(Laporan Final Penelitian)

Analisis Sentimen Pengguna Twitter Terhadap Calon Presiden Anies Baswedan Menggunakan Algoritma Naïve Bayes

Rudi Setiawan

Introduction

Di Indonesia, pemilihan presiden menjadi salah satu perayaan yang sangat penting bagi seluruh lapisan masyarakat, tahun 2024 digadang-gadang menjadi tahun kembalinya pesta demokrasi bagi seluruh rakyat Republik Indonesia, karena pada tahun tersebut masa jabatan Presiden dan Wakil Presiden Republik Indonesia periode 2019-2024 akan berakhir. Terdapat banyak survei mengenai bakal calon presiden 2024 yang tersebar di sosial media sejak masa pra deklarasi dimulai. Salah satu tokoh yang paling awal dideklarasikan sebagai bakal calon presiden 2024 adalah Anies Baswedan yang belum lama ini telah merampungkan masa baktinya sebagai Gubernur DKI Jakarta periode 2017-2022 dengan berbagai kebijakan dan program kerja yang cukup efektif dalam menyelesaikan permasalahan-permasalahan di wilayah DKI Jakarta [1].

Munculnya Anies Baswedan sebagai bakal calon presiden yang diusung oleh partai Nasional Demokrat menuai pro dan kontra ditengah masyarakat. Dengan keterbukaan akses teknologi, masyarakat dapat dengan mudah menyampaikan pendapatnya didepan publik melalui media sosial. Salah satu media sosial yang banyak digunakan oleh masyarakat Indonesia adalah Twitter. Twitter merupakan layanan jaringan sosial yang memungkinkan para penggunanya untuk mengirim dan membaca sebuah pesan yang berbasis teks dengan pengguna lainnya menggunakan perangkat komputer maupun *mobile*. Saat ini, Twitter banyak digunakan untuk membahas isu-isu hangat yang baru saja terjadi, mulai dari isu terkait *entertainment*, politik, sosial, hingga pemerintahan [2]. Berdasarkan hasil Survei Penetrasi dan Perilaku Pengguna Internet 2022 milik Asosiasi Penyelenggara Jasa Internet Indonesia atau APJII, jumlah penduduk yang telah terkoneksi internet dalam periode 20212022 berhasil menyentuh 210.026.769 jiwa dari total populasi 272.682.600 jiwa penduduk Indonesia di tahun 2021. Dari jumlah keseluruhan tersebut, sebanyak 98,02% adalah pengguna media sosial, 92,21% pengguna menggunakan internet untuk mengakses informasi/berita, dan 90,21% digunakan untuk bekerja dan bersekolah dari rumah [3].

Dalam dunia sosial media, tidak heran jika banyak pengguna yang memberikan pendapat negatif ataupun positif terhadap suatu isu atau pemberitaan, sama halnya dengan pemberitaan mengenai bakal calon presiden 2024. Terpilihnya Anies Baswedan sebagai bakal calon presiden 2024 yang diusung oleh partai Nasdem berhasil menuai berbagai macam komentar dari para pengguna Twitter di Indonesia, mulai dari dukungan, masukan, sampai dengan kritikan. Opini-opini yang beredar pada sosial media twitter dapat dikategorikan berdasarkan kelas sentimennya masing-masing dengan menggunakan algoritma klasifikasi. Beberapa algoritma klasifikasi yang dapat digunakan dalam proses analisis sentimen, di antaranya *Random Forest, Decision Tree, K-Nearest Neighbors, Naïve Bayes, dan Logistic Regression* [4,5]. Algoritma naïve bayes merupakan metode klasifikasi yang menerapkan teori probabilitas dan statistika, ditemukan oleh Thomas Bayes yang merupakan seorang ilmuan asal Inggris [6]. Naïve Bayes dapat bekerja dengan jumlah data yang sedikit dengan penanganan kelas yang banyak, memiliki struktur yang sederhana sehingga cepat dalam pemrosesannya karena pelatihan dan klasifikasi dapat dilakukan dengan satu kali iterasi data [7].

Pada penelitian sebelumnya mengenai klasifikasi opini tentang calon presiden dan wakil presiden periode 2019-2024 yang dilakukan oleh [8] menunjukkan hasil Joko Widodo dan Ma'aruf Amin memiliki nilai sentimen positif tertinggi, yaitu sebesar 25% dibandingkan dengan polaritas sentimen positif Prabowo Subianto dan Sandiaga Uno yang hanya menghasilkan nilai 5.1%. Pada tahun 2021 penelitian mengenai analisis sentimen menggunakan *naïve bayes classifier* oleh [9] menunjukkan hasil nilai harmonik antara recall dan precision sebesar 0,92 yang memiliki arti

bahwa sistem sudah bekerja dengan baik dalam mendeteksi sentimen-sentimen yang ada. Penelitian tahun 2023 mengenai analisis sentiment terhadap calon Presiden Indonesia 2024 menggunakan algoritma LSTM yang dilakukan oleh [10] terhadap 3 tokoh kandidat bakal calon Presiden yaitu Ganjar Pranowo, Prabowo Subianto dan Ridwan Kamil menunjukkan hasil akurasi 82% pada model Ganjar Pranowo, 82% pada model Prabowo Subianto dan 89% akurasi pada model Ridwan Kamil. Ditahun yang sama penelitian yang berjudul analisis sentimen calon presiden 2024 menggunakan algoritma Support Vector Machine pada media sosial Twitter dengan pengambilan data pada rentang tanggal 17-25 Oktober 2022 didapati hasil sentimen positif yang diperoleh calon presiden Ganjar Pranowo lebih tinggi daripada calon presiden lainnya yaitu 55%, Prabowo 30% dan Anies Baswedan 15%, sementara sentimen negatif Anies Baswedan lebih tinggi 89% daripada Ganjar 8% dan Prabowo 3% [11].

Pada penelitian ini algoritma Naïve Bayes dipilih karena kelebihannya dalam mengklasifikasikan tanpa memerlukan data dalam jumlah besar berdasarkan polaritas positif dan negatif dari komentar yang diperoleh dari Twitter. Input yang digunakan dalam penelitian ini berupa data *tweets* berbahasa Indonesia yang nantinya akan diklasifikasikan menggunakan Naïve Bayes Classifier untuk mendukung proses klasifikasi serta analisis sentimen dari setiap *tweets*. Manfaat penelitian ini adalah untuk mendapatkan gambaran seberapa besar opini positif dan negative masyarakat terhadap pencalonan Anies Baswedan sebagai calon Presiden pada Pemilihan Presiden 2024.

Literature Review

A. Analisis Sentimen

Analisis sentimen bertujuan untuk menganalisa suatu pendapat, sentimen, evaluasi, sikap, penilaian dan emosi yang disampaikan oleh seseorang guna mengetahui apakah pembicara atau penulis tersebut memiliki pendapat atau sepakat dengan suatu topik, produk, layanan, individu, ataupun kegiatan tertentu [12]. Analisis sentimen memiliki beberapa metode klasifikasi seperti *Naïve Bayes Classifier*, *Support Vector Machine*, dan *Maximum Entropy*. Namun, metode klasifikasi yang lebih sering digunakan oleh para peneliti dalam proses analisis sentimen adalah *Naïve Bayes Classifier* karena dinilai lebih mudah untuk digunakan, waktu pemrosesan yang singkat, serta memiliki tingkat akurasi yang cukup tinggi [13].

