

# Hoax Detector of Covid 19 Indonesia in twitter using Rocchio Classification Method

Revi Chandra Riana<sup>1</sup>, Yuliant Sibaroni<sup>2</sup>

<sup>1,2,3</sup> Universitas Telkom, Bandung

revicrr@students.telkomuniversity.ac.id<sup>1</sup>, yuliantsibaroni@telkomuniversity.ac.id<sup>2</sup>

---

## Abstrak

Perkembangan teknologi membuat komunikasi semakin mudah dengan menggunakan media sosial, begitupun berita yang tersebar didalamnya tidak dapat tervalidasi segala isinya. Hoax adalah berita yang diterbitkan tanpa mengacu pada berita yang benar atau berita yang salah. Untuk mengetahui hoax atau tidak suatu berita, kita bisa menggunakan metode klasifikasi khususnya klasifikasi Rocchio. Dengan akurasi yang cukup tinggi untuk mengklarifikasi berita, diharapkan hoax dapat dengan mudah teridentifikasi menggunakan klasifikasi Rocchio. Untuk kasus ini, metode n-gram digunakan sebagai tambahan parameter untuk mengukur tingkat akurasi dan keefektifan dari klasifikasi Rocchio. Hasil akhir berupa evaluasi dari tingkat akurasi Klasifikasi Rocchio untuk pengujian dengan unigram sebesar 0,87, untuk bigram sebesar 0,77, dan untuk trigram sebesar 0,66. Berdasarkan pengujian yang dilakukan menggunakan n-gram, disimpulkan bahwa gabungan urutan n-gram menentukan akurasi yang didapatkan, semakin banyak urutan n-gram akurasi akan semakin kecil

**Kata kunci :** *Hoax, Klasifikasi, Rocchio, Data, Twitter.*

---

## Abstract

The development of technology makes communication easier by using social media, as well as the news that is spread in it cannot be validated in all its contents. Hoax is news that is published without referring to true or false news. To find out whether a hoax is news or not, we can use a classification method, especially the Rocchio classification. With a high enough accuracy to clarify news, it is hoped that hoaxes will be identified easily using Rocchio classification. In this case, the n-gram method is used as an additional parameter to measure the accuracy and effectiveness of Rocchio classification. Final result is accuracy evaluation of Rocchio Classification for testing with a unigram of 0.87, a bigram of 0.77, and a trigram of 0.66. Based on the tests using n-grams, it was concluded that the combined sequence of n-grams determines the accuracy obtained, the more sequences of n-grams the accuracy will be smaller

**Keywords:** *Hoax, Classification, Rocchio, Data, Twitter.*

---

## 1. Pendahuluan

### 1.1 Latar Belakang

Kemajuan teknologi yang pesat membuat komunikasi lebih mudah dengan menggunakan media sosial seperti Twitter, Whatsapp, Instagram, Facebook dan media sosial lainnya. Sosial media bekerja sebagai jalur komunikasi di jaringan internet yang membuat pengguna bisa saling berinteraksi tanpa batasan jarak dan waktu. Twitter adalah platform media sosial yang memiliki basis berupa *microblogging* dan diluncurkan tanggal 13 Juli tahun 2006 [1]. Sebuah tweet memuat pesan singkat dengan maksimum 280 karakter [2]. Dalam sosial media twitter terdapat hal positif dan negative. Selain membuat masyarakat lebih mudah dalam berbagi informasi dan saling berhubungan, bisa juga membuat pengaruh negatif, terhadap pengguna, seperti penyalahgunaan informasi seperti hoax, penipuan dan lainnya

Hoax adalah berita yang diterbitkan tanpa mengacu pada berita yang benar atau berita yang salah [3]. Pada umumnya, hoax terjadi atau digunakan ketika seseorang mencoba mengelabui pihak lain untuk mempercayai sebuah kejadian yang tidak pernah terjadi [4]. Dalam hal ini, covid 19 di twitter menjadi sasaran utama hoaks bagi pihak yang ingin membuat kepanikan dan ketakutan diantaranya hoaks tentang kalung penyembuh, konspirasi covid adalah akal-akalan pemerintah dan sejenisnya.

Menurut survey yang dilakukan tentang hoax secara nasional yang dilakukan bahwa jalur utama berita hoaks tersebar adalah sebesar 34,90% dari situs web, aplikasi chatting 62,80% dan 92,40% untuk sosial media sebagai persentase tertinggi [5]. Untuk mengetahui berita hoax dan tidaknya pada internet, akan diperlukan sebuah metode berupa klasifikasi berita yang bisa dilakukan secara manual atau

otomatis oleh sistem. Klasifikasi dokumen adalah sebuah metode untuk mencari kumpulan model dalam dokumen yang kemudian akan dilakukan pemisahan dan pendeskripsian data sesuai kategori yang dimiliki oleh dokumen yang akan diklasifikasi [6].

Mengacu kepada riset [5], dalam pendeteksian hoax, terdapat sebuah metode yang lebih baik dalam beberapa aspek dibanding metode Naïve Bayes yang bernama metode Rocchio. Pada penelitian Aulia dalam sebuah studi yang berbeda, akurasi rata-rata yang didapatkan untuk Multinomial Naïve Bayes sebanyak 65,835% [7]. Berdasarkan hasil riset tersebut, klasifikasi Naïve Bayes memiliki sebuah kekurangan bernama underflow dimana nilai peluang yang dihasilkan sangat kecil dibanding dengan klasifikasi Rocchio, maka klasifikasi Rocchio akan mempunyai hasil lebih unggul untuk pengujian antara hoax dengan non-hoax [7]. Klasifikasi Rocchio merupakan klasifikasi linear yang berbasis pendekatan hipotesis, dimana dokumen antara satu kelas dan kelas lain tidak akan mirip satu sama lain sehingga overlap dapat dihindari antar kelas yang berbeda. Maka dari itu, akan dilakukan penelitian berita hoax tentang Covid-19 di twitter menggunakan klasifikasi Rocchio. Tujuannya untuk mengetahui tingkat akurasi dan keefektifan metode klasifikasi Rocchio untuk klasifikasi berita hoax mengenai Covid-19 Indonesia di twitter.

