

**PERBANDINGAN ALGORITMA *NAIVE BAYES* DENGAN  
*SUPPORT VECTOR MACHINE* DALAM KLASIFIKASI  
ANALISIS SENTIMEN PADA APLIKASI TIKTOK**

**THESIS**

Oleh :  
**IKA AROFATUL HIDAYAH**  
200605210025



**PROGRAM STUDI MAGISTER INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2024**

**PERBANDINGAN ALGORITMA *NAIVE BAYES* DENGAN *SUPPORT VECTOR MACHINE* DALAM KLASIFIKASI ANALISIS SENTIMEN  
PADA APLIKASI TIKTOK**

**THESIS**

**Diajukan kepada:  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Untuk Memenuhi Salah Satu Persyaratan Dalam  
Memperoleh Gelar Magister Komputer (M.Kom)**

**Oleh :  
IKA AROFATUL HIDAYAH  
200605210025**

**PROGRAM STUDI MAGISTER INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2024**

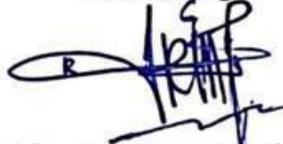
**PERBANDINGAN ALGORITMA *NAIVE BAYES* DENGAN *SUPPORT VECTOR MACHINE* DALAM KLASIFIKASI ANALISIS SENTIMEN  
PADA APLIKASI TIKTOK**

**THESIS**

**Oleh:  
IKA AROFATUL HIDAYAH  
200605210025**

Telah diperiksa dan disetujui untuk diuji:  
Tanggal: 10 Juni 2024

Pembimbing I,



Dr. Ririen Kusumawati, S.Si., M.Kom  
NIP. 19720309 200501 2 002

Pembimbing II,



Dr. M. Imamuddin, Lc., MA  
NIP. 19740602 200901 1 010

Mengetahui,  
Ketua Program Studi Magister Informatika  
Fakultas Sains dan Teknologi  
Universitas Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian  
NIP. 19740424 2009001 1 008

**PERBANDINGAN ALGORITMA *NAIVE BAYES* DENGAN *SUPPORT VECTOR MACHINE* DALAM KLASIFIKASI ANALISIS SENTIMEN  
PADA APLIKASI TIKTOK**

THESIS

Oleh:  
**IKA AROFATUL HIDAYAH**  
200605210025

Telah Dipertahankan di Depan Dewan Penguji Thesis  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Magister Komputer (M.Kom)  
Tanggal: 10 Juni 2024

**Susunan Dewan Penguji**

Penguji I

Dr. Zainal Abidin, M.Kom

NIP. 19760613 200501 1 004

Penguji II

Dr. M. Ainul Yaqin, M.Kom

NIP. 19761013 200604 1 004

Pembimbing I

Dr. Ririen Kusumawati, S.Si., M.Kom

NIP. 19720309 200501 2 002

Pembimbing II

Dr. M. Imamuddin, Lc., MA

NIP. 19740602 200901 1 010

**Tanda Tangan**



Mengetahui dan Mengesahkan  
Ketua Program Studi Magister Informatika  
Fakultas Sains dan Teknologi  
Universitas Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian  
NIP. 19740424 2009001 1 008

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Ika Arofatul Hidayah

NIM : 200605210025

Program Studi : Magister Informatika

Fakultas : Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa Thesis yang saya tulis ini benar – benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar Pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Thesis ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 4 Juni 2024  
Yang membuat pernyataan,



Ika Arofatul Hidayah  
NIM 200605210025

## KATA PENGANTAR

*Assalamu'alaikum Wr.Wb.*

Syukur alhamdulillah penulis haturkan kehadiran Allah SWT yang telah melimpahkan Rahmat dan Hidayah-Nya, sehingga penulis dapat menyelesaikan studi di Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang sekaligus menyelesaikan Thesis ini dengan baik.

Selanjutnya penulis haturkan ucapan terima kasih seiring do'a dan harapan jazakumullah ahsanal jaza' kepada semua pihak yang telah membantu terselesaikannya Thesis ini. Ucapan terima kasih ini penulis sampaikan kepada:

1. Ibu Dr. Ririen Kusumawati, S.Si., M.Kom dan Bapak Dr. M. Imamuddin, Lc., MA selaku dosen pembimbing Thesis, yang telah banyak memberikan pengarahan dan pengalaman yang berharga.
2. Segenap sivitas akademika Program Studi Magister Informatika, terutama seluruh Bapak/ Ibu dosen, terima kasih atas segenap ilmu dan bimbingan.
3. Ayahanda dan Ibunda tercinta yang senantiasa memberikan doa dan restunya kepada penulis dalam menuntut ilmu.
4. Semua pihak yang ikut membantu dalam menyelesaikan Thesis ini baik berupa materil maupun moril.

Penulis menyadari bahwa dalam penyusunan Thesis ini masih terdapat kekurangan dan penulis berharap semoga Thesis ini bisa memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi.

*Wassalamu'alaikum Wr. Wb.*

Malang, 04 Juni 2024  
Penulis,

## DAFTAR ISI

<b>HALAMAN SAMPUL .....</b>	<b>ii</b>
<b>HALAMAN PERSETUJUAN .....</b>	<b>iii</b>
<b>PERNYATAAN KEASLIAN TULISAN .....</b>	<b>v</b>
<b>KATA PENGANTAR.....</b>	<b>vi</b>
<b>DAFTAR ISI .....</b>	<b>vii</b>
<b>DAFTAR GAMBAR .....</b>	<b>x</b>
<b>DAFTAR TABEL .....</b>	<b>xii</b>
<b>ABSTRAK .....</b>	<b>xv</b>
<b>ABSTRACT .....</b>	<b>xvi</b>
<b>الملخص.....</b>	<b>xvii</b>
<b>BAB I .....</b>	<b>1</b>
<b>PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Pernyataan Masalah.....	7
1.3 Tujuan Penelitian.....	8
1.4 Batasan Masalah.....	8
1.5 Manfaat Penelitian.....	9
<b>BAB II.....</b>	<b>10</b>
<b>TINJAUAN PUSTAKA.....</b>	<b>10</b>
2.1 Analisis Sentimen.....	10
2.2 Kerangka Teori .....	27
<b>BAB III.....</b>	<b>31</b>
<b>METODOLOGI PENELITIAN .....</b>	<b>31</b>
3.1 Prosedur Penelitian .....	31
3.2 Pengumpulan Data.....	32
3.3 Pre-Processing.....	33
3.4 Lexicon.....	35
3.4.1 Pembobotan Term Frequency-Inverse Document Frequency.....	36
3.4.2 Klasifikasi Naive Bayes dan Support Vector Machine.....	36

3.4.3 Uji Coba.....	39
<b>BAB IV .....</b>	<b>42</b>
<b>PEMROSESAN DATA AWAL.....</b>	<b>42</b>
4.1 Pengumpulan Data.....	42
4.2 Case Folding dan Cleaning.....	42
4.3 Tokenisasi.....	44
4.4 Normalisasi.....	44
4.5 Stopword Removal.....	46
4.6 Stemming.....	47
4.7 Lexicon dan Labelling.....	48
4.8 Pembobotan Term Frequency-Inverse Document Frequency.....	49
<b>BAB V.....</b>	<b>51</b>
<b>KLASIFIKASI NAIVE BAYES.....</b>	<b>51</b>
5.1 Algoritma Naive Bayes.....	51
5.2 Uji Coba Algoritma Naive Bayes.....	52
5.2.1 Percobaan Naive Bayes Menggunakan Dataset 80:20.....	53
5.2.2 Percobaan Naive Bayes Menggunakan Dataset 70:30.....	54
5.2.3 Percobaan Naive Bayes Menggunakan Dataset 60:40.....	56
5.3 Performa Algoritma Naive Bayes.....	57
<b>BAB VI .....</b>	<b>59</b>
<b>KLASIFIKASI SUPPORT VECTOR MACHINE.....</b>	<b>59</b>
6.1 Algoritma Support Vector Machine.....	59
6.2 Uji Coba Algoritma Support Vector Machine.....	60
6.2.1 Percobaan Support Vector Machine Kernel Linear Menggunakan Dataset 80:20.....	61
6.2.2 Percobaan Support Vector Machine Kernel Linear Menggunakan Dataset 70:30.....	62
6.2.3 Percobaan Support Vector Machine Kernel Linear Menggunakan Dataset 60:40.....	64
6.2.4 Percobaan Support Vector Machine Kernel Polynomial Menggunakan Dataset 80:20.....	65

6.2.5 Percobaan Support Vector Machine Kernel Polynomial	
Menggunakan Dataset 70:30.....	67
6.2.6 Percobaan Support Vector Machine Kernel Polynomial	
Menggunakan Dataset 60:40.....	69
6.2.7 Percobaan Support Vector Machine Kernel Radial Basis Function	
Menggunakan Dataset 80:20.....	70
6.2.8 Percobaan Support Vector Machine Kernel Radial Basis Function	
Menggunakan Dataset 70:30.....	72
6.2.9 Percobaan Support Vector Machine Kernel Radial Basis Function	
Menggunakan Dataset 60:40.....	73
6.3 Performa Algoritma Support Vector Machine.....	75
<b>BAB VII.....</b>	<b>77</b>
<b>HASIL PEMBAHASAN.....</b>	<b>77</b>
7.1 Hasil dan Pembahasan.....	77
7.2 Perbandingan Performa Naïve Bayes dan Support Vector Machine.....	78
7.3 Integrasi Perspektif Al-Qur'an dan Hadist.....	82
<b>BAB VIII.....</b>	<b>86</b>
<b>PENUTUP.....</b>	<b>86</b>
8.1 Kesimpulan.....	86
8.2 Saran.....	86
<b>DAFTAR PUSTAKA.....</b>	<b>88</b>

## DAFTAR GAMBAR

<b>Gambar 2. 1</b> Sistem Pemodelan Pengolahan Data.....	20
<b>Gambar 2. 2</b> Implementasi Klasifikasi Teks.....	23
<b>Gambar 2. 3</b> Kerangka Teori .....	27
<b>Gambar 3. 1</b> Alur Penelitian.....	31
<b>Gambar 3. 2</b> Alur Proses Pengujian Algoritma.....	37
<b>Gambar 4. 1</b> Hasil Case Folding dan Cleaning .....	43
<b>Gambar 4. 2</b> Hasil Tokenisasi .....	44
<b>Gambar 4. 3</b> Hasil Normalisasi.....	45
<b>Gambar 4. 4</b> Hasil Stopword Removal.....	46
<b>Gambar 4. 5</b> Hasil Stemming .....	47
<b>Gambar 4. 6</b> Hasil Lexicon dan Labelling .....	48
<b>Gambar 4. 7</b> Grafik Persentasi Perbandingan Dataset.....	49
<b>Gambar 5. 1</b> Alur Klasifikasi Algoritma Naive Bayes.....	52
<b>Gambar 5. 2</b> Confusion Matrix Algoritma Naive Bayes 80:20.....	54
<b>Gambar 5. 3</b> Confusion Matrix Algoritma Naive Bayes 70:30.....	55
<b>Gambar 5. 4</b> Confusion Matrix Algoritma Naive Bayes 60:40.....	57
<b>Gambar 6. 1</b> Alur Klasifikasi Algoritma Support Vector Machine.....	60
<b>Gambar 6. 2</b> Confusion Matrix Algoritma Support Vector Machine Linear 80:20.....	62
<b>Gambar 6. 3</b> Confusion Matrix Algoritma Support Vector Machine Linear 70:30.....	63
<b>Gambar 6. 4</b> Confusion Matrix Algoritma Support Vector Machine Linear 60:40.....	65
<b>Gambar 6. 5</b> Confusion Matrix Algoritma Support Vector Machine Polynomial 80:20.....	66
<b>Gambar 6. 6</b> Confusion Matrix Algoritma Support Vector Machine Polynomial	

70:30.....	68
<b>Gambar 6. 7</b> Confusion Matrix Algoritma Support Vector Machine Polynomial 60:40. ....	69
<b>Gambar 6. 8</b> Confusion Matrix Algoritma Support Vector Machine Radial Basis Function 80:20.....	71
<b>Gambar 6. 9</b> Confusion Matrix Algoritma Support Vector Machine Radial Basis Function 70:30.....	72
<b>Gambar 6. 10</b> Confusion Matrix Algoritma Support Vector Machine Radial Basis Function 60:40.....	74
<b>Gambar 7. 1</b> Perbandingan Naïve Bayes dan Support Vector Machine.....	79
<b>Gambar 7. 2</b> Rata – rata Performa Naïve Bayes dan Support Vector Machine...	80

## DAFTAR TABEL

<b>Tabel 2. 1</b> Term Frequency.....	26
<b>Tabel 2. 2</b> Document Frequency.....	26
<b>Tabel 2. 3</b> Inverse Document Frequency.....	26
<b>Tabel 2. 4</b> Term Frequency- Inverse Document Frequency.....	27
<b>Tabel 2. 5</b> Daftar Artikel Penelitian.....	28
<b>Tabel 3. 1</b> Data Ulasan/Komentar Aplikasi TikTok.....	32
<b>Tabel 3. 2</b> Contoh Proses Pre-Processing Data.....	35
<b>Tabel 3. 3</b> Hasil Pelabelan Kelas Sentimen.....	38
<b>Tabel 3. 4</b> Confusion Matrix.....	39
<b>Tabel 3. 5</b> Split Data Training dan Testing.....	41
<b>Tabel 4. 1</b> Data Ulasan/Komentar Aplikasi TikTok.....	42
<b>Tabel 4. 2</b> Contoh Perhitungan TF- Inverse Document Frequency.....	50
<b>Tabel 5. 1</b> Split Data untuk Algoritma Naïve Bayes.....	53
<b>Tabel 5. 2</b> Pembagian Jumlah Data 80:20.....	53
<b>Tabel 5. 3</b> Classification Report Algoritma Naïve Bayes 80:20.....	54
<b>Tabel 5. 4</b> Pembagian Jumlah Data 70:30.....	55
<b>Tabel 5. 5</b> Classification Report Algoritma Naïve Bayes 70:30.....	56
<b>Tabel 5. 6</b> Pembagian Jumlah Data 60:40.....	56
<b>Tabel 5. 7</b> Classification Report Algoritma Naïve Bayes 60:40.....	57
<b>Tabel 5. 8</b> Performa Naïve Bayes.....	58
<b>Tabel 6. 1</b> Split Data untuk Algoritma Support Vector Machine.....	61
<b>Tabel 6. 2</b> Pembagian Jumlah Data Support Vector Machine Linear 80:20.....	61
<b>Tabel 6. 3</b> Classification Report Algoritma Support Vector Machine Linear 80:20.....	62

<b>Tabel 6. 4</b> Pembagian Jumlah Data Support Vector Machine Linear 70:30.....	63
<b>Tabel 6. 5</b> Classification Report Algoritma Support Vector Machine Linear 70:30.....	64
<b>Tabel 6. 6</b> Pembagian Jumlah Data Support Vector Machine Linear 60:40.....	64
<b>Tabel 6. 7</b> Classification Report Algoritma Support Vector Machine Linear 60:40.....	65
<b>Tabel 6. 8</b> Pembagian Jumlah Data Support Vector Machine Polynomial 80:20.....	66
<b>Tabel 6. 9</b> Classification Report Algoritma Support Vector Machine Polynomial 80:20.....	67
<b>Tabel 6. 10</b> Pembagian Jumlah Data Support Vector Machine Polynomial 70:30.....	67
<b>Tabel 6. 11</b> Classification Report Algoritma Support Vector Machine Polynomial 70:30.....	68
<b>Tabel 6. 12</b> Pembagian Jumlah Data Support Vector Machine Polynomial 60:40.....	69
<b>Tabel 6. 13</b> Classification Report Algoritma Support Vector Machine Polynomial 60:40.....	70
<b>Tabel 6. 14</b> Pembagian Jumlah Data Support Vector Machine Radial Basis Function 80:20.....	70
<b>Tabel 6. 15</b> Classification Report Algoritma Support Vector Machine Radial Basis Function 80:20.....	71
<b>Tabel 6. 16</b> Pembagian Jumlah Data Support Vector Machine Radial Basis Function 70:30.....	72
<b>Tabel 6. 17</b> Classification Report Algoritma Support Vector Machine Radial Basis Function 70:30.....	73
<b>Tabel 6. 18</b> Pembagian Jumlah Data Support Vector Machine Radial Basis Function 60:40.....	73
<b>Tabel 6. 19</b> Classification Report Algoritma Support Vector Machine Radial Basis Function 60:40.....	74

<b>Tabel 6. 20</b>	Performa Support Vector Machine.....	75
<b>Tabel 7. 1</b>	Tabel Perbandingan Algoritma Naïve Bayes dan Support Vector Machine.....	78
<b>Tabel 7. 2</b>	Tabel Rata - Rata Hasil Percobaan.....	80

## ABSTRAK

Arofatul Hidayah, Ika. 2024. Perbandingan Algoritma *Naïve Bayes* Dengan *Support Vector Machine* Dalam Klasifikasi Analisis Sentimen Pada Aplikasi TikTok. Thesis. Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Ririen Kusumawati, S.Si., M.Kom (II) Dr. M. Imamuddin, Lc., MA

Kata kunci: Analisis Sentimen, *Naive Bayes*, *Support Vector Machine*, TikTok

Di era digital saat ini, aplikasi media sosial seperti TikTok telah menjadi aspek penting dalam kehidupan masyarakat. TikTok memungkinkan pengguna untuk membuat dan berbagi video pendek, sehingga menjadi fenomena global dengan jutaan pengguna aktif. Namun, aplikasi ini juga telah menjadi subjek dari berbagai tanggapan dan opini dari masyarakat. Penelitian ini bertujuan untuk mengklasifikasikan sentimen masyarakat terhadap aplikasi TikTok berdasarkan komentar di Playstore dengan menggunakan algoritma *Naïve Bayes* dan *Support Vector Machine* (SVM). Metode penelitian ini dilakukan dengan mengumpulkan data komentar dari Playstore dengan menggunakan teknik scraping, sehingga menghasilkan 5.000 data ulasan. Tahapan pra-pemrosesan data meliputi case folding, tokenisasi, normalisasi, penghilangan stopword, stemming, dan pelabelan data menggunakan leksikon. Data yang telah diolah kemudian dibobot menggunakan Term Frequency - Inverse Document Frequency (TF-IDF) sebelum diklasifikasikan menggunakan algoritma *Naïve Bayes* dan SVM. Evaluasi kinerja algoritma dilakukan dengan menggunakan Confusion Matrix untuk mengukur akurasi, presisi dan recall. Hasil penelitian menunjukkan bahwa algoritma SVM memiliki akurasi yang lebih tinggi (84%) dibandingkan dengan *Naïve Bayes* (79%). SVM juga menunjukkan nilai precision dan recall yang lebih baik dalam mengklasifikasikan sentimen positif dan negatif dari ulasan pengguna. Dari hasil pengujian yang telah dilakukan, algoritma SVM lebih efektif dibandingkan dengan *Naïve Bayes* dalam analisis sentimen pada aplikasi TikTok. Penelitian ini memberikan wawasan tentang bagaimana sentimen publik dapat diukur dan dianalisis, serta menggarisbawahi pentingnya memilih algoritma yang tepat untuk analisis sentimen data pada platform media sosial.

## ABSTRACT

Arofatul Hidayah, Ika. 2024. Comparison Of Naive Bayes Algorithm With Support Vector Machine In Classification Of Sentiment Analysis On The Tiktok Application. Thesis. Master of Informatics Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisors: (I) Dr. Ririen Kusumawati, S.Si., M.Kom (II) Dr. M. Imamuddin, Lc., MA

Keywords: Sentiment Analysis, *Naïve Bayes*, *Support Vector Machine*, TikTok

In the current digital era, social media applications such as TikTok have become an important aspect of people's lives. TikTok allows users to create and share short videos, making it a global phenomenon with millions of active users. However, this application has also been the subject of various responses and opinions from the public. This research aims to classify public sentiment towards the TikTok application based on comments on Playstore using the Naïve Bayes algorithm and Support Vector Machine (SVM). This research method involves collecting comment data from Playstore using scraping techniques, resulting in 5,000 review data. Data pre-processing stages include case folding, tokenization, normalization, stopword removal, stemming, and data labeling using a lexicon. The data that has been processed is then weighted using Term Frequency - Inverse Document Frequency (TF-IDF) before being classified using the Naïve Bayes and SVM algorithms. Algorithm performance evaluation is carried out using the Confusion Matrix to measure accuracy, precision and recall. The research results show that the SVM algorithm has higher accuracy (84%) compared to Naïve Bayes (79%). SVM also shows better precision and recall values in classifying positive and negative sentiment from user reviews. From the results of the tests that have been carried out, the SVM algorithm is more effective than Naïve Bayes in sentiment analysis of the TikTok application. This research provides insight into how public sentiment can be measured and analyzed, and underscores the importance of choosing the right algorithm for data sentiment analysis on social media platforms.

## الملخص

عرفة الهداية، إيكّا. 2024. مقارنة خوارزمية المصنف البايزي الساذج مع آلة المتجه الداعم في تصنيف تحليل المشاعر على تطبيق تيك توك. رسالة الماجستير. قسم المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: د. ريرين كوسوماواي، الماجستير. المشرف الثاني: د. محمد إمام الدين، الماجستير.

**الكلمات الرئيسية:** تحليل المشاعر، بايزي ساذج، آلة المتجه الداعم، تيك توك.

في العصر الرقمي اليوم، أصبحت تطبيقات الوسائط الاجتماعية مثل تيك توك جانبا مهما من حياة الناس. يتيح تيك توك للمستخدمين إنشاء مقاطع فيديو قصيرة ومشاركتها، مما يجعلها ظاهرة عالمية مع ملايين المستخدمين النشطين. ومع ذلك، فقد كان هذا التطبيق أيضا موضوع ردود وآراء مختلفة من الجمهور. تهدف هذه الرسالة إلى تصنيف المشاعر العامة تجاه تطبيق تيك توك بناء على التعليقات على متجر جوجل (*Google Playstore*) باستخدام خوارزمية المصنف البايزي الساذج وآلة المتجه الداعم. تم تنفيذ طريقة البحث هذه من خلال جمع بيانات التعليقات من متجر جوجل باستخدام تقنية الكشط، مما أدى إلى 5000 بيانات مراجعة. تتضمن مراحل المعالجة المسبقة للبيانات طي الحالة، والتمييز، والتطبيع، وإزالة الكلمات التوقفية، والوقف، ووضع العلامات على البيانات باستخدام المعاجم. ثم تم ترجيح البيانات المعالجة باستخدام معامل تردد المصطلح - معكوس تردد الوثيقة (TF-IDF) قبل تصنيفها باستخدام خوارزمية المصنف البايزي الساذج وآلة المتجه الداعم. تم إجراء تقييم أداء الخوارزمية باستخدام مصفوفة الارتباك لقياس الدقة والثبات والاستدعاء. أظهرت النتائج أن خوارزمية المصنف البايزي الساذج كانت ذات دقة أعلى (82%) مقارنة بالمصنف البايزي الساذج (79%). أظهرت آلة المتجه الداعم أيضا دقة أفضل وقيمة الاستدعاء في تصنيف المشاعر الإيجابية والسلبية من مراجعات المستخدمين. من نتائج الاختبارات التي تم إجراؤها، تعد خوارزمية آلة المتجه الداعم أكثر فاعلية من المصنف البايزي الساذج في تحليل المشاعر على تطبيق تيك توك. قدمت هذه الرسالة نظرة ثاقبة حول كيفية قياس المشاعر العامة وتحليلها، وتؤكد على أهمية اختيار الخوارزمية المناسبة لتحليل مشاعر البيانات على منصات التواصل الاجتماعي.

