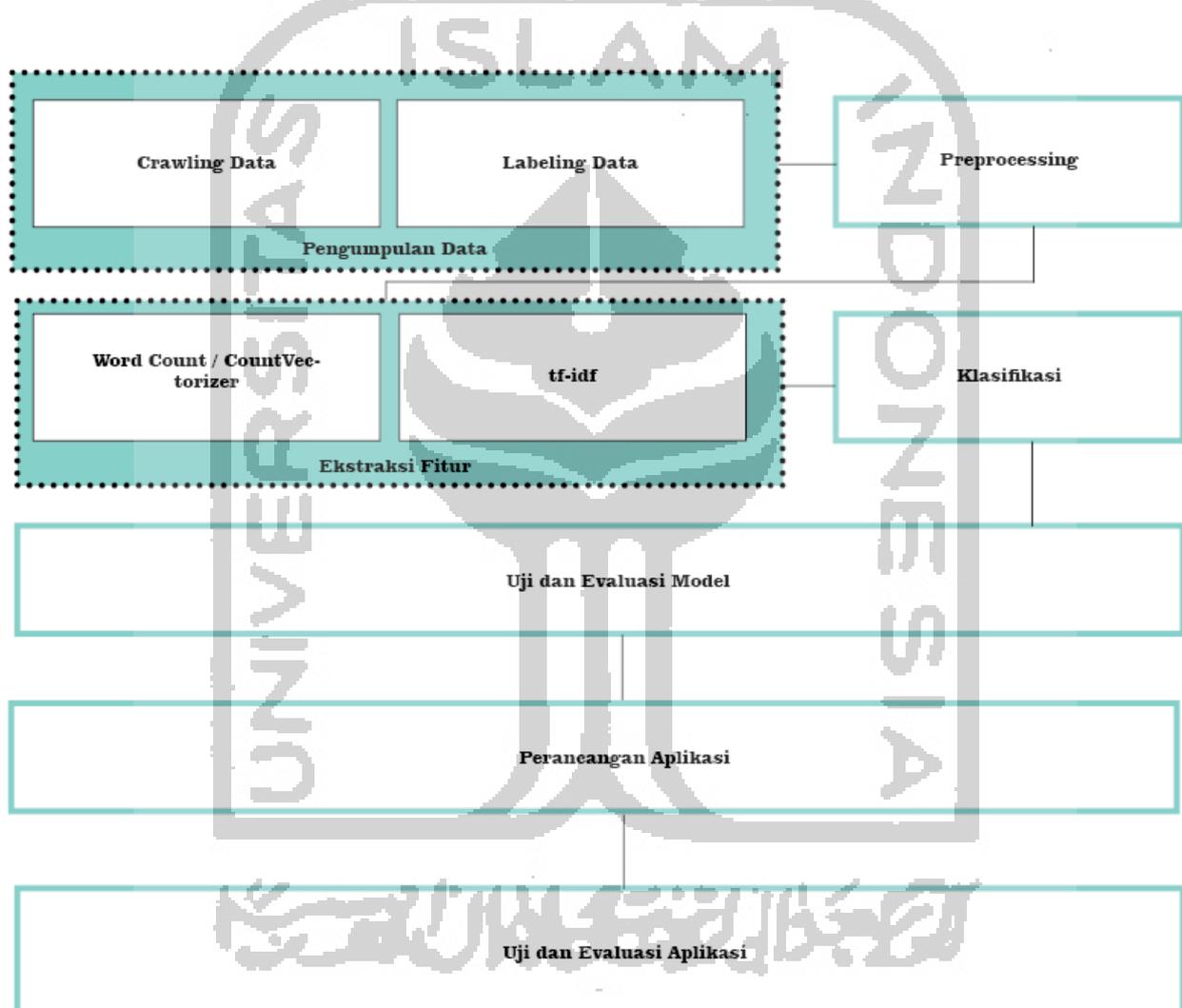


BAB III METODOLOGI PENELITIAN

3.1 Alur Pengerjaan Penelitian

Penelitian dilakukan secara *step by step* agar mencapai tujuan penelitian yang diinginkan. Secara umum, alur pengerjaan penelitian dapat dilihat pada gambar 3.1



