

## Analisis Sentimen Pendapat Masyarakat Terhadap Aksi *People Power* (22 Mei 2019) Melalui Media Sosial Twitter Menggunakan Algoritma *Random Forest*

Aqil Farras\*<sup>1</sup>, Liza Trisnawati

<sup>1,2</sup>Universitas Abdurrah Pekanbaru

e-mail: \*<sup>1</sup>[aqil.farras@univrab.ac.id](mailto:aqil.farras@univrab.ac.id), <sup>2</sup>[lizatrisnawati@univrab.ac.id](mailto:lizatrisnawati@univrab.ac.id)

**Abstract** – Twitter is a social networking and microblogging (short blog) service site. Users of the Twitter social media network utilize it to send and read messages no longer than 140 characters, with the term used for sending messages being tweet. The type of information being given varies from entertainment to education. On the Twitter website there exists a comment system aimed towards the tweet. The comment feature is often used to send aspirations or responses both positive and negative. In this research, the Random Forest algorithm is used to classify Twitter comments that contain positive and negative elements in tweets regarding the People Power Act (22th of May, 2019). The Random Forest model is trained and validated with the Cross-Validation method with the K-Fold technique utilizing 3036 YouTube comments labelled manually as data. The validation result done on the Random Forest model produces an accuracy score of 0.696 or 69%.

**Abstrak** – Twitter merupakan layanan jejaring sosial dan *microblogging* (blog ringkas). Para pengguna memakainya untuk berkirim pesan dan membaca pesan yang tidak lebih dari 140 karakter saja, istilah yang digunakan untuk mengirimkan pesan disebut tweet. Informasi yang diberikan beragam, mulai dari hiburan hingga edukasi. Pada situs Twitter terdapat fitur komentar untuk ditujukan terhadap tweet tersebut. Fitur komentar banyak digunakan untuk melontarkan aspirasi atau tanggapan yang positif maupun negatif. Pada penelitian ini, algoritma *Random Forest* digunakan untuk mengklasifikasi komentar Twitter yang memuat unsur positif dan memuat unsur negatif dalam tweet tentang aksi *People Power* (22 Mei 2019) Model *Random Forest* dilatih dan divalidasi dengan metode *cross-validation* dengan teknik *K-Fold* menggunakan data sebanyak 3036 data komentar YouTube dan diberi label manual. Hasil validasi yang telah dilakukan pada model *Random Forest* mendapatkan nilai akurasi sebesar 0.696 atau 69%.

**Kata Kunci** – Komentar Positif dan Negatif, Twitter, Pengklasifikasi, Random Forest

### I. PENDAHULUAN

Kebebasan berpendapat di Indonesia sudah tertulis dalam UUD 1945 pasal 28E ayat (3) yang menyatakan, “Setiap orang berhak atas kebebasan berserikat, berkumpul, dan mengeluarkan pendapat”. Kemerdekaan mengemukakan pendapat merupakan sebagian dari hak asasi manusia. Oleh sebab itu, dijamin oleh Deklarasi Universal Hak-Hak Asasi Manusia PBB dalam penegasan pasal 19 dan 20. Salah satu kebebasan kita berpendapat terjadi saat Pilpres 2019 dimana terdapat 2 kandidat presiden dan wakil presiden serta semua rakyat Indonesia bebas untuk memilih pasangan yang mereka unggulkan. Terlihat antusias para pendukung kedua calon presiden dan waktiil presiden mulai dari debat 17 Januari 2019 sampai pada Pemilu yang dilaksanakan pada 17 April 2019. Dalam Rapat Pleno Terbuka Rekapitulasi Perhitungan Secara Nasional yang telah resmi di tutup pada Selasa, 21 Mei 2019, disebutkan bahwa pasangan Jokowi-Ma’ruf unggul atas Prabowo-Sandi. Selisih suara sah Jokowi-Ma’ruf dengan Prabowo-Sandi yakni sebesar 16,957,123. Hasil rekapitulasi KPU secara nasional ini terdiri atas perolehan suara di 34 provinsi dan 130 panitia pemilihan luar negeri (PPLN).

Banyak pihak yang tidak puas dengan hasil yang diumumkan oleh KPU maka alasan terjadinya aksi *People Power* pada 22 Mei 2019 untuk berpendapat atau menyuarakan isi pikiran terhadap KPU. Dasar hukum demonstrasi adalah pasal 28 UUD 1945 dan UU No. 9 Tahun 1998 sehingga para peserta demonstrasi memiliki

legalitas dalam aksinya. Seiring perkembangan teknologi dan maraknya media sosial, semakin luas pula kebebasan berpendapat di dalam komunitas. Kita sebagai makhluk modern dengan mudah menuangkan isi pikiran, pendapat, dan argumen kita di media sosial. Dan dikarenakan media sosial bersifat luas dan terbuka, pendapat kita tersebut dapat dilihat oleh masyarakat luas, namun karena itu terkadang apabila kita merasa tidak senang atau kecewa terhadap suatu pihak, secara tidak sadar kita dapat menuangkannya di dalam media sosial baik dalam bentuk komentar positif maupun negatif. Komentar positif merupakan komentar memuji sifat baik dan tidak menyinggung perasaan orang lain, sementara komentar negatif merupakan komentar yang berupa kritikan ataupun kata-kata yang tidak benar dan tidak sopan, seperti salah satu contohnya yaitu Twitter yang mejadi wadah komentar atau gambar yang di post pada akun tersebut.