## 2. Studi Terkait

Penelitian terkait hoax pernah dilakukan oleh Petkovic [8], Vukovic [9], Chen [10]. Penelitian yang dilakukan Petkovic [8] menghasilkan sistem pendeteksi hoax otomatis untuk Croatians CARNet CERT menggunakan metode pengenalan edit jarak yang memiliki kekurangan berupa waktu komputasi yang lama. Sementara menurut penelitian Vukovic [9], penelitian ini menghasilkan sistem pendeteksi hoax cerdas menggunakan artificial neural network sebagai dasar dengan metode perbandingan antara pesan hoax yang telah dideteksi dan diklasifikasi terhadap pesan hoax yang sudah diketahui polanya. Namun apabila terdapat pola hoax baru, maka sistem tidak dapat mendeteksi hoax tersebut. Hoax sendiri menurut Chen[10], dapat membuat banyak individu terpengaruh dengan merusak kredibilitas dan citra atas suatu hal yang terkena hoax. Penelitian Chen[10] menghasilkan sistem pendeteksian hoax menggunakan Jarak Levenshtein, sistem melakukan pendeteksian hoax dengan membandingkan e-mail yang sudah di ekstraksi terhadap database yang sudah ada. Sebagai fitur nilai tambah, tingkat positif hoax juga ditentukan oleh bagaimana respon penerima terhadap email yang dideteksi memiliki kemungkinan hoax. Maka dari itu, sistem memiliki nilai prediktif yang tinggi namun lemah dalam mengidentifikasi email asli yang tidak ada di database. Sementara penelitian Rasywir dan Purwarianti [11] menghasilkan sistem pendeteksian klasifikasi hoax menggunakan pendekatan statistik. Metode yang digunakan untuk mencari hasil terbaik dalam membangun sistem klasifikasi hoax adalah perbandingan. Penelitian ini membandingkan beberapa metode dalam setiap proses diantaranya dari algoritma pembelajaran mesin, praproses, ekstraksi fitur, dan seleksi fitur. Dengan hasil berupa praproses terbaik dengan tanpa stemming dan stopwords, ekstraksi fitur terbaik adalah unigram, dan seleksi fitur terbaik didapatkan dengan operasi menggunakan union untuk mutual dan information gain. Sementara untuk algoritma pembelajaran mesin, didapatkan algoritma Naïve Bayes dengan akurasi 91.36% dibanding C4.5 dan SVM.

Penelitian mengenai berita-berita hoaks yang menggunakan bahasa indonesia dan bahasa inggris menggunakan klasifikasi Rocchio belum ditemukan, akan tetapi menurut hasil penelitian Manning [13], klasifikasi Rocchio mempunyai 6 cara untuk mengkalkulasi batas kelas dengan memanfaatkan centroid untuk menentukan batas antar kelas. Oleh karena itu, klasifikasi Rocchio dipilih sebab memiliki keunggulan dengan tidak adanya underflow seperti pada klasifikasi Naive Bayes [14].

Oleh karena itu, penelitian jurnal Hoax Detector Covid-19 Indonesia di twitter menggunakan Klasifikasi Rocchio mempunyai tujuan untuk mengklasifikasikan berita hoaks bahasa Indonesia dengan mengacu kepada keunggulan perbandingan tingkat akurasi dari klasifikasi Rocchio terhadap Multinomial Naïve Bayes berdasarkan penelitian yang dilakukan pada tahun 1997 oleh Joachims [7] dengan data yang bersumber dari media sosial dan pesan siaran serta media-media berita online seperti CNN [7].

Jurnal Hoax Detector Covid-19 Indonesia di twitter menggunakan Klasifikasi Rocchio ini dikembangkan berdasarkan riset tentang hoaks detector oleh Pantouw [12], yang akan bertujuan untuk mencari keunggulan nilai akurasi dan keefektifan pengerjaan dataset dari klasifikasi Rocchio untuk deteksi hoax Covid-19 di twitter dibanding riset-riset sebelumnya.

## 2.1 Ujaran Kebencian

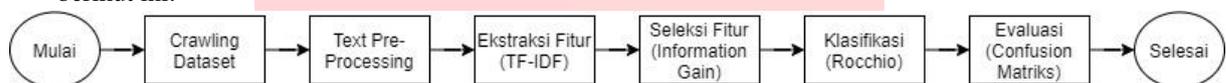
Ujaran kebencian adalah ujaran yang didalamnya terdapat unsur kebencian, bersifat menyerang suatu pihak dan dengan emosi yang berapi-api dengan maksud untuk menyebabkan efek khusus yang bersifat negatif, secara langsung maupun tidak langsung, yang memiliki tujuan untuk menginspirasi suatu pihak untuk melakukan kekerasan, menyakiti suatu individu dan kelompok lain [23].

## 2.2 Twitter

Twitter baru-baru ini muncul sebagai media sosial yang populer untuk pengguna mengutarakan pendapat tentang berbagai hal, termasuk lelucon, berita, pandangan mereka dan bahkan suasana hati mereka [15]. Twitter adalah sistem berbagi informasi, tempat pengguna mengikuti pengguna lain. Informasi tersebut terdiri dari pesan teks pendek yang disebut tweet. Dengan antarmuka yang sederhana dimana hanya kurang dari 200 pesan karakter dapat dibagikan. Pesan yang dibagikan pengguna segera dikirim ke pengikutnya yang memungkinkan mereka untuk menyebarkan informasi yang diterima. Selain diterima oleh pengikut, tweet juga dapat diambil melalui sistem pencarian dan alat lainnya.

## 3. Sistem yang Dibangun

Gambaran umum untuk sistem hoaks detektor direpresentasikan dengan rancangan diagram alir berikut ini.