Gambar 3.1 Alur penelitian

Langkah pertama yang dilakukan pada penelitian ini adalah pengumpulan data yang terdiri dari *Crawling Data* dan *Labeling Data*. Data dikumpulkan dari media sosial *Twitter* berupa cuitan-cuitan pengguna berbahasa Indonesia lalu kemudian dilakukan proses *labeling* yaitu membagi data cuitan menjadi dua kelas, yaitu kelas *cyberbullying* dengan label nol (0) dan kelas *non-cyberbullying* dengan label satu (1).

Langkah selanjutnya adalah tahap pembersihan data cuitan, atau yang dikenal dengan tahap *preprocessing*. Pada langkah ini, data cuitan dibersihkan hingga menjadi teks yang lebih terstruktur. Langkah *preprocessing* yang dilakukan antara lain:

- a. Menghilangkan URL
- b. Menghapus karakter NON-ASCII
- c. Menghapus angka, simbol dan tanda baca
- d. Menghapus *hashtag*, *username*, dan RT
- e. *Normalize*
- f. Penyeragaman huruf ke dalam bentuk *lowercase*
- g. Menghapus *stopwords*.

Langkah ke tiga adalah melakukan ekstraksi fitur untuk memberikan pembobotan pada tiap-tiap token pada baris data sehingga bisa diolah pada tahap selanjutnya. Metode ekstraksi fitur yang digunakan adalah metode *TF-IDF*.

Tahap selanjutnya ialah tahap klasifikasi. *Classifier* yang digunakan pada penelitian ini adalah *Multinomial Naïve Bayes*. Baris-baris data yang sudah dilakukan pembobotan pada tahap sebelumnya menjadi masukan untuk algoritma *Multinomial Naïve Bayes*.

Setelah sampai tahap empat, maka dilakukan tahap selanjutnya yakni tahap pengujian dan evaluasi. Pada tahap ini, dilakukan pengujian terhadap performa model yang direpresentasikan dalam *confussion matrix* dan menjadi tolok ukur evaluasi. Jika model dianggap sudah memiliki performa baik, maka dilanjutkan ke tahap selanjutnya.

Tahap selanjutnya adalah pembuatan aplikasi. Model yang sudah dibuat sebelumnya dijadikan inti dari aplikasi. Aplikasi didesain dengan *framework Bootstrap* dan *Django*.

Tahap terakhir ialah uji dan evaluasi aplikasi dengan menggunakan metode *black box testing*. Tahap ini dilakukan untuk mengetahui performa aplikasi, apakah sudah sesuai dengan tujuan penelitian atau belum.