B. Text Mining

Text mining merupakan suatu proses mengekstraksi atau mencari informasi untuk menggali pengetahuan dari sumber data tekstual [14], Metode ini sering digunakan untuk menganalisis dan memahami teks dalam jumlah besar secara otomatis, memungkinkan identifikasi pola, tren, dan pengetahuan yang sulit atau tidak praktis jika dilakukan secara manual [15]. Beberapa aspek utama dari *text mining* meliputi pengambilan informasi, penguraian teks, pengenalan entitas, analisis sentimen, klasifikasi teks, pengelompokan teks, ekstraksi informasi, pemodelan topik, rekomendasi teks. *Text mining* telah menjadi bagian penting dalam berbagai aplikasi, termasuk analisis media sosial dan pemrosesan bahasa alami. Dengan keberadaan *text mining* memungkinkan organisasi dan individu menemukan informasi berharga dari sekumpulan data yang ada sehingga dapat digunakan untuk proses penunjang keputusan yang lebih baik bagi organisasi.

C. Klasifikasi

Klasifikasi merupakan proses menilai suatu objek data yang memiliki kesamaan karakteristik dalam suatu dokumen dan memasukkan data-data tersebut ke dalam kelas tertentu [16]. Adapun beberapa metode klasifikasi yang sering digunakan diantaranya *Random Forest, Decision Tree, K-Nearest Neighbors, Naïve Bayes, dan Logistic Regression*.

Dalam proses klasifikasi terdapat dua tugas utama yang dilakukan [17], yaitu:

1. Pengembangan model yang digunakan sebagai prototype yang akan disimpan kedalam memori.
2. Penggunaan model yang telah dibuat untuk membantu proses klasifikasi dari suatu objek data guna mengetahui penempatan kelas objek data tersebut.

D. Pra-pemrosesan Data

Pra-pemrosesan merupakan serangkaian proses yang digunakan untuk menyeleksi data dalam bentuk teks agar menjadi sekumpulan data yang lebih terstruktur dengan melalui beberapa tahapan, seperti *case folding, tokenizing,*

filtering, dan *stemming*, pra-pemrosesan penting dilakukan untuk membuat data lebih mudah dibaca dan bebas dari kesalahan data [18]. Adapun tahapan-tahapan dalam *pre-processing* menurut [19] yaitu.

1. *Case folding*, merupakan tahapan di mana akan terdapat proses perubahan teks yang semula huruf besar menjadi huruf kecil, serta menghilangkan seluruh tanda baca yang terdapat pada kalimat tersebut.
2. *Tokenizing*, merupakan tahapan di mana setiap kata-kata yang terdapat dalam suatu kalimat akan dipisahkan berdasarkan spasi yang telah ditentukan.
3. *Filtering*, merupakan tahapan pembuangan kata-kata tidak penting dengan memanfaatkan stopwords berbahasa Indonesia.
4. *Stemming*, merupakan tahapan mengubah suatu kata menjadi ke dalam bentuk kata dasar.
5. *Stopword*, merupakan tahapan pembuangan kata-kata tidak penting dengan memanfaatkan pendekatan *bag-of-words*.

E. Naïve Bayes

Naïve Bayes Classifier atau *Bayesian Classification* merupakan sebuah algoritma dan salah satu teknik klasifikasi yang memadukan penggunaan ilmu statistika dan teori peluang guna menyelesaikan permasalahan suatu kasus *supervised learning* [20]. Rumus perhitungan probabilitas pada metode *Naïve Bayes* ditunjukkan pada Persamaan 1.

$$P(H|X) = \frac{P(X|H)\,P(H)}{P(X)} \qquad (1)$$

Keterangan:
X = Data *class* yang belum diketahi nilainya
H = Hypothesis data X merupakan suatu *class* spesifik
P(H|K) = Probabilitas hipotesis H berdasarkan kondisi X (*posterior probability*)
P(H) = Probabilitas hipotesis H (prior probability)
P(X|H) = Probabilitas X berdasarkan kondisi tertentu
P(X) = Probabilitas dari X

Persamaan 1 juga dapat dituliskan dalam bentuk Persamaan 2.

$$posterior = \frac{prior\ x\ likelihood}{evidence} \qquad (2)$$

Nilai posterior yang telah dihasilkan akan dibandingkan dengan nilai posterior yang dihasilkan dari kelas lain agar nantinya dapat ditentukan kelas dari masing-masing klasifikasi sampel.

F. Confusion Matrix

Confusion matrix merupakan suatu metode yang umumnya banyak digunakan oleh para peneliti dalam proses perhitungan tingkat akurasi dari suatu proses klasifikasi [21]. *Confusion matrix* disajikan dalam bentuk tabel yang di dalamnya memuat berbagai informasi dalam bentuk nilai yang berhasil diprediksi oleh sistem klasifikasi [22]. Contoh penyajian nilai pada *confusion matrix* dapat dilihat pada Tabel 1.

Tabel 1. Confusion Matrix

	Class Prediksi	
	True	*False*
True	*True Positive* (TP)	*False Negative* (FN)
False	*False Positive* (FP)	*True Negative* (TN)

Adapun nilai yang dihasilkan oleh *confusion matrix* sebagai berikut.

1. *Accuracy* merupakan hasil dari perbandingan antara nilai prediksi dan nilai aktual yang didapatkan. Rumus perhitungan *accuracy* ditunjukkan pada Persamaan 3.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

2. *Precision* merupakan metode perhitungan yang digunakan untuk mengetahui seberapa besar perbandingan atau ketepatan yang dihasilkan oleh sistem dalam memberikan prediksi pada kumpulan data dari tiap kelas. Rumus perhitungan nilai *precision* ditunjukkan pada Persamaan 4.

$$Precision = \frac{TP}{FP+TP} \quad (4)$$

3. *Recall* merupakan metode perhitungan dalam confusion matrix yang digunakan untuk mengetahui persentase keberhasilan sistem dalam memberikan prediksi pada keseluruhan data. Rumus perhitungan nilai *recall* ditunjukkan pada Persamaan 5.

$$Recall = \frac{TP}{FN+TP} \quad (5)$$

G. Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency atau biasa dikenal dengan pembobotan kata TF-IDF merupakan salah satu metode perhitungan yang digunakan untuk menentukan sejauh mana keterhubungan suatu kata (*term*) terhadap dokumennya dengan cara memberikan nilai bobot pada setiap kata yang muncul [23]. Perhitungan TF-IDF ditunjukkan pada Persamaan 6, 7, dan 8.

$$tf = 0.5 + 0.5 \times \frac{tf}{max(tf)} \quad (6)$$

$$idf_t = log\left(\frac{D}{dft}\right) \quad (7)$$

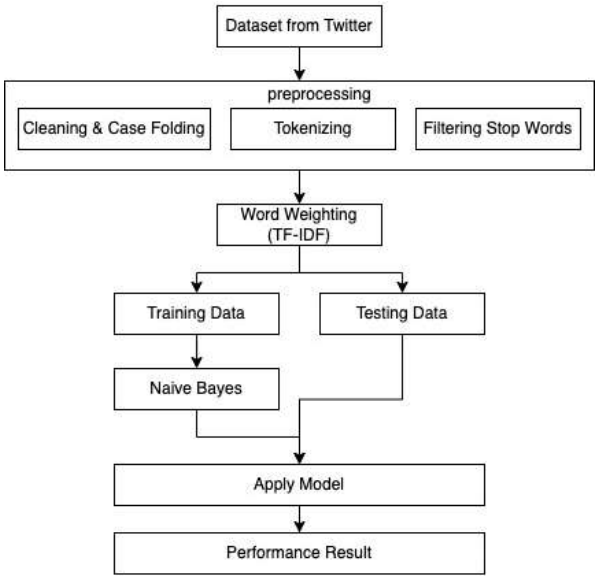
$$W_{d.t} = tf_{d.t} \times idf_{d.t} \quad (8)$$