Gambar 1. Diagram Alir Sistem

### 3.1 Dataset

Dari hasil tweet yang dihasilkan tiap-tiap harinya via pengguna twitter, tweet-tweet ini bisa dijadikan sumber-sumber informasi untuk pengumpulan data dengan cara melakukan proses crawling data via fitur pengembang Twitter menggunakan Application Program Interface (API) yang telah disediakan oleh twitter. API twitter hanya bisa diakses melalui permintaan otentikasi bertahap. Twitter memperbolehkan membuka atau mengakses tweet dan permintaannya sesuai protocol via pengguna dari Twitter yang memiliki izin sah. Akses masuk ke dalam API twitter memiliki batasan sesuai tipe akun pengembang twitter. Batasan ini diaplikasikan terhadap tingkatan untuk pengguna dan tingkatan untuk aplikasi [16]. Penggunaan data tweet yang dikumpulkan sebanyak 12.162 tweet yang dilakukan oleh 2 orang dengan pembagian 50% dari total tweet per orang menggunakan 1 keyword Covid dan 2 hashtag yaitu Covid dan Covid-19, data yang akan dikumpulkan akan dimasukkan dalam bentuk CSV dengan label 1 adalah hoax dan 0 adalah non hoax. Labeling ulang dari data sebelumnya akan dilakukan setelah data hasil labeling secara individual oleh 2 orang digabungkan menjadi 12.162 tweet, dengan hasil labeling berupa 7.068 tweet (58%) adalah non-hoax dan 5.095 tweet (42%) adalah hoax, kemudian proses labeling dilakukan oleh 2 orang dengan 1 orang sebagai pelabel dan 1 orang lagi sebagai pengkonfirmasi tweet hoax berdasarkan indikasi atas website-website yang kredibel seperti website-website dibawah :

Tabel 1. Website Kredibel

Tweet
Kepanikan dan Ketakutan Massal (Liputan6.com, <a href="https://m.liputan6.com/news/read/3867707">https://m.liputan6.com/news/read/3867707</a> )
Nama samaran(Kompasiana.com, <a href="https://www.kompasiana.com/komentar/andinimentari/5de24f97097f361813220a42/terjerat-pidana-karena-berita-hoax">https://www.kompasiana.com/komentar/andinimentari/5de24f97097f361813220a42/terjerat-pidana-karena-berita-hoax</a> )
12 Ciri-ciri hoax ( <a href="https://www.beritasatu.com/digital/547545/">https://www.beritasatu.com/digital/547545/</a> )

Tabel 2. Contoh Dataset

No.	Author	Tweet	Label	Indikator
1.	@hailanhamid	Virus Covid-19 Indonesia merupakan hasil Konspirasi Dokter-Dokter dan Pemerintah Indonesia	1 (Hoax)	Terdapat kata-kata provokatif dalam kalimat tweet, akun twitter tidak kredibel, tidak netral karena menyerang suatu kubu
2.	Berita Covid	"Supply oksigen di Jawa dan Bali dapat dimaksimalkan pemasokannya, mengingat adanya peningkatan kasus Covid-19 yang jumlahnya hingga 6 sampai 8 kali lipat, ujar Presiden Jokowi.	0 (Non-Hoax)	Tweet tidak memiliki indikasi hoax seperti ujaran kebencian, kata provokatif. Tweet berdasarkan akun yang kredibel (memiliki tanda verified), tweet bersifat netral tidak menyerang pihak manapun
3.	SinarPlus	Jokowi: Virus Covid-19 Bisa Masuk Melalui Handphone	1 (Hoax)	Tweet berdasarkan akun yang kurang kredibel, tweet dapat menyebabkan kepanikan mengingat masyarakat banyak yang menggunakan handphone tanpa bukti dan riset yang jelas.
4.	Harian Metro	Kementrian Kesehatan Indonesia: 67 Persen Masyarakat Indonesia mempercayai bahwa Vaksin Ampuh Atasi Virus Corona-Nasional <a href="https://t.co/oujrjHTuB">https://t.co/oujrjHTuB</a> #Covid #Covid19	0 (Non-hoax)	Tweet berdasarkan akun yang kredibel dan isi tweet didukung dengan fakta dalam link yang diberikan berupa voting

### 3.2 Pre-processing

Pre-processing adalah salah satu komponen penting pada banyak algoritma text mining [17]. Penstrukturan data agar dapat lebih mudah digunakan pada proses selanjutnya. Terdapat 5 tahapan pada proses preprocessing ini yaitu, Cleansing adalah penghilangan karakter yang tidak diperlukan seperti angka dan isyarat baca, Case Folding berupa perubahan huruf menjadi huruf kecil terhadap seluruh huruf yang diproses, Tokenizing berupa pemotongan teks dari dokumen menjadi bagian yang lebih kecil yang bernama token dan menghilangkan karakter khusus seperti simbol, Stopword Removal berupa penghapusan kata yang tidak sesuai dan tidak berhubungan dengan hoax detector, dan terakhir yaitu, Stemming adalah suatu proses perubahan kata yang telah ditokenisasi sebelumnya dari kata turunan menjadi kata dasar. Proses perubahan stemming ini dilakukan dengan menghapus imbuhan-imbuhan pada kata (affixes) yang terdiri dari imbuhan pada awal kata (prefixes), imbuhan sisipan

(infixes), imbuhan pada akhiran kata (suffixes) dan imbuhan pada awal dan akhir kata (confixes) pada kata turunan.



3.2.1 *Cleansing*

Penghilangan karakter yang tidak diperlukan seperti angka dan isyarat baca. Berikut adalah contoh dari *Cleansing*.

**Table 3. Contoh Cleansing**

Sebelum	Sesudah
Kota Bandung menjadi kota penambahan kasus terbanyak sebanyak 1.500 kasus	Kota Bandung menjadi kota penambahan kasus terbanyak sebanyak kasus

3.2.2 *Case Folding*

Perubahan huruf menjadi huruf kecil terhadap seluruh huruf yang ada di kalimat yang di proses. Berikut contoh dari *Case Folding*:

**Table 4. Contoh Case Folding**

Sebelum	Sesudah
Kota Bandung menjadi kota penambahan kasus terbanyak sebanyak kasus	kota bandung menjadi kota penambahan kasus terbanyak sebanyak kasus

3.2.3 *Tokenizing*

Proses pembuatan token kata dari pemotongan teks dari dokumen [13].

