IMPLEMENTASI TEXT MINING UNTUK MENDETEKSI HATE SPEECH PADA TWITTER

Disusun Oleh:

N a m a : Setyo Legianto

NIM : 14523195

PROGRAM STUDI TEKNIK INFORMATIKA – PROGRAM SARJANA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM INDONESIA

2019

HALAMAN PENGESAHAN DOSEN PEMBIMBING

IMPLEMENTASI TEXT MINING UNTUK MENDETEKSI HATE SPEECH PADA TWITTER

TUGAS AKHIR

Disusun Oleh:

N a m a : Setyo Legianto
NIM : 14523195

Yogyakarta, 02 Oktober 2019

Pembimbing 1,

(Yudi Prayudi S.Si., M. Kom.)

Pembimbing 2,

(Ahmad Fathan Hidayatullah, S.T., M. Cs.)

HALAMAN PENGESAHAN DOSEN PENGUJI

IMPLEMENTASI TEXT MINING UNTUK MENDETEKSI HATE SPEECH PADA TWITTER

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Teknik Informatika di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 02 Oktober 2019

Tim Penguji

Pembimbing 1

Yudi Prayudi S.Si., M.Kom.

Pembimbing 2

Ahmad Fathan Hidayatullah, S.T., M. Cs.

Anggota 1

Zainudin Zukhri, S.T., M.I.T.

Anggota 2

Dhomas Hatta Fudholi, S.T., M. Eng., Ph.D.

Mengetahui,

Ketua Program Studi Teknik Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia

(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Setyo Legianto

NIM : 14523195

Tugas akhir dengan judul:

IMPLEMENTASI TEXT MINING UNTUK MENDETEKSI HATE SPEECH PADA TWITTER

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta. 02 Oktober 2019

(Setyo Legianto)

HALAMAN PERSEMBAHAN

Alhamdulillahirobbil'alamin atas segala nikmat yang telah diberikan kepada kita. Salawat serta salam kita haturkan kepada junjungan kita Nabi Muhammad SAW yang kita nantikan safa'atnya di yaumul akhir nanti.

Terima kasih yang amat besar saya ucapkan kepada kedua orang tua saya yang telah mengasuh dan mendidik saya sejak di dalam kandungan sampai pada saat ini juga. Semoga beliau senantiasa diberikan kesehatan, kebahagiaan dan panjang umur.

Terima kasih kepada Dosen Pembimbing pertama saya, Bapak Yudi Prayudi yang memberikan semangat serta masukkan terhadap penelitian saya. Terima kasih juga kepada Dosen Pembimbing kedua saya bapak Ahmad Fathan Hidayatullah yang selalu melatih, membimbing saya dengan sabar dan selalu memerhatikan saya di setiap pertemuan bimbingan maupun pada saat tidak dapat bertemu dengan langsung.

Terima kasih kepada semua pihak yang tidak bisa disebutkan satu persatu atas dukungan dan bantuannya baik secara langsung maupun tidak langsung.

HALAMAN MOTO

“Mudahkan dan jangan mempersulit, berikan kabar gembira dan jangan membuat manusia lari HR. Bukhari”

“Sakitmu, lelahmu, dan sedihmu adalah penggugur dosamu”

“Jangan pernah merasa bahwa diri kita besar, karena kita mempunya Allah Yang Maha Besar”

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh

Alhamdulillahi Robbil Alamin, adalah kalimat yang bisa terucap kepada Allah SWT, karena dengan segala rahmat dan karunia-Nya lah peneliti bisa bertahan menahan segala gangguan dan godaan dalam menyelesaikan laporan penelitian dengan judul "Implementasi Text Mining Untuk Mendeteksi *Hate Speech* pada Twitter". Salawat serta salam juga tidak lupa selalu tercurahkan kepada Nabi Muhammad SAW, sebagai panutan seluruh umat di segala penjuru dunia.

Adapun dalam penyelesaian tugas akhir ini, banyak pihak yang terlibat dalam penyelesaiannya. Oleh karena itu, peneliti ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. ALLAH SWT , yang telah memberikan kelancaran dalam segala pengerjaan laporan tugas akhir ini.
2. Bapak Bejo dan Ibu Sudarmini orang tua dan keluarga atas segala doa dan dukungan selama saya melakukan tugas akhir
3. Mba Erva Darmayanti selaku kakak kandung saya yang selalu support serta memotivasi saya.
4. Bapak Dr. Raden Teduh Dirgahayu, S.T., M. Sc. selaku Kaprodi Teknik Informatika
5. Bapak Yudi Prayudi S.Si., M.Kom., selaku Dosen Pembimbing Pertama Tugas Akhir di Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
6. Bapak Ahmad Fathan Hidayatullah, S.T., M. Cs., selaku Dosen Pembimbing Kedua Tugas Akhir di Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
7. Seluruh jajaran staf dan dosen Teknik Informatika Universitas Islam Indonesia.
8. Kepada teman-teman Keluarga Wacana, yang selalu memberikan dukungan dan semangat positif.
9. Teman-teman dari kota Samarinda yang telah memberikan motivasi dan semangat
10. Kepada semua teman-teman Jurusan Teknik Informatika angkatan 2014.
11. Serta semua orang yang selalu mendukung dan mendoakan yang tidak bisa disebutkan satu per satu.

Peneliti sadar banyak terdapat kekurangan dalam pembuatan tugas akhir ini. Namun peneliti selalu berharap tugas akhir ini dapat bermanfaat atau mungkin bisa dikembangkan menjadi hal yang lebih besar lagi, sehingga dapat memberikan dampak yang baik untuk dunia, terlebih khusus untuk negara tercinta Indonesia.

Yogyakarta, 02 Oktober 2019

(Setyo Legianto)

SARI

Twitter adalah salah satu dari media sosial, aplikasi yang berbasis microblogging. Microblogging merupakan jenis media sosial yang memfasilitasi pengguna untuk menulis dan memublikasikan aktivitas atau pendapat secara bebas. Dengan adanya media sosial, salah satunya adalah Twitter. Setiap orang dapat saja saling berbagi informasi terhadap orang lain tanpa harus bertemu satu dengan yang lainnya dan juga memiliki kebebasan untuk mengemukakan pendapat. Tetapi dengan media sosial pengguna juga dapat mempengaruhi hal buruk pengguna lain dengan membuat dan menyebarkan informasi yang bersifat tuduhan, fitnah, berita hoax, maupun SARA, semua itu masuk kategori ujaran kebencian atau Hate Speech.

Penelitian ini dilakukan untuk mengetahui performa algoritme Naive Bayes Classifier dalam melakukan proses klasifikasi berdasarkan twitt atau status pengguna Twitter. Sumber data pada penelitian ini menggunakan Twitter.

Uji model penelitian ini dilakukan dengan menggunakan bantuan library python yaitu MultinomialNaiveBayes. Dalam proses uji model, besarnya data tes yang diambil adalah 33% dari data training yang dilakukan secara acak. Evaluasi model yang dilakukan pada penelitian ini menggunakan 5-fold cross validation dengan hasil akurasi ***71.0%.***

Kata kunci: Twitter, Microblogging, Hate Scpeech, Library, Naive Bayes Classifier, Cross Validation.

GLOSARIUM

Pre-processing	Perlakuan awal terhadap data untuk dijadikan bahan *training*.
Training	Mengolah data untuk dijadikan model.
Model	Hasil dari *training* yang digunakan untuk mengklasifikasikan bahasa.
Dataset	Data yang digunakan dalam pembentukan model.
	Word vector Matriks kata yang menandakan kata tertentu terdapat dalam dokumen dengan membandingkan dokumen dengan seluruh kata dari seluruh dokumen.
Klasifikasi	Penentuan kelas secara otomatis menggunakan model.
Mendeteksi	Menemukan objek negatif atau positif di dokumen.

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi pada bidang informasi telah membuat berbagai aplikasi media sosial bermunculan seperti halnya Facebook, Twitter, Instagram dan lain-lain. Menurut Andreas Kaplan dan Michael Heinlein, mendefinisikan "media sosial adalah suatu pengelompokan *software* atau perangkat lunak berbasis Internet yang berada di atas dasar ideologi dan teknologi Web 2.0, serta yang dapat memungkinkan *user* atau pengguna untuk penciptaan dan pertukaran "user-generated content" (Kaplan & Haenlein, 2010). Terdapat beberapa jenis media sosial yang berkembang sampai saat ini dan penggunanya(*user*) masih terbilang juga cukup banyak yang aktif, salah satunya adalah situs jejaring sosial Twitter.

"Twitter adalah salah satu dari media sosial, aplikasi yang berbasis microblogging. Microblogging merupakan jenis media sosial yang memfasilitasi pengguna untuk menulis dan memublikasikan aktivitas atau pendapat." (Viani, 2017). Melanjutkan penelitian dari Afiandiary data yang diperoleh pada Januari 2016 berdasarkan kisaran umur 20-25 tahun, peringkat pertama masih ditempati oleh Facebook sebesar 86,1%, kedua adalah Instagram sebanyak 75,8% dan ketiga adalah sebanyak Twitter sebanyak 41,5%. Pada tahun 2016 30,1 juta pengguna. Pada tugas akhir ini, peneliti memilih subject Twitter karena media tersebut sangatlah simple dalam penggunaannya serta dapat mengirimkan pesan teks mencapai 140 karakter.

Dengan adanya media sosial Twitter. Setiap orang dapat saja saling berbagi informasi terhadap orang lain tanpa harus bertemu satu dengan yang lainnya dan juga memiliki kebebasan untuk mengemukakan pendapat. Dengan media sosial pengguna juga dapat mempengaruhi hal buruk pengguna lain dengan membuat dan menyebarkan informasi yang bersifat tuduhan, fitnah, berita hoax, maupun SARA. "Dalam media sosial dikenal istilah Ucapan kebencian atau dikenal dengan *Hate Speech*, yang makin populer saat ini, hal ini disebabkan gesekan atau perbedaan yang mewakili kelompok-kelompok tertentu baik Suku, Agama, Ras, Etnis, Golongan." (Rohman, 2016). Atas dasar berbagai permasalahan di media sosial tersebut akhirnya pemerintah Indonesia membuat aturan terkait berbagai kejahatan yang terjadi di sosial media. (Pemerintah Indonesia, 2008).

Berdasarkan permasalahan di atas peneliti akan membangun sistem untuk mengalasis sentiment tweet dan pengklasifikasian atau mengelompokkan tweet tersebut mengandung *hate speech* dengan menggunakan metode text mining dan metode Naïve Bayes sebagai pengklasifikasi. "Text mining adalah salah satu perkembangan dari analisis teks prosesnya dikerjakan secara otomatis dengan komputasi oleh computer berguna sebagai penggalian informasi yang berkualitas mencari inti sari dari suatu rangkaian teks atau kata yang terangkum dalam sebuah dokumen" (Han, Kamber, & Pei, 2011). Metode ini akan peneliti gunakan sebagai pengolah data text dalam tweet pengguna Twitter dari hasil *crawling* data secara *real time* pada Twitter. Selain itu pengguna juga menggunakan metode Naïve Bayes. Metode Naïve Bayes Classifier dapat mengklasifikasi dari hasil Crawling data di Twitter dengan *Hate Space* atau perkataan kebencian. Dengan sistem tersebut diharapkan dapat bermanfaat untuk mengetahui performa dari algoritme *Naive Bayes Classifier* sebagai pengklasifikasi pengguna media sosial Twitter terkait *Hate Speech* dalam tweet pengguna Twitter.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, maka rumusan masalah dalam penelitian ini adalah

a. Bagaimana menganalisis sentimen pada *tweet* dan pengklasifikasian *tweet* menggunakan metode *Naïve Bayes Classifier*?
b. Bagaimana akurasi model *Naïve Bayes Classifier* dengan menambahkan teknik *cross validation* untuk teknik menvalidasi data ?
c. Bagaimana perbandingan dari akurasi model *Naïve Bayes Classifier* dengan model *Logistic Regression* sebagai model klasifikasi ?

1.3 Tujuan Penelitian

Adapun tujuan penelitian ini adalah :

a. Peneliti dapat mengolah data Twitter untuk memperoleh inti dari tweet pengguna menggunakan metode *text mining*.
b. Melakukan pengklasifikasian tweet berbahasa indonesia dengan menggunakan metode *Naïve Bayes Classifier*.
c. Mengetahui akurasi model klasifikasi menggunakan metode *Naïve Bayes Classifier* dengan tambahan fitur *Unigram* sebagai pengujian .

d. Mencari metode terbaik untuk membangun sistem pengklasifikasian pada tweet dengan membandingkan metode *Naïve Bayes Classifier* dengan *Logistic Regression*.

1.4 Batasan Masalah

Berdasarkan dari rumusan masalah di atas, terdapat beberapa batasan masalah untuk sistem tersebut sehingga ruang pengerjaan tidak terlalu melebar. Berikut adalah batasan masalah yang akan dilakukan:

a. Sistem yang dibuat menggunakan metode *text mining* sebagai pengolahan data text hasil dari *crawling data* di Twitter .
b. Jumlah data yang di *crawling* hanya 2500 tweet.
c. Hasil akhir dari penelitian adalah analisis sentimen dari metode *Naïve Bayes Classifier* sebagai pengklasifikasi data Twitter.
d. Data yang di- *Crawling* pada Twitter berupa Bahasa Indonesia.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah peneliti mengharapkan dapat membantu pihak-pihak yang ingin menganalisa tweet oleh pengguna Twitter serta mengetahui performa dari algoritme *Naive Bayes Classifier* sebagai pengklasifikasi berdasarkan status tweet pengguna Twitter.

1.6 Metodologi Penelitian

Metode penelitian dalam penelitian membangun Implementasi Text Mining Untuk Mendeteksi *Hate Speech* pada Twitter ini menggunakan metode penelitian *waterfall.* Metode ini juga sering disebut dengan metode air terjun. Metode *waterfall* adalah permodelan suatu sistem informasi dengan cara sistematis dan urut, metode ini dimulai dari tahap pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi, dan pengujian sistem. Permodelan sistem terebut sangat cocok dimanfaatkan agar sistem tetap terjaga karena pengembangan metode ini terstruktur.

