

BAB 3

ANALISIS DAN DISAIN

Pada bab ini akan dijelaskan mengenai analisis, gambaran arsitektur keseluruhan, dan disain sistem analisis sentimen.

3.1. Analisis

Aplikasi ini merupakan aplikasi untuk menganalisis Tweet terhadap *provider* telekomunikasi, apakah masuk dalam kategori positif, negatif ataupun netral. Tweet yang diklasifikasi hanya terbatas pada bahasa Indonesia, dan data akan diambil dengan menggunakan *Twitter API*.

Terdapat dua tipe data yaitu *data training* dan *data testing*. *Data training* yaitu kumpulan *Tweet* tentang *provider x* dan *y* yang diambil pada jangka waktu tertentu kemudian dilakukan klasifikasi secara *manual (manual judgment)*. Sistem telah menyediakan sebuah *form* untuk memudahkan *user* untuk melakukan klasifikasi secara *manual*. *Manual judgment* dilakukan oleh tiga orang dan hasilnya akan dibandingkan. *Data testing* yaitu data yang diperoleh dari *Twitter*, dan *data* tersebut akan digunakan untuk melakukan *testing* klasifikasi secara otomatis.

Ada beberapa tahapan pada untuk membentuk *data training*, diantaranya yaitu *manual tagging*, *pre-processing*, *selection*. Pada tahap awal, kalimat – kalimat dikategorikan kedalam 3 *file* teks yang berbeda. Setelah itu dilakukan tahap *pre-processing* pada setiap kalimat di masing - masing *file* teks dan akan menghasilkan *term-term*. Lalu dilanjutkan dengan proses *selection*, pada proses ini kumpulan *term* tersebut akan dihitung dengan rumus dan akan diurutkan dari yang terbesar. Hasilnya akan diambil sebanyak *n* terbesar dan akan menjadi atribut pada *data training*.

3.1.1. Contoh Penerapan Analisis

Pada sub bab ini akan menjelaskan contoh bagaimana penerapan analisis suatu kasus. Terdapat kumpulan kalimat yang akan dikategorikan secara manual dalam 3 bagian, yaitu positif, negatif, dan juga netral. Pada tabel 3.1 adalah contoh kalimat yang sudah dikategorikan:

Tabel 3.1 Kumpulan Kalimat

Positif	bersabar, selalu semangat, tetap kuat, dan semua akan indah pada waktunya.. Jgn menyerah dengan keadaan semangat!! bahagia itu mdh. Semangat!! Bisa!! Terima kasih Tuhan atas segala rahmat yang telah Tuhan berikan kepadaku.
Negatif	Aduh...Cape istirahat dlu kecewa banget, udah cape2 ngerjain tetapi hasil tdk sesuai.. ga ada niat bljr
Netral	Selamat siang

Setelah sekumpulan kalimat tersebut telah dikategorikan secara manual atau yang biasa disebut dengan *manual tagging*, maka langkah selanjutnya yaitu proses *pre-processing*. *Pre-processing* terdiri dari beberapa proses yaitu *tokenizing*, *formalization*, *stopping*, dan *n-grams*.

3.1.1.1. Tokenizing

Pada proses *tokenizing* setiap kalimat akan dibagi menjadi beberapa bagian (*token*). Tidak hanya itu, dalam tahap ini semua tanda baca akan dihapus. Dan juga setiap kata yang berawalan dengan huruf kapital akan diubah menjadi huruf kecil. Tabel 3.2 merupakan hasil kalimat setelah dilakukan *tokenizing*.

Tabel 3.2 Kalimat setelah dilakukan Tokenizing

Positif	bersabar selalu semangat tetap kuat dan semua akan indah pada waktunya jgn menyerah dengan keadaan semangat bahagia itu mdh semangat bisa terima kasih tuhan atas segala rahmat yang telah tuhan berikan kepadaku
Negatif	aduh cape istirahat dlu kecewa banget udah cape ngerjain tetapi hasil tdk sesuai ga ada niat bljr
Netral	selamat siang

3.1.1.2. Formalization

Proses selanjutnya yaitu proses *formalization*. Pada tahap ini setiap *token* dengan kata yang tidak baku akan diubah menjadi kata baku. Misalnya kata ‘jgn’ diubah menjadi ‘jangan’, kata ‘ga’ diubah menjadi ‘tidak’. Kamus *formalization* yang digunakan didapat dari sebuah *library natural language processing* Tim Lab Grafika dan Intelegensia Buatan dari Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung khusus untuk metode “*IndonesianSentenceFormalization*” serta ditambahkan beberapa kata yang perlu ditambahkan untuk penelitian seperti kata ‘trobek’ menjadi ‘masalah’ atau misalkan kata ‘lemot’ menjadi ‘lambat. Tabel 3.3 merupakan hasil setelah dilakukan proses *formatlization*.

Tabel 3.3 Kalimat setelah proses Formalization

Positif	bersabar selalu semangat tetap kuat dan semua akan indah pada waktunya jangan menyerah dengan keadaan semangat bahagia itu mudah semangat bisa terima kasih tuhan atas segala rahmat yang telah tuhan berikan kepadaku
Negatif	aduh cape istirahat dulu kecewa banget sudah cape ngerjain tetapi hasil tidak sesuai tidak ada niat belajar
Netral	selamat siang

3.1.1.3. Stopping

Tahap selanjutnya yaitu *stopping*. Pada tahap akan menghapus seluruh kata penghubung, artinya setiap kata-kata yang tidak bermakna akan dihapus. Kata-kata yang akan dihapus yaitu ‘dan’, ‘akan’, ‘pada’, ‘dengan’, ‘itu’, ‘yang’, ‘kau’, ‘banget’, ‘tetapi’, ‘telah’, ‘dulu’ dan ‘makin’. Tabel 3.4 merupakan hasil dari proses *stopping*.

Tabel 3.4 Kalimat setelah proses Stopping

Positif	bersabar selalu semangat tetap kuat semua indah pada waktunya jangan menyerah keadaan semangat bahagia mudah semangat bisa terima kasih tuhan atas segala rahmat tuhan berikan kepadaku
Negatif	aduh cape istirahat

	kecewa banget sudah cape ngerjain hasil tidak sesuai tidak ada niat belajar
Netral	selamat siang

3.1.1.4. N-Grams

Proses selanjutnya yaitu *n-grams*. Pada tahap ini sistem akan mengambil sejumlah *n* kata sebagai suatu *term* dan menghitung berapa banyak kata itu muncul. *N-grams* yang digunakan yaitu *bigram* dan *unigram*. Hasil *n-grams* dapat dilihat pada tabel 3.5.

Tabel 3.5 Kalimat Setelah Proses Unigram dan Bigram

Class	Kalimat	Unigram	Bigram
Positif	bersabar selalu semangat tetap kuat semua indah pada waktunya jangan menyerah keadaan semangat	bersabar selalu semangat tetap kuat semua indah waktunya jangan menyerah keadaan semangat	bersabar selalu selalu semangat semangat tetap tetap kuat kuat semua semua indah indah waktunya waktunya jangan jangan menyerah menyerah keadaan keadaan semangat
Positif	bahagia mudah semangat bisa	bahagia mudah semangat bisa	bahagia mudah mudah semangat semangat bisa
Positif	terima kasih tuhan atas segala rahmat berikan kepadaku	terima kasih tuhan atas segala rahmat Tuhan berikan kepadaku	terima kasih kasih tuhan tuhan atas atas segala segala rahmat rahmat tuhan tuhan berikan berikan kepadaku
Negatif	aduh cape istirahat	aduh cape istirahat	aduh cape cape istirahat
Negatif	kecewa banget sudah cape	kecewa banget	kecewa banget banget sudah

Class	Kalimat	Unigram	Bigram
	ngerjain hasil tidak sesuai	sudah cape ngerjain hasil tidak sesuai	sudah cape cape ngerjain ngerjain hasil hasil tidak tidak sesuai
Negatif	tidak ada niat belajar	tidak ada niat belajar	tidak ada ada niat niat belajar
Netral	selamat siang	selamat siang	selamat siang

Tabel 3.5 merupakan hasil setelah dilakukan *bigram* dan *unigram*. Setelah itu, *term* akan dihitung sesuai dengan kemunculan di setiap dokumen. Pada tabel di bawah ini merupakan hasil frekuensi yang muncul pada setiap *term*.

