, bapak, masih, banyak, orang, yang, mbutuhkan, kcl

e. Filtering

Label +	Label -
pendukung, cerdas, gubernur, lainya, berharap, kebijakan, psbb, diterapkan, bijak, salurkan, langsung, bantuan, biar, adil, merata	presiden, baik, tidak, bantuan, langsung, tunai, bantuan, sosial, tidak, tepat, sasaran, tidak, adil, bikin, sakit, hati, penyaluran, bantuannya, rt
neng, syukurnya, individu, lembaga, bergerak, menyalurkan, bantuan,	jokowi, proses, penyaluran, bantuan, ditengah, pandemic,

membutuhkan, hikmah, dibalik, corona	virus, corona, tidak, merata, daerah, orang, mbutuhkan, kcl
--------------------------------------	---

f. Stemming

Label +	Label -
dukung, cerdas, gubernur, lai, harap, bijak, terap, bijak, salur, langsung, bantu, biar, adil, rata	presiden, baik, tidakbantu, langsung, tunai, bantu, sosial, tidaktepat, sasar, tidakadil, bikin, sakit, hati, salur, bantu
neng, syukur, individu, lembaga, gerak, salur, bantu, butuh, hikmah, balik, corona	proses, salur, bantu, tengah, pandemi, virus, corona, tidakrata, daerah, orang

3. Tahap Laplace Correction

Diketahui :

a. Jumlah kata per kata pada data training

▪ Positif

adil	baik	balik	bantu	biar	bijak	bikin	butuh	cerdas	corona
1	0	1	2	1	2	0	1	1	1

daerah	dukung	gerak	gubernur	harap	hati	hikmah	individu	lai	langsung
0	1	1	1	1	0	1	1	1	1

lembaga	neng	orang	pandemi	presiden	proses	rata	sakit	salur	sasar
1	1	0	0	0	0	1	0	2	0

sosial	syukur	tengah	terap	tidakadil	tidakbantu	tidakrata	tidaktepat
0	1	0	1	0	0	0	0

tunai	virus
0	0

- Negatif

adil	baik	balik	bantu	biar	bijak	bikin	butuh	cerdas	corona
0	1	0	3	0	0	1	0	0	1

daerah	dukung	gerak	gubernur	harap	hati	hikmah	individu	lai	langsung
1	0	0	0	0	1	0	0	0	1

lembaga	neng	orang	pandemi	presiden	proses	rata	sakit	salur	sasar
0	0	1	1	1	1	0	1	2	1

sosial	syukur	tengah	terap	tidakadil	tidakbantu	tidakrata	tidaktepat
1	0	1	0	1	1	1	1

tunai	virus
1	1

b. Pembobotan Laplace Correction

- Positif

Ditahap ini penulis akan memberikan contoh perhitungan kata positif (bijak) di pembobotan laplace correction.

Diketahui :

Jumlah semua kata (positif) : 25

Jumlah kata (positif) : 40

Jumlah kata “bijak” (positif) : 2

Rumus:

$P(\text{kata}|+)$  :  $(\text{Jumlah kata per-kata}+1) /$   
 $(\text{jumlah kata} + \text{jumlah semua kata})$

$P(\text{bijak}|+)$  :  $(2+1) / (40+25)$

:  $3 / 65$

: 0.046153

adil	baik	balik	bantu	biar	Bijak	bikin	butuh
0.0307	0.0153	0.0307	0.0461	0.0307	0.0461	0.0153	0.0307

cerdas	corona	daerah	dukung	gerak	gubernur	harap	hati
0.0307	0.0307	0.0153	0.0307	0.0307	0.0307	0.0307	0.0153

hikmah	individu	lai	langsung	lembaga	neng	orang	pandemi
0.0307	0.0307	0.0307	0.0307	0.0307	0.0307	0.0153	0.0153

presiden	proses	rata	sakit	Salur	Sasar	sosial	syukur
0.0153	0.0153	0.0307	0.0153	0.0461	0.0153	0.0153	0.0307

tengah	terap	tidakadil	tidakbantu	tidakrata	tidaktepat	tunai	virus
0.0153	0.0307	0.0153	0.0153	0.0153	0.0153	0.0153	0.0153

- Negatif

Ditahap ini penulis akan memberikan contoh perhitungan 1 kata negatif (bijak) di pembobotan laplace correction.

Diketahui :

Jumlah semua kata (negatif) : 25

Jumlah kata (negatif) : 40

Jumlah kata “bijak” (negatif) : 0

Rumus:

$P(\text{kata}|-) = \frac{(\text{Jumlah kata per-kata}+1)}{(\text{jumlah kata} + \text{jumlah semua kata})}$

$P(\text{bijak}|-) = \frac{(0+1)}{(40+25)}$

: 1 / 65

: 0.0153

adil	baik	balik	bantu	biar	bijak	bikin	butuh
0.0153	0.0307	0.0153	0.0615	0.0153	0.0153	0.0307	0.0153

cerdas	corona	daerah	dukung	gerak	gubernur	harap	hati
0.0153	0.0307	0.0307	0.0153	0.0153	0.0153	0.0153	0.0307

hikmah	individu	lai	langsung	lembaga	neng	orang	pandemi
0.0153	0.0153	0.0153	0.0307	0.0153	0.0153	0.0307	0.0307

presiden	proses	rata	sakit	salur	sasar	sosial	syukur
0.0307	0.0307	0.0153	0.0307	0.0461	0.0307	0.0307	0.0153

tengah	terap	tidakadil	tidakbantu	Tidakrata	tidaktepat	tunai	virus
0.0307	0.0153	0.0307	0.0307	0.0307	0.0307	0.0307	0.0307

## B. Proses Testing

### 1. Daftar Data Testing

Berikut adalah data testing yang sudah di labelkan.

a. Data testing Doc-1 mempunyai label seperti berikut :

label postaging = “opini”.

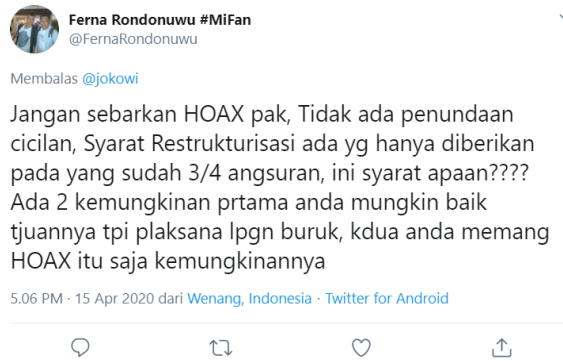
label naïve bayes = “positif”.



b. Data testing Doc-2 mempunyai label seperti berikut :

label postaging = “opini”.

label naïve bayes = “negatif”.



c. Data testing Doc-3 mempunyai label seperti berikut :

label postaging = “bukan opini”.

di data testing ini tidak ada label naïve bayes, karena data testing ini memiliki label postaging bukan opini.



## 2. Preprocessing

a. Cleaning

Doc-1	Doc-2
Yth Pak skrg lah saatnya mengeluarkan kartu PRA KERJA dan SEMBAKO MURAH spt dlm janji kampanye bpk turunkan harga karena hrg minyak lgi turun kebijakan tsb akan sangat	Jangan sebarkan HOAX pak Tidak ada penundaan cicilan Syarat Restrukturisasi ada yg hanya diberikan pada yang sudah angsuran ini syarat Ada kemungkinan prtama anda mungkin baik tjuannya tpi plaksana lpgn buruk kdua

membantu masyarakat dlm situasi spt ini cc	anda memang HOAX itu saja kemungkinannya
--	--

Doc-3
Presiden Jokowi Minta Program Kartu Pra Kerja Diprioritaskan bagi Warga Terkena PHK dlvr pic twitter

b. Casefolding

Doc-1	Doc-2
yth pak skrg lah saatnya mengeluarkan kartu pra kerja dan sembako murah spt dlm janji kampanye bpk turunkan harga karena hrg minyak lgi turun kebijakan tsb akan sangat membantu masyarakat dlm situasi spt ini cc	jangan sebar hoax pak tidak ada penundaan cicilan syarat restrukturisasi ada yg hanya diberikan pada yang sudah angsuran ini syarat ada kemungkinan prtama anda mungkin baik tjuannya tpi plaksana lpgn buruk kdua anda memang hoax itu saja kemungkinannya

Doc-3
presiden jokowi minta program kartu pra kerja diprioritaskan bagi warga terkena phk dlvr pic twitter

c. Tokenizing

Doc-1	Doc-2
yth, pak, skrg, lah, saatnya, mengeluarkan, kartu, pra,	jangan, sebar, hoax, pak, tidak, ada, penundaan, cicilan,



kerja, dan, sembako, murah, spt, dlm, janji, kampanye, bpk, turunkan, harga, karena, hrg, minyak, lgi, turun, kebijakan, tsb, akan, sangat, membantu, masyarakat, dlm, situasi, spt, ini, cc	syarat, restrukturisasi, ada, yg, hanya, diberikan, pada, yang, sudah, angsuran, ini, syarat, ada, kemungkinan, prtama, anda, mungkin, baik, tjuannya, tpi, plaksana, lpgn, buruk, kdua, anda, memang, hoax, itu, saja, kemungkinannya
--	--

Doc-3
presiden, jokowi, minta, program, kartu, pra, kerja, diprioritaskan, bagi, warga, terkena, phk, dlvr, pic, twitter

d. Normalisasi

Doc-1	Doc-2
yth, pak, sekarang, lah, saatnya, mengeluarkan, kartu, pra, kerja, dan, Sembilan, bahan, pokok, murah, seperti, dalam, janji, kampanye, bapak, turunkan, harga, karena, harga, minyak, lagi, turun, kebijakan, tersebut, akan, sangat, membantu, masyarakat, dalam, situasi, seperti, ini, cc	jangan, sebar, isu, bohong, pak, tidak, ada, penundaan, cicilan, syarat, restrukturisasi, ada, yang, hanya, diberikan, pada, yang, sudah, angsuran, ini, syarat, ada, kemungkinan, pertama, anda, mungkin, baik, tujuannya, tapi, pelaksana, lapan, buruk, kedua, anda, memang, isu, bohong, itu, saja, kemungkinannya

Doc-3
presiden, jokowi, minta, program, kartu, pra, kerja,

diprioritaskan, bagi, warga, terkena, phk, dlvr, pic, twitter,
---

e. Filtering

Doc-1	Doc-2
yth, mengeluarkan, kartu, pra, kerja, sembilan, bahan, pokok, murah, janji, kampanye, turunkan, harga, harga, minyak, turun, kebijakan, membantu, masyarakat, situasi, cc	jangan, sebarkan, HOAX, pak, Tidak, ada, penundaan, cicilan, syarat, restrukturisasi, ada, yg, sebarkan, isu, bohong, tidak, penundaan, cicilan, syarat, restrukturisasi, angsuran, syarat, pertama, baik, tujuannya, pelaksana, lapangan, buruk, kedua, isu, bohong

Doc-3
presiden, jokowi, program, kartu, pra, kerja, diprioritaskan, warga, terkena phk, dlvr, pic, twitter,

f. Stemming

Doc-1	Doc-2
keluar, kartu, kerja, sembilan, bahan, pokok, murah, janji, kampanye, turun, harga, harga, minyak, turun, bijak, bantu, masyarakat, situasi	sebar, isu, bohong, tidaktunda, cicil, syarat, restrukturisasi, angsur, syarat, baik, tuju, laksana, lapang, buruk, isu, bohong

Doc-3
presiden, program, kartu, kerja, prioritas, warga, kena

### 3. Opinion Detection (POSTagging dan Rule Opinion)

Tahap ini untuk menentukan tweet yang termasuk opini atau bukan. Rule yang digunakan untuk mengetahui apakah suatu tweet termasuk opini atau bukan ditunjukkan pada table berikut:

rule	contoh
RB JJ	sangat bagus
SC VB	jika memilih, semoga berjalan
NN JJ	kerja bagus, kerja bobrok
SC NNP	semoga sukses
RB VB	sangat memalukan
NEG JJ	tidak bagus, tidak transparan
JJ VB	bagus sekali
NEG VB	tidak mengerti, tidak memahami
RB JJ	sangat lambat, sangat bagus
MD VB	perlu dikembangkan, perlu diganti
PRP VB	saya suka
VB JJ	membuat bingung
JJ RB	bagus sekali
VB VB	membuat pusing

Gambar 4.5 Rule Opini.

Jika tweet mengandung rule berikut, maka tweet tersebut bisa di kategorikan sebagai opini, jika tweet tidak mengandung rule berikut maka tweet dikategorikan bukan opini, jika tweet dikategorikan bukan opini maka tweet tersebut tidak akan diproses di tahap kalsifikasi naïve bayes.

#### a. Tahap PosTagging Viterbi

Contoh perhitungan detail dari tahap postagging bisa dilihat pada bab 4.2.1, di bab ini penulis langsung menginputkan hasil dari POSTaging.

##### 1. Doc-1

keluar	kartu	Kerja	sembian	bahan	Pokok	murah	janji	kampanye
VB	NN	NN	CD	NN	NN	JJ	NN	NN

turun	harga	Harga	minyak	turun	bijak	bantu	masyarakat	situasi
VB	NN	NN	NN	VB	JJ	X	NN	NN

## 2. Doc-2

sebar	isu	bohong	tidak	tunda	cicil	syarat	restrukturisasi	angsur
X	NN	JJ	NEG	X	X	NN	NN	VB

syarat	baik	tuju	laksana	lapang	buruk	isu	bohong
NN	JJ	X	X	JJ	JJ	NN	VB

## 3. Doc-3

presiden	program	Kartu	Kerja	prioritas	warga	kena
NNP	NN	NN	NN	NN	NN	VB

## b. Tahap Rule Opini

### 1. Doc-1

Tweet	Tag	Termasuk Opini ?
keluar kartu	VB NN	Bukan
kartu kerja	NN NN	Bukan
kerja sembian	NN CD	Bukan
Sembian bahan	NN NN	Bukan
Bahan pokok	NN NN	Bukan
Pokok murah	NN JJ	Termasuk
Murah janji	JJ NN	Bukan
Janji kampanye	NN NN	Bukan
Kampanye turun	NN VB	Bukan
Turun harga	VB NN	Bukan
Harga harga	NN NN	Bukan

Harga minyak	NN NN	Bukan
Minyak turun	NN VB	Bukan
Turun bijak	VB JJ	Termasuk
Bijak bantu	JJ X	Bukan
Bantu masyarakat	X NN	Bukan
Masyarakat situasi	NN NN	Bukan

## 2. Doc-2

Tweet	Tag	Termasuk Opini ?
Sebar isu	X NN	Bukan
Isu bohong	NN JJ	Termasuk
Bohong tidak	JJ NEG	Bukan
Tidak tunda	NEG X	Bukan
Tunda cicil	X X	Bukan
Cicil syarat	X NN	Bukan
Syarat restrukturisasi	NN NN	Bukan
Restrukturisasi angsur	NN VB	Bukan
Angsur syarat	VB NN	Bukan
Syarat baik	NN JJ	Termasuk
Baik tuju	JJ X	Bukan
Tuju laksana	X X	Bukan
Laksana lapang	X JJ	Bukan
Lapang buruk	JJ JJ	Bukan
Buruk isu	JJ NN	Bukan
Isu bohong	NN VB	Bukan

## 3. Doc-3

Tweet	Tag	Termasuk Opini ?
Presiden program	NNP NN	Bukan
Program kartu	NN NN	Bukan
Kartu kerja	NN NN	Bukan

Kerja prioritas	NN NN	Bukan
Prioritas warga	NN NN	Bukan
Warga kena	NN VB	Bukan

Kesimpulan Opinion Detection :

1. Doc-1 adalah opini karena salah satu tag dari data tweet Doc-1 ada yang mengandung rule opinion, tag tersebut yaitu tag NN JJ dan VB JJ pada kalimat “pokok murah” dan “turun bijak”. Doc-1 akan di proses di naïve bayes karena Doc-1 termasuk opini.

