

# Perbandingan Algoritma Klasifikasi SVM dan Naive Bayes Dalam Analisis Sentimen Pembelajaran Daring di Masa Pandemi COVID-19 di Twitter

1<sup>st</sup> Arif Jundi Firdausi

Fakultas Informatika

Universitas Telkom

Bandung, Indonesia

jundifirdausi@students.telkomuniversity.ac.id

2<sup>nd</sup> Widi Astuti

Fakultas Informatika

Universitas Telkom

Bandung, Indonesia

widiwdu@telkomuniversity.ac.id

3<sup>rd</sup> Adiwijaya

Fakultas Informatika

Universitas Telkom

Bandung, Indonesia

adiwijaya@telkomuniversity.ac.id

## Abstrak

Pandemi Covid-19 menciptakan banyak perubahan, salah satunya adalah sistem pendidikan di Indonesia. Yang semula sistem pendidikan yang dilaksanakan adalah pembelajaran secara tatap muka atau datang ke sekolah. Namun setelah pandemi, pembelajaran dilakukan secara daring dan dilaksanakan di rumah masing-masing. Perubahan yang mendadak serta berskala nasional ini menciptakan banyak sekali opini di masyarakat terutama opini yang berada di media sosial twitter. Pada penelitian ini bertujuan untuk menganalisis sentimen masyarakat terhadap pembelajaran daring pada masa pandemi COVID-19 di Indonesia dengan membandingkan metode klasifikasi SVM dan Naive Bayes. Dengan harapan dapat mengetahui metode mana yang paling baik performansinya. Dari penelitian ini dihasilkan metode SVM lebih baik dari pada Naive Bayes dengan nilai terbaik yang didapatkan oleh metode SVM adalah dengan presentasi 60/40 dan tanpa menggunakan sentimen netral serta menggunakan kernel RBF mendapatkan akurasi 0,72 atau 72%.

**Kata Kunci:** covid-19, daring, opini, sentimen, naïve bayes, svm

## Abstract

*The Covid-19 pandemic has created many changes, one of which is the education system in Indonesia. Initially, the education system implemented was face-to-face learning or coming to school. However, after the pandemic, learning was carried out boldly and carried out in their respective homes. The changes that came along with this national service created a lot of opinion in the community on Twitter social media. This study aims to analyze public sentiment towards courageous learning during the COVID-19 pandemic in Indonesia by comparing the SVM and Naive Bayes classifications. With the hope of knowing which method works best. From this research, the SVM method is better than Naive Bayes with the best value obtained by the SVM method is the presentation of 60/40 and without using neutral sentiment and using the RBF kernel to get an accuracy of 0.72 or 72%.  
**Keywords:** covid-19, online, opinion, sentiment, naïve bayes, svm*

## I. PENDAHULUAN

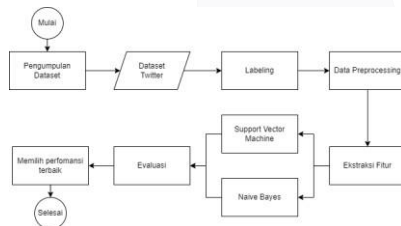
Setelah kasus COVID-19 di Indonesia menembus angka 1 juta, pemerintah terus berupaya dalam melaksanakan kebijakan Pembatasan Sosial Berskala Besar (PSBB) [1]. Melalui Kementerian Pendidikan dan Kebudayaan Pemerintah telah melarang semua jenjang pendidikan untuk melaksanakan pembelajaran tatap muka (konvensional) dan memerintahkan untuk menyelenggarakan pembelajaran secara daring (Surat Edaran Kemendikbud Dikti No. 1 tahun 2020). Perubahan pembelajaran tatap muka ke pembelajaran daring pada skala nasional menciptakan respons di masyarakat. Dari banyaknya respons tersebut penulis akan meneliti dan

menganalisis opini publik terhadap pembelajaran daring pada masa pandemi COVID-19 di twitter berbahasa Indonesia. Sentimen analisis dapat dikerjakan menggunakan berbagai macam metode pembelajaran mesin. Seperti *Support vector machine (SVM)*, *Naive Bayes (NB)*, *K Nearest neighbor (KNN)* dan *Random forest*. Merujuk pada penelitian [2] penulis menggunakan klasifikasi SVM dan NB untuk text categorization dengan menggunakan dataset yang seimbang, hasil yang ditunjukkan SVM menunjukkan keunggulan dibandingkan NB. Merujuk pada penelitian [4] penggunaan metode NB dengan Sentimen Pembelajaran Daring Pada Twitter menghasilkan presisi 97,15%. Namun pada penelitian [2] menyimpulkan bahwa SVM memberikan keunggulan dari pada NB dalam hal

klasifikasi. Tujuan dari penelitian ini adalah mengimplementasikan *SVM dan NB* pada sebuah sentimen pembelajaran daring di masa pandemi COVID-19 di Twitter serta mencari performansi terbaik dari perbandingan klasifikasi SVM dan NB dengan menggunakan dataset hasil crawling twitter berbahasa Indonesia dengan pelabelan atau pengelompokan sentimen tiga kelas, yaitu positif, netral dan negatif.

## II. KAJIAN TEORI

Beberapa hasil studi terkait yang telah dilakukan diantaranya adalah pada penelitian [2], penelitian ini melakukan perbandingan algoritma klasifikasi SVM dan NB untuk text categorization dengan menggunakan dataset yang seimbang, hasil yang ditunjukkan SVM menunjukkan kinerja lebih unggul dibandingkan NB. Dalam penelitian [3] perbandingan algoritma klasifikasi SVM dan NB berdasarkan Sentiment Analysis menggunakan Review Dataset menunjukkan hasil yang berbeda, dimana hasil recall dan f1-score yang dihasilkan NB lebih unggul dari pada SVM. Merujuk pada penelitian [4] didapatkan dengan metode NB dengan Sentimen Pembelajaran Daring Pada Twitter performansi terbaik. Dengan demikian diharapkan hasil penelitian ini dapat memberikan



Gambar 1. Flowchart Sistem

### a. Pengumpulan Dataset

Dataset yang digunakan bersumber dari twitter dengan dilakukan pengumpulan data.

Table 1. Labeling pada Tweet

Label	Tweet
Positif	support system yang dibutuhkan sekarang bukan lagi doi, tapi jaringan provider yang mendukung saat lagi jam kuliah online.
Netral	sedang berada di fase gumoh kuliah online tapi terlalu nyaman di rumah buat kuliah offline <a href="https://t.co/8yrowksdxy">https://t.co/8yrowksdxy</a>
Negatif	kuliah online gini punya rumah plosok minim sinyal sumpahhhh gaenak banget!!!udah pasang wifi ya tapi sama ajaaa g\xe2\x80\xa6 <a href="https://t.co/uq2luzmnkh">https://t.co/uq2luzmnkh</a>

Hasil dari proses labeling ini sebagai berikut:

menghasilkan nilai presisi 97,15%. Dengan hasil penelitian [4] menggunakan metode NB maka akan dilakukan penelitian lanjutan dengan membandingkan metode NB terhadap SVM. Dengan Algoritma pembobotan yang digunakan untuk proses ekstraksi fitur adalah TF-IDF karena pada penelitian [5], Penerapan TF-IDF menjadikan algoritma perhitungan lebih efektif. Selain itu penggunaan N-Gram pada ekstraksi fitur TF-IDF menjadi lebih akurat dan terhindar dari kesalahan perhitungan. Pada penelitian [6] TF-IDF pada fitur word level berkinerja sekitar 3-4% lebih baik karena mempertimbangkan masing-masing dan setiap kata sama pentingnya.