Information:
D = Document ke – *d*
t = *Term* ke – t dari suatu dokumen
W = Bobot dokumen ke – d terhadap *term* ke – t
Tf = Banyaknya *term* i pada suatu dokumen
idf = *Inversed Document Frequency*
df = Banyaknya dokumen yang mengandung *term* i

Method

A. Tahapan Penelitian

Tahapan penelitian diawali dengan proses pengumpulan data dari *Twitter*. Proses pengumpulan data atau *crawling* data dilakukan dengan *query* bertopik “anies baswedan 2024”, “anies baswedan” dan “anies pilpres”. Tahapan penelitian yang dilakukan pada penelitian ini ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

Setelah data didapatkan, tahapan selanjutnya adalah pra-pemrosesan data. Pada tahap ini terdapat beberapa sub proses, yang meliputi *cleaning* data atau pembersihan data dengan cara menghapus karakter khusus dan tanda baca yang tidak diperlukan, *case folding* mengubah semua huruf yang terdapat dalam suatu kalimat menjadi huruf kecil (*lowercase*), *tokenizing* memisahkan tiap-tiap kata, dan *filtering* untuk membuang kata-kata tidak diperlukan menggunakan daftar *stopwords* dalam bahasa Indonesia. Tahap berikutnya adalah pengelompokkan tiap data *tweets* atau pelabelan data berdasarkan dengan labelnya masing-masing yang dilakukan secara manual. Adapun label kelas atau sentimen yang digunakan pada penelitian ini yaitu kelas berlabel positif dan kelas berlabel negatif.

Tahap penelitian selanjutnya adalah merupakan tahap pembobotan kata atau pemberian nilai pada masing-masing kata yang terkandung dalam suatu dokumen menggunakan rumus perhitungan *term frequency-inverse document frequency* (TF-IDF). Data yang telah memiliki pembobotan kata dibagi menjadi 2 bagian yaitu data latih dan data uji. Penerapan metode *naïve bayes classifier* pada data latih digunakan untuk proses klasifikasi pada proses pengujian model *naïve bayes classifier* dan akan didapatkan nilai akurasi klasifikasi menggunakan *confusion matrix*.

A. Pengumpulan Data

Data yang digunakan pada penelitian ini adalah kumpulan *tweets* yang ditulis atau diposting oleh para pengguna Twitter pada bulan Oktober 2022 hingga Januari 2023 dengan keyword “anies baswedan 2024”, “anies baswedan” dan “anies pilpres”. Total data yang didapatkan sebanyak 7000 *tweets* yang ditulis dalam Bahasa Indonesia. Data *tweets* yang semula terkumpul sejumlah 7000 data berkurang menjadi 1133 dikarenakan proses penghapusan data duplikat atau data yang memiliki dua atau lebih kalimat yang sama. Tabel 2 merupakan contoh hasil pengumpulan data.

Tabel 2. Hasil Pengumpulan Data

Cetak Sejarah, Anies Baswedan Diangkat Jadi Anggota Dewan Universitas Oxford https://t.co/T4Bhs36mdN,"1614021608800399361"
Dalam sejarahnya, ini pertama kali seorang berasal Indonesia diundang untuk menjadi anggota suatu dewan (board) di Universitas Oxford. https://t.co/JQqW8o9FhI,"1614191858850336768"
Anies Baswedan Diangkat jadi Anggota Dewan Universitas Oxford, Eks Gubernur DKI Jakarta Buat Ini di Instagram https://t.co/Y6EaH5XxxZ,"1614036216730750977"
RT @merahputihsuara: Komisi Pemberantasan Korupsi (KPK) akan mengusut dugaan korupsi Bantuan Sosial (Bansos) pandemi Covid-19 tahun 2020 sa…,"1614692038733369344"

RT @Muhammad2553: @alisyarief Untuk mendulang suara di Jawa Timur & Jawa Tengah, hanya pasangan; Anies R Baswedan + AHY yang kuat ...,"1614691614638878721"

@Uki23 https://t.co/puALApTe4A [Inilah 26 Prestasi/Penghargaan Gubernur DKI Jakarta Anies Baswedan - Fakta Kini] https://t.co/HvxcrAySee,"1614691032884740096"

RT @sutanmangara: Saatnya Indonesia Bangkit Bersama ANIES BASWEDAN ! https://t.co/11M8NOE0z9,"1614686795467292674"

B. Pra-pemrosesan Teks

Text pre-processing merupakan serangkaian proses penyeleksian data dalam bentuk teks agar menjadi sekumpulan data yang lebih terstruktur sebelum data masuk ke dalam proses pembobotan kata dengan menggunakan rumus perhitungan TF-IDF dan proses klasifikasi menggunakan *naïve bayes classifier*. Adapun tahap-tahap dari *text pre-processing* yang telah dilakukan adalah sebagai berikut.

1. Pembersihan data dan *Case Folding*

   Pada tahap ini, dilakukan penghapusan seluruh karakter khusus yang ada pada teks dan penghapusan tanda baca yang terdapat pada teks (Tabel 3) dan dilakukan perubahan teks yang semula menggunakan huruf besar menjadi huruf kecil (Tabel 4).

Table 3. Hasil Penghapusan Tanda Baca

No.	Sebelum	Sesudah
1	@AgusYudhoyono @PDemokrat Pak Anies Baswedan pilihan rakyat Indonesia pemimpin yang baik dan cerdas presiden di tahun 2024	Pak Anies Baswedan pilihan rakyat Indonesia pemimpin yang baik dan cerdas presiden di tahun 2024
2	@Relawananies Semoga presiden yang berikutnya pada tahun 2024 mengangkat Anies Baswedan sebagai menteri luar negeri.	Semoga presiden yang berikutnya pada tahun 2024 mengangkat Anies Baswedan sebagai menteri luar negeri

Table 4. Hasil Proses *Case Folding*

No.	Sebelum	Sesudah
1	Pak Anies Baswedan pilihan rakyat Indonesia pemimpin yang baik dan cerdas presiden di tahun 2024	pak anies baswedan pilihan rakyat indonesia pemimpin yang baik dan cerdas presiden di tahun 2024
2	Semoga presiden yang berikutnya pada tahun 2024 mengangkat Anies Baswedan sebagai menteri luar negeri	semoga presiden yang berikutnya pada tahun 2024 mengangkat anies baswedan sebagai menteri luar negeri

2. *Tokenizing*

   Tokenizing merupakan tahapan di mana setiap kata-kata yang terdapat dalam suatu kalimat akan dipisahkan berdasarkan spasi yang telah ditentukan. Tabel 5 merupakan hasil proses dari *tokenizing*.