2. Doc-2 adalah opini karena salah satu tag dari data tweet Doc-2 ada yang mengandung rule opini, tag tersebut yaitu tag NN JJ dan NN JJ pada kalimat “isu bohong” dan “syarat baik”. Doc-2 akan di proses di naïve bayes karena Doc-2 bukan opini.

3. Doc-3 adalah bukan opini karena salah satu tag dari data tweet Doc-3 tidak ada yang mengandung rule opini. Doc-3 tidak di proses di naïve bayes karena Doc-3 bukan opini.

#### 4. Klasifikasi Naïve Bayes

Fungsi Klasifikasi Naïve Bayes pada sistem adalah untuk mengkategorikan opini negatif dan opini positif pada suatu tweet yang berlabel opini, Di proses kalsifikasi, penulis akan mencotohkan perhitungan kalsifikasi naïve bayes.

Diketahui :

Bobot laplace correction sentiemen positif =

Default = 0.0153 (Default digunakan jika kata dari data testing tidak ada di data training atau data training berupa 0).

Adil	baik	balik	bantu	Biar	bijak	Bikin	Butuh
0.0307	0.0153	0.0307	0.0461	0.0307	0.0461	0.0153	0.0307

Cerdas	corona	Daerah	dukung	Gerak	gubernur	harap	Hati
0.0307	0.0307	0.0153	0.0307	0.0307	0.0307	0.0307	0.0153

hikmah	individu	Lai	langsung	lembaga	Neng	Orang	pandemi
0.0307	0.0307	0.0307	0.0307	0.0307	0.0307	0.0153	0.0153

presiden	proses	Rata	Sakit	Salur	sasar	sosial	syukur
0.0153	0.0153	0.0307	0.0153	0.0461	0.0153	0.0153	0.0307

Tengah	terap	Tidakadil	tidakbantu	tidakrata	tidaktepat	Tunai	virus
0.0153	0.0307	0.0153	0.0153	0.0153	0.0153	0.0153	0.0153

Bobot laplace correction sentiemen negatif =

Default = 0.0153 (Default digunakan jika kata dari data testing tidak ada di data training atau data training berupa 0).

Adil	baik	balik	bantu	Biar	bijak	Bikin	butuh
0.0153	0.0307	0.0153	0.0615	0.0153	0.0153	0.0307	0.0153

Cerdas	corona	daerah	dukung	Gerak	gubernur	harap	hati
0.0153	0.0307	0.0307	0.0153	0.0153	0.0153	0.0153	0.0307

hikmah	individu	lai	langsung	lembaga	neng	Orang	pandemi
0.0153	0.0153	0.0153	0.0307	0.0153	0.0153	0.0307	0.0307

presiden	proses	rata	Sakit	Salur	sasar	sosial	syukur
0.0307	0.0307	0.0153	0.0307	0.0461	0.0307	0.0307	0.0153

Tengah	terap	tidakadil	tidakbantu	tidakrata	tidaktepat	tunai	virus
0.0307	0.0153	0.0307	0.0307	0.0307	0.0307	0.0307	0.0307

Hasil Preprocessing data testing:

Doc-1	Doc-2
keluar, kartu, kerja, sembilan, bahan, pokok, murah, janji, kampanye, turun, harga, harga, minyak, turun, bijak, bantu, masyarakat, situasi	sebar, isu, bohong, tidaktunda, cicil, syarat, restrukturisasi, angsur, syarat, baik, tuju, laksana, lapang, buruk, isu, bohong

Rumus Klasifikasi Naïve Bayes :

$$P(C|d) = P(C) \prod_{i=1}^n P(W_i | C)$$

Doc-1
<p>Probabiliti(+) = P(keluar +)* P(kartu +)* P(kerja +)* P(sembilan +)* P(bahan +)* P(pokok +)* P(murah +)* P(janji +)* P(kampanye +)* P(turun +)* P(harga +)* P(harga +)* P(minyak +)* P(turun +)* P(bijak +)* P(bantu +)* P(masyarakat +)* P(situasi +)</p> <p>Probabiliti(+) = 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0461 * 0.0461 * 0.0153 * 0.0153 = 2.0979836042984E-32</p>
<p>Probabiliti(-) = P(keluar -)* P(kartu -)* P(kerja -)* P(sembilan -)* P(bahan -)* P(pokok -)* P(murah -)* P(janji -)* P(kampanye -)* P(turun -)* P(harga -)* P(harga -)* P(minyak -)* P(turun -)* P(bijak -)* P(bantu -)* P(masyarakat -)* P(situasi -)</p> <p>Probabiliti(+) = 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0153 * 0.0461 * 0.0615 * 0.0153 * 0.0153 = 9.3243715746591E-33</p>
<p>Kesimpulan : Doc-1 berupa positif karena probability di sentiment positif lebih besar dari pada probability di sentimen negatif.</p>





b. Analisis Kebutuhan Perangkat Keras

Tabel 4.3 Kebutuhan Perangkat Keras

No.	Perangkat	Keterangan
1.	<i>Processor</i>	<i>INTEL CORE I5</i>
2.	<i>Memory</i>	<i>8GB</i>
3.	<i>Hardisk</i>	<i>1TB</i>

#### 4.4 Analisis Kebutuhan Fungsional

Analisa kebutuhan fungsional berisikan proses-proses yang dilakukan oleh sistem. Dalam hal ini user sebagai pengguna dapat menggunakan layanan-layanan pada sistem, antara lain :

- Sistem dapat melakukan scraping tweet di aplikasi web twitter.
- Sistem dapat menyimpan data tweet yang diambil dari scraping ke database.
- Sistem dapat melakukan proses *preprocessing* pada dataset.
- Sistem dapat melakukan proses perhitungan Laplace Correction.
- Sistem dapat melakukan proses POSTagging Viterbi.
- Sistem dapat melakukan klasifikasi *tweet* ke dalam klasifikasi positif dan negative menggunakan metode Naïve Bayes.
- Sistem dapat menampilkan grafik dari *tweet* yang telah diklasifikasi.

#### 4.5 Perancangan Sistem

##### 4.5.1 Database

*Database* digunakan untuk menyimpan dataset *tweet* yang didapatkan dari Scraping web twitter dan untuk menyimpan semua hasil proses algoritma yang digunakan di aplikasi sistem. Ada 11 table yang digunakan untuk penyimpanan data pada aplikasi penulis, diantaranya yaitu : *tb\_datatraining*, *tb\_katadasar*, *tb\_laplacecorrection*, *tb\_naivebayes*, *tb\_normalisasi*, *tb\_postagging*, *tb\_preprocessing*, *tb\_preprocessingtraining*, *tb\_rule\_opini*, *tb\_scraper*, *tb\_stopword*.

Tabel 4.4 Atribut Tabel *tb\_datatraing*

No.	Atribut	Tipe Data
1.	ID	char(20)
2.	url	Text

3.	Datetime	Text
4.	Text	Text
5	user_id	char(20)
6	usernameTweet	Text
7	Sentiment	varchar(100)

Tabel 4.4 merupakan tabel yang digunakan untuk menyimpan data tweet hasil scraping, dan data di table ini dijadikan dataset data training.

Tabel 4.5 Atribut Tabel *tb\_katadasar*

No.	Atribut	Tipe Data
1.	Id	int(11)
2.	Kata	varchar(100)
3.	Kategori	varchar(100)

Tabel 4.5 merupakan tabel yang digunakan untuk menyimpan *datasets* kata dasar yang akan digunakan untuk proses preprocessing filtering.

Tabel 4.6 Atribut Tabel *tb\_laplacecorrection*

No.	Atribut	Tipe Data
1.	id_laplacecorrection	int(11)
2.	Kata	Longtext
3.	jumlah_kata	int(20)
4.	Bobot	varchar(20)
5.	Sentiment	varchar(30)

Tabel 4.6 merupakan tabel yang digunakan untuk menyimpan pembobotan kata setelah proses *laplace correction* telah selesai.

Tabel 4.7 Atribut Tabel *tb\_naivebayes*

No.	Atribut	Tipe Data
1.	id_naivebayes	int(11)
2.	id_tweet	char(20)
3.	detail_bobot_naive_bayes_positif	Longtext
4.	detail_bobot_naive_bayes_negatif	Longtext
5.	hasil_naive_bayes_sentimen_positif	Text
6.	hasil_naive_bayes_sentimen_negatif	Text
7.	label_naive_bayes_ekspektasi	Text

8.	status_naive_bayes	varchar(100)
----	--------------------	--------------

Tabel 4.7 merupakan tabel yang digunakan untuk menyimpan hasil dari kalsifikasi naïve bayes.

Tabel 4.8 Atribut Tabel *tb\_normalisasi*

No.	Atribut	Tipe Data
1.	id_normalisasi	int(11)
2.	kata_gaul	varchar(500)
3.	Normalisasi	Text

Tabel 4.8 merupakan tabel yang digunakan untuk menyimpan kata gau atau kata tidak baku beserta kata bakunya.

Tabel 4.9 Atribut Tabel *tb\_postagging*

No.	Atribut	Tipe Data
1.	id_postagging	int(11)
2.	id_tweet	char(20)
3.	stemming_postagging	Longtext
4.	Tag	Longtext
5.	rule_deteksi	Text
6.	label_postagging_ekspektasi	varchar(200)
7.	status_postagging	varchar(100)

Tabel 4.9 merupakan tabel yang digunakan untuk menyimpan tag dari tweet yang telah melalui proses postaging viterbi.

Tabel 4.10 Atribut Tabel *tb\_preprocessing*

No.	Atribut	Tipe Data
1.	id_preprocessing	int(11)
2.	id_tweet	char(20)
3.	Cleaning	Longtext
4.	Casefolding	Longtext
5.	Normalisasi	Longtext
6.	Filtering	Longtext
7.	Stemming	Longtext

Tabel 4.10 merupakan tabel yang digunakan untuk menyimpan hasil preprocessing data testing.

Tabel 4.11 Atribut Tabel *tb\_preprocessingtraining*

No.	Atribut	Tipe Data
1.	id_preprocessing	int(11)
2.	id_tweet	varchar(100)
3.	Cleaning	Longtext
4.	Casefolding	Longtext
5.	Normalisasi	Longtext
6.	Filtering	Longtext
7.	Stemming	Longtext

Tabel 4.11 merupakan tabel yang digunakan untuk menyimpan hasil preprocessing data training.

Tabel 4.12 Atribut Tabel *tb\_rule\_opini*

No.	Atribut	Tipe Data
1.	id_rule_opini	int(11)
2.	Rule	varchar(100)
3.	Contoh	varchar(100)

Tabel 4.12 merupakan tabel yang digunakan untuk menyimpan data rule opini.

Tabel 4.13 Atribut Tabel *tb\_scraper*

No.	Atribut	Tipe Data
1.	ID	char(20)
2.	url	Text
3.	Datetime	Text
4.	Text	Text
5.	user_id	char(20)
6.	usernameTweet	Text
7.	Status	varchar(100)
8.	Label_postagging_realita	varchar(100)
9.	Label_naive_bayes_realita	varchar(100)

Tabel 4.13 merupakan tabel yang digunakan untuk menyimpan data tweet hasil scraping, dan data di table ini dijadikan dataset data testing.

Tabel 4.14 Atribut Tabel *tb\_stopword*

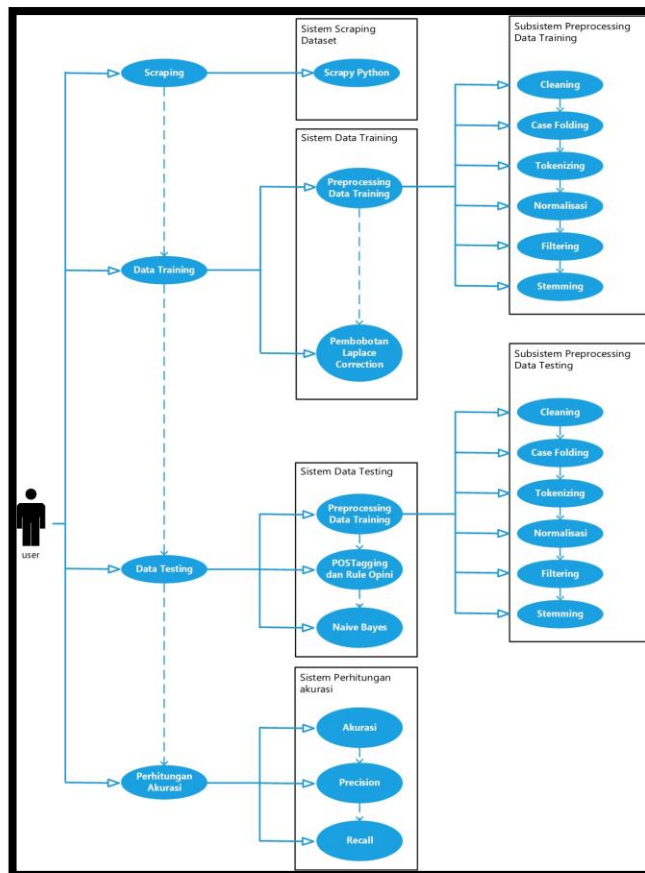
No.	Atribut	Tipe Data
1.	id_stopword	int(11)
2.	Kata	varchar(100)

Tabel 4.14 merupakan tabel yang digunakan untuk menyimpan data stopwords yang akan digunakan pada proses preprocessing filtering.

#### 4.5.2 Use Case

*Use Case* merupakan sebuah teknik yang digunakan dalam pengembangan sebuah software atau sistem informasi untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan, *Use Case* menjelaskan interaksi yang terjadi antara ‘aktor’ — inisiator dari interaksi sistem itu sendiri dengan sistem yang ada, sebuah *Use Case* direpresentasikan dengan urutan langkah yang sederhana.

*Use Case* Digunakan untuk mengidentifikasi siapa yang berinteraksi dengan sistem dan apa yang harus dilakukan sistem, berikut detail *Use Case* yang dirancang oleh penulis:



Gambar 4.6 Use Case.

a. Deskripsi Use Case

Tabel 4.15 Deskripsi Use Case

No.	Kategori Use Case	Use Case	Deskripsi
1.	<i>Scraping</i>	Scrapy Python	Metode pengambilan data tweet yang digunakan penulis adalah metode scraping menggunakan library scrapy python, metode scraping mengambil data dari html aplikasi tweeter.
2.	<i>Data Training</i>	Preprocessing Data Training	Tujuan Prerprocessing Data Training adalah untuk menghilangkan noise (kata yang tidak digunakan) pada suatu tweet, hasil dari proses preprocessing data training akan diproses di pembobotan kata laplace correction. Ada 6 metode preprocessing yang digunakan penulis untuk menghilangkan noise, diantaranya yaitu: cleaning, casefolding, tokenizing, normalisasi, filtering dan stemming.
		Pembobotan Kata Laplace Correction	Pembobotan kata Lapace Correction adalah pembobotan per kata dari kata yang berasal dari hasil preprocessing data training.
3	<i>Data Testing</i>	Preprocessing Data Testing	Tujuan Prerprocessing Data Testing adalah untuk menghilangkan noise (kata yang tidak digunakan) pada suatu tweet, hasil dari proses preprocessing data testing akan diproses di POSTagging, Rule Opini dan Naïve Bayes. Ada 6 metode preprocessing yang digunakan penulis untuk menghilangkan noise, diantaranya yaitu: cleaning, casefolding, tokenizing, normalisasi, filtering dan stemming.