## III. METODE

Sistem yang dirancang dalam penelitian ini bertujuan untuk membandingkan algoritma klasifikasi SVM dan NB dalam analisis sentimen pembelajaran daring pada masa pandemi Covid-19. Dataset yang digunakan bersumber dari hasil pengumpulan data. Dataset dilakukan *preprocessing* dan dilakukan fitur ekstraksi. Dari hasil kedua algoritma klasifikasi tersebut dipilih yang memiliki

informasi dan referensi untuk pengembangan penelitian selanjutnya.

Proses ini menggunakan *keyword* sesuai judul penelitian seperti “daring”, “online”, “sekolah online”. serta mengambil cuitan pada waktu surat edaran Kemendikbud Dikti No. 1 tahun 2020 dikeluarkan hingga beberapa bulan selanjutnya.

### b. Proses Labeling

Labeling dilakukan secara online dengan 3 orang responden yang memberikan label di setiap tweet dengan label positif, negatif dan netral. Lalu ditentukan label tiap tweet dari mayoritas banyaknya label yang diberikan di dalam satu tweet tersebut.

**Table 2. Total Jumlah Sentimen**

	Positif	Negatif	Netral	Total
Jumlah	2110 tweet	1750 tweet	1705 tweet	5.565 tweet

c. *Data Preprocessing*

*Data Preprocessing* memainkan peran penting. Terutama untuk dapat mengoptimalkan performansi[7]. Proses ini terdiri dari *cleansing data*, *tokenization*, *stopword removal* dan *stemming*.

i. *Cleansing Data*

Data kotor akan dibersihkan dengan menghapus karakter spesial, *url*, angka, emoji dan *whitespace*. lalu kata diubah menjadi huruf kecil. Hasil dari *cleansing data* dapat dilihat dari tabel dibawah ini:

**Table 3. Contoh Hasil Cleansing Data**

Sebelum Cleansing Data	Sesudah Cleansing Data
Sistem ulangannya gmn sih? Masih online dirumah atau udh di sekolah?? Sorry tanya gw bkn generasi covid soalnya wkwk	sistem ulangannya gmn sih masih online dirumah atau udh di sekolah sorry tanya gw bkn generasi covid soalnya wkwk

ii. *Tokenization*

Data lalu dilakukan *tokenize* untuk memisahkan kalimat menjadi beberapa kata. Hasil dari *tokenize* dapat dilihat dari tabel dibawah ini:

**Table 4. Contoh Hasil Tokenization**

Sebelum Tokenisasi	Sesudah Tokenisasi
sistem ulangannya gmn sih masih online dirumah atau udh di sekolah sorry tanya gw bkn generasi covid soalnya wkwk	['sistem', 'ulangannya', 'gmn', 'sih', 'masih', 'online', 'dirumah', 'atau', 'udh', 'di', 'sekolah', 'sorry', 'tanya', 'gw', 'bkn', 'generasi', 'covid', 'soalnya', 'wkwk']

iii. *Stopword Removal*

Data lalu dilakukan *stopword removal* bahasa indonesia. Hasil dari *stopword removal* dapat dilihat dari tabel dibawah ini:

**Table 5. Contoh Hasil Stopword Removal**

Sebelum Stopword Removal	Sesudah Stopword Removal
['sistem', 'ulangannya', 'gmn', 'sih', 'masih', 'online', 'dirumah', 'atau', 'udh', 'di', 'sekolah', 'sorry', 'tanya', 'gw', 'bkn', 'generasi', 'covid', 'soalnya', 'wkwk']	['sistem', 'ulangannya', 'gmn', 'online', 'dirumah', 'udh', 'sekolah', 'sorry', 'gw', 'bkn', 'generasi', 'covid', 'wkwk']

iv. *Stemming*

Data selanjutnya dilakukan proses *stemming*. Hasil dari *stemming* dapat dilihat dari tabel dibawah ini:

**Table 6. Contoh Hasil Stemming**

Sebelum Stemming	Sesudah Stemming
['sistem', 'ulangannya', 'gmn', 'online', 'dirumah', 'udh', 'sekolah', 'sorry', 'gw', 'bkn', 'generasi', 'covid', 'wkwk']	['sistem', 'ulang', 'gmn', 'online', 'rumah', 'udh', 'sekolah', 'sorry', 'gw', 'bkn', 'generasi', 'covid', 'wkwk']

#### d. Ekstraksi Fitur

Pada penelitian ini metode ekstraksi fitur yang digunakan adalah TF-IDF yang merupakan metode untuk menghitung bobot setiap kata yang dihitung dari

seberapa seringnya suatu kata muncul dalam sebuah dokumen. Tujuannya adalah menggambarkan sebuah identitas objek yang membantu dalam penelitian [8]. dengan formula pada persamaan berikut[9] :

$$Wdt = TFdt \times IDFft$$

Keterangan:

$Wdt$  = Nilai dokumen ke-d pada kata ke-t

$TFdt$  = Jumlah kata yang dicari dalam suatu dokumen

$IDFft$  = Inverse Document Frequency ( $\log(\frac{N}{df})$ )

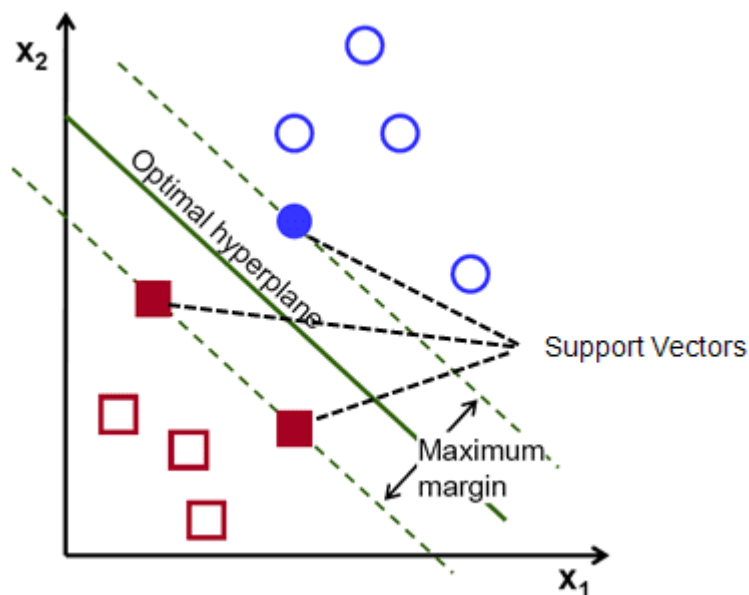
$N$  = Jumlah dokumen

$df$  = Jumlah dokumen yang mengandung kata yang dicari

#### e. Classification

##### i. Support Vector Machine

SVM adalah algoritma *supervised classifier* yang menggunakan *hyperplane* yang memisahkan antara suatu kelas dengan kelas lain. Secara konsep sederhana SVM akan mencari support vektor dan lalu mencari garis pembatas dengan margin paling maksimal[10]. yang dapat diilustrasikan sebagai berikut:



Gambar 2. Representasi Klasifikasi SVM

##### ii. Naïve Bayes

Algoritma yang memiliki ciri utama yaitu asumsi yang kuat dimana fitur dataset tidak berhubungan dengan fitur lainnya pada dataset yang sama[11]. Persamaan pemilihan kelas atau perhitungan klasifikasi dengan metode NB dapat didefinisikan dalam persamaan(3) dengan mencari hasil *Prior* di persamaan (1) dan *Conditional Probability* dalam persamaan (2)[12].

$$\hat{R}(c) = \frac{N_c}{N} \quad (1)$$

$$P(w|c) = \frac{\text{count}(w,c)+1}{\text{count}(c)+|V|} \quad (2)$$

$$P(c|d) \text{ a } P(c) P(w|c) \quad (3)$$

Keterangan :