Tabel 3.6 Tabel *Unigram* Positif

No	Term unigram dan bigram	Frekuensi
1	bersabar	1
2	selalu	1
3	semangat	3
4	tetap	1
5	kuat	1
6	semua	1
7	indah	1
8	waktunya	1
9	jangan	1
10	menyerah	1
11	keadaan	1
12	bahagia	1
13	mudah	1
14	bisa	1
15	terima	1
16	kasih	1
17	tuhan	2
18	atas	1
19	segala	1
20	rahmat	1
21	berikan	1
22	kepadaku	1

Tabel 3.7 Tabel *Bigram* Positif

No	Term unigram dan bigram	Frekuensi
1	selalu semangat	1
2	semangat tetap	1
3	bersabar selalu	1
4	tetap kuat	1
5	kuat semua	1
6	semua indah	1
7	indah pada	1
8	pada waktunya	1
9	waktunya jangan	1
10	jangan menyerah	1
11	menyerah keadaan	1
12	keadaan semangat	1
13	bahagia mudah	1
14	mudah semangat	1
15	semangat bisa	1
16	terima kasih	1
17	kasih tuhan	1
18	atas segala	1
19	segala rahmat	1
20	rahmat berikan	1
21	berikan kepadaku	1

Tabel 3.8 Tabel *Unigram* Negatif

No	Term unigram dan bigram	Frekuensi
1	aduh	1
2	cape	2
3	istirahat	1
4	kecewa	1
5	banget	1
6	sudah	1
7	ngerjain	1
8	hasil	1
9	tidak	2
10	sesuai	1
11	ada	1
12	niat	1
13	belajar	1

Tabel 3.9 Tabel *Bigram* Negatif

No	Term unigram dan bigram	Frekuensi
1	aduh cape	1

No	Term unigram dan bigram	Frekuensi
2	cape istirahat	1
3	kecewa banget	1
4	banget sudah	1
5	sudah cape	1
6	cape ngerjain	1
7	ngerjain hasil	1
8	hasil tidak	1
9	tidak sesuai	1
10	tidak ada	1
11	ada niat	1
12	niat belajar	1

Tabel 3.10 Tabel *N-grams* Netral

No	Term unigram dan bigram	Frekuensi
1	selamat	1
2	siang	1
3	selamat siang	1

3.1.1.5. Mutual Information

Pada sub bab ini akan menjelaskan tahap selanjutnya setelah tahap *preprocessing*. Setelah frekuensi kemunculan setiap *term* diketahui, maka akan dilakukan penyeleksian *term*. Caranya dengan menghitung berapa banyak jumlah *term x* di *class y* dan *class non y*. Lalu dimasukkan ke dalam rumus *mutual information*. Contoh untuk *term* ‘semangat’, jumlah *term* ‘semangat’ pada *class* ‘positif’ ada 3, sedangkan pada *class* selain positif (negatif, netral) ada 0. Sehingga nilai N_{11} dan N_{10} pada rumus MI secara berurutan adalah 3 dan 0. Untuk mencari nilai N_{01} dengan cara menghitung jumlah *term* selain *term* ‘semangat’ pada *class* positif, sedangkan N_{00} menghitung jumlah *term* selain *term* ‘semangat’ pada *class* selain positif yaitu pada *class* negatif dan netral. Nilai N_{01} dan N_{00} secara berurutan yaitu 43 dan 30. Lalu setiap variabel tersebut akan dimasukkan ke dalam rumus MI. Tujuan dari MI yaitu memilih *term* yang mempunyai peranan penting dalam setiap *class*.

Tabel 3.11 Tabel Hasil Mutual Information *Class* Positif

Term	Hasil MI
semangat	0.008751027

Term	Hasil MI
tuhan	0.003773987
indah	0.00053165

Tabel 3.12 Tabel Hasil Mutual Information *Class* Negatif

Term	Hasil MI
cape	0.023696812
tidak	0.023696812
kecewa	0.01070534

Tabel 3.13 Tabel Hasil Mutual Information *Class* Netral

Term	Hasil MI
selamat	0.074252384
siang	0.074252384
selamat siang	0.074252384

Tabel 3.11 , 3.12, dan 3.13 merupakan hasil dari penghitungan *mutual information* yang sudah diambil tiga terbesar dari masing-masing *class*. Jadi *term* yang paling berperan pada *class* positif yaitu ‘semangat’ dan ‘tuhan’, pada *class* negatif yaitu ‘cape’ dan ‘tidak’, serta pada netral yaitu ‘siang’ dan ‘selamat siang’.

3.1.1.6. Proses Training Naïve Bayes

Term yang sudah terseleksi digunakan untuk *keyword* pada proses *training Naïve Bayes*. Tabel 3.14 merupakan tabel untuk proses *training Naïve Bayes* dengan mengambil dua teratas pada setiap *class*.

Tabel 3.14 Tabel Data Training *Naïve Bayes*

Semangat	Tuhan	Cape	Tidak	Selamat Siang	Siang	Class

Sistem akan memeriksa setiap kalimat untuk setiap *term*. Jika sebuah kalimat mengandung *keyword* maka akan diberi nilai 1 sedangkan jika tidak mengandung *keyword* maka akan diberi nilai 0. Berikut contoh-contoh kalimat dalam pembentukan data *training* :

1. Mendapat pencerahan dari Tuhan membuatku menjadi semangat kembali.
2. Cape banget sih kuliah, banyak sekali yang harus diselesaikan.
3. Siang ini mendapat dukungan dari teman-teman membuatku menjadi semangat.
4. Basket dilapangan terbuka pada siang hari membuat menjadi cepat cape.
5. Selamat siang para hadirin sekalian.

Tabel 3.15 Tabel Proses Naïve Bayes

No	Tuhan	Semangat	Cape	Tidak	Selamat Siang	Siang	Class
1.	1	1	0	0	0	0	Positif
2.	0	0	1	0	0	0	Negatif
3.	0	1	0	0	0	1	Positif
4.	0	0	1	0	0	1	Negatif
5.	0	0	0	0	1	1	Netral

Langkah selanjutnya, sistem akan menghitung total kemunculan masing-masing *keyword* pada setiap *class*, lalu dibagi dengan total keseluruhan kemunculan atribut dari masing-masing *class*. Sehingga didapatkan nilai probabilitas sebagai berikut:

Tabel 3.16 Tabel Nilai Probabilistik

Class	Tuhan		Semangat		Cape		Tidak		Selamat Siang		Siang	
	1	0	1	0	1	0	1	0	1	0	1	0
Positif	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{2}{2}$	$\frac{0}{2}$	$\frac{0}{2}$	$\frac{2}{2}$	$\frac{0}{2}$	$\frac{2}{2}$	$\frac{0}{2}$	$\frac{2}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
Negatif	$\frac{0}{2}$	$\frac{2}{2}$	$\frac{0}{2}$	$\frac{2}{2}$	$\frac{2}{2}$	$\frac{0}{2}$	$\frac{0}{2}$	$\frac{2}{2}$	$\frac{0}{2}$	$\frac{2}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
Netral	$\frac{0}{1}$	$\frac{1}{1}$	$\frac{0}{1}$	$\frac{1}{1}$	$\frac{0}{1}$	$\frac{1}{1}$	$\frac{0}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{0}{1}$	$\frac{1}{1}$	$\frac{0}{1}$

Pada tabel 3.16 terdapat nilai probabilistik yang bernilai 0. Oleh karena itu perlu dilakukan penambahan agar tidak ada nilai probabilistik yang bernilai nol. Proses penambahan jika terdapat nilai 0 disebut dengan *Laplacian Smoothing*. Setiap nilai akan ditambahkan satu. Berikut hasil dari proses *Laplacian Smoothing* pada tabel 3.17 :

Tabel 3.17 Tabel Hasil Nilai Probabilistik dengan *Laplacian Smoothing*

Class	Tuhan		Semangat		Cape		Tidak		Selamat		Siang	
	1	0	1	0	1	0	1	0	1	0	1	0
Positif	2/4	2/4	3/4	1/4	1/4	3/4	1/4	3/4	1/4	3/4	2/4	2/4
Negatif	1/4	3/4	1/4	3/4	3/4	1/4	1/4	3/4	1/4	3/4	2/4	2/4
Netral	1/3	2/3	1/3	2/3	1/3	2/3	1/3	2/3	2/3	1/3	2/3	1/3

3.1.1.7. Testing

Setelah proses *training* selesai maka akan dilakukan pengujian. Data yang digunakan pada saat *testing* yaitu data yang berasal dari *twitter*. Setiap kalimat dari data *testing* akan dimasukkan ke dalam kategori positif, negatif atau netral. Berikut adalah contoh kalimat untuk data *testing*:

“Semangat untuk teman-teman yang bertanding pada siang hari ini, Tuhan sertai kalian!”

Langkah pertama yaitu mengubah kalimat testing ini ke dalam format *Naïve Bayes* dan dihitung menggunakan rumus $P(Y|X) = P(X_i|Y) * P(Y)$. Berikut proses *testing*:

Tabel 3.18 Tabel Data Testing dalam Format *Naïve Bayes*

Semangat	Tuhan	Cape	Tidak	Selamat	Siang	Siang
1	1	0	0	0		1

Lalu cari nilai $P(Y)$ pada masing-masing *class*:

- $P(\text{'Positif'}) = \frac{2}{5} = 0.4$
- $P(\text{'Negatif'}) = \frac{2}{5} = 0.4$
- $P(\text{'Netral'}) = \frac{1}{5} = 0.2$

Tahap selanjutnya mencari nilai $P(X_i|Y)$:

- a) $P(X_i | \text{'Positif'}) = \frac{2}{4} * \frac{3}{4} * \frac{2}{4} = 0.1875$
- b) $P(X_i | \text{'Negatif'}) = \frac{1}{4} * \frac{1}{4} * \frac{2}{4} = 0.03125$
- c) $P(X_i | \text{'Netral'}) = \frac{1}{3} * \frac{1}{3} * \frac{2}{3} = 0.0740741$

Tahap terakhir yaitu nilai $P(X_i|Y)$ akan dikalikan dengan $P(Y)$:

- a) $P(X_i | \text{'Positif'}) * P(\text{'Positif'}) = 0.4 * 0.25 = 0.075$
- b) $P(X_i | \text{'Negatif'}) * P(\text{'Negatif'}) = 0.4 * 0.03125 = 0.0125$
- c) $P(X_i | \text{'Netral'}) * P(\text{'Netral'}) = 0.2 * 0.0740741 = 0.014815$

Dari hasil diatas maka dapat disimpulkan contoh kalimat *training* di atas masuk dalam kategori positif, karena hasil dari $P(X_i|Y) * P(Y)$ pada *class* positif merupakan hasil yang paling besar yaitu 0.075. Sedangkan pada *class* negatif dan netral secara berurutan bernilai 0.0125 dan 0.014815.

