BAB III
METODOLOGI PENELITIAN

3.1 Road Map Penelitian

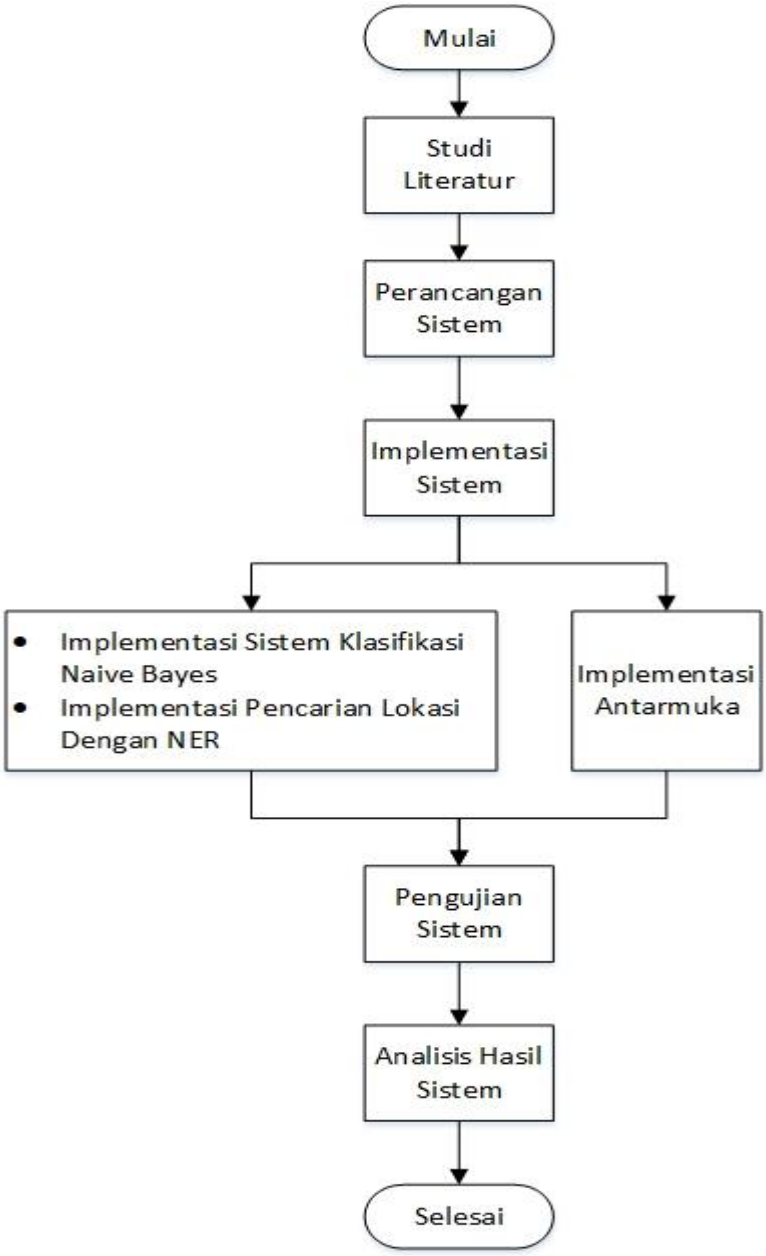
Road map penelitian merupakan gambaran yang dibuat oleh penulis untuk menjelaskan keseluruhan aktivitas yang dapat dikerjakan dalam membangun sebuah sistem klasifikasi dan deteksi lokasi kasus demam berdarah di Indonesia. Pada penelitian kali ini penulis akan membangun sistem klasifikasi dan deteksi lokasi kasus demam berdarah di Indonesia pada tahap pertama yaitu melakukan klasifikasi kasus demam berdarah dan menemukan lokasi yang memiliki kasus demam berdarah dan kemudian ditampilkan pada peta dalam bentuk *marker* di tiap lokasinya. Penelitian ini merupakan penelitian pertama yang dikerjakan oleh penulis sehingga belum adanya aktivitas pembangunan sistem selanjutnya seperti penjumlahan kasus demam berdarah di tiap lokasi, dan membuat sistem informasi geografis persebaran demam berdarah di Indonesia. Berikut *road map* dari penelitian dapat dilihat pada Gambar 3.1.

Gambar 3.1 *Road map* penelitian

3.2 Alur Penelitian

Penelitian yang baik adalah penelitian yang mempunyai alur pada penelitian agar proses implementasi dapat dilakukan dengan baik. Oleh karena itu diperlukannya sebuah diagram alur penelitian untuk menjadi pedoman dalam melakukan tahapan-tahapan yang ada pada penelitian. Diagram alur penelitian memiliki tahapan kerja yang tersusun secara sistematis sehingga hasil yang didapatkan dari penelitian dapat memiliki hasil sesuai dengan tujuan dari penelitian yang dibuat. Berikut merupakan

diagram alur penelitian yang akan dilakukan dalam penyusunan tugas akhir yang dapat dilihat pada Gambar 3.2.