3.2 Uraian Metodologi

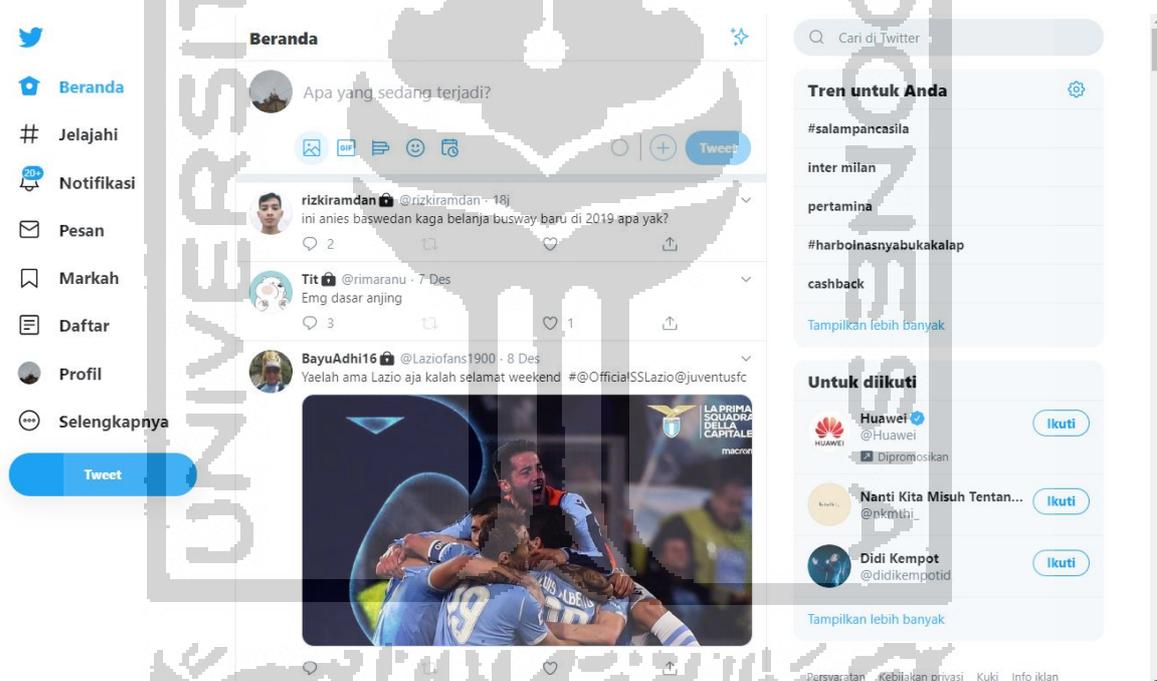
Pada bagian ini, akan dijelaskan secara rinci tahapan-tahapan yang dilakukan untuk mencapai tujuan penelitian.

3.2.1 Pengambilan Data

Data yang digunakan pada penelitian ini adalah data cuitan pada media sosial *Twitter*. Alasan dari dipilihnya *Twitter* sebagai sumber data adalah sebagai berikut:

1. *Twitter* merupakan media social yang masih memiliki banyak pengguna di Indonesia berdasarkan data dari *Country Industry Head Twitter Indonesia* (Clinten, 2019).
2. Pengguna *Twitter* dari berbagai kalangan, mulai dari anak-anak, remaja, orang dewasa, pemerintah, dan lembaga negara.
3. *Twitter* memberikan akses kepada peneliti untuk menggunakan datanya melalui *Twitter API*.

Cuitan yang dihimpun berupa cuitan pengguna berbahasa Indonesia. Data diambil menggunakan teknik *crawling* menggunakan bahasa pemrograman *Python*.



Gambar 3.1 Halaman beranda *Twitter*

Data yang dikumpulkan berupa cuitan-cuitan yang diambil menggunakan kata-kata kunci yang didapatkan oleh penulis melalui sebaran *Google Form* pada beberapa pengguna media sosial. Kata kunci yang didapatkan dari hasil survei di antaranya adalah *belagu, dungu, goblok, iq jongkok, jelek, jijik, kampungan, kontet, ngartis, norak, sombong, songong, tolol, udik*. Data yang didapat kemudian disimpan dengan format *xlsx*.

Setelah terkumpul, data cuitan dibagi menjadi dua kelas yang berbeda yaitu kelas cuitan *cyberbullying* dengan label satu (1) dan cuitan *non-cyberbullying* dengan label nol (0). Proses pelabelan dilakukan sendiri oleh penulis dengan pelabelan manual.

Tabel 3.1 Contoh data yang dihimpun

Cuitan	Label
@Hilmi28 @prabowo Semoga Allah SWT melindungi dan menjaga pak @prabowo bang @sandiuno barakallohu fiikum	0
@humantolol kata papa juga gaboleh sombong	0
@ilovebillies kata mama ga boleh sombong	0
@bilonk Duh belagu amat Wkwk	1
@danillajpr Danilla gendut	1
@bbylibur @seolhyunlibur @sooyaaalibur @tehalibur @holidayjailed beginilah gambaran kalau iq anda jongkok 🙇🙇🙇	1
@beervodkas SONGONG	1
@BEJOS09958347 @ardi_riau @mohmahfudmd Di si dungu	1
@berteman_mari Biar mereka pada makan batu aja jendral...songong, arogan, blegug ☐☐	1
@BATMANXBABALL YaAllah udik sekali diriku😅	0
@beauthingy Gapernah beli di olshop (apalagi yg import bangkok dll) krn pasti ngatung jelek gt:(0
@bobbyifaa Kalo di liat dari followers yg udah 4k si keknya doi ngartis 😊	0
@bombasterwit_ Ya gaboleh nge tolol toloin diri sendiri mending nge toloin aku aja 😊☐	0