		POSTagging, Rule Opini	Tujuan POSTagging adalah untuk memberi tag pada kata, dan tujuan dari rule opini adalah untuk mendeteksi apakah tweet dari data testing termasuk opini atau bukan opini.
		Naïve Bayes	Tujuan Naïve Bayes adalah untuk mendeteksi apakah tweet dari data testing termasuk opini positif atau opini negative.
4	Perhitungan Akurasi	Akurasi	Akurasi merupakan rasio prediksi Benar (positif dan negatif) dengan keseluruhan data.
		<i>Precision</i>	<i>Precision</i> Merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif.
		<i>Recall</i>	<i>Recall</i> merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif.

b. Skenario *Use Case*

1. Scraping

a. Scrapy Python.

Tabel 4.16 Skenario *Use Case* Scrapy Python

<b>Identifikasi</b>	
<b>Nomor</b>	1a
<b>Nama</b>	Scrapy Python
<b>Tujuan</b>	Untuk mendapatkan data tweet dari tweeter melalui HTML aplikasi tweeter dengan menggunakan library <i>python</i> .
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu dataset</b>	Melakukan proses scraping
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data tweet dari tweeter.



2. Data Training
  - a. Preprocessing Data Training.

Tabel 4.17 Skenario *Use Case* Preprocessing Data Training

<b>Identifikasi</b>	
<b>Nomor</b>	2a
<b>Nama</b>	Preprocessing Data Training
<b>Subsistem</b>	Cleaning, Casefolding, Tokenizing, Normalisasi, Filtering dan Steeming.
<b>Tujuan</b>	Tujuan Preprocessing Data Training adalah untuk menghilangkan noise (kata yang tidak digunakan) pada suatu tweet data training.
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu preprocessing</b>	Melakukan proses Preprocessing Data Testing
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data tweet yang sudah melalui process data training.

- b. Pembobotan Laplace Correction.

Tabel 4.18 Skenario *Use Case* Pembobotan laplace Correction

<b>Identifikasi</b>	
<b>Nomor</b>	2b
<b>Nama</b>	Pembobotan Laplace Correction
<b>Tujuan</b>	Tujuan Pembobotan laplace Correction adalah untuk membobotkan kata hasil dari pemrosesan preprocessing data testing.
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu preprocessing</b>	Melakukan proses Pembobotan Laplace Correction
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data pembobotan laplace correction.

3. Data Testing
  - a. Preprocessing Data Testing.

Tabel 4.19 Skenario *Use Case* Preprocessing Data Testing

<b>Identifikasi</b>	
<b>Nomor</b>	3a
<b>Nama</b>	Preprocessing Data Testing
<b>Subsistem</b>	Cleaning, Casefolding, Tokenizing, Normalisasi, Filtering dan Steeming.
<b>Tujuan</b>	Tujuan Prerprocessing Data Testing adalah untuk menghilangkan noise (kata yang tidak digunakan) pada suatu tweet data testing.
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu preprocessing</b>	Melakukan proses Preprocessing Data Testing
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data tweet yang sudah melalui process data testing.

b. POSTagging dan Rule Opini.

Tabel 4.20 Skenario *Use Case* POSTagging dan Rule Opini

<b>Identifikasi</b>	
<b>Nomor</b>	3b
<b>Nama</b>	POSTagging dan Rule Opini
<b>Tujuan</b>	Tujuan POSTagging adalah untuk memberi tag pada kata, dan tujuan dari rule opini adalah untuk mendeteksi apakah tweet dari data testing termasuk opini atau bukan opini.
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu preprocessing</b>	Melakukan proses POSTagging dan Rule Opini
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data tweet yang sudah melalui process data POSTagging dan Rule Opini.

c. Naïve Bayes.

Tabel 4.21 Skenario *Use Case* Naïve Bayes

<b>Identifikasi</b>	
<b>Nomor</b>	3d
<b>Nama</b>	Naïve Bayes
<b>Tujuan</b>	Tujuan Naïve Bayes adalah untuk mendeteksi apakah tweet dari data testing termasuk opini positif atau opini negative.

<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu preprocessing</b>	Melakukan proses Postagging dan Rule Opini
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data tweet yang sudah melalui process Naïve Bayes.

4. Hitung Akurasi

a. Akurasi.

Tabel 4.22 Skenario *Use Case* Akurasi

<b>Identifikasi</b>	
<b>Nomor</b>	4a
<b>Nama</b>	Akurasi
<b>Tujuan</b>	Tujuan akurasi adalah untuk memperlihatkan perbandingan rasio prediksi Benar (positif dan negatif) dengan keseluruhan data.
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu preprocessing</b>	Melakukan proses Akurasi
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data akurasi tweet.

b. Precision.

Tabel 4.23 Skenario *Use Case* Precision

<b>Identifikasi</b>	
<b>Nomor</b>	4b
<b>Nama</b>	Precision
<b>Tujuan</b>	Tujuan Precision adalah untuk memperlihatkan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif.
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu preprocessing</b>	Melakukan proses Precision
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data precision tweet.

c. Recall.

Tabel 4.24 Skenario *Use Case* Recall

<b>Identifikasi</b>	
<b>Nomor</b>	4c
<b>Nama</b>	Recall
<b>Tujuan</b>	Tujuan Precision adalah untuk memperlihatkan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif.
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>User</i> ditampilkan beberapa menu pilihan
<b>Aksi Aktor</b>	Reaksi Sistem
<b>Memilih menu preprocessing</b>	Melakukan proses Recall
<b>Kondisi Akhir</b>	<i>User</i> mendapatkan data recall tweet.

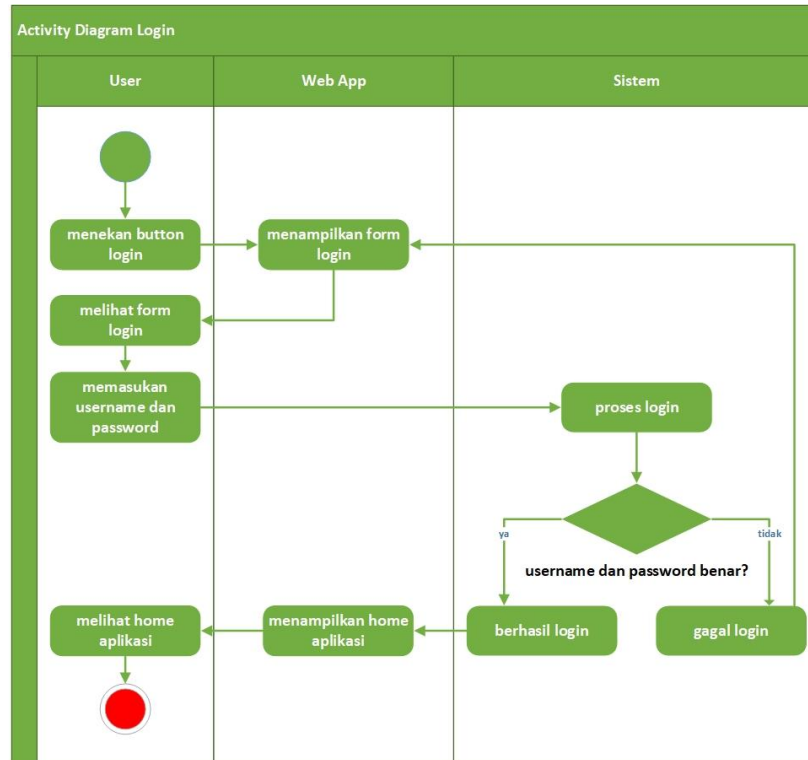
#### 4.5.3 Activity Diagram

*Activity Diagram* atau *Diagram* aktivitas adalah bentuk visual dari alur kerja yang berisi aktivitas dan tindakan, yang juga dapat berisi pilihan, atau pengulangan. Dalam Unified Modeling Language(UML), diagram aktivitas dibuat untuk menjelaskan aktivitas komputer maupun alur aktivitas dalam organisasi. Selain itu diagram aktivitas juga menggambarkan alur kontrol secara garis besar. Yang perlu diperhatikan yaitu diagram aktivitas bukan menggambarkan aktivitas sistem yang dilakukan aktor, tetapi menggambarkan aktivitas yang dapat dilakukan oleh sistem.

Tujuan utama dari Activity Diagram adalah menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses agar lebih mudah dipahami. Disini penulis menjelaskan aktivitas user dan aktivitas sistem secara beurutan dari aktivitas pertama yaitu login sampai aktivitas terakhir yaitu perhitungan akurasi, berikut detail *activity diagram* yang digunakan pada sistem:

a. *Activity Diagram Login*

*Activity Diagram Login* menggambarkan apa saja yang bisa dilakukan sistem disaat proses login.

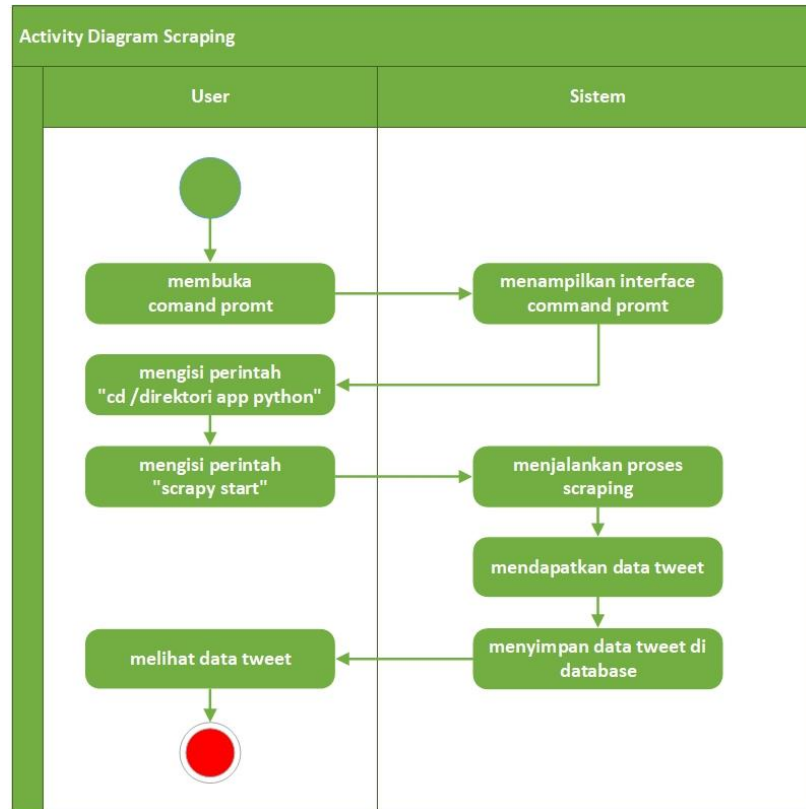


Gambar 4.7 Activity Diagram Login.

*Activity Diagram Login* dimulai dari user menekan button login, web app menampilkan form login, user melihat form login, user memasukan username dan password, sistem melakukan proses login, jika username dan password salah maka gagal login dan akan diarahkan ke web app menampilkan form login, jika berhasil maka web app menampilkan home aplikasi, user melihat home aplikasi.

b. *Activity Diagram Scraping*

*Activity Diagram Scraping* menggambarkan apa saja yang bisa dilakukan sistem disaat proses login.



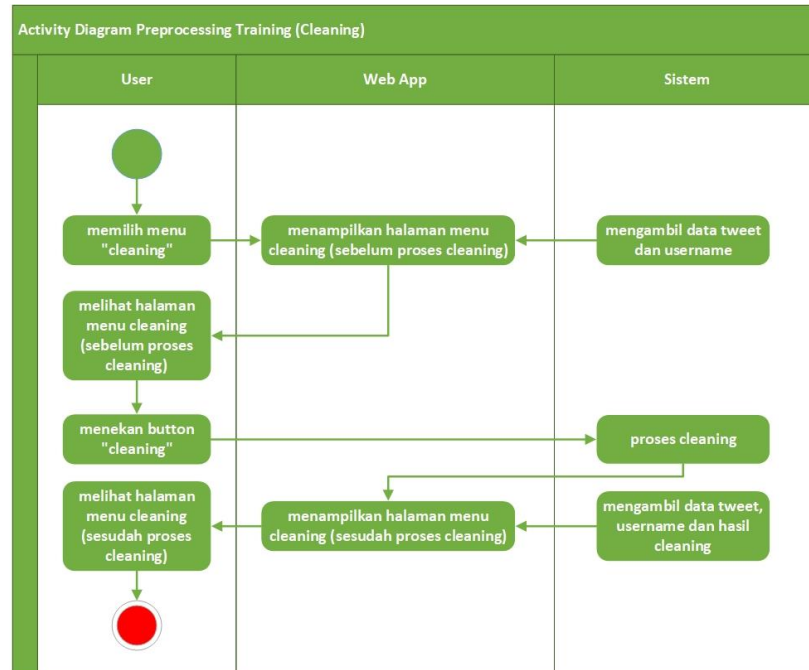
Gambar 4.8 Activity Diagram Scraping.

*Activity Diagram Scraping* dimulai dari user membuka comand promt, sistem menampilkan interface command promt, user mengisi perintah "cd /direktori app python", user mengisi perintah "scrapy start", sistem menjalankan proses scraping, sistem mendapatkan data tweet, sistem menyimpan data tweet di database, user melihat data tweet.

c. *Activity Diagram Preprocessing Training*

1. *Cleaning*

*Activity Diagram Preprocessing Training Cleaning* menggambarkan apa saja yang bisa dilakukan sistem disaat proses cleaning di data training.

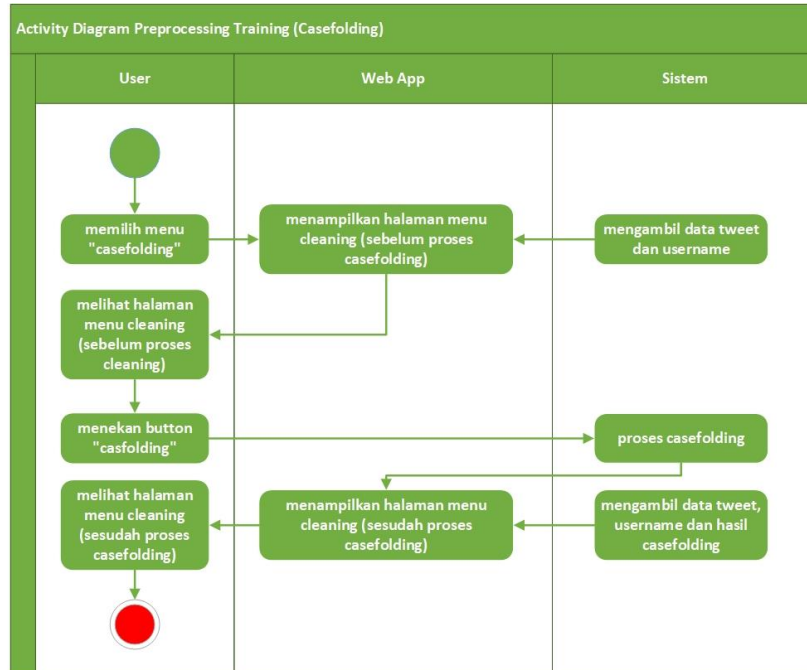


Gambar 4.9 Activity Preprocessing Cleaning Data Training.