$w$	= Kata
$c$	= Kelas (positif, netral, negatif)
$N_c$	= Dokumen yang memiliki kelas
$N$	= Total dokumen
$ V $	= Jumlah dari vocabulary di tiap dokumen
$P(c d)$	= <i>Posterior Probability</i> , probabilitas kesempatan kelas c untuk menjawab d
$P(w c)$	= <i>Conditional Probability</i> (kemungkinan), probabilitas kesempatan muncul kata w di kelas c
$P(c)$	= <i>Class Prior Probability</i> , probabilitas kesempatan muncul kelas c

Perhitungan klasifikasi dengan metode NB dengan data kontinyu digunakan rumus *Densitas Gauss* :

$$P(X_i = x_i | Y = y_i) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} \frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}$$

Keterangan

$P$	= Peluang
$X_i$	= Atribut ke-i
$x_i$	= Nilai Atribut ke-i
$Y$	= Kelas yang dicari
$y_i$	= Sub-kelas yang dicari
$\mu$	= mean, menyatakan rata-rata dari seluruh atribut
$\sigma$	= Deviasi Standar, menyatakan varian dari seluruh atribut

#### f. Evaluasi

Evaluasi performansi bertujuan mengukur nilai keberhasilan dari sebuah sistem yang dibangun[13] Penelitian ini mencari performansi terbaik dari klasifikasi SVM dan NB. Metode yang digunakan adalah *Confusion Matrix*. Metode ini dapat menemukan nilai akurasi, presisi, *recall* dan *f1-score*. Beberapa istilah yang digunakan sebagai berikut :

<i>True positive (TP)</i>	: Data positif yang terdeteksi dengan benar.
<i>True negative (TN)</i>	: Data negatif yang terdeteksi dengan benar.
<i>False positif (FP)</i>	: Data negatif yang terdeteksi sebagai data positif.
<i>False negative (FN)</i>	: Data positif yang terdeteksi sebagai data negatif.

#### 1. Akurasi

Akurasi adalah tingkat keakuratan dalam mengklasifikasikan dengan benar. dapat dirumuskan sebagai berikut:

$$\text{Akurasi} = \frac{TP + TN}{TP + FP + FN + TN}$$

#### 2. Presisi

Presisi adalah tingkat kebenaran antara informasi yang diminta dengan jawaban sistem. dapat dirumuskan sebagai berikut:

$$Presisi = \frac{TP}{TP + FP}$$

### 3. Recall

Recall adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. dapat dirumuskan sebagai berikut:

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

### 4. F1-Score

F1-Score adalah perbandingan rata-rata presisi dan recall yang sudah di bobotkan, dapat dirumuskan sebagai berikut:

$$F1 - Score = 2 \times \frac{Recall \times Presisi}{Recall + Presisi}$$

## IV. HASIL DAN PEMBAHASAN

### A. Skenario Pengujian

Pada penelitian ini evaluasi yang dilakukan pada sistem yang dibangun menggunakan beberapa skenario untuk mengetahui seberapa jauh keberhasilan dari penelitian yang dilakukan. Data yang telah diolah akan dibagi menjadi dua bagian yaitu data latih dan data test. Dengan menjalankan 4 skenario.

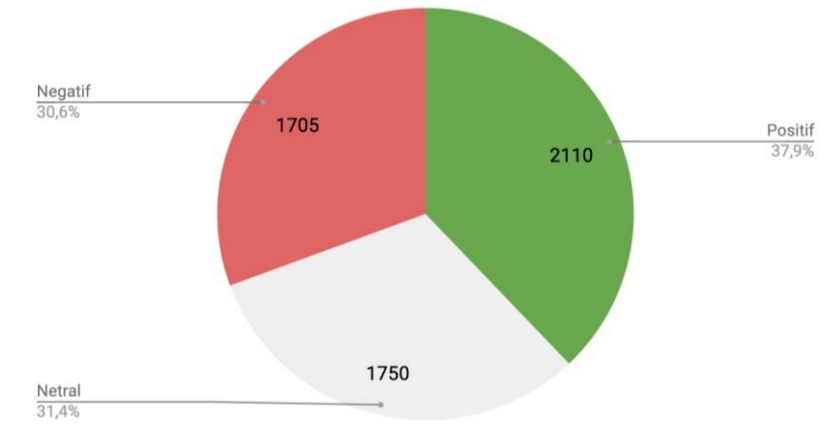
**Table 7. Skenario yang dilakukan**

Skenario	Keterangan
Skenario 1	Menggunakan sentimen netral
Skenario 2	Tidak Menggunakan sentimen netral
Skenario 3	Tidak Menggunakan sentimen netral dan NB( <i>fit_prior=False</i> )
Skenario 4	Tidak Menggunakan sentimen netral dan SVM dengan kernel trick

**Table 8. Jenis Persentase Perbandingan Data Uji dan Data Set**

Presentasi	Perbandingan Data		Jumlah Data	
	Data Latih	Data Uji	Data Latih	Data Uji
Presentasi 1	80%	20%	4452	1113
Presentasi 2	70%	30%	3895	1670
Presentasi 3	60%	40%	3339	2226
Presentasi 4	50%	50%	2782	2781

Sebelum proses klasifikasi, penulis memeriksa distribusi data untuk menghindari data yang *imbalance*, Tujuannya untuk mengoptimalkan performansi setiap metode klasifikasi yang dilakukan.



**Gambar 3. Distribusi Data Dalam Bentuk Diagram**

Dilihat bahwa persebaran data yang diolah berdasarkan kelas sentimen cukup seimbang dengan presentasi kelas negatif 31%, kelas netral 31% dan kelas positif 38%.

**B. Proses Klasifikasi**

**4.2.1 Naïve Bayes**

Berdasarkan pembagian data pada tabel 8 dengan presentasi 80/20, didapatkan hasil:

**Table 9. Hasil NB 80/20 Menggunakan Sentiment Neutral**

Menggunakan Sentiment <i>Neutral</i>												
<i>Precision</i>				<i>Recall</i>				<i>F1-Score</i>				<i>Accuracy</i>
Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	
0.45	0.73	0.45	0.54	0.00	0.31	0.93	0.41	0.00	0.42	0.57	0.33	0.44

**Table 10. Hasil NB 80/20 Tidak Menggunakan Sentiment Neutral**

Tidak Menggunakan Sentiment <i>Neutral</i>									
<i>Precision</i>			<i>Recall</i>			<i>F1-Score</i>			<i>Accuracy</i>
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.62	0.60	0.62	0.62	0.48	0.60	0.54	0.66	0.60	0.61

Dilihat dari tabel diatas, penghilangan sentimen Neutral memiliki akurasi, Precision, recall dan f1-score yang lebih tinggi dibandingkan menggunakan sentimen Neutral. Perbedaan hasil akurasi sekitar 0.17.

**Table 11. Hasil NB 80/20 Tidak Menggunakan Sentiment Neutral dan Fit Prior False**

Tidak Menggunakan Sentiment <i>Neutral</i> dan <i>Fit Prior False</i>									
<i>Precision</i>			<i>Recall</i>			<i>F1-Score</i>			<i>Accuracy</i>
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.61	0.60	0.60	0.49	0.71	0.60	0.54	0.65	0.59	0.60

Dengan presentasi data 70/30, didapatkan hasil:

**Table 12. Hasil NB 70/30 Menggunakan Sentiment Neutral**

Menggunakan Sentiment <i>Neutral</i>												
<i>Precision</i>				<i>Recall</i>				<i>F1-Score</i>				<i>Accuracy</i>
Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	
0.41	0.65	0.00	0.35	0.93	0.33	0.00	0.42	0.57	0.44	0.00	0.34	0.34

**Table 13. Hasil NB 70/30 Tidak Menggunakan Sentiment Neutral**

Tidak Menggunakan Sentiment <i>Neutral</i>									
<i>Precision</i>			<i>Recall</i>			<i>F1-Score</i>			<i>Accuracy</i>
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.61	0.60	0.60	0.49	0.71	0.60	0.54	0.65	0.59	0.60

Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.60	0.60	0.61	0.48	0.73	0.60	0.54	0.66	0.60	0.61

Dilihat dari tabel diatas, penghilangan sentimen Neutral memiliki akurasi, Precision, recall dan f1-score yang lebih tinggi dibandingkan menggunakan sentimen Neutral. Perbedaan hasil akurasi sekitar 0.27.