3.2. Gambaran Keseluruhan

Pada bagian ini akan dijelaskan mengenai fitur pada aplikasi yang akan dibangun.

3.2.1. Persyaratan Antarmuka Eksternal

Perangkat lunak ini memiliki persyaratan-persyaratan antar muka eksternal sebagai berikut:

1. Dibutuhkan koneksi internet untuk mengambil data dari *Twitter*.
2. Data kalimat yang diambil harus berbahasa Indonesia

3.2.2. Antarmuka dengan Pengguna

Pada bagian ini akan dijelaskan tentang persyaratan antarmuka dengan pengguna atau *user*:

1. *User* dapat melihat hasil dari setiap tahap dalam *preprocessing*
2. *User* dapat melakukan klasifikasi secara otomatis.

3.2.3. Antarmuka Perangkat Keras

Di bagian ini akan dijelaskan antarmuka perangkat keras yang dipakai baik saat dilakukan pengembangan maupun untuk *end-user* yang dapat dilihat pada daftar di bawah ini :

3.2.3.1. Spesifikasi Antarmuka Perangkat Keras Saat Pengembangan

Antarmuka Perangkat Keras yang digunakan saat pengembangan aplikasi memiliki spesifikasi sebagai berikut :

1. *Processor* Intel i5-2410M, 2.3Ghz
2. *RAM* 4GB.
3. *Harddisk* 1 TB.
4. *Mouse*
5. *Keyboard*

3.2.3.2. Spesifikasi Antarmuka Perangkat Keras *End-User*

Antarmuka Perangkat Keras minimal yang digunakan untuk *end-user* untuk penggunaan aplikasi adalah sebagai berikut :

1. *Microsoft Windows 7.*
2. *RAM 2GB.*
3. *Hard Disk 80 GB.*
4. *Processor Intel Core 2 Duo.*
5. *Monitor*
6. *Epson TM-u220 Series.*
7. *Mouse*
8. *Keyboard*

3.2.4. Antarmuka Perangkat Lunak

Di bagian ini akan dijelaskan antarmuka perangkat lunak yang diperlukan untuk menjalankan perangkat lunak ini sebagai berikut :

1. Sistem operasi *Windows 7* (direkomendasikan)/ *Windows 8*
2. *Visual Studio 2012*
3. *SQL Server Management Studio*

3.3. Disain Perangkat Lunak

Disain Perangkat Lunak terdiri atas pemodelan perangkat lunak, disain penyimpanan data dan disain antarmuka, dengan penjelasan sebagai berikut :

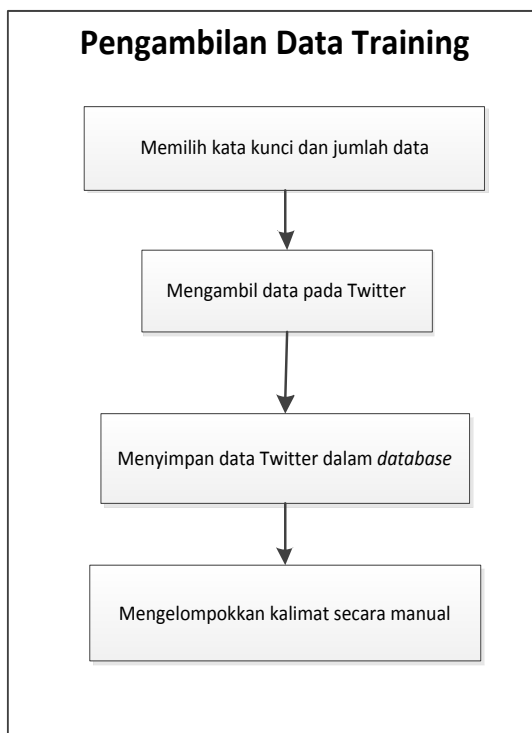
3.3.1. Pemodelan Perangkat Lunak

Berikut adalah pemodelan perangkat lunak yang akan diterapkan pada sistem analisis sentimen.

3.3.1.1. Arsitektur Perangkat Lunak

Arsitektur pada sistem analisis sentimen terhadap *provider* akan dibagi menjadi tiga bagian yaitu arsitektur pengambilan *data training*, arsitektur pemilihan *term*, dan arsitektur sistem analisis.

3.3.1.1.1 Arsitektur Pembuatan Data Training

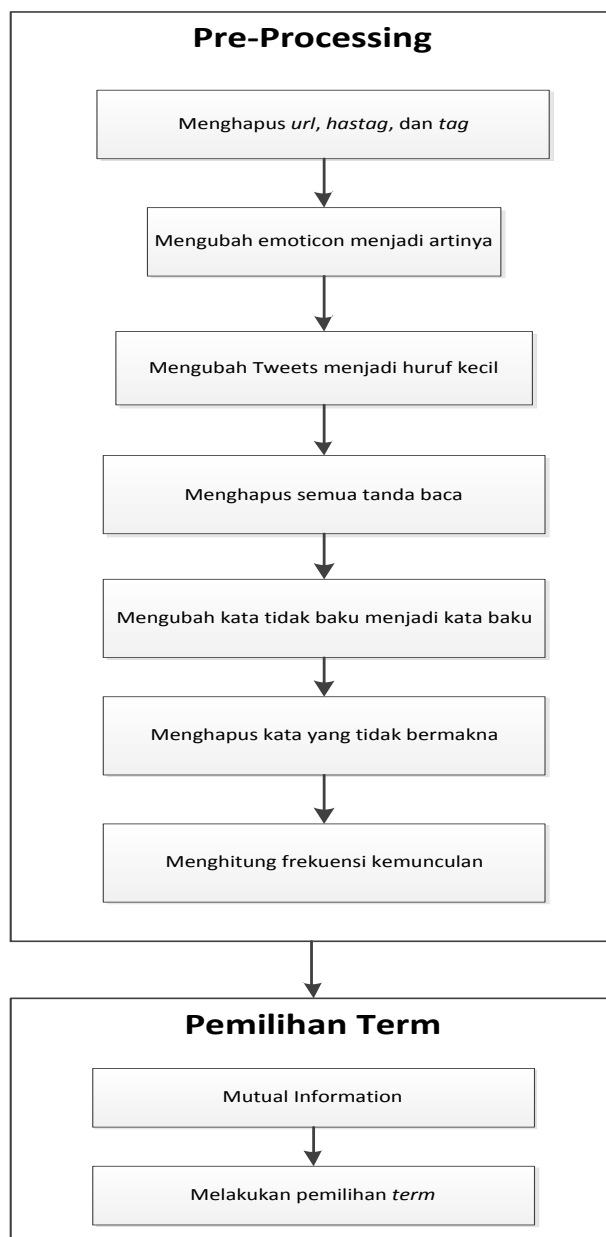


Gambar 3.1 Arsitektur Pengambilan Data Training

Gambar 3.1 menunjukkan arsitektur proses dalam pengambilan *data training* yang akan dibuat pada sistem. Langkah-langkah proses pengambilan *data training* adalah sebagai berikut :

1. *User* memilih kata kunci dan jumlah data yang akan diambil dalam pencarian *Tweets*.
2. Sistem akan mengambil data pada Twitter dengan menggunakan Twitter API. Diperlukan *consumer key*, *consumer secret*, *access token*, dan *access token secret*.
3. Data yang sudah diambil tersebut akan disimpan pada *database*.
4. Data yang sudah disimpan tersebut dikategorikan secara manual dalam tiga kategori, yaitu positif, negatif, atau netral.

3.3.1.1.2 Arsitektur Pemilihan Term



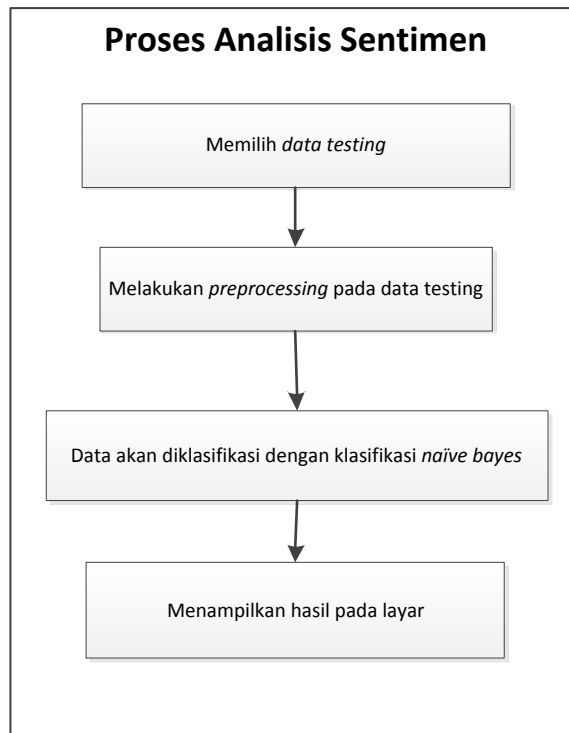
Gambar 3.2 Arsitektur Pemilihan Term

Gambar 3.2 menunjukkan arsitektur pemilihan *term* yang akan dibuat dalam sistem. Langkah-langkah proses pemilihan *term* adalah sebagai berikut:

1. Sebelum proses pemilihan *term* dilakukan *preprocessing* pada *data training* yang sudah dikelompokkan menjadi tiga kategori. Berikut tahap-tahap pada *preprocessing* :
 - a. Menghapus seluruh *url* (<http://contoh.com>), *hashtag* ([#provider](#)), dan *tag* ([@username](#)).