Adapun beberapa langkah penyelesaian yang dapat dilakukan dalam pembangunan sistem tersebut adalah :

a. Pengumpulan Data

   Pada pengumpulan data, peneliti akan mengumpulkan data dengan mencari referensi jurnal dan diktat yang berhubungan dengan topik penelitian ini. Pengumpulan data juga

dilakukan dengan mengumpulkan data *tweet* dengan mengambil data langsung *(Crawling data)* Twitter menggunakan API (*Application Interface*) pada Twitter.

b. Analisis Kebutuhan

Pada analisis kebutuhan, peneliti akan menentukan apa saja yang akan dibutuhkan dalam pembangunan sistem seperti kebutuhan *input, proses* dan *output.* Dalam hal ini peneliti menganalisa sentimen terhadap tweet dari pengguna Twitter terhadap suatu topik berdasarkan *hastag* sebagai sumber data. Setelah data terkumpul peneliti akan mempersiapkan *tools* pendukung untuk menganalisis sentimen seperti *Anaconda* sebagai pendistribusi *Python, Python package library* sebagai bahasa pemrograman yang peneliti gunakan, dan *library* yang dapat sebagai pendukung melakukan analisis sentimen menggunakan metode *Naive Bayes Classifier*.

c. Analisis Perancangan

Pada analisis perancangan peneliti akan membuat gambaran rancangan alur pengerjaan analisis data yang nantinya berguna mempermudah orang lain untuk memahami proses analisis data sentimen. Cara yang dilakukan peneliti yaitu dengan membuat diagram alur analisis sentimen.

d. Implementasi

Pada implementasi peneliti akan melakukan analisis kebutuhan dan analisis perancangan setelah dari proses tersebut peneliti akan mengimplementasikan kebutuhan sebagai analisis sentimen ini sesuai dengan alur yang dibuat pada saat melakukan proses analisis perancangan.

e. Pengujian Sistem dan evaluasi sistem

Pada pengujian sistem, sistem akan diuji terlebih dahulu sebelum digunakan. Hal tersebut berguna meminimalisir adanya kesalah dalam sistem. Sistem akan diuji dengan mengambil data tweet di Twitter secara *realtime* pengambilan tersebut berdasarkan fitur pencarian *hastag* pada Twitter, setelah data diambil sistem akan memberikan laporan hasil pengambilan berupa nama akun, tanggal, tweet, dan lain-lain. Setelah mendapatkan laporan hasil pengambilan data, data tersebut akan dilakukan pengujian apakah hasil dari data tersebut sesuai dengan analisis sentimen tujuan atau tidak. Pengujian ini juga dapat mengetahui kekurangan dan kelemahan analisis sentimen yang sudah berjalan. Jika dalam

proses pengujian tidak berjalan maksimal atau tidak berhasil dan tidak sesuai akan kembali lagi ke dalam proses implementasi dan dilakukan pengujian kembali.

1.7 Sistematis Penulisan

Sistematika penulisan sebagai gambaran singkat struktur penulisan laporan, serta isi setiap struktur bagian. Struktur bagian tersebut dijelaskan sebagai berikut :

BAB I PENDAHULUAN dalam bab ini berisi terkait latar belakang, rumusan masalah , tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian dan langkah penyelesaian.

BAB II LANDASAN TEORI dalam bab ini berisi terkait teori dasar yang berkaitan dengan analisis sentimen dengan menggunakan metode Naive Bayes Classifier, dan tinjuan pustaka yang nantinya sebagai dasar pengacuan pengerjaan tugas akhir.

BAB III METODOLOGI PENELITIAN dalam bab ini berisi terkait langkah-langkah penyelesaian masalah dari tahap pengumpulan data, analisis kebutuhan, dan pengimplementasian.

BAB IV HASIL DAN PEMBAHASAN dalam bab ini berisi terkait hasil dari penyelesaian masalah dari sistem di atas dan juga pembahasan sistem.

BAB V SIMPULAN DAN SARAN dalam bab ini berisi terkait rangkuman dari hasil analisis sistem kinerja pada bagian sebelumnya serta saran-saran yang perlu diperhatikan guna pengembangan sistem.

BAB II
LANDASAN TEORI

2.1 Hate Speech

Hate speech adalah suatu ujaran kebencian yang dilakukan di berbagai media, yang membuat semakin populer karena perbedaan yang sampai mewakili berbagai kelompok seperti suku, ras, etnis dan agama (Rohman, 2016). *Hate speech* ini biasanya semakin meningkat intensitasnya di media sosial menjelang pemilihan umum kepala daerah. Dasar yang paling banyak menyebabkan perselisihan atau perbedaan adalah masalah sara(suku, ras, agama diantara golongan). Kejahatan ini memilki potensi mengancam ke stabilitas negara dan keamanan. Terkait dengan permasalahan di atas pemerintah mengeluarkan aturan terkait penanganan ujaran kebencian *(Hate Speech).*

2.2 Twitter

Twitter adalah media jejaring sosial unik yang memfasilitasi penggunanya untuk dapat mengirim dan menerima terkait segala aktivitas, opini, serta segala sesuatu hal terhadap pengguna lainnya secara publik yang biasa disebut *tweet* atau juga dapat mengirim pesan secara pribadi dalam komunitas *Twitter*. Komunitas Twitter itu adalah:

a. *Following*

   Following adalah komunitas ini diartikan dengan mengikuti pengguna media jejaring sosial Twitter lainnya. pengguna juga dapat melihat *tweets* yang ditampilkan oleh semua pengguna yang diikuti. Dengan mengikuti pengguna lain di *Twitter* dapat diartikan pengguna berlangganan dengan tampilan *tweets* mereka.

b. *Followers*

   Followers adalah pengguna lain yang membaca tampilan *tweets* pengguna dan mengikuti pengguna di media jejaring sosial *Twitter*. *Followers* atau pengikut dapat melihat *tweets* yang pengguna kirim ke jejaring sosial *Twitter*.

Tweets adalah kiriman pesan singkat yang memiliki panjang yang terdiri dari 140 karakter, sehingga gampang untuk difilter (Crow Communications, 2011).

2.3 Sentimen *Analysis*

Sentiment Analysis (SA) atau biasa di sebut juga sebagai *opinion mining* adalah suatu riset komputasi nal dari emosi yang diungkapkan atau diekspresikan berupa tulisan (*tekstual*) dan *opini sentiment* (Zulfa & Winarko, 2017). *Sentiment Analysis* (SA) merupakan suatu proses untuk memahami data, mengolah data dan mengekstrak data tekstual secara otomatis dengan tujuan mendapatkan informasi sentimen atau intisari dari data yang terdapat di dalam suatu kalimat opini. *Sentiment Analysis* (SA) ini sendiri untuk melihat pendapat atau kecenderungan opini terhadap suatu masalah atau objek oleh seseorang, apakah kecenderungan tersebut mengarah ke hal positif atau negatif (Rozi, Pramono, & Dahlan, 2012).

Sentiment Analysis (SA) dibedakan berdasarkan sumber dari datanya, Adapun beberapa level yang paling banyak digunakan dalam penelitian adalah *sentiment analysis* (SA) berdasarkan level elemen dan sentimen *analysis* (SA) berdasarkan level kalimat (Falahah & Nur, 2015). Berdasarkan sumber datanya *sentiment analysis* dibagi menjadi 2 kelompok besar yaitu :

a. *Coarse-grained Sentiment Analysis*

   Sentiment analysis ini dilakukan pada level dokumen. Secara garis besar *sentiment analysis* jenis ini fokus utama dengan seluruh isi dokumen yang akan di analisis sebagai sentimen negatif dan sentimen negatif (Falahah & Nur, 2015).

b. *Fined-grained Sentiment Analysis*

   Sentiment analysis ini dilakukan pada level kalimat. Dalam *sentiment analysis* ini fokus untuk menganalisis data dari setiap kalimat (Falahah & Nur, 2015).

2.4 *Text Mining*

Text mining merupakan konsep terapan dalam teknik *data mining* untuk mencari pola inti suatu teks, dengan tujuan mendapatkan informasi yang terkandung dalam suatu teks yang dapat di manfaatkan dengan tujuan tertentu. Dari ketidak ter aturan suatu data teks dan banyaknya kandungan kata-kata imbuhan serta kiasan dalam suatu data teks, dalam proses text *mining* memerlukan tahapan-tahapan untuk mendapatkan data teks yang lebih terstruktur.

Tahapan proses yang harus di lewati *text mining* di bagi menjadi 5 bagian untuk memperoleh hasil yang diinginkan. Adapun 5 proses tersebut yang harus dijalankan dalam *text mining*, adalah (Hidayatullah, 2014):

a. *Text preprocessing*

Tahapan awal dalam *text mining* adalah *text prepocessing* dengan tujuan mempersiapkan data teks yang nantinya akan mengalami pengolahan data teks berikutnya. Selain itu biasanya dalam proses *text preprocessing* ini juga menggunakan *case folding*, yaitu pengubahan data teks pada karakter huruf besar dalam data teks menjadi huruf kecil.

b. *Text transformation*

Dalam tahap ini hasil yang di dapatkan dari proses *text preprocessing* akan dilakukan proses transformasi. Proses transformasi ini dilakukan dengan mengurangi jumlah dari setiap kata dalam data teks *stop word removal* dan mengubah kata-kata menjadi kata dasar dalam data teks *stemming.*

Stop word removal adalah suatu kata yang memiliki keunikan kata dari data teks seperti kata sambung, serta kata kepunyaan yang nantinya pada proses transformasi kata-kata tersebut tidak akan dihitung. Selain itu proses *stop word removal* dapat mengurangi beban kinerja sistem, karena kata yang akan di ambil adalah kata-kata yang dianggap penting.

Stemming adalah suatu proses dalam teks transformasi yang digunakan sebagai memproses kata-kata di dalam data teks agar menjadi kata dasar.

c. *Feature selection*

Dalam tahapan *feature selection* adalah tahapan penting dalam *text mining.* Karena dalam tahap ini dilakukan proses pembuangan beberapa *term atau kata* yang tidak terkait sehingga memperoleh *term* atau kata penting sebagai wakil kumpulan dokumen yang di analisis. Dalam *feature selection* terdapat beberapa metode yang digunakan, diantaranya adalah sebagai berikut:

1. *Document Frequency*

   Document Frequency adalah seberapa banyak kemunculan suatu *term* atau kata dalam data dokumen yang akan dianalisis.

2. *Term Frequency*

   Term frequency ($tf_{t,d}$) adalah menghitung banyaknya kemunculan *term* atau kata dalam suatu *corpus* terhadap suatu bobot *term t* atau kata pada dokumen *d.*

3. *Term Frequency-Inverse Document Frequency* (TF-IDF)

   TF-IFD itu sendiri terdiri dari *Term Frequency* dan *Inverse Document Frequency.*

d. *Pattern discovery*

Tahap *pattern discovery* berguna untuk menemukan suatu *knowledge* atau pola dengan menggunakan beberapa teknik data *mining* sebagai contoh *classification* dan *clustering.*

e. *Interpretation*

Tahapan terakhir ini adalah melakukan proses interpretasi ke sebuah bentuk kemudian di evaluasi.

2.5 *Naive Bayes Classifier*

Naïve Bayes Classifier adalah algoritme yang terdapat dalam teknik data *mining* yang menerapkan teori *Naïve Bayes* dalam klasifikasi, semua itu mendasarkan pada nilai suatu atribut secara kondisional saling bebas jika diberikan suatu nilai *output* (Ridwan, Suryono, & Sarosa, 2013). *Naïve Bayes Classifier* yaitu suatu metode pengklasifikasian berakar pada teorema *bayes*. Teorema *bayes* adalah pendekatan statistik yang fundamental dalam *pattern recoginition* (pengenalan pola).

Keuntungan menggunakan metode *Naïve Bayes Classifier* adalah metode ini hanya memerlukan nilai atau jumlah data pelatihan (*Training Data*) yang kecil sebagai penentu estimasi parameter yang nantinya diperlukan dalam proses klasifikasi data (Manalu, Sianturi, & Manalu, 2017). Berikut adalah persamaan 2.1 *Teorema Bayes* (Hidayatullah, 2014):

$$P(H|E) = \frac{P(E|H).P(H)}{P(E)} \qquad (2.1)$$

Keterangan :

E = Data yang belum diketahui *class*nya

H = Suatu *class* spesifikasi hipotesis data E

P(H|E) = *probabilitas posterior*, *probabilitas* maka akan muncul H jika diketahui E

P(E|H) = *probabilitas posterior*, *probabilitas* maka akan muncul E jika diketahui H

P(H) = *probabilitas prior*, *probabilitas* kejadian H

P(E) = *probabilitas prior*, *probabilitas* kejadian E

Peraturan dari Naïve Bayes Classifier :

Jika $P(h1|e) < P(h2|e)$, maka e dapat diklasifikasikan h2. $P(h1|e)$ mengindentifikasi probabilitas h1 berdasarkan terjadi pada kondisi e, begitu pula sebaliknya dengan h1. Klasifikasikan dari e sesuai dengan probabilitas terbesar antara probabilitas e dengan semua kelas.

2.6 *Cross Validation*

Cross Validation adalah salah satu teknik sebagai penilaian memvalidasi kan keakuratan dari suatu model yang dibuat berdasarkan dataset tertentu. Pembuatan model ini biasanya

bertujuan sebagai penentu prediksi maupun pengklasifikasian terhadap suatu data baru yang dapat dikatakan belum pernah muncul di dalam dataset. Data yang dipergunakan sebagai proses pembuatan model dapat disebut juga sebagai data latih atau data *training*, sedangkan data yang akan sebagai validasi model disebut sebagai data *test*. Salah satu metode *Cross-Validation* yang paling banyak digunakan adalah *K-Fold Cross Validation. K-fold* bekerja melipat data sebanyak K dan melakukan proses mengulang sebanyak K juga.

2.7 *Performance Evaluation Measure*

Performance Evaluation Measure (PEM) atau juga bisa disebut sebagai pengukuran evaluasi performa. Pengukuran evaluasi performa adalah suatu proses tahapan yang berguna sebagai pengukur performa suatu sistem. *Performance Evaluation Measure* ini banyak di pergunakan dalam kasus training data. Dibuatnya proses ini bertujuan untuk mengevaluasi model yang sudah dibuat. Beberapa perhitungan yang terdapat dalam *Performance Evaluation Measure* untuk menemukan nilai *Performance Evaluation Measure,* biasanya diterapkan secara parsial ataupun sebagai kombinasi. Beberapa perhitungan yang terdapat dalam *Performance Evaluation Measure* seperti (Amin , 2012):

a. *Precision*.

   Precision adalah tingkat ketepatan atau ketelitian dari hasil antara pengujian request pengguna dengan jawaban sistem.

b. *Recall*.

   Recall adalah ukuran ketepatan atau ketelitian antara informasi yang sama dengan informasi yang sudah pernah ada sebelumnya.

c. *Accuration.*

   Accuration adalah sebagai pembanding antara informasi yang dijawab oleh sistem dengan benar dengan keseluruhan informasi.

Rumus *precision* (pre) :

$$pre = \frac{TP}{TP + FP} \tag{2.2}$$

Rumus *recall (rec) :*

$$rec = \frac{TP}{TP + FN} \tag{2.3}$$

Rumus *accuration (acc) :*

$$acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.4)$$

Performance Evaluation Measure biasanya digambarkan dalam bentuk tabel atau *confusion matrix*, tabel ini berisi dari hasil pengujian model yang telah melalui proses perbandingan dengan *dataset*, tabel ini terdiri dari kelas *true* dan *false*, seperti pada Tabel 2.1.