Analisis sentiment merupakan proses memahami, mengekstrak, dan mengolah data tekstual secara otomatis untuk mendapatkan informasi sentiment yang terkandung dalam suatu kalimat opini. Analisis sentiment dilakukan untuk melihat beberapa opini seseorang yang ditujukan kepada kejadian People Power 22 Mei 2019, yang dapat dikategorikan menjadi positif, netral, atau negatif. Terdapat beberapa metode analisis yang dapat digunakan, contohnya Naïve Bayes, Maximum Entropy, Neural Network, ataupun Support Vector Machine (SVM). Dengan demikian, penulis tertarik untuk menggunakan metode Random Forest untuk melakukan analisis sentiment. Pemilihan metode Random Forest dilakukan karena Random Forest merupakan teknologi pre-processing dalam klasifikasi fitur, yang menambah skalabilitas, akurasi, dan efisiensi yang tentunya sangat dalam proses klasifikasi sebuah teks seperti tree. Sebagai alat klasifikasi, Random Forest dianggap efisien dan sederhana, serta sensitive pada seleksi fitur mudah dan sederhana dalam pengaplikasiannya.

Oleh karena itu, diperlukan Analisis Sentimen Pendapat Masyarakat Terhadap Aksi People Power (22 Mei 2019) Melalui Media Sosial Twitter Menggunakan Algoritma Random Forest dengan tujuan menganalisis pendapat masyarakat terhadap aksi People Power (22 Mei 2019) melalui media sosial Twitter menggunakan algoritma Random Forest.

## II. PENELITIAN YANG TERKAIT

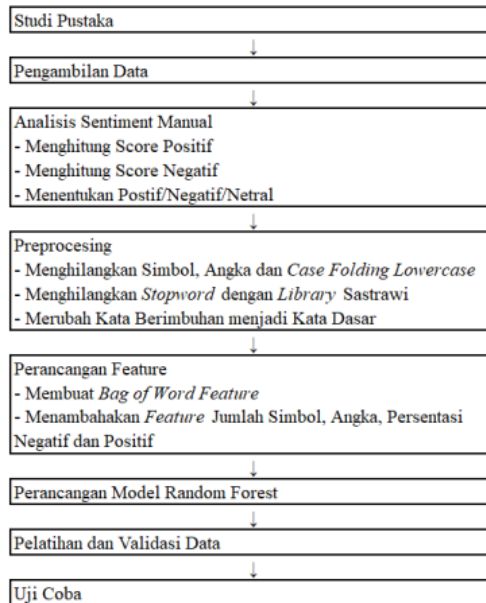
Penelitian ini didasari dari pengembangan yang dilakukan daripada beberapa penelitian terdahulu, salah satunya adalah penelitian berjudul ‘Pendeteksian Sarkasme Pada Proses Analisis Sentimen Menggunakan Random Forest Classifier’ oleh Debby Alita dan Auliya Rahman Isnain, dimana mereka meneliti menggunakan metode *Support Vector Machine* yang dilanjutkan dengan ekstraksi fitur dan kemudian diklasifikasikan dengan menggunakan metode Random Forest Classifier. Hasil penelitian ini berupa peningkatan nilai rata-rata akurasi sebesar 16.61%, nilai presisi sebesar 5.45%, nilai recall sebesar 9.64%, dan nilai f1-score sebesar 11.27% dengan jumlah data sebanyak 2027. Pengembangan yang dilakukan pada penelitian ini adalah penggunaan pendapat masyarakat mengenai aksi People Power (22 Mei 2019) sebagai objek penelitian dikarenakan berupa topik yang lebih viral dan memiliki potensi menghasilkan lebih banyak data yang dapat digunakan dalam proses analisis sentimen. [1]

Selanjutnya adalah penelitian yang berjudul ‘Analisis Sentimen Terhadap Aplikasi Ruangguru Menggunakan Algoritma Naïve Bayes, Random Forest, dan Support Vector Machine’ oleh Evita Fitri yang membahas mengenai penggunaan ketiga algoritma tersebut untuk melakukan analisis sentiment terkait kualitas aplikasi Ruangguru. Hasil yang diperoleh adalah peningkatan akurasi sebesar 7.16% sebagai tahap terakhir model klasifikasi Random Forest dengan performa terbaik. Pengembangan yang dilakukan pada penelitian ini adalah penggunaan K-Fold sebagai salah satu metode validasi model Random Forest serta objek penelitian yang berbeda, yaitu pendapat masyarakat terhadap aksi People Power (22 Mei 2019) dengan melakukan scraping pada data komentar Twitter. [2]

## III. METODE PENELITIAN

### A. Kerangka Kerja Penelitian

Kerangka kerja penelitian yang digunakan dalam penelitian ini adalah sebagai berikut:



Gambar. 1 Metode Penelitian

1. Studi Pustaka  
Sebelum melakukan penelitian, diperlukan pengkajian pengetahuan yang ada dalam kepustakaan dan referensi agar menjadi landasan penelitian.
2. Pengambilan Data  
Pada tahap ini, hal yang dilakukan adalah mengumpulkan data komentar yang berasal dari Twitter dengan menggunakan tools seperti Tweepy.
3. Analisis Sentimen Manual  
Pada tahap ini, dilakukan pengelompokan kata mulai dari positif dan negatif yang diberi score pada masing-masing kata yang telah diambil sebelumnya.
4. Preprocessing  
Pada tahap ini, data-data yang telah diambil dirapikan sebelum digunakan dalam model Random Forest seperti menghilangkan simbol, angka, dan *case folding lowercase*. Selain itu, dilakukan pengambilan huruf a-z dan A-z, case folding langsung menggunakan fungsi *built-in* Python (`.lower()`), menghilangkan stopwords menggunakan library sastrawi `stopword.remove(data)`, serta merubah kata berimbuhan menjadi kata dasar dalam fungsi library terdapat `tokenizing`.
5. Perancangan Fitur  
Pada tahap ini, dirancang fitur berupa Bag of Words untuk merubah kalimat menjadi vektor serta menambahkan fitur jumlah simbol, jumlah angka, persentase kata positif, serta persentase kata negatif.
6. Perancangan Model Random Forest  
Pada tahap ini dilakukan seleksi fitur yang berdampak atas implementasi model Random Forest dan menghapus yang tidak memiliki dampak penting atas implementasi model Random Forest.
7. Training dan Validasi  
Pada tahap ini, melakukan penelitian dengan mempelajari fitur dengan algoritma Random Forest dan melakukan Cross-Validating dengan metode K-Fold Validation.
8. Uji Coba  
Pada tahap ini, dilakukan testing prediksi pada sejumlah data tweet baru yang melihat kata-kata yang negatif dan positif. [3]

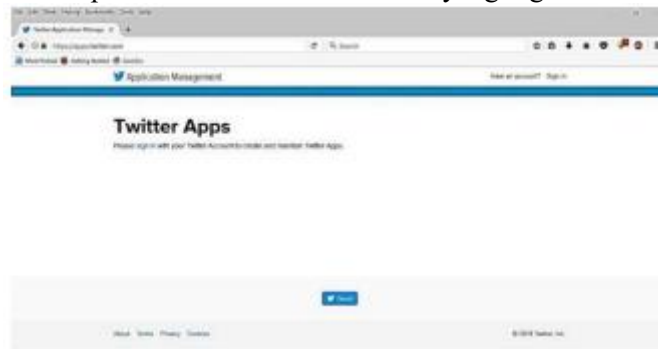
## IV. HASIL DAN PEMBAHASAN

### 3.1 Pengambilan Data

Pengambilan data tweet pada layanan Twitter API dilakukan menggunakan bahasa pemrograman Python. Sebelum dapat melakukan proses pengambilan data, pertama-tama harus membuat dan mendaftarkan aplikasi pada Twitter untuk mendapatkan kunci akses pada Twitter API.

#### a. Membuat Koneksi Twitter API

Langkah pertama yang harus dilakukan adalah membuat aplikasi pada Twitter dengan mengunjungi situs <https://apps.twitter.com/> untuk mendapatkan kunci dan token akses yang digunakan untuk mengakses Twitter API.



Gambar. 2 Situs Twitter Apps

Kemudian, dilanjutkan dengan membuat aplikasi baru dengan memilih menu *Create New App* dan mengisi semua kotak input yang ada pada *Application Details*. Setelah semua form yang terlampir sudah terisi, maka akan tampil halaman detail dari aplikasi yang telah dibuat. [4]



Gambar. 3 Detail Aplikasi yang Dibuat

Aplikasi yang telah didaftarkan tadi memiliki kunci dan token akses yang meliputi *consumer key*, *consumer secret*, *access token*, dan *access token secret* yang nantinya akan digunakan sebagai koneksi dan akses dari Twitter API dalam melakukan pengambilan data tweet pada Twitter. Kunci dan token ini bersifat rahasia dan hanya milik satu pengguna.



Gambar. 4 Kunci dan Token Aplikasi yang Didaftarkan

Setelahnya kita membutuhkan library Python yang dapat mengimplementasikan pemanggilan API Twitter yaitu salah satunya adalah Library Tweepy dan Library Drive untuk mengakses media penyimpanan yang ada pada Google Drive dengan path '/content/gdrive'. Import keperluan data mining melalui API Twitter yaitu Tweepy sebagai Library untuk menggunakan API Twitter, kemudian OAuthHandler yang akan menangani konfigurasi API seperti kunci dan token.

#### b. Mengambil Data dari Twitter

Jika otentikasi pada Twitter berhasil, maka selanjutnya adalah pengambilan tweet dengan berdasarkan kata kunci yang telah ditentukan. Tweet yang diambil yaitu tweet berbahasa Indonesia, diambil secara acak baik dari user biasa maupun akun official media Twitter. Topik yang dibahas dalam analisis sentimen ini adalah 'pendapat masyarakat terhadap aksi People Power (22 Mei 2019) melalui media sosial Twitter' dan beberapa kata pencarian yang digunakan untuk melakukan pencarian tweet.

```

2 RT @AdvokatBS 2019-07-15 6:53:
  Gak ada beda dg
3 #gwpengikutulan 2019-07-15 6:50
4 Relawan Minta P 2019-07-15 6:47
5 Pendukung Minti 2019-07-15 6:43
6 Kecuali anda dag 2019-07-15 6:40
7 @evylagi Ch... 2019-07-15 6:39
8 RT @profrotigul:
  yang dikejar oleh 2019-07-15 6:38
9 Woooooogh! malik
  "Prabowo harus! 2019-07-15 6:36
10 Adakah hubunga
  https://t.co/spX3I 2019-07-15 6:36
11 @UtuhWibowo I- 2019-07-15 6:32
  Prof jadi kasihan
  yang dikejar oleh
12 Relawan Tuntut 2019-07-15 6:32
13 Agustus, Komna: 2019-07-15 6:28
14 @TeddyGusnaid
  Jangan hanya ke 2019-07-15 6:28
15 RT @CNNindoni 2019-07-15 6:25
16 Dalam Rusuh 21 2019-07-15 6:24
17 RT @AndiAnef_ 2019-07-15 6:18
18 10 anggota polisi 2019-07-15 6:15
  prabowo dituntut