Gambar 3.2 Diagram alur penelitian

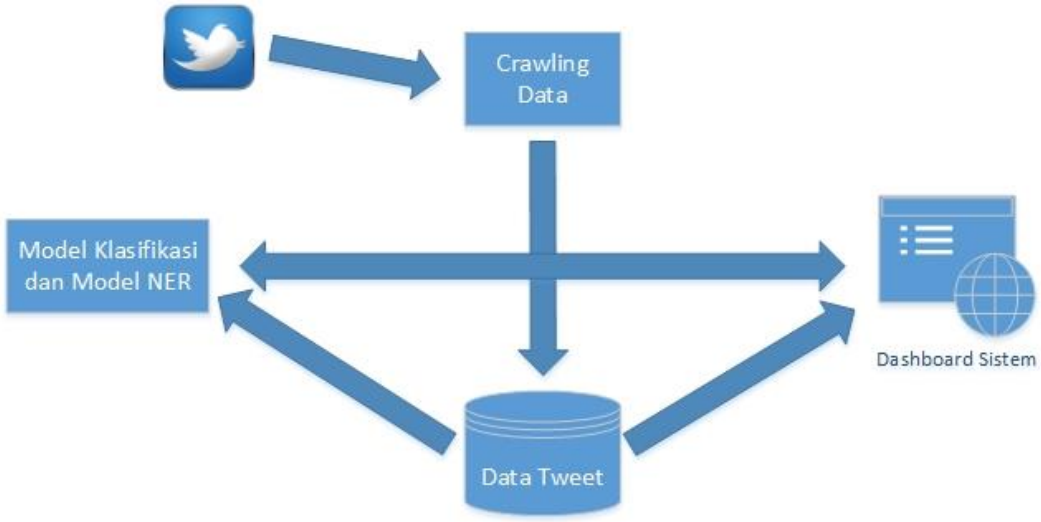
3.3 Studi Pustaka

Tahapan ini dilakukan penulis dalam mengumpulkan berbagai informasi dari jurnal, penelitian terkait, dan buku untuk menunjang penelitian yang sedang dilakukan oleh penulis yaitu klasifikasi dan deteksi lokasi kasus demam berdarah

di Indonesia dengan menggunakan data Twitter dan juga untuk menentukan metode yang akan digunakan dalam penelitian. Informasi yang menunjang penelitian akan digunakan sebagai sumber rujukan awal penulis dalam mengerjakan penelitian. Dari hasil studi pustaka yang dilakukan penulis maka didapatkan metode Naïve Bayes untuk mengklasifikasikan kasus penyakit demam berdarah di Indonesia dengan menggunakan data Twitter dan juga metode Named Entity Recognition untuk melakukan deteksi lokasi yang memiliki kasus demam berdarah.

3.4 Gambaran Umum Sistem

Sistem yang akan dibangun oleh penulis pada penelitian ini yaitu klasifikasi dan deteksi lokasi kasus demam berdarah di Indonesia berdasarkan data *tweet* dengan menggunakan metode Naïve Bayes dan Named Entity Recognition. Berikut gambaran umum sistem dapat dilihat pada Gambar 3.3.



Gambar 3.3 Gambaran umum sistem

Proses dari sistem dimulai dari pengambilan *tweet* dengan cara *crawling* yaitu melakukan pengumpulan data berdasarkan kata kunci dengan bantuan *tools* dan API Twitter. Data *tweets* yang sudah terkumpul kemudian akan digunakan sebagai data *train* dan juga data *test*. Data *train* akan digunakan untuk pelatihan model klasifikasi dan model NER. Setelah model berhasil didapatkan maka akan dilakukan pengujian untuk melihat kinerja model klasifikasi dan model deteksi

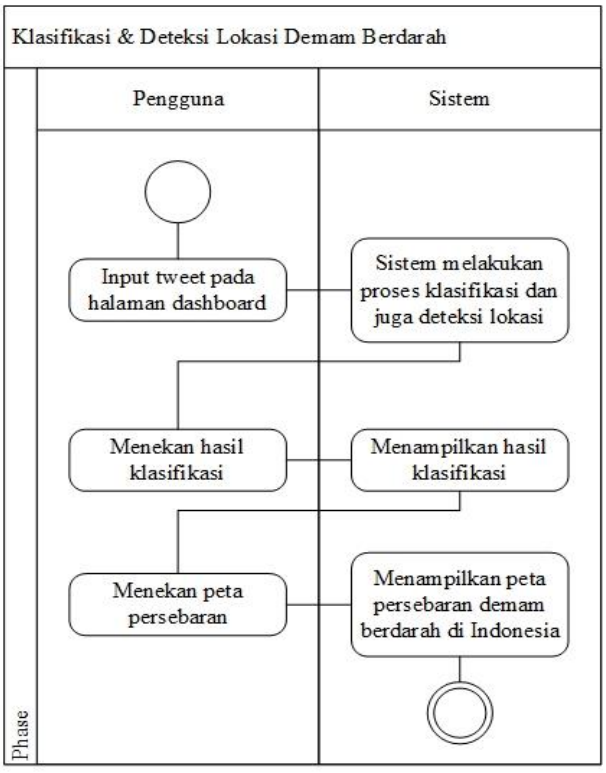
lokasi. Setelah didapatkan kinerja model yang baik maka selanjutnya model akan digunakan pada aplikasi untuk melakukan klasifikasi pada *tweet* dan deteksi lokasi pada *tweet* yang dinyatakan sebagai kasus dbd kemudian ditampilkan ke dalam bentuk peta.

3.5 Perancangan Sistem

Pada tahap ini penulis akan menjelaskan fungsi dari sistem dan juga proses yang akan dilakukan penulis pada tahap implementasi sistem. Berikut perancangan sistem yang akan dibangun oleh penulis.

3.5.1 Activity Diagram

Pada sistem yang dibuat aktivitas yang dapat dilakukan pengguna adalah melakukan input tweet dan kemudian pengguna dapat melihat hasil klasifikasi dan deteksi lokasi kasus demam berdarah berdasarkan input pengguna. Berikut diagram aktivitas yang dilakukan pengguna dapat dilihat pada Gambar 3.4.