Tabel 2.1 *Confusion Matrix*

	Class	
True Class	***Positive***	***Negative***
Positive	TP	FN
Negative	FP	TN

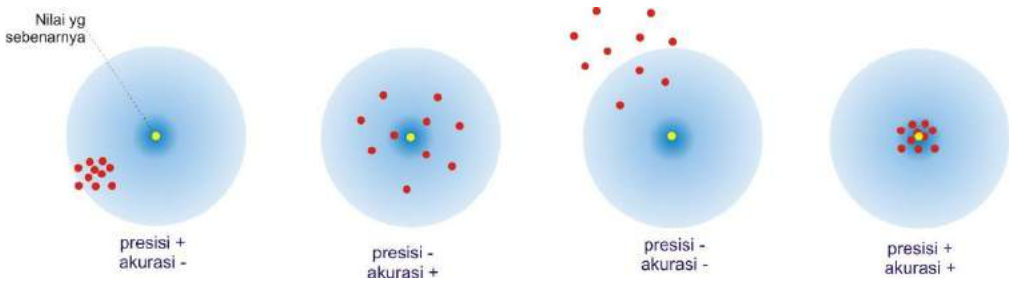
Keterangan:

TP (*true positive*) : contoh data bernilai positif yang diprediksi benar sebagai positif

TN (*true negative*) : contoh data bernilai negatif yang diprediksi benar sebagai negatif

FP (*false positive*) : contoh data bernilai negatif yang diprediksi salah sebagai positif

FN (*false negative*) : contoh data bernilai positif yang diprediksi salah sebagai negative



Gambar 2.1 Ilustrasi Gambar *Precision* dan *Accuracy*

Dari ilustrasi Gambar 2.1 di atas dapat dijelaskan gambaran persebaran data dengan *precision* dan *accuracy*. Dapat diilustrasikan dengan permisalan di bawah ini:

Misalkan peneliti ingin mengukur kinerja terhadap mesin pemisah ikan yang memiliki tugas sebagai pemisah antara ikan arwana dari semua ikan yang telah dikumpulkan oleh

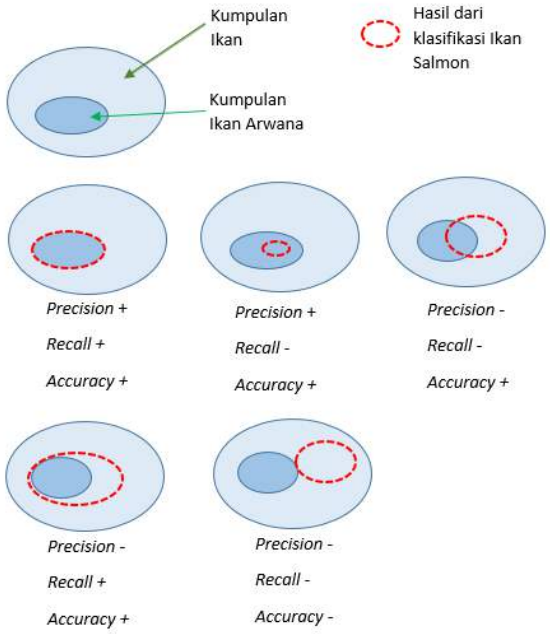
peneliti. Proses pengujiannya akan memasukkan 100 ikan arwana dan 900 adalah ikan-ikan lainnya (bukan ikan arwana). Dari proses memasukkan tadi hasil dari mesin memisahkan 110 yang terdeteksi bahwa itu adalah ikan dan hanya 90 ikan yang terdeteksi sebagai ikan arwana, sedangkan 20 lainnya adalah ikan lainnya (bukan ikan arwana), dapat diperjelas dengan melihat Tabel 2.2.

Tabel 2.2 Contoh Hasil *Confusion Matrix*

		Nilai Sebenarnya	
		True	***False***
Nilai Prediksi	***True***	90	20
	False	10	880

Dapat dilihat dari Tabel 2.2 di atas bisa dihitung dengan menggunakan persamaan (2.2), persamaan (2.3), dan persamaan (2.4) di atas. Dari kasus Tabel 2.2 di atas dapat disimpulkan bahwa mesin tersebut memiliki nilai *precision* sebesar 82%, nilai *recall* 90%, dan nilai *accuracy* sebesar 97%

Dari hasil kasus di atas bisa dapat disimpulkan gambaran seperti Gambar 2.3 di atas, apabila membandingkan dari nilai *precision,* nilai *recall* dan nilai *accuracy* :



Gambar 2.2 Perbandingan Precision, nilai Recall dan nilai Accuracy

2.8 Penelitian Serupa

Dalam pembuatan penelitian peneliti, ada beberapa penelitian sebelumnya yang sudah pernah ada dilakukan oleh orang lain yang mirip dan bahkan dijadikan sebagai acuan dari penelitian. Beberapa penelitian yang serupa dapat dilihat sebagai berikut:

a. Terdapat penelitian yang menganalisis terkait kepribadian seseorang (Sarwani & Mahmudy, 2015). kepribadian seseorang adalah hal penting untuk mengambil suatu kesimpulan atau keputusan yang berdampak baik atau buruk. Sistem ini mengambil salah satu layanan sosial yang masih bisa populer hingga saat ini yaitu *Twitter*. *T*witter hingga saat ini masih aktif menghasilkan 110 juta *tweet* per hari dan masih mempunyai lebih dari 200 juta pengguna. Dalam memproses data *Twitter* untuk menganalisis kepribadian seseorang sangat dibutuhkan metodologi yang tepat sebagai menentukan ke akuratan dari hasil. *Tweet* pada *T*witter adalah kumpulan kata yang tidak baku yang nantinya perlu diolah agar menjadi data kata yang dapat diproses. Oleh karena itu sebagai pengolahan data diperlukannya suatu proses *pre-processing* sebagai awal pengolahan kata yang kemudian akan diteruskan ke proses klasifikasi. Metode yang digunakan sebagai klasifikasi adalah metode *Naïve Bayes Classifier*. Metode tersebut dipilih karena memberikan kemudahan dan sederhana dalam proses pengolahan data serta memberikan tingkat ke akurasi yang baik. Hasil dari penelitian ini adalah menyatakan bahwa kepribadian karakter seseorang dapat diketahui dari postingan *tweet Twitter* mereka.

b. Terdapat penelitian berkaitan dengan kenaikan popularitas media jejaring sosial terus meningkat dalam beberapa tahun terakhir seperti *Twitter, Facebook,* dan *Youtube*. Salah satu dari beberapa media jejaring sosial tersebut dapat dimanfaatkan dalam bidang pemilihan umum adalah *Twitter* (Hidayatullah, 2014). Data statistik menunjukkan sejak kemunculan *Twitter* tahun 2006 terus mengalami peningkatan, *Twitter* sendiri mempunyai seratus juta lebih pengguna aktif 50 persen dari pengguna melakukan posting dan *sign in* setiap hari dengan 250 *tweets* lebih di-*posting.* Kebiasaan mem*posting tweet* pengguna mejadi salah satu sebagai acuan menentukan sentimen pengguna terhadap tokoh publik. Adapun metode yang dipergunakan sebagai mengklasifikasi data kata adalah *Naïve Bayes Classifier* dengan fitur tambahan fitur negasi berguna mengetahui negasi pada postingan *tweet*. Dengan adanya penelitian ini dapat membantu berbagai pihak yang ingin mengerti dan mengetahui tanggapan publik terkait tokoh publik yang layak untuk dapat maju sebagai pilpres dengan melalui media postingan tweet pada *Twitter*. Selain dari pada itu,

peneliti ini dapat dijadikan sebagai referensi penelitian fitur negation dalam penelitian sentimen analisis.

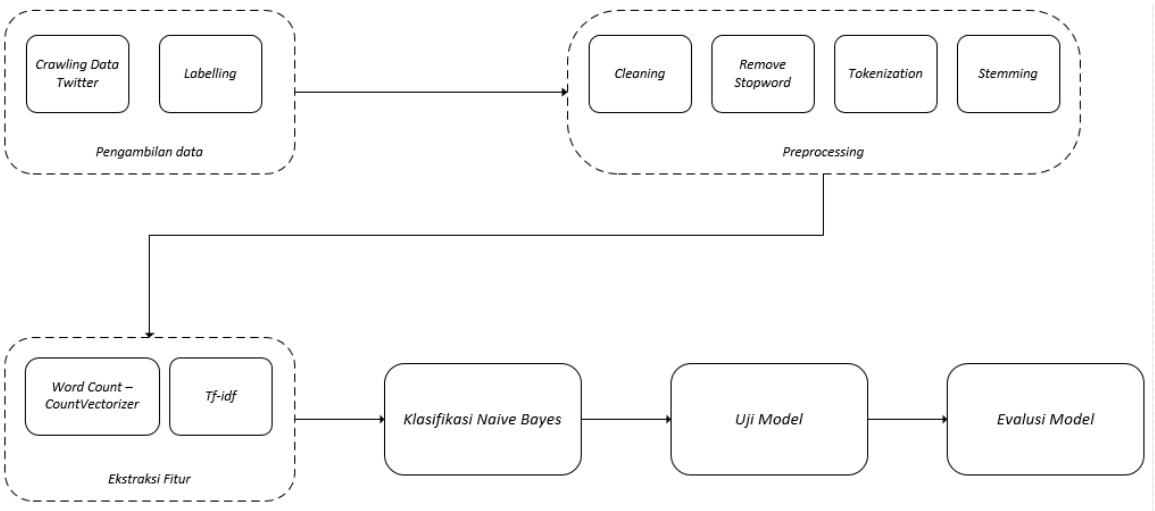
c. Penelitian menganalisis peran Twitter yang memiliki pengaruh yang sangat besar sebagai kesuksesan atau kehancuran citra seseorang. (Buntoro, 2016). Banyaknya gerakkan-gerakan yang dikerjakan di Twitter dapat mempengaruhi dari perspektif positif hingga perspektif negatif. Penelitian ini menganalisis *hastag* atau tagar pada Twitter dengan menggunakan dua sentimen yaitu *HateSpeech* dan *GoodSpeech*. Proses yang digunakan untuk menganalisis data di penelitian ini yaitu *Naïve Bayes Classifier (NBC)* dan *Support Vector Machine (SVM)* dengan mungumpulkan 522 *tweet.* Hasil akurasi tertinggi didapatkan saat menggunakan metode klasifikasi Support Vector Machine (SVM) dengan *tokenisasi unigram*, *stopword list* Bahasa Indonesia dan emoticons, dengan nilai rata-rata akurasi mencapai 66,6%, nilai presisi 67,1%, nilai recall 66,7% nilai TP rate 66,7% dan nilai TN rate 75,8%.

d. Dalam penelitian analisis sentimen terhadap media sosial, analisis sentimen adalah salah satu proses untuk menentukan emosi, opini dan sikap yang dicerminkan seseorang dari teks biasanya analisis ini untuk mengklasifikasi menjadi opini negatif dan opini positif (Cindo, Rini, & Ernitita, 2019). Selain itu analisis ini juga dapat digunakan sebagai menganalisis opini terkait produk atau layanan dan bisa juga topik tertentu di berbagai media, dalam penelitian ini menggunakan 3 objek penelitian data Twitter, Facebook, dan Web Scraping. Peneliti menggunakan metode *Naïve Bayes Classifier* dan *Support Vector Machine* pada saham perusahaan. Selain itu peneliti juga membandingkan beberapa metode seperti *logistic regression* dan *lexical-based*. Hasil akhir yang diperoleh *logistic regression* lebih unggul 93% dibandingkan dengan *Naïve Bayes Classifier* 88.20%, *SVM* 85.20%, dan *lexical-based* 92% pada tahun 2014 hingga 2018 terkait saham.

BAB III
METODOLOGI PENELITIAN

3.1 Alur Pengerjaan Tugas Akhir

Perancangan alur pengerjaan tugas akhir adalah gambaran umum terkait alur dari penelitian yang akan dilakukan dalam pengerjaan tugas akhir dari awal hingga akhir. Alur kerja dari pengerjaan tugas akhir penelitian dapat dilihat pada Gambar 3.1 berikut:



Gambar 3.1 Alur Pengerjaan Tugas Akhir

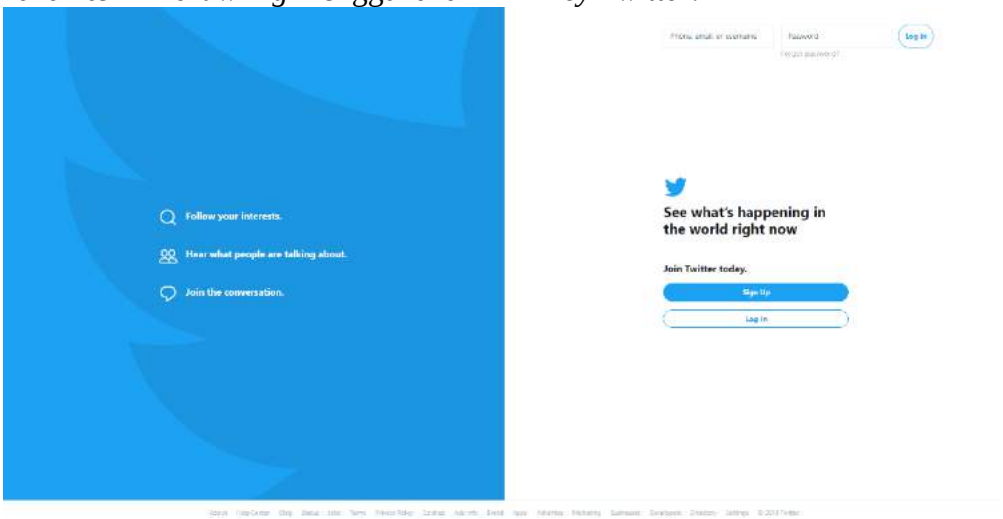
Alur pertama dalam pengerjaan tugas akhir penelitian adalah mendapatkan data dari postingan atau *tweet* pengguna Twitter dengan menggunakan teknik *crawling* data kemudia setelah semua data dikumpulkan selanjutnya tahap proses *labelling* data untuk menentukan sentimen terhadap postingan pengguna Twitter yang didapatkan. Langkah kedua, dilakukan proses *preprocessing* berguna sebagai menyeleksi data serta mengubahnya menjadi data yang lebih terstruktur. Pada proses *preprocessing* terdapat 4 tahapan yang dilakukan, yaitu *Cleaning, Remove Stopword, Tokenization* dan *Stemming.* Pada tahapan *Cleaning* berguna sebagai membersihkan kata-kata yang tidak diperlukan guna mengurangi *noise* seperti *html*, *link*, dan *script.* Selain kata-kata yang tidak perlu dihilangkan pada tahap ini juga menghilangkan tanda baca seperti titik(.), koma(,) dan juga tanda baca yang lainnya. Selain menghilangkan kata-kata dan tanda baca, pada tahan *Cleaning* juga mengubah kata menjadi

lower-case (huruf kecil) semua. Tahapan kedua adalah tahap *Remove Stopword* dalam tahap ini kata-kata yang kurang bermakna atau tidak mempunyai arti akan dilakukan penghapusan, seperti kata: saya, dan, atau. Selanjutnya masuk pada tahapan ke tiga yaitu tahap *Tokenization* digunakan sebagai indentifikasi kata-kata yang terdapat di dalam teks menjadi beberapa urutan yang terpotong oleh spasi atau juga dengan karakter spesial. Tahapan terakhir pada proses *preprocessing* adalah tahap *Stemming*, pada tahapan ini mengubah kata yang berimbuhan kembali ke kata bentuk aslinya. Langkah ketiga adalah proses ekstraksi fitur dalam proses ini dilakukan pembuatan fitur sebagai mempermudah bekerjanya proses *learning Naïve Bayes Classifier*. Langkah keempat adalah proses pengklasifikasian data menggunakan metode *Naïve Bayes Classifier* proses ini data akan diklasifikasi berdasarkan sentimen yang terdapat dalam dokumen. Setelah proses klasifikasi akan menghasilkan model yang nantinya akan dipergunakan sebagai menunjukkan ketepatan hasil pengklasifikasi. Langkah kelima adalah uji model sebagai pengukuran nilai performa pengklasifikasian yang telah dikerjakan. Langkah terakhir, setelah uji model selesai maka evaluasi model dengan cara melihat tingkat akurasi metode menggunakan *confusion matrix* dan tabel akurasi serta presisi pada setiap model.