```

Gambar. 5 Hasil Tweet

### 3.2. Analisa Sentimen Manual

Pengambilan data untuk sentimen analisis dimulai dari Mei sampai Juni sebanyak 3036 komentar yang ada di dalam Twitter mengenai aksi People Power (22 Mei 2019) yang dilakukan dengan menggunakan tool Comment Scraper buatan Philip Klostermann dengan memasukkan url dari komentar yang terkait untuk diambil data komentarnya dan diberi output berupa kumpulan data yang berformat file JSON dan CSV. Setelah data pembagian waktu di tampilkan untuk mengetahui kapan data di ambil, dikumpulkan pelabelan dilakukan manual dimana label 0 adalah netral, label 1 adalah positif, dan 2 adalah negatif. [5]

#### a. Pengumpulan Daftar Kata Positif dan Negatif

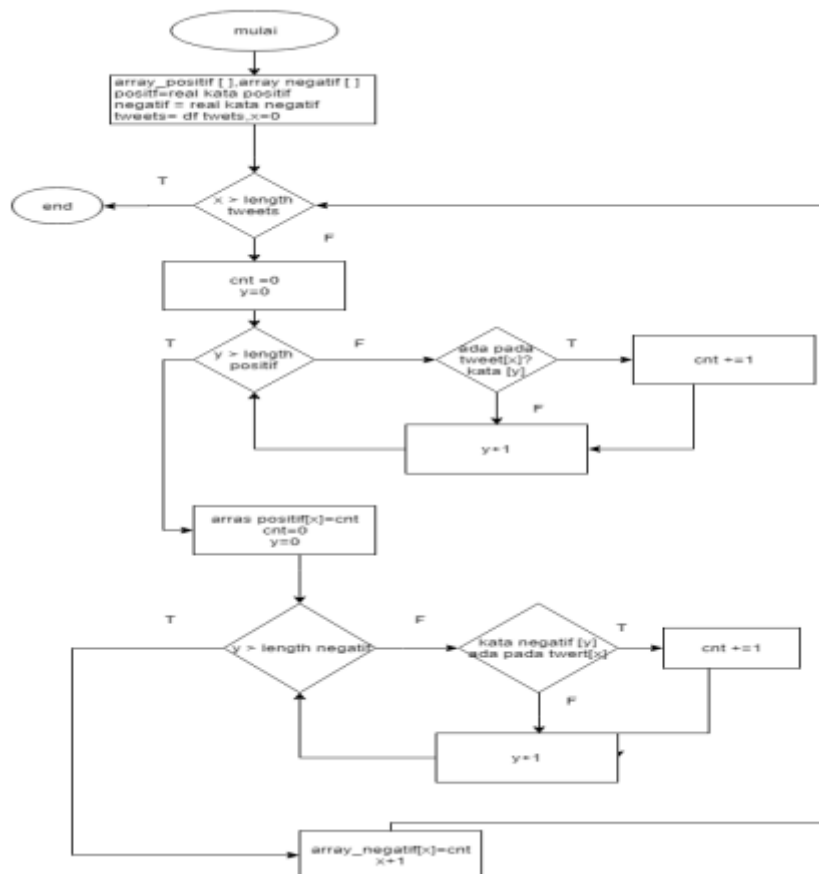
Secara umum pengumpulan data terbagi 2, yaitu pengumpulan data kata-kata sentimen positif dan negatif atau korpus dalam bahasa Indonesia yang sudah dibuat Nazir. Ada kurang lebih 1000 kata positif dan 2000 kata negatif. Beberapa contoh kata yang terkait dengan sentimen positif dan negatif terdapat pada tabel dibawah.

Tabel 1. Contoh Kata Positif dan Negatif

Positif	Negatif
Ahli	Acak
Akurat	Bohong
Amanah	Curang
Aspirasi	Radikal
Ramah	Rasis

b. Menentukan Sentimen dan Jumlah Kata Positif dan Negatif

Data yang telah diambil di Twitter dipisahkan menjadi kalimat positif dan negatif. Berikut adalah flowchart untuk sentimen manual.