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dalam era digital saat ini, aplikasi media sosial menjadi salah satu aspek penting dalam kehidupan masyarakat. Salah satu aplikasi media sosial yang tengah populer di Indonesia adalah TikTok. TikTok adalah *platform* media sosial yang memungkinkan pengguna untuk membuat dan membagikan video pendek dengan durasi singkat, biasanya berkisar antara 15 hingga 60 detik (Indriyani, Fauzi, & Faisal, 2023). Aplikasi ini telah menjadi fenomena global dengan jutaan pengguna aktif di seluruh dunia. Pengguna TikTok dapat membuat berbagai jenis konten, mulai dari tarian, *lip-sync*, komedi, tutorial, konten pendidikan, hingga tren terkini. TikTok menawarkan berbagai fitur kreatif dan alat editing yang memungkinkan pengguna untuk membuat video yang menarik dan menghibur dengan mudah (Isnan, Elwirehardja, & Pardamean, 2023). Namun, seperti halnya *platform* media sosial lainnya, TikTok juga menjadi subjek berbagai tanggapan dan opini dari masyarakat.

Analisis sentimen merupakan teknik *Natural Language Processing* yang digunakan untuk menentukan nada emosional di balik sebuah teks. Ini adalah alat penting yang dapat membantu memahami opini umum, emosi, dan respons terhadap suatu produk, layanan, ataupun ide (Fide, Suparti, & Sudarno, 2021). Analisis ini sering digunakan untuk memeriksa ulasan, umpan balik, dan media sosial untuk mendapatkan wawasan tentang sikap publik. Dengan pertumbuhan data yang tidak terstruktur di internet, seperti komentar dan postingan di media

sosial, analisis sentimen telah menjadi kunci untuk strategi bisnis dan penelitian sosial, memungkinkan organisasi untuk merespon secara lebih efektif terhadap kebutuhan dan keinginan pelanggan.

Teknologi ini menggunakan algoritma untuk mengkategorikan teks menjadi positif, negatif, atau netral. Selain itu, nuansa budaya dan demografis juga mempengaruhi cara ekspresi emosi dilakukan, yang menuntut model yang adaptif dan inklusif untuk menganalisis data dari berbagai sumber dan latar belakang dengan tepat. Pendekatan ini harus mampu membedakan antara komentar yang benar-benar mengungkapkan sentimen negatif dan mereka yang hanya menggunakan bahasa yang secara tradisional dianggap sebagai negatif dalam konteks yang humoris atau sarat ironi.

TikTok telah menjadi fenomena global yang merevolusi penggunaan media sosial, menjadikannya topik penelitian yang sangat menarik. Sebagai platform yang menarik jutaan pengguna aktif, TikTok memperkenalkan format video singkat yang inovatif, yang telah mengubah cara konten dibuat dan dikonsumsi. Interaksi pengguna dengan platform ini mencerminkan pergeseran dalam konsumsi media digital, khususnya di kalangan generasi muda, yang lebih memilih konten yang cepat, mudah diakses, dan sangat interaktif. TikTok tidak hanya menjadi platform hiburan tetapi juga ruang untuk ekspresi politik, sosial, dan kultural yang memungkinkan pengguna untuk berbagi dan mempengaruhi opini secara global.

Analisis sentimen terhadap TikTok dapat memberikan wawasan tentang bagaimana nilai-nilai dan norma sosial dibentuk dan direfleksikan melalui media digital modern. Studi tentang sentimen masyarakat terhadap TikTok juga bisa

mengungkapkan bagaimana aplikasi ini mempengaruhi isu-isu seperti identitas generasi, interaksi antar budaya, dan dinamika globalisasi. Melalui TikTok, generasi muda terutama berkesempatan untuk mengekspresikan dan mempromosikan isu-isu yang penting bagi mereka, seperti kesadaran lingkungan, hak-hak sosial, dan keadilan. Ini menunjukkan potensi TikTok sebagai alat pemberdayaan dan perubahan sosial.

Karakteristik komentar pada aplikasi TikTok di Playstore sangat menarik untuk diteliti karena mencerminkan persepsi pengguna yang beragam terhadap aplikasi tersebut. Komentar ini meliputi berbagai aspek seperti kepuasan pengguna terhadap fitur baru, keluhan tentang bug atau masalah kinerja, serta tanggapan terhadap kebijakan privasi dan keamanan aplikasi. Analisis terhadap komentar ini dapat memberikan *insight* tentang bagaimana perubahan atau pembaruan pada aplikasi mempengaruhi sentimen pengguna. Selain itu, komentar juga sering mencakup aspek sosial dan budaya, memberikan gambaran tentang bagaimana TikTok digunakan dan diterima di berbagai lapisan masyarakat.

Aplikasi TikTok dapat di unduh dan diakses melalui *Google Play* atau yang sering dikenal dengan *PlayStore*. Selain itu, pengguna *smartphone Android* juga dapat memberikan penilaian dalam bentuk rating dan ulasan terhadap aplikasi yang terdapat di *PlayStore*, termasuk salah satunya adalah aplikasi TikTok. Oleh sebab itu, penelitian ini bertujuan untuk melihat opini publik atau masyarakat terhadap aplikasi TikTok berdasarkan ulasan atau komentar yang diberikan. Ulasan pengguna atau masyarakat biasanya dapat berupa ulasan positif ataupun ulasan negatif, sehingga dari ulasan yang diberikan terhadap

aplikasi yang ada di *PlayStore* dapat digunakan sebagai data utama untuk mengetahui sentimen publik atau masyarakat terhadap aplikasi yang diberikan ulasan.

Dalam melakukan analisis sentimen, sangat penting untuk berhati-hati karena proses ini sensitif dan berkaitan erat dengan pelabelan karakter seseorang yang berpotensi menimbulkan stigma negatif. Kebebasan berpendapat dalam Islam adalah suatu hal yang diakui dan dihormati yang mana dijelaskan dalam Al-Qur'an surat Al-Kahfi ayat 29 Allah berfirman:

وَقُلِ الْحَقُّ مِنْ رَبِّكَ فَمَنْ شَاءَ فَلْيُؤْمِنْ وَمَنْ شَاءَ فَلْيُكْفُرْ

Artinya: "Dan katakanlah: Kebenaran itu datangnya dari Tuhanmu; maka barangsiapa yang ingin beriman, hendaklah ia beriman, dan barangsiapa yang ingin kafir, biarlah ia kafir."

Tafsir surat Al-Kahfi ayat 29 : "Menegaskan bahwa manusia memiliki kebebasan untuk memilih keyakinan dan pendapatnya sendiri. Pentingnya menyampaikan kebenaran yang datang dari Allah dan memberikan kebebasan kepada manusia untuk memilih jalan mereka. Ayat ini juga mengajarkan tentang tanggung jawab individu atas pilihan mereka dan konsekuensi yang akan mereka hadapi di akhirat".

Dalam pemanfaatan big data untuk analisis sentimen, penting untuk berhati-hati agar tidak terjadi prasangka buruk saat menentukan label komentar. Proses ini membutuhkan keakuratan dan kecepatan, terutama karena data yang besar dan tidak terstruktur serta perubahan data yang cepat. Oleh karena itu, diperlukan mesin dengan kinerja tinggi yang dapat melakukan proses secara otomatis dan berkala, seperti teknologi *machine learning*, untuk mengurangi intervensi manusia dan meningkatkan efisiensi. Dalam kaitannya dengan penerapan klasifikasi *teks* untuk analisis sentimen menggunakan teknologi

*machine learning*, telah dilakukan sejumlah penelitian terdahulu yang membahas tentang membandingkan berbagai metode klasifikasi dalam analisis sentimen untuk menemukan model terbaik dan menghasilkan analisis yang akurat. Memilih dan menerapkan metode klasifikasi teks untuk analisis sentimen sangat penting untuk mencapai ketelitian dan kehati-hatian dalam pemrosesan. Hal ini bertujuan untuk menghindari prasangka buruk dan mendapatkan pemahaman yang mendalam dari data yang benar-benar berguna dan dapat dimanfaatkan.

Seperti penelitian yang dilakukan Sola Fide dkk (2021) menggunakan algoritma *Support Vector Machine* dengan *kernel Radial Basis Function* (RBF) dan metode asosiasi untuk analisis sentimen ulasan aplikasi TikTok di *Google Play Store* yang memperoleh hasil akurasi 90.62% dan kappa 81.24% dan penelitian yang dilakukan oleh Friska Aditia Indriyani dkk (2023) menggunakan algoritma *naïve bayes* dan *support vector machine* untuk analisis sentimen tiktok, hasil akhir memperoleh nilai akurasi sebesar 79% untuk metode *naïve bayes* dan sebesar 84% untuk *support vector machine*.

Dari sini dapat diketahui bahwa algoritma *naïve bayes* dan *support vector machine* mempunyai performa yang bagus dalam proses klasifikasi sentimen masyarakat, selain itu proses klasifikasi sentimen juga dipengaruhi oleh data dan *preprocessing* data yang digunakan. Pada penelitian ini, tahapan *preprocessing* yang akan digunakan yakni *Case Folding*, *Cleaning*, *Tokenization*, *Stop Word Removal*, *Normalization*, dan *Stemming*. Selain itu penelitian ini juga bertujuan untuk membandingkan hasil akurasi yang diperoleh algoritma *naïve bayes* dan

*support vector machine* dalam proses klasifikasi sentimen masyarakat terhadap aplikasi TikTok.

Untuk memperoleh data ulasan di *PlayStore* tentunya sangat sulit dan membutuhkan waktu yang lama jika dilakukan *copy & paste* setiap ulasan / komentar yang ada, mengingat jumlah ulasan / komentar aplikasi yang tentunya sangat banyak sekali. Maka dari itu dalam pengumpulan data awal akan dilakukan teknik *scrapping* (penambangan teks) data. Teknik *scraping*, atau yang sering disebut juga *web scraping* merupakan proses pengambilan informasi atau data dari situs web secara otomatis menggunakan bot atau program komputer. Teknik ini memungkinkan pengguna untuk mengekstrak data yang terstruktur atau tidak terstruktur dari berbagai situs web dan menyimpannya dalam format yang dapat diakses dan dianalisis lebih lanjut.

Klasifikasi sentimen melibatkan analisis teks yang seringkali memiliki dimensi tinggi dan fitur yang jarang. *Naive Bayes* dan *Support Vector Machine* memiliki cara berbeda dalam menangani fitur tersebut, sehingga perbandingan mereka dapat menunjukkan metode yang lebih efektif untuk dataset tertentu. *Naive Bayes* biasanya cepat dan efisien dalam hal komputasi (Fitri, Andreswari, & Hasibuan, 2019; Sudhir & Suresh, 2021), sementara *Support Vector Machine* bisa lebih akurat namun memerlukan lebih banyak sumber daya (Fide, Suparti, & Sudarno, 2021; Irfani, Triyanto, Hartanto, & Kusnawi, 2020). Membandingkan keduanya dapat mengungkap akurasi yang terbaik. *Naive Bayes* menggunakan pendekatan probabilistik (Farhana, 2021; Rahab, Zitouni, & Djoudi, 2021), sementara *Support Vector Machine* adalah algoritma margin-based yang mencari hyperplane optimal untuk memisahkan kelas. Perbedaan pendekatan ini dapat

mempengaruhi kinerja mereka pada dataset sentimen. *Support Vector Machine* memiliki beberapa *kernel* yang dapat menangani overfitting lebih baik dalam beberapa kasus dibandingkan *Naive Bayes*, yang mengasumsikan independensi antar fitur.

Berdasarkan uraian diatas, penelitian ini mencoba melakukan klasifikasi sentimen masyarakat terhadap aplikasi TikTok berdasarkan komentar yang terdapat di *Playstore* menggunakan algoritma *naïve bayes* dan *support vector machine*. Hasil penelitian ini diharapkan dapat mengetahui kinerja algoritma *naïve bayes* dan *support vector machine* dalam mengklasifikasikan positif dan negatif serta menghasilkan perbandingan nilai akurasi, presisi dan recall dari data tersebut.

## **1.2 Pernyataan Masalah**

Dalam pengolahan dan pemanfaatan *big data* berbentuk teks dari media sosial, penggunaan teknik data mining dengan *machine learning* untuk analisis sentimen seringkali menemui kendala, khususnya pada tahap *preprocessing* data. Mengingat pentingnya analisis sentimen dan tantangan yang ada, penelitian ini membahas bagaimana mengukur kinerja algoritma *Naïve Bayes* dan *Support Vector Machine* untuk klasifikasi analisis sentimen pada aplikasi TikTok?

### 1.3 Tujuan Penelitian

Berdasarkan penjelasan latar belakang serta pernyataan masalah di atas adalah untuk membandingkan performa atau kinerja algoritma *Naïve Bayes* dan *Support Vector Machine* untuk klasifikasi analisis sentimen pada aplikasi TikTok.

### 1.4 Batasan Masalah

Batasan dalam penelitian ini, terdapat pada beberapa aspek sebagai berikut:

- a. Penelitian ini hanya akan memfokuskan pada analisis sentimen ulasan atau komentar pengguna di *Google Play Store* terhadap aplikasi TikTok.
- b. Data yang digunakan dalam penelitian ini diambil dari ulasan atau komentar yang terdapat di *Google Play Store* saja.
- c. Metode pengumpulan data awal menggunakan teknik scraping untuk menghindari kerumitan dalam pengumpulan data yang manual.
- d. Penelitian ini menggunakan algoritma *Naïve Bayes* dan *Support Vector Machine* untuk mengklasifikasikan sentimen ulasan aplikasi TikTok.
- e. Tahap *pre-processing data* meliputi *Case Folding*, *Cleaning*, *Tokenization*, *Stop Word Removal*, *Normalization*, dan *Stemming*.

## 1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

- a. **Memahami Sentimen Masyarakat:** Penelitian ini akan memberikan pemahaman yang lebih baik tentang bagaimana masyarakat Indonesia merespons dan mempersepsikan aplikasi TikTok.
- b. **Rekomendasi dan Pemecahan Masalah:** Hasil dari penelitian ini dapat digunakan sebagai dasar untuk memberikan rekomendasi kepada pihak terkait, termasuk pengembang aplikasi, regulator, dan pengguna, dalam meningkatkan pengalaman pengguna, memperbaiki layanan, serta mengelola dampak sosial dari penggunaan TikTok di Indonesia.
- c. **Kontribusi Ilmiah:** Penelitian ini diharapkan dapat memberikan kontribusi pada literatur tentang analisis sentimen dalam konteks media sosial di Indonesia. Informasi dan wawasan yang dihasilkan dapat menjadi sumber referensi bagi peneliti dan akademisi yang tertarik pada topik terkait.
- d. **Pemahaman Lebih Lanjut tentang Dampak Sosial:** Penelitian ini dapat memberikan pemahaman yang lebih mendalam tentang dampak sosial dari penggunaan TikTok di Indonesia, termasuk pengaruhnya terhadap pola perilaku, budaya digital, dan hubungan sosial antar individu. Hal ini dapat menjadi landasan untuk pengembangan kebijakan yang lebih baik dalam mengelola dampak sosial dari aplikasi media sosial.

## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Analisis Sentimen

Terdapat beberapa penelitian terdahulu yang telah meneliti tentang analisis sentimen, diantaranya:

Sola Fide dkk (2021) melakukan penelitian tentang analisis sentimen ulasan TikTok di *Google PlayStore* menggunakan algoritma *Support Vector Machine* dengan *kernel Radial Basis Function* dan metode asosiasi. Penelitian tersebut menganalisis sentimen publik *positif atau negatif* terhadap aplikasi TikTok di Indonesia. Beberapa proses yang dilakukan dalam analisis sentimen tersebut meliputi pengambilan data dengan proses *scrapping data* di *Google Play*, *preprocessing data* (*case folding, cleaning, dan normalisasi kata*), penilaian sentimen, *feature selection* (*stopwords, stemming, dan tokenizing*), pembobotan *Term Frequency – Inverse Document Frequency*, pembuatan data latih serta data uji, dan klasifikasi data menggunakan *Support Vector Machine* dan algoritma *asosiasi*. Hasil pengujian yang telah dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi 90.62% dan kappa 81.24%, hal ini menunjukkan klasifikasi yang hampir sempurna. Dalam penelitian tersebut, sentimen positif umumnya berkaitan dengan konten, sedangkan sentimen negatif terkait masalah teknis seperti pendaftaran dan pemblokiran akun (Fide, Suparti, & Sudarno, 2021).

Penelitian yang dilakukan oleh Hichem Rahab dkk (2021) melakukan analisis sentimen atau opini dari komentar yang ditulis dalam bahasa Arab di

situs web surat kabar Aljazair (Rahab, Zitouni, & Djoudi, 2021). Analisis sentimen pada penelitian tersebut bertujuan untuk mengklasifikasikan opini masyarakat ke dalam kategori tertentu untuk mempermudah pemahaman fenomena di balik teks opini tersebut. Dua korpus digunakan dalam penelitian ini yakni SANA (komentar dari tiga surat kabar Aljazair) dan OCA (korpus yang tersedia secara bebas untuk analisis sentimen). Pendekatan yang digunakan dalam penelitian tersebut adalah *supervised learning* dengan menggunakan tiga algoritma klasifikasi: *Support Vector Machines*, *Naïve Bayes*, dan *k-Nearest Neighbors*. Proses anotasi dilakukan dengan metode *MATTER* yang melibatkan beberapa iterasi untuk menyelesaikan proses anotasi. Validasi menggunakan metode *10-fold cross-validation* untuk mengevaluasi dan membandingkan algoritma. Penelitian ini tersebut terdiri dari dua kali eksperimen: 1) Model dan anotasi menggunakan pendekatan *MATTER*. *Pre-processing* menggunakan tokenisasi, stemming, penghapusan *stop words*, dan pembobotan kata menggunakan *Term Frequency-Inverse Document Frequency*. Evaluasi dilakukan dengan menghitung presisi, recall, dan akurasi untuk setiap klasifikasi. Hasil menunjukkan bahwa *Support Vector Machines* memberikan kinerja terbaik dengan metode TO dan BTO tanpa *stemming*, sementara *Naïve Bayes* dan *k-Nearest Neighbors* menunjukkan hasil terbaik dengan menggunakan *Term Frequency* dan *Term Frequency-Inverse Document Frequency*. 2) Model yang sama digunakan tanpa perubahan. *Anotasi* diarahkan untuk hanya mempertimbangkan konten komentar. Evaluasi menunjukkan peningkatan akurasi dengan penggunaan stemming ringan.

Hasil penelitian diatas menunjukkan bahwa kinerja algoritma berbeda berdasarkan korpus yang digunakan. Untuk *korpus OCA*, yang terdiri dari ulasan film, *Term Frequency* dan *Term Frequency – Inverse Document Frequency* memberikan hasil terbaik untuk *Support Vector Machine* dan *k-Nearest Neighbors*. Sementara untuk korpus *SANA*, efek stemming ringan lebih signifikan dan bi-gram serta tri-gram tidak memberikan peningkatan yang signifikan.

Friska Aditia Indriyani dkk (2023) membahas tentang analisis sentimen aplikasi tiktok menggunakan algoritma *naïve bayes* dan *support vector machine*. Ulasan atau komentar masyarakat dalam penelitian tersebut dikelompokkan menjadi ulasan positif dan negatif (Indriyani, Fauzi, & Faisal, 2023). Proses pengumpulan data pada penelitian tersebut dilakukan menggunakan teknik *web scraping* dan berhasil mengumpulkan sebanyak 2000 data komentar yang paling relevan. Selanjutnya melakukan *pre-processing data* yang mencakup *Case Folding*, *Tokenization*, *Stop Word Removal*, dan *Stemming*. Pembobotan data diterapkan menggunakan metode *Term Frequency – Inverse Document Frequency* untuk menghitung bobot dari setiap term berdasarkan frekuensinya dalam dokumen. Proses klasifikasi kemudian dijalankan menggunakan algoritma *Naïve Bayes* dan *Support Vector Machines* untuk mengkategorikan data tersebut. Tahapan terakhir yang melakukan evaluasi dua algoritma tersebut dengan menggunakan *Confusion Matrix* untuk mengukur efektivitas dan akurasi dari metode klasifikasi yang digunakan. Data yang telah melalui proses *pre-processing* dibagi dengan persentase 20% (400) data testing dan 80% (1600) data training. Hasil akurasi yang diperoleh algoritma *Naïve Bayes* sebesar 79%

dan sebesar 84% untuk *Support Vector Machine*. Dilihat dari nilai akurasi yang diperoleh masing-masing algoritma, dapat disimpulkan bahwa algoritma *Support Vector Machine* lebih baik daripada algoritma *Naïve Bayes*.

Junda Alfiah Zulqornain dkk (2021) dalam penelitiannya juga membahas tentang analisis sentimen masyarakat terhadap aplikasi tiktok menggunakan algoritma *Naïve Bayes* dan *Categorical Proportional Difference*. Aplikasi TikTok merupakan aplikasi yang dapat membagikan video, tetapi biasanya juga muncul video yang memiliki unsur vulgar yang dapat dilihat oleh anak dibawah umur karena aplikasi tersebut ini tidak terdapat batasan usia pemakai. Oleh sebab itu penelitian tersebut bertujuan untuk menganalisa sentimen pada ulasan aplikasi TikTok untuk membantu orang tua dalam pemilihan aplikasi untuk anaknya. Dataset yang digunakan dalam penelitian tersebut diambil dari *Google PlayStore* dari rating 1-5 yang berjumlah 1000 data dan mempunyai 2 kelas yakni positif dan negatif. Hasil akurasi terbaik yang diperoleh dalam penelitian tersebut yakni 0,729947 (73%), nilai *precision* sebesar 0,746854 (75%), nilai *recall* sebesar 0,926118 (93%), dan nilai *f-measure* 0,824511 (82%) (Zulqornain, Indriati, & Adikara, 2021).