**Table 5. Contoh Tokenizing**

Sebelum	Sesudah
kota bandung menjadi kota penambahan kasus terbanyak sebanyak kasus	“kota” “bandung” “menjadi” “kota” “penambahan” “kasus” “terbanyak” “sebanyak” “kasus”

3.2.4 *Stopword Removal*

Pada proses ini berupa penghapusan kata yang tidak sesuai dan tidak berhubungan dengan hoax detector dan penghapusan kata-kata yang mengandung ujaran kebencian. Berikut contoh dari *Stopword Removal* :

**Table 6. Contoh Stopword Removal**

Sebelum	Sesudah
“kota” “bandung” “menjadi” “kota” “penambahan” “kasus” “terbanyak” “sebanyak” “kasus”	“kota” “bandung” “kota” “penambahan”

3.2.5 *Stemming*

Proses untuk merubah kata yang telah ditokenisasi sebelumnya dari kata turunan menjadi kata dasar. Berikut contoh dari *Stemming* :

**Table 7. Contoh Stemming**

Sebelum	Sesudah
“kota” “bandung” “kota” “penambahan”	“kota” “bandung” “kota” “tambah”

**3.3 Ekstraksi Fitur (TF-IDF Unigram, Bigram, Trigram)**

Pada tahap ini dilakukan perubahan atau pengestrakan data berupa dokumen teks menjadi bentuk vektor untuk digunakan dalam proses selanjutnya, untuk jurnal ini akan mengekstrak dataset twitter tentang Covid-19. Ekstraksi fitur digunakan untuk menemukan fitur paling ringkas dan informatif untuk meningkatkan efisiensi dalam pemrosesan data [18]. Untuk metode yang digunakan adalah metode TF-IDF, sebuah metode yang memiliki fungsi untuk mengkalkulasikan bobot tiap-tiap kata umum yang dipergunakan. Metode ini memiliki kelebihan berupa efisiensi yang baik, penggunaan yang mudah serta hasil akurasi yang tinggi. Metode ini mengkalkulasikan nilai TF dan IDF untuk tiap-tiap kata yang sudah menjadi bentuk token dalam dokumen yang terdapat pada korpus. Sederhanya, metode TF-IDF ini dipergunakan untuk mencari banyaknya jumlah kata yang muncul dalam sebuah dokumen [19]. Berikut adalah rumus TF-IDF :

$$TF_{t,d} = \sum_{x \in d} f_t(x) \quad (1)$$

Frekuensi kemunculan *term* dalam dokumen  $d_i$ ,  
dimana nilai  $f_t(x)$  adalah 1 jika  $x = t$  dan 0 jika  $x \neq t$ .

$$IDF_t = \log\left(\frac{|D|}{df_t}\right) \quad (2)$$

$IDF_t$  adalah jumlah dokumen yang mengandung *term*  $t$   
 $|D|$  adalah jumlah dokumen  
 $df(t)$  adalah *term*  $t$

$$W_{ij} = tf_{ij} \times \log\left(\frac{n}{df_j}\right) \quad (3)$$

$W_{ij}$  : bobot term ( $t_j$ ) terhadap dokumen ( $d_i$ )  
 $tf_{ij}$  : jumlah kemunculan term ( $t_j$ ) dalam dokumen ( $d_i$ )  
 $n$  : jumlah semua dokumen yang ada dalam database  
 $df_j$  : jumlah dokumen yang mengandung term ( $t_j$ )

Kemudian setelah dilakukan perhitungan seperti diatas, akan dilakukan L1 Normalisasi atau dapat dikenal juga dengan *Jarak Manhattan*. L1 Normalisasi ini berfungsi untuk jarak antara vektor berdasarkan perbedaan absolut dari komponen vektor. Sementara untuk melanjutkan perhitungan TF-IDF sebelumnya, dikarenakan pada library tidak dilakukan normalisasi maka fungsi CountVectorizer digunakan untuk mendapatkan vektor TF, kemudian baru dapat dilakukan pengaplikasian normalisasi menggunakan rumus sebagai berikut [25] :

$$\|x\|_1 = \sum_i |x_i| \quad (4)$$

$\|x\|_1$  : Normalisasi L1  
 $|x_i|$  : Hasil dari TF

Lalu dilakukan perkalian TF dan IDF agar mendapatkan vektor TFIDF yang akan digunakan pada tahap selanjutnya.

Model n-gram adalah model probabilistic untuk melakukan prediksi item selanjutnya dalam urutan pada item. Item ini dapat berupa huruf/karakter, n-gram merupakan wadah dari kumpulan kata yang memiliki panjang  $n$  kata [24]. TF-IDF yang digunakan penulis menggunakan fitur kombinasi Ngram, yaitu unigram, bigram, trigram kemudian ada gabungan n-gram berupa uni+bigram, bi+trigram dan uni+bi+trigram.

### 3.4 Seleksi Fitur (Information Gain)

Dari hasil TF-IDF sebelumnya pada proses Ekstraksi Fitur, akan dilakukan penghilangan fitur-fitur yang tidak berguna diantaranya data yang redundant dan tidak relevan pada Seleksi Fitur yang menggunakan metode Information Gain. Manfaatnya untuk meningkatkan kinerja pada tahap klasifikasi [20]. Seleksi fitur ini akan memilih data dari hasil TF-IDF menggunakan teknik Information Gain untuk dipisah menjadi bagian baru bernama selected feature yang akan berbentuk chart, disana akan terlihat perbandingan antara dua chart, yaitu: original feature (data yang asli) dan selected feature (data yang sudah melalui proses seleksi fitur). Dimana data dari selected feature ini nantinya akan dilanjutkan untuk diproses pada bagian klasifikasi. Beberapa manfaat dari Seleksi Fitur sebagai berikut :

- Menambah kualitas data
- Meningkatkan akurasi model
- Meningkatkan performa untuk mendapatkan akurasi yang lebih baik
- Mempercepat waktu proses menjalankan algoritma
- Mengurangi storage yang digunakan