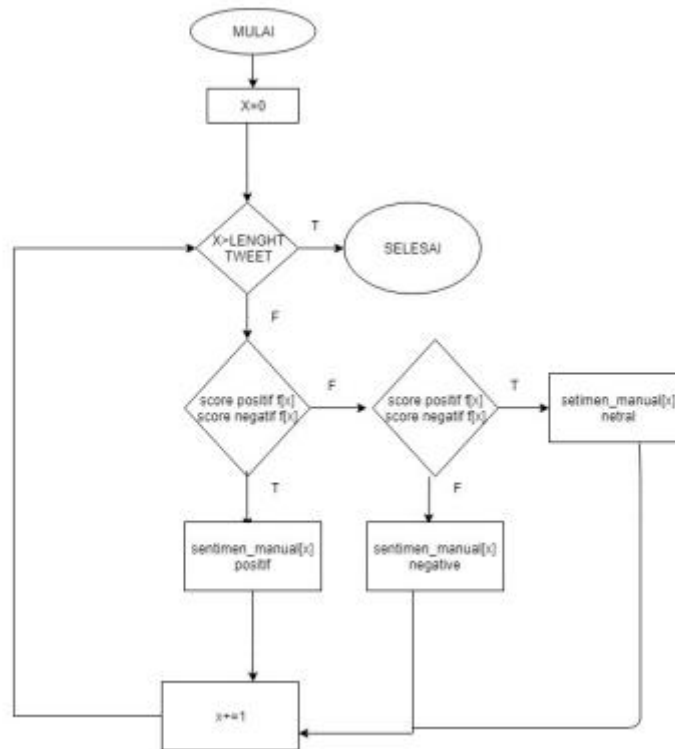
Gambar. 6 Flowchart Sentimen Manual

Berdasarkan flowchart, akan dihitung kemunculan kata-kata positif dan negatif pada setiap tweet yang ada pada variabel df dengan menggunakan perulangan bersarang. Perulangan pertama akan mengulang setiap baris/tweet yang akan dimasukkan pada variabel x. Pada perulangan pertama dideklarasikan variabel cnt bernilai 0. Kemudian dilakukan perulangan kedua yaitu perulangan untuk setiap kata-kata positif pada variabel bertipe string positif yang di split atau dipisah berdasarkan '|'. Method split akan merubah string menjadi tipe data list/array. Variabel y akan berisi setiap elemen list/array hasil split() method pada string positif. Kemudian, dilakukan percabangan apabila kata yang ada pada variabel y terdapat (in) pada x (tweet) maka cnt akan ditambah dengan 1 yang berarti ketika pada tweet mengandung sebuah kata positif maka akan dihitng dengan menambahkan 1 pada variabel penampung cnt. Perulangan y dilakukan 2 kali untuk melakukan perhitungan kemunculan kata negatif setelah kata positif. Kemudian, hasil dari perhitungan akan dimasukkan kedalam list/array penampung arr\_positif, arr\_negatif yang nantinya akan diinput kedalam pandas dataframe atau table yang telah dibuat sebagai kolom baru. [6]

Unnamed: #	tweets	created_at	positive_word	negative_word	sentimen_manual
0	0 Tiga Saksi Melihat Wajah Penembak Kerusuhan 22...	2019-07-15 06:55:55	0	2	negatif
1	1 Jenderal Polisi Minta Komnas HAM Serahkan Data ...	2019-07-15 06:55:29	0	1	negatif
2	2 RT @AdvokatBS: Kecua! anda dapat menggerakkan ...	2019-07-15 06:53:58	2	1	positif
3	3 Gak ada beda dgn bekas-2 Jejak Kerusuhan 21-22...	2019-07-15 06:50:18	0	4	negatif
4	4 Relawan Minta Prabowo Tanggung Jawab Atas Kema...	2019-07-15 06:47:38	1	3	negatif

Gambar. 7 Hasil Output Sentimen Manual

Setelah melakukan analisis sentimen manual, selanjutnya adalah menghitung jumlah kalimat positif, negatif, dan netral pada tweet.



Gambar. 8 Flowchart Score Sentimen Manual

Berdasarkan flowchart, akan dibuat kolom pada variabel df yang berisi hasil percabangan untuk menentukan score. Hasil sentimen manual dibuat dengan python list comprehension dengan percabangan jika jumlah kata positif pada sebuah tweet lebih besar dari jumlah kata negatif maka positif, dan jika jumlah kata positif sama dengan jumlah kata negatif, maka netral. Selain dari itu adalah negatif.

Perhitungan dilakukan dengan menggunakan fungsi len() dengan python list comprehension didalamnya yang mengambil data pada dataframe. Baris pertama akan mengambil data yang memiliki nilai netral pada kolom sentimen\_manual, baris kedua akan mengambil data yang memiliki nilai positif pada kolom sentimen\_manual, dan baris ketiga akan mengambil nilai negatif yang kemudian dihitung jumlah elemen dalam listnya dengan len().

```

↳ tweets netral : 810
   tweets positif : 575
   tweets negatif : 1651
  
```

Gambar. 9 Hasil Output Score Sentimen Manual

### 3.3. Preprocessing

Pada tahap ini, dilakukan proses pembersihan kata yang ada pada tweet yang sudah dipisahkan menjadi negatif dan positif. Preprocessing dilakukan agar data emnjadi data yang ideal dan sesuai untuk diimplementasikan pada model Random Forest. Preprocessing terbagi menjadi beberapa tambahan dengan mengikuti pedoman flowchart sebagai berikut.



Gambar. 10 Flowchart Preprocessing

a. Melakukan Case Folding

Case Folding adalah kalimat yang akan di uji dalam bentuk umum sebuah kalimat diubah menjadi huruf kecil atau lower case dengan tujuan mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar.

Tabel 2. Contoh Case Folding

Teks dari Twitter	Hasil Case Folding
RT@detikcom: Jenderal Polri Minta Komnas HAM Serahkan Data	rt@detikcom: jenderal polri minta komnas ham serahkan data

b. Melakukan Tokenizing

Tokenizing adalah tahap memecah suatu kalimat atau teks menjadi satuan kata yang menyusunnya.