3.2 Uraian Metodologi

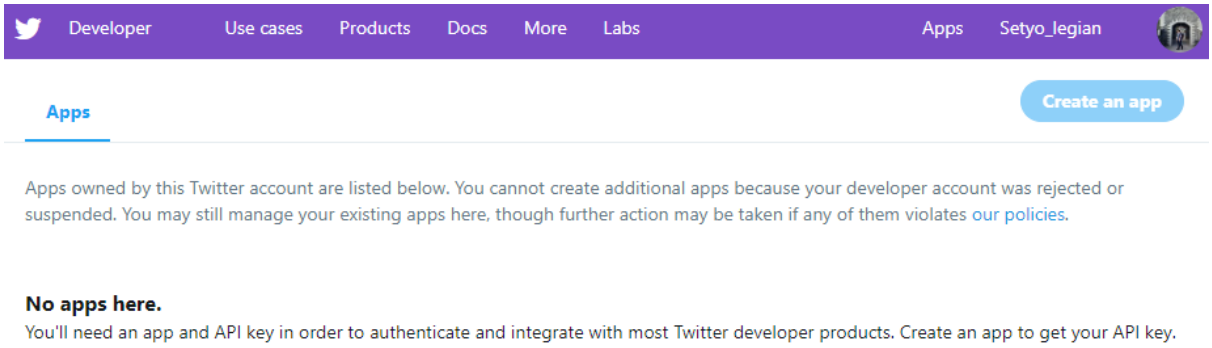
3.2.1 Pengambilan Data

Data yang digunakan dalam penelitian ini adalah data postingan pengguna Twitter yang terdapat pada situs *Twitter.com*. Data yang dikumpulkan berupa data teks yang diambil menggunakan teknik *crawling* menggunakan *API Key Twitter*.



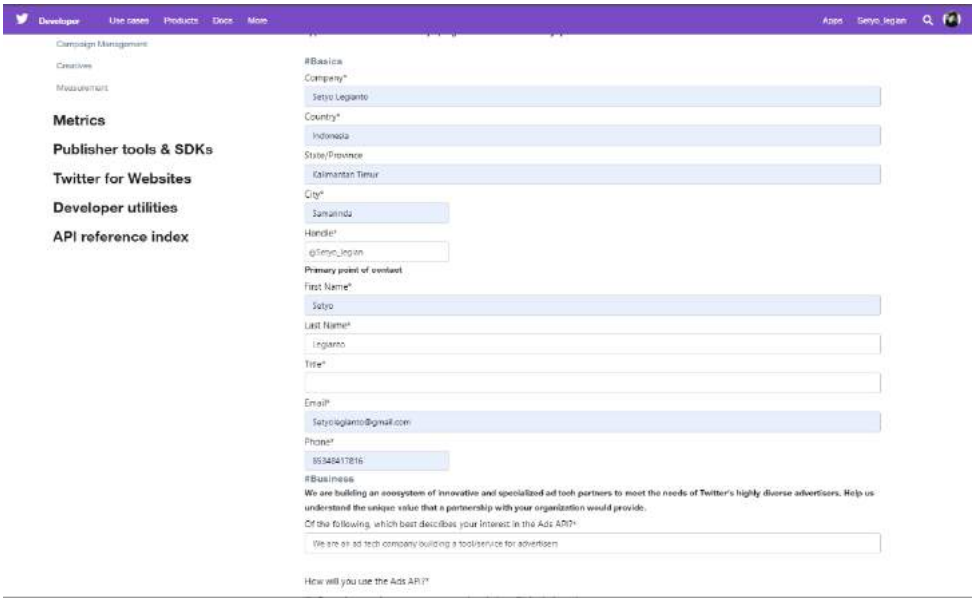
Gambar 3.2 Halaman Awal Twitter

Tahap awal untuk melakukan proses pengambilan data dari Twitter, peneliti harus memiliki *key number* dan *secret number* dari *API Key Twitter*. Mendapatkan *API Key Twitter* peneliti harus mendaftarkan atau melakukan pengajuan terhadap pihak *developer* atau pengembang Twitter untuk mendapatkan *API Key Twitter* tersebut seperti Gambar 3.3 di bawah.



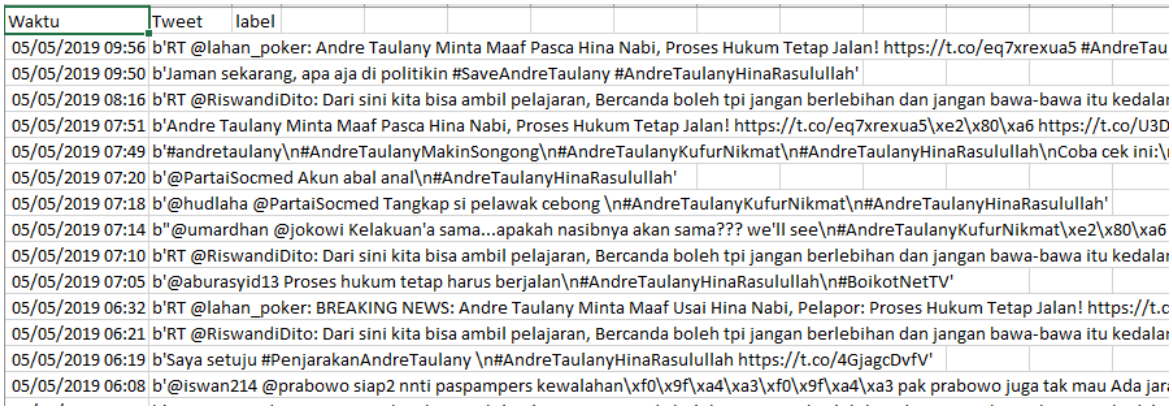
Gambar 3.3 Halaman *Developer* Twitter

Setelah mengakses website *https://developer.twitter.com/en/apps* nantinya diminta untuk mengisi form pengajuan terkait permintaan *API Key Twitter* yang berisi tentang kegunaan *API Key Twitter* yang peneliti *request.* Seperti Gambar 3.4 di bawah contoh halaman pada saat pengguna *Request API Key Twitter*.



Gambar 3.4 Halaman *Request API Key Twitter*

Setelah *API Key Twitter* didapatkan maka peneliti sudah dapat melakukan proses *crawling* data.



Waktu	Tweet	label
05/05/2019 09:56	b'RT @lahan_poker: Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5 #AndreTau	
05/05/2019 09:50	b'Jaman sekarang, apa aja di politikin #SaveAndreTaulany #AndreTaulanyHinaRasulullah'	
05/05/2019 08:16	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalar	
05/05/2019 07:51	b'Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5\xe2\x80\xa6 https://t.co/U3D	
05/05/2019 07:49	b'#andretaulany\n#AndreTaulanyMakinSongong\n#AndreTaulanyKufurNikmat\n#AndreTaulanyHinaRasulullah\nCoba cek ini:\(	
05/05/2019 07:20	b'@PartaiSocmed Akun abal anal\n#AndreTaulanyHinaRasulullah'	
05/05/2019 07:18	b'@hudlaha @PartaiSocmed Tangkap si pelawak cebong \n#AndreTaulanyKufurNikmat\n#AndreTaulanyHinaRasulullah'	
05/05/2019 07:14	b"@umardhan @jokowi Kelakuan'a sama...apakah nasibnya akan sama??? we'll see\n#AndreTaulanyKufurNikmat\xe2\x80\xa6	
05/05/2019 07:10	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalar	
05/05/2019 07:05	b'@aburasyid13 Proses hukum tetap harus berjalan\n#AndreTaulanyHinaRasulullah\n#BoikotNetTV'	
05/05/2019 06:32	b'RT @lahan_poker: BREAKING NEWS: Andre Taulany Minta Maaf Usai Hina Nabi, Pelapor: Proses Hukum Tetap Jalan! https://t.c	
05/05/2019 06:21	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalar	
05/05/2019 06:19	b'Saya setuju #PenjarakanAndreTaulany \n#AndreTaulanyHinaRasulullah https://t.co/4GjagcDvfV'	
05/05/2019 06:08	b'@iswan214 @prabowo siap2 nnti paspampers kewalahan\xf0\x9f\xa4\xa3\xf0\x9f\xa4\xa3 pak prabowo juga tak mau Ada jar	

Gambar 3.5 Hasil *Crawling* Data

Hasil dari proses *crawling* data dalam file excel pada Gambar 3.5 di atas yang nantinya akan proses *labelling* data untuk menentukan klasifikasi pendapat atau pandangan dari hasil tweet yang telah di *crawling* tadi. Pada proses *labelling* ini dibedakan menjadi 2 kelas. Yaitu *class positif* dan *class negative.* Contoh dari proses *labelling* data seperti Tabel 3.1 di bawah.

Tabel 3.1 Contoh Data Hasil *Labelling*

Tweet	**Clean Text**	**Label**
b'RT @lahan_poker: Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5 #AndreTaulanyKufurNikmat #Andr\xe2\x80\xa6'	andre taulany minta maaf pasca hina nabi proses hukum tetap jalan andretaulanykufur nikmat andr	0
b'Kontol kau Andre ...anjing...biadab....taik babi ko memang udah bauk haram lagi #AndreTaulanyHinaRasulullah'	kontol kau andre anjing biadab taik babi ko udah bauk haram andretaulanyhinara sulullah	1
b'@detikcom Cebong bangsat\n#AndreTaulanyHinaRasulullah'	cebong bangsat andretaulanyhinara sulullah	1

b'baperan. habib kok baperan. harusnya lbh kenal agama lbh bijaksana #AndreTaulanyHinaRasulullah https://t.co/t9GZ7T5v39'	baperan habib baperan kenal agama bijaksana andretaulanyhinara sulullah	1
b'@aburasyid13 Proses hukum tetap harus berjalan\n#AndreTaulanyHinaRasulullah\n#BoikotNetTV'	proses hukum tetap jalan andretaulanyhinara sulullah boikotnettv	0

Dalam kasus ini *class positif* label 1 menyatakan bahwa tweet tersebut adalah kata-kata yang mengandung unsur *hatespeech* atau ujaran kebencian, sedangkan *class negative* berlabel kan 0 adalah kata-kata yang netral atau tidak mengandung unsur *hatespeech.*

3.2.2 Tahapan Proses *Preprocessing*

Preprocessing adalah tahapan proses untuk membersihkan data dari kata-kata atau tweet yang tidak di perlukan serta kata-kata yang tidak memiliki makna. Proses ini dilakukan sesuai dengan isi data dari proses pengambilan data atau *crawling* data Twitter. Adapun proses beberapa langkah dari proses *preprocessing* memilki urutan sebagai berikut :

a. *Cleaning*

Cleaning adalah proses penghapusan simbol, tanda baca, huruf kapital dan bilangan angka yang sering muncul pada tweet pengguna Twitter sehingga data tersebut menjadi data yang tidak efektif dan tidak memiliki arti. Proses ini dijalankan menggunakan program, sehingga *cleaning* ini berjalan secara otomatis sebelum menyimpan hasil *decode* dalam bentuk file excel. Contoh penerapan proses *cleaning* dapat dilihat seperti Tabel 3.2 di bawah.

Tabel 3.2 Contoh Hasil *Cleaning*

Tweet* Sebelum *Cleaning	***Tweet* Sesudah *Cleaning***
b'RT @lahan_poker: Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5 #AndreTaulanyKufurNikmat #Andr\xe2\x80\xa6'	andre taulany minta maaf pasca hina nabi proses hukum tetap jalan andretaulanykufurnikmat andra
b'Kontol kau Andre ...anjing...biadab....taik babi ko memangnya udah bauk haram lagi #AndreTaulanyHinaRasulullah'	kontol kau andre anjing biadab taik babi ko memangnya udah bauk

	haram lagi andretaulanyhinarasulullah
b'@detikcom Cebong bangsat\n#AndreTaulanyHinaRasulullah'	cebong bangsat andretaulanyhinarasulullah
b'baperan. habib kok baperan. harusnya lbh kenal agama lbh bijaksana #AndreTaulanyHinaRasulullah https://t.co/t9GZ7T5v39'	baperan habib baperan kenal agama bijaksana andretaulanyhinarasulullah
b'@aburasyid13 Prosesnya hukum tetap harus berjalan\n#AndreTaulanyHinaRasulullah\n#BoikotNetTV'	prosesnya hukum tetap jalan andretaulanyhinarasulullah boikotnettv

b. *Remove Stopword*

Remove Stopword adalah proses pehapusan kata-kata yang kurang bermakna atau kata yang tidak memiliki arti seperti kata dan, atau, kamu, saya. Contoh proses penerapan pada tahap *Remove Stopword* dapat dilihat pada Tabel 3.3 di bawah.

Tabel 3.3 Contoh Data Hasil *Remove Stopword*

Tweet Sebelum *Remove Stopword*	**Tweet Sesudah *Remove Stopword***
andre taulany minta maaf pasca hina nabi proses hukum tetap jalan andretaulanykufurnikmat andra	andre taulany minta maaf pasca hina nabi proses hukum tetap jalan andretaulanykufurnikmat andra
kontol kau andre anjing biadab taik babi ko memangnya udah bauk haram lagi andretaulanyhinarasulullah	kontol kau andre anjing biadab taik babi ko udah bauk haram andretaulanyhinarasulullah
cebong bangsat andretaulanyhinarasulullah	cebong bangsat andretaulanyhinarasulullah

baperan habib kok baperan harus lbh kenal agama lbh bijaksana andretaulanyhinarasulullah	baperan habib baperan kenal agama bijaksana andretaulanyhinarasulullah
prosesnya hukum tetap harus jalan andretaulanyhinarasulullah boikotnettv	prosesnya hukum tetap jalan andretaulanyhinarasulullah boikotnettv

c. *Tokenization*

Tokenization adalah proses untuk memecahkan kalimat untuk menjadi beberapa bagian yang dinamakan *token*. Sebuah *token* dapat dianggap menjadi satu bentuk sebuah kata, frasa, atau suatu elemen yang berarti. Contoh proses pada tahap *tokenization* dapat dilihat pada Tabel 3.4 di bawah.