Table 5. Hasil Proses *Tokenizing*

No.	Sebelum	Sesudah
1.	pak anies baswedan pilihan rakyat indonesia pemimpin yang baik dan cerdas presiden di tahun 2024	|anies| |baswedan| |pilihan| |rakyat| |Indonesia| |pemimpin| |yang| |baik| |cerdas| |presiden| |tahun|
2.	semoga presiden yang berikutnya pada tahun 2024 mengangkat anies baswedan sebagai menteri luar negeri	|semoga| |presiden| |yang| |berikutnya| |pada| |tahun| |mengangkat| |anies| |baswedan| |sebagai| |menteri| |luar| |negeri|

3. *Filtering*

   Pada tahap ini dilakukan penghapusan kata-kata yang tidak diperlukan menggunakan daftar *stopwords* berbahasa Indonesia. Tabel 6 merupakan hasil proses *filtering*.

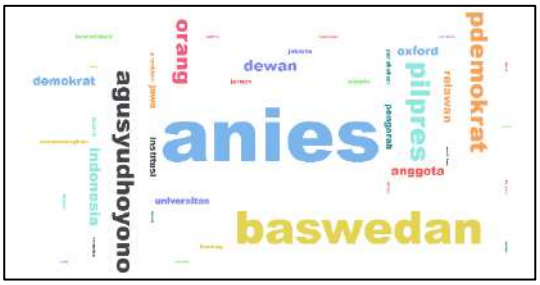
Table 6. Hasil Proses *Filtering*

No.	Sebelum	Sesudah

1.	|anies| |baswedan| |pilihan| |rakyat| |Indonesia| |pemimpin| |yang| |baik| |cerdas| |presiden| |tahun|	|anies| |baswedan| |pilihan| |rakyat| |Indonesia| |pemimpin| |cerdas| |presiden|
2.	|semoga| |presiden| |yang| |berikutnya| |pada| |tahun| |mengangkat| |anies| |baswedan| |sebagai| |menteri| |luar| |negeri|	|semoga| |presiden| |mengangkat| |anies| |baswedan| |menteri| |negeri|

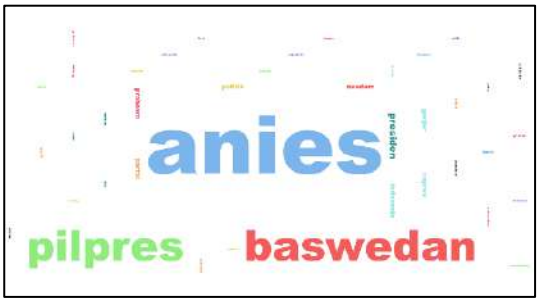
C. Visualisasi Kata

WordCloud merupakan salah satu metode yang terdapat pada perangkat lunak RapidMiner bertujuan untuk memberikan gambaran atau visualisasi dari suatu data dalam bentuk teks secara menarik namun tetap informatif dengan memberikan tampilan daftar kata yang ukuran besar-kecilnya bergantung pada frekuensi kemunculan kata tersebut dalam suatu data *tweets* [24].



Gambar 2. *WordCloud* Sebelum *Pre-processing*

Gambar 2. Merupakan hasil visualisasi wordcloud terhadap 40 kata teratas yang paling sering muncul dalam kumpulan *tweets* yang membahas mengenai anies baswedan sebagai bakal calon presiden 2024 sebelum dilakukannya text pre-processing. Dalam *WordCloud* tersebut menunjukkan bahwa kata yang paling sering muncul adalah "anies" dengan frekuensi kemunculan sebanyak 2159 kali dan "baswedan" dengan frekuensi kemunculan sebanyak 1363 kali. Gambar 3. merupakan hasil visualisasi *wordcloud* terhadap 40 kata teratas yang paling sering muncul setelah dilakukannya text pre-processing. Dalam *WordCloud* terlihat bahwa kata "anies" dan "baswedan" masih menjadi kata yang paling sering muncul dalam data *tweets* dengan frekuensi kemunculan sebanyak 675 dan 364 kali.



Gambar 3. *WordCloud* Sesudah *Pre-processing*

D. Pelabelan Data

Pelabelan data merupakan salah satu rangkaian tahapan penelitian, dimana pada tahap ini data *tweets* yang telah melalui prapemrosesan akan dikelompokkan berdasarkan labelnya masing-masing secara manual. Tahap pelabelan data dilakukan secara manual berdasarkan pendapat pribadi dengan melihat kandungan makna atau kata yang berkonotasi negatif atau positif yang terdapat pada *tweets*. Pemberian label pada data dilakukan untuk proses pembelajaran dan melatih model agar dapat memahami pola atau karakteristik dari setiap data. Contoh hasil pelabelan data ditunjukkan pada Tabel 7.

Table 7. Contoh Pelabelan Data

No	Tweets	Label
1	pemimpin berprestasi kerjanya dihargai diakui nasional internasional anies baswedan presiden	Positive
2	pdip menang capres menghentikan langkahnya anies baswedan presiden megawati soekarnoputri maju puan ganjar berat	Negative

E. Term Frequency Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency (*TF-IDF*) merupakan salah satu rangkaian tahapan penelitian yang bertujuan untuk mengetahui sejauh apa keterhubungan suatu kata terhadap dokumennya dengan cara memberikan nilai bobot pada setiap kata yang muncul pada dokumen tersebut. Tabel 8 merupakan contoh empat dokumen yang berasal dari data *tweets* yang telah didapatkan sebelumnya, pada empat dokumen ini akan diberikan nilai bobot pada setiap kata yang muncul. Proses perhitungan TF-IDF dilakukan di dalam *jupyter notebook* menggunakan bahasa pemrograman Python.

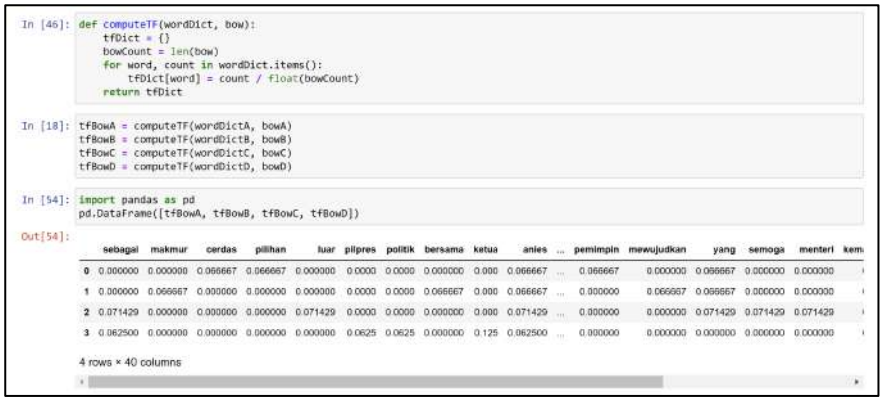
Table 8. Kumpulan Dokumen

Doc A	pak anies baswedan pilihan rakyat indonesia pemimpin yang baik dan cerdas presiden di tahun 2024
Doc B	mewujudkan masyarakat yang adil dalam kemakmuran makmur dalam keadilan bersama anies rasyid baswedan pemilu 2024
Doc C	semoga presiden yang berikutnya pada tahun 2024 mengangkat anies baswedan sebagai menteri luar negeri
Doc D	ketua dmi dan wakil ketua dmi manfaatkan dmi sebagai kendaraan politik anies baswedan di pilpres 2024

Berikut ini tahapan-tahapan dari proses perhitungan pembobotan kata pada empat dokumen yang telah ditentukan menggunakan *Term Frequency-Inverse Document Frequency* (*TF-IDF*).