**Table 14. Hasil NB 70/30 Tidak Menggunakan Sentiment Neutral dan Fit Prior False**

Tidak Menggunakan Sentiment <i>Neutral</i> dan <i>Fit Prior False</i>									
<i>Precision</i>			<i>Recall</i>			<i>F1-Score</i>			<i>Accuracy</i>
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.61	0.60	0.60	0.49	0.71	0.60	0.54	0.65	0.59	0.60

Dengan presentasi data 60/40, didapatkan hasil:

**Table 15. Hasil NB 60/40 Menggunakan Sentiment Neutral**

Menggunakan Sentiment <i>Neutral</i>												
<i>Precision</i>				<i>Recall</i>				<i>F1-Score</i>				<i>Accuracy</i>
Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	
0.42	0.67	1.00	0.69	0.93	0.35	0.00	0.43	0.58	0.46	0.00	0.45	0.46

**Table 16. Hasil NB 60/40 Tidak Menggunakan Sentiment Neutral**

Tidak Menggunakan Sentiment <i>Neutral</i>									
<i>Precision</i>			<i>Recall</i>			<i>F1-Score</i>			<i>Accuracy</i>
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.64	0.61	0.62	0.50	0.73	0.62	0.56	0.66	0.61	0.62

Dilihat dari tabel diatas, penghilangan sentimen Neutral memiliki akurasi, Precision, recall dan f1-score yang lebih tinggi dibandingkan menggunakan sentimen Neutral. Perbedaan hasil akurasi sekitar 0.26.

**Table 17. Hasil NB 60/40 Tidak Menggunakan Sentiment Neutral dan Fit Prior False**

Tidak Menggunakan Sentiment <i>Neutral</i> dan <i>Fit Prior False</i>									
<i>Precision</i>			<i>Recall</i>			<i>F1-Score</i>			<i>Accuracy</i>
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.64	0.61	0.62	0.51	0.73	0.62	0.56	0.66	0.61	0.62

Dengan presentasi data 50/50, didapatkan hasil:

**Table 18. Hasil NB 50/50 Menggunakan Sentiment Neutral**

Menggunakan Sentiment <i>Neutral</i>												
<i>Precision</i>				<i>Recall</i>				<i>F1-Score</i>				<i>Accuracy</i>
Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	
0.45	0.77	0.42	0.54	0.75	0.48	0.23	0.49	0.56	0.59	0.29	0.48	0.50

**Table 19. Hasil NB 50/50 Tidak Menggunakan Sentiment Neutral**

Tidak Menggunakan Sentiment <i>Neutral</i>									
<i>Precision</i>			<i>Recall</i>			<i>F1-Score</i>			<i>Accuracy</i>
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.65	0.61	0.63	0.50	0.75	0.62	0.57	0.67	0.62	0.63

Dilihat dari tabel diatas, penghilangan sentimen Neutral memiliki akurasi, Precision, recall dan f1-score yang lebih tinggi dibandingkan menggunakan sentimen Neutral. Perbedaan hasil akurasi sekitar 0.13.



**Table 20. Hasil NB 50/50 Tidak Menggunakan Sentiment Neutral dan Fit Prior False**

Tidak Menggunakan Sentiment <i>Neutral</i> dan <i>Fit Prior False</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.47	0.50	0.48	0.47	0.50	0.48	0.47	0.50	0.48	0.48

4.2.2 Support Vector Machine

Berdasarkan pembagian data pada tabel 8 dengan presentasi 80/20, didapatkan hasil:

**Table 21. Hasil SVM 80/20 Menggunakan Sentiment Neutral**

Menggunakan Sentiment <i>Neutral</i>												
Precision				Recall				F1-Score				Accuracy
Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	
0.45	0.73	0.44	0.54	0.74	0.53	0.23	0.50	0.56	0.61	0.30	0.49	0.51

**Table 22. Hasil SVM 80/20 Tidak Menggunakan Sentiment Neutral**

Tidak Menggunakan Sentiment <i>Neutral</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.81	0.67	0.74	0.54	0.88	0.71	0.65	0.76	0.71	0.72

Dilihat dari tabel diatas, penghilangan sentimen *Neutral* memiliki akurasi, Precision, recall dan f1-score yang lebih tinggi dibandingkan menggunakan sentimen *Neutral*. Perbedaan hasil akurasi sekitar 0.21.

**Table 23. Hasil SVM 80/20 Tidak Menggunakan Sentiment Neutral dan Kernel Linear**

Tidak Menggunakan Sentiment <i>Neutral</i> dan <i>Kernel Linear</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.68	0.62	0.65	0.47	0.80	0.70	0.55	0.69	0.62	0.64

**Table 24. Hasil SVM 80/20 Tidak Menggunakan Sentiment Neutral dan Kernel Polynomial**

Tidak Menggunakan Sentiment <i>Neutral</i> dan <i>Kernel Polynomial</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.74	0.68	0.71	0.59	0.81	0.70	0.66	0.74	0.62	0.70

**Table 25. Hasil SVM 80/20 Tidak Menggunakan Sentiment Neutral dan Kernel Sigmoid**

Tidak Menggunakan Sentiment <i>Neutral</i> dan <i>Kernel Sigmoid</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.45	0.49	0.47	0.47	0.46	0.47	0.46	0.47	0.47	0.47

Dengan presentasi data 70/30, didapatkan hasil:

**Table 26. Hasil SVM 70/30 Menggunakan Sentiment Neutral**

Menggunakan Sentiment <i>Neutral</i>												
Precision				Recall				F1-Score				Accuracy
Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	
0.45	0.73	0.48	0.55	0.78	0.51	0.21	0.50	0.57	0.60	0.29	0.49	0.51

**Table 27. Hasil SVM 70/30 Tidak Menggunakan Sentiment Neutral**

Tidak Menggunakan Sentiment <i>Neutral</i>									
Precision			Recall			F1-Score			Accuracy



0.74	0.68	0.71	0.61	0.79	0.70	0.66	0.73	0.70	0.70
------	------	------	------	------	------	------	------	------	------

**Table 35. Hasil SVM 60/40 Tidak Menggunakan Sentiment Neutral dan Kernel Sigmoid**

Tidak Menggunakan Sentiment <i>Neutral</i> dan Kernel <i>Sigmoid</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.46	0.49	0.47	0.47	0.48	0.47	0.46	0.48	0.47	0.47

Dengan presentasi data 50/50, didapatkan hasil:

**Table 36. Hasil SVM 50/50 Menggunakan Sentiment Neutral**

Menggunakan Sentiment <i>Neutral</i>												
Precision				Recall				F1-Score				Accuracy
Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	Positive	<i>Neutral</i>	Negative	Avg	
0.42	0.42	0.77	0.54	0.75	0.48	0.23	0.49	0.56	0.59	0.29	0.48	0.50

**Table 37. Hasil SVM 50/50 Tidak Menggunakan Sentiment Neutral**

Tidak Menggunakan Sentiment <i>Neutral</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.77	0.62	0.72	0.57	0.84	0.70	0.65	0.75	0.70	0.71

Dilihat dari tabel diatas, penghilangan sentimen *Neutral* memiliki akurasi, Precision, recall dan f1-score yang lebih tinggi dibandingkan menggunakan sentimen *Neutral*. Perbedaan hasil akurasi sekitar 0.21.