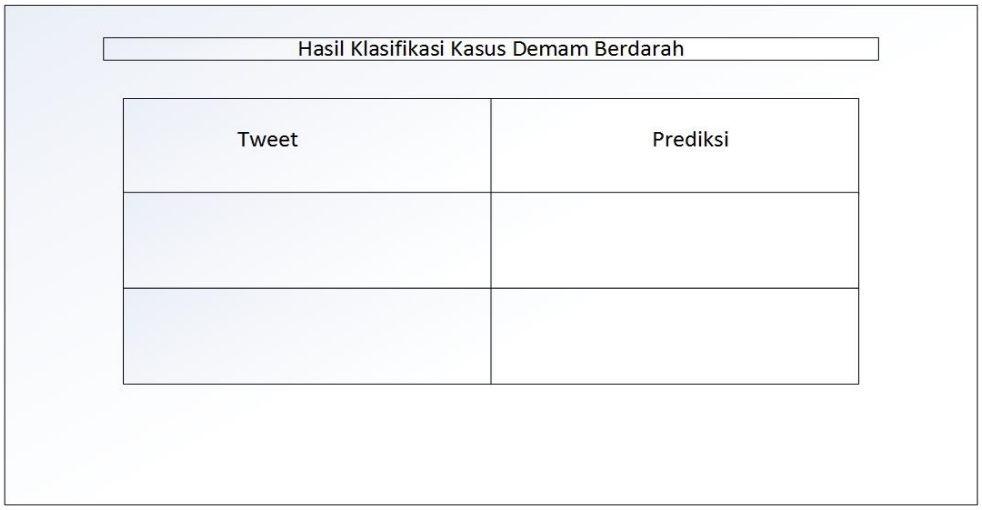
Gambar 3.4 *Activity* diagram

3.5.2 Perancangan Antarmuka

Pada tahap ini penulis akan merancang antarmuka dari sistem klasifikasi dan deteksi lokasi kasus demam berdarah di Indonesia dengan tujuan memudahkan dalam segi penggunaan oleh pengguna (*user*). Antar muka akan dibangun menjadi sebuah halaman website. Antar muka sistem terbagi menjadi 3 yaitu halaman utama, halaman hasil klasifikasi dan halaman persebaran demam berdarah di Indonesia.

Gambar 3.5 Tampilan halaman utama

Pada Gambar 3.5 adalah desain halaman utama yang berguna sebagai tempat pengguna melakukan input tweet yang akan diklasifikasi dan deteksi. Pada halaman utama terdapat *textbox* yang berfungsi sebagai tempat pengguna melakukan *input tweet*. Pada halaman utama terdapat 3 *button* diantaranya berguna untuk menjalankan proses klasifikasi dan deteksi lokasi, berguna untuk menampilkan hasil klasifikasi yang ada pada Gambar 3.6, dan berguna untuk menampilkan peta persebaran demam berdarah di Indonesia yang ada pada Gambar 3.7.

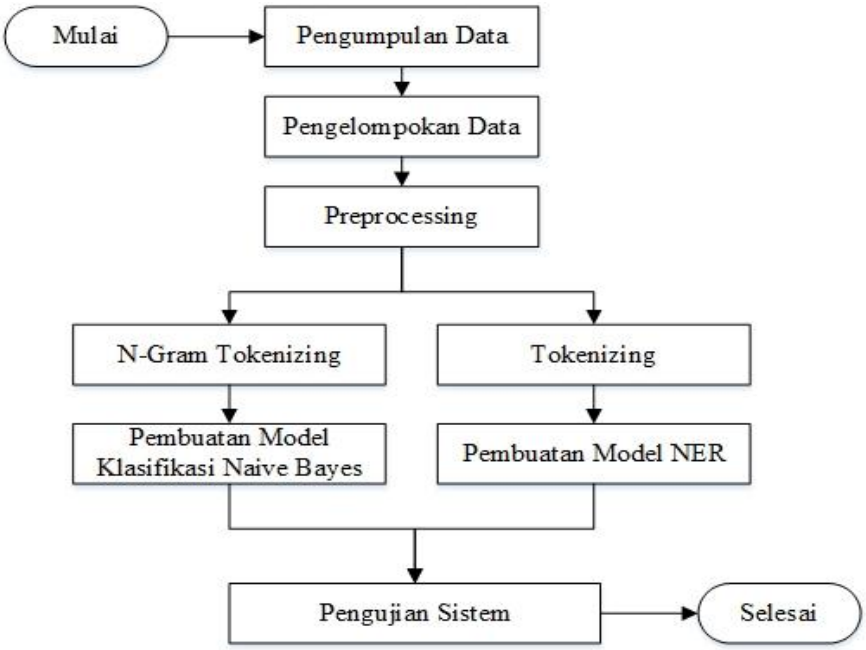


Gambar 3.6 Tampilan halaman hasil klasifikasi

Gambar 3.7 Tampilan Halaman Peta Persebaran Demam Berdarah di Indonesia

3.5.3 Perancangan Sistem Klasifikasi dan Pencarian Lokasi

Pada penelitian yang dibuat oleh penulis terdapat 2 proses yang akan dilakukan oleh sistem yaitu menemukan *tweet* berlabel kasus dengan melakukan klasifikasi dan juga melakukan deteksi lokasi dari *tweet* dengan melakukan ekstraksi informasi. Berikut rancangan sistem dapat dilihat pada Gambar 3.8.



Gambar 3.8 Flowchart rancangan sistem klasifikasi dan pencarian lokasi

Dari Gambar 3.8 dapat dilihat bahwa pada penelitian ini akan digunakan 2 metode dengan fungsi yang berbeda. Pada metode Naive Bayes akan dibuat model yang kemudian digunakan untuk melakukan klasifikasi *tweet* yang menyatakan kasus demam berdarah atau bukan kasus demam berdarah. Pada metode Named Entity Recognition akan dibuat model yang digunakan untuk melakukan deteksi lokasi dari *tweet* yang diklasifikasikan sebagai kasus demam berdarah.

3.5.4.1 Pengumpulan Data

Pada tahap ini akan dilakukan proses pengambilan data yang dibutuhkan untuk mengerjakan penelitian ini. Data yang digunakan adalah *tweet* berbahasa Indonesia yang diambil dari Twitter yang membahas tentang kasus demam berdarah di Indonesia. Proses pengambilan data dilakukan secara periodik yaitu setiap 1 minggu sekali dengan memanfaatkan API Twitter. Kata kunci yang digunakan dalam pengambilan data yaitu ”Demam Berdarah”, “DBD”, dan “Dengue”. Setelah data yang dibutuhkan telah didapatkan maka selanjutnya akan dilakukan pemberian label yang dilakukan secara manual oleh penulis sesuai dengan kelas yang telah

ditentukan. Adapun contoh data *tweet* yang telah diambil dan juga diberikan label dapat dilihat pada Tabel 3.1.