3.2.2 Preprocessing

Preprocessing berfungsi untuk membersihkan data cuitan yang berupa teks dari kata-kata atau simbol yang kurang penting, sehingga menjadi data teks yang terstruktur. Hal yang dilakukan diantaranya *cleaning*, *remove stopwords*, dan *stemming* yang akan dijelaskan sebagai berikut:

a. Cleaning

Tidak semua cuitan yang didapatkan merupakan teks yang terstruktur dan sesuai dengan kaidah penulisan yang benar. Bagian ini berfungsi untuk membersihkan cuitan dari *URL*, karakter NON-ASCII, angka, simbol, tanda baca, *hashtag*, *username*, dan RT. Proses ini dieksekusi dengan bahasa pemrograman *Python* pada IDE *Jupyter Notebook* yang hasilnya disimpan dalam file berformat *xlsx*.

Contoh cuitan yang telah melalui proses *cleaning* dapat dilihat pada tabel 3.2.

Tabel 3.2 Penerapan *cleaning* pada cuitan

Sebelum	Sesudah
@VIVAcoid Alhamdulillah.... terimakasih ya rob!.orang sombong tak layak jadi wakil rakyat.	alhamdulillah terimakasih iya rob orang sombong tak layak jadi wakil rakyat
@vngnst Nanti aku dikatain sombong	nanti aku dikatakan sombong
@xsehunsi Dih udah pagi ☹️ Kok sehun sombong sama kac y!	dih sudah pagi kok sehun sombong sama kac y
@tubirfess Mereka gak sadar lebih kampungan dari si mc yang mereka sendiri kata katin, bego.	mereka tidak sadar lebih kampungan dari si mc yang mereka sendiri kata katin bego
boro2 bales dm/mt, tidur aja kurang. nanti ya dibales, jangan ngatain sombong. kurang aktif lo nya kl kaga paham anw, good morning 🙏	boro2 balas dm makan teman tidur saja kurang nanti iya dibalas jangan katakan sombong kurang aktif kamu nya kalau tidak paham anw good morning

b. *Normalize*

Cuitan pada media social *Twitter* biasanya ditulis menggunakan penulisan yang tidak sesuai dengan standar penulisan yang benar, mulai dari penyingkatan terhadap kata, menggunakan istilah-istilah kekinian, menggunakan Bahasa daerah dan menggunakan istilah-istilah asing. Pada bagian ini dilakukan normalisasi kata-kata dalam data cuitan yang penulisannya tidak dalam penulisan yang semestinya. Contoh penerapan tahap ini dapat dilihat pada tabel 3.3.

Tabel 3.3 *Normalize Slangword* pada cuitan

Sebelum	Sesudah
males followan sm rp sombong	males followan sama rp sombong
Cakep si tp sombong	Cakep si tapi sombong
cuyun jelek kalau ga chaeyon nunak	cuyun jelek kalau tidak chaeyon nunak
iya jdi kaya sombong gt ya kan	iya jadi kaya sombong begitu iya kan
gpp masih iq jongkok mas dari pada iq ndlosor	tidak apa-apa masih iq jongkok mas dari pada iq ndlosor
o yaampun udik bgt ak	o yaampun udik banget saya
iq nya ga kesemutan jongkok terus	iq nya tidak kesemutan jongkok terus
temen yg di ig kadang so ngartis	temen yang di instagram kadang sok ngartis
gue suka gaya elo om	saya suka gaya kamu om
heboh ndiri bantah ndiri klo gak tolol ya bukan cebong	heboh sendiri bantah sendiri kalau tidak tolol iya bukan cebong
loe idiot apa mmg dungu dr kecil ya gk paham jg klo org ngomong	kamu idiot apa memang dungu dari kecil iya tidak paham juga kalau orang bicara