**Table 38. Hasil SVM 50/50 Tidak Menggunakan Sentiment Neutral dan Kernel Linear**

Tidak Menggunakan Sentiment <i>Neutral</i> dan Kernel <i>Linear</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.68	0.62	0.65	0.47	0.80	0.70	0.55	0.69	0.62	0.64

**Table 39. Hasil SVM 50/50 Tidak Menggunakan Sentiment Neutral dan Kernel Polynomial**

Tidak Menggunakan Sentiment <i>Neutral</i> dan Kernel <i>Polynomial</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.73	0.68	0.70	0.61	0.78	0.69	0.66	0.72	0.69	0.70

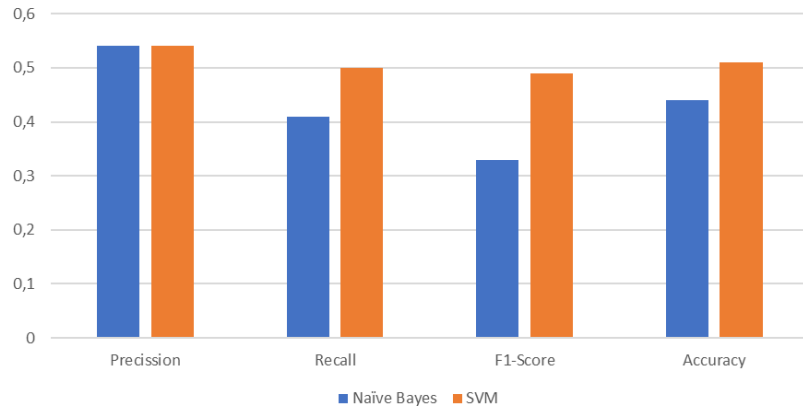
**Table 40. Hasil SVM 50/50 Tidak Menggunakan Sentiment Neutral dan Kernel Sigmoid**

Tidak Menggunakan Sentiment <i>Neutral</i> dan Kernel <i>Sigmoid</i>									
Precision			Recall			F1-Score			Accuracy
Positive	Negative	Avg	Positive	Negative	Avg	Positive	Negative	Avg	
0.47	0.50	0.48	0.47	0.50	0.48	0.47	0.50	0.47	0.48

C. Analisis Hasil Pengujian

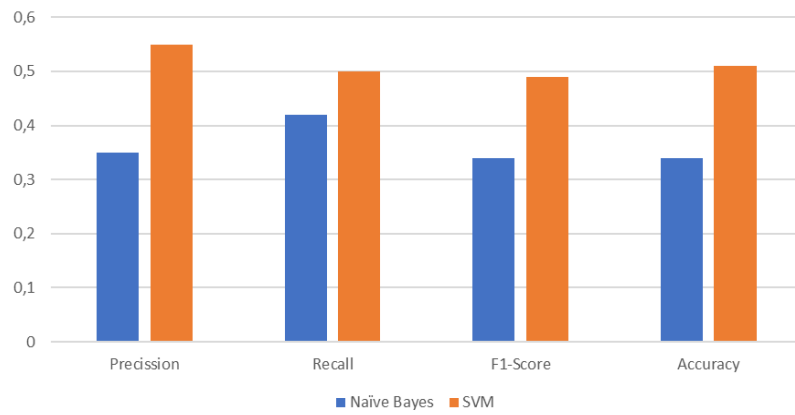
Setelah dilakukannya klasifikasi penulis akan mencari performansi yang terbaik antara algoritma SVM dan NB dengan mengambil akurasi minimal yang didapatkan dari setiap percobaan yang dilakukan di poin 4.2.

4.3.1 Analisis Hasil Pengujian Skenario 1



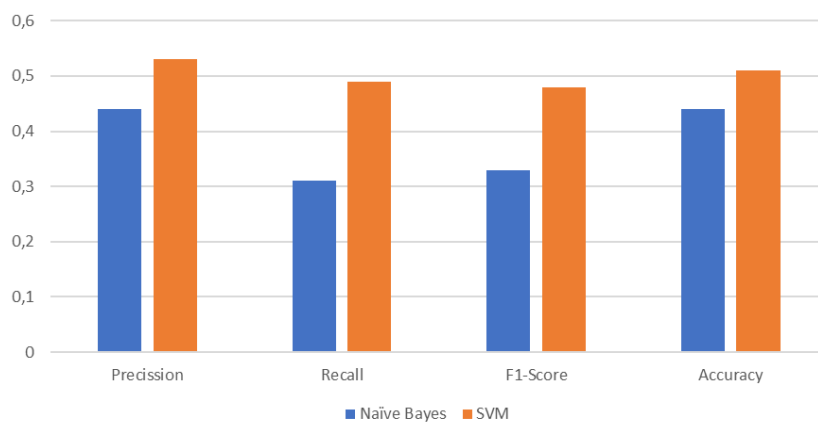
**Gambar 4. Diagram Perbandingan SVM dan NB (80/20)**

Dilihat dari diagram diatas presisi antara keduanya tidak terlampau jauh namun Recall, F1-Score dan Akurasi dari keduanya lebih baik performansi dari SVM.



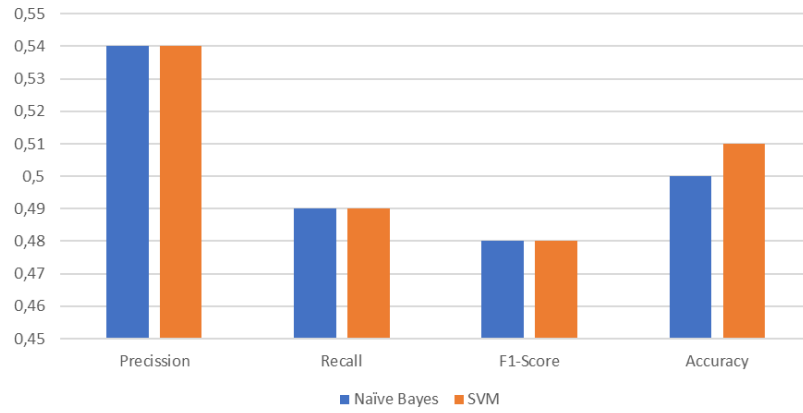
**Gambar 5. Diagram Perbandingan SVM dan NB (70/30)**

Dilihat dari diagram diatas Presisi, Recall, F1-Score dan Akurasi. SVM lebih baik.



**Gambar 6. Diagram Perbandingan SVM dan NB (60/40)**

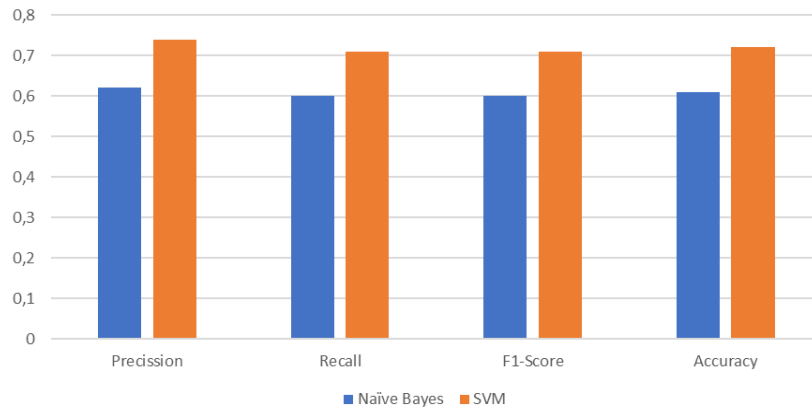
Dilihat dari diagram diatas Recall, F1-Score dan Akurasi dari SVM lebih baik.