- b. Mengubah *emoticon* pada *Tweets* sesuai dengan arti dalam *emoticon* tersebut.
 - c. Mengubah semua huruf kapital menjadi huruf kecil pada *Tweets*.
 - d. Menghapus semua tanda baca atau karakter khusus selain huruf dan angka. Seperti koma, kutip, tanda seru, tanda tanya, titik dua, kutip, dan lain-lain.
 - e. Mengubah semua kata tidak baku menjadi kata baku. Misalnya kata 'ngga', 'ga', 'gak', 'kaga' menjadi kata 'tidak'.
 - f. Menghapus semua kata-kata yang tidak memiliki makna maupun kata-kata yang paling sering muncul tetapi tidak memiliki peran dalam masing-masing kategori.
 - g. Setiap kata akan dihitung berdasarkan frekuensi kemunculannya pada masing-masing *class*(positif, negatif, netral).
2. Pemilihan *term* dibagi menjadi dua tahap, yaitu:
- a. Menghitung *mutual information* pada seluruh kata.
 - b. Mengambil n *term* pada masing-masing kelas yang memiliki nilai *mutual information* tertinggi pada setiap kelas.

3.3.1.1.3 Arsitektur Analisis Sentimen



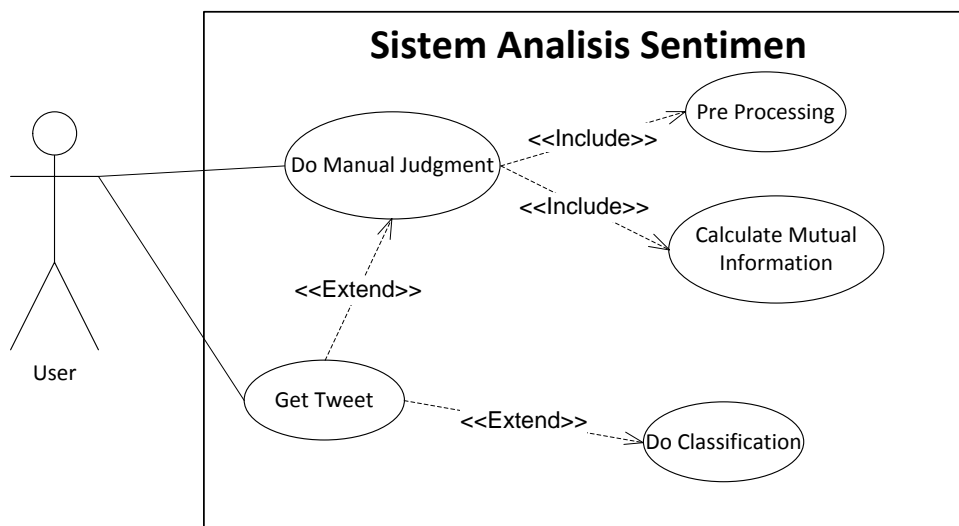
Gambar 3.3 Arsitektur Proses Analisis Sentimen

Gambar 3.3 menunjukkan arsitektur proses analisis sentimen yang akan diterapkan dalam sistem. Tahap-tahap analisis sentimen sebagai berikut:

1. Sistem menyediakan dua macam *data testing* yaitu, *data testing* yang diambil secara manual lalu disimpan dalam *database* dan *data testing* yang diambil secara otomatis dalam jangka waktu tertentu.
2. Sebelum dilakukan pengklasifikasian, *data testing* akan melalui tahap *preprocessing* sampai atribut-atribut tersebut bisa dibandingkan dengan *term* yang sudah dipilih pada *data training*.
3. Sistem akan melakukan perhitungan dengan metode *naive bayes*. Lalu masing-masing kalimat akan dimasukkan ke dalam kelas yang nilai probabilitasnya paling besar.
4. Sistem akan menampilkan hasil pada layar yang dapat dilihat oleh konsumen.

3.3.1.2. Use Case

Pada sub bab ini akan menjelaskan tentang *use case* diagram dari sistem utama. Gambar 3.4 merupakan gambaran *use case* sistem.



Gambar 3.4 Use Case

3.3.1.3. Use Case Skenario

Pada sub bab ini akan menjelaskan *use case* skenario dari *use case* pada bab 3.3.1.2 di atas.

3.3.1.3.1 Use Case Do Manual Judgment

- Use-Case* : *Do Manual Judgment*
- Description* : *User* mengklasifikasi kalimat secara manual ke dalam kategori positif, negatif, atau netral.
- Participating Actor* : *User*
- Quality Requirement* : *User* dapat mengklasifikasikan kalimat secara manual.
- Main Course* : Sistem akan menyimpan kategori tiap teks sesuai yang sudah diklasifikasi oleh *user*.
- Pre-Condition* : Menampilkan *data Tweets* yang sudah diambil sesuai dengan *Id* yang dipilih.
- Post-Condition* : Mengubah kategori setiap kalimat sesuai dengan pilihan *user* lalu disimpan di database agar bisa dilakukan *preprocessing* pada *data training*.

3.3.1.3.2 Use Case Preprocessing

- Use-Case* : *Pre-processing*
- Description* : Sistem akan melakukan tahap *pre-processing* pada suatu *file*.
- Participating Actor* : Sistem
- Quality Requirement* : Sistem berhasil melakukan *pre-processing*
- Main Course* : Sistem menampilkan setiap *term* beserta frekuensi kemunculannya.
- Pre-Condition* : Menampilkan kalimat sebelum dilakukan tahap *pre-processing*.
- Post-Condition* : Data *term* beserta frekuensi kemunculannya akan digunakan pada tahap *mutual information*.

3.3.1.3.3 Use Case Calculate Mutual Information

- Use-Case* : *Calculate Mutual Information*
- Description* : Sistem akan menghitung nilai dari setiap *term*.
- Participating Actor* : Sistem
- Quality Requirement* : Sistem berhasil menghitung nilai setiap *term* dengan tepat.
- Main Course* : Sistem melakukan penyeleksian setiap *term*.
- Pre-Condition* : Hasil *term* pada tahap *pre-processing* akan digunakan pada tahap *mutual information*.
- Post-Condition* : Hasil dari *mutual information* akan digunakan pada tahap klasifikasi dengan *naïve bayes*.

3.3.1.3.4 Use Case Get Tweets

- Use-Case* : *Get Tweet*
- Description* : *User* mengambil data dari Twitter dengan menggunakan Twitter API sesuai topik yang dipilih.
- Participating Actor* : *User*
- Quality Requirement* : *User* masuk ke dalam sistem '*Get Tweet*'.

- Main Course* : Sistem mengambil data dan menampilkan data-data yang berhasil diambil. Setelah itu sistem menyimpan data tersebut ke dalam *database*.
- Pre-Condition* : *User* memilih berapa banyak data yang akan diambil beserta dengan topik nya.
- Post-Condition* : Sistem menyimpan data dalam database lalu menampilkan data.

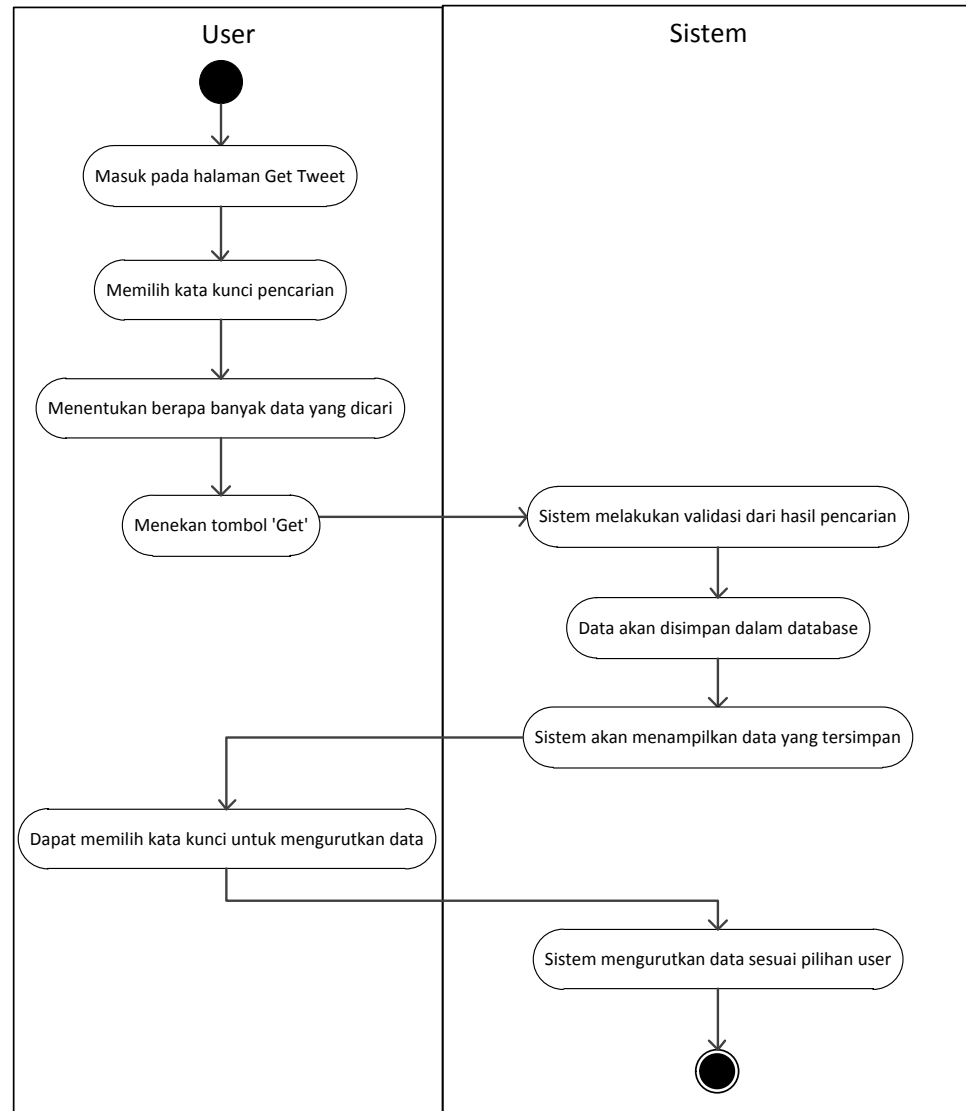
3.3.1.3.5 Use Case Do Classification

- Use-Case* : *Do Classification*
- Description* : Memisahkan setiap *term* menjadi keyword dengan probabilitiknya masing-masing.
- Participating Actor* : Sistem
- Quality Requirement* : Sistem berhasil menyeleksi *term* menjadi sebuah *keyword*.
- Main Course* : Setiap *term* pada setiap kelas akan diseleksi menjadi sebuah *keyword*.
- Pre-Condition* : Hasil *term* pada tahap *mutual information* akan diurutkan dari yang terbesar hingga yang terkecil.
- Post-Condition* : N teratas akan dijadikan sebuah *keyword* dan setiap kalimat *data training* akan ditransformasi menjadi format *naïve bayes*.