Tabel 3. Contoh Tokenizing

Hasil Case Folding	Hasil Tokenizing
rt@detikcom: jenderal polri minta komnas ham serahkan data	rt, @detikcom:, jenderal, polri, minta, komnas, ham, serahkan, data

c. Filtering

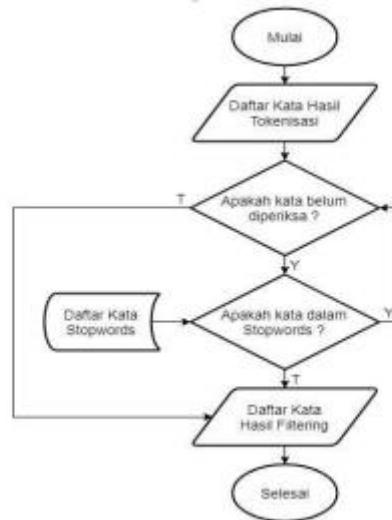
Filtering adalah tahap pengambilan kata-kata penting dari hasil tokenizing. Proses filtering dapat menggunakan stoplist (membuang kata kurang penting) atau wordlist (menyimpan kata penting). Stoplist atau Stopword adalah kata-kata yang tidak deskriptif yang dapat dibuang dalam pendekatan *bag of words*. Bag of Words adalah representasi penyederhanaan yang digunakan dalam pemrosesan bahasa alami dan pengambilan informasi. Juga dikenal sebagai model ruang vector. Dalam model ini, sebuah teks direpresentasikan sebagai tas dari kata-katanya, mengabaikan tata bahasa atau bahkan urutan kata tetapi menjaga keberagaman kata proses ini dibantu dengan menggunakan library sastrawi.

Tabel 4. Contoh Filtering

Hasil Tokenizing	Hasil Filtering
rt, @detikcom:, jenderal, polri, minta, komnas, ham, serahkan, data	jenderal, polri, minta, komnas, ham, serahkan, data

Langkah-langkah proses filtering adalah filterisasi kata yang dilakukan sebanyak jumlah kata pada hasil tokenisasi, melakukan pengecekan pada daftar stopwords bahasa Indonesia, apabila terdapat kata pada daftar tersebut maka kata tersebut akan dihapus pada daftar hasil filterisasi. Jika tidak, maka perulangan akan berlanjut pada kata selanjutnya. Perulangan akan berhenti apabila sudah tidak ada kata lagi yang akan diperiksa. [7]





Gambar. 11 Flowchart Filterisasi Kata

d. Stemming

Stemming merupakan salah satu tahapan dalam proses text processing. Stemming bertujuan untuk mentransformasikan kata menjadi kata dasarnya (root) dengan menghilangkan semua imbuhan kata (affix) meliputi awalan kata (prefix), sisipan kata (infix), akhiran kata (suffix), dan/atau menghilangkan awalan dan akhiran kata (confix) pada kata turunan. Salah satu library yang bisa digunakan dalam melakukan proses stemming bahasa Indonesia adalah dengan menggunakan Library Python Sastrawi. Library ini merupakan pengembangan dari Library Python Sastrawi dimana library tersebut menerapkan algoritma Nazief dan Adriani. Tahapan algoritma tersebut meliputi pemeriksaan root kata, dimana apabila kata merupakan root maka proses berhenti. Selanjutnya adalah menghilangkan imbuhan belakang, imbuhan turunan, dan awalan turunan. Bila belum menemukan kata, maka dianalisis untuk menentukan kata tersebut dimasukkan ke tabel disambiguitas kolom terakhir atau tidak. Bila semua proses diatas gagal, maka algoritma mengambilkan kata aslinya.

Tabel 5. Contoh Stemming

Hasil Tokenizing	Hasil Stemming
rt, @detikcom:, jenderal, polri, minta, komnas, ham, serahkan, data	jenderal, polri, minta, komnas, ham, serah, data

3.4. Perancangan Fitur

Tahap ini meliputi pengelompokkan kata-kata untuk dijadikan fitur dengan metode CountVectorizer (menghitung frekuensi kemunculan suatu kata) dalam Bag of Word. Dimana kata-kata yang dijadikan fitur dipilih berdasarkan frekuensi munculnya kata pada suatu kumpulan data sebanyak dua kali kemunculan. Kata yang dihasilkan dari tahap stemming sebelumnya dicek berdasarkan minimal kemunculan kata dari seluruh kalimat yaitu 0.0025 atau 0.25%.

Setelah kata melalui proses diatas, langkah selanjutnya adalah mencari kata didalam kalimat yang memiliki nilai positif maupun negatif. Tabel hasil proses Bag of Words dapat dilihat seperti dibawah.