Tabel 3.1 Contoh data Twitter

No	Tweets	Label
1	Kasus DBD Meningkat di Karimun https://t.co/qUglQ85r8m #Kepri #Batam #Berita	Kasus
2	Di tengah pandemi kasus DBD meningkat https://t.co/1TIOSnR7Rl	Bukan Kasus
3	Di tengah pandemi, DBD menjadi perhatian. https://t.co/24ta2vLFbm	Bukan Kasus
4	Kasus DBD Meningkat di Pasaman https://t.co/Jcmtte1QHI	Kasus

3.5.4.2 Pengelompokan Data

Data *tweet* yang sudah dikumpulkan kemudian dikelompokkan menjadi 2 bagian yaitu menjadi data *train* yang akan digunakan dalam membuat model klasifikasi dan model deteksi lokasi dan juga data *test* yang akan digunakan untuk menguji model klasifikasi dan deteksi lokasi yang sudah dibuat.

3.5.4.3 Preprocessing

A. Cleaning

Pada proses ini akan dilakukan penghapusan kata – kata yang tidak diperlukan dalam sebuah *tweet* yang akan digunakan dalam proses klasifikasi dan deteksi lokasi. Kata – kata yang akan dihilangkan adalah *hashtag, url, mention, username,* dan *emoticon* dengan tujuan untuk mengurangi frekuensi kemunculan kata yang tidak ada hubungannya dalam klasifikasi pada *tweet* dan deteksi lokasi pada *tweet.* Proses *cleaning* dilakukan penulis secara otomatis dengan memanfaatkan *tools*. Berikut contoh *cleaning* dapat dilihat pada Tabel 3.2.

Tabel 3.2 Contoh cleaning

Tweet Sebelum *Cleaning*	Tweet Sesudah *Cleaning*
Kasus DBD Meningkat di Karimun https://t.co/qUglQ85r8m #Kepri #Batam #Berita	Kasus DBD Meningkat di Karimun

B. Case Folding

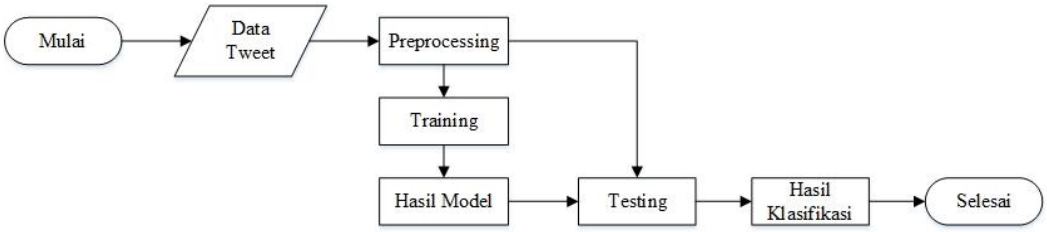
Pada proses ini akan dilakukan pengubahan teks pada *tweet* kedalam bentuk huruf kecil atau sebaliknya karena tidak semua *tweet* yang di posting menggunakan huruf yang sama. Oleh karena itu peran dari case folding dibutuhkan agar keseluruhan teks dalam sebuah data *tweet* menjadi suatu bentuk standar dan menghindari terjadinya kemunculan kata yang terlalu banyak dikarenakan ada kata yang sama tetapi dikarenakan bentuk huruf yang berbeda dan kecilnya nilai frekuensi kemunculan tiap kata. Berikut contoh *case folding* dapat dilihat pada Tabel 3.3.

Tabel 3.3 Contoh case folding

Tweet Sebelum *Case Folding*	Tweet Sesudah *Case Folding*
Kasus DBD Meningkat di Karimun	kasus dbd meningkat di karimun

3.5.4.4 Perancangan Klasifikasi Naïve Bayes

Pada tahap ini penulis akan menjelaskan proses klasifikasi untuk menemukan *tweet* dengan kelas kasus demam berdarah menggunakan metode Naïve Bayes. Pada klasifikasi menggunakan metode Naïve Bayes akan dilakukan dua proses tahapan yaitu *training* dan *testing*. Pada tahap *training* akan dilakukan pelatihan untuk menghasilkan sebuah model klasifikasi. Pada tahap *testing* akan dilakukan pengujian model klasifikasi yang didapatkan pada tahap *training*. Hasil pada tahap ini adalah penentuan klasifikasi kelas pada *tweet* termasuk dalam kelas kasus demam berdarah atau bukan kasus demam berdarah. Berikut *flowchart* proses klasifikasi dengan menggunakan algoritma Naïve Bayes dapat dilihat pada Gambar 3.9.



Gambar 3.9 Flowchart proses klasifikasi naive bayes

A. Preprocessing

Pada tahapan ini akan dilakukan *preprocessing* pada dataset agar data dapat diterima dalam format yang tepat untuk diproses dengan menggunakan metode Naïve Bayes. Pada proses ini akan dilakukan proses *tokenizing* dengan memanfaatkan fitur n-gram yaitu pemisahan atau pemecahan kalimat *tweet* menjadi potongan kata dengan menerapkan *unigram* dan *bigram*. Berikut contoh n-gram *tokenizing* dapat dilihat pada Tabel 3.4 dan Tabel 3.5.

Tabel 3.4 Contoh *tokenizing* dengan *unigram*

Tweet Sebelum *Tokenizing*	**Tweet Sesudah *Tokenizing***
kasus dbd meningkat di karimun	kasus, dbd, meningkat, di, karimun

Tabel 3.5 Contoh *tokenizing* dengan *bigram*

Tweet Sebelum *Tokenizing*	**Tweet Sesudah *Tokenizing***
kasus dbd meningkat di karimun	kasus dbd, dbd meningkat, meningkat di, di karimun

B. Training

Pada tahapan ini data yang telah melewati proses *preprocessing* akan digunakan sebagai data *training* (latih) untuk tahap *training*. *Training* merupakan sebuah tahapan dalam pembentukan sebuah model dengan cara mengumpulkan pengetahuan dari data latih yang digunakan. Proses *training* akan dilakukan dengan menggunakan pemrograman *python* dan akan dilakukan dengan menggunakan 2 skenario. Berikut proses *training* dari tiap skenario :

1. Training Skenario 1

Pada skenario 1 akan dilakukan proses *training* dengan menggunakan dataset yang telah melalui proses *preprocessing* dengan menerapkan fitur *unigram.* Berikut merupakan contoh bentuk data latih berjumlah 4 dengan 2 kelas yaitu kelas kasus dan bukan kasus yang telah melewati proses *preprocessing* dengan fitur *unigram* dan telah dilakukan pembobotan dapat dilihat pada Tabel 3.6.