**Gambar 7. Diagram Perbandingan SVM dan NB (50/50)**

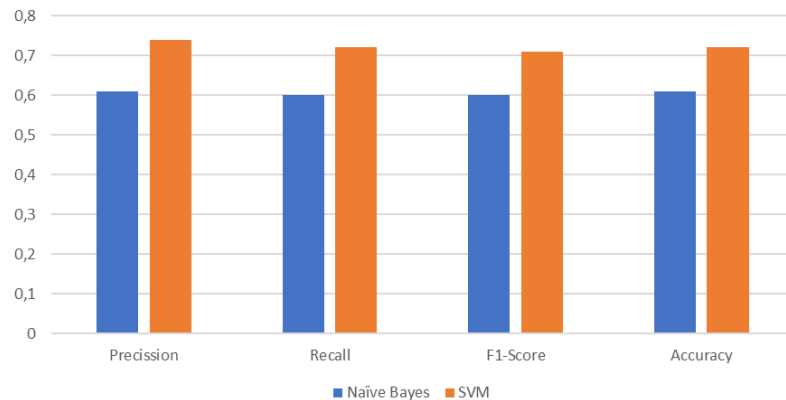
Dilihat dari diagram diatas, Recall, F1-Score dan Presisi dari keduanya hampir seimbang namun SVM lebih unggul dari sisi Akurasi.

#### 4.3.2 Analisis Hasil Pengujian Skenario 2



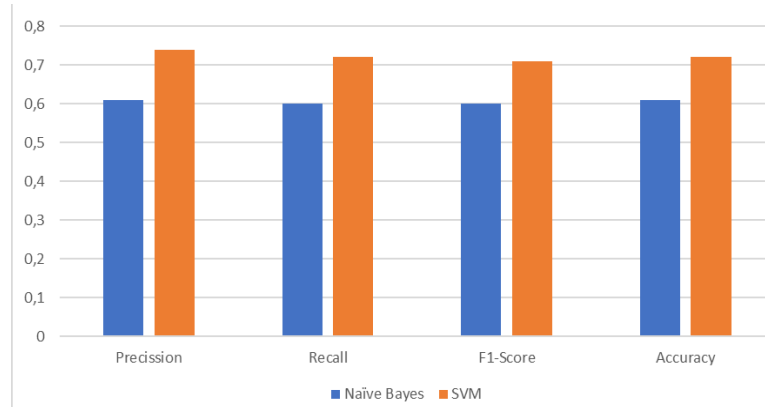
**Gambar 8. Diagram Perbandingan SVM dan NB (80/20)**

Dilihat dari diagram diatas, Recall, F1-Score, Akurasi dan Presisi jauh diungguli oleh SVM.



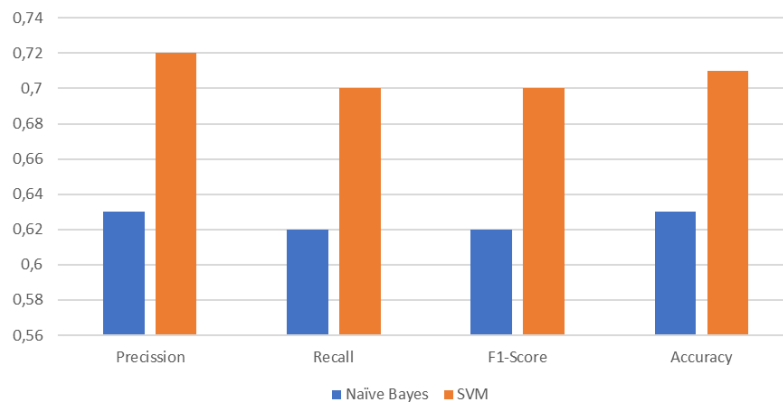
**Gambar 9. Diagram Perbandingan SVM dan NB (70/30)**

Dapat dilihat dari diagram diatas, Recall, F1-Score, Akurasi dan Presisi jauh diungguli oleh SVM.



**Gambar 10. Diagram Perbandingan SVM dan NB (60/40)**

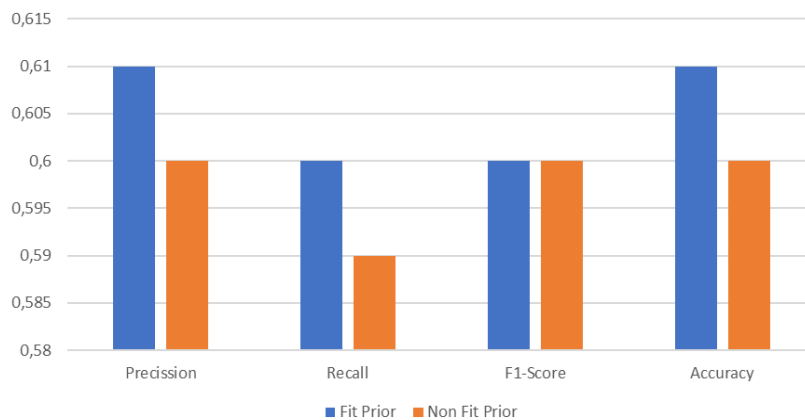
Dilihat dari diagram diatas, Recall, F1-Score, Akurasi dan Presisi jauh diungguli oleh SVM.



**Gambar 11. Diagram Perbandingan SVM dan NB (50/50)**

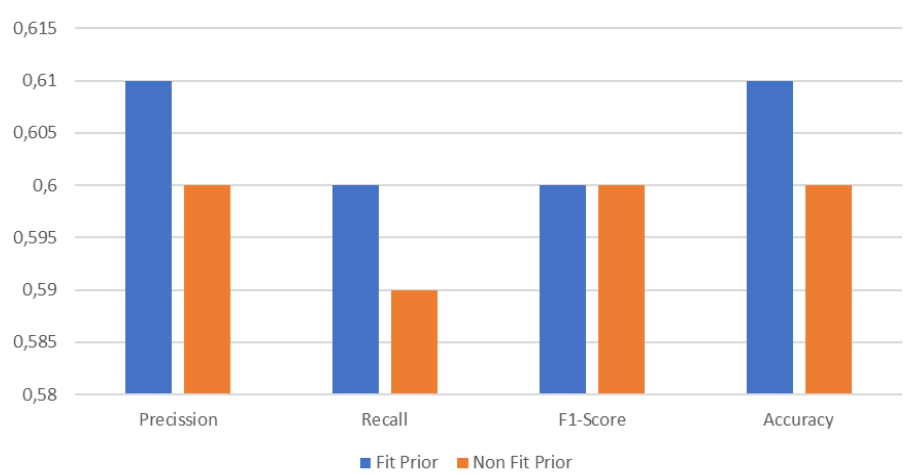
Dapat dilihat dari diagram diatas, Recall, F1-Score, Akurasi dan Presisi jauh diungguli oleh SVM.

#### 4.3.3 Analisis Hasil Pengujian Skenario 3



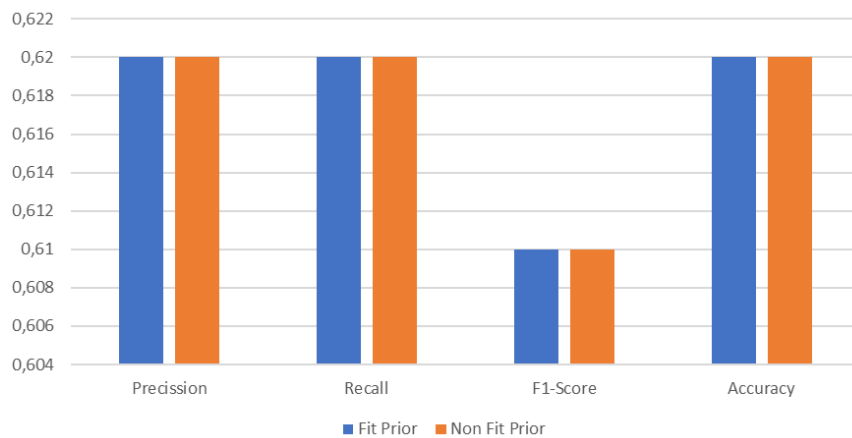
**Gambar 12. Diagram Perbandingan Fit Prior NB (70/30)**

Dilihat dari penggunaan Fit Prior dan tidak menggunakan Fit Prior, Recall, F1-Score, Akurasi dan Presisi jauh diungguli oleh penggunaan Fit Prior.