3.3.1.4. Activity Diagram

Pada sub bab ini akan dibahas mengenai *activity diagram* pada yang diambil dari *use case* pada bab 3.3.1.2.

3.3.1.4.1 Activity Diagram Get Tweets



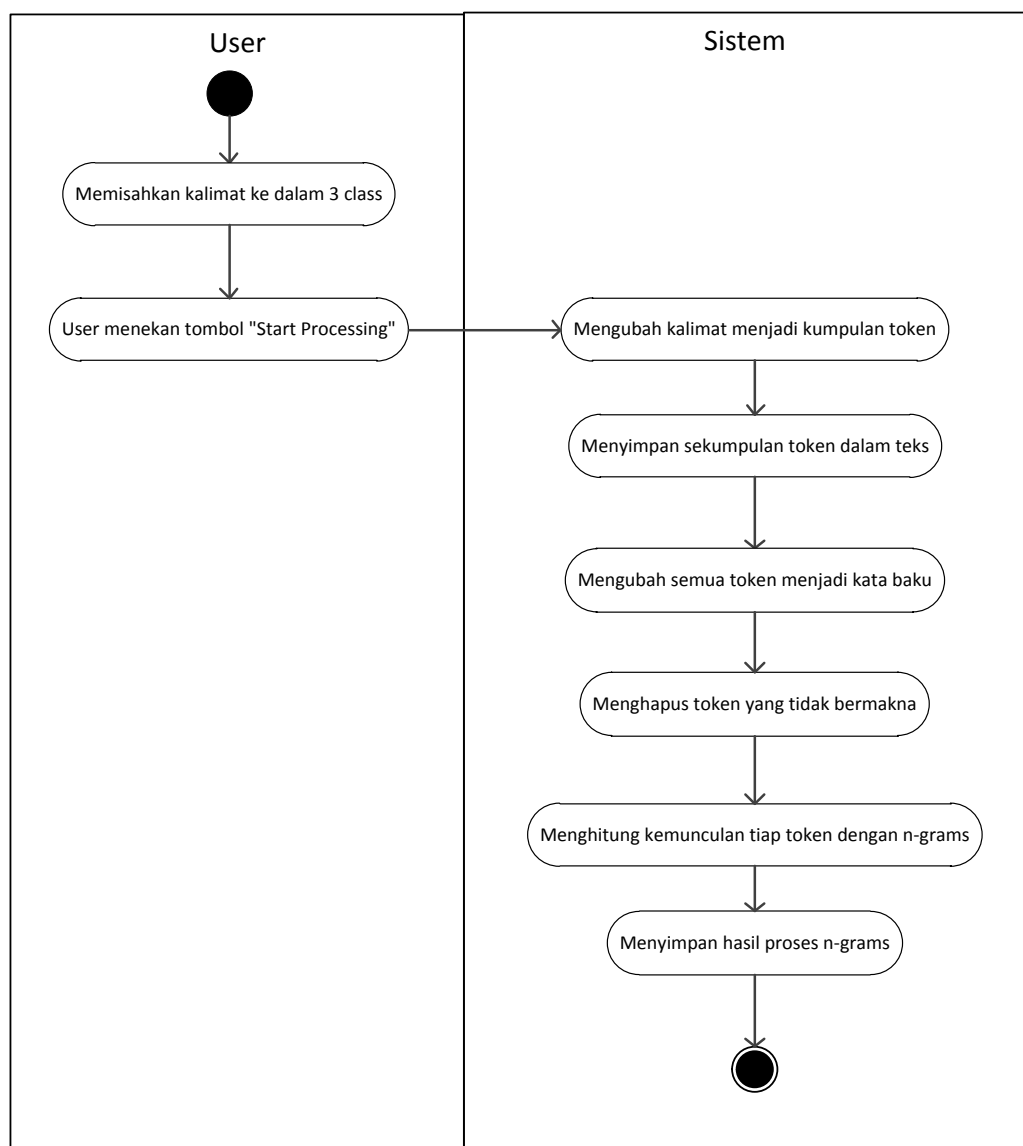
Gambar 3.5 Activity Diagram Get Tweets

Pada gambar 3.5 menunjukkan *activity diagram* pada proses *get Tweets*. Pertama-tama *user* masuk pada halaman “*Get Tweets*”. Pada halaman ini *user* bisa memilih kata kunci pencarian dan menentukan berapa banyak data yang akan dicari dalam Twitter. Setelah itu sistem akan menggunakan Twitter API untuk melakukan pencarian data. Tidak semua data hasil pencarian dimasukkan dalam database, tetapi sistem memilih data dengan ketentuan tertentu. Pertama sistem tidak akan

menyimpan *Tweet* yang memiliki “*ID_Tweet*” yang sudah ada dalam database. Kedua sistem tidak menyimpan semua *retweet* (RT), sehingga *Tweet* dengan dua karakter pertama “RT” tidak akan disimpan dalam database. Dan yang ketiga sistem tidak akan menyimpan *Tweets* dari *provider* tertentu.

Data yang sudah disimpan pada *database* akan ditampilkan oleh sistem. *User* dapat mengurutkan data sesuai kata kunci yang sudah disediakan oleh sistem. Kata kunci tersebut tersedia dalam bentuk *combo box* yang berisi “*ID*”, “*Author*”, “*Text*”, “*Date*”, dan “*Category*”.

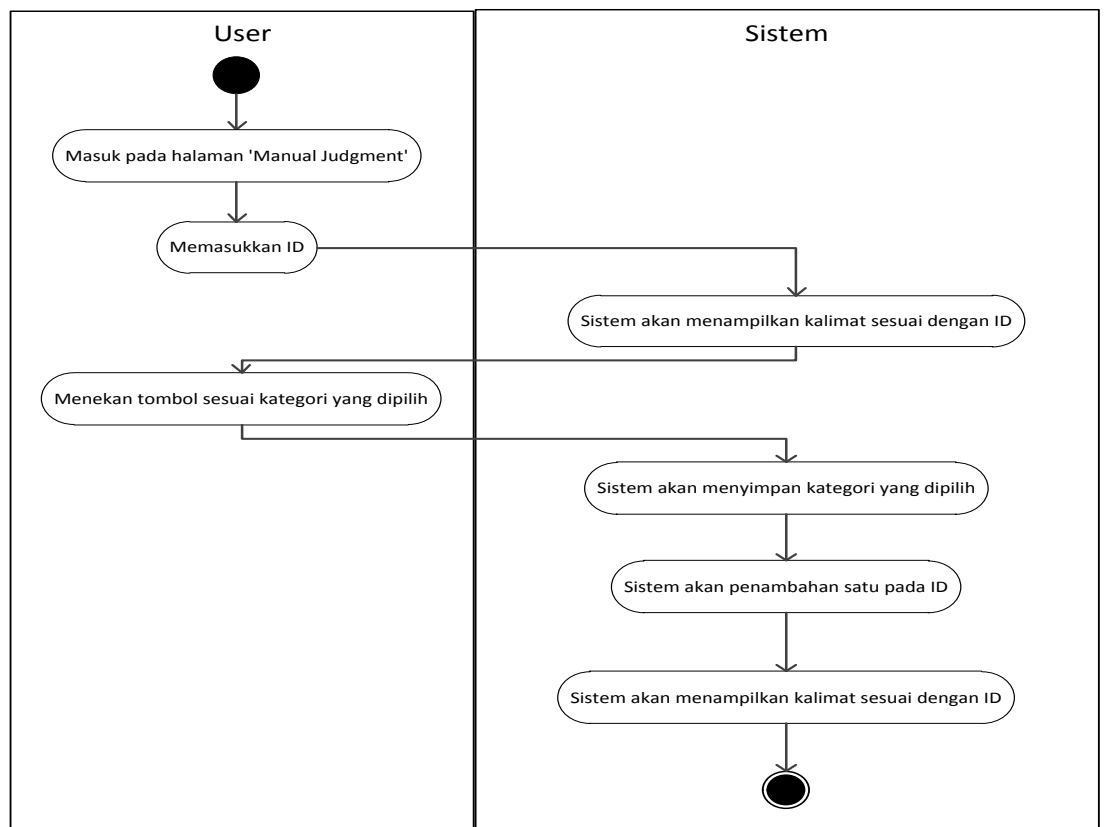
3.3.1.4.2 Activity Diagram Pre Processing



Gambar 3.6 Activity Diagram Pre Processing

Gambar 3.6 adalah gambaran *activity diagram* pada proses *preprocessing*. Pertama-tama *user* memisahkan kalimat-kalimat ke dalam 3 *class* yaitu positif, negatif, dan netral secara manual. Lalu *user* menekan tombol “*Start Processing*”, sistem akan mengubah seluruh kalimat menjadi sekumpulan token dan akan disimpan menurut *class*-nya masing-masing. Setelah proses *tokenizing* selesai, sekumpulan token yang bukan merupakan kata baku akan diubah menjadi kata baku. Pada tahap *pre-processing* ini sistem akan menghapus setiap token (kata) yang tidak memiliki makna, seperti kata penghubung. Proses terakhir pada *preprocessing* adalah *n-grams*. Sistem akan menghitung setiap frekuensi kemunculan masing-masing token pada setiap kelas, lalu hasil *preprocessing* akan disimpan dalam sistem.