Tabel 3.4 Contoh Data Hasil *Tokenization*

Tweet Sebelum *Tokenization*	**Tweet Sesudah *Tokenization***
andre taulany minta maaf pasca hina nabi proses hukum tetap jalan andretaulanykufurnikmat andra	[‘andre’,‘taulany’, ‘minta’,‘maaf’,‘pasca’, ‘hina’,‘nabi’,‘proses’,‘hukum’, ‘tetap’,‘jalan’, ‘andretaulanykufurnikmat’,‘andra’]
kontol kau andre anjing biadab taik babi ko udah bauk haram andretaulanyhinarasulullah	[‘kontol’,‘kau’,‘andre’,‘anjing’,‘biadab’,‘taik’,‘babi’,‘ko’,‘udah’,‘bauk’,‘haram’,‘andretaulanyhinarasulullah’]
cebong bangsat andretaulanyhinarasulullah	[‘cebong’,‘bangsat’,‘andretaulanyhinarasulullah’]
baperan habib baperan kenal agama bijaksana andretaulanyhinarasulullah	[‘baperan’,‘habib’,‘baperan’,‘kenal’,‘agama’,‘bijaksana’,‘andretaulanyhinarasulullah’]
prosesnya hukum tetap jalan andretaulanyhinarasulullah boikotnettv	[‘prosesnya’,‘hukum’,‘tetap’,‘jalan’,‘andretaulanyhinarasulullah’,‘boikotnettv’]

d. *Stemming*

Stemming adalah proses untuk merubah kata kedalam bentuk dasar dengan menghapus kata imbuhannya di depan maupun di belakang kata. Contoh proses penerapan *stemming* dapat dilihat pada Tabel 3.5 di bawah.

Tabel 3.5 Contoh Data Hasil *Stemming*

Tweet Sebelum *Stemming*	**Tweet Sesudah *Stemming***
[‘andre’,‘taulany’, ‘minta’,‘maaf’,‘pasca’, ‘hina’,‘nabi’,‘proses’,‘hukum’, ‘tetap’,‘jalan’, ‘andretaulanykufurnikmat’,‘andra’]	andre taulany minta maaf pasca hina nabi proses hukum tetap jalan andretaulanykufurnikmat andra
[‘kontol’,‘kau’,‘andre’,‘anjing’,‘biadab’,‘taik’,‘babi’,‘ko’,‘udah’,‘bauk’,‘haram’,‘andretaulanyhinarasulullah’]	kontol kau andre anjing biadab taik babi ko udah bauk haram andretaulanyhinarasulullah
[‘cebong’,‘bangsat’,‘andretaulanyhinarasulullah’]	cebong bangsat andretaulanyhinarasulullah
[‘baperan’,‘habib’,‘baperan’,‘kenal’,‘agama’,‘bijaksana’,‘andretaulanyhinarasulullah’]	baperan habib baperan kenal agama bijaksana andretaulanyhinarasulullah
[‘prosesnya’,‘hukum’,‘tetap’,‘jalan’,‘andretaulanyhinarasulullah’,‘boikotnettv’]	proses hukum tetap jalan andretaulanyhinarasulullah boikotnettv

3.2.3 Ekstraksi Fitur

Dalam kasus ini setelah semua tweet dikumpulkan dari hasil *crawling* dan hasil data tersebut telah melewati proses *preprocessing*, proses selanjutnya adalah membuat fitur yang berguna untuk mempermudah proses pengklasifikasian data tweet tersebut biasa proses ini dibuat dengan proses ekstraksi fitur. Pada proses ekstraksi fitur ini terdapat dua proses yang dilakukan yaitu proses pembuatan *word vector* dalam proses ini sistem akan mengubah suatu teks menjadi representasi *vector* dan proses pembobotan kata menggunakan *Tf-idf(term*

frequency–inverse document frequency). Word vector ini sendiri dapat diartikan dalam Bahasa Indonesia sebagai vektor kata proses pembuatan kalimat yang sudah ada menjadi sekumpulan *array* menjadi suatu matriks, setiap baris matriks tersebut mewakili dari baris dokumen, sedangkan setiap kolom matriks akan mewakili seluruh kata yang terdapat di dalam teks dari suatu data tweet. Setalah semua kata diproses dan berubah menjadi vektor kata, selanjutnya adalah proses pemberian bobot dari setiap kata pada setiap kalimat atau dokumen menggunakan metode *Uni gram* dan *Tf-idf(term frequency–inverse document frequency)* menggunakan rumus yang sudah dijelaskan pada bab sebelumnya. Jika proses pembobotan selesai maka *dataset* dapat digunakan dalam *training* menggunakan perhitungan *Naïve Bayes Classifier.*

3.2.4 Klasifikasi

Dalam kasus ini peneliti menggunakan Metode *Naïve Bayes Classifier* sebagai metode pengklasifikasian data Twitter untuk mendapatkan sentimen analisis. Untuk melakukan proses pengklasifikasian ini diperlukan hasil dari data yang sudah diolah dari proses sebelumnya yaitu hasil dari proses *preprocessing* dan hasil dari pembobotan kata dengan *Tf-idf.* Setelah data berhasil di- *training* maka akan dilakukan proses pengujian menggunakan data*test* sebagai pengujian hasil ketepatan klasifikasi yang dilakukan.

Selain menggunakan metode *Naïve Bayes Classifier* dalam penelitian ini juga menggunakan metode *logistic regression model.* Metode *logistic regression model* ini digunakan sebagai pembanding terhadap metode *Naïve Bayes Classifier*. *Logistic regression model* adalah suatu metode regresi (metode melihat pengaruh antara dua atau lebih variabel), *Logistic regression* ini menghubungkan antara satu atau beberapa variabel bebas (variabel *independen*) dengan variabel *dependen* yang kategori variabel ini 0 dan 1.

3.2.5 Uji Model

Proses uji model dapat dilakukan ketika proses dari *training* data telah selesai dikerjakan. Pengujian model ini dilakukan untuk mengetahui bagaimana kinerja model. Jumlah data yang dijadikan sebagai bahan pengujian diambil dari data *training* sebesar 33% sama dengan 0,33. Pengambilan data ini dilakukan secara *random* dengan menggunakan bantuan *library* dari *Phyton.* Setelah semua proses selesai maka sistem akan menampilkan besar akurasi dari model yang dikerjakan.

3.2.6 Evaluasi Model

Evaluasi model ini berguna untuk mengetahui tingkat keakurasian dari kinerja model. Untuk mendapatkan tingkat keakurasian dari kinerja model dalam kasus ini digunakan metode *confusion Matrix* dan tabel akurasi serta melihat presisi untuk setiap model. Setalah *datatest* dilakukan pengujian terhadap data *training,* maka akan menghasilkan beberapa kelas dari *datatest,* biasa disebut prediksi kelas. Kemudian prediksi kelas tadi yang sebenarnya berasal dari *datatest* sebelumnya tadi disembunyikan, sehingga dapat ditampilkan dan dihitung nilai dari *accuration*, *precision*, *recall*, dan *f1-score.*

BAB IV
HASIL DAN PEMBAHASAN

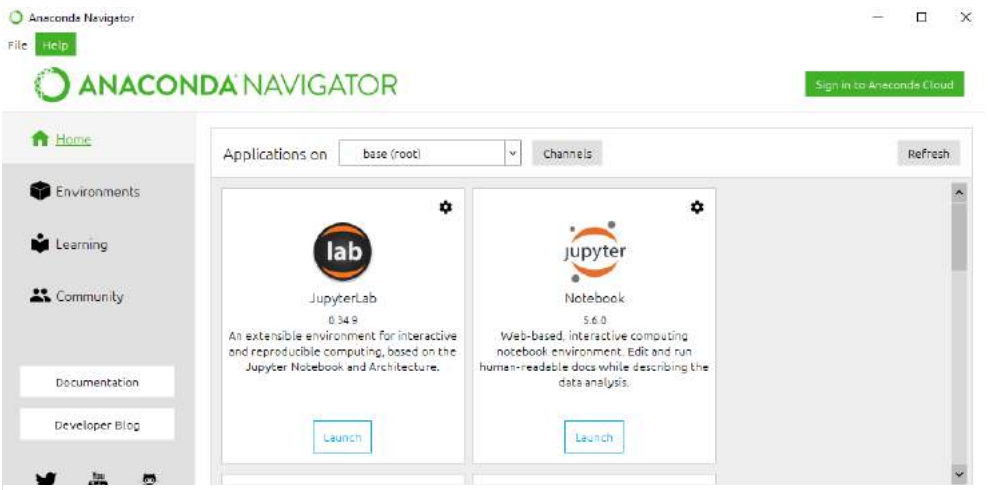
4.1 Pengambilan Data (*Crawling* data)

Pengambilan data ini atau proses *crawling* data Twitter ini menggunakan *API Key Twitter* dan proses pengambilan data Twitter dibantu dengan bahasa pemrograman *phyton. API Key Twitter* adalah *Application Programming Interface(API)* dalam *API* ini suatu layanan berisi sekumpulan perintah, fungsi, komponen dan juga protokol yang disediakan untuk mempermudah programme pada saat membangun suatu sistem perangkat lunak. *API Key Twitter* itu sendiri memiliki suatu *consumer keys, consumer secret, access key, dan accsess secret. Consumer keys, access key, dan accsess secret* tersebut digunakan untuk mengakses data Twitter yang dibutuhkan oleh programme pada Gambar 4.1 di bawah.

```
consumer_key = "I0E7xGaPsVIIyg0sq8tfw"
consumer_secret = "frk7rjhTVHEuRErf4V2h93xZ6eSAr2myy9gH4RaU"
access_key = "228245421-FPyC4SFufgkHDmHy7gTpEqm36mbFepYmkQ2p54xf"
access_secret = "cYc2xsbeKiTAlb8RuQ9btHZxERGNoFrlazKBPTnQ"
```

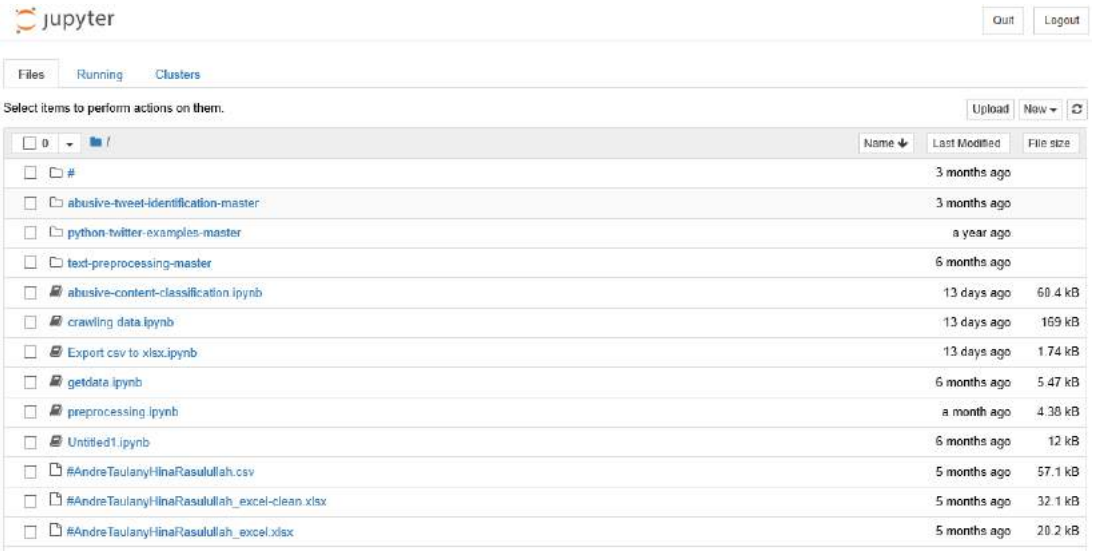
Gambar 4.1 *API Key Twitter*

Selain membutuhkan *API Key Twitter* untuk dapat melakukan pengambilan data Twitter disini peneliti menggunakan *tools* pendukung untuk menganalisis sentimen seperti *Anaconda* sebagai pendistribusi *Python*. Tampilan dari *anaconda* pada Gambar 4.2 di bawah.



Gambar 4.2 Tampilan *Anaconda*

Di sini peneliti menggunakan *Anaconda* sebagai *tools* karena di dalam *anaconda* sudah terdapat *Jupyter Notebook.* *Jupyter Notebook* biasa juga disebut *jupyter* ini adalah pengembangan dari *Ipython* atau *Interactive Python.* *Jupyter Notebook* ini suatu editor dalam bentuk web aplikasi yang berjalan di *localhost* komputer, adapun beberapa hal yang dapat dilakukan oleh *Jupyter Notebook* seperti menulis kode *python, equations, visualisasi* dan bisa juga sebagai *markdown editor*. Tampilan dari *Jupyter Notebook* pada Gambar 4.3 di bawah.



Gambar 4.3 Tampilan *Jupyter Notebook*

Dengan adanya *Jupyter Notebook* sekarang peneliti dapat melakukan proses pengkodingan menggunakan bahasa pemrograman *Python* seperti Gambar 4.4 di bawah.

```
In [5]: import tweepy
        import csv
        import pandas as pd
        from pandas import ExcelWriter

        #Twitter API credentials

        consumer_key = "I0E7xGaPsVIIyg0sq8tfw"
        consumer_secret = "frk7rjhTVHEuRErf4V2h93xZ6eSAr2myy9gH4RaU"
        access_key = "228245421-FPyC4SFufgkHDmHy7gTpEqm36mbFepYmkQ2p54xf"
        access_secret = "cYc2xsbeKiTAlb8RuQ9btHZxERGNoFrlazKBPTnQ"
```

Gambar 4.4 *Source Code* Pemanggilan Python *Library* Proses *Crawling*

Pada Gambar 4.4 di atas proses pendeklarasian *library tweepy, library csv, library pandas. library tweepy* adalah suatu API yang disediakan oleh pihak Twitter untuk dapat mengakses dan mengambil data-data yang ada di dalam Twitter menggunakan bahasa pemrograman Phyton. *Library Csv* (*Command Separated Values)* adalah *library* yang menyediakan layanan

baca dan menulis suatu data bertipe file csv atau excel. *Library pandas* adalah library pada Phyton yang berguna untuk pengolahan data. Setelah itu masukkan *consumer keys, consumer secret, access key, dan accsess secret* yang telah didapatkan pada Gambar 4.1 di atas. Setelah memasukkan kode di atas, selanjutnya memasukkan kode proses pada Gambar 4.5 di bawah.

```
In [6]: auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
        auth.set_access_token(access_key, access_secret)
        api = tweepy.API(auth,wait_on_rate_limit=True)

In [7]: csvFile = open('#andretaulanykufurnikmat.csv', 'a')
        csvWriter = csv.writer(csvFile)

In [ ]: for tweet in tweepy.Cursor(api.search,q="#andretaulanykufurnikmat",count=500,
                                  lang="id").items():
            print (tweet.created_at, tweet.text)
            csvWriter.writerow([tweet.created_at, tweet.text.encode('utf-8')])
```

Gambar 4.5 *Source Code* Proses *Crawling*

Setelah menjalankan semua kode di atas maka akan didapat file excel yang belum diolah. Seperti Gambar 4.6 di bawah contoh file dari hasil *crawling* data.