Tabel 3.6 Contoh data latih bentuk *unigram*

Kata	***tf(K)***	***tf(BK)***
kasus	2	1
dbd	2	2
meningkat	2	1
di	2	2
karimun	1	0
pasaman	1	0
tengah	0	2
pandemi	0	2
menjadi	0	1
perhatian	0	1

Keterangan :

tf(K) : *Term Frequency* (frekuensi kata) pada kelas kasus.

tf(BK) : *Term Frequency* (frekuensi kata) pada kelas bukan kasus.

Setelah data latih pada tabel 3.6 telah didapatkan maka selanjutnya adalah menghitung besar nilai probabilitas dari tiap kelas dan juga nilai probabilitas dari setiap kata yang muncul pada masing-masing kelas. Berikut perhitungan yang akan dilakukan :

Perhitungan nilai probabilitas untuk tiap kelas :

P (K) $= \frac{2}{4} = 0{,}5$

P (BK) $= \frac{2}{4} = 0{,}5$

Perhitungan nilai probabilitas terhadap kelas kasus dari tiap kata :

P (kasus | K) $= \frac{2+1}{10+10} = 0,15$

P (dbd | K) $= \frac{2+1}{10+10} = 0,15$

P (meningkat | K) $= \frac{2+1}{10+10} = 0,15$

P (di | K) $= \frac{2+1}{10+10} = 0,15$

P (karimun | K) $= \frac{1+1}{10+10} = 0,1$

P (pasaman | K) $= \frac{1+1}{10+10} = 0,1$

P (tengah | K) $= \frac{0+1}{10+10} = 0,05$

P (pandemi | K) $= \frac{0+1}{10+10} = 0,05$

P (menjadi | K) $= \frac{0+1}{10+10} = 0,05$

P (perhatian | K) $= \frac{0+1}{10+10} = 0,05$

Perhitungan nilai probabilitas terhadap kelas bukan kasus dari tiap kata :

P (kasus | BK) $= \frac{1+1}{12+10} = 0,0909$

P (dbd | BK) $= \frac{2+1}{12+10} = 0,1363$

P (meningkat | BK) $= \frac{1+1}{12+10} = 0,0909$

P (di | BK) $= \frac{2+1}{12+10} = 0,1363$

P (karimun | BK) $= \frac{0+1}{12+10} = 0,0454$

P (pasaman | BK) $= \frac{0+1}{12+10} = 0,0454$

P (tengah | BK) $= \frac{2+1}{12+10} = 0,1363$

P (pandemi | BK) $= \frac{2+1}{12+10} = 0,1363$

P (menjadi | BK) $= \frac{1+1}{12+10} = 0,0909$

P (perhatian | BK) $= \frac{1+1}{12+10} = 0,0909$

2. Training Skenario 2

Pada skenario 2 akan dilakukan proses *training* dengan menggunakan dataset yang telah melalui proses *preprocessing* dengan menerapkan fitur *bigram.* Berikut merupakan contoh bentuk data latih berjumlah 4 dengan 2 kelas yaitu kelas kasus dan bukan kasus yang telah melewati proses *preprocessing* dengan fitur *bigram* dan telah dilakukan pembobotan dapat dilihat pada Tabel 3.7.

Tabel 3.7 Contoh data latih bentuk bigram

Kata	*tf(K)*	*tf(BK)*
kasus dbd	2	1
dbd meningkat	2	1
meningkat di	2	0
di karimun	1	0
di pasaman	1	0
di tengah	0	2
tengah pandemi	0	2
dbd menjadi	0	1
menjadi perhatian	0	1
pandemi kasus	0	1

Keterangan :

tf(K) : *Term Frequency* (frekuensi kata) pada kelas kasus.

tf(BK) : *Term Frequency* (frekuensi kata) pada kelas bukan kasus.

Setelah data latih pada Tabel 3.7 telah didapatkan maka selanjutnya adalah menghitung besar nilai probabilitas dari tiap kelas dan juga nilai probabilitas dari setiap kata yang muncul pada masing-masing kelas. Berikut perhitungan yang akan dilakukan :

Perhitungan nilai probabilitas untuk tiap kelas :

P (K) $= \frac{2}{4} = 0{,}5$

P (BK) $= \frac{2}{4} = 0{,}5$

Perhitungan nilai probabilitas terhadap kelas kasus dari tiap kata :

P (kasus dbd| K) $= \frac{2+1}{8+10} = 0{,}166$

P (dbd meningkat| K) $= \frac{2+1}{8+10} = 0{,}166$

P (meningkat di| K) $= \frac{2+1}{8+10} = 0{,}166$

P (di karimun| K) $= \frac{1+1}{8+10} = 0{,}111$

P (di pasaman | K) $= \frac{1+1}{8+10} = 0{,}111$

P (di tengah | K) $= \frac{0+1}{8+10} = 0{,}055$

P (tengah pandemi | K) $= \frac{0+1}{8+10} = 0{,}055$

P (dbd menjadi | K) $= \frac{0+1}{8+10} = 0{,}055$

P (menjadi perhatian | K) $= \frac{0+1}{8+10} = 0{,}055$

P (pandemi kasus | K) $= \frac{0+1}{8+10} = 0{,}055$

Perhitungan nilai probabilitas terhadap kelas bukan kasus dari tiap kata :

P (kasus dbd| K) $= \frac{1+1}{9+10} = 0{,}105$

P (dbd meningkat| K) $= \frac{1+1}{9+10} = 0{,}105$

P (meningkat di| K) $= \frac{0+1}{9+10} = 0{,}052$

P (di karimun| K) $= \frac{0+1}{9+10} = 0{,}052$

P (di pasaman | K) $= \frac{0+1}{9+10} = 0{,}052$

P (di tengah | K) $= \frac{2+1}{9+10} = 0{,}157$

P (tengah pandemi | K) $= \frac{2+1}{9+10} = 0{,}157$

P (dbd menjadi | K) $= \frac{1+1}{9+10} = 0{,}105$

P (menjadi perhatian | K) $= \frac{1+1}{9+10} = 0{,}105$

P (pandemi kasus | K) $= \frac{1+1}{9+10} = 0{,}105$

C. Testing

Pada tahap *testing* akan dilakukan dengan menggunakan 2 skenario pengujian dengan menggunakan data yang akan diuji dengan menggunakan model yang telah didapatkan pada tahap sebelumnya yaitu pada tahap *training*. Berikut adalah data yang akan digunakan sebagai data uji (*testing*) dapat dilihat pada Tabel 3.8.