Analisis sentimen terhadap aplikasi TikTok juga dilakukan oleh Dian Ardiansyah dkk (2023) menggunakan algoritma *K-Nearest Neighbor* dan *Support Vector Machine* yang dioptimasi dengan algoritma *Particle Swarm Optimization*. Dalam penelitian tersebut, data ulasan aplikasi TikTok digunakan untuk membandingkan kinerja algoritma *k-Nearest Neighbors* dan *Support Vector Machines* yang diotimasi menggunakan *Particle Swarm Optimization* dalam hal sentimen positif dan negatif. Tahapan dalam penelitian tersebut

meliputi proses pengambilan data ulasan dari Playstore dengan teknik *scrapping*, pelabelan secara manual, *preprocessing* (*Cleansing, Tokenizing, Case Transformation, Filter Stopwords, Filter Tokens*), pemodelan dan evaluasi. Metode *Cross Validation* digunakan untuk menguji hasil dari kedua algoritma tersebut, dan hasil pengujian menunjukkan bahwa algoritma *Support Vector Machines* yang dioptimalkan menggunakan *Particle Swarm Optimization* mencapai akurasi tertinggi dibandingkan dengan algoritma *k-Nearest Neighbors*. Akurasi *Support Vector Machines* yang dioptimalkan menggunakan *Particle Swarm Optimization* mencapai 88,20% dengan AUC sebesar 0,91, sedangkan algoritma *K-Nearest Neighbor* memiliki akurasi 83,40% dengan AUC sebesar 0,903. Algoritma *K-Nearest Neighbor* yang dioptimalkan menggunakan *Particle Swarm Optimization* mencapai akurasi 69,20% dengan AUC sebesar 0,77. Dengan demikian, dapat disimpulkan bahwa penggunaan algoritma *Support Vector Machine* yang dioptimalkan menggunakan *Particle Swarm Optimization* memberikan tingkat akurasi yang paling tinggi dalam penelitian ini (Ardiansyah, Saepudin, Aryanti, Fitriani, & Royadi, 2023).

Winda Yulita dkk (2021) melakukan analisis sentimen terhadap opini masyarakat tentang vaksin covid-19 menggunakan algoritma *Naïve Bayes Classifier*. Wabah virus corona telah mengakibatkan langkah-langkah yang belum pernah terjadi sebelumnya, memaksa pihak berwenang untuk mengambil keputusan terkait penerapan lockdown di beberapa daerah yang terkena dampak pandemi. Media sosial menjadi sarana penting bagi individu saat menghadapi tantangan ini. Pada tanggal 9 November 2020, ketika vaksin pertama dengan tingkat efektivitas lebih dari 90% diumumkan, media sosial bereaksi dan orang-

orang di seluruh dunia mulai mengekspresikan pendapat mereka tentang vaksinasi. Penelitian tersebut bertujuan untuk menganalisis opini tentang vaksinasi COVID-19 di Indonesia dengan menganalisis 3780 tweet yang berkaitan dengan topik vaksinasi menggunakan algoritma *Naïve Bayes Classifier*. Tahapan yang dilakukan dalam penelitian ini meliputi *collection and labeling data, preprocessing (Cleansing, Convert Negation, Convert Emoticon, Case Folding, Tokenization, Filtering/ stopwords, Stemming)*, pembobotan *Term Frequency - Inverse Document Frequency*, klasifikasi, dan evaluasi. Evaluasi yang digunakan dalam penelitian tersebut menggunakan *Confusion matrix*. Dari analisis yang telah dilakukan, mayoritas *tweet* menunjukkan sikap positif (60,3%), sedangkan jumlah *tweet* yang bersikap netral (34,4%) lebih banyak daripada yang menunjukkan penolakan (5,4%). Akurasi hasil analisis mencapai 0,93 (93%) (Yulita, Nugroho, & Algifari, 2021).

Penelitian terkait analisis sentimen juga dilakukan oleh Samsir dkk (2021) terkait dengan sentimen masyarakat terhadap pembelajaran daring pada *Twitter* di masa pandemi Covid-19. Dalam penelitiannya, Pandemi memaksa semua lembaga pendidikan mengadopsi pembelajaran daring sebagai strategi utama, namun hal ini menjadi kontroversial karena tantangan adaptasi yang singkat. Transisi tiba-tiba dari pembelajaran tatap muka ke daring menyebabkan berbagai respons di masyarakat. Penelitian tersebut bertujuan untuk menganalisis pandangan masyarakat terhadap pembelajaran daring pada masa pandemi COVID-19 di Indonesia pada awal November 2020. Metode penelitian melibatkan *scrapping* teks dari dokumen *Twitter* dan dianalisis menggunakan algoritma *Naïve Bayes*. Hasil menunjukkan bahwa pembelajaran daring

memiliki sentimen positif sebesar 30%, sentimen negatif sebesar 69%, dan netral sebesar 1% pada periode tersebut. Sentimen negatif yang tinggi dipicu oleh ketidakpuasan masyarakat terhadap pembelajaran daring, seperti yang tercermin dalam tweet yang menyatakan 'stres' dan 'malas', kata-kata yang sering muncul dalam percakapan (Samsir, Ambiyar, Verawardina, Edi, & Watrianthos, 2021).

Faizal Fakhri Irfani dkk (2020) melakukan analisis terhadap platform Ruangguru menggunakan algoritma *Support Vector Machine*. Dalam penelitiannya, dataset yang digunakan diambil dari website *Google Playstore* yang berupa data teks review yang berjumlah sebanyak 2000 review. Sebelum pembuatan model untuk proses klasifikasi sentimen, dataset dilakukan *pre-processing* yang meliputi *Case Folding*, *Tokenize*, *Cleansing*, *Filtering*, *Translation*, *Stopword Removal*, dan *Stemming*. Setelah proses *pre-processing* dilakukan selanjutnya dilakukan pembobotan kata dengan menggunakan *Term Frequency - Inverse Document Frequency*. Proses pembuatan model dan uji coba menggunakan algoritma *Support Vector Machine*. Data hasil *pre-processing* selanjutnya dilakukan pengujian dan evaluasi dengan menggunakan *K-Fold Cross Validation*. Melalui penggunaan metode *K-Fold Cross Validation*, ditemukan bahwa akurasi tertinggi terjadi pada nilai *K-Fold* 6, 9, dan 10, dengan nilai akurasi mencapai 0,902. Namun, pada *K-Fold 10*, presisinya lebih tinggi daripada nilai *K-Fold* lainnya, dengan nilai presisi sebesar 0,903. Tingkat akurasi dalam penelitian ini berada pada kisaran 90%. Hasil dari beberapa pengujian menunjukkan bahwa sentimen masyarakat terhadap aplikasi Ruangguru cenderung positif (Irfani, Triyanto, Hartanto, & Kusnawi, 2020).

Penelitian yang serupa terkait sentimen masyarakat terhadap *platform* Ruangguru juga dilakukan oleh Hindun Habibatul Mubaroroh dkk (2022) dengan menggunakan algoritma *Naïve Bayes Classifier* dengan normalisasi kata *levenshtein distance*. Data yang dikumpulkan sebanyak 1500 ulasan namun data yang termasuk spam dihilangkan sehingga dataset yang digunakan sebanyak 1484 data ulasan. Pada penelitiannya metode *levenshtein distance* digunakan saat *pre-processing* setelah dilakukan *case folding* dan *cleansing*, yang dilanjutkan dengan proses pemberian label positif maupun negatif terhadap ulasan yang ada. Setelah itu dilanjutkan dengan proses tahap *stopwords removal*, *stemming*, *tokenizing*, dan pembobotan *Term Frequency - Inverse Document Frequency*. Pengujian data menggunakan metode *10-fold cross validation*. Dengan menerapkan metode *10-fold cross validation*, algoritma *Naïve Bayes* menghasilkan rata-rata akurasi sebesar 88,20%, dengan nilai akurasi tertinggi terjadi pada fold ke-8, mencapai 94%. Hal ini mengindikasikan bahwa algoritma *Naïve Bayes* efektif dalam mengklasifikasikan opini pengguna aplikasi Ruangguru (Mubaroroh, Yasin, & Rusgiyono, 2022).

Indah Novitasari dkk (2022) juga melakukan analisis sentimen masyarakat terhadap *tweet* ruang guru. Dalam penelitian ini, analisis sentimen akan dilakukan dengan mengklasifikasikan ke dalam label sentimen positif, negatif, dan netral, menggunakan algoritma *Naive Bayes Classifier*. Algoritma ini dipilih karena kemampuan klasifikasinya yang memuaskan, bahkan dengan jumlah data training yang sedikit, serta kemampuan yang akurat dan cepat dalam menangani kumpulan data besar (Han, Kamber, & Pei, 2012; Mccallum & Nigam, 2001). Tahapan penelitian meliputi pengumpulan data, *preprocessing*, pengolahan data,

akurasi, dan evaluasi. Hasil penelitian menunjukkan bahwa prediksi yang dihasilkan cenderung dominan pada sentimen positif. Dari pengujian menggunakan metode *Naive Bayes Classifier*, didapatkan nilai presisi sebesar 71%, *recall* 69%, dan *F1-score* 69%, dengan tingkat akurasi sebesar 69% (Novitasari, Kurniawan, Dewi, & Misinem, 2022).

Analisis sentimen melibatkan penilaian opini yang berkaitan dengan kejadian saat ini. Dalam konteks penelitian, analisis sentimen digunakan untuk mengevaluasi dan memahami respon emosional seseorang terhadap berbagai kondisi, seperti pelayanan produk, kinerja organisasi, atau peristiwa fenomenal. Tujuan dari analisis sentimen adalah untuk memperoleh wawasan tentang pandangan masyarakat terhadap isu tertentu serta untuk mengidentifikasi tren atau kecenderungan dalam pembicaraan publik (Pawar, Jawale, & Kyatanavar, 2016; Wenando, Hayami, & Anggrawan, 2020).

Analisis sentimen dalam penelitian ini merupakan komentar masyarakat Indonesia di *plyastore* tentang aplikasi TikTok untuk mengetahui emosi seseorang yang meliputi positif dan negatif terhadap aplikasi TikTok.

Analisis sentimen merupakan proses klasifikasi yang kompleks karena bahasa sering kali mengandung kata-kata yang ambigu. Dalam analisis sentimen, terdapat berbagai tingkat analisis yang berbeda, seperti Wenando dkk (2020):

a. Kalimat

Tingkat kalimat melibatkan evaluasi kalimat untuk menentukan apakah masing-masing kalimat memiliki sentimen positif, negatif, atau netral. Secara umum, kalimat yang netral tidak mengandung pendapat atau opini.

b. Dokumen

Tingkat dokumen akan melakukan proses klasifikasi untuk menentukan apakah dokumen tersebut memiliki sentimen positif atau negatif. Saat menganalisis dokumen, perlu diperhatikan bahwa dokumen yang hanya berisi opini tentang satu entitas tidak dapat diterapkan pada dokumen yang mengandung lebih dari satu entitas.

c. Entitas serta Aspek

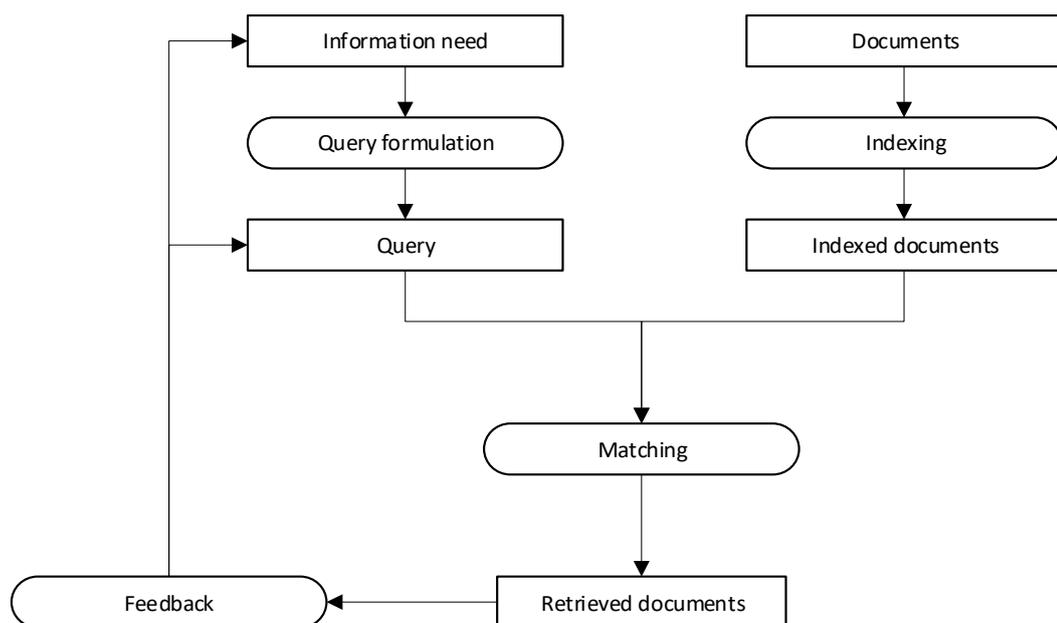
Tingkat aspek terdiri dari dua langkah yang belum menetapkan dengan pasti apa yang disukai dan tidak disukai oleh orang, tetapi lebih fokus pada gagasan dan opini yang terkandung dalam komentar positif dan negatif.

Selain itu, analisis sentimen erat kaitannya dengan ungkapan emosi seseorang yang dituangkan dalam bentuk ekspresi maupun kalimat/ucapan. Makna dari kalimat/ucapan tersebut mengindikasikan kecenderungan alami yang mutlak dalam kehidupan makhluk hidup. Emosi mencerminkan keadaan dan ekspresi yang unik dalam kondisi psikologis dan biologis, serta kecenderungan untuk berperilaku. Emosi merupakan tanggapan terhadap rangsangan dari lingkungan luar maupun dalam diri seseorang. Emosi seperti kegembiraan, kasih sayang, dan kebahagiaan memicu perubahan fisik yang terlihat, seperti tertawa. Sebaliknya, dalam keadaan sedih, seseorang cenderung menunjukkan perilaku marah, kesal, ataupun menangis.

*Information Retrieval System* atau yang biasa dikenal sebagai sistem pemerolehan data, merupakan perangkat lunak yang menyimpan dan mengelola data dari berbagai dokumen, termasuk teks dan multimedia. Sistem ini tidak selalu dapat memberikan jawaban langsung atau akurat atas suatu pertanyaan,

tetapi biasanya memberikan lokasi dokumen yang mungkin mengandung data yang relevan dengan pertanyaan yang diajukan. Sistem pemerolehan data yang efektif akan menganggap dokumen yang relevan sebagai dokumen yang penting. Namun, sistem ini tidak mempertimbangkan subjektivitas opini pengguna dalam menentukan relevansi. Pengguna dapat menggunakan *query* yang spesifik dalam sistem dan memperhitungkan berbagai dokumen yang ditemukan dengan cara yang berbeda, baik menyukainya maupun tidak (Worth, 2010).

Terdapat tiga proses yang harus didukung oleh sistem pemerolehan data: pertama, menafsirkan konten dokumen; kedua, memahami data yang diinginkan oleh pengguna; dan ketiga, membandingkan kedua aspek tersebut.



Gambar 2.1. Sistem Pemodelan Pengolahan Data

Pada Gambar 2.1. diatas merupakan alur proses deskripsi yang menghasilkan dokumen. Sistem pemerolehan data menggunakan metode yang secara penuh mengidentifikasi setiap kata dalam teks, baik dalam huruf kapital maupun huruf kecil. Proses deskripsi ini memungkinkan penyimpanan dokumen ke dalam sistem. Namun, terkadang sistem hanya menyimpan sebagian contoh

judul dan abstrak, dan menambahkan informasi posisi dokumen yang sesungguhnya ke dalam data.

Pengguna tidak hanya mencari hiburan, tetapi juga memiliki kebutuhan akan informasi dan data. Proses perumusan kebutuhan informasi sering kali merupakan tahap dimana pengguna merumuskan pertanyaan atau *query* yang lebih luas untuk mendapatkan pemahaman yang lebih baik dari sistem tentang data yang mereka butuhkan, yang kemudian bisa menjadi dasar umpan balik informasi. Dalam proses *query* dan penilaian dokumen, ada bagian yang disebut sebagai *Process Matching* (proses pencocokan). *Process Matching* umumnya menghasilkan daftar dokumen yang terstruktur. Penempatan dokumen yang terstruktur dengan baik di bagian atas daftar dapat meningkatkan relevansi dan memperpendek waktu baca pengguna. Berikut merupakan beberapa proses information retrieval (Worth, 2010) :

a. *Tokenization*

Suatu dokumen unit mempunyai lapisan perkata khusus dari tugas tokenisasi untuk memecah lapisan kata jadi terpisah yang disampaikan suatu token. Berikut contohnya:

Input: analisis sentimen masyarakat, terhadap aplikasi tiktok.

Output: |analisis| |sentimen| |masyarakat| |terhadap| |aplikasi| |tiktok|

Setelah dilakukan tokenisasi tanda baca titik(.), koma (,), seru(!), pagar atau hashtag (#), dan add (@) dan symbol lainnya.

b. *Stopword Removal*

*Stopword Removal* merupakan *preprocessing text mining* pada saat melakukan eliminasi suatu kata. Kata-kata yang di eliminasi merupakan

kata yang sangat kerap timbul dalam sesuatu dokumen yang mempunyai penilaian sedikit serta membantu memilah dokumen diperlukan tahapan dalam menghilangkan kata sambungan maupun perkataan yang sering muncul tapi tidak bermakna. seperti: ini, itu, yaitu, maka, walaupun, dan lain-lainnya.

c. *Normalization*

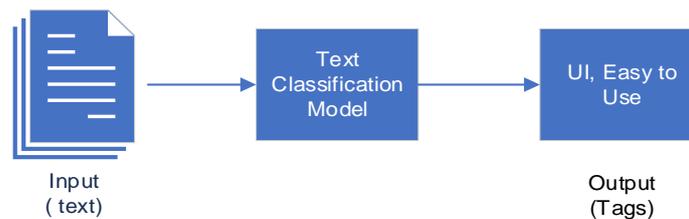
Merupakan proses persamaan dalam kata eja yang mempunyai arti yang sama, contoh kata yang dicari 'india' ketika mengharapkan akan memperoleh 'I N. D. I. A'. Dalam bahasa Indonesia banyak ditemukan kata tidak jelas seperti 'slama' dalam artian 'selama'. Sehingga dalam menggunakan kata ejaan yang benar harus berpedoman pada Kamus Besar Bahasa Indonesia (KBBI).

d. *Stemming*

Tujuan dari *stemming* adalah untuk mengurangi infleksi (pembuatan kata baru) akibat akumulasi sesuatu imbuhan akhiran serta sisipan sehingga mengubah suatu kata ke bentuk kata dasarnya. *Stemming* merupakan langkah dari *preprocessing* bacaan yang digunakan untuk menciptakan kata dasar tanpa ada imbuhan kata apapun. Seperti kata "mbingungkan" menjadi "bingung" dan "membuat" menjadi "buat" dan lainnya.

Klasifikasi teks merupakan suatu proses yang digunakan untuk menentukan kategori di mana suatu teks harus ditempatkan. Dalam melakukan klasifikasi teks, metode yang digunakan meliputi penyusunan teks, pengelompokan teks, dan kategorisasi kata. Sebagai contoh, tulisan dapat dikelompokkan berdasarkan tema, keluhan pelanggan dapat dikelompokkan

berdasarkan relevansi sumbernya, dan percakapan dapat dikelompokkan berdasarkan bahasa yang digunakan (Roldós, 2020). Alat yang digunakan dalam proses klasifikasi mengambil input, menganalisis data, dan kemudian secara otomatis menetapkan tag yang relevan, mirip dengan proses yang diilustrasikan dalam Gambar 2.2.



Gambar 2.2. Implementasi klasifikasi teks

Klasifikasi teks dengan *deep learning* melibatkan pembelajaran dari pola-pola yang ada dalam data sebelumnya. Dengan menggunakan contoh yang telah diberi label sebagai data latih, metode *deep learning* dapat mengidentifikasi hubungan yang berbeda antara bagian-bagian teks, di mana *output*-nya adalah *tag* yang telah ditentukan sebelumnya untuk input berupa teks. Sebagai contoh, menentukan kamus dengan kata-kata seperti "ini, adalah, itu, tidak, luar biasa, kurang baik, bola basket," dan membuat vektor dari teks "ini luar biasa," maka akan memiliki representasi vektor berikut dari teks tersebut: (1, 1, 0, 0, 1, 0, 0) (Roldós, 2020).

*Term Frequency - Inverse Document Frequency* yang disingkat TF-IDF adalah metode untuk menentukan hal penting dari sebuah kata terhadap data yang telah di ekstraksi (Bafna, Pramod, & Vaidya, 2016; Cheng, Yang, Zhao, & Gao, 2020). Metode akan berjalan dengan menghitung jumlah kata yang unik dari kata serta dibandingkan total kata dengan metode yang diterapkan (Hananto, Pinandito, & Kharisma, 2018).