c. *Stemming*

Tahap *Stemming* adalah tahap untuk mengubah kata dalam cuitan ke dalam bentuk kata dasarnya dengan menghapus imbuhan. Contoh penerapan *Stemming* dapat dilihat pada tabel 3.4.

Tabel 3.4 *Stemming* pada data cuitan

Sebelum	Sesudah
aku dikatain sombong	aku kata sombong
mereka tidak sadar kampungan mc yang mereka sendiri katain bego	mereka tidak sadar kampung mc yang mereka sendiri katain bego
boro balas dm makan teman tidur saja dibalas jangan katakan sombong aktif kamu tidak paham anw good morning	boro balas dm makan teman tidur saja balasan kata sombong aktif kamu tidak paham anw good morning

d. *Removing Stopwords*

Setelah melalui proses *Cleaning*, teks akan diproses pada tahap *Removing Stopwords*. *Stopwords* berisi kumpulan kata-kata yang kurang penting dalam teks cuitan. Tahap ini berfungsi untuk menghapus kata-kata yang tidak memiliki makna. Pada tahap ini juga dilakukan proses pemecahan kata-kata pada kalimat cuitan menjadi bagian yang dinamakan token, yang berguna untuk pembobotan di tahap selanjutnya. Contoh penerapan tahap ini dapat dilihat pada tabel 3.5.

Tabel 3.5 *Removing Stopwords* pada cuitan

Sebelum	Sesudah
alhamdulillah terimakasih iya rob orang sombong tak layak jadi wakil rakyat	terimakasih orang sombong tak layak wakil rakyat
nanti aku dikatain sombong	aku dikatakan sombong
dih sudah pagi kok sehung sombong sama kacu	dih sudah sehung sombong kacu
mereka tidak sadar lebih kampungan dari si mc yang mereka sendiri kata katain bego	mereka tidak sadar kampungan mc yang mereka sendiri katain bego
boro balas dm makan teman tidur saja kurang nanti iya dibales jangan ngatain sombong kurang aktif kamu nya kalau tidak paham anw good morning	boro balas dm makan teman tidur saja dibalas jangan katakan sombong aktif kamu tidak paham anw good morning

3.2.3 Ekstraksi Fitur

Setelah data cuitan melalui tahap *preprocessing* dan bentuknya menjadi terstruktur, maka tahap selanjutnya ialah ekstraksi fitur. Algoritma NBC tidak bisa memproses data karena masih berbentuk teks. Pada tahap ini cuitan yang berbentuk teks direpresentasikan ke dalam bentuk *vector*. Kalimat-kalimat cuitan yang berupa *array* dijadikan dalam bentuk matriks, di mana baris matriks merepresentasikan baris teks pada dokumen dan kolom matriks merepresentasikan kata-kata pada dokumen. Setelah teks berubah menjadi vektor kata, langkah selanjutnya adalah memberikan pembobotan terhadap setiap kata pada tiap kalimat menggunakan metode *tf-idf*.

Langkah pertama pada metode *tf-idf* adalah menghitung nilai dari *Term Frequency (TF)* yaitu banyaknya kata baku (*term*) yang muncul dalam semua dokumen *dataset*. Sebagai contoh, terdapat kata baku ‘mewah’, ‘ramai’, ‘indah’, ‘murah’ dan tiga kalimat yang disimbolkan dengan K1, K2, dan K3 pada *dataset*. Tabel *Term Frequency* dapat dilihat pada tabel 3.1.