Tabel 3.8 Contoh data testing

Tweet	Kelas
Kasus DBD di kota Tasik Terus Naik	?

1. Testing Skenario Pertama

Pada skenario pertama data uji pada tabel 3.8 akan masuk ke tahap *preprocessing* yang menerapkan fitur *unigram* terlebih dahulu agar bentuk data uji sesuai dengan data latih yang telah disiapkan pada proses *training* skenario pertama. Berikut bentuk data uji setelah proses *preprocessing* dapat dilihat pada Tabel 3.9.

Tabel 3.9 Contoh data testing setelah preprocessing bentuk *unigram*

Tweet	Kelas
Kasus, dbd, di, kota, tasik, terus, naik	?

Setelah data latih yang telah melalui tahap *preprocessing* pada Tabel 3.9 didapatkan, selanjutnya akan dilakukan perhitungan nilai probabilitas dengan menggunakan algoritma Naïve Bayes untuk menentukan data uji masuk dalam kelas kasus atau bukan kasus. Proses perhitungan dilakukan dengan cara mengalikan hasil perhitungan dari nilai probabilitas tiap kelas dan juga nilai probabilitas masing-masing kata yang telah dilakukan pada tahap *training*. Berikut perhitungan yang akan dilakukan :

$$P(\text{uji}|K) = P(K) \times P(\text{kasus}|K) \times P(\text{dbd}|K) \times P(\text{di}|K) \times P(\text{kota}|K) \times P(\text{tasik}|K) \times P(\text{terus}|K) \times P(\text{naik}|K)$$

$$P(\text{uji}|K) = \frac{2}{4} \times \frac{3}{20} \times \frac{3}{20} \times \frac{3}{20} \times \frac{1}{20} \times \frac{1}{20} \times \frac{1}{20} \times \frac{1}{20} = \frac{54}{5120000000} = 0{,}00000001054$$

P (uji|BK) = P(BK) × P(kasus|BK) × P(dbd|BK) × P(di|BK) × P(kota|BK) × P(tasik|BK) × P(terus|BK) × P(naik|BK)

P (uji|BK) = $\frac{2}{4} \times \frac{2}{22} \times \frac{3}{22} \times \frac{3}{22} \times \frac{1}{22} \times \frac{1}{22} \times \frac{1}{22} \times \frac{1}{22} = \frac{36}{9977431552}$

= 0,000000003608

Dari perhitungan yang telah dilakukan didapatkan hasil bahwa data uji masuk kedalam kelas kasus dengan besar nilai probabilitas yang didapatkan sebesar 0,00000001054.

2. Testing Skenario Kedua

Pada skenario kedua data uji pada Tabel 3.8 akan masuk ke tahap *preprocessing* yang menerapkan fitur *bigram* terlebih dahulu agar bentuk data uji sesuai dengan data latih yang telah disiapkan pada proses *training* skenario kedua. Berikut bentuk data uji setelah proses *preprocessing* yang dapat dilihat pada Tabel 3.10.

Tabel 3.10 Contoh data testing setelah preprocessing bentuk bigram

Tweet	Kelas
kasus dbd. dbd di, di kota, kota tasik, tasik terus, terus naik	?

Setelah data latih yang telah melalui tahap *preprocessing* pada tabel 3.10 didapatkan, selanjutnya akan dilakukan perhitungan nilai probabilitas dengan menggunakan algoritma Naïve Bayes untuk menentukan data uji masuk dalam kelas kasus atau bukan kasus. Proses perhitungan dilakukan dengan cara mengalikan hasil perhitungan dari nilai probabilitas tiap kelas dan juga nilai probabilitas masing-masing kata yang telah dilakukan pada tahap *training*. Berikut perhitungan yang akan dilakukan :

P (uji|K) = P(K) × P(kasus dbd |K) × P(dbd di|K) × P(di kota|K) × P(kota tasik|K) × P(tasik terus|K) × P(terus naik|K)

P (uji|K) = $\frac{2}{4} \times \frac{3}{18} \times \frac{1}{18} \times \frac{1}{18} \times \frac{1}{18} \times \frac{1}{18} \times \frac{1}{18} = \frac{6}{7558272}$ = 0,000007938

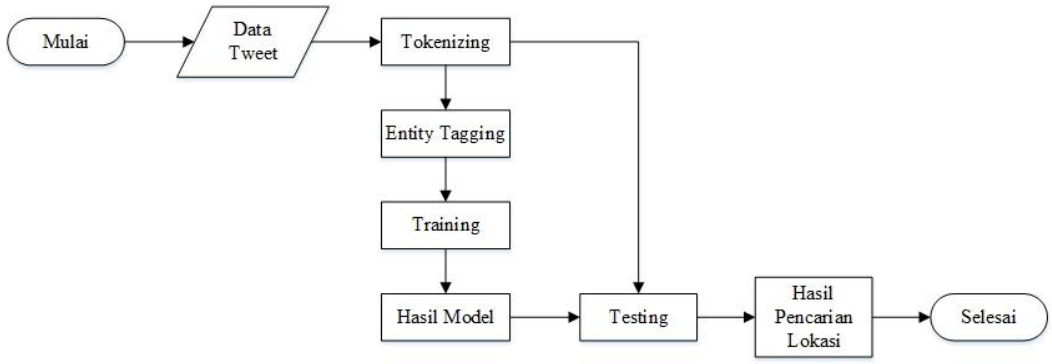
P (uji|BK) = P(BK) × P(kasus dbd|BK) × P(dbd di|BK) × P(di kota|BK) × P(kota tasik|BK) × P(tasik terus|BK) × P(terus naik|BK)

$$P(\text{uji}|\text{BK}) = \frac{2}{4} \times \frac{2}{19} \times \frac{1}{19} \times \frac{1}{19} \times \frac{1}{19} \times \frac{1}{19} \times \frac{1}{19} = \frac{4}{9904396} = 0{,}0000004038$$

Dari perhitungan yang telah dilakukan didapatkan hasil bahwa data uji masuk kedalam kelas kasus dengan besar nilai probabilitas yang didapatkan sebesar 0,000007938.