Information Gain digunakan untuk menghitung pengaruh fitur pada sebuah kalimat. Untuk mendapatkan Information Gain diperlukan perhitungan entropy sebelum dan sesudah data dipisahkan [20]. Adapun rumus Information Gain sebagai berikut :

$$\text{Entropy}(S) = \sum_{i=1}^k (P_i) \log_2 (P_i) \quad (5)$$

S : dataset  
k = jumlah S  
Pi = jumlah sampel i

$$\text{Entropy}(S,A) = \sum_{i=1}^v \left( \frac{S_v}{S} * \text{Entropy}(S_v) \right) \quad (6)$$

v : Semua nilai yang mungkin dari atribut A  
Sv : subset dari S dimana atribut A bernilai v

$$\text{Gain}(S,A) = \text{Entropy}(S) - \text{Entropy}(S,A) \quad (7)$$

Gain (S,A) : Nilai information gain  
Entropy (S) : Nilai entropy sebelum pemisahan  
Entropy (S,A) : Nilai entropy setelah pemisahan

Setelah dilakukan perhitungan dengan Information Gain, akan mendapatkan fitur yang diseleksi yang memiliki nilai gain tersendiri untuk setiap fitur. Hanya fitur yang sudah diseleksi dan mendapat nilai Gain yang tinggi dan sesuai yang dapat digunakan menjadi fitur data latih baru yang selanjutnya akan dilakukan perhitungan dalam proses selanjutnya yaitu klasifikasi Rocchio.

### 3.5 Klasifikasi Rocchio

Selanjutnya hasil dari seleksi fitur akan disave dalam bentuk sparse matriks yang kemudian akan dipanggil menggunakan library train\_test\_split yang berfungsi memisahkan data yang berbentuk dalam sparse matriks tadi menjadi data latih dan data uji. Kemudian klasifikasi akan dilakukan untuk data uji

yang akan menghasilkan confusion matriks pada tahap Evaluasi. Klasifikasi Rocchio adalah klasifikasi dengan perbandingan antara sebuah dokumen dan daftar term yang terdiri dari positif dan negatif untuk tiap-tiap kategori yang dibandingkan, lalu akan dilakukan klasifikasi berdasarkan bobot dari term positif dan negatif yang dilakukan sebelumnya. Kemudian dilakukan perhitungan bobot berdasarkan fitur yang sudah diseleksi menggunakan perkalian TF-IDF, kemudian nilai bobot yang didapat dari perhitungan TF-IDF tersebut dikalkulasikan terhadap centroid dari kelas masing-masing. Hasil perhitungan nilai dari centroid tersebut akan dimasukkan kedalam tabel centroid Rocchio yang terdiri dari nilai centroid dan term. Centroid kelas  $c$  dihitung dengan persamaan :

Tabel 8. Alur algoritma Rocchio

TrainRocchio(C, D)
1. for each $c_j \in C$
2. do $D_j \leftarrow \{d: \langle d, c_j \rangle \in D\}$
3. $u_j \leftarrow \frac{1}{ D_j } \sum_{d \in D_j} v^-(d)$
4. return $\{u_1, \dots, u_j\}$
Apply Rocchio ( $\{u_1, \dots, u_j\}, d$ )
1. return $\arg \min_j,  u_j - v^-(d) $

Klasifikasi rochio menggunakan rumus tersebut untuk menentukan batas-batas antar kelas yang menggunakan centroid sebagai pembatas antar kelas dalam dataset. Dimana inputnya berupa dua buah dokumen, dokumen C dan dokumen D. Untuk setiap kelas centroid  $c_j$  dalam C maka, himpunan dokumen  $D_j$  akan memiliki input berupa  $d$ (kata) yang terdiri dari gabungan antara kata dan kelas centroid  $c_j$  dalam dokumen D. Kemudian vektor  $j$  yang telah dinormalisasi akan mendapatkan input berupa hasil dari vektorisasi himpunan dokumen  $D_j$  yang dikalikan dengan jumlah kata dari himpunan dokumen  $D_j$  kemudian dikalikan dengan vektor kata( $d$ ). Lalu akan menghasilkan output berupa vektor dari index 1 hingga index  $j$ . Kemudian pada bagian apply rochio akan memiliki input dari hasil vektor  $i$  hingga  $j$  yang sebelumnya sudah ada dan kata ( $d$ ). Setelah diaplikasikan akan memiliki output berupa pengembalian index minimum  $j$ , dan hasil dari pengurangan nilai panjang vektor  $j$  dan vektor  $d$ . Dalam algoritma diatas, hasil aplikasi rochio berupa perhitungan kemiripan sebuah centroid kelas yang menggunakan jarak sebagai parameter perhitungan. Ketika menggunakan perhitungan jarak ( $|u_j - v^-(d)|$ ), maka yang akan dilakukan adalah pencarian dari kelas yang mempunyai jarak terkecil terhadap kueri.

Cosine similarity digunakan dalam perbandingan kemiripan antara dokumen satu dengan dokumen lainnya, dengan perbandingan berupa query dengan dokumen latih untuk kasus ini [22]. Perhitungan cosine similarity diawali dengan perkalian scalar antara kueri terhadap dokumen lalu dilakukan penjumlahan, lalu dilakukan perkalian dari panjang dokumen dengan panjang kueri yang telah dikuadratkan, kemudian dihitung dengan akar pangkat dua. Tahap terakhir adalah perkalian scalar tersebut dibagi dengan hasil perkalian antara panjang dokumen terhadap kueri [21]. Adapun Cosine Similarity memiliki rumus seperti dibawah ini :

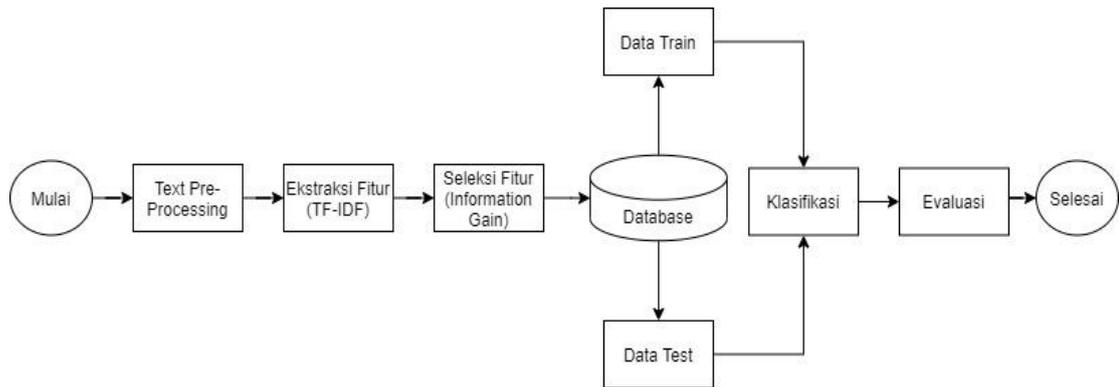
$$sim(d1, d2) = \frac{v(d1).v(d2)}{\|v(d1)\|.\|v(d2)\|} \quad (8)$$