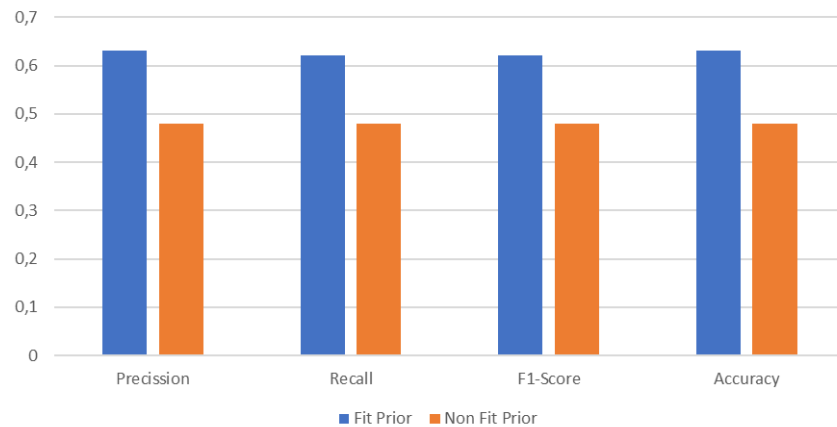
**Gambar 13. Diagram Perbandingan Fit Prior NB (70/30)**

Dilihat dari penggunaan Fit Prior dan tidak menggunakan Fit Prior, Recall, F1-Score, Akurasi dan Presisi jauh diungguli oleh menggunakan Fit Prior.



**Gambar 14. Diagram Perbandingan Fit Prior NB (60/40)**

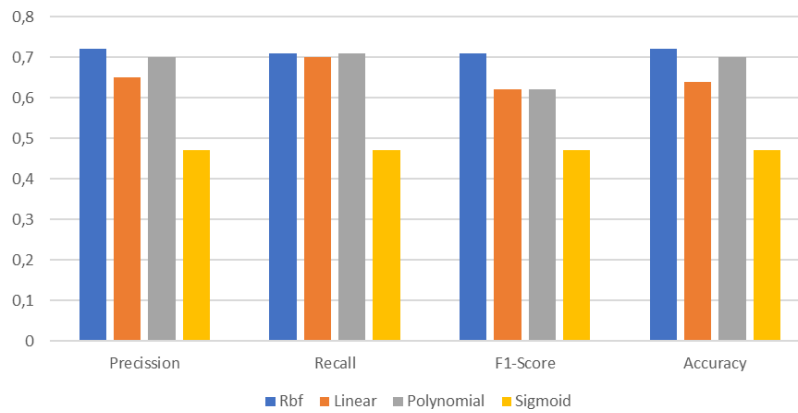
Dilihat dari penggunaan Fit Prior dan tidak menggunakan Fit Prior, Recall, F1-Score, Akurasi dan Presisi hasilnya serupa.



**Gambar 15. Diagram Perbandingan Fit Prior NB (50/50)**

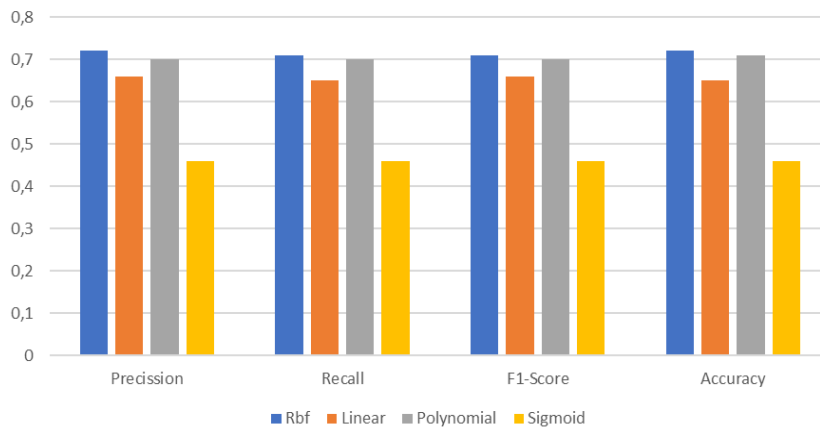
Dilihat dari penggunaan Fit Prior dan tidak menggunakan Fit Prior, Recall, F1-Score, Akurasi dan Presisi jauh diungguli oleh menggunakan Fit Prior.

4.3.4 Analisis Hasil Pengujian Skenario 4



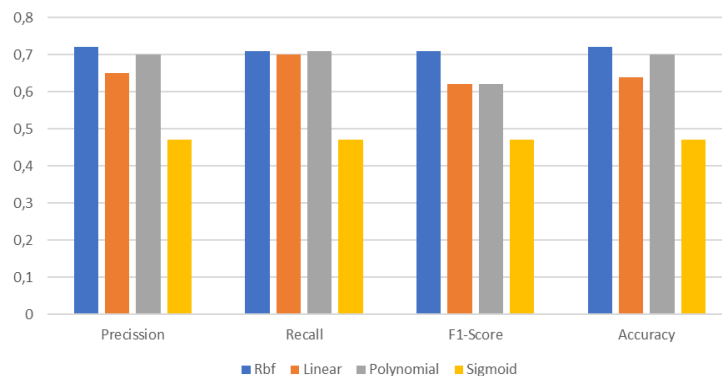
**Gambar 16. Diagram Perbandingan Kernel Trick SVM (80/20)**

Dilihat dari diagram diatas, kernel Rbf lebih unggul dari pada kernel lainnya dari sisi Recall, F1-Score, Akurasi dan Presisi.



**Gambar 17. Diagram Perbandingan Kernel Trick SVM (70/30)**

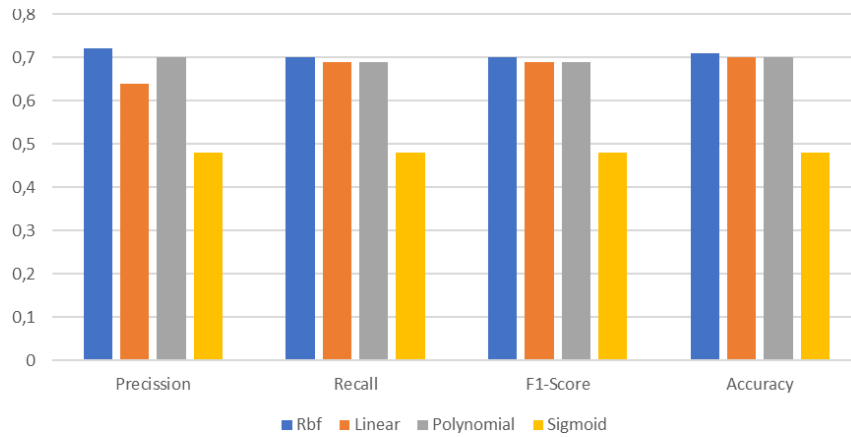
Dilihat dari diagram diatas, kernel Rbf lebih unggul dari pada kernel lainnya dari sisi Recall, F1-Score, Akurasi dan Presisi.



**Gambar 18. Diagram Perbandingan Kernel Trick SVM (60/40)**

Dilihat dari diagram diatas, dilihat kernel Rbf lebih unggul dari pada kernel lainnya dari sisi Recall, F1-Score, Akurasi dan Presisi.