3.5.4.5 Perancangan Deteksi Lokasi Dengan NER

Pada tahapan ini penulis akan menjelaskan rancangan deteksi lokasi pada *tweet* untuk mengetahui wilayah yang memiliki kasus demam berdarah di Indonesia dengan menggunakan Named Entity Recognition. Proses yang akan dilakukan meliputi 2 proses yaitu *training* dan *testing*. Pada proses *training* akan dilakukan pelatihan untuk menghasilkan sebuah model deteksi lokasi. Pada proses *testing* akan dilakukan pengujian model yang didapatkan pada saat *training*. Hasil pada tahap ini adalah lokasi yang tercantum pada *tweet*. Berikut flowchart pencarian lokasi dengan menggunakan metode Named Entity Recognition dapat dilihat pada Gambar 3.10.



Gambar 3.10 *Flowchart* deteksi lokasi dengan NER

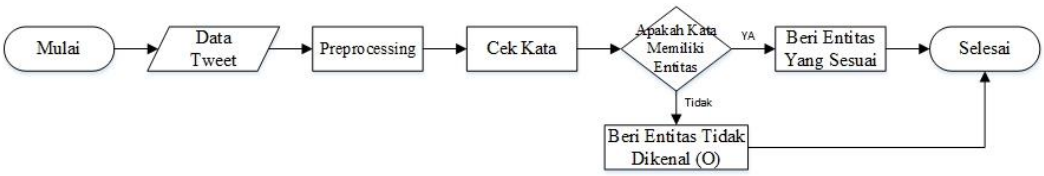
A. Tokenizing

Pada tahap pertama penulis akan melakukan tokenisasi pada data latih. Data latih akan diubah menjadi potongan kata. Berikut merupakan data latih yang sudah melalui proses tokenisasi yang dapat dilihat pada Tabel 3.11.

Tabel 3.11 Contoh tokenisasi pada NER

Tweet Sebelum di Tokenisasi	Tweet Setelah Di tokenisasi
Kasus dbd meningkat di karimun	kasus, dbd, meningkat, di, karimun

B. Entity Tagging



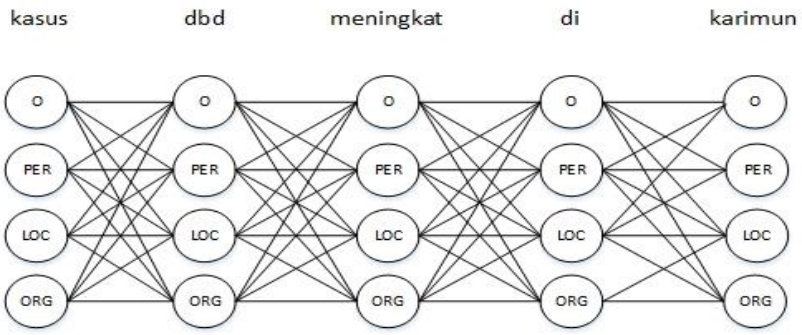
Gambar 3.11 *Flowchart entity tagging*

Pada tahap *entity tagging* berdasarkan Gambar 3.11 data *tweet* yang sudah melalui proses *preprocessing* selanjutnya akan dilakukan pemberian entitas secara manual dengan melakukan pengecekan kata untuk mengidentifikasi jenis entitas pada tiap kata. Format entitas yang digunakan pada proses Named Entity Recognition adalah *token/tag*. Pemberian entitas mengacu pada aturan yang sudah ditentukan. Aturan yang akan digunakan untuk proses Named Entity Recognition dapat dilihat pada Tabel 3.12.

Tabel 3.12 Aturan NER

No	Tag	Keterangan
1	O	Tidak Dikenali
2	Orang	Nama Orang
3	Lokasi	Nama Tempat
4	Organisasi	Nama Organisasi

Proses penentuan entitas akan dilakukan dengan melihat kata yang ada disekitar entitas tersebut yang menunjukkan entitas lokasi, entitas orang, dan entitas organisasi. Kata yang menunjukkan entitas tempat yaitu “di”,”ke”,”menuju” dan sebagainya. Kata yang menunjukkan entitas orang yaitu “bapak”, “ibu”, dan sebagainya. Kata yang menunjukkan entitas organisasi yaitu “universitas”, “sekolah”, dan sebagainya. Tujuan dari melihat kata disekitar entitas adalah untuk menghindari keambiguan entitas. Berikut yang akan dilakukan dalam proses anotasi pada data *tweet* dapat dilihat pada Gambar 3.12.



Gambar 3.12 Proses *entity tagging*

Dari proses yang telah dilakukan pada Gambar 3.12 dilakukan proses pengecekan entitas pada tiap kata. Kata "di" pada contoh menunjukkan nama dari sebuah lokasi dikarenakan setelah dilakukan pengecekan letak kata "di" berada sebelum kata "karimun" yang memiliki kegunaan sebagai penunjuk tempat. Maka pada pengecekan tersebut dinyatakan bahwa entitas dari kata "karimun" adalah lokasi. Berikut adalah data *tweet* yang sudah melewati proses anotasi dapat dilihat pada Tabel 3.13.