$v(d1)$  : Nilai vektor centroid untuk setiap kelas

$v(d2)$  : Nilai vektor data uji

$\|v(d1)\|$  : Nilai panjang vektor centroid

$\|v(d2)\|$  : Nilai panjang vektor data uji



**3.6 Confusion Matriks**

Tahap terakhir evaluasi adalah mengeluarkan hasil berupa model dari machine learning yang menggunakan metode confusion matriks dengan hasil akhirnya adalah precision, recall, dan accuracy dari hasil dataset yang telah dilakukan data latih dan data uji pada tahap klasifikasi.

$$Precision = \frac{TP}{TP+FP} \tag{10}$$

$$Recall = \frac{TP}{TP+FN} \tag{11}$$

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{12}$$

- TP : True Positive
- FP : False Positive
- TN : True Negative
- FN : False Negative

**4. Evaluasi Sistem**

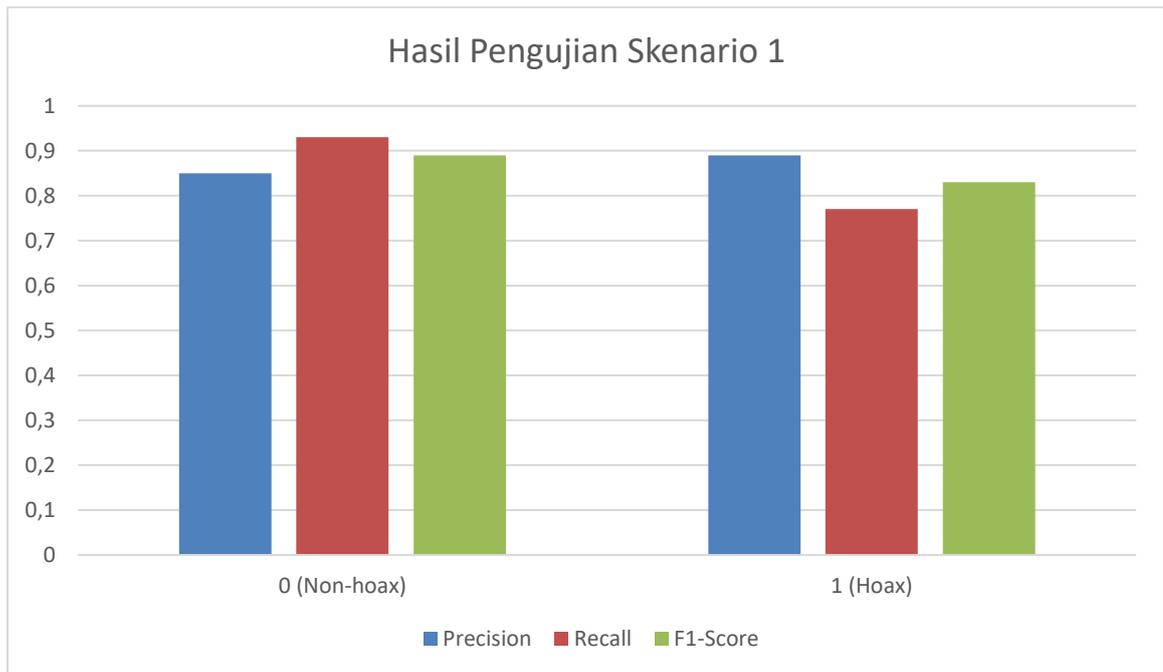
Evaluasi sistem ini bertujuan untuk mengetahui hasil dari tujuan awal yang akan dicapai berupa hasil akurasi hoaks detektor dari metode klasifikasi Rocchio tentang Covid-19 Indonesia di twitter yang berupa model machine learning terbagi menjadi 3, yaitu: Unigram, Bigram, dan Trigram

**4.1 Hasil Pengujian Skenario 1**

Pada bagian ini dilakukan analisis terhadap hasil akhir berupa model dari hasil uji unigram. Pengujian dilakukan dengan memerhatikan nilai precision, recall dan support sebagai faktor utama. Dapat dilihat bahwa hasil pengujian menunjukkan label 0 sebagai non-hoax memiliki akurasi sebanyak 0.87 yang cukup tinggi sebagai kelebihan dari klasifikasi Rocchio yaitu akurasi yang tinggi.

Tabel 9. Hasil Pengujian Skenario 1 Unigram

Label	Precision	Recall	F1-Score	Support
0 (Non-Hoax)	0.85	0.93	0.89	2315
1 (Hoax)	0.89	0.77	0.83	1699
Accuracy			0.87	4014
Macro Avg	0.87	0.85	0.86	4014
Weighted Avg	0.87	0.87	0.86	4014