*Activity Diagram Preprocessing Training Cleaning* dimulai dari user memilih menu “cleaning”, web app menampilkan halaman menu cleaning (sebelum proses cleaning) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu cleaning (sebelum proses cleaning), user menekan button “cleaning”, sistem memproses proses cleaning, web app menampilkan halaman menu cleaning (sesudah proses cleaning) bersamaan dengan sistem mengambil data tweet username dan hasil cleaning, user melihat melihat halaman menu cleaning (sesudah proses cleaning).

## 2. Casefolding

*Activity Diagram Preprocessing Training Casefolding* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *casefolding* di data training.



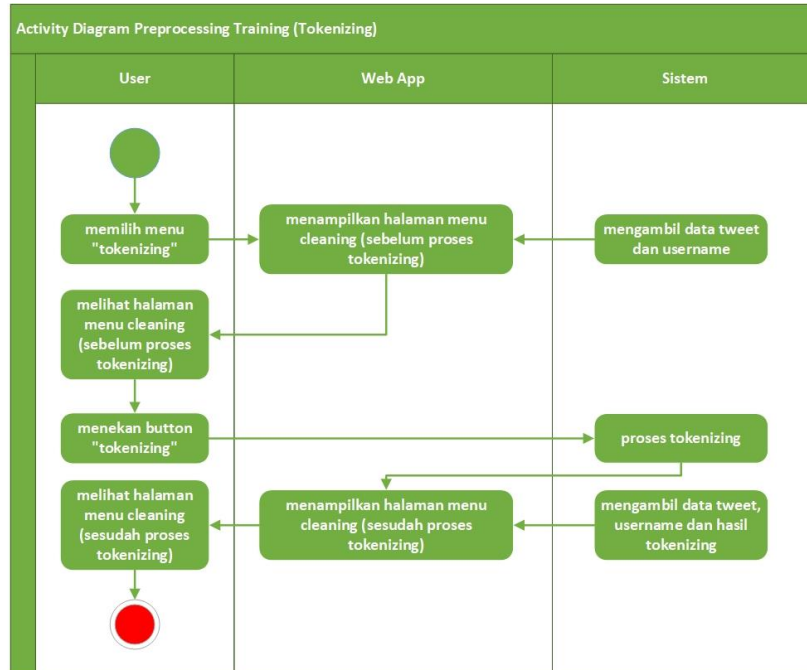
Gambar 4.10 Activity Preprocessing Casefolding Data Training.

*Activity Diagram Preprocessing Training Casefolding* dimulai dari user memilih menu “*casefolding*”, web app menampilkan halaman menu *casefolding* (sebelum proses *casefolding*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *casefolding* (sebelum proses *casefolding*), user menekan button “*casefolding*”, sistem memproses proses *casefolding*, web app menampilkan halaman menu *casefolding* (sesudah proses *casefolding*) bersamaan dengan sistem mengambil data tweet username dan hasil *casefolding*, user melihat melihat halaman menu cleaning (sesudah proses *casefolding*).



### 3. Tokenizing

*Activity Diagram Preprocessing Training Tokenizing* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *tokenizing* di data training.

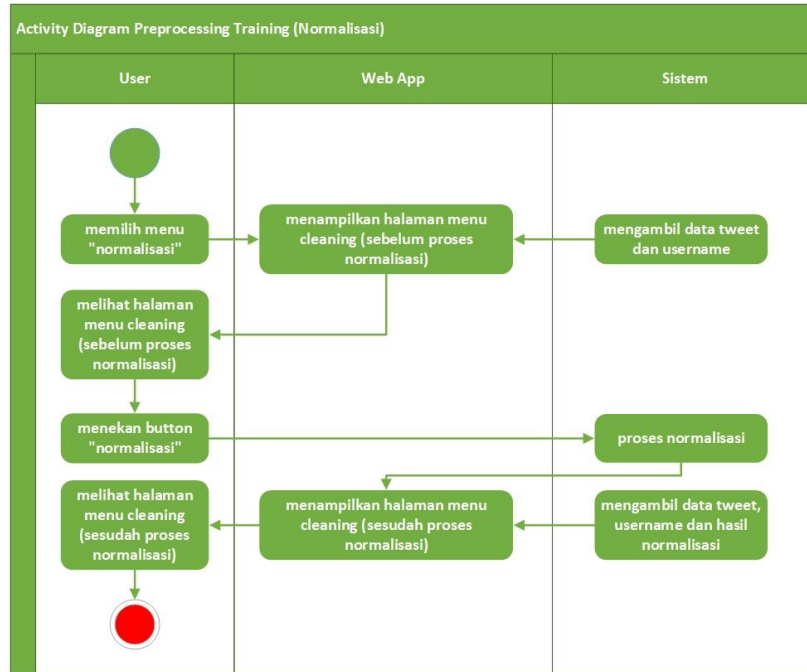


Gambar 4.11 Activity Preprocessing Tokenizing Data Training.

*Activity Diagram Preprocessing Training Tokenizing* dimulai dari user memilih menu "*tokenizing*", web app menampilkan halaman menu *tokenizing* (sebelum proses *tokenizing*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *tokenizing* (sebelum proses *tokenizing*), user menekan button "*tokenizing*", sistem memproses proses *tokenizing*, web app menampilkan halaman menu *tokenizing* (sesudah proses *tokenizing*) bersamaan dengan sistem mengambil data tweet username dan hasil *tokenizing*, user melihat melihat halaman menu cleaning (sesudah proses *tokenizing*).

#### 4. Normalisasi

*Activity Diagram Preprocessing Training Normalisasi* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *normalisasi* di data training.

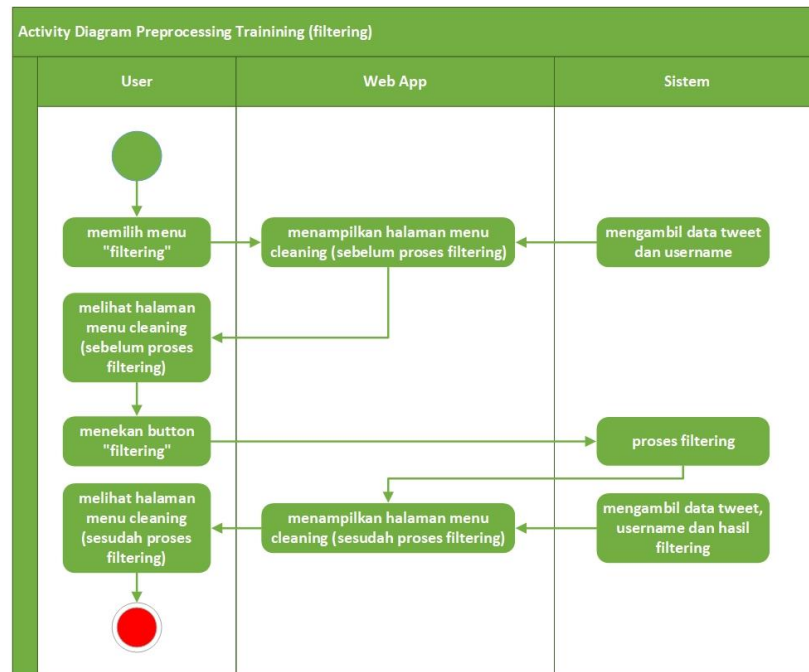


Gambar 4.12 Activity Preprocessing Normalisasi Data Training.

*Activity Diagram Preprocessing Training Normalisasi* dimulai dari user memilih menu “*normalisasi*”, web app menampilkan halaman menu *normalisasi* (sebelum proses *normalisasi*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *normalisasi* (sebelum proses *normalisasi*), user menekan button “*normalisasi*”, sistem memproses proses *normalisasi*, web app menampilkan halaman menu *normalisasi* (sesudah proses *normalisasi*) bersamaan dengan sistem mengambil data tweet username dan hasil *normalisasi*, user melihat melihat halaman menu cleaning (sesudah proses *normalisasi*).

## 5. Filtering

*Activity Diagram Preprocessing Training Filtering* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *filtering* di data training.

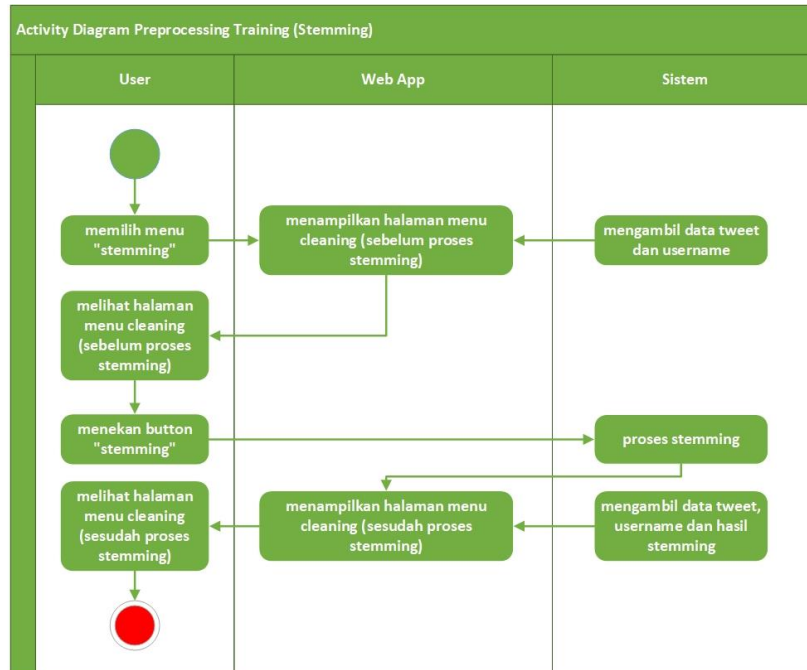


Gambar 4.13 Activity Preprocessing Filtering Data Training.

*Activity Diagram Preprocessing Training Filtering* dimulai dari user memilih menu "*filtering*", web app menampilkan halaman menu *filtering* (sebelum proses *filtering*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *filtering* (sebelum proses *filtering*), user menekan button "*filtering*", sistem memproses proses *filtering*, web app menampilkan halaman menu *filtering* (sesudah proses *filtering*) bersamaan dengan sistem mengambil data tweet username dan hasil *normalisasi*, user melihat melihat halaman menu cleaning (sesudah proses *filtering*).

## 6. Stemming

*Activity Diagram Preprocessing Training Stemming* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *stemming* di data training.

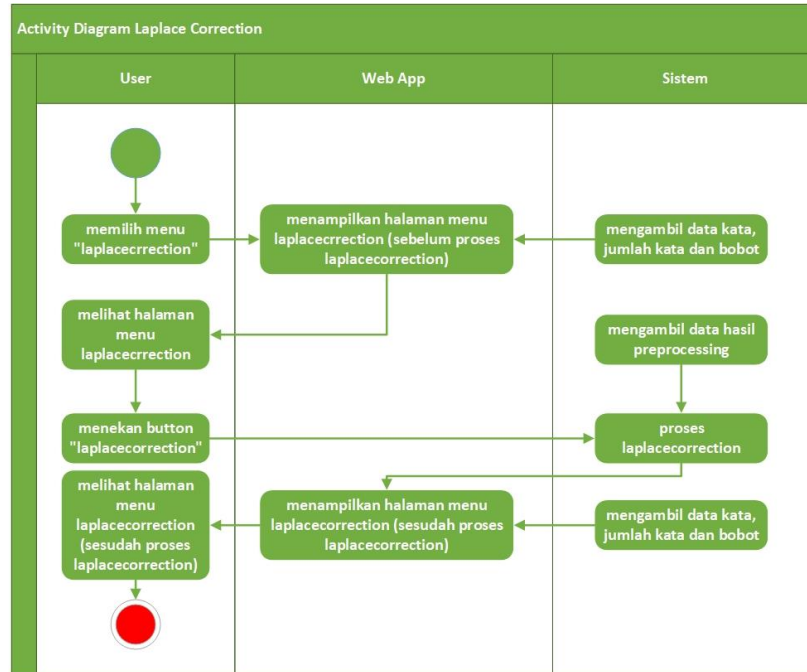


Gambar 4.14 Activity Preprocessing Stemming Data Training.

*Activity Diagram Preprocessing Training Stemming* dimulai dari user memilih menu "*stemming*", web app menampilkan halaman menu *stemming* (sebelum proses *stemming*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *stemming* (sebelum proses *stemming*), user menekan button "*stemming*", sistem memproses proses *stemming*, web app menampilkan halaman menu *stemming* (sesudah proses *stemming*) bersamaan dengan sistem mengambil data tweet username dan hasil *stemming*, user melihat melihat halaman menu cleaning (sesudah proses *stemming*).

d. *Activity Diagram Laplace Correction*

*Activity Diagram Laplace Correction* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *Laplace Correction* di data training.



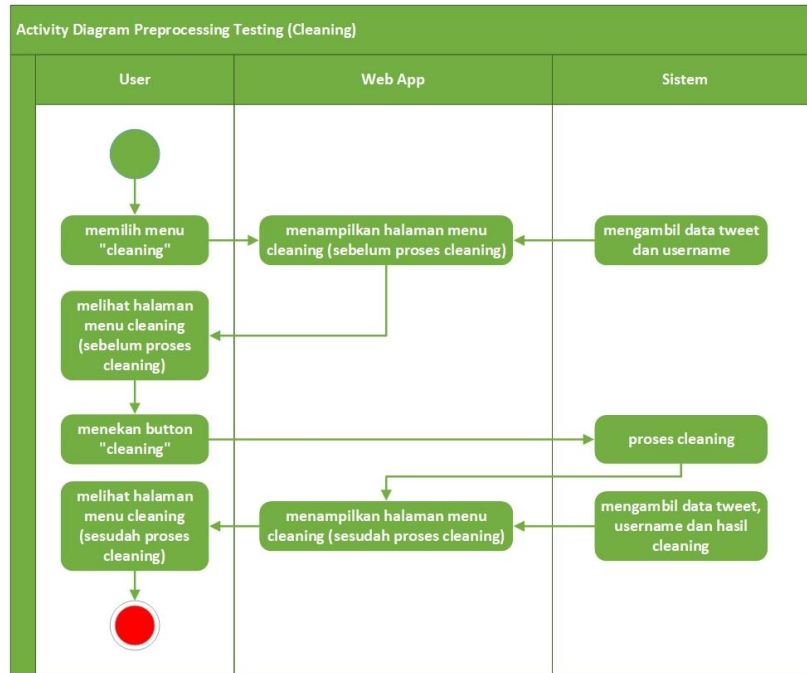
Gambar 4.15 Activity Laplace Correction.

*Activity Diagram Laplace Correction* dimulai dari user memilih menu “*laplace correction*”, web app menampilkan halaman menu *laplace correction* (sebelum proses *laplace correction*) bersamaan dengan sistem mengambil data kata, jumlahkata dan bobot , user melihat halaman menu *stemming* (sebelum proses *stemming*), user menekan button “*stemming*”, sistem memproses proses *stemming*, web app menampilkan halaman menu *stemming* (sesudah proses *stemming*) bersamaan dengan sistem mengambil data tweet username dan hasil *stemming*, user melihat melihat halaman menu cleaning (sesudah proses *stemming*).

e. *Activity Diagram Preprocessing Testing*

1. *Cleaning*

*Activity Diagram Preprocessing Testing Cleaning* menggambarkan apa saja yang bisa dilakukan sistem disaat proses cleaning di data testing.