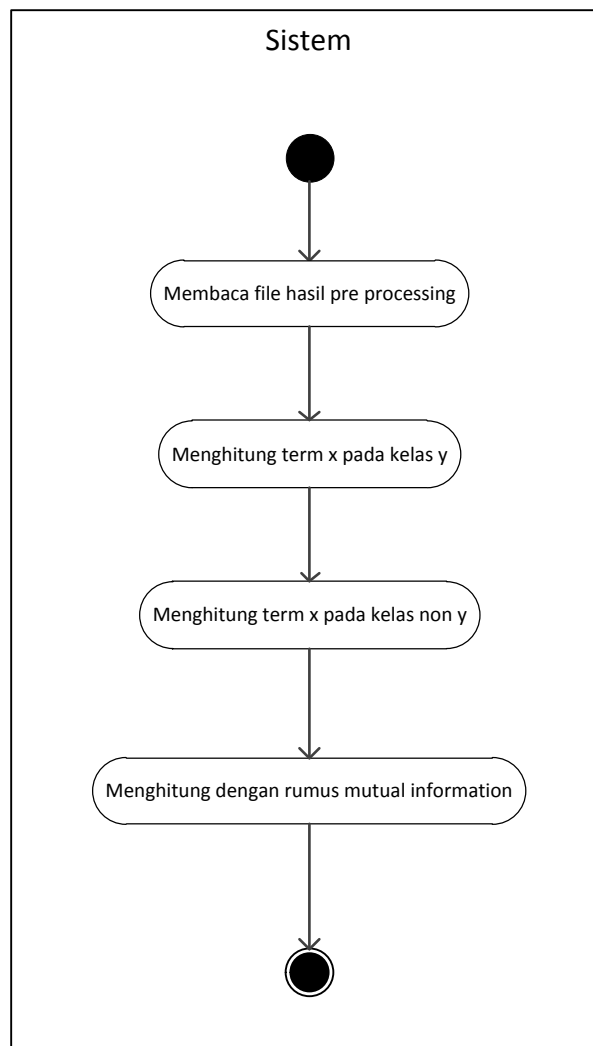
3.3.1.4.3 Activity Diagram Manual Judgment



Gambar 3.7 Activity Diagram Manual Judgment

Pada gambar 3.7 menunjukkan *activity diagram* pada proses *manual judgment*. Pertama-tama *user* masuk pada halaman “*Manual Judgment*”. *User* dapat memilih *id* data yang akan dilakukan pemilihan kategori. Sistem akan menampilkan teks sesuai *id* yang dimasukan. *User* dapat membaca teks tersebut lalu dapat memilih data tersebut masuk dalam kategori mana. Terdapat empat tombol untuk melakukan *manual judgment* diantara lain, tombol “*Positive*”, “*Negative*”, “*Netral*”, dan “*Skip*”. Tombol “*Positive*”, “*Negative*”, dan “*Netral*” digunakan untuk memilih kategori. Jika tombol ditekan maka *id* akan bertambah satu secara otomatis, lalu sistem akan menampilkan data sesuai dengan *id*.

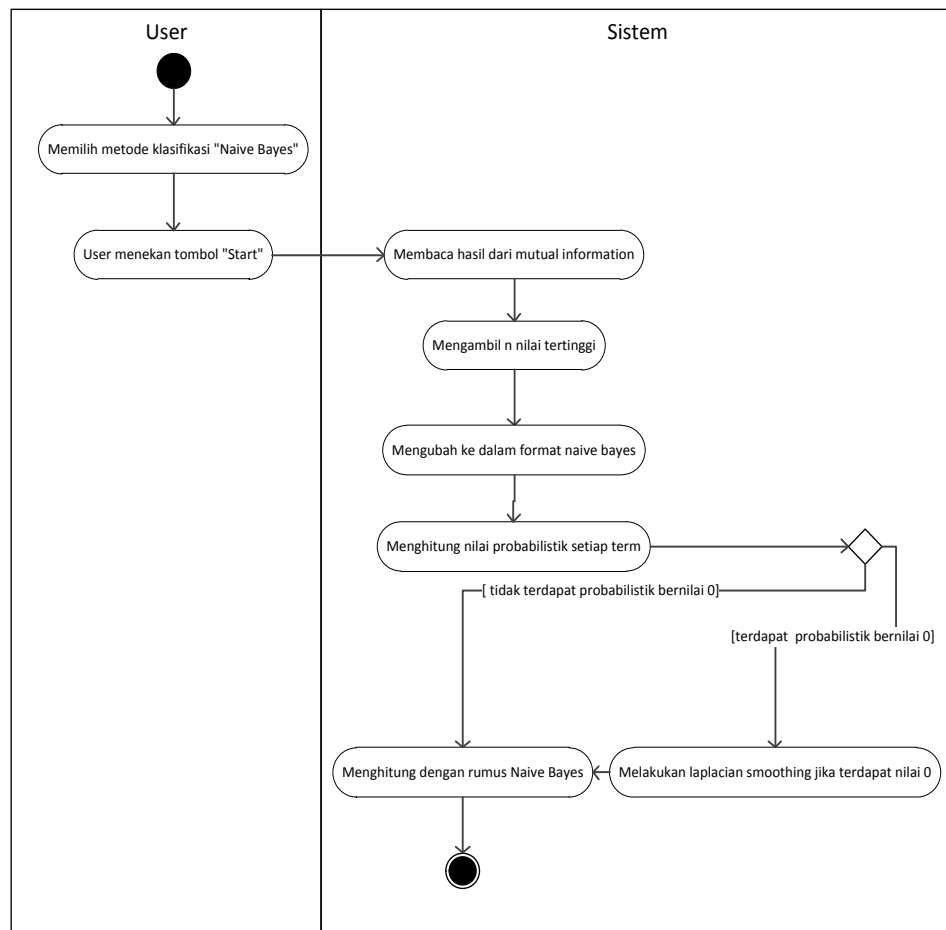
3.3.1.4.4 Activity Diagram Calculate Mutual Information



Gambar 3.8 Activity Diagram Mutual Information

Gambar 3.8 merupakan *activity diagram* pada proses *mutual information*. Setiap *term* akan dihitung akan dihitung frekuensi kemunculannya pada *class y* maupun kelas *non y*. Sistem akan menghitung setiap *term* dengan rumus *mutual information*.

3.3.1.4.5 Activity Diagram Do Classification



Gambar 3.9 Activity Diagram Do Classification

Gambar 3.9 menunjukkan *activity diagram* pada proses *training Naive Bayes*. Pertama *user* memilih *data testing*, lalu menekan tombol ‘*Start*’. Sistem akan membaca hasil dari *mutual information* pada *data training*, dan mengambil dua nilai tertinggi dari masing-masing *class*. Setiap *term* yang memiliki nilai tertinggi akan dihitung nilai probabilitas dalam format *Naive Bayes*. Jika terdapat nilai probabilitas yang bernilai 0 maka akan dilakukan penambahan pada setiap nilai,

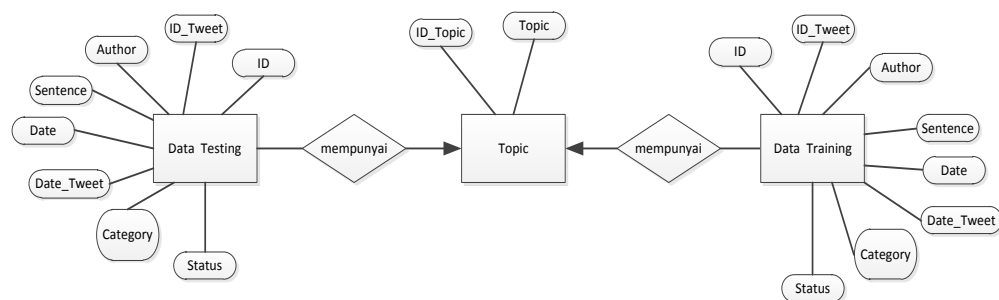
proses ini disebut dengan *laplacian smoothing*. Setelah itu akan dilakukan perhitungan dengan rumus *Naïve Bayes*.

3.3.2. Disain Penyimpanan Data

Disain penyimpanan data dijelaskan melalui ERD dan ditransformasikan dalam tabel.

3.3.2.1. Entity Relationship Diagram (ERD)

Pada sub bab ini akan menggambarkan ERD pada sistem. Gambar ERD ditunjukkan pada gambar 3.10.