1. *Term Frequency* (*TF*)

   Proses perhitungan pembobotan kata dilakukan di dalam *jupyter notebook* menggunakan bahasa pemrograman *python*. Adapun tampilan dari *source code* yang digunakan dalam proses perhitungan *term frequency* beserta hasil yang diperoleh ditunjukkan pada Gambar 4.



```
In [46]: def computeTF(wordDict, bow):
             tfDict = {}
             bowCount = len(bow)
             for word, count in wordDict.items():
                 tfDict[word] = count / float(bowCount)
             return tfDict

In [18]: tfBowA = computeTF(wordDictA, bowA)
         tfBowB = computeTF(wordDictB, bowB)
         tfBowC = computeTF(wordDictC, bowC)
         tfBowD = computeTF(wordDictD, bowD)

In [54]: import pandas as pd
         pd.DataFrame([tfBowA, tfBowB, tfBowC, tfBowD])

Out[54]:
```

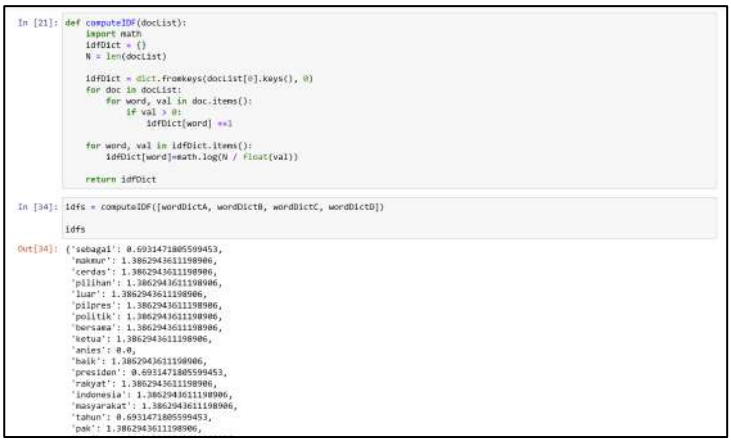
	sebagai	makmur	cerdas	pilihan	luar	pilpres	politik	bersama	ketua	anies	...	pemimpin	mewujudkan	yang	semoga	menteri	kem
0	0.000000	0.000000	0.066667	0.066667	0.000000	0.0000	0.0000	0.000000	0.000	0.066667	...	0.066667	0.000000	0.066667	0.000000	0.000000	
1	0.000000	0.066667	0.000000	0.000000	0.000000	0.0000	0.0000	0.066667	0.000	0.066667	...	0.000000	0.066667	0.066667	0.000000	0.000000	
2	0.071429	0.000000	0.000000	0.000000	0.071429	0.0000	0.0000	0.000000	0.000	0.071429	...	0.000000	0.000000	0.071429	0.071429	0.071429	
3	0.062500	0.000000	0.000000	0.000000	0.000000	0.0625	0.0625	0.000000	0.125	0.062500	...	0.000000	0.000000	0.000000	0.000000	0.000000	

4 rows × 40 columns

Gambar 4. Hasil Perhitungan *Term Frequency*

2. *Inverse Document Frequency* (*IDF*)

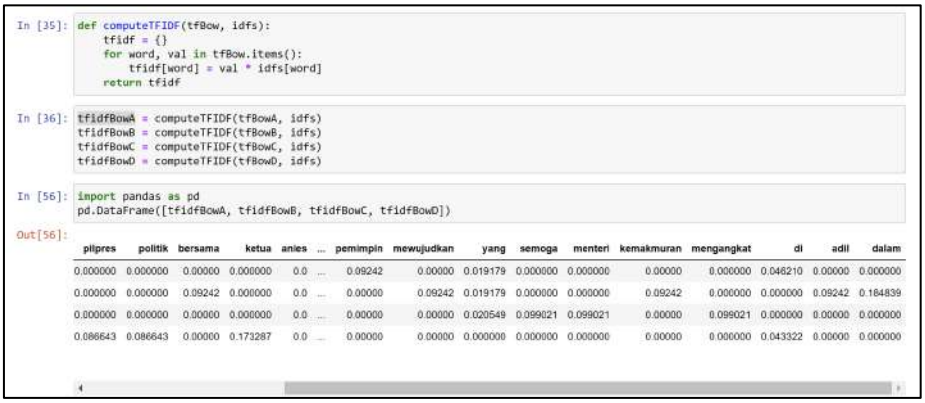
Tahap perhitungan *inverse document frequency* atau *IDF* bertujuan untuk memberikan keterangan kontribusi dari masing-masing kata ke dalam dokumen yang telah ditentukan. Adapun tampilan dari *source code* yang digunakan dalam proses perhitungan *inverse document frequency* beserta dengan hasil yang diperoleh ditunjukkan pada Gambar 5.



Gambar 5. Hasil Perhitungan *Inverse Document Frequency*

3. Pembobotan dari Hasil *TF* dan *IDF*

Tahap pembobotan kata ini merupakan gabungan dari hasil perhitungan *term frequency* dan *inverse document frequency* yang bertujuan untuk mengetahui gambaran nilai dari kumpulan data. Adapun tampilan dari *source code* yang digunakan dalam proses perhitungan pembobotan kata beserta dengan hasil yang diperoleh ditunjukkan pada Gambar 6.



Gambar 6. Hasil pembobotan kata TF-IDF

Hasil perhitungan yang telah didapatkan dari proses perhitungan *term frequency*, *inverse document frequency*, dan bobot kata dengan mengintegrasikan *term frequency* dan *inverse document frequency* disajikan dalam bentuk table yang dapat dilihat pada Tabel 9, Tabel 10, Tabel 11, dan Tabel 12.

Table 9. Calculation of TF-IDF Value in Document "A"

Words	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	B	C	D		A	A	A
anies	1	1	1	1	4	0.067	0.0	0.0
baswedan	1	1	1	1	4	0.067	0.0	0.0
presiden	1	0	1	0	2	0.067	0.693	0.046
ketua	0	0	0	2	1	0.0	1.386	0.0
adil	0	1	0	0	1	0.0	1.386	0.0
cerdas	1	0	0	0	1	0.067	1.386	0.092
indonesia	1	0	0	0	1	0.067	1.386	0.092
keadilan	0	1	0	0	1	0.0	1.386	0.0

Table 10. Calculation of TF-IDF Value in Document "B"

Words	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	B	C	D		B	B	B
anies	1	1	1	1	4	0.067	0.0	0.0
baswedan	1	1	1	1	4	0.067	0.0	0.0
presiden	1	0	1	0	2	0.0	0.693	0.0
ketua	0	0	0	2	1	0.0	1.386	0.0
adil	0	1	0	0	1	0.067	1.386	0.092
cerdas	1	0	0	0	1	0.0	1.386	0.0
indonesia	1	0	0	0	1	0.0	1.386	0.0
keadilan	0	1	0	0	1	0.067	1.386	0.092

kemakmuran	0	1	0	0	1	0.0	1.386	0.0
kendaraan	0	0	0	1	1	0.0	1.386	0.0
makmur	0	1	0	0	1	0.0	1.386	0.0
manfaatkan	0	0	0	1	1	0.0	1.386	0.0
masyarakat	0	1	0	0	1	0.0	1.386	0.0
mengangkat	0	0	1	0	1	0.0	1.386	0.0
menteri	0	0	1	0	1	0.0	1.386	0.0
mewujudkan	0	1	0	0	1	0.0	1.386	0.0
negeri	0	0	1	0	1	0.0	1.386	0.0
pemilu	0	1	0	0	1	0.0	1.386	0.0
pemimpin	1	0	0	0	1	0.067	1.386	0.092
pilihan	1	0	0	0	1	0.067	1.386	0.092
pilpres	0	0	0	1	1	0.0	1.386	0.0
politik	0	0	0	1	1	0.0	1.386	0.0
rakyat	1	0	0	0	1	0.067	1.386	0.092
rasyid	0	1	0	0	1	0.0	1.386	0.0
semoga	0	0	1	0	1	0.0	1.386	0.0
wakil	0	0	0	1	1	0.0	1.386	0.0