Tabel 6. Tabel Bag of Words

Kalimat	Minta	Data	Kematian	...
jenderal polri minta komnas ham serahkan data	1	1	0	0
relawan minta prabowo tanggung jawab atas kematian	1	0	1	0

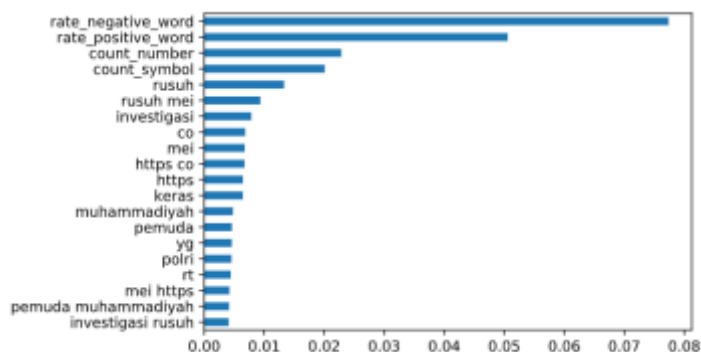
YE ang	YE jadi	YE laku	YE mati	YE sdh	YE tewas	YE tinggal	yn	yuk	yuk gt	zona	zona dinamika	count_symbol	count_number	rate_positive_word	rate_negative_word
0	0	0	0	0	0	0	0	0	0	0	0	10	2	0.000000	0.133333
0	0	0	0	0	0	0	0	0	0	0	0	7	5	0.058824	0.058824
0	0	0	0	0	0	0	0	0	0	0	0	4	0	0.076923	0.076923
0	0	0	0	0	0	0	0	0	0	0	0	9	7	0.000000	0.266667
0	0	0	0	0	0	0	0	0	0	0	0	10	6	0.090909	0.136364

Gambar. 12 Hasil Output Bag of Words

### 3.5. Perancangan Model Random Forest

Setelah seleksi kalimat sebelumnya, langkah berikutnya adalah melakukan *machine learning* algoritma Random Forest. Dari 3036 data yang diambil sebagai contoh, data yang digunakan di tampilan dari 20 data terakhir. Variabel model akan menampung algoritma dari library ensemble scikit learn, yaitu RandomForestClassifier. Parameter *n\_estimators* merupakan jumlah tree pada suatu forest dalam algoritma Random Forest. Kemudian *n\_jobs* merupakan jumlah eksekusi yang dilakukan paralel, bernilai -1 berarti menggunakan semua processor untuk melakukan pekerjaan paralel. *Random\_state* merupakan seed atau inisialisasi nilai random agar tetap sama dimulai dari seed 0. Metode *fit()* dapat dipanggil dari objek model yang akan melakukan training dan pelatihan model dari data input yang diberikan, vektor dari tweet dan label/target dari tweet. Untuk mengoptimalkan algoritma Random Forest, digunakan metode untuk mencari feature yang paling penting atau berpengaruh pada penentuan target/label pada model. *Feature\_importance* akan menampung semua fitur yang terdapat pada model secara terurut dari yang paling tinggi (paling berpengaruh) sampai yang terendah (tidak berpengaruh) dengan menggunakan metode *sort\_values()*. Kemudian, untuk menampilkan grafik *feature\_importance* dapat menggunakan metode *plot.barh()* yang berarti bar horizontal terhadap 20 data terakhir (*tail(20)*).

Deklarasi variabel *selected\_feature* yang menampung semua fitur pada *feature\_importance* yang nilai pengaruhnya lebih dari nilai rata-rata dari semua fitur dengan memanggil *feature\_importance.mean()* untuk mendapatkan nilai rata-rata. Kemudian, mencetak jumlah fitur yang didapat dan jumlah fitur yang dipilih.

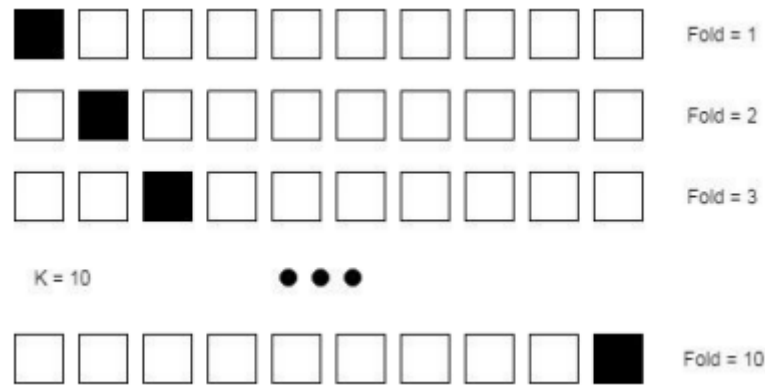


Gambar. 13 Urutan Fitur yang Diseleksi Berdasarkan *Feature Importance*

### 3.6. Pelatihan dan Validasi Data

Model Random Forest berkerja dengan membuat *Decision Tree* yang berjumlah '*n\_estimators*' yaitu pada penelitian ini sebanyak 100 buah. Tahap-tahap pembuatan tree pada model Random Forest pertama dimulai dengan pembuatan data bootstrap (data sampling acak dengan penggantian) pada setiap tree, dan kemudian mengambil fitur pada data secara acak dengan maksimal fitur yang diambil dengan operasi pengakaran dari fitur pada data lebih.

Validasi dilakukan menggunakan metode K-Fold Validation dengan nilai *k* = 10. Metode validasi ini dilakukan dengan cara membagi dataset menjadi 10 bagian yang sama. Dalam 10 iterasi, setiap bagian dari dataset digunakan sebagai data uji, sedangkan bagian lainnya digunakan sebagai data pelatihan.