Ketika menghitung frekuensi kata, masalah umum yang muncul adalah istilah yang sering muncul mulai mendominasi teks, tetapi mungkin tidak memiliki 'konten informasi' yang diperlukan agar model dapat membedakan dengan benar (Bafna et al., 2016; Neogi, Garg, Mishra, & Dwivedi, 2021). *Inverse Document Frequency* adalah metrik untuk mengukur signifikansi sebuah kata. Kita memerlukan nilai *Inverse Document Frequency* karena hanya menghitung TF tidak cukup untuk menghargai signifikansi kata-kata tersebut. Proses *Term Frequency - Inverse Document Frequency* dibagi menjadi dua proses yakni proses TF (*Term Frequency*) dan proses IDF (*Inverse Document Frequency*).

a. *Term Frequency*

Term Frequency digunakan untuk memastikan pembobotan kata dalam sebuah dokumen berdasarkan frekuensi kemunculannya dalam dokumen tersebut. Semakin sering sebuah kata muncul dalam dokumen, semakin tinggi pembobotannya. Ketika kumpulan dokumen semakin besar, bobot suatu kata dalam menilai dokumen juga akan meningkat. Adapun persamaan yang digunakan untuk menghitung *Term Frequency* yakni sebagai berikut (Neogi, Garg, Mishra, & Dwivedi, 2021).

$$TF(t, d) = \frac{N(t,d)}{T}$$

Di sini,  $TF(t,d)$  mewakili frekuensi istilah dari istilah  $t$  dalam dokumen  $d$ ,  $N(t,d)$  adalah jumlah kemunculan istilah  $t$  dalam dokumen  $d$ , dan  $T$  adalah jumlah total istilah dalam dokumen tersebut. Jadi, untuk setiap dokumen dan kata, nilai  $TF(t,d)$  yang berbeda akan diberikan.

b. *Inverse Document Frequency*

*Inverse Document Frequency* diperlukan karena ketika sebuah kata muncul secara umum dalam banyak dokumen, kata tersebut kehilangan signifikansinya. Meskipun kejadian kata yang jarang muncul harus dipertimbangkan dalam memberikan bobot pada kumpulan dokumen. Adapun persamaan yang digunakan untuk menghitung *Inverse Document Frequency* yakni sebagai berikut (Neogi, Garg, Mishra, & Dwivedi, 2021).

$$IDF(t) = \frac{\log N}{N(t)}$$

Persamaan diatas menunjukkan perhitungan IDF(t), yang merupakan inverse document frequency dari istilah t, N adalah jumlah dokumen, dan N(t) adalah jumlah dokumen yang mengandung istilah t. Inverse Document Frequency dibutuhkan karena *term* (kata) yang banyak timbul sebagai dokumen, mampu dianggap sebagai term universal hingga tidak berarti nilainya. Meskipun faktor kejarangan membawa kata term scarcity wajib dicermati dalam pemberian bobot pada koleksi dokumen. Adapun persamaan yang digunakan untuk menghitung *Term Frequency - Inverse Document Frequency* yakni sebagai berikut (Neogi, Garg, Mishra, & Dwivedi, 2021).

$$TF - IDF = TF * IDF$$

Adapun langkah-langkah proses pembobotan *Term Frequency - Inverse Document Frequency* yakni sebagai berikut:

- a. Menghitung jumlah TF (*Term Frequency*) setiap kata pada dokumen pada tabel 2.1.

Tabel 2.1. Term Frequency

Kata	TF (Term Frequency)					
	D6	D1	D2	D3	D4	D5
Belajar	0	1	1	1	1	0
Rumah	0	1	0	1	1	0
Saja	0	1	0	0	0	0
Tetap	1	1	0	1	0	0
Semangat	1	1	0	1	0	0

- b. Menghitung DF (*Document Frequency*) pada tabel 2.2.

Tabel 2.2. Document Frequency

Kata	TF (Term Frequency)						DF
	D6	D1	D2	D3	D4	D5	
Belajar	0	1	1	1	1	0	4
Rumah	0	1	0	1	1	0	3
Saja	0	1	0	0	0	0	1
Tetap	1	1	0	1	0	0	3
Semangat	1	1	0	1	0	0	3

- c. Menghitung IDF (*Inverse Document Frequency*) pada tabel 2.3.

Tabel 2.3. Inverse Document Frequency

Kata	IDF
	$\text{Log}(v/df)$
Belajar	$\log(6/4)^{10} = 0.176$
Rumah	$\log(6/3)^{10} = 0.301$
Saja	$\log(6/1)^{10} = 0.778$
Tetap	$\log(6/3)^{10} = 0.301$
Semangat	$\log(6/3)^{10} = 0.301$

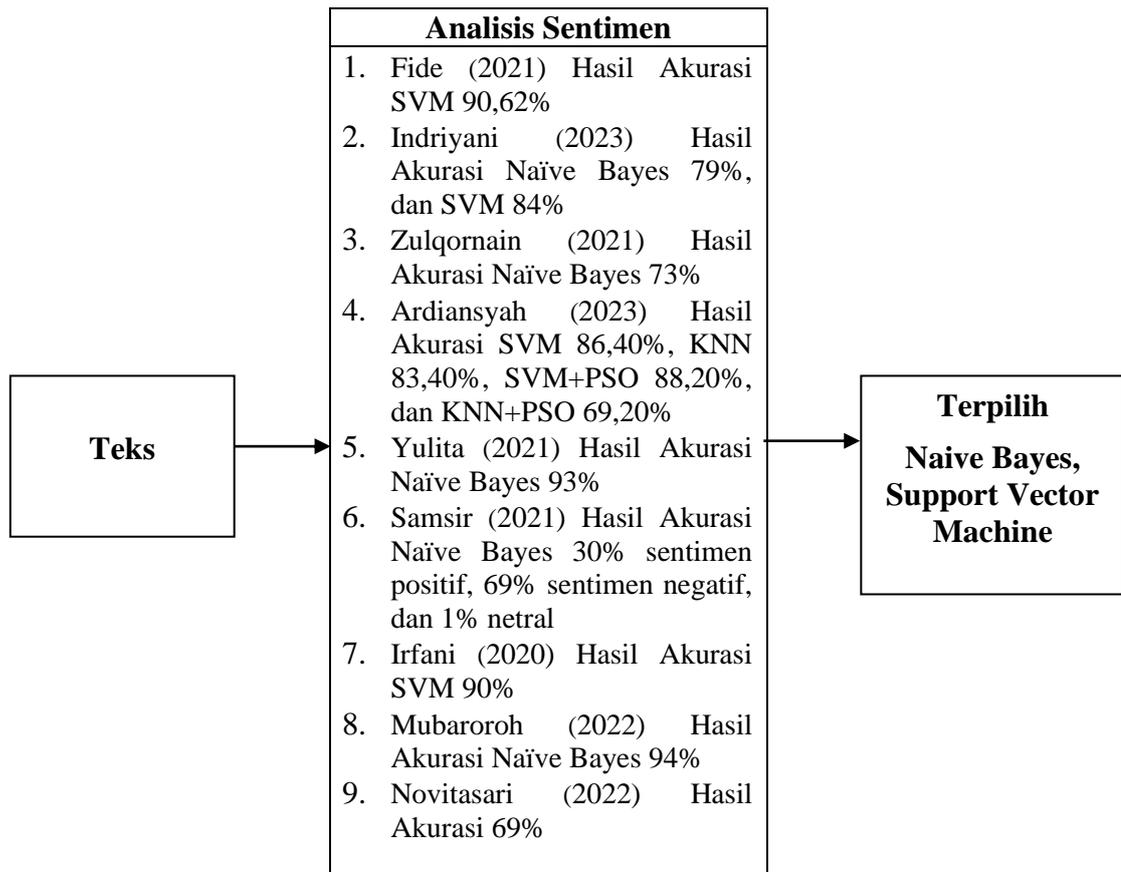
- d. Menghitung TF-IDF (*Term Frequency - Inverse Document Frequency*) pada tabel 2.4.

Tabel 2.4. Term Frequency - Inverse Document Frequency

TF-IDF( <i>TERM FREQUENCY</i> )					
D6	D1	D2	D3	D4	D5
0	0.176	0.176	0.176	0.176	0
0	0.301	0	0.301	0.301	0
0	0.778	0	0	0	0
0.301	0.301	0	0.301	0	0
0.301	0.301	0	0.301	0	0

## 2.2. Kerangka Teori

Kerangka teori dalam penelitian ini didasarkan pada referensi beberapa jurnal yang tersaji secara jelas dalam Gambar 2.3.



Gambar 2.3. Kerangka Teori

Dalam Gambar 2.3. terlihat bahwa metode yang dipilih untuk penelitian ini adalah algoritma *Naïve Bayes & Support Vector Machine*, karena kedua algoritma tersebut menunjukkan tingkat akurasi yang tinggi. Rangkuman artikel penelitian yang menjadi dasar penelitian ini tersaji dalam tabel 2.5.

Tabel 2.5. Daftar Artikel Penelitian

No	Nama Peneliti	Judul Penelitian	Metode	Hasil
1	Sola Fide, Suparti, Sudarno (2021)	Analisis Sentimen Ulasan Aplikasi TikTok di Google Play Menggunakan Metode Support Vector Machine (SVM) dan Asosiasi	SVM dan Asosiasi	Hasil pengujian yang telah dilakukan, algoritma SVM memperoleh hasil akurasi 90.62% dan kappa 81.24%
2	Friska Aditia Indriyani, Ahmad Fauzi, Sutan Faisal	Analisis Sentimen Aplikasi Tiktok Menggunakan Algoritma Naïve Bayes dan Support Vector Machine	Naïve Bayes dan Support Vector Machine	Hasil akurasi yang diperoleh algoritma Naïve Bayes sebesar 79% dan sebesar 84% untuk Support Vector Machine
3	Junda Alfiah Zulqornain, Indriati, Putra Pandu Adikara (2023)	Analisis Sentimen Tanggapan Masyarakat Aplikasi Tiktok Menggunakan Metode Naïve Bayes dan Categorical Proportional Difference (CPD)	Naïve Bayes dan Categorical Proportional Difference (CPD)	Hasil akurasi terbaik yang diperoleh dalam penelitian tersebut yakni 0,729947 (73%), nilai <i>precision</i> sebesar 0,746854 (75%), nilai <i>recall</i> sebesar 0,926118 (93%), dan nilai <i>f-measure</i> 0,824511 (82%)
4	Dian Ardiansyah, Atang Saepudin, Riska Aryanti, Eka Fitriani, Royadi (2023)	Analisis Sentimen Review Pada Aplikasi Media Sosial Tiktok Menggunakan Algoritma K-NN dan SVM Berbasis PSO	K-NN dan SVM Berbasis PSO	Nilai akurasi SVM 86,40%, akurasi SVM+PSO memiliki akurasi 88,20%. Sedangkan akurasi algoritma K-NN 83,40% dan akurasi K-NN+PSO 69,20%
5	Winda Yulita, Eko Dwi Nugroho, Muhammad Habib Algifari, (2021)	Analisis Sentimen Terhadap Opini Masyarakat Tentang Vaksin Covid-19 Menggunakan Algoritma Naïve Bayes Classifier	Naïve Bayes Classifier	Dari analisis yang telah dilakukan, mayoritas tweet menunjukkan sikap positif (60,3%), jumlah tweet yang bersikap netral (34,4%) lebih banyak daripada yang menunjukkan penolakan (5,4%). Akurasi hasil analisis mencapai 0,93 (93%)
6	Samsir, Ambiyar, Unung Verawardina,	Analisis Sentimen Pembelajaran Daring Pada Twitter di Masa	Naïve Bayes	Hasil menunjukkan bahwa pembelajaran daring memiliki sentimen

No	Nama Peneliti	Judul Penelitian	Metode	Hasil
	Firman Edi, Ronal Watrianthos (2021)	Pandemi COVID-19 Menggunakan Metode Naive Bayes		positif sebesar 30%, sentimen negatif sebesar 69%, dan netral sebesar 1% pada periode tersebut
7	Faizal Fakhri Irfani, Mohamad Triyanto, Anggit Dwi Hartanto, Kusnawi (2020)	Analisis Sentimen Review Aplikasi Ruangguru Menggunakan Algoritma Support Vector Machine	Support Vector Machine	Pada K-Fold 10, presisinya lebih tinggi daripada nilai K-Fold lainnya, dengan nilai presisi sebesar 0,903. Tingkat akurasi dalam penelitian ini berada pada kisaran 90%
8	Hindun Habibatul Mubaroroh, Hasbi Yasin, Agus Rusgiyono (2022)	Analisis Sentimen Data Ulasan Aplikasi Ruangguru Pada Situs Google Play Menggunakan Algoritma Naive Bayes Classifier Dengan Normalisasi Kata Levenshtein Distance	Naive Bayes Classifier	Algoritma <i>Naive Bayes</i> menghasilkan rata-rata akurasi sebesar 88,20%, dengan nilai akurasi tertinggi terjadi pada fold ke-8, mencapai 94%
9	Indah Novitasari, Tri Basuki Kurniawan, Deshinta Arrova Dewi, Misinem (2022)	Analisis Sentimen Masyarakat Terhadap Tweet Ruang Guru Menggunakan Algoritma Naive Bayes Classifier (NBC)	Naive Bayes Classifier	Dari pengujian menggunakan metode <i>Naive Bayes Classifier</i> , didapatkan nilai presisi sebesar 71%, <i>recall</i> 69%, dan <i>F1-score</i> 69%, dengan tingkat akurasi sebesar 69%
10	Hichem Rahab, Abdelhafid Zitouni, Mahieddine Djoudi (2021)	SANA: Sentiment analysis on newspapers comments in Algeria	Supports vector machines, naive Bayes, k-nearest neighbors	Hasil penelitian menunjukkan kinerja algoritma berbeda berdasarkan korpus yang digunakan. Untuk korpus OCA, yang terdiri dari ulasan film, TF dan TF-IDF memberikan hasil terbaik untuk SVM dan KNN. Sementara untuk korpus SANA, efek stemming ringan lebih signifikan dan bi-gram serta tri-gram tidak memberikan peningkatan yang signifikan.

Berdasarkan penelitian sebelumnya dan rangkuman tabel diatas, parameter pada algoritma *Support Vector Machine* yang digunakan adalah menggunakan *Kernel Linear*. Dalam penelitian ini akan menggunakan perubahan pada *Kernel*

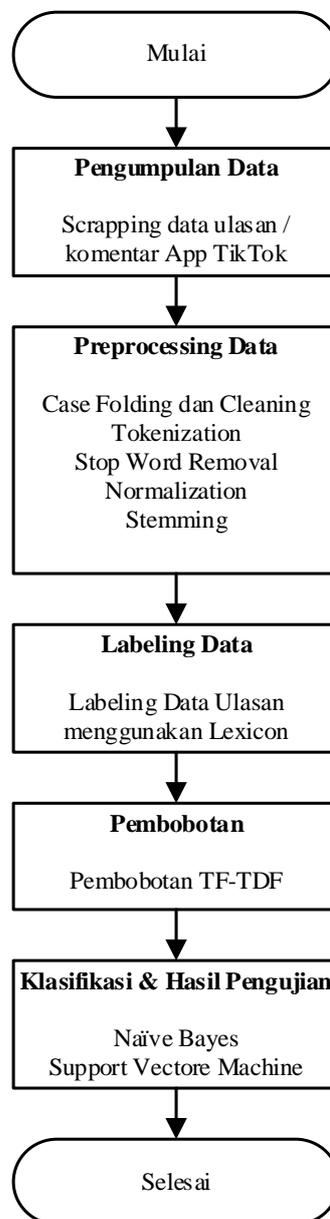
yang terdapat pada Algoritma *Support Vector Machine* yang meliputi *Kernel Linear, Polynomial dan Radial Basis Function* (RBF) untuk mengetahui *Kernel* yang cocok untuk klasifikasi sentimen TikTok serta mengetahui akurasi yang optimal dari beberapa kernel yang digunakan.

## BAB III

### METODOLOGI PENELITIAN

#### 3.1. Prosedur Penelitian

Pada penelitian ini dibutuhkan prosedur yang benar agar penelitian ini dapat berjalan dengan efektif. Berikut alur penelitian ini dapat dilihat pada gambar 3.1.



Gambar 3.1. Alur Penelitian

### 3.2. Pengumpulan Data

Data yang digunakan dalam penelitian ini menggunakan data ulasan / komentar masyarakat Indonesia terhadap Aplikasi Tik-Tok di *Google PlayStore*. Pengumpulan data dilakukan dengan cara *Scrapping* data komentar Tik-Tok menggunakan *Google Colaboratory* dengan ketentuan hanya melakukan *scrapping* dengan id-Indonesia (komentar masyarakat Indonesia saja). Jumlah data yang terkumpul yakni sebanyak 313.187 data dari tanggal 28 Juni 2023 hingga 31 Desember 2023, namun data yang digunakan dalam penelitian ini sebanyak 5.000 dataset, hal ini dikarenakan keterbatasan *resource* yang digunakan pada penelitian ini. Berikut sebagian data hasil *scrapping* ulasan / komentar TikTok yang telah dikumpulkan dapat dilihat pada tabel 3.1.

Tabel 3.1. Data Ulasan / Komentar App TikTok

No	Komentar	Skor	Tanggal
1	baguuzzz bgtttz aks sukaaaak 🤔 🤔 🤔 🤔 🤔 🤔	5	2023-12-31 23:59:24
2	Aplikasi gak jelas, padahal enggak pernah ngirim/ngetik aneh aneh malah ditanggihkan nungguin lama tetep gak bisa, gak bisa ngirim pesan, enggak bisa terima pesan tolong ya aplikasi nya di perbaiki lagi	1	2023-12-31 23:58:40
3	Skrang Tiktok Terlalu sensitif biar kata tidak mengandung kata Negatif... Biar pun bahasa daerah yang kata membangun Semangat... Sedikit-Sediki di hapus.. Biar pun di Banding" tetap aja di hapus....Tolong di perhatikan dan belajar bahasa daerah	1	2023-12-31 23:58:33
4	Keren	4	2023-12-31 23:52:35
5	Terlalu banyak pelanggaran yg di berikan Padahal vidio buat sendiri tapi kena pelanggaran sampai 3 point Tolong di perbaiki sistem nya	1	2023-12-31 23:52:09
6	👍👍👍👍👍👍👍👍	1	2023-06-28 20:25:38
7	Baguss bngtt 😊	5	2023-06-28 20:25:10
8	Aplikasi ini sangat bagus...	5	2023-06-28 20:24:47
9	Tidak bisa di download	1	2023-06-28 20:24:07

Setelah proses pengumpulan data yang dilakukan, selanjutnya melakukan pembersihan data duplikat berdasarkan data nama pengguna yang berkomentar serta menghapus teks unique seperti *URL*, *emoji*, dan *simbol*. Kemudian melakukan *preprocessing data* dengan memecah kalimat jadi perkata (*Tokenization*), lalu dilakukan proses perbaikan kata yang tidak baku menjadi baku (*Normalization*), dan membuang kata yang banyak kemunculannya / kata penghubung (*Stopword Removal*) serta membuang kata imbuhan dari kata yang memiliki imbuhan (*Stemming*).

### **3.3. Pre-Processing**

*Preprocessing* merupakan proses mengubah data yang tidak baik menjadi baik untuk memperoleh sumber keaslian informasi yang akan dilakukan proses selanjutnya. Pada tahapan ini dilakukan beberapa tahapan diantaranya membersihkan, memperbaiki dan menggabungkan data. Penelitian ini akan melakukan *preprocessing data* dengan memecah kalimat jadi perkata (*Tokenization*), lalu dilakukan proses perbaikan kata yang tidak baku menjadi baku (*Normalization*), dan membuang kata yang banyak kemunculannya / kata penghubung (*Stopword Removal*) serta membuang kata imbuhan dari kata yang memiliki imbuhan (*Stemming*). Berikut langkah-langkah dan penjelasan dari proses *preprocessing* tersebut:

#### a. *Case Folding* dan *Cleaning*

Berfungsi untuk mengubah huruf yang tidak *lowercase* atau huruf kapital menjadi huruf kecil semua atau disebut dengan *lowercase* serta menghapus teks unique seperti *URL*, *emoji*, dan *simbol*.

b. *Tokenization*

Berfungsi untuk memecah perkata pada kalimat yang telah diproses pembersihan kata yaitu membuang angka, tanda baca dan simbol pada kalimat.

c. *Stopword Removal*

Berfungsi membuang kata dasar yang memiliki kemunculan sangat banyak serta tidak bermakna seperti “yang”, “dan”, “di”, “dari”, dan lain-lainnya.

d. *Normalization*

Berfungsi untuk memperbaiki kata yang tidak baku menjadi baku, seperti ‘slama’ dalam artian ‘selama’. Sehingga dalam menggunakan kata ejaan yang benar harus berpedoman pada Kamus Besar Bahasa Indonesia (KBBI).

e. *Stemming*

Berfungsi untuk membuang kata imbuhan pada kata bermunculan seperti "menjadikan" maka hasilnya "jadi".

Adapun contoh dari beberapa tahapan di atas dapat dilihat pada tabel 3.2.