Tabel 3.6 Tabel *term frequency*

	(K1)	(K2)	(K3)
Mewah	2	1	0
Ramai	1	1	0
Indah	0	0	1
Murah	1	1	2

Setelah *TF* diketahui, langkah selanjutnya adalah mencari tahu nilai *Document Frequency (DF)* yaitu jumlah baris data atau dokumen yang mengandung kata baku (*term*). Contoh dari nilai *DF* berdasar data pada contoh sebelumnya seperti yang terlihat pada tabel 3.2.

Tabel 3.7 Contoh tabel *DF*

<i>Term</i>	<i>Document Frequency</i>
Mewah	2
Ramai	1
Indah	1
Murah	1

Setelah DF diketahui, langkah selanjutnya adalah menghitung nilai *Inverse Document Frequency* (IDF) dengan persamaan di bawah ini.

$$IDF = \log\left(\frac{D}{DF_j}\right) \quad (3.1)$$

Di mana D adalah jumlah seluruh dokumen, sedangkan DF_j adalah nilai DF untuk tiap-tiap *term*.

Setelah nilai TF dan IDF diketahui, maka langkah terakhir adalah menghitung pembobotan dengan mengali nilai TF dengan IDF seperti yang terlihat pada persamaan di bawah ini.

$$W_{ij} = TF_{ij} \times \log\left(\frac{D}{DF_j}\right) \quad (3.2)$$

Keterangan:

W_{ij} = bobot j terhadap dokumen d

TF_{ij} = nilai TF *term* j pada dokumen d

D = jumlah baris data atau dokumen pada *dataset*

DF_j = jumlah dokumen yang mengandung *term* j .

3.2.4 Klasifikasi

Setelah dilakukan pembobotan pada tahap ekstraksi fitur, data cuitan siap untuk diproses menggunakan algoritma *Multinomial Naïve Bayes*. Model multinomial memperhitungkan frekuensi setiap kata yang muncul pada dokumen. Misal terdapat dokumen d dan himpunan kelas c . Untuk memperhitungkan kelas dari dokumen d , maka dapat dihitung dengan persamaan berikut (Rahman, Wiranto, & Doewes, 2017).

$$P(c|\text{term dokumen } d) = P(c) \times P(t_1|c) \times P(t_2|c) \times P(t_n|c) \quad (3.3)$$

Keterangan:

$P(c)$ = Probabilitas *Prior* dari kelas c

$P(t_n|c)$ = Probabilitas kata ke- n dengan diketahui kelas c

$$P(c) = \frac{N_c}{N} \quad (3.4)$$

Keterangan:

N_c = Jumlah kelas c pada seluruh dokumen

N = Jumlah seluruh dokumen

Sementara persamaan *Multinomial* yang digunakan dengan pembobotan *TF-IDF* adalah sebagai berikut:

$$P(tn|c) = \frac{Wct + 1}{(\sum W' \epsilon V W' ct) + B'} \quad (3.5)$$

Keterangan:

Wct = Nilai pembobotan *TF-IDF* atau W dari *term* t di kategori c

$\sum W' \epsilon V W' ct$ = Jumlah total W dari keseluruhan *term* yang berada pada kategori c

B' = Jumlah W kata unik (nilai *idf* tidak dikali dengan *tf*) pada seluruh dokumen

Setelah probabilitas *term* diketahui, langkah selanjutnya adalah menghitung probabilitas sebenarnya dari dokumen d dengan persamaan

$$P(c = 0 | term \text{ dokumen } d) = \frac{P(0 | term \text{ dokumen } d)}{p(d)} \quad (3.6)$$

$$P(c = 1 | term \text{ dokumen } d) = \frac{P(1 | term \text{ dokumen } d)}{p(d)} \quad (3.7)$$

$$P(d) = P(1 | term \text{ dokumen } d) + P(0 | term \text{ dokumen } d) \quad (3.8)$$

Keterangan:

$P(c = 0 | term \text{ dokumen } d)$ = Probabilitas *non-cyberbullying*

$P(c = 1 | term \text{ dokumen } d)$ = Probabilitas *cyberbullying*

$p(d)$ = Probabilitas dokumen

Agar lebih mudah dipahami, berikut dituliskan sebuah contoh kasus bagaimana sebuah teks diklasifikasikan dengan menggunakan algoritma *Multinomial Naïve Bayes*.