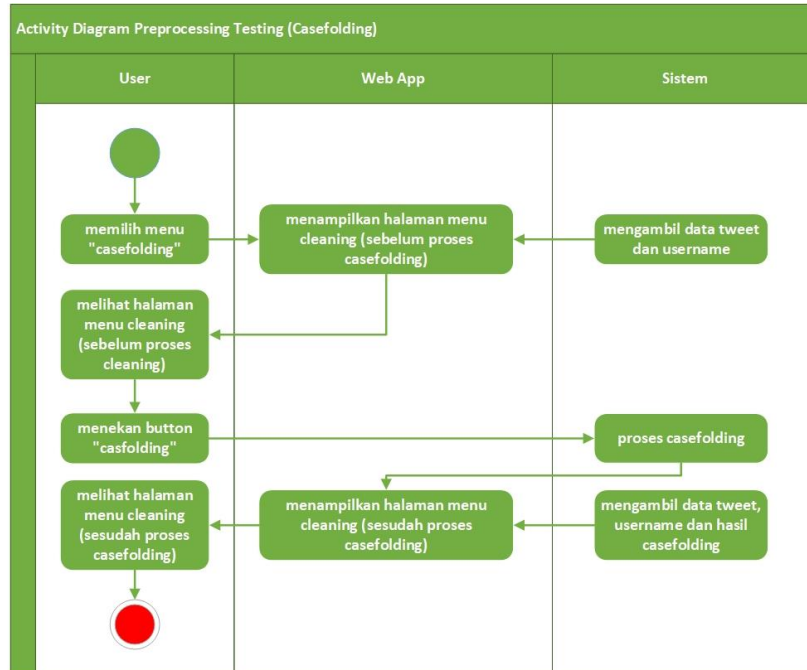


Gambar 4.16 Activity Preprocessing Cleaning Data Testing.

*Activity Diagram Preprocessing Testing Cleaning* dimulai dari user memilih menu “cleaning”, web app menampilkan halaman menu cleaning (sebelum proses cleaning) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu cleaning (sebelum proses cleaning), user menekan button “cleaning”, sistem memproses proses cleaning, web app menampilkan halaman menu cleaning (sesudah proses cleaning) bersamaan dengan sistem mengambil data tweet username dan hasil cleaning, user melihat melihat halaman menu cleaning (sesudah proses cleaning).

## 2. Casefolding

*Activity Diagram Preprocessing Testing Casefolding* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *casefolding* di data testing.

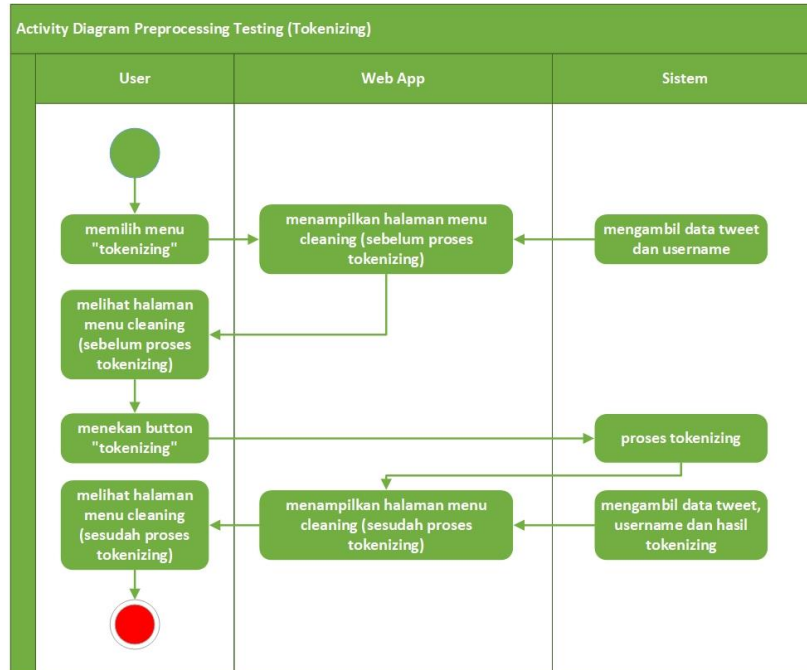


Gambar 4.17 Activity Preprocessing Casefolding Data Testing.

*Activity Diagram Preprocessing Testing Casefolding* dimulai dari user memilih menu "casefolding", web app menampilkan halaman menu *casefolding* (sebelum proses *casefolding*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *casefolding* (sebelum proses *casefolding*), user menekan button "casefolding", sistem memproses proses *casefolding*, web app menampilkan halaman menu *casefolding* (sesudah proses *casefolding*) bersamaan dengan sistem mengambil data tweet username dan hasil *casefolding*, user melihat melihat halaman menu cleaning (sesudah proses *casefolding*).

### 3. Tokenizing

*Activity Diagram Preprocessing Testing Tokenizing* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *tokenizing* di data testing.



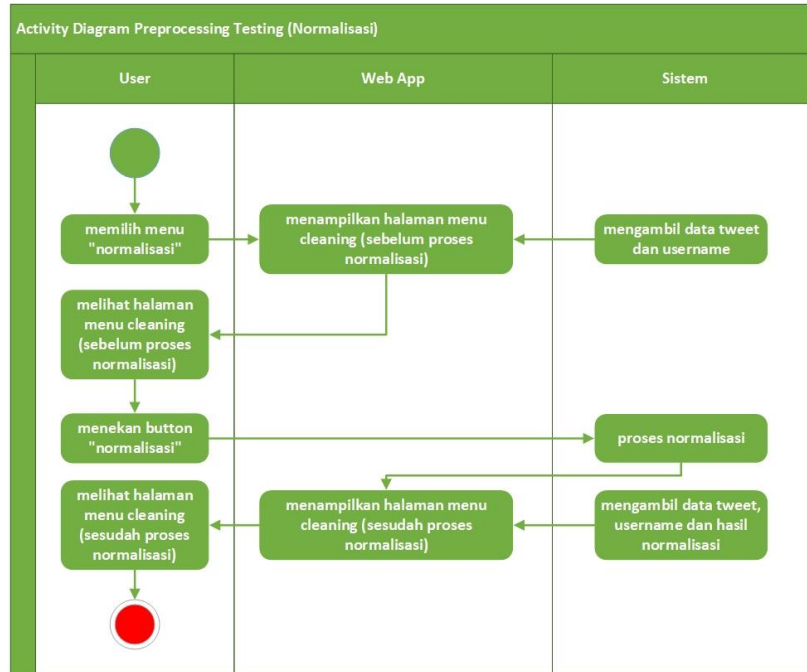
Gambar 4.18 Activity Preprocessing Tokenizing Data Testing.

*Activity Diagram Preprocessing Testing Tokenizing* dimulai dari user memilih menu "*tokenizing*", web app menampilkan halaman menu *tokenizing* (sebelum proses *tokenizing*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *tokenizing* (sebelum proses *tokenizing*), user menekan button "*tokenizing*", sistem memproses proses *tokenizing*, web app menampilkan halaman menu *tokenizing* (sesudah proses *tokenizing*) bersamaan dengan sistem mengambil data tweet username dan hasil *tokenizing*, user melihat melihat halaman menu cleaning (sesudah proses *tokenizing*).



#### 4. Normalisasi

*Activity Diagram Preprocessing Testing Normalisasi* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *normalisasi* di data testing.

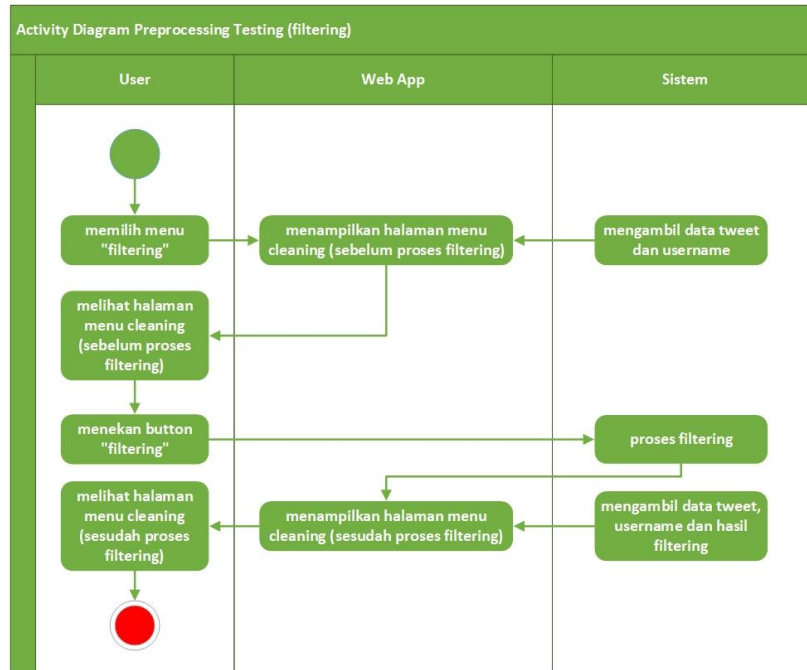


Gambar 4.19 Activity Preprocessing Normalisasi Data Testing.

*Activity Diagram Preprocessing Testing Normalisasi* dimulai dari user memilih menu “*normalisasi*”, web app menampilkan halaman menu *normalisasi* (sebelum proses *normalisasi*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *normalisasi* (sebelum proses *normalisasi*), user menekan button “*normalisasi*”, sistem memproses proses *normalisasi*, web app menampilkan halaman menu *normalisasi* (sesudah proses *normalisasi*) bersamaan dengan sistem mengambil data tweet username dan hasil *normalisasi*, user melihat melihat halaman menu cleaning (sesudah proses *normalisasi*).

## 5. Filtering

*Activity Diagram Preprocessing Testing Filtering* menggambarkan apasaja yang bisa dilakukan sistem disaat proses *filtering* di data testing.

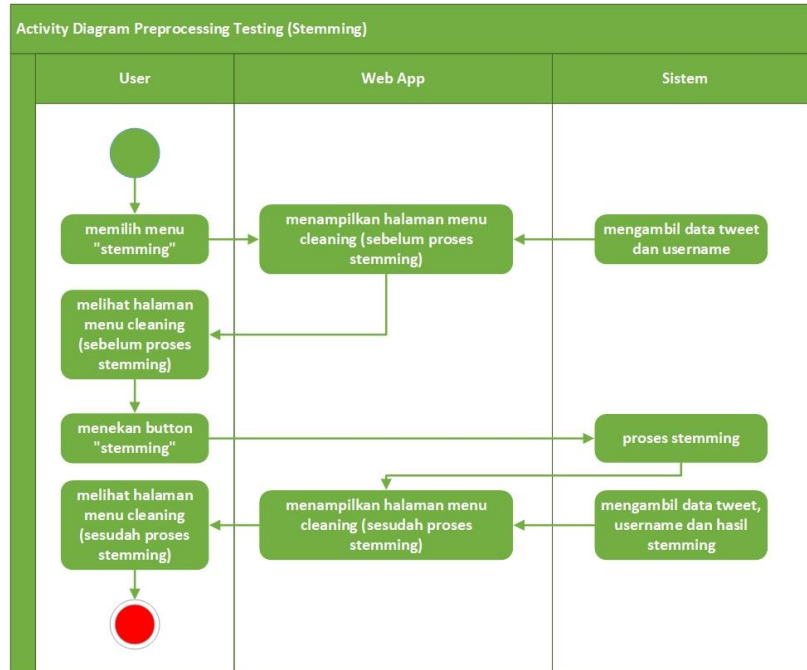


Gambar 4.20 Activity Preprocessing Filtering Data Testing.

*Activity Diagram Preprocessing Testing Filtering* dimulai dari user memilih menu "*filtering*", web app menampilkan halaman menu *filtering* (sebelum proses *filtering*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *filtering* (sebelum proses *filtering*), user menekan button "*filtering*", sistem memproses proses *filtering*, web app menampilkan halaman menu *filtering* (sesudah proses *filtering*) bersamaan dengan sistem mengambil data tweet username dan hasil *normalisasi*, user melihat melihat halaman menu cleaning (sesudah proses *filtering*).

## 6. Stemming

*Activity Diagram Preprocessing Testing Stemming* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *stemming* di data testing.

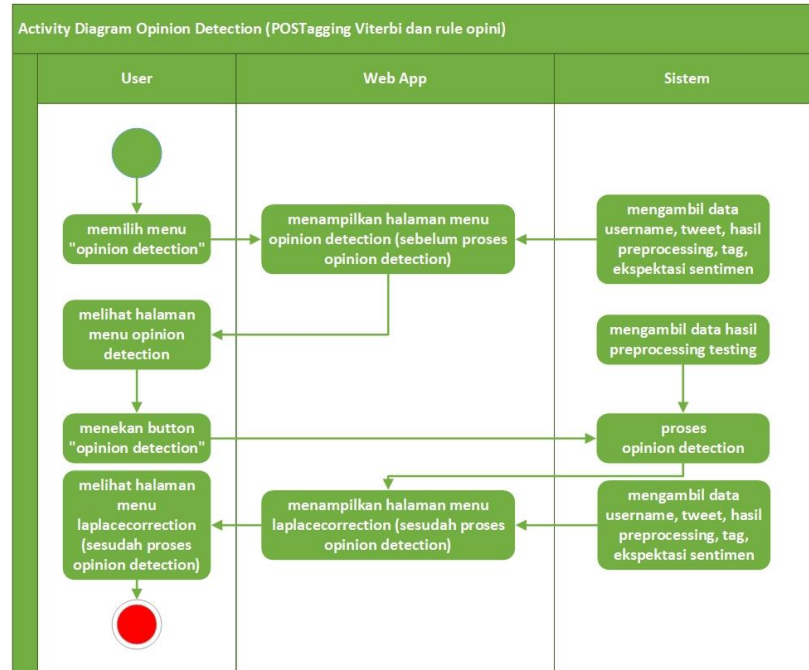


Gambar 4.21 Activity Preprocessing Stemming Data Testing.

*Activity Diagram Preprocessing Testing Stemming* dimulai dari user memilih menu "*stemming*", web app menampilkan halaman menu *stemming* (sebelum proses *stemming*) bersamaan dengan sistem mengambil data tweet dan username, user melihat halaman menu *stemming* (sebelum proses *stemming*), user menekan button "*stemming*", sistem memproses proses *stemming*, web app menampilkan halaman menu *stemming* (sesudah proses *stemming*) bersamaan dengan sistem mengambil data tweet username dan hasil *stemming*, user melihat melihat halaman menu cleaning (sesudah proses *stemming*).

f. *Activity Diagram Opinion Detection (POSTagging dan Rule Opini)*

*Activity Diagram Opinion Detection* menggambarkan apasaja yang bisa dilakukan sistem disaat proses *Opinion Detection* di data testing.