Tabel 3.2. Contoh Proses Pre-Processing Data

No	Komentar	Case Folding	Stopword Removal	Stemming
1	baguuzzz bgtttz aks sukaaaak 🤔😞❤️🔥 ❤️🔥	baguuzzz bgtttz aks sukaaaak 🤔😞❤️🔥❤️🔥	baguuzzz bgtttz aks sukaaaak 🤔😞❤️🔥❤️🔥	bagus banget aku suka
2	Aplikasi gak jelas, padahal enggak pernah ngirim/ngetik aneh aneh malah ditanggihkan nungguin lama tetep gak bisa, gak bisa ngirim pesan, enggak bisa terima pesan tolong ya aplikasi nya di perbaiki lagi	aplikasi gak jelas, padahal enggak pernah ngirim/ngetik aneh aneh malah ditanggihkan nungguin lama tetep gak bisa, gak bisa ngirim pesan, enggak bisa terima pesan tolong ya aplikasi nya di perbaiki lagi	aplikasi gak jelas, enggak kirim/ngetik aneh aneh ditanggihkan nungguin gak bisa, gak bisa ngirim pesan, enggak bisa terima pesan tolong aplikasi perbaiki lagi	aplikasi gak jelas, enggak kirim/ketik aneh aneh tangguh tunggu gak bisa, gak bisa kirim pesan, enggak bisa terima pesan tolong aplikasi baik lagi
3	Skrang Tiktok Terlalu sensitif biar kata tidak mengandung kata Negatif... Biar pun bahasa daerah yang kata membangun Semangat... Sedikit-Sediki di hapus.. Biar pun di Banding" tetap aja di hapus....Tolong di perhatikan dan belajar bahasa daerah	skrang tiktok terlalu sensitif biar kata tidak mengandung kata negatif... biar pun bahasa daerah yang kata membangun semangat... sedikit- sediki di hapus.. biar pun di banding" tetap aja di hapus....tolong di perhatikan dan belajar bahasa daerah	skrang tiktok sensitif kata mengandung kata negatif... bahasa daerah kata membangun semangat... sedikit- sediki hapus.. banding" tetap hapus....tolong perhatikan belajar bahasa daerah	skrang tiktok sensitif kata mengandung kata negatif... bahasa daerah kata bangun semangat... sedikit hapus.. banding" tetap hapus....tolong perhati belajar bahasa daerah

### 3.4. Lexicon

Berfungsi untuk mengidentifikasi sebuah kata dalam kalimat yang memiliki sifat positif, negatif atau netral dengan menghitung nilai *polarity* (Pradhan, Senapati, & Sahu, 2022). Cara melakukan untuk mengidentifikasi kata untuk mengetahui sentimen pada kata, bisa menggunakan dataset lexicon berbahasa Indonesia. Langkah pertama dalam proses ini adalah pengembangan atau pemilihan lexicon yang sesuai. Setelah lexicon disiapkan, proses labeling dimulai dengan analisis teks yang akan dilabeli. Teks ini dipecah menjadi unit-unit yang lebih kecil, biasanya kata-kata, yang kemudian dicocokkan dengan entri yang ada dalam lexicon. Setiap kata dalam teks diberi skor berdasarkan

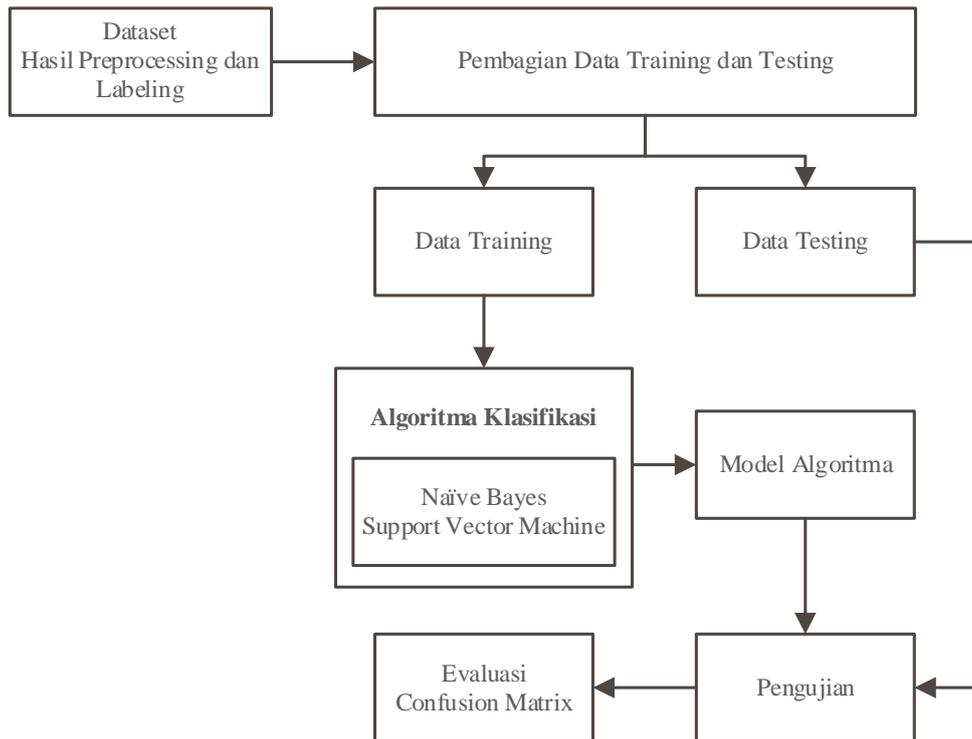
nilai sentimen yang terkait dalam lexicon. Misalnya, kata "bahagia" mungkin memiliki nilai positif, sedangkan kata "sedih" memiliki nilai negatif. Proses ini memungkinkan peneliti untuk secara otomatis mengumpulkan skor dari semua kata dalam teks dan mengagregasi skor tersebut untuk mendapatkan representasi sentimen keseluruhan dari teks.

#### **3.4.1 Pembobotan Term Frequency - Inverse Document Frequency**

Pembobotan kata *Term Frequency - Inverse Document Frequency* yang bertujuan untuk menghitung nilai bobot masing-masing kata setiap dokumen. Pada tahapan ini dibagi menjadi 2 proses yaitu *Term Frequency* dan *Inverse Document Frequency*. *TF (Term Frequency)* menghitung jumlah kemunculan setiap kata pada dokumen serta dengan kemunculan kata tersebut paling banyak maka nilai kata itu paling besar. *IDF (Inverse Document Frequency)* menghitung jumlah dokumen pada setiap kata yang jarang kemunculan pada sebuah dokumen yang dianggap nilai paling besar. Jika kata tersebut banyak kemunculan kata pada dokumen maka hasil nilainya kecil.

#### **3.4.2 Klasifikasi Naive Bayes dan Support Vector Machine**

Alur proses klasifikasi sentimen pada penelitian ini dapat dilihat pada gambar 3.2.



Gambar 3.2. Alur Proses Pengujian Algoritma

Pengujian dilakukan dengan perbandingan data 80:20, 70:30, dan 60:40 untuk data *training* (latih) dan data *testing* (uji). Hal ini bertujuan untuk mengetahui hasil akurasi dari kinerja algoritma *Naïve Bayes* dan *Support Vector Machine* dalam proses klasifikasi sentimen masyarakat terhadap aplikasi TikTok.

#### a. *Naïve Bayes*

Proses klasifikasi menggunakan metode *naive bayes* melibatkan dua jenis data: *data training* dan *data testing*. *Data training* berfungsi untuk menghitung probabilitas setiap kata dalam dataset. Kemudian, *data testing* digunakan untuk membandingkan kata-kata dalam data testing tersebut dengan probabilitas yang telah diketahui dari data training, guna menilai kinerja model.

#### b. *Support Vector Machine*

Proses klasifikasi menggunakan *Support Vector Machine* melibatkan dua set data: data training dan data testing. Data training digunakan untuk melatih model *Support Vector Machine* agar dapat membangun sebuah *hyperplane* yang optimal yang memisahkan kelas-kelas data. Setelah model terlatih, data testing digunakan untuk mengevaluasi seberapa baik model *Support Vector Machine* dapat mengklasifikasikan data baru berdasarkan *hyperplane* yang telah dibentuk. Tujuannya adalah untuk memverifikasi akurasi dan efektivitas model dalam melakukan klasifikasi. Parameter yang digunakan pada algoritma *Support Vector Machine* ini dengan melakukan perubahan pada parameter *Kernel*. *Kernel* yang digunakan meliputi *Kernel Linear*, *Polynomial* dan *Radial Basis Function (RBF)*.

Pelabelan yang dilakukan pada penelitian analisis sentimen aplikasi TikTok ini dibagi menjadi 2 kelas, yakni kelas negatif dan positif. Adapun hasil pelabelan kelas sentimen pada data yang digunakan dapat dilihat pada tabel 3.3.

Tabel 3.3. Hasil Pelabelan Kelas Sentimen

<b>Kelas</b>	<b>Jumlah</b>
Negatif	2579
Positif	2421

Berdasarkan tabel 3.3. diketahui ulasan tertinggi terdapat pada kelas Negatif dengan jumlah 2.579 dan kelas Positif berjumlah 2.421. Kelas sentimen Negatif merupakan ulasan pengguna yang tidak puas atau kecewa terhadap aplikasi TikTok, sedangkan Kelas sentimen Positif merupakan ulasan pengguna yang puas terhadap aplikasi TikTok.

### 3.4.3 Uji Coba

Untuk mengetahui kinerja algoritma *Naive Bayes* dan *Support Vector Machine*, penelitian ini menggunakan *Confusion Matrix*. *Confusion Matrix* menggunakan tabel matriks untuk menunjukkan hasil klasifikasi data testing berdasarkan data training. Berikut ini adalah tabel *Confusion Matrix* yang memiliki empat kombinasi berbeda antara nilai prediksi dan nilai aktual.

Tabel 3.4. Confusion Matrix

Data Aktual	Data Prediksi	
	Positif	Negatif
Positif	True Positive (TP)	False Negative (FN)
Negatif	False Positive (FP)	True Negative (TN)

Keterangan:

- *True Positive* (TP) adalah data positif yang diprediksi benar.
- *True Negative* (TN) adalah data negatif yang diprediksi benar.
- *False Positive* (FP) adalah data negatif tapi diprediksi sebagai data positif.
- *False Negative* (FN) adalah data positif tapi diprediksi sebagai data negatif.

Adapun yang umum digunakan dalam penilaian dan evaluasi model klasifikasi sebagai berikut:

- a. Akurasi digunakan untuk mengukur seberapa sering model prediksi benar secara keseluruhan. Ini adalah rasio prediksi benar (baik positif maupun negatif) terhadap total prediksi. Persamaan yang digunakan yakni sebagai berikut:

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

b. *Precision* digunakan untuk mengukur seberapa banyak dari prediksi positif yang benar-benar positif. Ini adalah rasio prediksi positif benar terhadap total prediksi positif. Persamaan yang digunakan yakni sebagai berikut:

$$Precision\ Positif = \frac{TP}{TP + FP}$$

$$Precision\ Negatif = \frac{TN}{TN + FN}$$

c. *Recall* digunakan untuk mengukur seberapa baik model menangkap semua contoh positif yang sebenarnya. Ini adalah rasio prediksi positif benar terhadap total contoh positif sebenarnya. Persamaan yang digunakan yakni sebagai berikut:

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

$$Recall\ Negatif = \frac{TN}{TN + FP}$$

Skenario uji coba performa algoritma bertujuan untuk menghasilkan model terbaik saat proses training sampai dengan ketika model diimplementasikan. Pada analisis sentimen ulasan aplikasi TikTok, pengujian dilakukan sebanyak 3 kali dengan membagi data training dan testing dengan porsi yang berbeda-beda, serta dengan mengubah parameter kernel algoritma *Support Vector Machine*. Parameter *kernel* pada Algoritma *Support Vector Machine* yang dilakukan pengujian antara lain adalah *Kernel Linear*, *Polynomial* dan *Radial Basis Function*.

Tabel 3.5. Split Data Training dan Testing

<b>Percobaan</b>	<b>Training</b>	<b>Testing</b>
Pertama	80%	20%
Kedua	70%	30%
Ketiga	60%	40%

## BAB IV

### PEMROSESAN DATA AWAL

#### 4.1. Pengumpulan Data

Data ulasan atau komentar aplikasi TikTok yang digunakan dalam penelitian ini sebanyak 5.000 data dari jumlah total data 313.186, hal ini dikarenakan keterbatasan *resource* yang digunakan pada penelitian ini. Berikut sebagian data ulasan / komentar TikTok yang telah dikumpulkan dapat dilihat pada tabel 4.1.

Tabel 4.1. Data Ulasan / Komentar App TikTok

No	Komentar	Skor	Tanggal
1	baguuzzz bgtttz aks sukaaaak 🤔 🤔 ❤️ 🤔 ❤️ 🤔	5	2023-12-31 23:59:24
2	Aplikasi gak jelas, padahal enggak pernah ngirim/ngetik aneh aneh malah ditanggihkan nungguin lama tetep gak bisa, gak bisa ngirim pesan, enggak bisa terima pesan tolong ya aplikasi nya di perbaiki lagi	1	2023-12-31 23:58:40
3	Skrang Tiktok Terlalu sensitif biar kata tidak mengandung kata Negatif... Biar pun bahasa daerah yang kata membangun Semangat... Sedikit-Sediki di hapus.. Biar pun di Banding" tetap aja di hapus....Tolong di perhatikan dan belajar bahasa daerah	1	2023-12-31 23:58:33
4	Keren	4	2023-12-31 23:52:35
5	Terlalu banyak pelanggaran yg di berikan Padahal vidio buat sendiri tapi kena pelanggaran sampai 3 point Tolong di perbaiki sistem nya	1	2023-12-31 23:52:09
6	👍👍👍👍👍👍👍👍	1	2023-06-28 20:25:38
7	Baguss bngtt 😊	5	2023-06-28 20:25:10
8	Aplikasi ini sangat bagus...	5	2023-06-28 20:24:47
9	Tidak bisa di download	1	2023-06-28 20:24:07

#### 4.2. Case Folding dan Cleaning

Pada proses ini dilakukan tahapan, tahapan pertama dilakukan dengan mengubah huruf besar menjadi huruf kecil, hal ini sangat perlu untuk dilakukan karena akan berdampak terhadap proses *pre-processing* selanjutnya. Selain itu

pada tahapan ini pula dilakukan proses pembersihan data dengan menghapus karakter atau kata-kata yang tidak relevan atau berisik dari teks yang dapat mencakup penghapusan tanda baca, angka, URL, dan simbol atau karakter khusus lainnya yang tidak memberikan nilai informasi yang berarti untuk analisis. Berikut proses case folding dan cleaning pada data ulasan/komentar aplikasi TikTok pada Gambar 4.1.

Komentar	case folding	cleaning
baguuzzz bgtttz aks sukaaaak 🤔🤔🤔🤔🤔	baguuzzz bgtttz aks sukaaaak	baguuzzz bgtttz aks sukaaaak
Aplikasi gak jelas, padahal enggak pernah ngiri...	aplikasi gak jelas padahal enggak pernah ngiri...	aplikasi gak jelas padahal enggak pernah ngiri...
Skrang Tiktok Terlalu sensitif biar kata tidak...	skrang tiktok terlalu sensitif biar kata tidak...	skrang tiktok terlalu sensitif biar kata tidak...
Keren	keren	keren
Pembela zionis	pembela zionis	pembela zionis

Gambar 4.1. Hasil Case Folding dan Cleaning

Berikut *source code* 4.1 yang digunakan untuk proses case folding dan cleaning:

```
# Load data dari Excel
data = pd.read_excel('/content/drive/MyDrive/tiktok_reviews -
data yang digunakan.xlsx', usecols=['Komentar'])
data.head()
data['cf'] = data['Komentar'].str.lower()
data['cf'] = data['cf'].str.replace(r'\.', ' ', regex=True)
data['cf'] = data['cf'].str.replace(r'\,', ' ', regex=True)
data['cf'] = data['cf'].astype(str).replace(r'^[a-zA-z0-9\s]', ' ', regex=True)

def clean_korpus(text):
    text = re.sub("([@#][A-Za-z0-9]+)|(\w+:\//\//S+)|([^\x00-\x7F]+)", " ", text)
    text = re.sub(r"\d+", "", text)
    text = text.strip()
    text = re.sub('\s+', ' ', text)
    text = re.sub(r"\b[a-zA-Z]\b", "", text)
    return text

data['clean'] = data['cf'].apply(clean_korpus)
data.head()
```

### 4.3. Tokenisasi

Pada tahapan ini kalimat yang telah dilakukan *case folding* dan *cleaning* di pecah menjadi masing-masing kata. Hal ini bertujuan untuk mempermudah saat proses normalisasi serta proses *stopword removal*. Hasil dari proses tokenisasi terdapat pada Gambar 4.2.

clean	tokenisasi
baguuzzz bgtttz aks sukaaaak	[baguuzzz, bgtttz, aks, sukaaaak]
aplikasi gak jelas padahal enggak pernah ngiri...	[aplikasi, gak, jelas, padahal, enggak, pernah...
skrang tiktok terlalu sensitif biar kata tidak...	[skrang, tiktok, terlalu, sensitif, biar, kata...
keren	[keren]
pembela zionis	[pembela, zionis]

Gambar 4.2. Hasil Tokenisasi

Berikut *source code* 4.2 yang digunakan untuk proses tokenisasi.

```
import nltk
from nltk.tokenize import word_tokenize

def word_tokenize_wrapper(text):
    return word_tokenize(text)

data['tokenisasi'] =
data['clean'].apply(word_tokenize_wrapper)
data.head(16)
```

### 4.4. Normalisasi

Normalisasi pada penelitian ini dilakukan dengan mengubah kata tidak baku menjadi kata baku sesuai dengan Kamus Besar Bahasa Indonesia (KBBI) dengan dicocokkan dengan dataset yang telah dimiliki (*kalimat tidak formal* → *kalimat formal*). Kelemahan pada proses normalisasi ini terdapat pada kamus bahasa yang terbatas sehingga akan ada kemungkinan kata yang gagal dalam proses perbaikan kata tidak baku menjadi kata baku hal ini karena banyaknya

kalimat tidak baku yang identik bahasa daerah. Hasil dari proses normalisasi terdapat pada Gambar 4.3.

tokenisasi	normalisasi
[baguuzzz, bgtttz, aks, sukaaaak]	[baguuzzz, bgtttz, aks, sukaaaak]
[aplikasi, gak, jelas, padahal, enggak, pernah...]	[aplikasi, tidak, jelas, padahal, tidak, perna...]
[skrang, tiktok, terlalu, sensitif, biar, kata...]	[sekarang, tiktok, terlalu, sensitif, biar, ka...]
[keren]	[keren]
[pembela, zionis]	[bela, zionis]

Gambar 4.3. Hasil Normalisasi

Berikut *source code* 4.3 yang digunakan untuk proses normalisasi.

```
kamus_alay =
pd.read_csv('/content/drive/MyDrive/new_kamusalay.csv',
encoding = 'latin-1')

alay_dict = dict(zip(kamus_alay['slang'],
kamus_alay['formal']))

# Creating dictionary from DataFrame
alay_dict = dict(zip(kamus_alay['slang'],
kamus_alay['formal']))

def normalized_term(document):
    failed_words = []
    processed_text = []
    for term in document:
        try:
            if term in alay_dict:
                processed_text.append(alay_dict[term])
            else:
                processed_text.append(term)
                failed_words.append(term) # Saving words
that failed to convert
        except KeyError:
            processed_text.append(term)
            failed_words.append(term) # Saving words that
failed to convert
    return processed_text, failed_words
```

```
# Normalisasi
data[['normalisasi', 'gagal_normalisasi']] =
pd.DataFrame(data['tokenisasi'].apply(normalized_term).tolist
(), index=data.index)
```

#### 4.5. Stopword Removal

Pada proses ini dilakukan penghapusan atau pembuangan kata penghubung seperti “di”, “yang”, “ke” dan lain-lainnya. Hasil dari proses Stopword Removal terdapat pada Gambar 4.4.

normalisasi	stopwords
[baguuzzz, bgtttz, aks, sukaaaak]	[baguuzzz, bgtttz, aks, sukaaaak]
[aplikasi, tidak, jelas, padahal, tidak, perna...]	[aplikasi, ngirimnetik, aneh, aneh, ditangguh...]
[sekarang, tiktok, terlalu, sensitif, biar, ka...]	[tiktok, sensitif, biar, mengandung, negatif, ...]
[keren]	[keren]
[bela, zionis]	[bela, zionis]

Gambar 4.4. Hasil Stopword Removal

Berikut *source code* 4.4 yang digunakan untuk proses stopwords removal.

```
import nltk
from nltk.corpus import stopwords

nltk.download('stopwords')
list_stopwords = stopwords.words('indonesian')
def stopwords_removal(words):
    return [word for word in words if word not in
list_stopwords]

data['stopwords'] =
data['normalisasi'].apply(stopwords_removal)
data.head(20)
```

#### 4.6. Stemming

Pada proses ini dilakukan perbaikan kata dengan menghilangkan kata imbuhan pada kata yang terdapat pada ulasan/komentar aplikasi TikTok. Sebelum proses stemming dilakukan, kalimat yang masih dalam bentuk array (tokenisasi) dilakukan penggabungan terlebih dahulu, kemudian dapat dilanjutkan proses *stemming*. Hasil *stemming* juga memiliki kelemahan pada kata-kata berimbuhan yang memiliki huruf ganda atau kesalahan penulisan. Contohnya adalah kata "Dkecewaknn" yang gagal dalam proses *stemming* pada bagian kata "knn" dan "D". Hasil dari proses Stemming terdapat pada Gambar 4.5.

stopwords	stemmed
[baguuzzz, bgtttz, aks, sukaaaaak]	baguuzzz bgtttz aks sukaaaaak
[aplikasi, ngirimnetik, aneh, aneh, ditangguh...]	aplikasi ngirimnetik aneh aneh tangguh tunggu...
[tiktok, sensitif, biar, mengandung, negatif, ...]	tiktok sensitif biar kandung negatif biar baha...
[keren]	keren
[bela, zionis]	bela zionis

Gambar 4.5. Hasil Stemming

Berikut *source code* 4.5 yang digunakan untuk proses *stemming*.