Tabel 3.13 Contoh anotasi data tweet

Tweet Sebelum di Anotasi	Tweet Setelah di Anotasi
kasus dbd meningkat di karimun	kasus/O dbd/O meningkat/O di/O karimun/lokasi

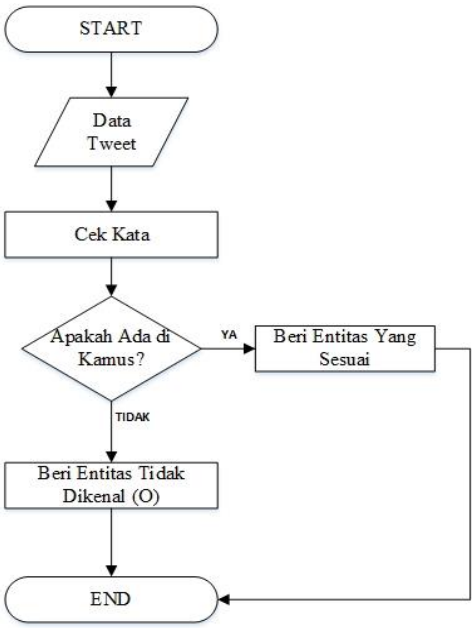
C. Training

Pada tahap *training* penulis akan melakukan pelatihan model NER bahasa Indonesia dengan menggunakan arsitektur dari *spaCy*. Data latih yang sudah dilakukan pemberian entitas oleh penulis selanjutnya dapat digunakan untuk melakukan proses *training* untuk mendapatkan model Named Entity Recognition yang akan digunakan pada penelitian ini. Setelah model didapatkan maka selanjutnya akan dilakukan proses pengujian pada model.

D. Testing

Setelah pemodelan Named Entity Recognition sudah dibuat maka selanjutnya penulis akan melakukan tahap *testing*. *Testing* dilakukan untuk mengukur kemampuan model yang didapatkan pada tahap *training* menggunakan dataset

yang tidak digunakan pada tahap *training*. Berikut adalah gambar flowchart dari alur proses testing dapat dilihat Gambar 3.13.



Gambar 3.13 Flowchart alur proses testing model

Dari Gambar 3.13 dapat dijelaskan bahwa alur yang akan dilakukan yaitu data *tweet* yang digunakan sebagai data *testing* akan dilakukan pengecekan kata dengan melihat pada kamus atau *corpus* yang didapatkan pada proses *training*. Jika kata tersebut ada pada kamus maka akan diberikan entitas sesuai dengan yang ada pada kamus dan jika kata tidak ada pada kamus maka kata tersebut akan diberikan entitas tidak dikenal.

3.6 Pengujian Sistem

Pada tahapan ini akan dilakukan pengujian pada sistem yang telah dibangun untuk melihat apakah sistem berfungsi sesuai dengan tujuan penelitian. Pengujian akan dilakukan pada sistem klasifikasi dan deteksi lokasi.

3.6.1 Pengujian Sistem Klasifikasi Naïve Bayes

Untuk mengevaluasi sistem klasifikasi akan dievaluasi menggunakan *confusion matrix*. *Confusion matrix* merupakan cara yang digunakan untuk menghitung apakah klasifikasi sudah berjalan dengan baik dalam mengenali data dengan menggunakan tabel matriks [30]. Pengujian yang akan dilakukan meliputi empat penilaian yaitu menghitung akurasi sistem antara nilai prediksi dengan nilai keseluruhan disebut dengan *accuracy,* menghitung ketepatan informasi yang diminta dengan balasan dari sistem disebut dengan *precision,* menghitung keberhasilan sistem dalam menemukan kembali informasi disebut dengan *recall,* dan menghitung perbandingan rata-rata nilai *precision* dan nilai *recall* disebut dengan *F1-Measure*. Berikut persamaan 3.1, 3.2, 3.3 dan 3.4 yang akan digunakan untuk menghitung kinerja model klasifikasi.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3.1)$$

$$Precision = \frac{TP}{TP+FP} \quad (3.2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3.3)$$

$$F1\text{-}Measure = \frac{2\ x\ (Precision\ x\ Recall)}{Precision+Recall} \quad (3.4)$$

Dimana : TP = Jumlah tweet bernilai benar yang diklasifikasikan bernilai benar.

TN = Jumlah tweet bernilai salah yang diklasifikasikan bernilai salah.

FP = Jumlah tweet bernilai benar yang diklasifikasikan bernilai salah.

FN = Jumlah tweet bernilai salah yang diklasifikasikan bernilai benar.

3.6.2 Pengujian Sistem Deteksi Lokasi Dengan NER

Untuk mengetahui kinerja model dalam mendeteksi lokasi kasus demam berdarah yang berasal dari *tweet* maka diperlukan evaluasi. Evaluasi kinerja NER akan dilakukan dengan menggunakan *confusion matrix* . Evaluasi kinerja meliputi 3 penilaian yaitu *Precision* untuk menghitung besarnya perbandingan jumlah entitas yang diprediksi benar dengan jumlah keseluruhan entitas yang berhasil di prediksi,

Recall untuk menghitung perbandingan entitas yang berhasil di prediksi dengan benar dengan jumlah entitas keseluruhan, dan *F1-Measure* untuk menghitung perbandingan antara nilai *precision* dan nilai *recall.* Berikut persamaan 3.5, 3.6 dan 3.7 yang akan digunakan untuk menghitung kinerja model deteksi lokasi.

$$Precision = \frac{TP}{TP+FP} \quad (3.5)$$

$$Recall = \frac{TP}{TP+FN} \quad (3.6)$$

$$F1\text{-}Measure = \frac{2\ x\ (Precision\ x\ Recall)}{Precision+Recall} \quad (3.7)$$

Dimana : TP = Jumlah entitas yang berhasil di prediksi dengan benar.

FP = Jumlah entitas yang berhasil di prediksi tetapi salah.

FN = Jumlah entitas yang tidak berhasil di prediksi.

3.7 Analisis Hasil Sistem

Pada tahap ini akan menentukan kesimpulan dari pengujian yang dilakukan pada sistem untuk melihat apakah klasifikasi dan deteksi lokasi kasus demam berdarah di Indonesia berdasarkan data *tweet* dengan menggunakan metode Naive Bayes dan Named Entity Recognition sudah dapat beroperasi sesuai dengan tujuan penelitian ini.