Waktu	Tweet	label
05/05/2019 09:56	b'RT @lahan_poker: Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5 #AndreTau	
05/05/2019 09:50	b'Jaman sekarang, apa aja di politikin #SaveAndreTaulany #AndreTaulanyHinaRasulullah'	
05/05/2019 08:16	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalar	
05/05/2019 07:51	b'Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5\xe2\x80\xa6 https://t.co/U3D	
05/05/2019 07:49	b'#andretaulany\n#AndreTaulanyMakinSongong\n#AndreTaulanyKufurNikmat\n#AndreTaulanyHinaRasulullah\nCoba cek ini:\	
05/05/2019 07:20	b'@PartaiSocmed Akun abal anal\n#AndreTaulanyHinaRasulullah'	
05/05/2019 07:18	b'@hudlaha @PartaiSocmed Tangkap si pelawak cebong \n#AndreTaulanyKufurNikmat\n#AndreTaulanyHinaRasulullah'	
05/05/2019 07:14	b"@umardhan @jokowi Kelakuan'a sama...apakah nasibnya akan sama??? we'll see\n#AndreTaulanyKufurNikmat\xe2\x80\xa6	
05/05/2019 07:10	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalar	
05/05/2019 07:05	b'@aburasyid13 Proses hukum tetap harus berjalan\n#AndreTaulanyHinaRasulullah\n#BoikotNetTV'	
05/05/2019 06:32	b'RT @lahan_poker: BREAKING NEWS: Andre Taulany Minta Maaf Usai Hina Nabi, Pelapor: Proses Hukum Tetap Jalan! https://t.c	
05/05/2019 06:21	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalar	
05/05/2019 06:19	b'Saya setuju #PenjarakanAndreTaulany \n#AndreTaulanyHinaRasulullah https://t.co/4GjagcDvfV'	
05/05/2019 06:08	b'@iswan214 @prabowo siap2 nnti paspampers kewalahan\xf0\x9f\xa4\xa3\xf0\x9f\xa4\xa3 pak prabowo juga tak mau Ada jar	

Gambar 4.6 File Excel Hasil *Crawling*

Pada saat proses pengambilan data Twitter, peneliti mengambil 3 sumber *Hastag* yang pada saat proses *crawling* data berada pada posisi *tranding topic* Twitter atau pada posisi pembahasan terbanyak pada tweet yaitu #andretaulanyhinarasulullah berjumlah 464 tweet pada tanggal 05/05/2019, #andretaulanykufurnikmat berjumlah 417 tweet pada tanggal 05/05/2019, #C1PlanoBabinsaAdalahKunci berjumlah 445 tweet pada tanggal 05/05/2019, anjing berjumlah 499 tweet pada tanggal 26/08/2019, babi berjumlah 398 tweet pada tanggal 01/09/2019, monyet berjumalah 277 tweet pada tanggal 01/09/2019. Semua data yang diambil berjumlah 2500 tweet.

Setelah proses *crawling* data selesai dan berhasil disimpan dalam bentuk file dokumen excel, selanjutnya masuk pada proses pelabelan data tweet. Dalam proses pelabelan data tweet ini peneliti menentukan nilai dari tweet pada Twitter yang bernilai positif dalam artian tweet tersebut mengandung kata-kata *hate speech* (kata-kata yang mengandung unsur hinaan atau ujaran kebencian, berkata kasar, membawa suku, ras, dan agama), dan bernilai negatif tweet tersebut tidak mengandung kata *hate specch.* Dalam proses labelling data ini seharusnya untuk menentukan nilai tersebut dilakukan oleh pakar bahasa atau yang terkait dibidangnya dan juga setidaknya membutuhkan dua orang atau lebih untuk menghindari perbedaan pendapat yang sama dalam menentukan sentimen tweet yang terdapat dari hasil data *crawling.* Namun demikian disini pada saat proses labelling peneliti hanya dilakukan dengan tiga orang mahasiswa yang bukan pakar dari bahasa ataupun psikologi dibidangnya. Konsekuensi dari penelitian dalam proses labelkan data ini adalah terdapatnya beberapa tweet yang tidak sama ataupun sesuai dengan sentimen analisis sebenarnya. Karena proses pelabelan data Twitter ini dilakukan secara manual, dan juga dalam proses pelabelan ini secara manual membutuhkan waktu yang tidak cepat, hal tersebut menjadi kelemahan pada saat jumlah data yang akan di proses pelabelan sangat banyak, seperti pada Gambar 4.7 di bawah menunjukkan hasil data yang sudah proses *labelling*.

Waktu	Tweet	label
2019-05-05 09:2	b'RT @lahan_poker: Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5 #AndreTaulanyKufurNikmat #Andr\xe2\x80\xa6'	0
2019-05-05 09:0	b'laman sekarang, apa aja di politikin #SaveAndreTaulany #AndreTaulanyHinaRasulullah'	1
2019-05-05 08:3	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalam keagamaan, sem\xe2\x80\xa6'	1
2019-05-05 07:3	b'Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5\xe2\x80\xa6 https://t.co/U3DCyCb0rY'	0
2019-05-05 07:3	b'#andretaulany\n#AndreTaulanyMakinSongong\n#AndreTaulanyKufurNikmat\n#AndreTaulanyHinaRasulullah\nCoba cek ini:\n\nAndre T\xe2\x80\xa6 https://t.co/2xcRkXGywW'	1
2019-05-05 07:3	b'@PartaiSocmed Akun abal anafy\n#AndreTaulanyHinaRasulullah'	1
2019-05-05 07:3	b'@hudiaha @PartaiSocmed Tangkap si pelawak cebong \n#AndreTaulanyKufurNikmat\n#AndreTaulanyHinaRasulullah'	1
2019-05-05 07:3	b"@umardhan @jokowi Kelakuan'2 sama...apakah nasibnya akan sama2?? we'll see\n#AndreTaulanyKufurNikmat\xe2\x80\xa6 https://t.co/3c9uW4oxtj"	1
2019-05-05 07:3	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalam keagamaan, sem\xe2\x80\xa6'	1
2019-05-05 07:2	b'@oburasyid13 Proses hukum tetap harus berjalan\n#AndreTaulanyHinaRasulullah\n#BoikotNetTV'	0
2019-05-05 06:3	b'RT @lahan_poker: BREAKING NEWS: Andre Taulany Minta Maaf Usai Hina Nabi, Pelapor: Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5 #SaveAn\xe2\x80\xa6'	0
2019-05-05 06:3	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalam keagamaan, sem\xe2\x80\xa6'	1
2019-05-05 06:3	b'Saya setuju #PenjarakanAndreTaulany \n#AndreTaulanyHinaRasulullah https://t.co/46jagcDvfV'	0
2019-05-05 06:3	b'@iswan214 @prabowo siap2 ont pospampers kewalahan\xf0\x9f\xa4\xa3\x80\x9f\xa4\xa3 pak prabowo juga tak mau Ada jarak antaru beliau dgn rakyatnya\xe2\x80\xa6 https://t.co/KPlbBQNyvB'	0
2019-05-05 06:3	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalam keagamaan, sem\xe2\x80\xa6'	1
2019-05-05 06:3	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalam keagamaan, sem\xe2\x80\xa6'	1
2019-05-05 06:3	b'BREAKING NEWS: Andre Taulany Minta Maaf Usai Hina Nabi, Pelapor: Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5\xe2\x80\xa6 https://t.co/fVAiRfjqFE'	0
2019-05-05 05:3	b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalam keagamaan, sem\xe2\x80\xa6'	1
2019-05-05 05:3	b'Andre Taulany menghina Kanjeng Nabi?\nTrus orang ini apa? PeMain MMA tah? Atau lenong? Orang yg gk sengaja km spont\xe2\x80\xa6 https://t.co/VYK21cjoN'	1
2019-05-05 05:3	b'Enak Betul Minta Maaf \nAhmad Dani aja di Bui cuma bilang Idiot. Ini jelas2 menghina Nabi Muhammad SAW masa gak ada\xe2\x80\xa6 https://t.co/LvkSD35KMR'	1
2019-05-05 05:3	b'@reltyWF @KPU_ID @bawaslu_RI gak bakal berani terima tantangan. \n\nIINAelectionVictim440KPPS\xe2\x80\xa6 https://t.co/cyWShfdXD5'	0
2019-05-05 04:3	b'RT @Anonymousreceh: Ayo TKN mana janji mu 100M nya, bakso aja kagak buyar belagu\n\n#C1PlanoBabinsaAdalahKunci\n#AndreTaulanyKufurNikmat \n#Im\xe2\x80\xa6'	1
2019-05-05 04:3	b'RT @Anonymousreceh: Ayo TKN mana janji mu 100M nya, bakso aja kagak buyar belagu\n\n#C1PlanoBabinsaAdalahKunci\n#AndreTaulanyKufurNikmat \n#Im\xe2\x80\xa6'	1
2019-05-05 04:3	b'RT @Anonymousreceh: Ini pada kemana hey 100M nya kemana bong cebong\n\n#C1PlanoBabinsaAdalahKunci\n#AndreTaulanyKufurNikmat \n#InvestigasiFatal\xe2\x80\xa6'	1
2019-05-05 04:3	b'Minta maaf sih harus. Tapi proses hukum mestinya jalan terus. Bercanda jangan berlebihan, apalagi diproduksi sebagai\xe2\x80\xa6 https://t.co/fNbwF3ftl'	0
2019-05-05 03:3	b'Proses hukum tetap jalan...\n#AndreTaulanyHinaRasulullah https://t.co/kmgJ4PJ22v'	0
2019-05-05 02:3	b'Mas Andre dtg dan menemui Kyai Cholil Nafis MUI Pusat di MUI pusat. Semoga semua kita makin berhati2 dlm berlisan.\xe2\x80\xa6 https://t.co/PQa69HiIF4'	0
2019-05-05 01:3	b'@erlex_wu Saya pribadi sempat saat ini,demi Allah belum bisa memaafkan manusia satu ini !!!\xe2\x80\xa6 https://t.co/ITyHnc7hVo'	1
2019-05-05 00:3	b'hukum harus tetap ditegakkan \xe2\x9c\x8a\xf0\x9f\x91\xa5\n\n#AndreTaulanyHinaRasulullah https://t.co/d3gRnAdc8'	0
2019-05-05 00:3	b'Harusnya MUELIS ULAMA INDONESIA buat penjara sendiri, biar menjarain orang2 penista agama, susah berharap ke pemer\xe2\x80\xa6 https://t.co/hu9dBTETgx'	1
2019-05-05 00:3	b'@putrabanten80 Sorry. Dah kadung ane haramkan nonton acara ente, apapun bentuknya.\n\n#AndreTaulanyKufurNikmat\xe2\x80\xa6 https://t.co/StxIYJNin2'	1

Gambar 4.7 Hasil data *crawling* dan *labelling*

4.2 *Preprocessing*

Pada saat proses *preprocessing* ini terdiri dari beberapa tahapan untuk menjadikan kalimat pada tweet menjadi bahasa yang baku, karena tidak sepenuhnya tweet tersebut hasil *crawling* menggunakan kata baku. Selain itu proses ini berguna sebagai untuk menghilangkan beberapa

bagian dari kalimat yang tidak berguna. Proses *preprocessing* ini dikerjakan menggunakan bantuan dari *library* pada bahasa pemrograman *Python 3*. Untuk mengerjakan proses *preprocessing* terdapat 4 tahapan proses untuk memperoleh hasil yang maksimal, sebagai berikut:

a. *Cleaning*

Pada proses *cleaning* ini berguna untuk mengurangi atau membersihkan data tweet dari kata atau kalimat yang tidak diperlukan seperti tanda baca, *unicode*, dan lain-lain. Proses *cleaning* ini terdapat 4 tahapan yang akan dilakukan oleh sistem untuk memperoleh hasil yang maksimal, seperti di bawah ini:

1. Membersihkan tanda baca
2. Membersihkan angka
3. Merubah huruf besar menjadi huruf kecil semua
4. Membersihkan kelebihan spasi

Beberapa kode program yang mengimplementasikan *cleaning* data dapat dilihat pada Gambar 4.8 di bawah.

```
def cleaning(str):
    #remove non-ascii
    str = unicodedata.normalize('NFKD', str).encode('ascii', 'ignore').decode('utf-8', 'ignore')
    str = re.sub("b'|b\"",'',str)
    #remove URLs
    str = re.sub(r'(?i)\b((?:https?://|www\d{0,3}[.]|[a-z0-9.\-]+[.][a-z]{2,4}/)(?:[^\s()<>]+|\(([^\s()<>]+
    #remove punctuations
    str = re.sub(r'[^\w]|_',' ',str)
    #remove digit from string
    str = re.sub("\S*\d\S*", "", str).strip()
    #remove digit or numbers
    str = re.sub(r"\b\d+\b", " ", str)
    #to lowercase
    str = str.lower()
    #Remove additional white spaces
    str = re.sub('[\s]+', ' ', str)

    return str
```

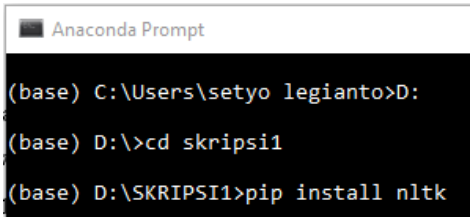
Gambar 4.8 kode program proses *cleaning*

Pada Gambar 4.8 di atas, keseluruhan dari proses *cleaning* dilakukan dengan menggunakan *regex* atau bisa juga disebut *regular expression.* Regex itus sendiri adalah konstruksi dalam suatu bahasa untuk mencocokkan teks berdasarkan pola tertentu, terutama untuk kasus-kasus kompleks.

b. *Remove Stopword*

Pada proses *remove stopword* ini berguna sebagai menghapus kata *stopword* atau biasa disebut juga kata penghubung dari suatu kalimat seperti yang, dan, tetapi dan sebagainya. Dalam proses penghapusan *stopword* ini, terlebih dahulu dilakukan mendefinisikan kata-kata yang

nantinya akan terhapus ketika proses ini dijalankan. Dalam hal ini, seluruh kata-kata yang sudah di definisikan tadi disimpan di dalam sebuah file yang dengan nama *stopword_id.* File ini disimpan pada folder *corpora* yang terdapat di dalam *nltk_data*. Dalam proses penghapusan *stopword* ini dibantu dengan *library nltk* yang terdapat pada bahasa pemrograman *python3.* Dalam hal ini, peneliti melakukan proses install *library nltk* menggunakan *pip* sebagai perintah pada Gambar 4.9 di bawah.