Table 11. Calculation of TF-IDF Value in Document "C"

kemakmuran	0	1	0	0	1	0.067	1.386	0.092
kendaraan	0	0	0	1	1	0.0	1.386	0.0
makmur	0	1	0	0	1	0.067	1.386	0.092
manfaatkan	0	0	0	1	1	0.0	1.386	0.0
masyarakat	0	1	0	0	1	0.067	1.386	0.092
mengangkat	0	0	1	0	1	0.0	1.386	0.0
menteri	0	0	1	0	1	0.0	1.386	0.0
mewujudkan	0	1	0	0	1	0.067	1.386	0.092
negeri	0	0	1	0	1	0.067	1.386	0.0
pemilu	0	1	0	0	1	0.067	1.386	0.092
pemimpin	1	0	0	0	1	0.0	1.386	0.0
pilihan	1	0	0	0	1	0.0	1.386	0.0
pilpres	0	0	0	1	1	0.0	1.386	0.0
politik	0	0	0	1	1	0.0	1.386	0.0
rakyat	1	0	0	0	1	0.0	1.386	0.0
rasyid	0	1	0	0	1	0.067	1.386	0.092
semoga	0	0	1	0	1	0.0	1.386	0.0
wakil	0	0	0	1	1	0.0	1.386	0.0

Table 12. Calculation of TF-IDF Value in Document "D"

Word	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	B	C	D		C	C	C
anies	1	1	1	1	4	0.714	0.0	0.0
baswedan	1	1	1	1	4	0.714	0.0	0.0
presiden	1	0	1	0	2	0.714	0.693	0.049
ketua	0	0	0	2	1	0.0	1.386	0.0
adil	0	1	0	0	1	0.0	1.386	0.0
cerdas	1	0	0	0	1	0.0	1.386	0.0
indonesia	1	0	0	0	1	0.0	1.386	0.0
keadilan	0	1	0	0	1	0.0	1.386	0.0
kemakmuran	0	1	0	0	1	0.0	1.386	0.0
kendaraan	0	0	0	1	1	0.0	1.386	0.0
makmur	0	1	0	0	1	0.0	1.386	0.0
manfaatkan	0	0	0	1	1	0.0	1.386	0.0
masyarakat	0	1	0	0	1	0.0	1.386	0.0
mengangkat	0	0	1	0	1	0.714	1.386	0.099
menteri	0	0	1	0	1	0.714	1.386	0.099
mewujudkan	0	1	0	0	1	0.0	1.386	0.0
negeri	0	0	1	0	1	0.714	1.386	0.099
pemilu	0	1	0	0	1	0.0	1.386	0.0
pemimpin	1	0	0	0	1	0.0	1.386	0.0
pilihan	1	0	0	0	1	0.0	1.386	0.0
pilpres	0	0	0	1	1	0.0	1.386	0.0
politik	0	0	0	1	1	0.0	1.386	0.0
rakyat	1	0	0	0	1	0.0	1.386	0.0
rasyid	0	1	0	0	1	0.0	1.386	0.0
semoga	0	0	1	0	1	0.714	1.386	0.099
wakil	0	0	0	1	1	0.0	1.386	0.0

Words	Word Appearance Frequency				DF	TF	IDF	Weight (W)
	A	B	C	D		D	D	D
anies	1	1	1	1	4	0.0625	0.0	0.0
baswedan	1	1	1	1	4	0.0625	0.0	0.0
presiden	1	0	1	0	2	0.0	0.693	0.0
ketua	0	0	0	2	1	0.125	1.386	0.173
adil	0	1	0	0	1	0.0	1.386	0.0
cerdas	1	0	0	0	1	0.0	1.386	0.0
indonesia	1	0	0	0	1	0.0	1.386	0.0
keadilan	0	1	0	0	1	0.0	1.386	0.0
kemakmuran	0	1	0	0	1	0.0	1.386	0.0
kendaraan	0	0	0	1	1	0.0625	1.386	0.086
makmur	0	1	0	0	1	0.0	1.386	0.0
manfaatkan	0	0	0	1	1	0.0625	1.386	0.086
masyarakat	0	1	0	0	1	0.0	1.386	0.0
mengangkat	0	0	1	0	1	0.0	1.386	0.0
menteri	0	0	1	0	1	0.0	1.386	0.0
mewujudkan	0	1	0	0	1	0.0	1.386	0.0
negeri	0	0	1	0	1	0.0	1.386	0.0
pemilu	0	1	0	0	1	0.0	1.386	0.0
pemimpin	1	0	0	0	1	0.0	1.386	0.0
pilihan	1	0	0	0	1	0.0	1.386	0.0
pilpres	0	0	0	1	1	0.0625	1.386	0.086
politik	0	0	0	1	1	0.0625	1.386	0.086
rakyat	1	0	0	0	1	0,0	1.386	0.0
rasyid	0	1	0	0	1	0.0	1.386	0.0
semoga	0	0	1	0	1	0.0	1.386	0.0
wakil	0	0	0	1	1	0.0625	1.386	0.086

F. Naïve Bayes

Multinomial Naïve Bayes is one of a series of text classification processes owned by Naïve Bayes by presenting a calculation method to find out the frequency of each word that appears in a document [19]. Table 11 is a collection of documents that have been given a weight value for each word that appears using the TF-IDF calculation method with an overall total weight in positive class documents of W(+) = 1,242, and the total number of W in negative class documents is W(-) = 1.048, and the total number of IDF in all classes is B' = 32.571.

Table 11. Word Weight

Words	W	
	Positive (c1)	Negative (c2)
anies	0	0
baswedan	0	0
presiden	0.046	0.049
ketua	0	0.173
adil	0.092	0
cerdas	0.092	0
indonesia	0.092	0
keadilan	0.092	0
kemakmuran	0.092	0
kendaraan	0	0.086
makmur	0.092	0
manfaatkan	0	0.086
masyarakat	0.092	0
mengangkat	0	0.099
menteri	0	0.099

mewujudkan	0.092	0
negeri	0	0.099
pemilu	0.092	0
pemimpin	0.092	0
pilihan	0.092	0
pilpres	0	0.086
politik	0	0.086
rakyat	0.092	0
rasyid	0.092	0
semoga	0	0.099
wakil	0	0.086

After successfully getting the prior probability calculation results, the next step in multinomial naïve Bayes is to do the calculation of the probability of each word to find out whether a word belongs to a certain category or class. Calculating Probability Can be done using the following equation [11]:

$$P(w|c) = \frac{Wct + 1}{(\sum w' \in VW\ ct') + B'}$$

Information:
Wct = Weight (W) often t is available in category c.
$(\sum w' \in VW\ ct')$ = The total amount of W from the total term that is inside category c.
B' = Total sum of IDF values on the entire document.