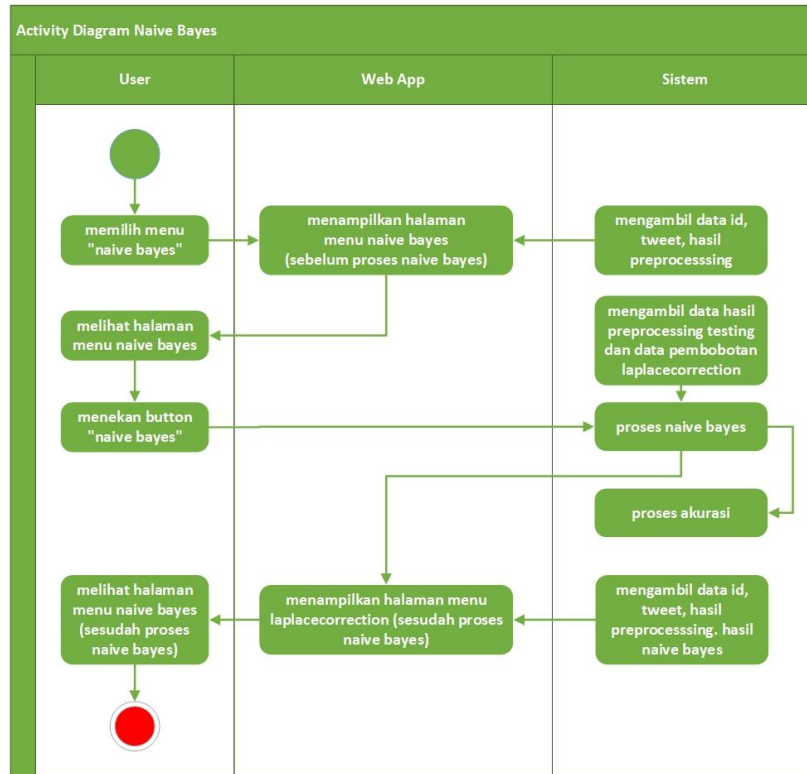


Gambar 4.22 Activity Opinion Detection.

*Activity Diagram Opinion Detection* dimulai dari user memilih menu “*opinion detection*”, web app menampilkan halaman menu opinion detection (sebelum proses opini detection) bersamaan dengan mengambil data username, tweet, hasil preprocessing, tag, ekspektasi sentiment, user melihat halaman menu opinion detection, user menekan button “*opinin detection*”, sistem proses opinion detection, web app menampilkan halaman menu laplacecorection (sesudah proses opinion detection) bersamaan dengan sistem menagmbil data username, tweet, hasil preprocessing tag, ekspektasi sentimen, user melihat halaman menu laplacecorection (sesudah proses opinion detection).

g. *Activity Diagram Naïve Bayes*

*Activity Diagram Naïve Bayes* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *Naïve Bayes* di data testing.

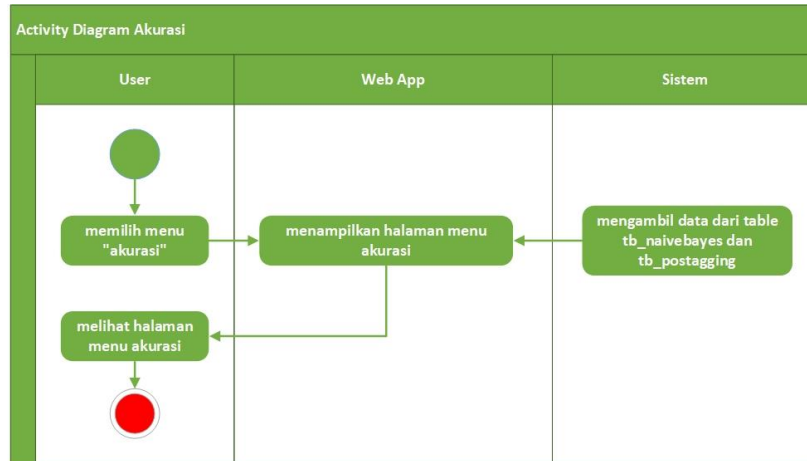


Gambar 4.23 Activity Naïve Bayes.

*Activity Diagram Naïve Bayes* dimulai dari user memilih menu “*naïve bayes*”, web app menampilkan halaman menu naïve bayes (sebelum proses naïve bayes) bersamaan dengan sistem mengambil data id, tweet, hasil preprocessing, user melihat halaman menu naïve bayes, user menekan button “*naïve bayes*”, sistem proses naïve bayes bersamaan dengan sistem mengambil data hasil preprocessing dan testing dan data pembobotan laplacecorrection dan sistem melakukan proses akurasi, web app menampilkan halaman menu lapalcecorrrection (sesudah proses naïve bayes) bersamaan dengan sistem mengambil data id tweet, hasil preprocessing dan hasil naïve bayes, user melihat halaman menu naïve bayes (sesudah proses naïve bayes).

h. *Activity Diagram Akurasi*

*Activity Diagram Akurasi* menggambarkan apa saja yang bisa dilakukan sistem disaat proses *akurasi* di data testing.



Gambar 4.24 Activity Akurasi.

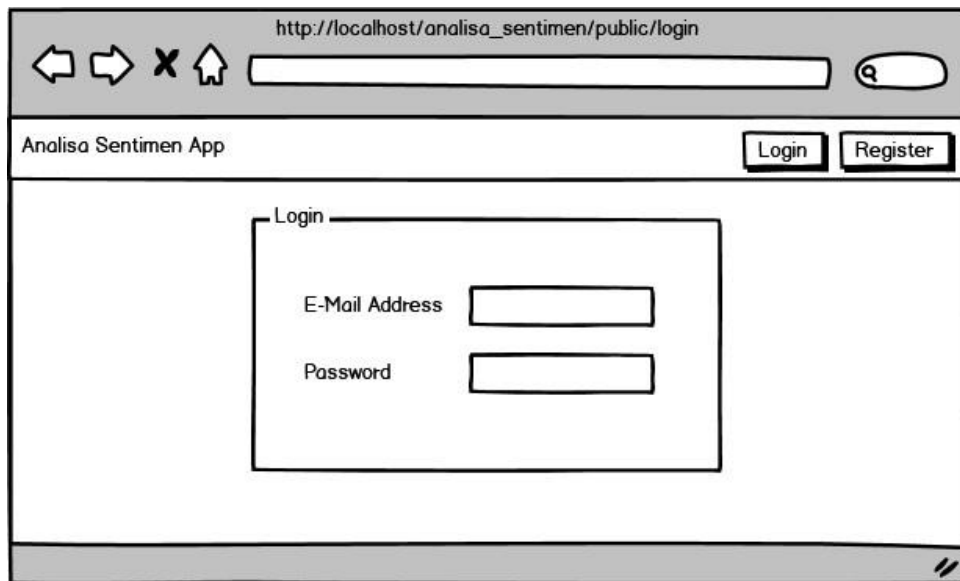
*Activity Diagram Akurasi* dimulai dari user memilih menu “*akurasi*”, web app menampilkan halaman menu akurasi bersamaan dengan sistem mengambil data dari table *tb\_naivebayes* dan *tb\_postagging*, user melihat halaman menu akurasi.

#### 4.5.4 Perancangan Interface Sistem

Perancangan *interface* yang akan dibuat pada sistem ini ada 18, yaitu, *Form Login*, *Menu Dashboard*, *Menu Data Training*, *Menu Preprocessing Cleaning Data Training*, *Menu Preprocessing Casefolding Data Training*, *Menu Preprocessing Tokenizing Data Training*, *Menu Preprocessing Normalisasi Data Training*, *Menu Preprocessing Filtering Data Training*, *Menu Preprocessing Stemming Data Training*, *Menu Laplace Correction*, *Menu Data testing*, *Menu Preprocessing Cleaning Data Testing*, *Menu Preprocessing Casefolding Data Testing*, *Menu Preprocessing Tokenizing Data Testing*, *Menu Preprocessing Normalisasi Data Testing*, *Menu Preprocessing Filtering Data Testing*, *Menu Preprocessing Stemming Data Testing*, *Menu Opinion Detection*, *Menu Naïve Bayes*, *Menu Akurasi*.

##### 3.5.4.1 Perancangan Form Login

Pada Form Login akan menampilkan Form Login yang berfungsi untuk masuk ke dalam sistem aplikasi.

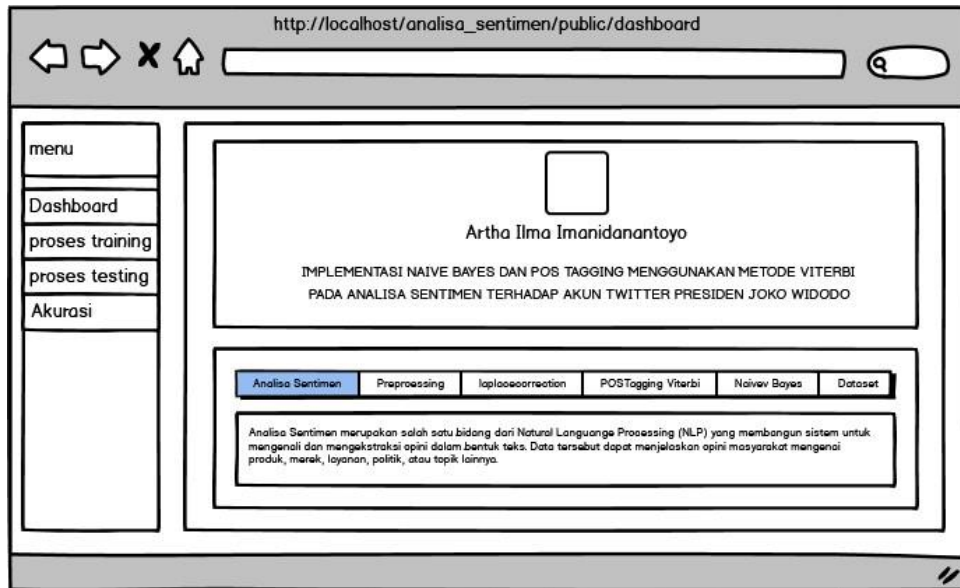


The image shows a web browser window with the address bar containing 'http://localhost/analisa\_sentimen/public/login'. The page title is 'Analisa Sentimen App'. In the top right corner, there are two buttons: 'Login' and 'Register'. The main content area features a 'Login' form with two input fields: 'E-Mail Address' and 'Password'. The browser's navigation icons (back, forward, stop, home) and a search bar are visible in the top left.

Gambar 4.25 Perancangan Form Login.

##### 3.5.4.2 Perancangan Menu Dashboard

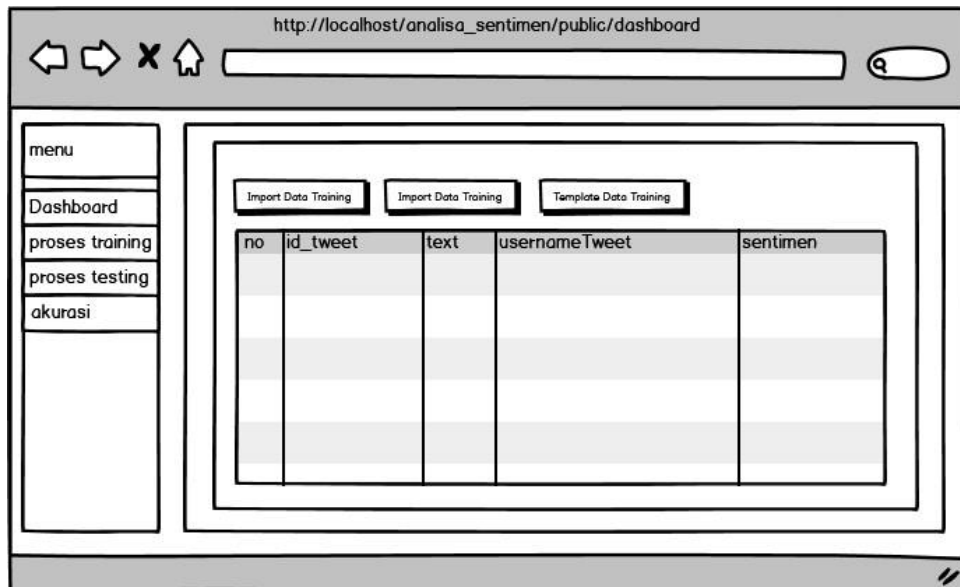
Pada Menu Dashboard akan menampilkan informasi algoritma apa saja yang berada di aplikasi penulis dan contoh algoritma tersebut beserta referensi algoritma.



Gambar 4.26 Perancangan Menu Dashboard.

### 3.5.4.3 Perancangan Menu Data Training

Pada Menu Data Training akan menampilkan Data Training yang akan diproses pada algoritma laplace correction yang hasilnya berupa bobot kata.

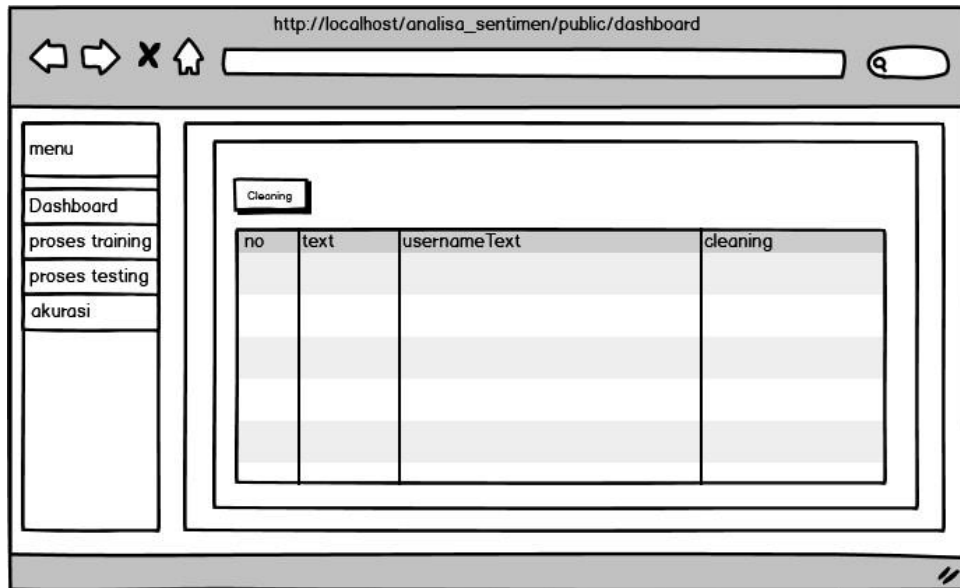


Gambar 4.27 Perancangan Menu Data Training.

### 3.5.4.4 Perancangan Menu Preprocessing Cleaning Data Training

Pada Menu Preprocessing Cleaning Data Training akan menampilkan Data Training yang sudah melalui tahap Preprocessing Cleaning.