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

# Inisialisasi stemmer Bahasa Indonesia
factory = StemmerFactory()
stemmer = factory.create_stemmer()

def stem_text(text_list):
    # Pastikan semua elemen adalah string kecuali float dan
    None
    filtered_text = [str(text) for text in text_list if not
isinstance(text, float) and text is not None]
    # Gabungkan list yang telah disaring menjadi string
    joined_text = ' '.join(filtered_text)
```

```

# Terapkan stemming pada teks gabungan
stemmed_text = stemmer.stem(joined_text)
return stemmed_text

# Membaca data
data['stemmed'] = data['stopwords'].apply(stem_text)
data.head(20)

```

Setelah proses *stemming* dilakukan, pada proses ini juga dilakukan penyimpanan data hasil *stemming* yang disimpan dalam bentuk excel. Hal ini bertujuan untuk mempermudah tahapan selanjutnya yakni *lexicon* kalimat.

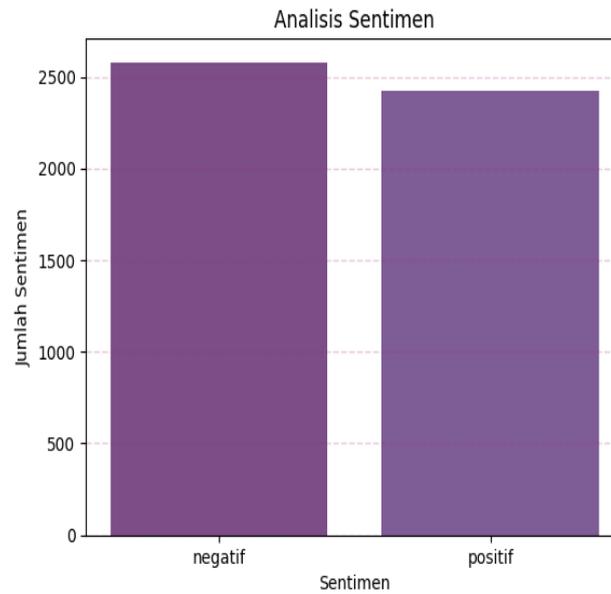
#### 4.7. Lexicon dan Labeling

Setelah dilakukan tahapan *preprocessing* hingga *stemming*, selanjutnya dilakukan dengan pencocokan kata pada *dataset lexicon* berbahasa Indonesia yang telah disiapkan untuk menghasilkan nilai *polarity* kalimat yang akan digunakan untuk penentuan sentimen positif dan negatif. Hasil dari proses *Lexicon dan Labeling* terdapat pada Gambar 4.6.

	stemmed	polarity_text	sentimen
	baguuzzz bgtttz aks sukaaaak	0	positif
	aplikasi ngirimnetik aneh aneh tangguh tunggu...	-4	negatif
	tiktok sensitif biar kandung negatif biar baha...	-14	negatif
	keren	0	positif

Gambar 4.6. Hasil Lexicon dan Labeling

Adapun perbandingan jumlah dataset setelah dilakukan *preprocessing* dan *lexicon* yakni sentimen negatif sebanyak 2579 (51.58%) dan sentimen positif sebanyak 2421 (48.42%) dengan total keseluruhan 5.000 data. Persentase data tersebut dapat dilihat pada Gambar 4.7.



Gambar 4.7. Grafik Persentasi Perbandingan Dataset

#### 4.8. Pembobotan Term Frequency-Inverse Document Frequency

Selanjutnya, pembobotan kata dilakukan dengan *Term Frequency-Inverse Document Frequency* untuk mengetahui frekuensi kemunculan term pada setiap dokumen. *Term Frequency* menghitung nilai probabilitas kemunculan term dalam setiap dokumen, seperti pada dokumen D1 hingga D5. DF menghitung jumlah kemunculan term dari keseluruhan dokumen D1 hingga D5, sedangkan perhitungan *Term Frequency-Inverse Document Frequency* menggunakan metode log dengan membagi total dokumen dengan jumlah DF, yaitu  $=\text{LOG}(\text{jumlah dokumen} / \text{jumlah DF})$ . *Term Frequency-Inverse Document Frequency* menghitung bobot kata pada masing-masing dokumen dengan mengalikan hasil *Inverse Document Frequency* dengan jumlah setiap term dalam dokumen. Berikut adalah hasil pembobotan kata *Term Frequency-Inverse Document Frequency* pada Tabel 4.2.

Tabel 4.2. Contoh Perhitungan Term Frequency-Inverse Document Frequency

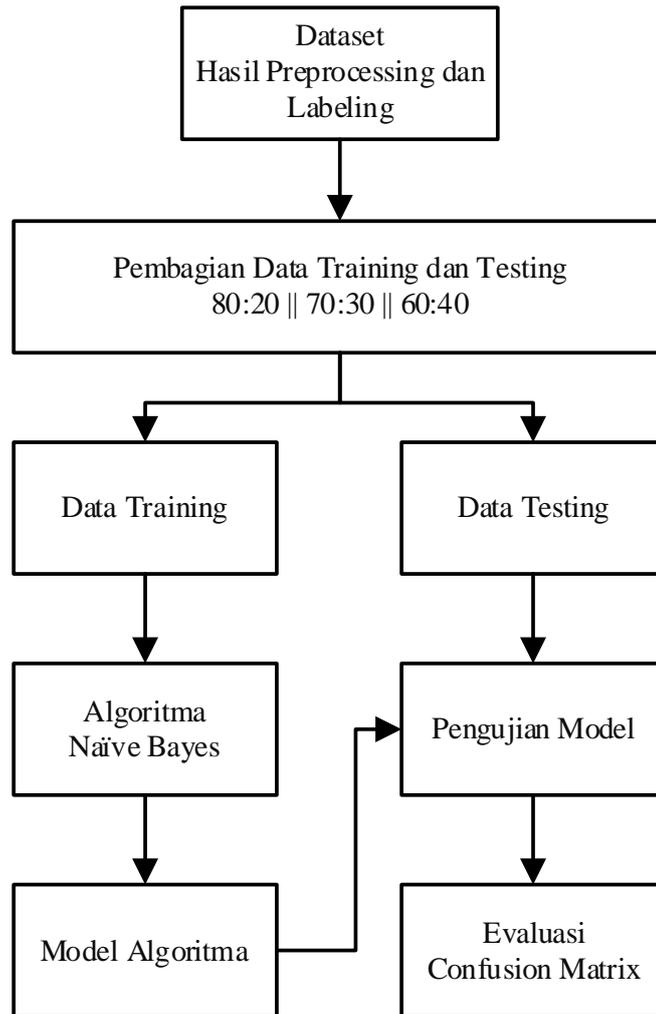
Term	Term Frequency (TF)					DF	IDF	Term Frequency x Inverse Document Frequency TF x IDF				
	D1	D2	D3	D4	D5			D1	D2	D3	D4	D5
Aplikasi	1	0	0	0	0	1	0,70	0,70	0,00	0,00	0,00	0,00
Ini	1	0	0	0	0	1	0,70	0,70	0,00	0,00	0,00	0,00
Sangat	1	0	1	0	0	2	0,40	0,40	0,00	0,40	0,00	0,00
Bagus	1	0	0	0	0	1	0,70	0,70	0,00	0,00	0,00	0,00
Bermanfaat	1	0	0	0	0	1	0,70	0,70	0,00	0,00	0,00	0,00
Terlalu	0	1	0	0	0	1	0,70	0,00	0,70	0,00	0,00	0,00
Banyak	0	1	0	0	0	1	0,70	0,00	0,70	0,00	0,00	0,00
Iklan	0	1	0	0	0	1	0,70	0,00	0,70	0,00	0,00	0,00
Mengganggu	0	1	0	0	0	1	0,70	0,00	0,70	0,00	0,00	0,00
Konten	0	0	1	0	0	1	0,70	0,00	0,00	0,70	0,00	0,00
Disediakan	0	0	1	0	0	1	0,70	0,00	0,00	0,70	0,00	0,00
Menghibur	0	0	1	0	0	1	0,70	0,00	0,00	0,70	0,00	0,00
Sering	0	0	0	1	0	1	0,70	0,00	0,00	0,00	0,70	0,00
Crash	0	0	0	1	0	1	0,70	0,00	0,00	0,00	0,70	0,00
Lambat	0	0	0	1	0	1	0,70	0,00	0,00	0,00	0,70	0,00
Fitur-fiturnya	0	0	0	0	1	1	0,70	0,00	0,00	0,00	0,00	0,70
Menarik	0	0	0	0	1	1	0,70	0,00	0,00	0,00	0,00	0,70
Inovatif	0	0	0	0	1	1	0,70	0,00	0,00	0,00	0,00	0,70

## BAB V

### KLASIFIKASI NAIVE BAYES

#### 5.1. Algoritma Naive Bayes

Algoritma *Naive Bayes* adalah salah satu metode klasifikasi yang berbasis pada Teorema Bayes dengan asumsi independensi antar fitur. Proses klasifikasi menggunakan algoritma ini melibatkan beberapa tahapan utama yang sistematis. Sebelum data digunakan dalam model, data tersebut harus dilakukan pre-processing. Proses ini melibatkan beberapa langkah penting seperti *case folding*, *tokenisasi*, *penghapusan stopwords* dan *stemming*. Setelah data dilakukan pre-processing, langkah selanjutnya adalah pembobotan kata menggunakan metode *Term Frequency-Inverse Document Frequency*. *Term Frequency-Inverse Document Frequency* memberikan bobot pada kata-kata berdasarkan frekuensi kemunculannya dalam dokumen tertentu dan seberapa umum kata tersebut muncul dalam keseluruhan kumpulan dokumen. Hal ini membantu dalam mengidentifikasi kata-kata yang lebih penting dan relevan untuk klasifikasi. Kemudian dibagi menjadi dua bagian pertama sebagai data training dan kedua sebagai data testing. Data training digunakan untuk melatih model, sementara data testing digunakan untuk mengevaluasi kinerja model. Perbandingan yang digunakan untuk data training dan data testing pada penelitian ini yakni 80 : 20. Adapun alur dari proses klasifikasi menggunakan algoritma naive bayes dapat di lihat pada Gambar 5.1.



Gambar 5.1. Alur Klasifikasi Algoritma Naive Bayes

## 5.2. Uji Coba Algoritma Naïve Bayes

Proses klasifikasi data teks komentar/ulasan aplikasi TikTok menggunakan *Naive Bayes* dibagi menjadi 2 bagian yakni data training dan data testing. Data yang digunakan merupakan data yang telah dilakukan *preprocessing* dan pelabelan. Percobaan dilakukan sebanyak 3 kali, dengan pembagian dataset yang dapat di lihat pada Tabel 5.1.

Tabel 5.1. Split Data untuk Algoritma Naive Bayes

<b>Percobaan</b>	<b>Training</b>	<b>Testing</b>
Pertama	80%	20%
Kedua	70%	30%
Ketiga	60%	40%

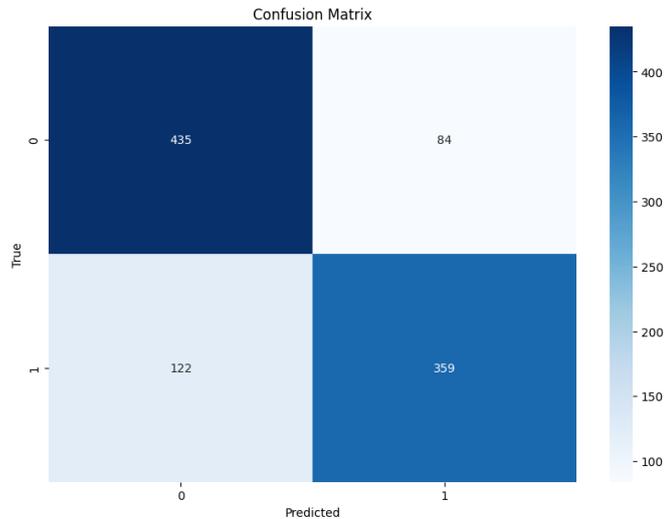
### 5.2.1. Percobaan Naïve Bayes Menggunakan Dataset 80:20

Perbandingan dataset yang digunakan menggunakan perbandingan 80% untuk data training dan 20% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 5.2.

Tabel 5.2. Pembagian Jumlah Data 80:20

<b>Kelas</b>	<b>Jumlah</b>	<b>Training 80%</b>	<b>Testing 20%</b>
Negatif	2579	2060	519
Positif	2421	1940	481
<b>Total</b>	<b>5000</b>	<b>4000</b>	<b>1000</b>

Dari hasil percobaan yang dilakukan, algoritma *Naive Bayes* memperoleh hasil akurasi yang cukup baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 79%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Naive Bayes* dengan menggunakan 1000 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 5.2.



Gambar 5.2. Confusion Matrix Algoritma Naive Bayes 80:20

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 435 dan yang terprediksi positif (salah) sebanyak 84. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 122 dan yang terprediksi positif (benar) sebanyak 359. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 5.3.

Tabel 5.3. Classification Report Algoritma Naive Bayes 80:20

Class	Precision	Recall	F1-Score
Negatif	0.78 (78%)	0.84 (84%)	0.81 (81%)
Positif	0.81 (81%)	0.75 (75%)	0.78 (78%)
Accuracy		0.79 (79%)	

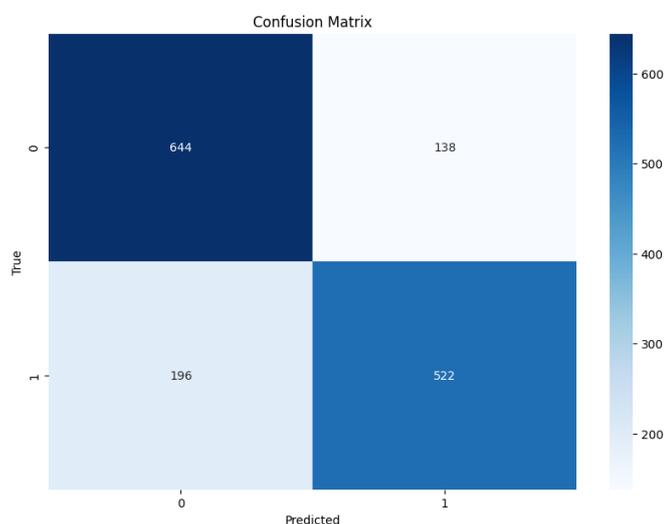
### 5.2.2. Percobaan Naïve Bayes Menggunakan Dataset 70:30

Perbandingan dataset yang digunakan menggunakan perbandingan 70% untuk data training dan 30% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 5.4.

Tabel 5.4. Pembagian Jumlah Data 70:30

Kelas	Jumlah	Training 70%	Testing 30%
Negatif	2579	1797	782
Positif	2421	1703	718
<b>Total</b>	<b>5000</b>	<b>3500</b>	<b>1500</b>

Dari hasil percobaan yang dilakukan, algoritma *Naive Bayes* memperoleh hasil akurasi yang cukup baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 78%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Naive Bayes* dengan menggunakan 1500 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 5.3.



Gambar 5.3. Confusion Matrix Algoritma Naive Bayes 70:30

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 644 dan yang terprediksi positif (salah) sebanyak 138. Sedangkan data ulasan positif yang terprediksi negatif

(salah) sebanyak 196 dan yang terprediksi positif (benar) sebanyak 522. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 5.5.

Tabel 5.5. Classification Report Algoritma Naive Bayes 70:30

Class	Precision	Recall	F1-Score
Negatif	0.77 (77 %)	0.82 (82 %)	0.79 (79 %)
Positif	0.79 (79 %)	0.73 (73 %)	0.76 (76%)
Accuracy		0.78 (78%)	

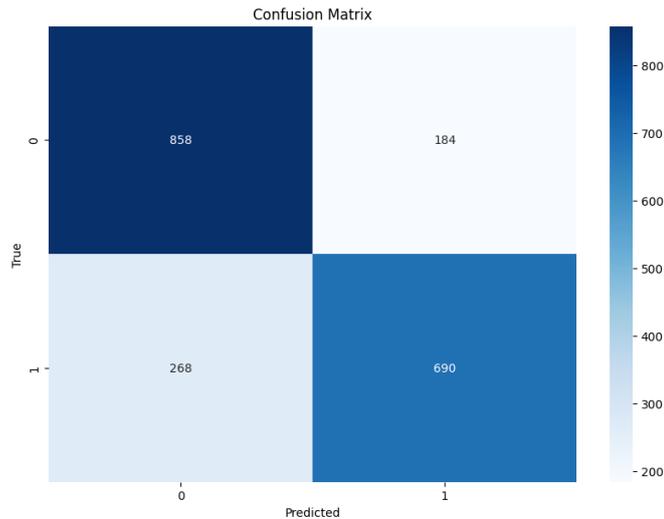
### 5.2.3. Percobaan Naïve Bayes Menggunakan Dataset 60:40

Perbandingan dataset yang digunakan menggunakan perbandingan 60% untuk data training dan 40% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 5.6.

Tabel 5.6. Pembagian Jumlah Data 60:40

Kelas	Jumlah	Training 60%	Testing 40%
Negatif	2579	1537	1042
Positif	2421	1463	958
<b>Total</b>	<b>5000</b>	<b>3000</b>	<b>2000</b>

Dari hasil percobaan yang dilakukan, algoritma *Naive Bayes* memperoleh hasil akurasi yang cukup baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 77%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Naive Bayes* dengan menggunakan 2000 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 5.4.



Gambar 5.4. Confusion Matrix Algoritma Naive Bayes 60:40

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 858 dan yang terprediksi positif (salah) sebanyak 184. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 268 dan yang terprediksi positif (benar) sebanyak 690. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 5.7.

Tabel 5.7. Classification Report Algoritma Naive Bayes 60:40

Class	Precision	Recall	F1-Score
Negatif	0.76 (76 %)	0.82 (82 %)	0.79 (79 %)
Positif	0.79 (79 %)	0.72 (72 %)	0.75 (75 %)
Accuracy		0.77(77%)	

### 5.3. Performa Algoritma Naive Bayes

Berdasarkan hasil uji coba menggunakan 3 percobaan maka rata rata performa *Naive Bayes* dapat dilihat pada Tabel 5.8.

Tabel 5.8. Performa Naive Bayes

NO	PERCOBAAN	DATA	PERFORMA NAÏVE BAYES						
			<i>Accuracy</i>	<i>Precision (Negatif)</i>	<i>Precision (Positif)</i>	<i>Recall (Negatif)</i>	<i>Recall (Positif)</i>	<i>f1-Score (Negatif)</i>	<i>f1-Score (Positif)</i>
1	Percobaan Pertama	80:20	<b>79%</b>	<b>78%</b>	<b>81%</b>	<b>84%</b>	<b>75%</b>	<b>81%</b>	<b>78%</b>
2	Percobaan Kedua	70:30	78%	77%	79%	82%	73%	79%	76%
3	Percobaan Ketiga	60:40	77%	76%	79%	82%	72%	79%	75%
<b>Rata Rata</b>			78%	77%	80%	<b>83%</b>	73%	80%	76%

Pada Tabel 5.8. diketahui nilai *accuracy*, *precision (negatif)* dan *precision (positif)* tertinggi di peroleh pada percobaan pertama yaitu 79%, 78% dan 81%, sedangkan nilai *recall (negatif)*, *recall (positif)*, *f1-score (negatif)* dan *f1-score (positif)* tertinggi di peroleh pada percobaan pertama yaitu 84%, 75%, 81% dan 78%. Rata rata performa *Naive Bayes* adalah *accuracy* sebesar 78%, nilai *precision (negatif)* 77%, nilai *precision (positif)* 80%, nilai *recall (negatif)* 83%, nilai *recall (positif)* 73%, nilai *f1-score (negatif)* 80% dan *f1-score (positif)* 76%.

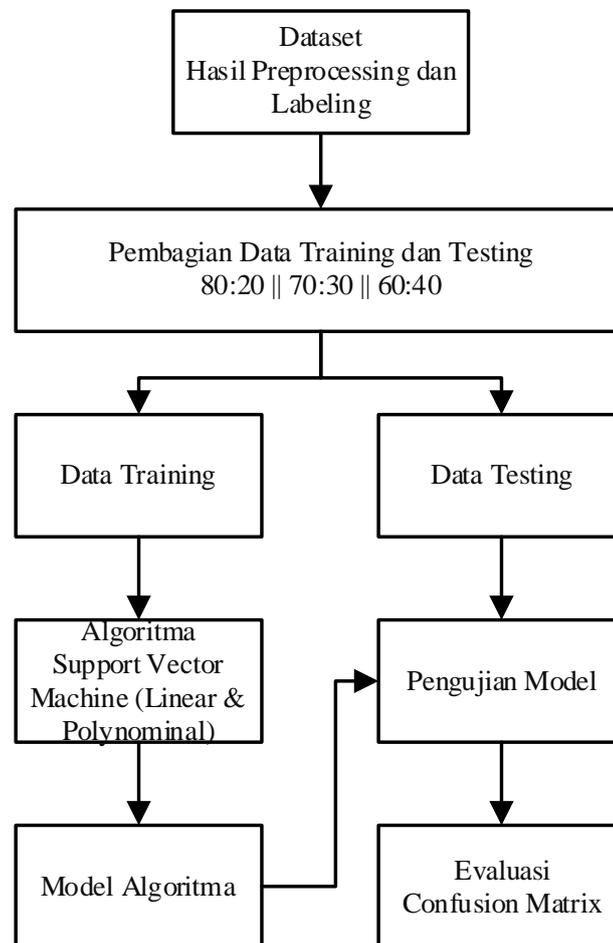
## BAB VI

### KLASIFIKASI SUPPORT VECTOR MACHINE

#### 6.1. Algoritma Support Vector Machine

Algoritma *Support Vector Machine* adalah salah satu metode klasifikasi yang kuat dan populer dalam *machine learning*. Algoritma ini bekerja dengan mencari *hyperplane* terbaik yang memisahkan kelas-kelas dalam data. Sebelum data digunakan dalam model, data tersebut harus dilakukan pre-processing. Proses ini melibatkan beberapa langkah penting seperti *case folding*, *tokenisasi*, *penghapusan stopwords* dan *stemming*. Setelah data dilakukan pre-processing, langkah selanjutnya adalah pembobotan kata menggunakan metode *Term Frequency-Inverse Document Frequency*. *Term Frequency-Inverse Document Frequency* memberikan bobot pada kata-kata berdasarkan frekuensi kemunculannya dalam dokumen tertentu dan seberapa umum kata tersebut muncul dalam keseluruhan kumpulan dokumen. Hal ini membantu dalam mengidentifikasi kata-kata yang lebih penting dan relevan untuk klasifikasi. Kemudian dibagi menjadi dua bagian pertama sebagai data training dan kedua sebagai data testing. Data training digunakan untuk melatih model, sementara data testing digunakan untuk mengevaluasi kinerja model. Perbandingan yang digunakan untuk data training dan data testing pada penelitian ini yakni 80 : 20. *Hyperplane* ini adalah garis atau permukaan yang memaksimalkan margin antara dua kelas yang berbeda. Algoritma *Support Vector Machine* menggunakan titik-titik data yang paling dekat dengan *hyperplane* (disebut *support vectors*) untuk menentukan posisi *hyperplane*. Adapun alur dari proses

klasifikasi menggunakan algoritma *Support Vector Machine* dapat di lihat pada Gambar 6.1.



Gambar 6.1. Alur Klasifikasi Algoritma Support Vector Machine

## 6.2. Uji Coba Algoritma Support Vector Machine

Proses klasifikasi data teks komentar atau ulasan aplikasi TikTok menggunakan *Support Vector Machine* dibagi menjadi 2 bagian yakni data training dan data testing. Data yang digunakan merupakan data yang telah dilakukan preprocessing dan pelabelan. Percobaan dilakukan sebanyak 3 kali untuk beberapa parameter *Kernel Support Vector Machine* diantaranya *Kernel*

*Linear, Polynomial dan Radial Basis Function (RBF)*. Adapun pembagian dataset dapat dilihat pada Tabel 6.1.

Tabel 6.1. Split Data untuk Algoritma Support Vector Machine

Percobaan	Training	Testing	Kernel
Pertama	80%	20%	- <i>Linear</i>
Kedua	70%	30%	- <i>Polynomial</i>
Ketiga	60%	40%	- <i>Radial Basis Function</i>

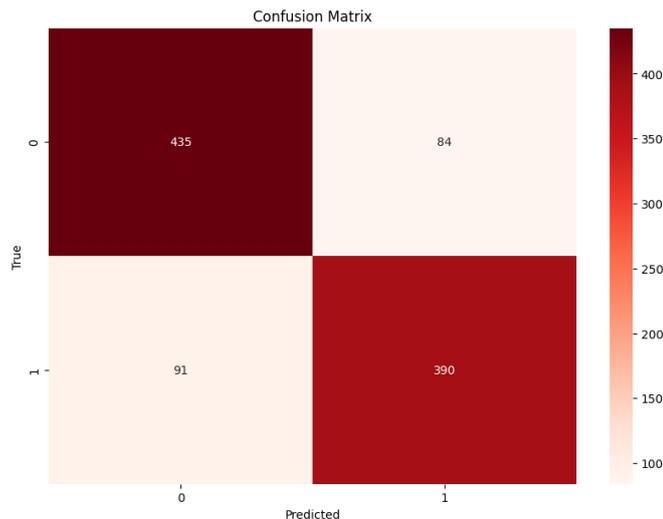
### 6.2.1. Percobaan Support Vector Machine Kernel Linear Menggunakan Dataset 80:20

Perbandingan dataset yang digunakan menggunakan perbandingan 80% untuk data training dan 20% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 6.2.

Tabel 6.2. Pembagian Jumlah Data Support Vector Machine Linear 80:20

Kelas	Jumlah	Training 80%	Testing 20%
Negatif	2579	2060	519
Positif	2421	1940	481
<b>Total</b>	<b>5000</b>	<b>4000</b>	<b>1000</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 82%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support Vector Machine* dengan menggunakan 1000 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 6.2.



Gambar 6.2. Confusion Matrix Algoritma Support Vector Machine Linear 80:20

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 435 dan yang terprediksi positif (salah) sebanyak 84. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 91 dan yang terprediksi positif (benar) sebanyak 390. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 6.3.