Gambar 4.9 proses install *library nltk*

Setelah proses instalasi selesai, maka peneliti mendeklarasikan *library nltk* terlebih dahulu seperti pada Gambar 4.10 di bawah.

```
import nltk
from nltk import word_tokenize, sent_tokenize
from nltk.corpus import stopwords
```

Gambar 4.10 Pendeklarasian *library nltk*

Selanjutnya proses pengimplementasian dari tahapan *remove stopword* pada kode program Gambar 4.11 di bawah.

```
def removeStopword(str):
    stop_words = set(stopwords.words('stopwords_id'))
    word_tokens = word_tokenize(str)
    filtered_sentence = [w for w in word_tokens if not w in stop_words]
    return ' '.join(filtered_sentence)
```

Gambar 4.11 Kode Program Proses *Remove Stopword*

c. *Tokenization*

Pada proses *tokenization* berguna sebagai pemisah kata, simbol, frase dan entias dari suatu teks. Dalam proses ini dilakukan juga menggunakan bantuan *library nltk* pada bahasa pemrograman

python3. Adapun pengimplementasian dari kode program *tokenization* dapat dilihat pada Gambar 4.12 di bawah.

```
def word_tokenization(str):
    str = word_tokenize(str)
    return str
```

Gambar 4.12 Kode Program Proses *Tokenization*

d. *Stemming*

Pada proses *stemming* berguna sebagai penghapusan kata imbuhan dari setiap kata, baik kata imbuhan yang berada di depan kata ataupun di belakang kata. Dalam proses *stemming* dikerjakan menggunakan bantuan dari *library sastrawi* yang terdapat dalam bahasa pemrograman *python3*. Pada proses ini peneliti melakukan instalasi *library sastrawi* terlebih dahulu dengan menggunakan perintah *pip* pada Gambar 4.13 di bawah.

```
(base) D:\SKRIPSI1>pip install sastrawi
```

Gambar 4.13 Proses Instalasi *library sastrawi*

Adapun proses instalasi selesai, maka peneliti perlu mendeklarasikan *library sastrawi* terlebih dahulu pada Gambar 4.14.

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
```

Gambar 4.14 Pendeklarasian *library sastrawi*

Selanjutnya proses pengimplementasian dari tahapan *stemming* pada kode program Gambar 4.1 di bawah.

```
def stemming(str):
    factory = StemmerFactory()
    stemmer = factory.create_stemmer()
    return stemmer.stem(str)
```

Gambar 4.15 Kode Program Proses *Stemming*

Setelah semua proses *preprocessing* dijalankan terhadap semua data, maka hasil dari *preprocessing* disimpan menjadi suatu *file* baru yang nantinya akan dijadikan sebagai *dataset* dalam proses pengklasifikasian. Adapun hasil dari proses *preprocessing* pada Gambar 4.16 di bawah.

Tweet	cleantext	label
b'RT @lahan_poker: Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5 #AndreTaulanyKufurNikmat #Andr\xe2\x80\xa6'	andre taulany minta maaf pasca hina nabi proses hukum tetap jalan andretaulanykufurnikmat andr	0
b'Jaman sekarang, apa aja di politikin #SaveAndreTaulany #AndreTaulanyHinaRasulullah'	jaman aja politikin saveandretaulany andretaulanyhinarasulullah	1
b'RT @RiswandiDito: Dari sini kita bisa ambil pelajaran, Bercanda boleh tpi jangan berlebihan dan jangan bawa-bawa itu kedalam keagamaan, sem\xe2\x80\xa6'	ambil ajar canda tpi lebih bawa bawa dalam agama	1
b'Andre Taulany Minta Maaf Pasca Hina Nabi, Proses Hukum Tetap Jalan! https://t.co/eq7xrexua5\xe2\x80\xa6 https://t.co/U3DCyCb0zY'	andre taulany minta maaf pasca hina nabi proses hukum tetap jalan	0
b'#andretaulany\n#AndreTaulanyMakinSongong\n#AndreTaulanyKufurNikmat\n#AndreTaulanyHinaRasulullah\nCoba cek ini:\n\nAndre T\xe2\x80\xa6 https://t.co/2xcRkXGywW'	andretaulany andretaulanymakinsongong andretaulanykufurnikmat andretaulanyhinarasulullah ncoba cek nandre	1
b'@PartaiSocmed Akun abal anal\n#AndreTaulanyHinaRasulullah'	akun abal anal andretaulanyhinarasulullah	1
b'@hudlaha @PartaiSocmed Tangkap si pelawak cebong \n#AndreTaulanyKufurNikmat\n#AndreTaulanyHinaRasulullah'	tangkap lawak cebong andretaulanykufurnikmat andretaulanyhinarasulullah	1

Gambar 4.16 Hasil *Preprocessing*

4.3 Ekstraksi Fitur

Pada proses ekstraksi fitur, proses pertama yang dilakukan oleh sistem setelah *tokenization* yaitu mengubah dataset menjadi suatu representasi *vector* dengan menggunakan *library* yang sudah disediakan oleh *Phyton* yang bernama *library Count Vectorizer*. Sebagai contoh penelitian menggunakan 3 komentar, diantarnya :

(Doc1)”Cowok itu bajunya bagus sekali”

(Doc2)”Mulutnya hancur banget seperti mulut anjing”

(Doc3)”Cowok itu sangat hancur”

Setelah sistem melakukan preprocessing terdapat 4 jumlah kata baku dari 3 kalimat di atas yaitu “Cowok”,”Bagus”,“Mulut”, dan ”Hancur”.

Setelah tahapan di atas dari setiap dokumen ditampilkan mejadi sebuah *vector* dengan elemen, ketika kata tersebut terdapat di dalam dokumen maka diberikan nilai 1, jika tidak ada maka diberikan nilai 0. Sebagai contoh terdapat pada Tabel 4.1 di bawah.

Tabel 4.1 Pembuatan *Word Vector*

	Cowok	**Bagus**	**Mulut**	**Hancur**
Doc1	1	1	0	0

Doc2	0	0	2	1
Doc3	1	0	0	1

Dokumen yang telah diubah menjadi *word vector* selanjutnya akan dihitung menggunakan rumus *TF-IDF*, dengan menggunakan rumus ini maka akan menghasilkan *word vector* yang memiliki nilai yang sudah terbobot. *TF* atau *Term Frequency* itu sendiri adalah banyaknya frekuensi kemunculan kata dari suatu *term* dalam dokumen bersangkutan, sedangkan *IDF* atau *Inverse Document Frequency* adalah perhitungan dari bagaimana *term* disebarkan atau didistribusikan secara luas dalam koleksi dokumen yang bersangkutan.

Proses perhitungan bobot kata dilakukan dengan proses awal menghitung *TF* atau *Term Frequency* terlebih dahulu. Dapat dilihat contoh pada Tabel 4.2 di bawah.

Tabel 4.2 Proses Perhitungan *TF* (*Term Frequency)*

	(Doc1)	***(Doc2)***	***(Doc3)***
Cowok	1	0	1
Bagus	1	0	0
Mulut	0	2	0
Hancur	0	1	1

Setelah proses perhitungan bobot *TF* selesai selanjutnya dilakukan proses menentukan *DF* atau *Document Frequency* yaitu dengan banyaknya *term* (*t*) muncul dalam semua dokumen. Maka akan memperoleh hasil seperti Tabel 4.3 di bawah.

Tabel 4.3 Proses Perhitungan *DF (Document Frequency)*

T (Term)	***DF (Document Frequency)***
Cowok	2
Bagus	1
Mulut	2
Hancur	2

Kemudian setelah proses *TF* dan *DF* kemudian dilanjutkan menghitung nilai *IDF* (*Inverse Document Frequency*) dengan cara menghitung nilai dari log hasil D atau jumlah dokumen dalam contoh kasus ini ada 3 dokumen, dari 3 dokumen tersebut dibagi dengan nilai *DF*

(*Document Frequency*). Maka akan menghasilkan nilai perhitungan seperti Tabel 4.4 di bawah.

Tabel 4.4 Proses *IDF (Inverse Document Frequency)*

T (Term)	***DF (Document Frequency)***	***D/DF***	***IDF (Inverse Document Frequency)***
Cowok	2	1,5	log 1,5 = 0,176
Bagus	1	3	log 3 = 0,477
Mulut	2	1,5	log 1,5 = 0,176
Hancur	2	1,5	log 1,5 = 0,176

Setelah mendapatkan nilai *IDF* (*Inverse Document Frequency*), selanjutnya dilanjutkan dengan menghitung *TF-IDF*. Seperti pada Tabel 4.5 di bawah.

Tabel 4.5 Contoh Proses Perhitungan *TF-IDF*

Q	***TF***			***DF***	***D/DF***	***IDF***	***IDF+1***	***W = TF*(IDF+1)***		
	Doc 1	***Doc 2***	***Doc 3***					***Doc 1***	***Doc 2***	***Doc 3***
Cowok	1	0	1	2	1,5	0,176	1,176	1,176	0	1,176
Bagus	1	0	0	1	3	0,477	1,477	1,477	0	0
Mulut	0	2	0	2	1,5	0,176	1,176	0	2,352	0
Hancur	0	1	1	2	1,5	0,176	1,176	0	1,176	1,176
Nilai Bobot Dari Setiap Dokumen								2,653	3,528	2,352

Hasil dari *word vector* yang sudah mendapatkan bobot dapat dilihat pada Tabel 4.6 di bawah.

Tabel 4.6 Contoh *Word Vector* yang sudah dibobotkan

	Pantai	**Bagus**	**Taman**	**Indah**
(Doc1)	1,176	1,477	0	0
(Doc2)	0	0	2,352	1,176
(Doc3)	1,176	0	0	1,176

4.4 Implementasi Klasifikasi *Naïve Bayes*

Pada proses ekstraksi fitur dan proses pengklasifikasian *Naïve Bayes* yang nantinya akan di compres menjadi satu *class pipeline vectorizer => transfomer => classifier*. Proses pengklasifikasian tersebut berjalan dengan bantuan *library* pada bahasa pemrograman Python3 yang mempunyai nama *library scikit-learn* untuk proses pengklasifikasian, selain itu terdapat *library numpy* dan juga *pandas* sebagai pembacaan data.

Untuk *library scikit-learn* disini yang digunakan adalah *Pipeline, CountVectorizer, Naïve Bayes, MultinomialNB, Confusion Matrix, TfidfTransformer*, dan *f1 Score*.

Untuk langkah awal pengerjaan proses ekstraksi fitur dan klasifikasi adalah dilakukan proses menginstall library yang diperlukan. Selanjutnya setelah semua library terinstall maka dilanjutkan ke proses mendeklarasi semua library yang akan digunakan. Adapun kode program untuk deklarasi pada Gambar 4.17 di bawah.

```
import pandas as pd
import numpy as np
from sklearn.naive_bayes import MultinomialNB
from sklearn.svm import LinearSVC, SVC
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.pipeline import Pipeline
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.metrics import classification_report,confusion_matrix, accuracy_score, f1_score, precision_score, recall_score
```

Gambar 4.17 Proses pendeklarasian *library* yang digunakan

Setelah selesai mendeklarasi *library* dilanjutkan dengan proses mengambil *dataset* yang akan dipergunakan sebagai data *training* menggunakan *library pandas.* Untuk kode program tersebut pada Gambar 4.18 di bawah.

```
data = pd.read_excel('AndreTaulanyHinaRasulullah_excel_label-clean.xlsx',encoding='Latin-1')
len(data)
```

Gambar 4.18 Proses memanggil *data set*

Selanjutnya proses pembuatan *class pipeline* yang di dalamnya terdapat 3 tahapan yaitu mengubah *dataset* yang hasil *crawling data* Twitter menjadi *representasi vector* (mengubah huruf menjadi angka) menggunakan *library CountVectorizer* dengan pembobotan menggunakan *word vector* dalam *library TfidfTransformer*, tahapan terakhir dilakukan

klasifikasi dengan menggunakan *library MultinominalNaiveBayes.* Proses dari pengimplementasian dari tiga proses pembuatan *class pipeline* pada Gambar 4.19 di bawah.

```
#Multinomial Naive Bayes
pipeline_mnb = Pipeline([
    ('vect', CountVectorizer()),
    ('tfidf', TfidfTransformer(use_idf=True, smooth_idf=True)),
    ('clf', MultinomialNB(alpha=1))
])

txt = data['cleantext'].values.astype('U')
#X_train, X_test, y_train, y_test = train_test_split(data['cleantext'], data['label'],  test_size=0.33, random_state = 0)
X_train, X_test, y_train, y_test = train_test_split(txt, data['label'],  test_size=0.33, random_state = 0)
pipeline_mnb.fit(X_train, y_train)
```

Gambar 4.19 Proses Pengimplementasian *Class Pipiline*

Pada proses pengklasifikasian data ini, peneliti menggunakan data tes yang diacak dari 33% atau 0,33 dari data *training.* Proses pengklasifikasian data ini dilakukan dengan menggunakan perhitungan *probabilitas* dari setiap kelas, maka peneliti baru bisa mendapatkan hasil jelas dari prediksi data yang di-*input.* Tahapan akhir setelah melakukan semua proses pengklasifikasian, maka barulah bisa menghitung dari performa dari *algoritme* yang dipergunakan.

4.5 Uji Model

Untuk mengetahui tingkatan dari performa *Algoritme Naïve Bayes*, maka peneliti melakukan pengujian terhadap model. Hasil dari klasifikasi nantinya akan ditampilkan dalam bentuk *confusion matrix.* Tabel yang ditampilkan di dalam *confusion matrix* ini terdiri dari kelas *predicted* dan juga kelas *actual.* Model dari *confusion matrix* ini dapat dilihat pada Tabel 4.7.

Tabel 4.7 Model *Confusion Matrix*

		Predict Class	
		Class A	***Class B***
Actual class	***Class A***	AA	AB
	Class B	BA	BB

Untuk mengetahui nilai dari akurasi model diperoleh dari banyak jumlah data yang tepat hasil klarifikasi dibagi dengan total dari data, seperti pada Gambar 4.20 di bawah.

$$\text{Akurasi} = \frac{AA+BB}{AA+AB+BA+BB}$$

Gambar 4.20 Hasil Akurasi

Pada saat proses pengujian model maka akan mendapatkan hasil dari nilai akurasi dan *confusion matrix* 2x2 pada Gambar 4.21 di bawah.

```
Total documents classified:  2500
Accuracy: 0.7103
Confusion matrix:
[[358  96]
 [143 228]]
             precision    recall  f1-score   support

          0       0.71      0.79      0.75       454
          1       0.70      0.61      0.66       371

avg / total       0.71      0.71      0.71       825
```

Gambar 4.21 Nilai Akurasi dan *Confusion Matrix* 2x2

Nilai akurasi yang didapatkan dari pengujian model sebesar 71.0% yang proses perhitungannya berdasarkan jumlah nilai dari diagonal *confusion matrix* dibagi dengan seluruh jumlah data. Karena jumlah pada data setiap kelas data *training* tidak seimbang, maka besarnya nilai akurasi bukanlah terpenting.

4.6 Evaluasi Model

Dalam proses evaluasi model ini dilakukan setelah uji model telah selesai dilakukan. Evaluasi model berguna sebagai menghitung performa dari metode yang dipilih. Pada proses uji model ini akan menghasilkan *confusion matrix* dengan ukuran 2x2 yang dapat dilihat pada Tabel 4.8 di bawah.

Tabel 4.8 Hasil *Confusion Matrix*

		Predict Class	
		Positive	***Negative***
Actual class	***Positive***	358	96
	Negative	143	228

Seperti pada Tabel 4.8 di atas, *confused matrix* matriks yang berukuran 2x2 setiap kolomnya mewakili nilai dari setiap kelas yaitu kelas *positive,* dan kelas negative.

Berdasarkan rumus yang terdapat dalam bab sebelumnya nilai presisi pada keseluruhan sistem bernilai sebesar **0.704** dan untuk nilai dari *recall* keseluruhan sistem berupa **0.615** sedangkan untuk nilai dari *f-1 Score* untuk pengevaluasian dalam informasi temu kembali yang dihitung mengombinasi nilai dari *presisi* dan *recall* yaitu sebesar **0.656**. Untuk menghitung proses menghitung dari nilai *presisi, recall* dan *f-1 score* pada sistem ini dapat pada Gambar 4.22 di bawah.