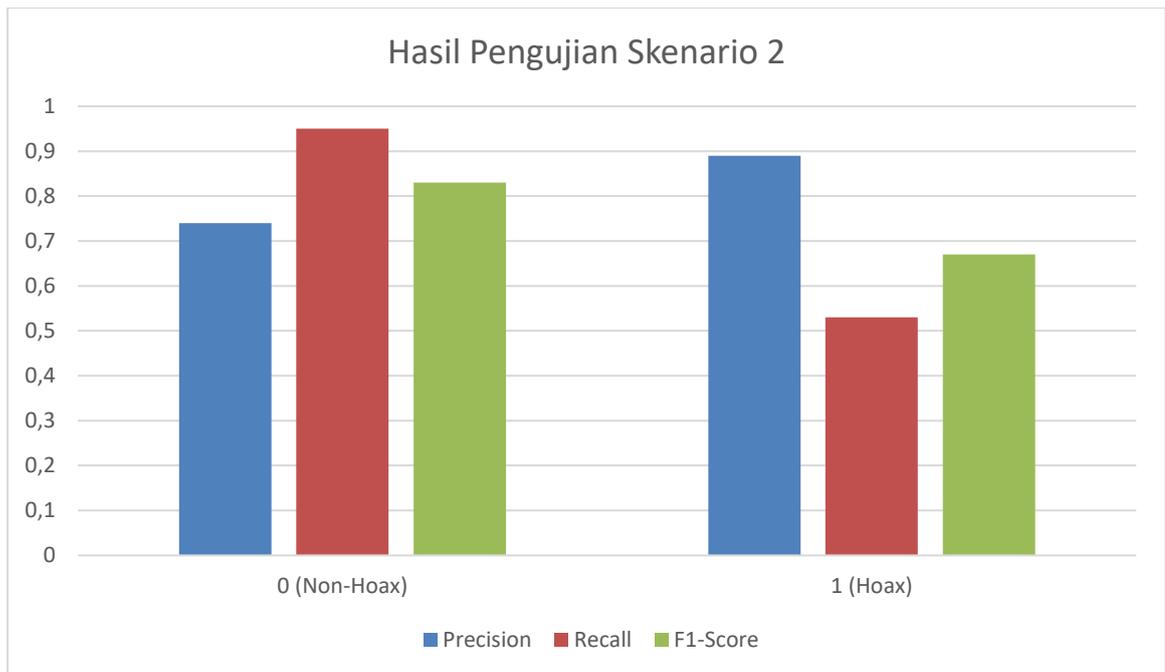


#### 4.2 Hasil Pengujian Skenario 2

Pada bagian ini dilakukan analisis terhadap hasil akhir berupa model dari hasil uji bigram. Untuk pengujian pada tipe bigram terdapat pengurangan nilai pada beberapa aspek termasuk akurasi berkurang sebanyak 0.10 point, sementara untuk hoax terdapat penurunan nilai recall dan F1-Score Pengujian terhadap bigram mendapatkan akurasi yang lebih kecil dibanding dengan unigram.

Tabel 10. Hasil Pengujian Skenario 2 Bigram

Label	Precision	Recall	F1-Score	Support
0 (Non-Hoax)	0.74	0.95	0.91	2315
1 (Hoax)	0.89	0.53	0.67	1699
Accuracy			0.77	4014
Macro Avg	0.80	0.74	0.75	4014
Weighted Avg	0.81	0.77	0.76	4014

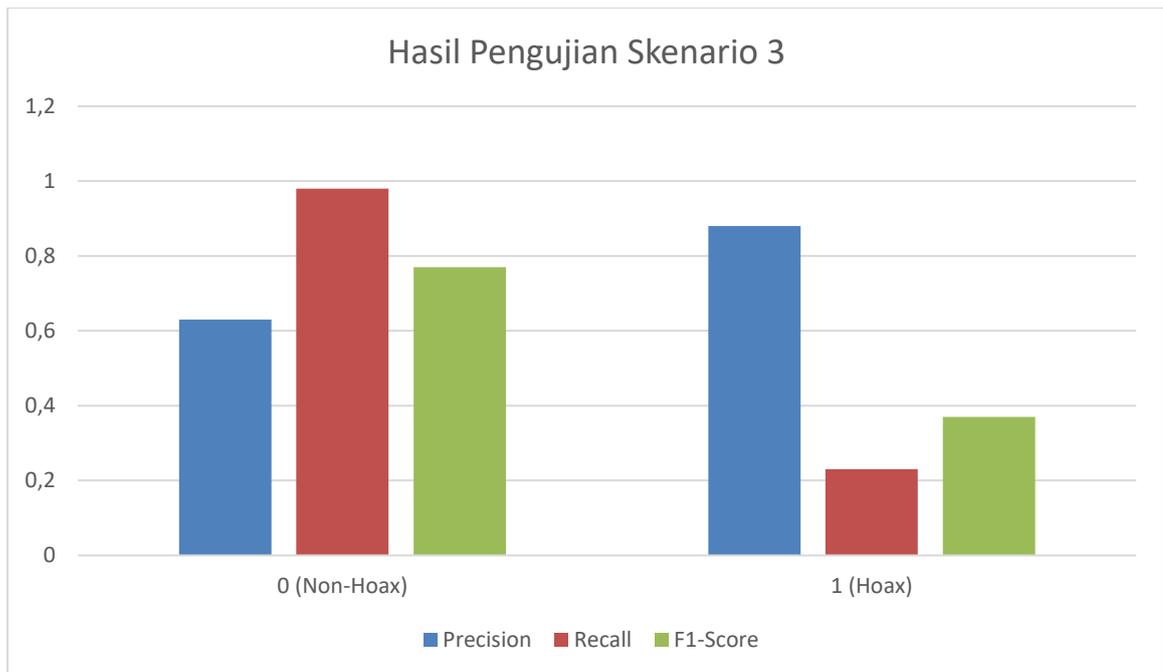


### 4.3 Hasil Pengujian Skenario 3

Pada bagian ini dilakukan analisis terhadap hasil akhir berupa model dari hasil uji trigram. Untuk pengujian pada tipe trigram nilai hasil pada label 0 cenderung menurun sementara pada label 1 bagian recall dan f-1 score mengalami penurunan lebih banyak. Untuk akurasi, semakin banyak kata N-gram dalam pengujian akan semakin kecil hasil akurasi yang dihasilkan, berdasarkan perbandingan antara unigram, bigram dan trigram. Akurasi trigram lebih kecil dari bigram 0,11 poin dan bigram lebih kecil dari unigram sebanyak 0.10 poin.