Gambar. 14 Ilustrasi Metode K-Fold Validation dengan Nilai K = 10

Hasil dari validasi silang dan pelatihan model dapat merepresentasikan seberapa besar nilai akurasi yang didapat pada model Random Forest. Method fit() digunakan untuk melatih model dengan 2 parameter yaitu vektor tweet dan label/target. Variabel `df_vect` berisi vektor yang memiliki fitur yang telah dipilih pada variabel `selected_feature`. Kemudian, validasi dilakukan dengan menggunakan `cross_validate` terhadap model, `df_vect`, dan label dengan menampung jumlah *scoring accuracy*. Perulangan dilakukan untuk mencetak semua nilai hasil dari `validasi_score` yang masing-masing memiliki 5 buah nilai dengan mean atau rata-rata dari setiap nilai tersebut sebagai nilai akurasi dan nilai waktu pada saat training dari validasi. [8]

Hasil dari validasi yang dilakukan pada model Random Forest mendapatkan nilai akurasi sebesar 0.696 atau 69%.

### 3.7. Uji Coba

Dilakukan pengujian terhadap data baru yang berupa komentar yang memuat unsur positif dan negatif, dimana terdapat beberapa ketidakakuratan pada suatu sampel pengujian karena akurasi yang didapat dari validasi model sebesar 0.6966547082285616 atau 69%.

	Tweets	Sentimen
0	polisi melakukan pelanggaran ham	Netral
1	demo berlangsung dengan baik di kawal oleh polisi	Positif
2	banyak korban akibat rusu	Negatif
3	demo berjalan rusuh	Negatif

Gambar. 15 Hasil Uji Coba

## V. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, model Random Forest dapat mengklasifikasi komentar Negatif dan Positif pada Twitter yang memuat tweet mengenai aksi People Power (22 Mei 2019) dengan mempelajari fitur-fitur pada data komentar Twitter yang telah dilakukan preprocessing. Dalam peningkatan keefektifan model Random Forest, dilakukan pembersihan data dengan metode stemming serta menambahkan fitur-fitur yang dapat digunakan untuk meningkatkan efektifitas seperti jumlah simbol, jumlah angka, tingkat kemunculan kata positif, dan tingkat kemunculan kata negatif pada suatu data.

Penyeleksian fitur-fitur yang memiliki dampak penting terhadap model juga dilakukan bertujuan untuk menghindari adanya fitur yang tidak berpengaruh untuk memperkecil jumlah fitur. Pada penelitian ini menggunakan metode Cross Validation untuk memvalidasi dan melatih model secara menyilang dengan teknik K-Fold yaitu membagi validasi menjadi k bagian (pada penelitian ini, digunakan  $k = 10$ ) dengan mendapatkan hasil dari tiap fold lalu dihitung rata-rata dari tiap validasi tersebut untuk melihat performa terbaik dari model Random Forest. Hasil pengujian pada data uji yang telah dilakukan pada model Random Forest mendapatkan nilai akurasi sebesar 0.696 atau 69%.

Beberapa saran yang dapat dilakukan selanjutnya oleh penelitian pengklasifikasian komentar yang mengandung unsur negatif dan positif menggunakan model Random Forest, yaitu menggunakan teknik *word embedding* agar tidak menghilangkan makna komentar, menggunakan data komentar yang tidak memuat bahasa daerah, menambahkan jumlah data agar data tetap dalam performa baik saat setelah terseleksi pada bagian

preprocessing atau pembersihan data, dan menggunakan metode *parameter tuning* untuk menentukan jumlah tree guna mengoptimalkan model Random Forest.

#### UCAPAN TERIMA KASIH

Ucapan terima kasih penulis kepada pihak yang membantu ataupun memberikan dukungan terkait dengan penelitian yang dilakukan seperti bantuan fasilitas penelitian, dana hibah, dan lainnya.

#### DAFTAR PUSTAKA

- [1] D. Alita dan A. R. Isnain, "Pendeteksian Sarkasme Pada Proses Analisis Sentimen Menggunakan Random Forest Classifier", In: *Jurnal Komputasi (JK)* 8.2 (2020).
- [2] E. Fitri, "Analisis Sentimen Terhadap Aplikasi Ruangguru Menggunakan Algoritma Naïve Bayes, Random Forest, dan Support Vector Machine", In: *Jurnal Transformatika (JT)* 18.1 (2020).
- [3] H. Februariyanti dan E. Zuliarso, "Klasifikasi Dokumen Berita Teks Bahasa Indonesia Menggunakan Ontologi", In: *Dinamik* 17.1 (2012).
- [4] A. Imaduddin dan T. Budi, "Pengenalan Karakter Huruf Hangul Korea Menggunakan Random Forest", In: *eProceedings of Engineering* 1.1 (2014).
- [5] H. Leidiana, "Penerapan Algoritma K-Nearest Neighbor Untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor", In: *PIKSEL (Penelitian Ilmu Komputer Sistem Embedded dan Logic)* 1.1 (2013), pp. 065-076.
- [6] T. Mardiana dan R. D. Nyoto, "Kluster Bag of Words Menggunakan Weka", In: *Jurnal Edukasi dan Penelitian Informatika (JEPIN)* 1.1 (2015).
- [7] A. Muzakir dan R. A. Wulandari, "Model Data Mining Sebagai Prediksi Penyakit Hipertensi Kehamilan dengan Teknik Decision Tree", In: *Scientific Journal of Informatics* 3.1 (2016), pp. 019-026.
- [8] D. B. Satmoko, P. Sukarno, dan E. M. Jaded, "Peningkatan Akurasi Pendeteksian Serangan DDoS Menggunakan Multiclassifier Ensemble Learning dan Chi-Square", In: *eProceedings of Engineering* 5.3 (2018).