Tabel 3.8 Sampel cuitan

kode	cuitan	label
K1	Kamu jelek banget sih udah gitu miskin juga, pantas saja kamu tidak punya teman.	1
K2	Dasar miskin, tidak usah kebanyakan menghayal ya.	1
K3	Kamu baik sekali, semoga Allah membalas kebaikanmu.	0
K4	Kamu rajin sekali, aku suka itu.	0

Diberikan 4 cuitan sebagai sampel, dan didapati kata baku (*term*) jelek, miskin, baik, dan rajin. Keempat sampel cuitan dijadikan sebagai data training untuk klasifikasi.

Tabel pembobotan tiap *term* dapat dilihat pada tabel di bawah ini

Tabel 3.9 Bobot tiap *term*

Kata Baku	TF					DF	D/DF	IDF	W=TF*(IDF+1)				
	K1	K2	K3	K4	C1				K1	K2	K3	K4	C1
Jelek	1	0	0	0	1	2	2	0.3	0,3	0	0	0	0.3
Miskin	1	1	0	0	1	3	1.3	0.1	0.1	0.1	0	0	0.1
Baik	0	0	1	0	1	2	2	0.3	0	0	0.3	0	0.3
Rajin	0	0	0	1	0	1	4	0.6	0	0	0	0.3	0
nilai bobot tiap dokumen									0.4	0.1	0.3	0.3	0.7

Kemudian, diberikan sebuah kalimat untuk diklasifikasikan dengan algoritma *Multinomial Naïve Bayes*.

“Kamu baik banget, cuma sayang kamu jelek dan miskin”

Diketahui:

t1 = “Jelek”

t2 = “Miskin”

t3 = “Baik”

$$P(c|term\ dokumen\ d) = P(c) \times P(t1|c) \times P(t2|c) \times P(tn|c)$$

$$P(1|term\ dokumen\ d) = P(1) \times P(t1|1) \times P(t2|1) \times P(t3|1)$$

$$P(1|term\ dokumen\ d) = 0.5 \times 0.72 \times 0.61 \times 0.55$$

$$P(1|term\ dokumen\ d) = 0.12$$

$$P(0|\text{term dokumen } d) = P(0) \times P(t1|0) \times P(t2|0) \times P(t3|0)$$