Tabel 11. Hasil Pengujian Skenario 3 Trigram

Label	Precision	Recall	F1-Score	Support
0 (Non-Hoax)	0.63	0.98	0.77	2315
1 (Hoax)	0.88	0.23	0.37	1699
Accuracy			0.66	4014
Macro Avg	0.76	0.60	0.57	4014
Weighted Avg	0.74	0.66	0.60	4014



## 5. Kesimpulan

### 5.1 Kesimpulan

Dari analisis yang dilakukan terhadap skenario pengujian yang dilakukan pada sistem hoax detector Covid-19 Indonesia menggunakan metode klasifikasi Rocchio, maka didapatkan kesimpulan sebagai berikut :

1. Pengimplementasian sistem dapat menghasilkan tingkat akurasi sebesar 0,87 untuk unigram, 0,77 untuk bigram, dan 0,66 untuk trigram. Dengan precision untuk pengambilan data lebih besar pada label 1 (hoax) sesuai dengan perbandingan pada dataset.
2. Precision, recall, f1-score, support dan accuracy diperoleh dari hasil data uji yang telah diubah menjadi sparse matriks pada bagian seleksi fitur yang sebelumnya sudah mengalami pembobotan pada tf-idf dan sudah dilakukan preprocessing.
3. Hasil akurasi dipengaruhi oleh banyaknya tipe N-gram yang dilakukan pada data uji, semakin banyak N-gram maka semakin kecil akurasi yang didapatkan.

### 5.2 Saran

Sistem hoax detector Covid-19 menggunakan klasifikasi Rocchio ini, untuk performansi kinerja sistem masih dapat lebih dikembangkan agar lebih efisien. Oleh karena itu, penulis menyarankan untuk pengembangan sistem dapat ditingkatkan dari aspek berikut :

1. Lebih memperbanyak dan menyeimbangkan dataset agar akurasi dapat lebih baik lagi
2. Pengujian sistem selanjutnya bisa diaplikasikan pada tipe data yang lain selain hoax

## REFERENSI

- [1] Asur, Sitaran dan Bernardo A. Huberman. Predicting The Future With Social Media. Proceeding WI-IAT '10 Proceeding of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. Volume 01. Halaman 492-499. 2010.
- [2] Rimjhim, & Chakraborty, R. (2018). Characterizing user reactions towards twitter's 280 character limit. ACM International Conference Proceeding Series.
- [3] Situngkir, Hokky. (2011). "Spread Of Hoax in Social Media". Bandung : Department Computational Sociology.
- [4] J. Han, M. Kamber and J. Pei, Data Mining Concepts and Techniques Third Edition, San Fransisco: Morgan Kauffman Publishers, 2012.

- [5] Marwan, M. R., & Ahyad. (2017). Analisis Penyebaran Berita Hoaks di Indonesia. Universitas Gunadarma
- [6] Mustofa, H., & Mahfudh, A. A. (2019). Klasifikasi Berita Hoax Dengan Menggunakan Metode Naive Bayes. *Walisongo Journal of Information Technology*, 1(1). <https://doi.org/10.21580/wjit.2019.1.1.3915>
- [7] Afriza, A., & Adisantoso, J. (2018). Metode Klasifikasi Rocchio untuk Analisis Hoax. *Jurnal Ilmu Komputer Dan Agri-Informatika*.
- [8] Petkovic, T Kostanj, Z dan Pale P. 2005. E-mail System for Automatic Hoax Recognition. Departement of Electronic and Information Processing Faculty of Electrical Engineering and Computing, University of Zagreb.
- [9] Vukovic, M, Pripuzic, K, dan Belani, H. 2009. An Intelligent Automatic Hoax Detection System. Knowledge-Based and Intelligent Information and Engineering Systems Lecture Notes in Computer Science, Vol 5711, 318-325.
- [10] Chen, Y Y, Yong, S P, dan Ishak A. 2014. Email Hoax Detection System Using Levenshtein Distance Method. *Journal of computers*, Vol 9. No 2, academy publisher.
- [11] Rasywir, E dan Purwarianti, A. 2015. "Eksperimen pada Sistem Klasifikasi Berita Hoax Berbahasa Indonesia Berbasis Pembelajaran Mesin". dalam: *Jurnal Cybermatika* 3 (2).
- [12] Pantouw, J C. 2017. "Perbandingan Klasifikasi Rocchio dan Multinomial Naive Bayes pada Analisis Sentimen Data Twitter Bahasa Indonesia". Skripsi. Departemen Ilmu Komputer, Institut Pertanian Bogor. 36pp.
- [13] Manning, C, Raghavan, P, dan Schutze, H. 2009. *Introduction to Information Retrieval*. Cambridge: Cambridge University Press. [Internet].
- [14] Joachims, T. 1997. "A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization" dalam: *International Conference on Machine Learning*, pp. 143- 151
- [15] T. Puspa Anggitasari, "Klasifikasi Spammer Pada Twitter Berdasarkan Perilaku Pengguna Menggunakan Algoritme C5.0," 2015.
- [16] Wahyunita, L. (2019). Reayasa Web Klasifikasi Rocchio pada Data Tidak Terstruktur. *Jurnal Komunika : Jurnal Komunikasi, Media Dan Informatika*. <https://doi.org/10.31504/komunika.v8i2.2016>
- [17] M. Allahyari, E. D. Trippe, and J. B. Gutierrez, "A Brief Survey of Text Mining: Classification , Clustering and Extraction Techniques," 2017.
- [18] I. Guyon, S. Gunn, and M. Nikravesh, "Feature Extraction," 2006.
- [19] Sierra, D. (2013). Algoritma TF — IDF . <https://medium.com/@dltsierra/algoritma-tf-idf-633e17d10a80>
- [20] L. Ladha and T. Deepa, "Feature Selection Methods and Algorithms," *Int. J. Comput. Sci. Eng.*, vol. 3, no. 5, pp. 1787–1797, 2011.
- [21] Komunitas Programmer. <https://www.payahtidur.com/project/cosine-similarity>
- [22] A. H. I. L. S. Azizah Zain, "Identifikasi Tweet Hoax yang Berhubungan dengan Pemilihan Presiden 2019 Menggunakan Naive Bayes Classifier," *Open Library Telkom*, pp. 4-14, 2019.
- [23] <https://berkas.dpr.go.id/sipinter/files/sipinter-2475-180-20210722101553.pdf>
- [24] L. S. R. Sedy Andrian Sugianto, "Pembuatan Aplikasi Predictive Text Menggunakan Metode N-Gram-Based," *Semantic Scholar*, 2018.
- [25] <https://yunusmuhammad007.medium.com/tf-idf-term-frequency-inverse-document-frequency-representasi-vector-data-text-2a4eff56cda>