```
txt = data['cleantext'].values.astype('U')
#X_train, X_test, y_train, y_test = train_test_split(data['cleantext'], data['label'],  test_size=0.33, random_state = 0)
X_train, X_test, y_train, y_test = train_test_split(txt, data['label'],  test_size=0.33, random_state = 0)
pipeline_mnb.fit(X_train, y_train)
predictions = pipeline_mnb.predict(X_test)

print("Accuracy: {}".format(accuracy_score(y_test, predictions)))
print("F1 Score: {}".format(f1_score(y_test, predictions)))
print("Precision score: {}".format(precision_score(y_test, predictions)))
print("Recall score: {}".format(recall_score(y_test, predictions)))
print("Confusion matrix: {}".format(confusion_matrix(y_test, predictions)))
```

Gambar 4.22 Proses Menghitung dari Nilai *Presisi, Recall* dan *F-1 score*

Dengan diketahuinya nilai dari *precision, recall,* dan *f-1 Score* dalam kinerja di keseluruhan sistem, maka dapat mengetahui kemampuan dari sistem untuk mencari ketepatan atau kebenaran dari informasi yang diminta oleh pengguna dengan hasil jawaban yang dikeluarkan oleh sistem dan memberitahu tingkat keberhasilan dari suatu sistem dalam menentukan kembali suatu informasi atau nilai *accuracy* sebesar **71%.**

Setelah proses di atas selesai, untuk performa dari metode pengklasifikasian dari setiap kelas dapat diketahui dengan *precision, recall,* dan *f-1 Score* di setiap kelasnya. Hasil dari *precision, recall,* dan *f-1 Score* memiliki ukuran penilaian sebesar 0-1. Semakin tinggi nilai maka semakin baik, dalam artian semakin mendekati angka 1 nilai dari 0 maka sistem semakin baik. Hasil dari proses pengevaluasian model keseluruhan sistem ini terdapat pada Gambar 4.23 di bawah.

```
Total documents classified:  2500
Accuracy: 0.7103
F1 Score: 0.6561
Precision score: 0.7037
Recall score: 0.6146
Confusion matrix:
[[358  96]
 [143 228]]
             precision    recall  f1-score   support

          0       0.71      0.79      0.75       454
          1       0.70      0.61      0.66       371

avg / total       0.71      0.71      0.71       825
```

Gambar 4.23 Hasil dari Proses Pengevaluasian Model

Hasil dari nilai *precision, recall,* dan *f-1 Score* di setiap kelas terdapat pada Tabel 4.9 di bawah.

Tabel 4.9 Hasil dari Nilai *Precision, Recall,* dan *F-1 score*

Jenis Klasifikasi	***Precision***	***Recall***	***F-1 Score***
Positif	0,71	0,79	0,75
Negatif	0,70	0,61	0,66

Dapat dilihat dari hasil evaluasi model dapat dilihat nilai *precision,* dan *recall* dari setiap kelas dapat dilihat tingkat kemampuan pemrosesan sistem dalam mencari tingkat ketepatan antara informasi yang diinginkan oleh pengguna sebagai kelas *positif* adalah "**71%**", dan untuk kelas *negatif* adalah "**70%**". Tingkat keberhasilan dari pemrosesan sistem dalam memperoleh kembali informasi kelas *positif* adalah "**79%**", untuk kelas *negatif* adalah "**61%**". Dengan nilai-nilai tersebut dapat dikatakan kinerja sistem dari keberhasilan sistem untuk menemukan kembali suatu informasi yang bernilai *positif* dan *negatif* dalam dokumen sangat rendah.

Untuk itu dilakukan proses pengujian ulang untuk menentukan hasil uji dan evaluasi yang maksimal dengan menggunakan *k-fold cross validation. K-fold cross validation* ini adalah metode *Cross Validation* yang digunakan melipat data sebanyak K dan men-iterasi(pengulangan) sebanyak K. Dalam penelitian ini pengujian menggunakan nilai K yaitu 5. Dalam 5 *fold,* data dibagi menjadi 5 *fold* berukuran kira-kira sama, sehingga sistem memiliki 5 *subset* data sebagai pengevaluasian kinerja algoritme atau model. Hasil pengujian sistem menggunakan metode 5 *fold cross validation* pada Gambar 4.24 di bawah.

```
Total documents classified:  2500
Accuracy: 0.7100
F1 Score: 0.7496
Precision score: 0.7668
Recall score: 0.7331
Confusion matrix:
[[138  66]
 [ 79 217]]
             precision    recall  f1-score   support

          0       0.64      0.68      0.66       204
          1       0.77      0.73      0.75       296

avg / total       0.71      0.71      0.71       500
```

Gambar 4.24 Hasil Pengujian 5 *K-Fold Cross Validation*

Dengan menggunakan *cross validation* dapat dilihat nilai dari *accuracy* tidak berubah dari sebelumnya yaitu sebesar **0,710** atau **71%**. Untuk dari hasil *precision, recall,* dan *f-1 score* mengalami perubahan hasil setiap *class* dapat dilihat pada Tabel 4.10 di bawah.

Tabel 4.10 Hasil *Precision, Recall,* dan *F-1 score*

Jenis Klasifikasi	***Precision***	***Recall***	***f-1 Score***
Positif	0,64	0,68	0,66
Negatif	0,77	0,73	0,75

Dari hasil evaluasi model menggunakan *fold validation* dilihat dari Tabel 4.10 di atas nilai dari *precision* dan *recall* di setiap *class* nya mengalami peningkatan dari kemampuan sistem untuk mencari ketepatan antara informasi yang pengguna minta untuk *precision* kelas *positif* **64%**, dan kelas negatif sebesar **77%**. Sedangkan dari tingkat keberhasilan sistem dalam menemukan suatu informasi kembali untuk hasil *recall* kelas *positif* **68%**, dan untuk kelas negatif **73%**. Dengan hasil dalam Tabel 4.10 di atas maka kinerja sistem tingkat keberhasilan sistem untuk menemukan kembali suatu informasi yang bernilai *positif* dan *negatif* dalam dokumen sangat rendah tetapi mengalami peningkatan dari evaluasi sebelumnya dapat dilihat dari nilai *precision* keseluruhan menjadi **76,7%**, nilai *recall* keseluruhan menjadi **73,3%**, dan juga nilai dari *f1-score* keseluruhan menjadi **74,9%**. Hasil dari nilai *K-fold validation* dapat dilihat pada Gambar 4.25 di bawah.

```
Accuracy: 0.7100
F1 Score: 0.7496
Precision score: 0.7668
Recall score: 0.7331
Confusion matrix:
[[138  66]
 [ 79 217]]
             precision    recall  f1-score   support

          0       0.64      0.68      0.66       204
          1       0.77      0.73      0.75       296

avg / total       0.71      0.71      0.71       500
```

Gambar 4.25 Hasil *Fold Validation*

Adapun peneliti membuat perbandingan antara hasil pengujian dari tingkat *accuracy, precision, recall,* dan *f1-score* terhadap beberapa model yang berbeda seperti pada Tabel 4.11 di bawah.

Tabel 4.11 Perbandingan Metode penelitian

Metode	**Fitur Extraction**	***Accuracy***	***Precession***	***Recall***	***F1-Score***
Naïve Bayes	*TF-IDF, CountVectorizer*	0,710	0,704	0,615	0,657
	CountVectorizer	0,697	0,658	0,679	0,668
	Bi gram	0,676	0,720	0,458	0,560
	TF-IDF, Bi gram	0,680	0,742	0,442	0,554
	Tri gram	0,642	0,740	0,315	0,442
	TF-IDF, Tri gram	0,651	0,798	0,299	0,435
	TF-IDF, CountVectorizer, K fold 5	**0,710**	**0,766**	**0,733**	**0,749**
SVM	*TF-IDF, CountVectorizer*	0,710	0,714	0,592	0,648
	CountVectorizer	0,710	0,736	0,555	0,632
	Bi gram	0,670	0,759	0,390	0,516
	TF-IDF, Bi gram	0,678	0,733	0,444	0,553

	Tri gram	0,649	0,797	0,296	0,432
	TF-IDF, Tri gram	0,651	0,798	0,299	0,435
	TF-IDF, CountVectorizer, K fold 5	0,562	0,789	0,355	0,489
Logistic Regression	*TF-IDF, CountVectorizer*	0,709	0,739	0,544	0,627
	CountVectorizer	0,716	0,729	0,587	0,650
	Bi gram	0,674	0,768	0,393	0,521
	TF-IDF, Bi gram	0,674	0,765	0,396	0,522
	Tri gram	0,650	0,797	0,297	0,432
	TF-IDF, Tri gram	0,650	0,797	0,297	0,432
	TF-IDF, CountVectorizer, K fold 5	0,548	0,769	0,337	0,469

Pada Tabel 4.11 di atas dapat dilihat hasil metode dan *fitur extraction* paling besar terdapat pada metode *Naïve Bayes* dengan menggunakan *fitur extraction TF-IDF, CountVectorizer, K fold 5* yaitu nilai *Accuracy* sebesar **0,710**, *Precession* sebesar **0,766**, *Recall* sebesar **0,733,** *F1-Score* **0,749**. Hasil metode dan *fitur extraction* paling kecil terdapat pada metode *Logistic Regression* dengan menggunakan *fitur extraction TF-IDF, CountVectorizer, K fold 5* yaitu nilai *Accuracy* sebesar **0,548**, *Precession* sebesar **0,769**, *Recall* sebesar **0,337,** *F1-Score* **0,469.**

BAB V
KESIMPULAN

5.1 Kesimpulan

Berdasarkan dari tahapan proses yang dijelaskan pada bab sebelumnya maka diperoleh hasil pengujian algoritme *Naïve Bayes Classifier* yang telah dilakukan, beberapa hal yang dihasilkan:

a. Pada penelitian Implementasi Text Mining Untuk Mendeteksi Hate Speech pada Twitter menggunakan algoritme *Naïve Bayes Classifier* terbukti algoritme ini yang akurat karena menghasilkan nilai akurasi **0,710** atau **71,0%.**
b. Dalam penelitian ini untuk memastikan dari hasil penelitian, maka dilakukan juga proses pengujian dengan menggunakan *K-Fold Cross Validation* dengan menggunakan nilai dari k sebesar 5 yang menghasilkan nilai dari akurasi sebesar **0,710** atau **71,0%.** Mengalami peningkatan pada nilai *precision* keseluruhan menjadi **76,7%**, nilai *recall* keseluruhan menjadi **73,3%**, dan juga nilai dari *f1-score* keseluruhan menjadi **74,9%.**
c. Selain menggunakan algoritme *Naïve Bayes Classifier* peneliti juga menggunakan algoritme *Logistic Regression Model* sebagai pembanding model algoritme mendapatkan nilai akurasi sebesar **0,709** atau **70,9%.**

5.2 Saran

Dari hasil yang dikerjakan dalam kasus ini masih mempunyai kekurangan dalam metode *Naive Bayes Classifier* untuk menentukan kemungkinan kasus data tersebut *Hate Speech*, diharapkan dalam penelitian berikutnya proses pengerjaannya dapat menggunakan metode atau algoritme klasifikasi yang lain yang berguna sebagai pembanding hasil uji model yang dipergunakan untuk mencari algoritme klasifikasi terbaik.

.

DAFTAR PUSTAKA

Amin , F. (2012). Sistem Temu Kembali Informasi dengan Metode Vector Space Model . *Jurnal Sistem Informasi Bisnis*.

Buntoro, G. A. (2016). Analisis Sentimen Hatespeech Pada Twitter Dengan Metode Naive Bayes Classifier Dan Support Vectore Machine. *Jurnal Dinamika Informatika*, 5.

Cindo, M., Rini, D. P., & Ernitita. (2019). Literatur Review: Metode Klasifikasi Pada Sentimen Analisis. *SAINTEKS 2019*.

Crow Communications. (2011). Twitter For Beginners. Social Media DIY Workshop for Small Business.

Falahah, & Nur, D. D. (2015). Pengembangan Aplikasi Sentiment Analysis Menggunakan Metode Naive Bayes. *SESINDO*.

Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Technique. *University of Illinois at Urbana-Champaign Simon Fraser University* .

Hidayatullah, A. F. (2014). Analisis Sentiment dan Klasifikasi Kategori Terhadap Tokoh Publik Pada Data Twitter Menggunakan Naive Bayes Classifier.

Kaplan, A. M., & Haenlein, M. (2010). Users of the world, unite! The challenges and opportunities of Social Media. *Business Horizons*.

Manalu, E., Sianturi, F. A., & Manalu, M. R. (2017, Desember). Penerapan Algoritma Naive Bayes Untuk Memprediksi Jumlah Produksi Barang Berdasarkan Data Persediaan Dan Jumlah Pemesanan Pada CV. Papadan Mama Pastries. *Manajemen Dan Informatika Pelita Nusantara, 1*, 2.

Pemerintah Indonesia. (2008). *Undang-Undang Informasi dan Transaksi Elektronik Nomor 11 Tahun 2008 Lembaga Negara Republik Indonesia.* Jakarta.

Ridwan, M., Suryono, H., & Sarosa, M. (2013, Juni). Penerapan Data Mining Untuk Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naive Bayes Classifier. *EECCIS, 7*, 1.

Rohman, F. (2016). Analisis Meningkatnya Kejahatan Cyberbullying dan Hatespeech Menggunakan Berbagai Media Sosisal dan Metode Pencegahannya. *Seminar Nasional Ilmu Pengetahuan dan Teknologi Komputer Nusa Mandiri*.

Rozi, I. F., Pramono, S. H., & Dahlan, E. A. (2012). Implementasi Opinion Mining (Analisis. *EECCIS*.

Sarwani, M. Z., & Mahmudy, W. F. (2015). Analisis Twitter Untuk Mengetahui Karakter Seseorang Menggunakan Algoritma Naive Bayes Classifier. *SESINDO*.

Viani, A. N. (2017). Pengaruh Twitter Terhadap Tingkat Partisipasi Politik Remaja dalam Pilkada Serentak 2015 pada Mahasiswa Fakultas Ilmu Komunikasi dan Informatika Universitas Muhammadiyah Surakarta Angkatan 2014.

Zulfa, I., & Winarko, E. (2017, July). Sentimen Analisis Tweet Berbahasa Indonesia dengan. *IJCCS, 11*, 2.

LAMPIRAN