By using the formula or equation above, the calculation results are obtained probability as follows:

Table 12. Likelihood Calculation Results

Words	$P(w\|c) = \frac{Wct + 1}{(\sum w' \in VW\ ct') + B'}$	
	Positive	Negative
anies	0.030	0.030
baswedan	0.030	0.030
presiden	0.076	0.079
ketua	0.030	0.203
adil	0.122	0.030
cerdas	0.122	0.030
indonesia	0.122	0.030
keadilan	0.122	0.030
kemakmuran	0.122	0.030
kendaraan	0.030	0.116
makmur	0.122	0.030
manfaatkan	0.030	0.116
masyarakat	0.122	0.030
mengangkat	0.030	0.129
menteri	0.030	0.129
mewujudkan	0.122	0.030
negeri	0.030	0.129
pemilu	0.122	0.030
pemimpin	0.122	0.030
pilihan	0.122	0.030
pilpres	0.030	0.116
politik	0.030	0.116
rakyat	0.122	0.030
rasyid	0.122	0.030
semoga	0.030	0.129
wakil	0.030	0.116

The next calculation step is to determine the probability of the document against the sentiment class by comparing the values in the positive class and the values in the negative class using the following equation:

Table 13. Document Opportunity Calculation Results

anies baswedan pilihan rakyat indonesia pemimpin cerdas presiden	
$P(positif\ (c1), doc\ A)$	$P(positif\ (c2), doc\ A)$

P(c1) x P(anies) x P(baswedan) x P(pilihan) x P(rakyat) x P(indonesia) x P(pemimpin) x P(cerdas) x P(presiden) = 0.5 x 0.030 x 0.030 x 0.122 x 0.122 x 0.122 x 0.122 x 0.122 x 0.076 = 9.243	P(c2) x P(anies) x P(baswedan) x P(pilihan) x P(rakyat) x P(indonesia) x P(pemimpin) x P(cerdas) x P(presiden) = 0.5 x 0.030 x 0.030 x 0.030 x 0.030 x 0.030 x 0.030 x 0.030 x 0.079 = 8.638

From the results of the calculation of the document probability which can be seen in table 13, it is found that the probability of document A for the positive sentiment class has a greater value, namely 9,243 compared to the probability of a document for the negative sentiment class, namely 8,638. Then the conclusion that can be drawn in this calculation process is that the sample data document A is classified as a positive sentiment.

Results and Discussion

G. Classification Algorithm Modeling

In this study, the RapidMiner application is one of the software used in modelling Naïve Bayes Classifier, because the RapidMiner application is considered to be able to provide an effective and efficient experience starting from the process of retrieving the necessary data or information to analyzing data sentiment. Model naïve bayes classifier which has been made can be seen in figure 8.

Cross Validation Operator is a set of RapidMiner operators, such as retrieve, split data, Naïve Bayes, apply model, and performance which is used with the aim of knowing the results of the accuracy of a data where each data has been given a sentiment or label in the previous stage. The following is a list of operators used in modelling the Naïve Bayes classifier.

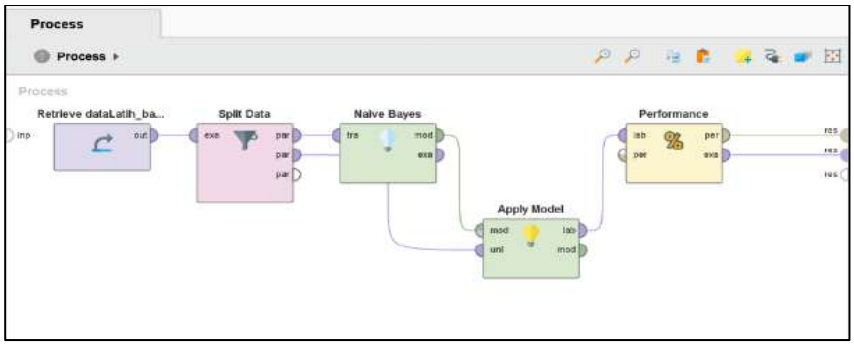


Figure 8. Cross Validation Operator

H. Algorithm Modeling Testing

The data testing process was carried out three times using the method confuse *matrix* contained in RapidMiner. The dataset used in this study has 585 tweets belonging to the class labelled positive and 548 tweets belonging to the class labelled negative, so the total data tweet a total of 1133 *tweets* which can be seen in the picture below.

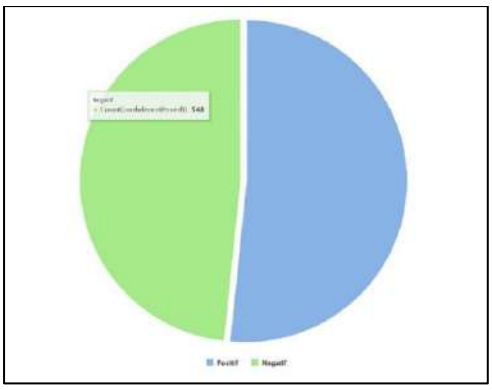


Figure 9. Negative Sentiment Analysis Results

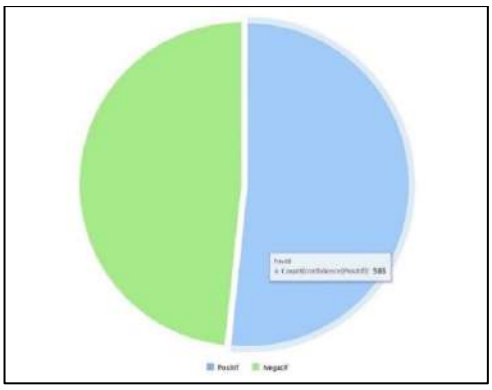


Figure 10. Positive Sentiment Analysis Results

The algorithm testing process *naïve bayes classifier* three times wherein the first test the ratio is used, namely 0.6 or 60% for training data and 0.4 or 40% for test data, then the second test uses the ratio 0.7 or 70% training data and 0.3 or 30% test data, and in the third test using a ratio of 0.8 or 80% training data and 0.2 or 20% test data. Model testing process *naïve bayes classifier* will produce a sentiment accuracy value of each data tweet.

1. Testing Model 1

Model testing *naïve bayes classifier* first used a ratio of 60% for training data and 40% for test data. The overall results of the accuracy level can be seen in Figure 11. The accuracy value obtained is 73.95%. Classification in this first test succeeded in producing 2,209 attributes derived from the weighting of 1,133 data tweets with 585 data labelled positive and 548 data labelled negative.

accuracy: 73.95%

	true Positif	true Negatif	class precision
pred. Positif	156	40	79.59%
pred. Negatif	78	179	69.65%
class recall	66.67%	81.74%	

Figure 11. Model Testing Results from Naive Bayes 1

2. Testing Model 2

Model testing *naïve bayes classifier* the second used a ratio of 0.7 or 70% for training data and 0.3 or 30% for test data. After the system applies the model to the test data, the overall accuracy level results are obtained which can be seen in Figure 12, Where the resulting accuracy value is 74.63%. The attributes produced in this second test are the same as the previous test, namely 2,209 attributes derived from the weighting results of 1,133 data tweets with 585 data labelled positive and 548 data labelled negative.