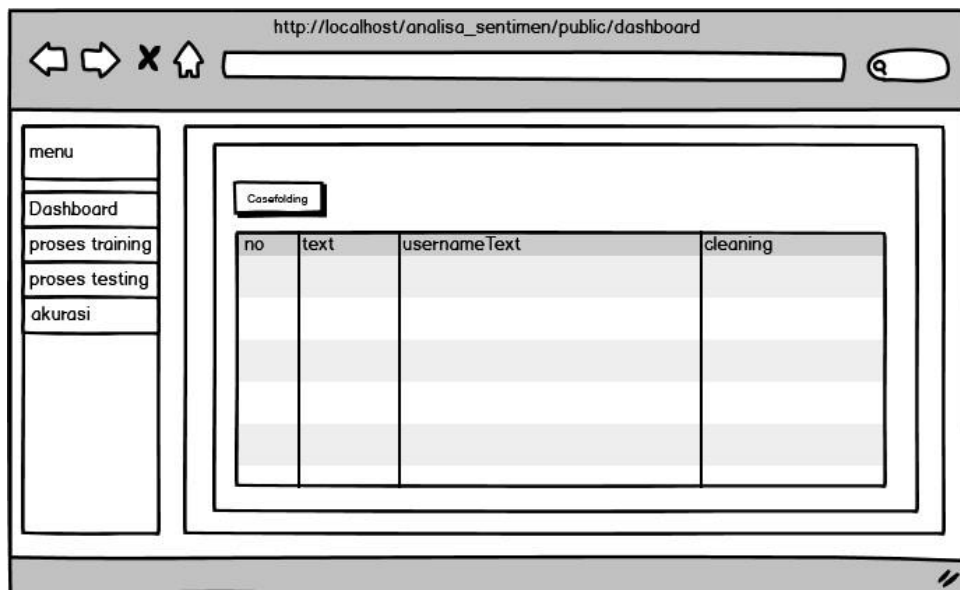




Gambar 4.28 Perancangan Menu Preprocessing Cleaning Data Training.

#### 3.5.4.5 Perancangan Menu Preprocessing Casefolding Data Training

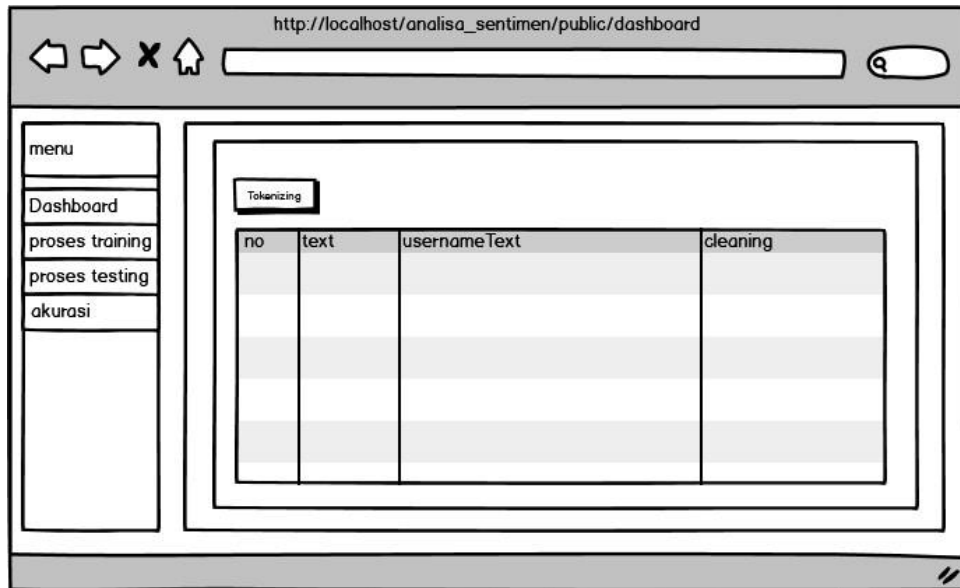
Pada Menu Preprocessing Casefolding Data Training akan menampilkan Data Training yang sudah melalui tahap Preprocessing Casefolding.



Gambar 4.29 Perancangan Menu Preprocessing Casefolding Data Training.

#### 3.5.4.6 Perancangan Menu Preprocessing Tokenizing Data Training

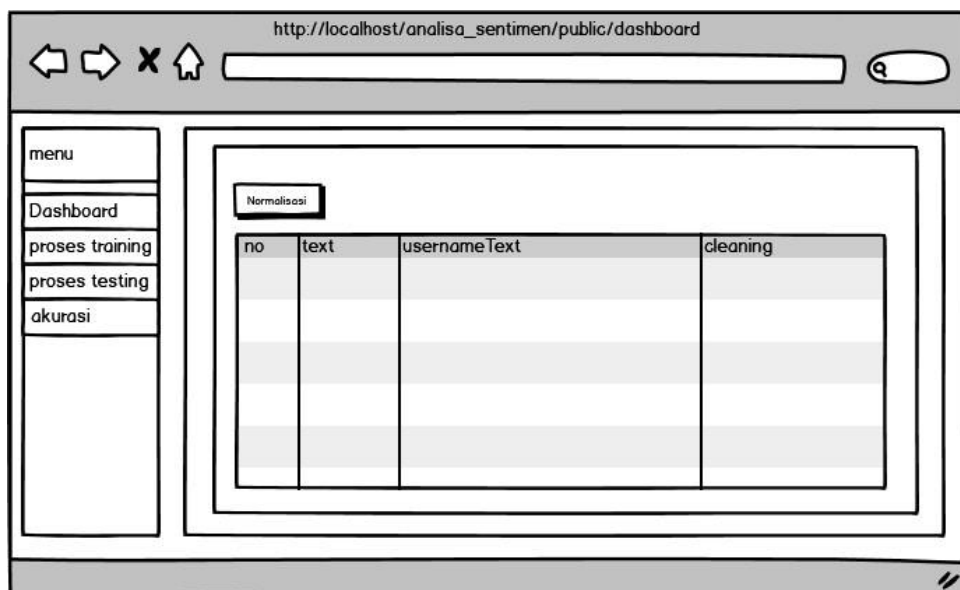
Pada Menu Preprocessing Tokenizing Data Training akan menampilkan Data Training yang sudah melalui tahap Preprocessing Tokenizing.



Gambar 4.30 Perancangan Menu Preprocessing Casefolding Data Training.

#### 3.5.4.7 Perancangan Menu Preprocessing Normalisasi Data Training

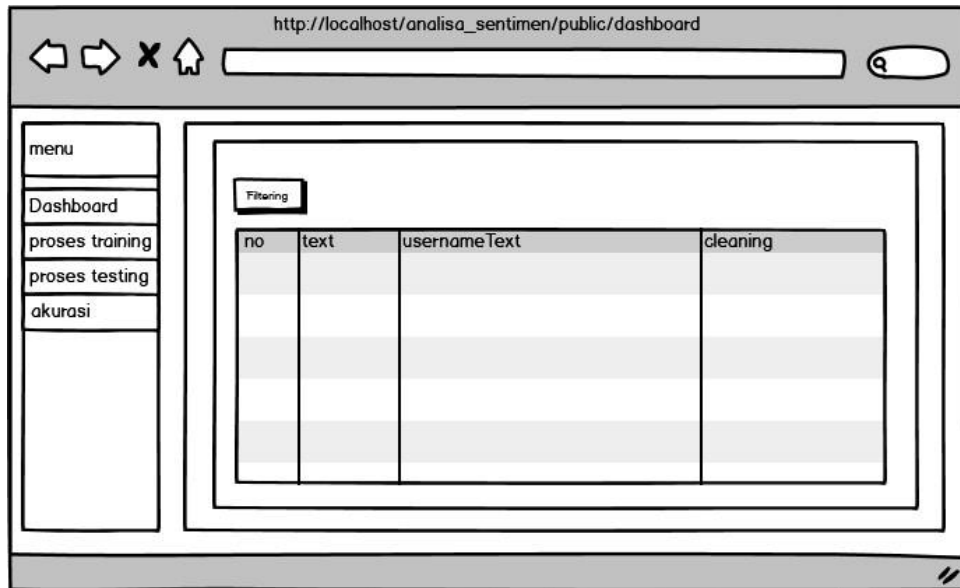
Pada Menu Preprocessing Normalisasi Data Training akan menampilkan Data Training yang sudah melalui tahap Preprocessing Normalisasi.



Gambar 4.31 Perancangan Menu Preprocessing Normalisasi Data Training.

#### 3.5.4.8 Perancangan Menu Preprocessing Filtering Data Training

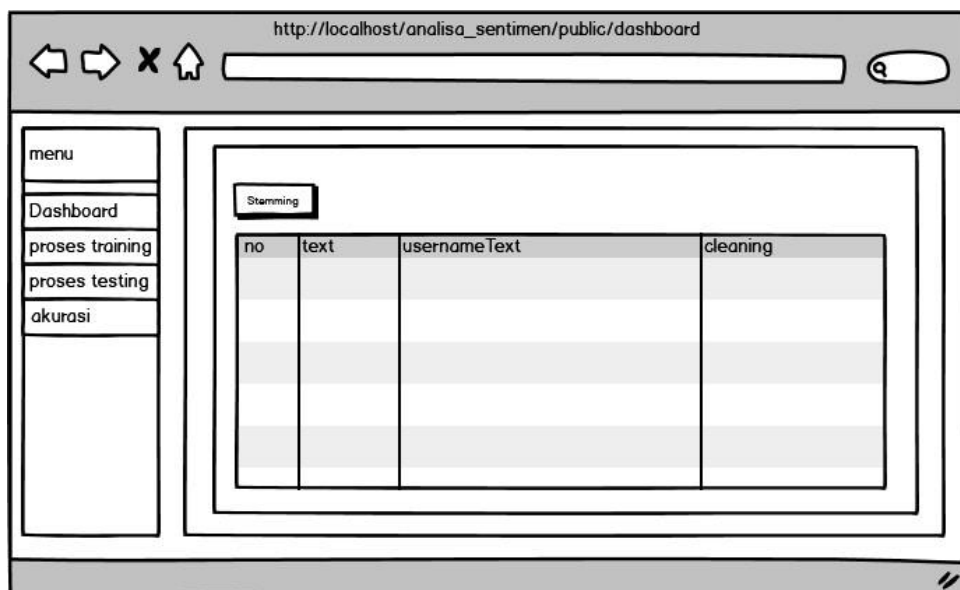
Pada Menu Preprocessing Filtering Data Training akan menampilkan Data Training yang sudah melalui tahap Preprocessing Filtering.



Gambar 4.32 Perancangan Menu Preprocessing Filtering Data Training.

#### 3.5.4.9 Perancangan Menu Preprocessing Stemming Data Training

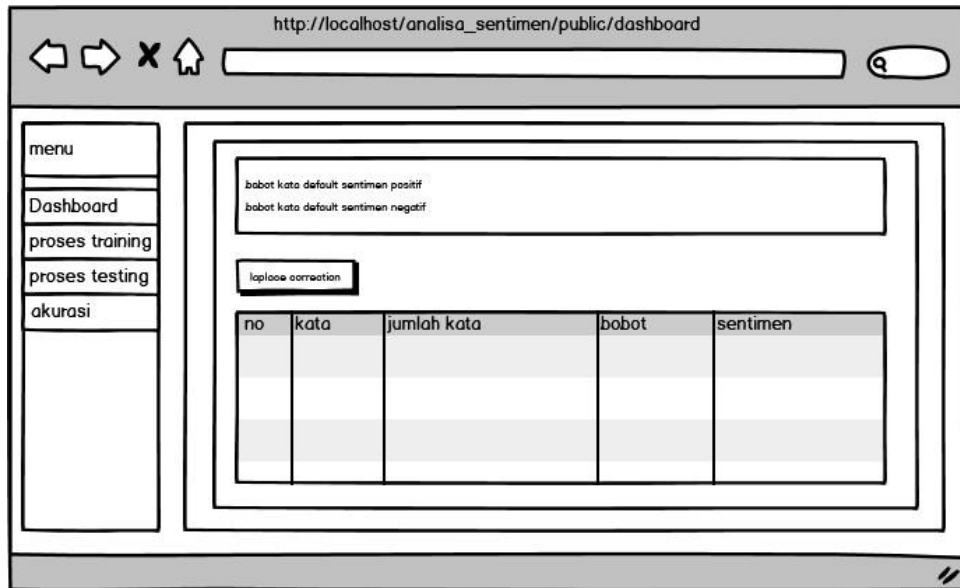
Pada Menu Preprocessing Stemming Data Training akan menampilkan Data Training yang sudah melalui tahap Preprocessing Stemming.



Gambar 4.33 Perancangan Menu Preprocessing Stemming Data Training.

#### 3.5.4.10 Perancangan Menu Laplace Correction

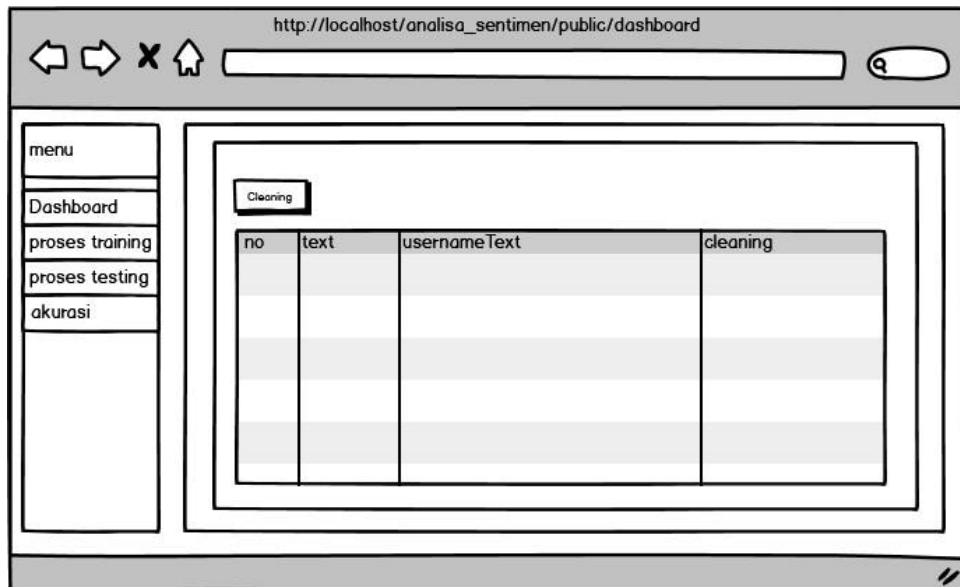
Pada Menu Laplace Correction akan menampilkan hasil data pembobotan laplace correction per kata.



Gambar 4.34 Perancangan Menu Laplace Correction.

#### 3.5.4.11 Perancangan Menu Preprocessing Cleaning Data Testing

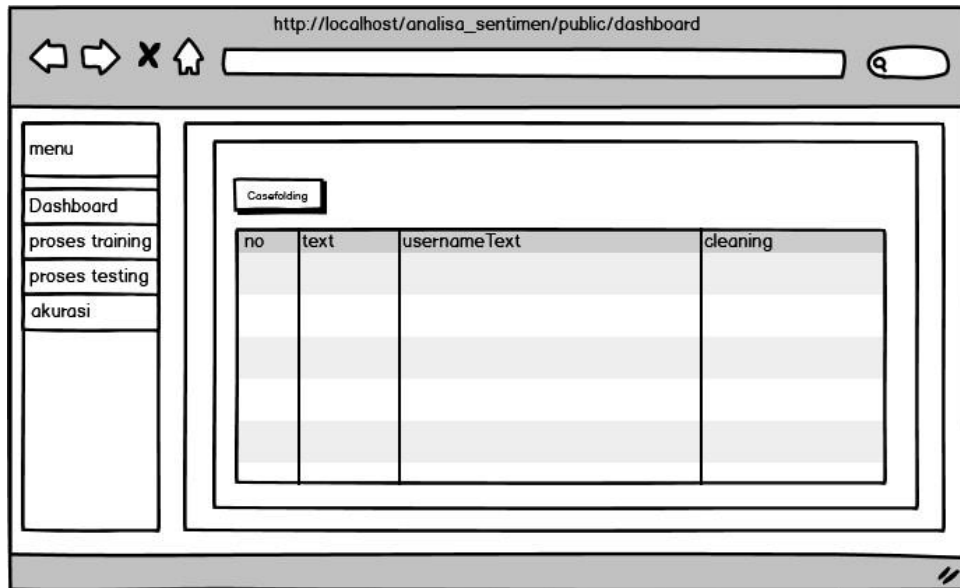
Pada Menu Preprocessing Cleaning Data Testing akan menampilkan Data Testing yang sudah melalui tahap Preprocessing Cleaning.