Tabel 6.3. Classification Report Algoritma Support Vector Machine Linear 80:20

Class	Precision	Recall	F1-Score
Negatif	0.83 (83%)	0.84 (84%)	0.83 (83%)
Positif	0.82 (82%)	0.81 (81%)	0.82 (82%)
Accuracy		0.82 (82%)	

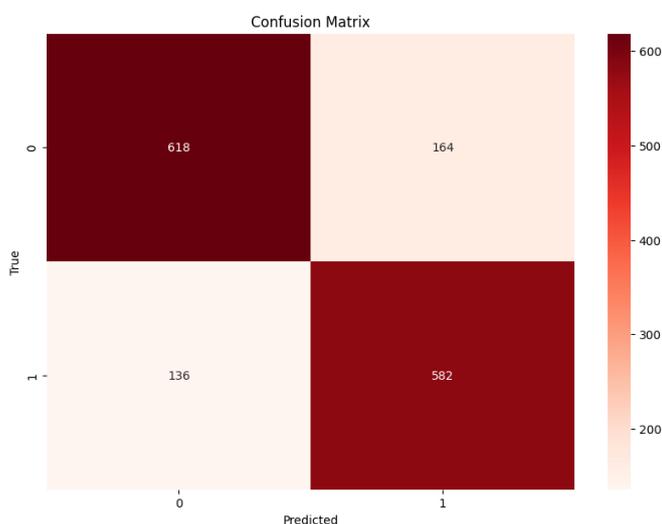
### 6.2.2. Percobaan Support Vector Machine Kernel Linear Menggunakan Dataset 70:30

Perbandingan dataset yang digunakan menggunakan perbandingan 70% untuk data training dan 30% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 6.4.

Tabel 6.4. Pembagian Jumlah Data Support Vector Machine Linear 70:30

Kelas	Jumlah	Training 70%	Testing 30%
Negatif	2579	1797	782
Positif	2421	1703	718
<b>Total</b>	<b>5000</b>	<b>3500</b>	<b>1500</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 80%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support Vector Machine* dengan menggunakan 1500 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 6.3.



Gambar 6.3. Confusion Matrix Algoritma Support Vector Machine Linear 70:30

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 618 dan yang terprediksi positif (salah) sebanyak 164. Sedangkan data ulasan positif yang terprediksi negatif

(salah) sebanyak 136 dan yang terprediksi positif (benar) sebanyak 582. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 6.5.

Tabel 6.5. Classification Report Algoritma Support Vector Machine Linear 70:30

Class	Precision	Recall	F1-Score
Negatif	0.82 (82%)	0.79 (79%)	0.80 (80 %)
Positif	0.78 (78%)	0.81 (81%)	0.80 (80 %)
Accuracy		0.80 (80 %)	

### 6.2.3. Percobaan Support Vector Machine Kernel Linear Menggunakan Dataset 60:40

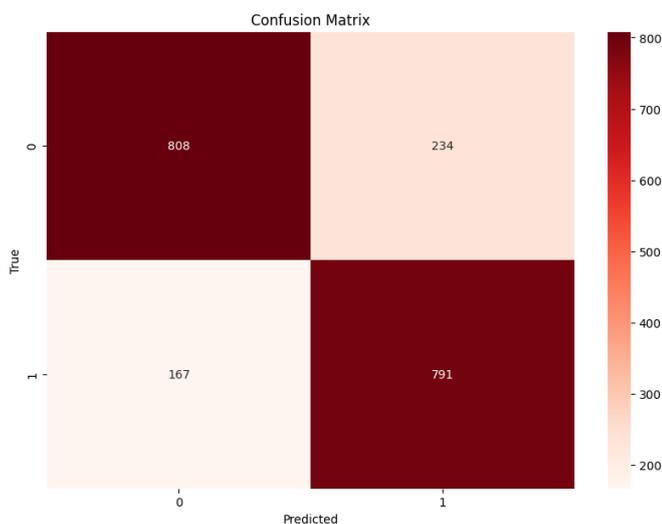
Perbandingan dataset yang digunakan menggunakan perbandingan 60% untuk data training dan 40% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 6.6.

Tabel 6.6. Pembagian Jumlah Data Support Vector Machine Linear 60:40

Kelas	Jumlah	Training 70%	Testing 30%
Negatif	2579	1537	1042
Positif	2421	1463	958
<b>Total</b>	<b>5000</b>	<b>3000</b>	<b>2000</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 80%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support Vector Machine* dengan menggunakan 2000 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma

tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 6.4.



Gambar 6.4. Confusion Matrix Algoritma Support Vector Machine Linear 60:40

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 808 dan yang terprediksi positif (salah) sebanyak 234. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 167 dan yang terprediksi positif (benar) sebanyak 791. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 6.7.

Tabel 6.7. Classification Report Algoritma Support Vector Machine Linear 60:40

Class	Precision	Recall	F1-Score
Negatif	0.83 (83 %)	0.78 (78 %)	0.80 (80 %)
Positif	0.77 (77 %)	0.83 (83 %)	0.80 (80 %)
Accuracy		0.80 (80 %)	

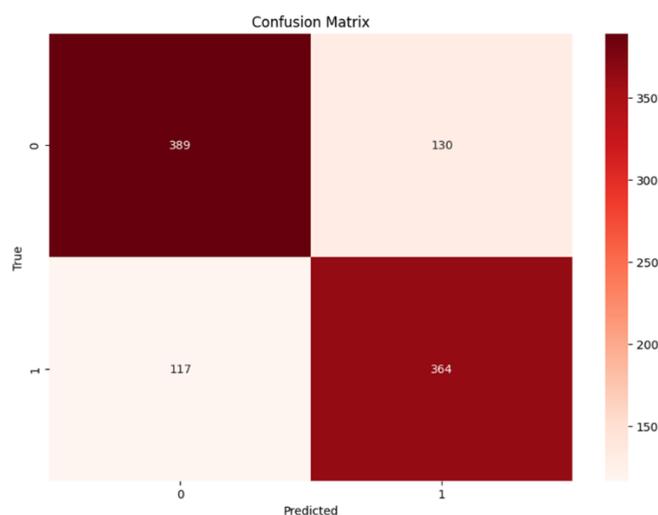
#### 6.2.4. Percobaan Support Vector Machine Kernel Polynomial Menggunakan Dataset 80:20

Perbandingan dataset yang digunakan menggunakan perbandingan 80% untuk data training dan 20% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 6.8.

Tabel 6.8. Pembagian Jumlah Data Support Vector Machine Polynomial 80:20

Kelas	Jumlah	Training 80%	Testing 20%
Negatif	2579	2060	519
Positif	2421	1940	481
<b>Total</b>	<b>5000</b>	<b>4000</b>	<b>1000</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 75%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support Vector Machine* dengan menggunakan 1000 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 6.5.



Gambar 6.5. Confusion Matrix Algoritma Support Vector Machine Polynomial 80:20

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 389 dan yang terprediksi positif (salah) sebanyak 130. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 117 dan yang terprediksi positif (benar) sebanyak 364. Adapun *precision*, *recall*, dan *f1-score* dapat dilihat pada Tabel 6.9.

Tabel 6.9. Classification Report Algoritma Support Vector Machine Polynomial 80:20

Class	Precision	Recall	F1-Score
Negatif	0.77 (77 %)	0.75 (75 %)	0.76 (76 %)
Positif	0.74 (74 %)	0.76 (76 %)	0.75 (75 %)
Accuracy		0.75 (75 %)	

### 6.2.5. Percobaan Support Vector Machine Kernel Polynomial Menggunakan Dataset 70:30

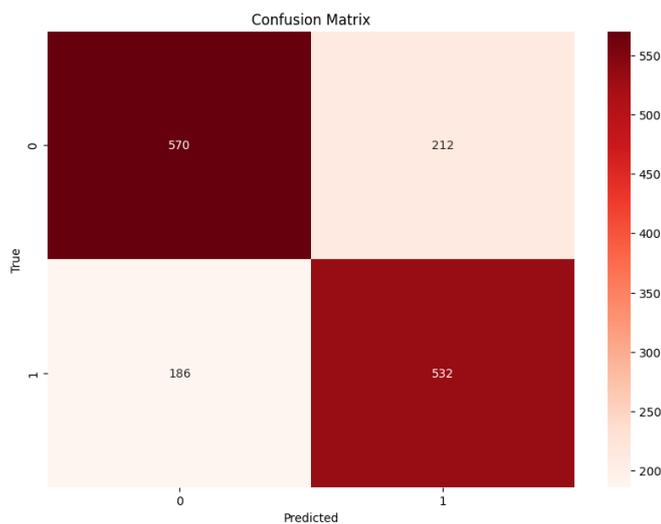
Perbandingan dataset yang digunakan menggunakan perbandingan 70% untuk data training dan 30% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 6.10.

Tabel 6.10. Pembagian Jumlah Data Support Vector Machine Polynomial 70:30

Kelas	Jumlah	Training 70%	Testing 30%
Negatif	2579	1797	782
Positif	2421	1703	718
<b>Total</b>	<b>5000</b>	<b>3500</b>	<b>1500</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 73%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support*

*Vector Machine* dengan menggunakan 1500 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 6.6.



Gambar 6.6. Confusion Matrix Algoritma Support Vector Machine Polynomial 70:30

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 570 dan yang terprediksi positif (salah) sebanyak 212. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 186 dan yang terprediksi positif (benar) sebanyak 532. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 6.11.

Tabel 6.11. Classification Report Algoritma Support Vector Machine Polynomial 70:30

Class	Precision	Recall	F1-Score
Negatif	0.82 (82%)	0.79 (79%)	0.80 (80 %)
Positif	0.78 (78%)	0.81 (81%)	0.80 (80 %)
Accuracy		0.73 (73 %)	

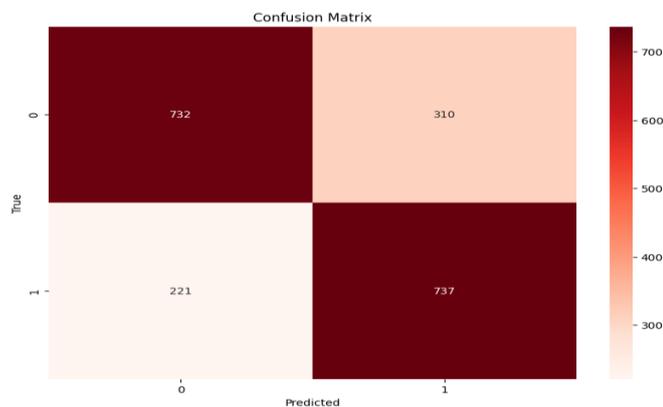
### 6.2.6. Percobaan Support Vector Machine Kernel Polynomial Menggunakan Dataset 60:40

Perbandingan dataset yang digunakan menggunakan perbandingan 60% untuk data training dan 40% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 6.12.

Tabel 6.12. Pembagian Jumlah Data Support Vector Machine Polynomial 60:40

Kelas	Jumlah	Training 70%	Testing 30%
Negatif	2579	1537	1042
Positif	2421	1463	958
<b>Total</b>	<b>5000</b>	<b>3000</b>	<b>2000</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 73%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support Vector Machine* dengan menggunakan 2000 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 6.7.



Gambar 6.7. Confusion Matrix Algoritma Support Vector Machine Polynomial 60:40

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 732 dan yang terprediksi positif (salah) sebanyak 310. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 221 dan yang terprediksi positif (benar) sebanyak 737. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 6.13.

Tabel 6.13. Classification Report Algoritma Support Vector Machine Polynomial 60:40

Class	Precision	Recall	F1-Score
Negatif	0.77 (77 %)	0.70 (70 %)	0.73 (73 %)
Positif	0.70 (70 %)	0.77 (77 %)	0.74 (74%)
Accuracy		0.73 (73 %)	

### 6.2.7. Percobaan Support Vector Machine *Kernel Radial Basis Function*

#### Menggunakan Dataset 80:20

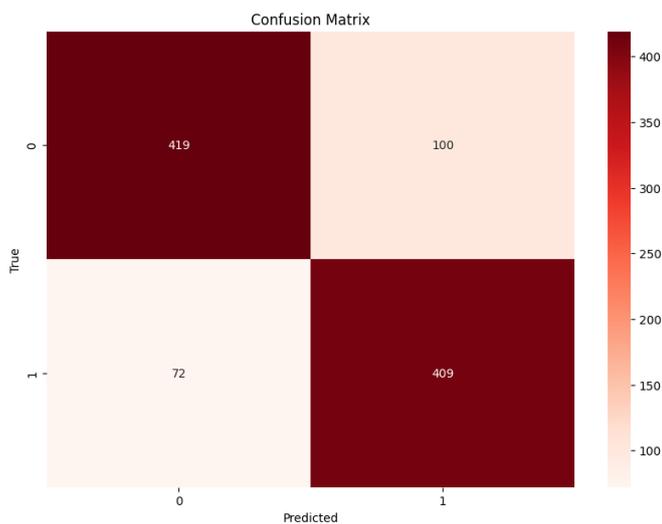
Perbandingan dataset yang digunakan menggunakan perbandingan 80% untuk data training dan 20% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat dilihat pada Tabel 6.14.

Tabel 6.14. Pembagian Jumlah Data Support Vector Machine Radial Basis Function 80:20

Kelas	Jumlah	Training 80%	Testing 20%
Negatif	2579	2060	519
Positif	2421	1940	481
<b>Total</b>	<b>5000</b>	<b>4000</b>	<b>1000</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 83%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support*

*Vector Machine* dengan menggunakan 1000 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 6.8.



Gambar 6.8. Confusion Matrix Algoritma Support Vector Machine Radial Basis Function 80:20

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 419 dan yang terprediksi positif (salah) sebanyak 100. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 72 dan yang terprediksi positif (benar) sebanyak 409. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 6.15.

Tabel 6.15. Classification Report Algoritma Support Vector Machine Radial Basis Function 80:20

Class	Precision	Recall	F1-Score
Negatif	0.85 (85 %)	0.81 (81 %)	0.83 (83 %)
Positif	0.80 (80 %)	0.85 (85 %)	0.83 (83 %)
Accuracy		0.83 (83 %)	

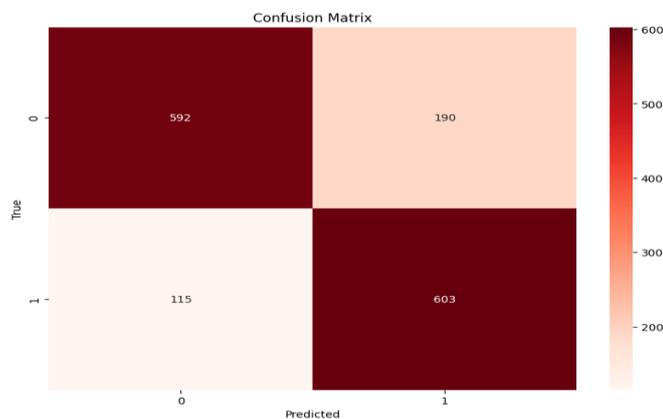
### 6.2.8. Percobaan Support Vector Machine *Kernel Radial Basis Function* Menggunakan Dataset 70:30

Perbandingan dataset yang digunakan menggunakan perbandingan 70% untuk data training dan 30% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat di lihat pada Tabel 6.16.

Tabel 6.16. Pembagian Jumlah Data Support Vector Machine Radial Basis Function 70:30

Kelas	Jumlah	Training 70%	Testing 30%
Negatif	2579	1797	782
Positif	2421	1703	718
<b>Total</b>	<b>5000</b>	<b>3500</b>	<b>1500</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 80%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support Vector Machine* dengan menggunakan 1500 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat dilihat pada Gambar 6.9.



Gambar 6.9. Confusion Matrix Algoritma Support Vector Machine Radial Basis Function 70:30

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 592 dan yang terprediksi positif (salah) sebanyak 190. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 115 dan yang terprediksi positif (benar) sebanyak 603. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 6.17.

Tabel 6.17. Classification Report Algoritma Support Vector Machine Radial Basis Function 70:30

Class	Precision	Recall	F1-Score
Negatif	0.84 (82%)	0.76 (79%)	0.80 (80 %)
Positif	0.76 (78%)	0.84 (81%)	0.80 (80 %)
Accuracy		0.80 (80 %)	

### 6.2.9. Percobaan Support Vector Machine *Kernel Radial Basis Function*

#### Menggunakan Dataset 60:40

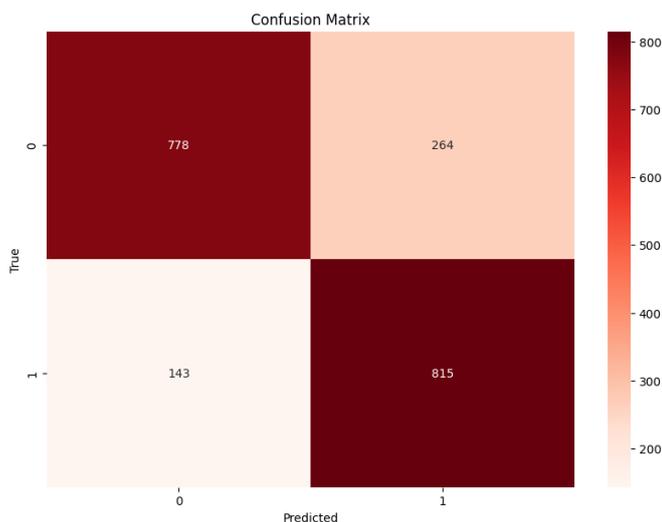
Perbandingan dataset yang digunakan menggunakan perbandingan 60% untuk data training dan 40% untuk data testing. Adapun perbandingan jumlah data yang digunakan dapat di lihat pada Tabel 6.18.

Tabel 6.18. Pembagian Jumlah Data Support Vector Machine Radial Basis Function 60:40

Kelas	Jumlah	Training 70%	Testing 30%
Negatif	2579	1537	1042
Positif	2421	1463	958
<b>Total</b>	<b>5000</b>	<b>3000</b>	<b>2000</b>

Dari hasil percobaan yang dilakukan, algoritma *Support Vector Machine* memperoleh hasil akurasi sangat baik dalam proses klasifikasi sentimen ulasan atau komentar aplikasi TikTok dengan tingkat akurasi sebesar 80%. Hal ini dibuktikan dengan hasil evaluasi dari proses pengujian model algoritma *Support*

*Vector Machine* dengan menggunakan 2000 data testing. Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mengetahui kinerja dari algoritma tersebut. Adapun hasil dari pengujian yang dilakukan dapat di lihat pada Gambar 6.10.



Gambar 6.10. Confusion Matrix Algoritma Support Vector Machine Radial Basis Function 60:40

Berdasarkan *confusion matrix*, dapat diketahui bahwa data ulasan negatif yang terprediksi negatif (benar) sebanyak 778 dan yang terprediksi positif (salah) sebanyak 264. Sedangkan data ulasan positif yang terprediksi negatif (salah) sebanyak 143 dan yang terprediksi positif (benar) sebanyak 815. Adapun precision, recall, dan f1-score dapat dilihat pada Tabel 6.19.

Tabel 6.19. Classification Report Algoritma Support Vector Machine Radial Basis Function 60:40

Class	Precision	Recall	F1-Score
Negatif	0.84 (77 %)	0.75 (70 %)	0.79 (73 %)
Positif	0.76 (70 %)	0.85 (77 %)	0.80 (74%)
Accuracy		0.80 (80 %)	

### 6.3. Performa Algoritma *Support Vector Machine*

Berdasarkan hasil uji coba menggunakan 3 percobaan dengan 3 parameter *kernel* maka rata rata performa *Support Vector Machine* dapat dilihat pada Tabel 6.20.

Tabel 6.20. Performa *Support Vector Machine*

NO	PARAMETER	PERCOBAAN	DATA	PERFORMA SUPPORT VECTOR MACHINE						
				Accuracy	Precision (Negatif)	Precision (Positif)	Recall (Negatif)	Recall (Positif)	f1-Score (Negatif)	f1-Score (Positif)
1	<i>SVM - Kernel Linear</i>	Percobaan Pertama	80:20	82%	83%	<b>82%</b>	<b>84%</b>	81%	83%	82%
2		Percobaan Kedua	70:30	80%	82%	78%	79%	81%	80%	80%
3		Percobaan Ketiga	60:40	80%	83%	77%	78%	83%	80%	80%
4	<i>SVM - Kernel Polynomial</i>	Percobaan Pertama	80:20	75%	77%	74%	75%	76%	76%	75%
5		Percobaan Kedua	70:30	80%	82%	78%	79%	81%	80%	80%
6		Percobaan Ketiga	60:40	80%	77%	70%	70%	77%	73%	74%
7	<i>SVM - Kernel Radial Basis Function</i>	Percobaan Pertama	80:20	<b>83%</b>	<b>85%</b>	80%	81%	85%	<b>83%</b>	<b>83%</b>
8		Percobaan Kedua	70:30	80%	84%	76%	76%	84%	80%	80%
9		Percobaan Ketiga	60:40	80%	84%	76%	75%	<b>85%</b>	79%	80%
<b>Rata Rata</b>				80%	<b>82%</b>	77%	77%	81%	79%	79%

Pada Tabel 6.20. diketahui nilai *accuracy* dan *precision (negatif)* tertinggi di peroleh pada percobaan pertama dengan *kernel radial basis function* yaitu 83%, dan 85%, sedangkan nilai *precision (positif)* dan *recall (negatif)* tertinggi di peroleh pada percobaan pertama dengan *kernel linear* yaitu 82% dan 84%, untuk nilai *recall (positif)* tertinggi di peroleh pada percobaan ketiga dengan *kernel radial basis function* yaitu 85%, *f1-score (negatif)* dan *f1-score (positif)* tertinggi di peroleh pada percobaan pertama dengan *kernel radial basis function* yaitu 83% dan 83%. Rata - rata performa *Support Vector Machine* adalah *accuracy* sebesar 80%, nilai *precision (negatif)* 82%, nilai *precision*

(positif) 77%, nilai *recall* (negatif) 77%, nilai *recall* (positif) 81%, nilai *f1-score* (negatif) 79% dan *f1-score* (positif) 79%

## BAB VII

### HASIL PEMBAHASAN

#### 7.1. Hasil dan Pembahasan

Dalam eksplorasi algoritma klasifikasi untuk mengelola data sentimen dari ulasan aplikasi TikTok, penelitian ini membandingkan dua metode yang populer yaitu *Naive Bayes* dan *Support Vector Machine*. Kedua algoritma ini diterapkan pada dataset yang sama, yang terdiri dari 5000 data yang dibagi dengan rasio 80%, 70% dan 60% untuk pelatihan dan 20%, 30% dan 40% untuk pengujian, untuk memastikan kedua model diuji dalam kondisi yang sama.

*Naive Bayes*, yang terkenal dengan efisiensinya dalam pembelajaran mesin berbasis probabilitas, menunjukkan kemampuan yang memadai dalam klasifikasi sentimen dengan mencapai akurasi 79%. Analisis lebih lanjut dengan menggunakan confusion matrix menunjukkan bahwa model ini memiliki kecenderungan lebih baik dalam mengidentifikasi ulasan negatif daripada positif, dengan *precision* dan *recall* masing-masing sebesar 78% dan 84% untuk ulasan negatif. Meskipun efektif, model ini memiliki beberapa keterbatasan, seperti asumsi independensi fitur yang mungkin tidak selalu tepat dalam analisis sentimen nyata. Sebaliknya, *Support Vector Machine* yang menggunakan pendekatan geometric untuk memaksimalkan margin antara kelas data, berhasil mencapai tingkat akurasi yang lebih tinggi pada 83%. Keunggulan *Support Vector Machine* dalam studi ini terlihat dari kemampuannya yang lebih baik dalam membedakan antara komentar positif dan negatif dengan menggunakan *hyperplane* yang optimal. *Support Vector Machine* menunjukkan *precision* dan