Gambar 3.10 Entity Relationship Diagram (ERD)

3.3.2.2. Transformasi ERD ke dalam Tabel

Pada sub bab ini akan menjelaskan tentang transformasi ERD pada bab 3.3.2.1. Akan dijelaskan melalui tabel dengan *heading* atribut, tipe data, dan keterangan. Berikut tabel-tabel hasil transformasi ERD pada bab 3.3.2.1

3.3.2.2.1 Tabel Topic

Tabel 3.19 merupakan hasil dari transformasi ERD pada tabel *topic*. Tabel *topic* memiliki atribut “*ID_topic*” sebagai *primary key* dan “*topic*”. Tipe data setiap atribut bisa dilihat pada tabel 3.19.

Tabel 3.19 Tabel Topic

Atribut	Tipe Data	Keterangan
ID_Topic	int	<i>Primary Key</i>
Topic	Nvarchar(50)	

3.3.2.2.2 Tabel Data Training

Tabel 3.20 merupakan hasil dari transformasi ERD pada tabel *tweet*. Tabel *data training* memiliki atribut “*ID*” sebagai *primary key*, “*ID_Tweet*”, “*Author*”,

“Text”, “Date”, “Date_Tweet”, “Category”, “Status”, dan “ID_Topic” sebagai *foreign key* dari tabel “topic”. Tipe data setiap atribut bisa dilihat pada tabel 3.20.

Tabel 3.20 Tabel Data Training

Atribut	Tipe Data	Keterangan
ID	int	<i>Primary Key</i>
ID_Tweet	bigint	
Author	nvarchar(50)	
Text	nvarchar(200)	
Date	datetime	
Date_Tweet	datetime	
Category	varchar(10)	
Status	varchar(5)	
ID_Topic	int	<i>Foreign key</i>

3.3.2.2.3 Tabel Data Testing

Tabel 3.21 merupakan hasil dari transformasi ERD pada tabel *tweet*. Tabel *data testing* memiliki atribut “ID” sebagai *primary key*, “ID_Tweet”, “Author”, “Text”, “Date”, “Date_Tweet”, “Category”, “Status”, dan “ID_Topic” sebagai *foreign key* dari tabel “topic”. Tipe data setiap atribut bisa dilihat pada tabel 3.21.

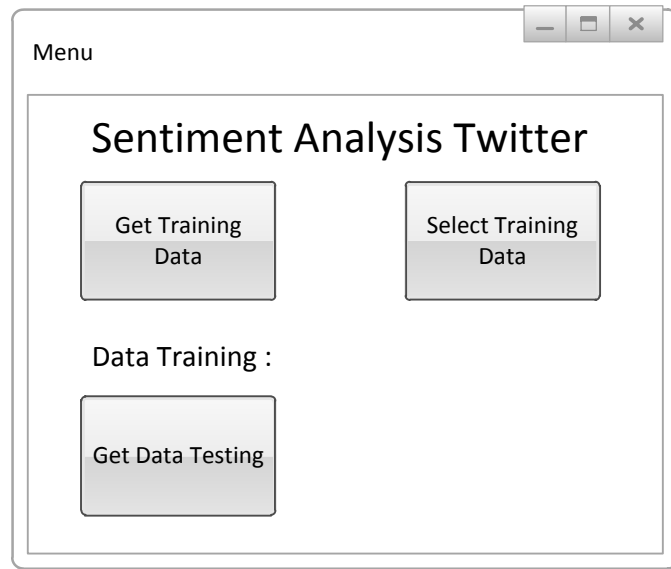
Tabel 3.21 Tabel Data Testing

Atribut	Tipe Data	Keterangan
ID	int	<i>Primary Key</i>
ID_Tweet	bigint	
Author	nvarchar(50)	
Text	nvarchar(200)	
Date	datetime	
Date_Tweet	datetime	
Category	varchar(10)	
Status	varchar(5)	
ID_Topic	int	<i>Foreign key</i>

3.3.3. Disain Antarmuka

Berikut ini adalah disain antarmuka pada sistem analisis sentimen.

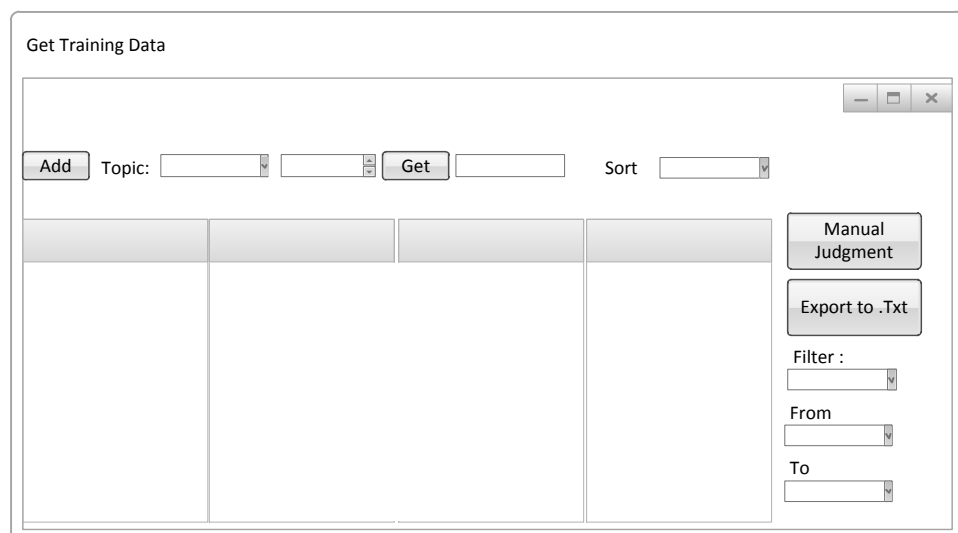
3.3.3.1. Rancangan Halaman Menu



Gambar 3.11 Rancangan Halaman Menu

Gambar 3.11 merupakan rancangan pada halaman “*Menu*”. Terdapat 3 buah *button* yaitu, “*Get Training Data*”, “*Select Data Training*”, dan “*Get Data Testing*”.

3.3.3.2. Rancangan Halaman Get Training Data

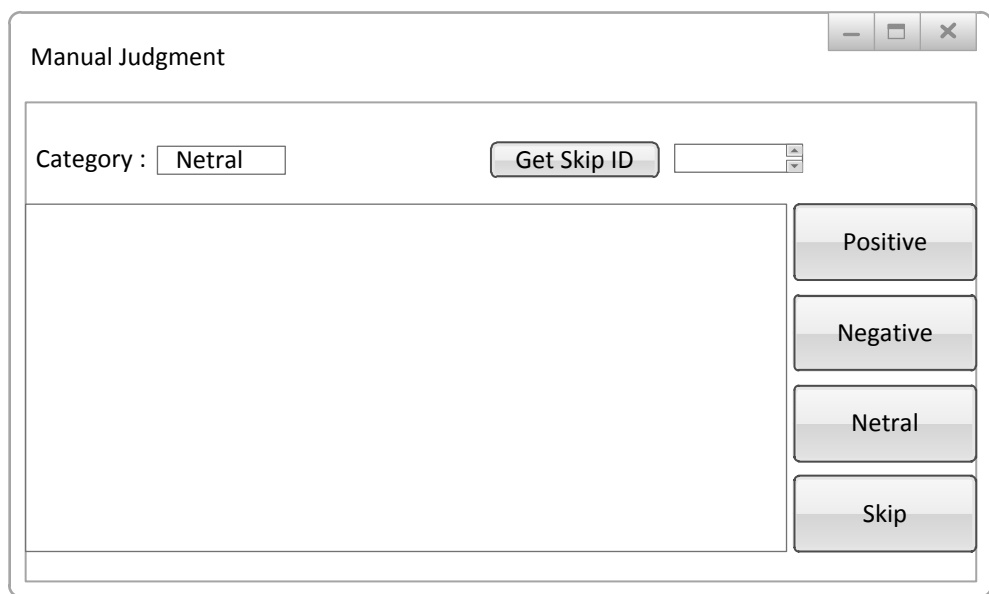


Gambar 3.12 Gambar Rancangan Get Training Data

Gambar 3.12 merupakan rancangan disain pada halaman “*Get Tweets*”. Halaman ini merupakan halaman untuk mengambil Tweets. Pada bagian atas halaman terdapat “*numeric up down*” untuk menentukan jumlah data yang dipilih dan sebuah *combo box* yang digunakan untuk memilih *topic* apa yang akan dipilih. Selain itu ada “*combo box sort*” yang digunakan untuk mengurutkan data sesuai kata yang *user* pilih. Dalam *combo box* terdapat kata kunci “*ID*”, “*Author*”, “*Text*”, “*Date*”, dan “*Category*”.

Di bagian tengah halaman terdapat “*data grid view Tweet*” untuk melihat data-data yang sudah tersimpan dalam *database*. Terdapat tujuh buah kolom yaitu, “*ID*”, “*ID_Tweet*”, “*Author*”, “*Text*”, “*Date*”, “*Category*”, dan “*Status*”. Di sebelah terdapat dua buah tombol, yang pertama tombol “*Manual Judgment*” yang berfungsi untuk masuk pada halaman “*Manual Judgment*”. Yang kedua tombol “*Export to .Txt*” yang berfungsi untuk *export* data yang ada dalam *database* ke dalam sebuah *file .txt* yang nantinya akan digunakan dalam pelatihan *data training*.