Gambar 4.35 Perancangan Menu Preprocessing Cleaning Data Testing.

#### 3.5.4.12 Perancangan Menu Preprocessing Casefolding Data Testing

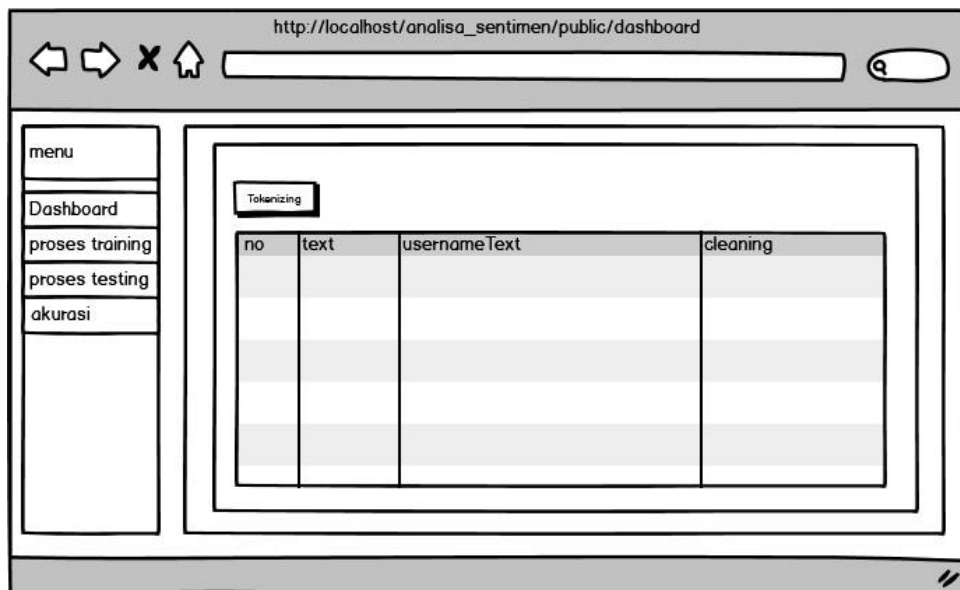
Pada Menu Preprocessing Casefolding Data Testing akan menampilkan Data Testing yang sudah melalui tahap Preprocessing Casefolding.



Gambar 4.36 Perancangan Menu Preprocessing Casefolding Data Testing.

#### 3.5.4.13 Perancangan Menu Preprocessing Tokenizing Data Testing

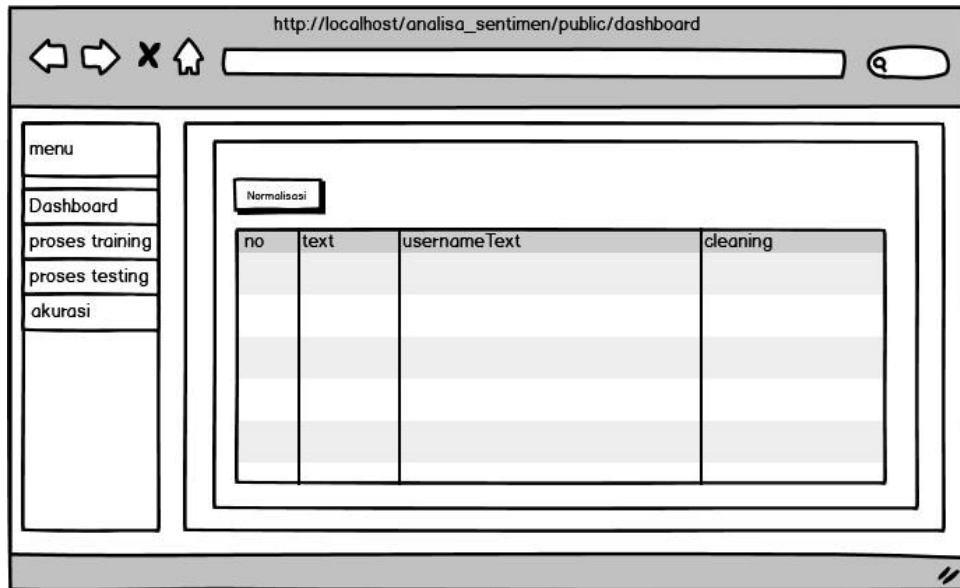
Pada Menu Preprocessing Tokenizing Data Testing akan menampilkan Data Testing yang sudah melalui tahap Preprocessing Tokenizing.



Gambar 4.37 Perancangan Menu Preprocessing Tokenizing Data Testing.

#### 3.5.4.14 Perancangan Menu Preprocessing Normalisasi Data Testing

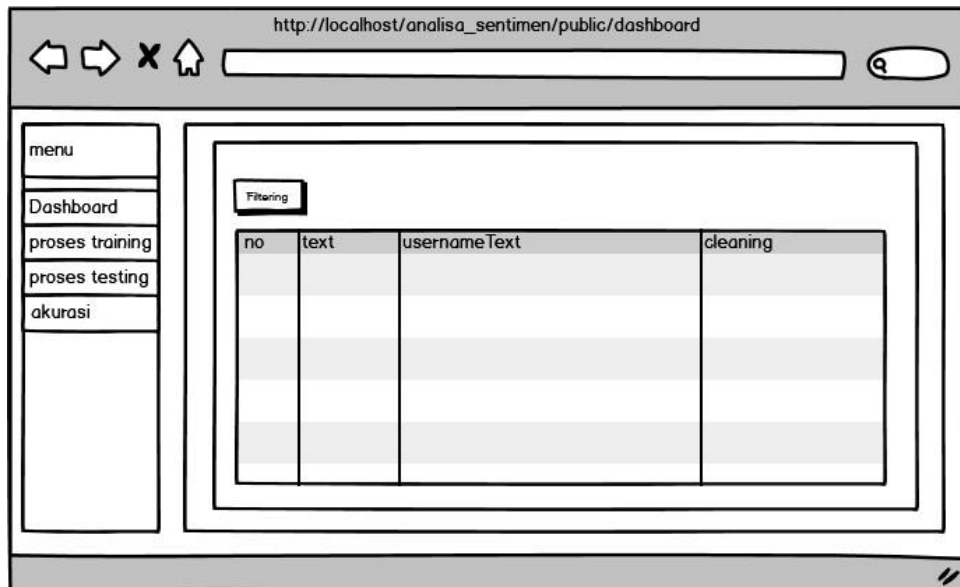
Pada Menu Preprocessing Normalisasi Data Testing akan menampilkan Data Testing yang sudah melalui tahap Preprocessing Normalisasi.



Gambar 4.38 Perancangan Menu Preprocessing Normalisasi Data Testing.

#### 3.5.4.15 Perancangan Menu Preprocessing Filtering Data Testing

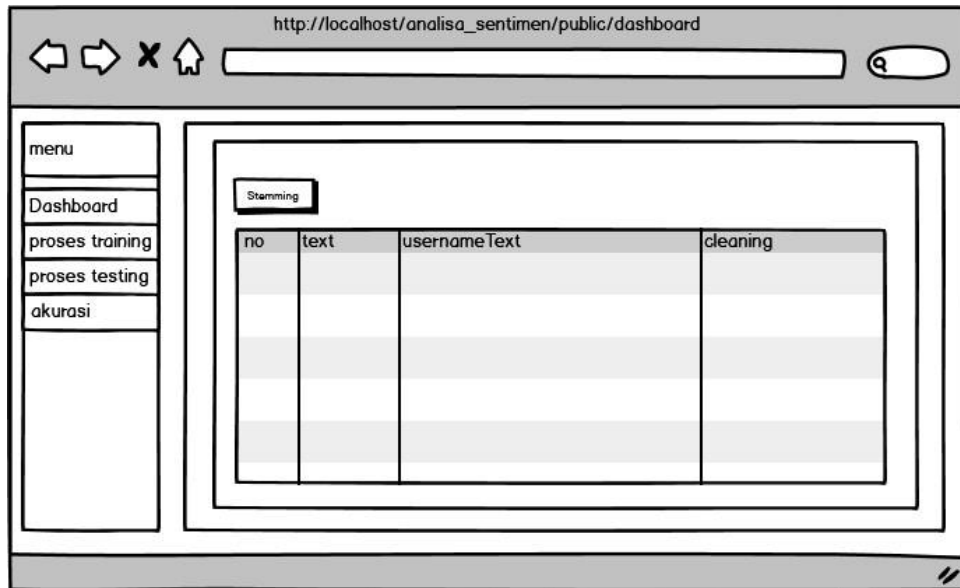
Pada Menu Preprocessing Filtering Data Testing akan menampilkan Data Testing yang sudah melalui tahap Preprocessing Filtering.



Gambar 4.39 Perancangan Menu Preprocessing Filtering Data Testing.

#### 3.5.4.16 Perancangan Menu Preprocessing Stemming Data Testing

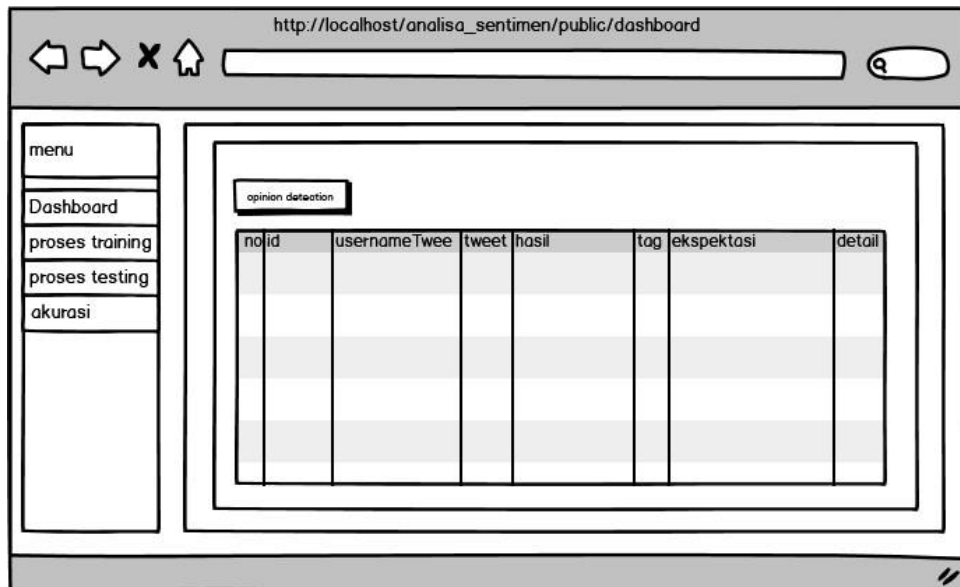
Pada Menu Preprocessing Stemming Data Testing akan menampilkan Data Testing yang sudah melalui tahap Preprocessing Stemming.



Gambar 4.40 Perancangan Menu Preprocessing Stemming Data Testing.

### 3.5.4.17 Perancangan Menu Opinion Detection

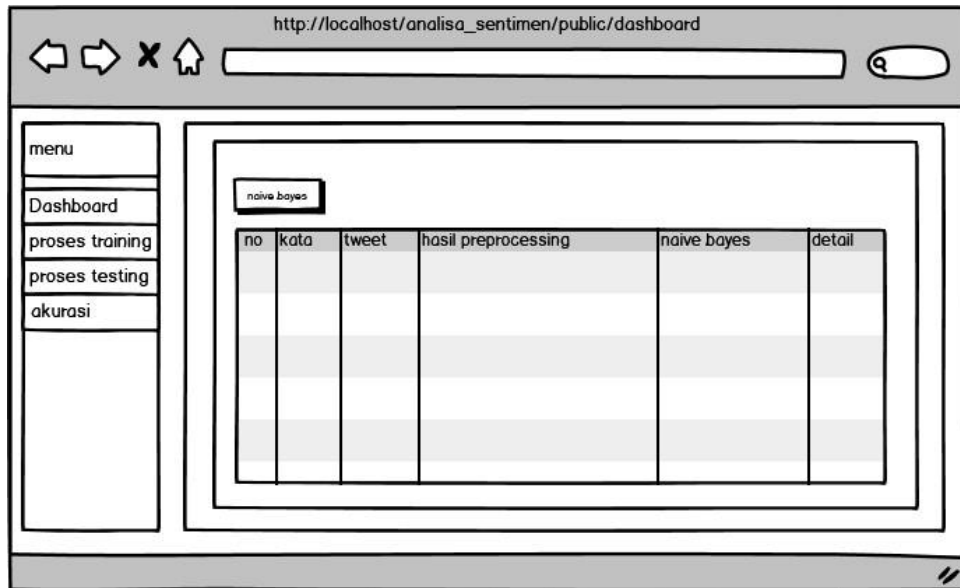
Pada Menu Opinion Detection akan menampilkan hasil dari proses Algoritma Opinion Detection.



Gambar 4.41 Perancangan Menu Opinion Detection.

### 3.5.4.18 Perancangan Menu Naïve Bayes

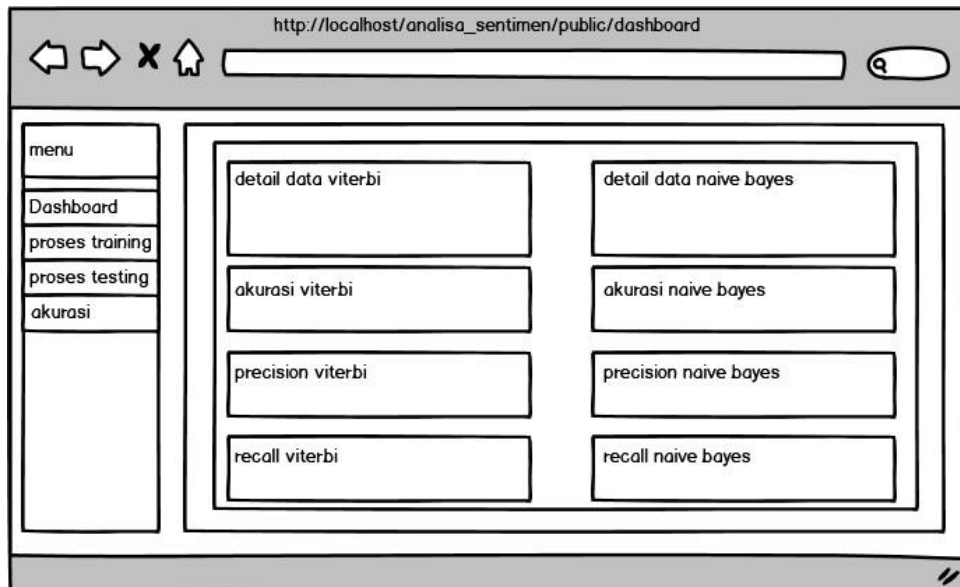
Pada Menu Naïve Bayes akan menampilkan hasil dari proses Algoritma Naïve Bayes.



Gambar 4.42 Perancangan Menu Naïve Bayes.

#### 3.5.4.19 Perancangan Menu Akurasi

Pada Menu Akurasi akan menampilkan hasil akurasi, precision dan recall dari proses Naïve Bayes dan proses POSTagging Viterbi.



Gambar 4.43 Perancangan Menu Akurasi.