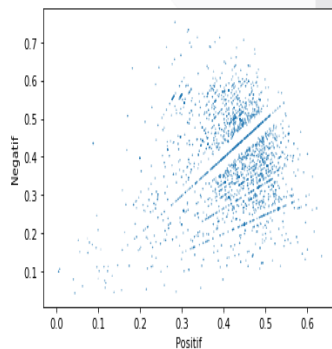


**Gambar 19. Diagram Perbandingan Kernel Trick SVM (50/50)**

Dilihat dilihat dari diagram diatas, kernel Rbf lebih unggul dari pada kernel lainnya dari sisi Recall, F1-Score, Akurasi dan Presisi.

**D. Diskusi dan Pembahasan**

Dalam diskusi ini, penulis menyoroti kinerja NB dan SVM. Ditemukan bahwa performansi SVM lebih unggul, terbukti dari skenario 1 dan skenario 2 performansi SVM selalu lebih unggul dari pada NB. Dilihat dari gambar 11 performansi SVM lebih superior dengan selisih akurasi hingga 9%, Recall 9%, F1-Score 9% dan Presisi lebih dari 10% dengan perbandingan 50/50. Keunggulan SVM dalam dikarenakan SVM hanya berfokus dalam mencari *hyperplane* dengan margin maksimal. Sedangkan NB lebih mengutamakan probabilitas kemunculan dari kata yang telah dipelajari sebelumnya (*prior*).



**Gambar 20. Persebaran Data Sebelum Terklasifikasi**

Melihat dari gambar 20, persebaran data yang akan diklasifikasi terlihat sudah cukup terpisah dimana banyak titik yang lebih condong ke sentimen positif (sumbu x) dan juga ke sentimen negatif (sumbu y). dimana ini akan mempermudah proses klasifikasi SVM dalam mencari *hyperplane* dengan margin maksimal. Sebagai contoh terpadat titik dengan nilai  $x = 0.6$  sedangkan  $y = 0.3$  maka saat SVM melakukan klasifikasi maka titik tersebut akan terklasifikasi menjadi sentimen positif (sumbu x). Oleh sebab itu, performansi antara algoritma klasifikasi SVM dan klasifikasi NB performansi SVM lebih unggul dalam klasifikasi sentimen pembelajaran daring di masa pandemi COVID-19 di twitter.

## V. KESIMPULAN

Berdasarkan hasil penelitian Perbandingan Algoritma Klasifikasi SVM dan NB Dalam Analisis Sentimen Pembelajaran Daring di Masa Pandemi COVID-19 di Twitter. dapat disimpulkan beberapa hal sesuai tujuan dari penelitian ini, diantaranya:

1. Nilai terbaik yang didapatkan oleh metode NB adalah dengan presentasi 60/40 dan tanpa menggunakan sentimen netral serta menggunakan Fit Prior mendapatkan akurasi 0,62 atau 62%.
2. Nilai terbaik yang didapatkan oleh metode SVM adalah dengan presentasi 60/40 dan tanpa menggunakan sentimen netral serta menggunakan kernel RBF mendapatkan akurasi 0,72 atau 72%.
3. Performansi terbaik antara metode NB dan SVM untuk klasifikasi sentimen pembelajaran daring di masa pandemi COVID-19 di Twitter adalah *Support Vector Machine*.

Saran untuk penelitian selanjutnya adalah dapat ditingkatkan jumlah data yang telah terlabeli sentimennya agar performansi yang dihasilkan dapat lebih baik, Penambahan proses normalisasi kata di dalam *preprocessing* sehingga data yang diolah akan lebih optimal

## REFERENSI

- [1] S. T. P. COVID-19, "Data Sebaran COVID-19 Indonesia," covid19.go.id, 2020. <https://covid19.go.id/> (accessed May. 1, 2020).
- [2] Sundus Hassan, Muhammad Rafi, Muhammad Shahid Shaikh "Comparing SVM and Naïve Bayes Classifiers for Text Categorization with Wikitology as knowledge enrichment," IEEE 14th International Multitopic Conference, 2012, doi: 10.1109/INMIC.2011.6151495.
- [3] A. M. Rahat, A. Kahir and A. K. M. Masum, "Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset," 2019 8th International Conference System Modeling and Advancement in Research Trends (SMART), 2019, pp. 266-270, doi: 10.1109/SMART46866.2019.9117512.
- [4] Samsir, Ambiyar, Unung Verawardina, Firman Edi, Ronal Watrianthos, "Analisis Sentimen Pembelajaran Daring Pada Twitter di Masa Pandemi COVID-19 Menggunakan Metode Naïve Bayes," Jurnal Media Informatika Budidarma, vol. 5, no. 1, pp. 157-163, 2021, doi: 10.30865/mib.v5i1.05.2604.
- [5] Muhammad Zidny Naf'an and Indra Hidayatulloh, "Ekstraksi TF-IDF NGRAM dari Komentar Pelanggan Produk Smartphone pada Website E- 29 commerce," Seminar Nasional Teknologi Informasi dan Multimedia, vol. 6, 2018, ISSN : 2302-3805.
- [6] Ravinder Ahujaa, Aakarsha Chuga, Shruti Kohlia, Shaurya Gupta, Pratyush Ahujaa, "The Impact of Features Extraction on the Sentiment Analysis," in International Conference on Pervasive Computing Advances and Applications – PerCAA 2019, vol. 152, pp. 341-348, 2019, doi: 10.1016/j.procs.2019.05.008.
- [7] Xiaochen Zheng , Meiqing Wang, Joaquín Ordieres-Meré, "Comparison of Data Preprocessing Approaches for Applying Deep Learning to Human Activity Recognition in the Context of Industry 4.0," in Sensors (SENSORS-BASEL), vol. 18, 2018, doi: /10.3390/s18072146
- [8] İ. İşeri, Ö. F. Atasoy and H. Alçiçek, "Sentiment classification of social media data for telecommunication companies in Turkey," 2017 International Conference on Computer Science and Engineering (UBMK), Antalya, 2017, pp. 1015-1019.
- [9] Robbi Rahim, Nuning Kurniasih, Muhammad Dedi Irawan, Yustria Handika Siregar, "Latent Semantic Indexing for Indonesian Text Similarity" International Journal of Engineering & Technology, 2018, Vol 7, pp.73-77. doi: 10.14419/ijet.v7i2.3.12619
- [10] A. Wibowo Haryanto, E. Kholid Mawardi, and Muljono, "Influence of Word Normalization and Chi-Squared Feature Selection on Support Vector Machine Text Classification," Proc. - 2018 Int. Semin. Appl. Technol. Inf. Commun. Creat. Technol. Hum. Life, iSemantic 2018, pp. 229-233, 2018, doi: 10.1109/ISEMANTIC.2018.8549748.
- [11] H. Zhang, "Exploring conditions for the optimality of naïve bayes," Int. J. Pattern Recognit. Artif. Intell., vol. 19, no. 2, pp. 183-198, 2005, doi: 10.1142/S0218001405003983.
- [12] Ainun Fauziyah Bahary, Yuliant Sibaroni, Mohamad Syahrul Mubarak, "Sentiment analysis of student responses related to information system services using Multinomial Naïve Bayes (Case study: Telkom University)," in 2nd International Conference on Data and Information Science, 2019, doi:10.1088/1742-6596/1192/1/012046.

[13] Suhariyanto, A. Firmanto and R. Sarno, "Prediction of Movie Sentiment Based on Reviews and Score on Rotten Tomatoes Using SentiWordnet," 2018 International

Seminar on Application for Technology of Information and Communication, 2018, pp. 202-206, doi: 10.1109/ISEMANTIC.2018.8549704.