$$P(0|\text{term dokumen } d) = 0.5 \times 0.52 \times 0.52 \times 0.68$$

$$P(0|\text{term dokumen } d) = 0.09$$

$$P(c = 0|\text{term dokumen } d) = \frac{P(0|\text{term dokumen } d)}{p(d)}$$

$$P(c = 0|\text{term dokumen } d) = \frac{0.09}{0.12 + 0.09}$$

$$P(c = 0|\text{term dokumen } d) = 0.43$$

$$P(c = 1|\text{term dokumen } d) = \frac{P(1|\text{term dokumen } d)}{p(d)}$$

$$P(c = 1|\text{term dokumen } d) = \frac{0.12}{0.12 + 0.09}$$

$$P(c = 1|\text{term dokumen } d) = 0.57$$

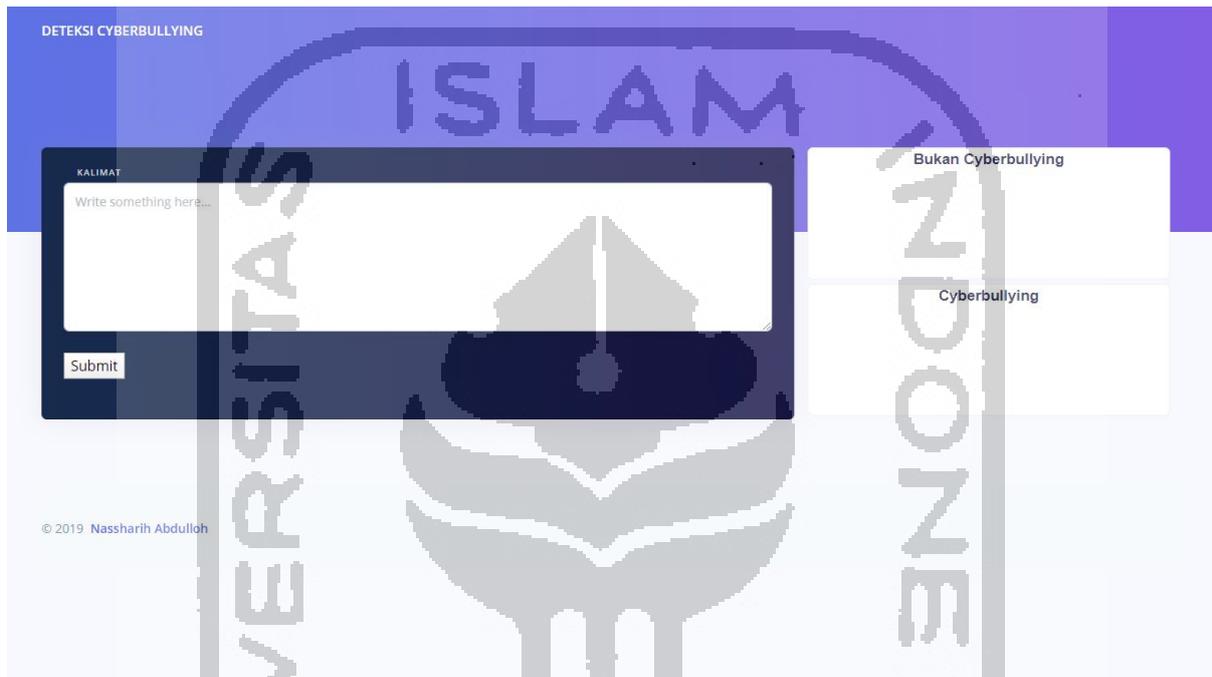
Berdasarkan perhitungan probabilitas masing – masing kelas, kalimat uji diklasifikasikan ke dalam kelas *cyberbullying* dengan probabilitas 0.57 yang lebih besar dibanding dengan probabilitas kelas *non-cyberbullying* yakni 0.43.

3.2.5 Uji dan Evaluasi Model

Model diuji dengan menggunakan data *testing* yang diperoleh dari sebagian data *training*. Data *testing* yang digunakan dipilih secara acak dengan bantuan *library* pada bahasa pemrograman *Python*. Evaluasi model akan dilakukan dengan melihat *confussion matrix* yang menjadi keluaran model. Dengan angka-angka yang tertera pada *confussion matrix*, variabel *accuracy*, *precision*, *recall*, dan *f1-score* dapat diketahui nilainya. Nilai pada variabel-variabel tersebut, menjadi tolok ukur performa model.

3.2.6 Perancangan Aplikasi

Setelah model dibuat dan menunjukkan performa yang baik, langkah selanjutnya ialah mengimplementasikan model ke dalam aplikasi berbasis *website* yang dibuat dengan *framework Django* dan *framework Bootstrap*. Aplikasi terdiri dari sebuah *textfield* dan hasil prediksi yang berupa probabilitas dari masing - masing kelas (*cyberbullying* dan bukan *cyberbullying*).



Gambar 3.2 Antarmuka aplikasi deteksi *cyberbullying*

3.2.7 Uji dan Evaluasi Aplikasi

Setelah antarmuka dan logika pemrograman di terapkan pada aplikasi, pengujian dan evaluasi dilakukan dengan teknik pengujian *black box testing*. Pengujian dilakukan dengan menuliskan sebuah kalimat pada *textfield* kemudian melihat hasil identifikasi pada bagian diagram.

Pengujian ini melibatkan lima orang sebagai *tester*, yang dipilih oleh penulis berdasarkan pemahamannya terhadap sistem dan logika pemrograman. *Tester* merupakan mahasiswa program studi Informatika UII.