3.3.3.3. Rancangan Halaman Manual Judgment

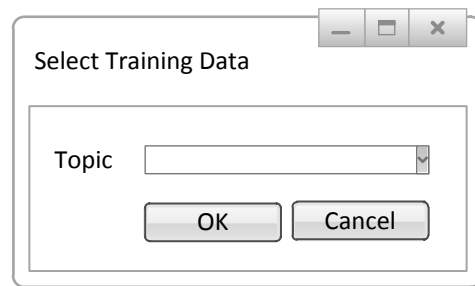


Gambar 3.13 Rancangan Halaman Manual Judgment

Gambar 3.13 merupakan rancangan disain pada halaman “*Manual Judgment*”. Sistem akan menampilkan data pada *text box* yang berada ditengah sesuai dengan *id* yang dimasukan oleh *user*. *Numeric up down* menu berguna untuk memasukan, menambah, atau mengurangi *id*. Terdapat empat buah tombol pada

sebelah kanan halaman. Tombol “*Positive*” berguna untuk mengubah kategori menjadi kategori positif, sedangkan tombol “*Negative*” dan “*Netral*” secara berurutan berguna untuk merubah kategori menjadi negatif dan netral. Pada sebelah kiri atas halaman terdapat sebuah *text box* yang berguna untuk meninformasikan kategori data. Tombol “*Skip ID*” berfungsi untuk mencari *id* mana yang belum terklasifikasi secara manual.

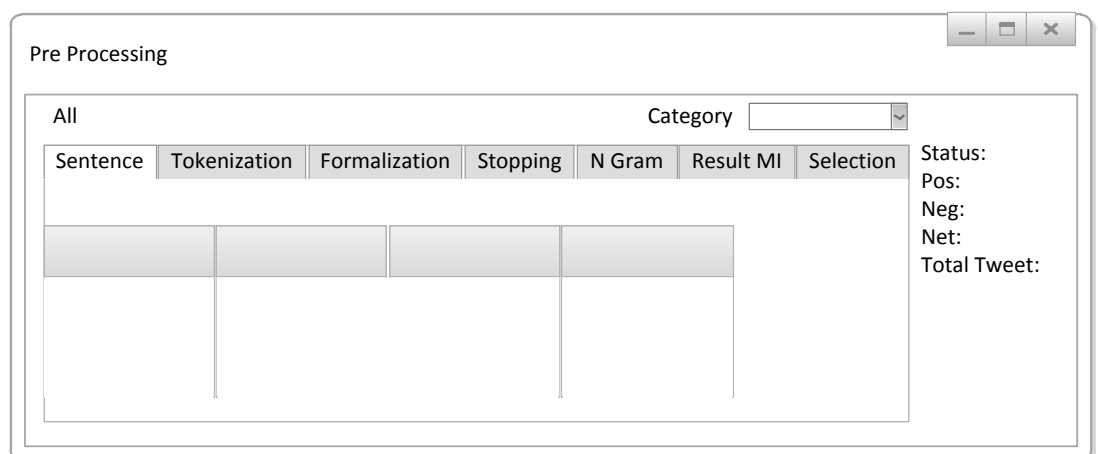
3.3.3.4. Rancangan Halaman Select Training Data



Gambar 3.14 Rancangan Halaman Select Training Data

Gambar 3.14 merupakan rancangan disain pada halaman “*Select Training Data*”. Pada halaman ini terdapat sebuah *combo box* yang berisi seluruh topik yang terdapat pada aplikasi. Terdapat dua buah *button* yaitu “*OK*” dan “*Cancel*”.

3.3.3.5. Rancangan Halaman Pre Processing



Gambar 3.15 Rancangan Halaman Pre Processing

Gambar 3.15 merupakan rancangan disain pada halaman “*Pre Processing*”. Pada halaman ini terdapat tujuh buah *data gridview* yaitu “*Sentence*”,

“*Tokenization*”, “*Formalization*”, “*Stopping*”, “*N Gram*”, “*Result MI*”, dan “*Selection*”. Masing-masing *data gridview* menampilkan hasil dari setiap tahap *pre processing* sampai tahap *selection*.

Terdapat sebuah *combo box* “*Category*” yang berisi kata “*All*”, “*Positive*”, “*Negative*”, dan “*Netral*”. *Combo box* ini berfungsi untuk menyaring data yang ditampilkan oleh sistem. Pada sebuah kanan halaman terdapat *label* yang akan memberikan informasi terhadap *user* tentang berapa banyak *tweets* dari masing kategori.

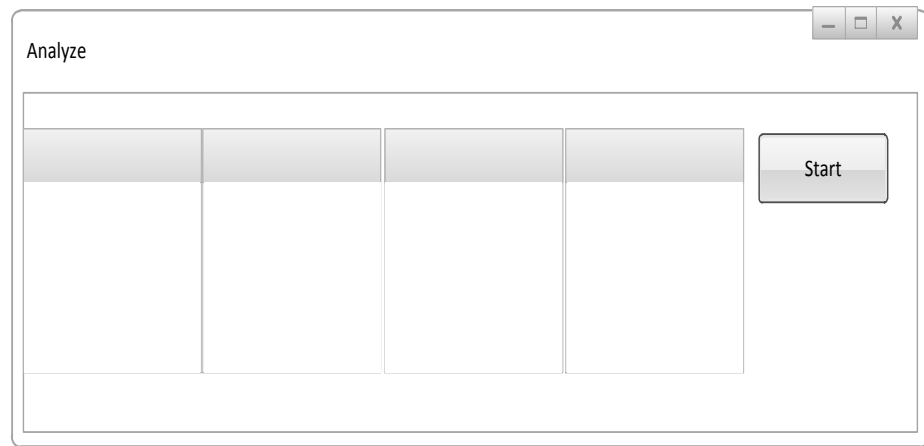
3.3.3.6. Rancangan Halaman Get Testing Data

The screenshot shows a web application window titled "Get Testing Data". At the top, there is a "Topic:" label followed by a dropdown menu, a "Get" button, and a "Sort" dropdown menu. Below this is a table with four columns and one row. To the right of the table are two buttons: "Analyze" and "Stream Tweet".

Gambar 3.16 Rancangan Halaman Get Testing Data

Pada gambar 3.16 merupakan rancangan disain pada halaman “*Get Testing Data*”. Halaman ini hampir mirip dengan halaman “*Get Tweet*”. Bedanya halaman ini digunakan untuk mengambil data dari Twitter untuk *data testing*. Sedangkan halaman “*Get Tweet*” digunakan untuk mengambil data dari Twitter yang digunakan untuk *data training*. Terdapat tombol “*Analyze*” dan “*Stream Tweet*” pada sebelah kanan halaman. Tombol “*Analyze*” berguna untuk masuk ke halaman “*Analyze*” sedangkan tombol “*Stream Tweet*” berguna untuk masuk ke halaman “*Stream Tweet*”.

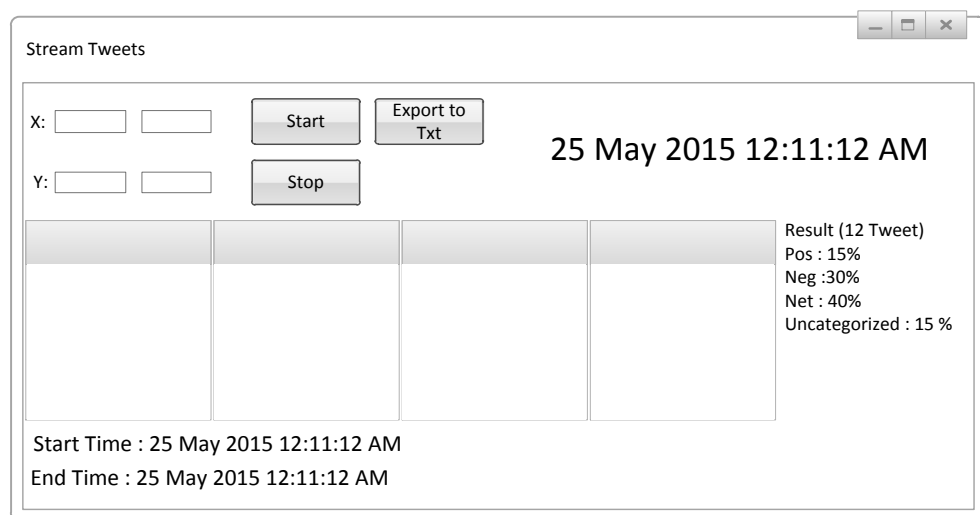
3.3.3.7. Rancangan Halaman Analyze



Gambar 3.17 Rancangan Halaman Analyze

Pada gambar 3.17 merupakan rancangan disain pada halaman “*Analyze*”. Halaman ini digunakan untuk melihat hasil *data testing* yang diambil secara manual pada halaman “*Get Data Testing*”. Terdapat sebuah *data gridview* yang memiliki kolom “*Sentence*” dan “*Result*”. Sistem akan menampilkan hasil dari masing-masing kalimat jika tombol “*Start*” sudah ditekan.

3.3.3.8. Rancangan Halaman Streaming Tweets



Gambar 3.18 Rancangan Halaman Streaming Tweet

Gambar 3.18 merupakan rancangan disain pada halaman “*Stream Tweets*”. Terdapat 4 buah *text box*, 2 buah *text box* untuk bagian “X” dan 2 buah *text box*

untuk bagian “Y”. Sistem akan mengambil data dari Twitter sesuai kata yang dimasukkan pada *text box*, setelah *user* menekan tombol “*Start*”. Pada halaman ini terdapat sebuah *data gridview* yang memiliki kolom “*Sentence*” dan “*Result*”. *Data gridview* ini akan menampilkan data yang diambil dari Twitter beserta hasil klasifikasi secara otomatis. Tombol “*Stop*” digunakan apabila *user* ingin memberhentikan sistem untuk mengambil data. Setelah tombol “*Stop*” ditekan akan muncul persentase hasil *stream* pada sebelah kanan halaman.