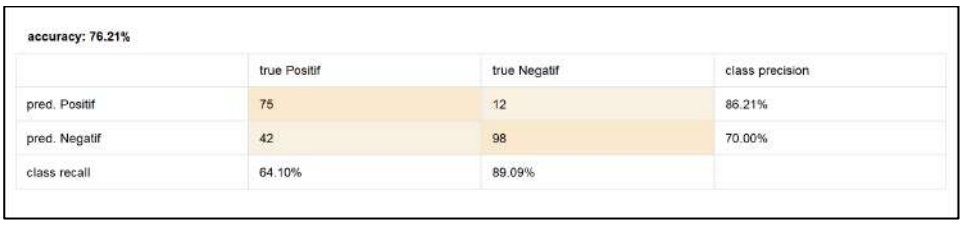
accuracy: 74.63%

	true Positif	true Negatif	class precision
pred. Positif	112	23	82.96%
pred. Negatif	63	141	69.12%
class recall	64.00%	85.98%	

Figure 12. Model Testing Results Naive Bayes 2

3. Testing Model 3

On model testing *naïve bayes classifiers* the third used a ratio of 0.8 or 80% for training data and 0.2 or 20% for test data. The overall results of the accuracy level can be seen in Figure 13, where the resulting accuracy value is 76.21% with the same number of attributes as the first and second tests.



accuracy: 76.21%

	true Positif	true Negatif	class precision
pred. Positif	75	12	86.21%
pred. Negatif	42	98	70.00%
class recall	64.10%	89.09%	

Figure 13. Model Testing Results Naïve Bayes 3

Based on the results of the three model testing processes that have been carried out, the average accuracy results are obtained which can be seen in table 14 below.

Table 14. Data Splitting Results

No.	Data Split Ratio	Accuracy
1.	60% : 40%	73.95%
2.	70% : 30%	74.63%
3.	80% : 20%	76.21%
Average		**74.93%**

From the table of results of the data splitting test it is known that the trials have been carried out on the classification model *naïve bayes* to 1133 data and the use of a data split ratio of 80% for training data and 20% for test data produces a higher accuracy value among the others, which is equal to 76.21% with a precision value class *recall* produced by true *positive* by 64.10%, and *class recall on true Negative* of 89.09%.

Conclusion

The process of testing the model which is carried out repeatedly using a combination of the number of ratios of test data and training data gives more optimal accuracy results with a comparison of the total values between positive and negative sentiments which are not far adrift proving that there are many Indonesian people who provide support for the 2024 Indonesian presidential candidate, but not a few also gave a statement of disagreement. The implementation of the Naïve Bayes classification algorithm can be considered quite good because it produces an accuracy percentage of 74.93%.

References

[1] N. Saputra, K. Nurbagja, and T. Turiyan, "Sentiment Analysis of Presidential Candidates Anies Baswedan and Ganjar Pranowo Using Naïve Bayes Method," J. Sisfotek Global., vol. 12, no. 2, pp. 114-119, 2022.

[2] H. Sujadi, "Analisis Sentimen Pengguna Media Sosial Twitter Terhadap Wabah Covid-19 Dengan Metode Naïve Bayes Classifier dan Support Vector Machine," J. Infotech., vol. 8, no. 1, pp. 22–27, 2022.

[3] APJII. (2022). P*rofil Internet Indonesia 2022* [online]. Available: https://apjii.or.id/survei /surveiprofilinternetindonesia2022-21072047

[4] M. Siahaan, "An Analysis of Contract Employee Performance Assessment Using Machine Learning," J. of Inf. and Tel. Eng., vol. 5, no. 1, pp. 121-131, 2021.

[5] M. Irwanto *et al*, "Sentiment Analysis and Classification of Forest Fires in Indonesia," Ilkom J. Ilmiah., vol. 15, no. 1, pp. 175-185. 2023.

[6] P. Arsi, B. A. Kusuma, and A. Nurhakim, "Analisis Sentimen Pindah Ibu Kota Berbasis Naive Bayes Classifier," J. Inf. Upgris., vol. 7, no. 1. pp. 1-6, 2021.

[7] R. Ardiansyah, "Analisis sentimen calon presiden dan wakil presiden periode 2019-2024 pasca debat pilpres di Twitter," J. ScientiCO: Comp. Sci and Inf., vol. 2, no. 1, pp. 21-28, 2019.

[8] R. Mubarok, "Analisis Sentimen Pengguna Twitter Terhadap Kebijakan Pemberlakukan Pembatasan Sosial Berskala Besar (PSBB) Dengan Metode Naïve Bayes," J. Sili. Seri. Sains. dan Tek., vol. 7, no. 1, 2021.

[9] R. Talib, M. K. Hanif, S. Ayesha, F. Fatima, "Text Mining: Techniques, Applications, and Issues," Int. J. of Adv. Comp. Sci. and Appl., vol. 7, no. 11, pp. 414-418, 2016.

[10] H. N. Alhazmi, "Text Mining in Online Social Networks: A Systematic Review," Int. J.of Comp. Sci. and Net. Sec., vol. 22, no. 3, pp. 396-404, 2022.

[11] D. Indriani, "Analisis Sentimen Pada Tweet Dengan Tagar# kpujangancurang Menggunakan Metode Naive Bayes," Ph. D. dissertation, Universitas Islam Riau., Riau., Indonesia., 2019.

[12] M. R. F. Sya'bani, U. Enri, and T. N. Padilah, "Analisis Sentimen Terhadap Bakal Calon Presiden 2024 Dengan Algoritme Naïve Bayes," J. Ris. Kom., vol. 9, no. 2, pp. 265-273, 2022.

[13] M. A. Rosari, W. Wasino, and T. Tony, "Analisis Sentimen Tanggapan Masyarakat Terhadapbantuan Sosialpemerintah Di Masa Pandemi Covid-19 Pada Platform Twitter," J. Ilmu. Kom. Dan Sist. Inf., vol. 10, no. 1, 2022.

[14] R. Nooraeni, A. B. Safiruddin, A. F. Afifah, K. D. Agung, and N. N. Rosyad, "Analisis Sentimen Publik Terhadap Sistem Zonasi Sekolah Menggunakan Data Twitter Dengan Metode Naïve Bayes Classification," J. Fak. Exac., vol. 12, no. 4, pp. 315-322, 2020.

[15] A. Deolika, K. Kusrini, and E. T. Luthfi, "Analisis Pembobotan Kata Pada Klasifikasi Text Mining," J. Tek. Inf., vol. 3, no. 2, pp. 179-184, 2019.

[16] R. T. Wahyuni, D. Prastiyanto, and E. Supraptono, "Penerapan algoritma cosine similarity dan pembobotan tf-idf pada sistem klasifikasi dokumen skripsi," J. Tek. Elek., vol. 9, no. 1, pp. 18-23, 2017.

[17] D. E. D. P. Sari, Y. A. Sari, and M. T. Furqon, "Pembentukan Daftar Stopword menggunakan Zipf Law dan Pembobotan Augmented TF-Probability IDF pada Klasifikasi Dokumen Ulasan Produk," J. Peng. Tek. Inf. Dan Ilmu Kom., vol. 4, no. 1, pp. 406-412, 2020.

[18] M. G. Pradana, "Penggunaan fitur wordcloud dan document term matrix dalam text mining," J. Ilmiah. Info., vol. 8, no. 1, pp. 38-43, 2020.

[19] A. H. Rahayu, and A. Sudrajat. "Naïve Bayes Performance in Analysis of Public Opinion Sentiment Against COVID-19," J. of Appl. Intell. Sys., vol. 7, no. 3, pp. 237-245, 2022.