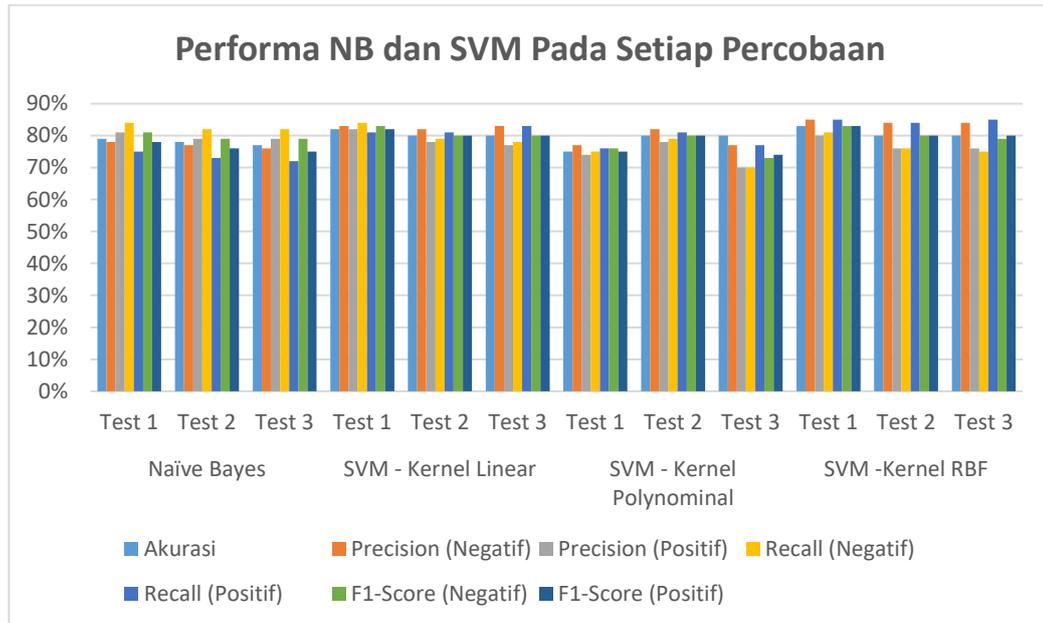
*recall* yang konsisten di atas 80% untuk kedua kelas, yang menunjukkan kinerja yang lebih seimbang dibandingkan dengan *Naive Bayes*. Adapun perbandingan dari kedua model algoritma ini yakni sebagaimana Tabel 7.1.

Tabel 7.1. Tabel Perbandingan Algoritma Naive Bayes dan Support Vector Machine

Kriteria	Naïve Bayes			SVM - Kernel Linear			SVM - Kernel Polynomial			SVM -Kernel RBF		
	80:20	70:30	60:40	80:20	70:30	60:40	80:20	70:30	60:40	80:20	70:30	60:40
Akurasi	79%	78%	77%	82%	80%	80%	75%	80%	80%	83%	80%	80%
Precision (Negatif)	78%	77%	76%	83%	82%	83%	77%	82%	77%	85%	84%	84%
Precision (Positif)	81%	79%	79%	82%	78%	77%	74%	78%	70%	80%	76%	76%
Recall (Negatif)	84%	82%	82%	84%	79%	78%	75%	79%	70%	81%	76%	75%
Recall (Positif)	75%	73%	72%	81%	81%	83%	76%	81%	77%	85%	84%	85%
F1-Score (Negatif)	81%	79%	79%	83%	80%	80%	76%	80%	73%	83%	80%	79%
F1-Score (Positif)	78%	76%	75%	82%	80%	80%	75%	80%	74%	83%	80%	80%

## 7.2. Perbandingan Performa Naïve Bayes dan Support Vector Machine

Setelah diketahui hasil pengujian *Naive Bayes* dan *Support Vector Machine* (menggunakan beberapa perubahan kernel) pada percobaan pertama sampai percobaan tiga, tahap selanjutnya adalah melakukan perbandingan performa *Naive Bayes* dan *Support Vector Machine* berdasarkan nilai *accuracy*, nilai *precision*, nilai *recall* dan nilai *f1-score*. Perbandingan performa *Naive Bayes* dan *Support Vector Machine* dapat dilihat pada gambar 7.1.



Gambar 7.1 Perbandingan *Naïve Bayes* dan *Support Vector Machines*

Pada gambar 7.1 diketahui pada percobaan pertama performa *Support Vector Machines* dengan *Kernel Radial Basis Function* unggul pada nilai *accuracy* (83%), nilai *precision* negatif (85%), nilai *recall* positif (85%), dan nilai *f1-score* positif dan negatif sebesar (83%). Sedangkan *Naïve Bayes* dan *Support Vector Machines* dengan *Kernel Linear* unggul pada nilai *recall* negatif yang sama (84%).

Pada percobaan kedua diketahui performa *Support Vector Machines* dengan 3 kernel yang digunakan sama sama unggul pada nilai akurasi (80%). Nilai *precision* negatif dan *recall* positif sebesar (84%) diperoleh *Support Vector Machines* dengan *Kernel Radial Basis Function* dan *f1-score* (80%). Sedangkan *Naïve Bayes* unggul pada nilai *akurasi* (78%) dan *recall* negatif (82%).

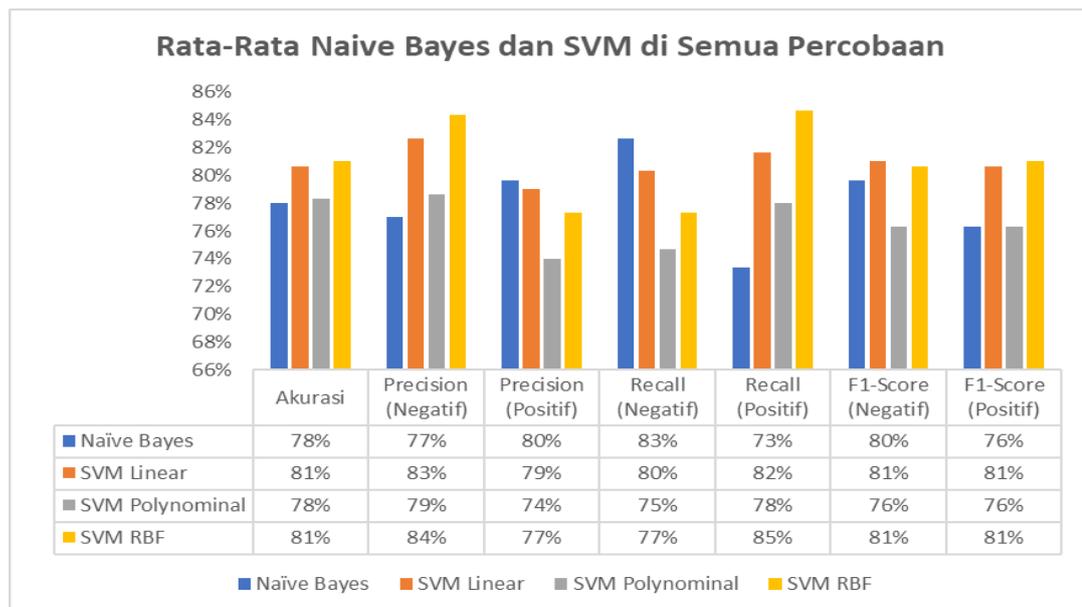
Pada percobaan ketiga diketahui performa *Support Vector Machine* dengan 3 kernel yang digunakan sama sama unggul pada nilai akurasi (80%). Pada nilai *precision* negatif dan *recall* positif, *Support Vector Machine* dengan

*Kernel Radial Basis Function* unggul dengan memperoleh nilai yang sama (85%). Sedangkan *Naive Bayes* unggul pada nilai *recall* negatif (82%). Adapun nilai rata-rata dari perbandingan 4 model diatas dapat diketahui pada tabel 7.2.

Tabel 7.2. Tabel Rata-Rata Hasil Percobaan

Kriteria	<i>Naive Bayes</i>	<i>SVM Linear</i>	<i>SVM Polynomial</i>	<i>SVM RBF</i>
<i>Akurasi</i>	78%	81%	78%	81%
<i>Precision (Negatif)</i>	77%	83%	79%	84%
<i>Precision (Positif)</i>	80%	79%	74%	77%
<i>Recall (Negatif)</i>	83%	80%	75%	77%
<i>Recall (Positif)</i>	73%	82%	78%	85%
<i>F1-Score (Negatif)</i>	80%	81%	76%	81%
<i>F1-Score (Positif)</i>	76%	81%	76%	81%

Berdasarkan tabel 7.2 diatas dapat diketahui bahwa algoritma *Support Vector Machine* dengan *Kernel Radial Basis Function* nilai rata-rata pada beberapa kriteria mengungguli beberapa algoritma yang digunakan. Adapun grafik nilai rata-rata dari perbandingan 4 model diatas dapat diketahui dan gambar 7.2



Gambar 7.2 Rata-Rata Performa *Naive Bayes* dan *Support Vector Machines*

Perbandingan 4 (empat) model ini menggambarkan bagaimana pemilihan algoritma dapat berdampak signifikan terhadap hasil klasifikasi sentimen. Meskipun *Naive Bayes* cepat dan efisien untuk dataset besar, *Support Vector Machine* menawarkan keakuratan yang lebih tinggi dengan kemampuan untuk menghandle data yang lebih kompleks dan *overlapped*. Keputusan antara kedua metode ini harus didasarkan pada kebutuhan spesifik penggunaan, dimana *Support Vector Machine* mungkin lebih cocok untuk aplikasi yang memerlukan tingkat diskriminasi yang tinggi antara sentimen yang berbeda. Kedua model, *Naive Bayes* dan *Support Vector Machine*, mengimplementasikan pendekatan yang berbeda terhadap masalah yang sama, yakni klasifikasi sentimen dari ulasan aplikasi TikTok. Dari analisis yang dilakukan, *Naive Bayes* menawarkan keuntungan dalam hal kecepatan dan kesederhanaan implementasi. Model ini ideal untuk skenario di mana kecepatan pemrosesan merupakan faktor kritis, dan sumber daya komputasi yang tersedia terbatas. Meskipun memiliki asumsi independensi yang bisa menjadi batasan dalam kasus penggunaan tertentu, teknik ini masih relevan untuk dataset yang memiliki banyak fitur yang secara statistik independen.

Di sisi lain, *Support Vector Machine* menonjol karena kemampuannya dalam menciptakan batas keputusan yang kuat, terutama dalam kasus-kasus di mana kelas data tidak mudah dipisahkan secara linier. Keakuratan yang lebih tinggi dalam model *Support Vector Machine* menunjukkan bahwa teknik ini mampu mengatasi kelemahan *Naive Bayes*, terutama dalam menangani fitur-fitur yang berkorelasi. Walaupun membutuhkan lebih banyak sumber daya komputasi dan waktu, keefektifannya dalam mengklasifikasikan ulasan dengan

lebih akurat menjadikannya pilihan yang lebih baik untuk aplikasi yang memerlukan tingkat prediksi yang lebih presisi.

Kesimpulannya, pemilihan antara *Naive Bayes* dan *Support Vector Machine* harus mempertimbangkan baik karakteristik data yang dihadapi maupun kebutuhan operasional dari aplikasi yang menggunakan model tersebut. Dalam konteks analisis sentimen di mana nuansa dan konteks kata-kata sangat penting, *Support Vector Machine* mungkin menawarkan pendekatan yang lebih robust. Namun, untuk implementasi yang membutuhkan respons cepat atau di mana sumber daya terbatas, *Naive Bayes* masih bisa menjadi pilihan yang sangat efektif (Fajrina, Syafriandi, Amalita, & Salma, 2023). Sebagai langkah lanjut, pengujian lebih lanjut dengan variasi parameter dan teknik pengolahan data pra-model dapat memberikan wawasan lebih dalam tentang bagaimana meningkatkan kinerja kedua model ini dalam berbagai skenario penggunaan.

### **7.3. Integrasi Perspektif Al-Qur'an dan Hadist**

Aspek *mu'amalah ma'a Allah* dalam mengomentari aplikasi TikTok berfokus pada bagaimana seseorang menjaga hubungan dan tanggung jawabnya kepada Allah SWT saat berinteraksi di platform tersebut. Dalam mengomentari TikTok, seseorang harus menjaga niat yang ikhlas dan bertujuan untuk menyebarkan kebaikan serta menghindari dosa. Setiap komentar harus disampaikan dengan rasa takut dan harap kepada Allah, memastikan bahwa tidak ada fitnah, ghibah, atau penghinaan terhadap orang lain yang dapat merusak hubungan sosial dan mengundang murka Allah. Selain itu, seseorang harus menegakkan keadilan, kebenaran, dan kebijaksanaan dalam berbicara,

selalu mengingat bahwa setiap kata yang diucapkan akan dipertanggungjawabkan di hadapan Allah pada hari kiamat. Dengan demikian, menjaga adab dan etika dalam berkomentar di TikTok merupakan bentuk ibadah dan penghormatan terhadap aturan Allah dalam kehidupan sehari-hari.

Aspek *mu'amalah ma'a an-nas* dalam mengomentari aplikasi TikTok melibatkan interaksi yang baik, adil, dan penuh hormat terhadap sesama pengguna platform tersebut. Dalam memberikan komentar, seseorang harus menghindari fitnah, ghibah, dan ucapan yang merendahkan atau menyakiti perasaan orang lain, sebagaimana diajarkan dalam Al-Qur'an dan Hadist. Manfaat dari memberikan komentar yang konstruktif di TikTok termasuk menciptakan lingkungan online yang positif dan mendidik, mendorong kreativitas yang bermanfaat, dan memperkuat rasa saling menghormati di antara pengguna. Komentar yang baik juga dapat memberikan dukungan moral, menawarkan saran yang berguna, dan membantu memperbaiki konten yang kurang sesuai, sehingga bersama-sama berkontribusi pada penggunaan media sosial yang lebih bertanggung jawab dan bermakna.

Dalam menyampaikan pendapat menurut Al-Qur'an dan Hadist, terdapat beberapa ketentuan dan batasan yang harus diperhatikan. Pertama, hindari fitnah dan penyebaran informasi palsu dengan selalu memverifikasi fakta (Surah Al-Hujurat 49:6). Kedua, jauhi ghibah (menggunjing) dan namimah (adu domba) serta berprasangka buruk (Surah Al-Hujurat 49:12). Ketiga, sampaikan pendapat dengan hikmah, bijaksana, dan kata-kata yang baik (Surah An-Nahl 16:125; Hadis Bukhari dan Muslim). Keempat, hindari perdebatan yang tidak bermanfaat (Hadis Tirmidzi). Kelima, jangan menyinggung atau menghina

orang lain (Surah Al-Hujurat 49:11). Keenam, jaga kerahasiaan informasi yang amanah (Surah An-Nisa' 4:58). Terakhir, hindari berbicara sembarangan tentang agama dan sampaikan hanya hal-hal yang bermanfaat dan benar (Hadis Abu Dawud). Dengan mengikuti ketentuan ini, pendapat dapat disampaikan dengan cara yang konstruktif dan sesuai dengan ajaran Islam.

Dalam konteks penilaian analisis sentimen terhadap mengomentari aplikasi atau teknologi dalam pandangan Al-Quran dan Hadist, terdapat prinsip yang bisa diambil sebagai dasar untuk kebebasan berpendapat yang mana dijelaskan dalam Al-Qur'an Surah Al-Hujurat (49:6) Allah berfirman:

يَا أَيُّهَا الَّذِينَ آمَنُوا إِن جَاءَكُمْ فَاسِقٌ بِنَبَأٍ فَتَبَيَّنُوا أَن تُصِيبُوا قَوْمًا بِجَهَالَةٍ فَتُصْحَبُوا عَلَىٰ مَا  
فَعَلْتُمْ نَادِمِينَ

*"Hai orang-orang yang beriman, jika datang kepadamu orang fasik membawa suatu berita, maka periksalah dengan teliti agar kamu tidak menimpakan suatu musibah kepada suatu kaum tanpa mengetahui keadaannya yang menyebabkan kamu menyesal atas perbuatanmu itu."*

Ayat ini menekankan pentingnya memverifikasi informasi sebelum menyebarkannya atau mengambil tindakan berdasarkan informasi tersebut, yang sangat relevan dalam konteks penilaian terhadap aplikasi atau berita di era digital saat ini. Ini relevan dalam menilai aplikasi, di mana pengguna harus memastikan kredibilitas dan keamanan aplikasi sebelum menggunakannya.

Dari sumber utama dalam Islam tersebut, dapat disimpulkan bahwa dalam memberikan komentar atau analisis terhadap aplikasi atau teknologi, umat Islam dianjurkan untuk:

- a. Memiliki pengetahuan yang cukup sebelum memberikan pendapat.

- b. Berbicara dengan baik atau diam jika tidak memiliki sesuatu yang baik untuk dikatakan.
- c. Berhati-hati dalam menyebarkan informasi atau memberikan komentar, untuk menghindari kesalahan atau menimpakan musibah kepada orang lain.
- d. Memperhatikan kebaikan dalam perkataan dan tindakan, sehingga memberikan manfaat kepada orang lain.

Dengan demikian, kebebasan berpendapat dalam Islam diimbangi dengan tanggung jawab moral untuk menggunakan kebebasan tersebut dengan bijaksana, sesuai dengan ajaran agama dan nilai-nilai etika.

## BAB VIII

### PENUTUP

#### 8.1. Kesimpulan

Penelitian ini membahas secara mendalam performa dua algoritma klasifikasi, *Naive Bayes* dan *Support Vector Machine* dalam menganalisis sentimen terhadap aplikasi TikTok berdasarkan ulasan/komentar di *Playstore*. Dari penelitian ini, di dapatkan bahwa *Support Vector Machine* dengan *Kernel Radial Basis Function* (RBF) menunjukkan performa yang lebih baik dengan akurasi 83%, sedangkan *Naive Bayes* mencapai akurasi 79%. Akurasi tersebut diperoleh dari split data 80:20 untuk training dan testing. Hal ini menunjukkan bahwa *Support Vector Machine* lebih efektif dalam mengklasifikasikan sentimen positif dan negatif pada dataset yang digunakan. Penelitian ini memberikan wawasan tentang bagaimana kedua algoritma ini memproses dan mengkategorikan data teks yang besar dan variatif, yang penting untuk pemahaman analisis sentimen di era digital.

#### 8.2. Saran

Untuk penelitian selanjutnya, disarankan untuk mengeksplorasi penggunaan dataset yang lebih besar dan beragam untuk menguji kemampuan dari algoritma klasifikasi yang digunakan. Penelitian lebih lanjut bisa mencakup pengembangan dan penerapan teknik *preprocessing* yang lebih canggih, seperti *word embedding* atau penggunaan model bahasa berbasis *neural network*, untuk meningkatkan akurasi klasifikasi. Selain itu, membandingkan dengan algoritma klasifikasi lain seperti *Decision Trees* atau *Neural Networks* dapat memberikan

perspektif yang lebih luas mengenai efektivitas berbagai metode dalam analisis sentimen. Pendekatan multi-algoritma dan *hybrid* bisa juga dipertimbangkan untuk menangani nuansa bahasa dan *slang* yang sering muncul dalam komentar online.

## DAFTAR PUSTAKA

- Ardiansyah, D., Saepudin, A., Aryanti, R., Fitriani, E., & Royadi. (2023). Analisis Sentimen Review Pada Aplikasi Media Sosial Tiktok Menggunakan Algoritma K-NN dan SVM Berbasis PSO. *Jurnal Informatika Kaputama (JIK)*, 7(2), 233–241.
- Bafna, P., Pramod, D., & Vaidya, A. (2016). Document Clustering: TF-IDF Approach. *International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 61–66.
- Cheng, L., Yang, Y., Zhao, K., & Gao, Z. (2020). Research and improvement of tf-idf algorithm based on information theory. *Advances in Intelligent Systems and Computing*, 905, 608–616. Springer Verlag. [https://doi.org/10.1007/978-3-030-14680-1\\_67](https://doi.org/10.1007/978-3-030-14680-1_67)
- Fajrina, D. P., Syafriandi, Amalita, N., & Salma, A. (2023). Sentiment Analysis of TikTok Application on Twitter using The Naïve Bayes Classifier Algorithm. *UNP Journal of Statistics and Data Science*, 1(5), 392–398. <https://doi.org/10.24036/ujsds/vol1-iss5/103>
- Farhana, S. (2021). Classification of Academic Performance for University Research Evaluation by Implementing Modified Naive Bayes Algorithm. *Procedia Computer Science*, 194, 224–228. Elsevier B.V. <https://doi.org/10.1016/j.procs.2021.10.077>
- Fide, S., Suparti, & Sudarno. (2021). Analisis Sentimen Ulasan Aplikasi Tiktok di Google Play Menggunakan Metode Support Vector Machine (SVM) dan Asosiasi. *Jurnal Gaussian*, 10(3), 346–358. Retrieved from <https://ejournal3.undip.ac.id/index.php/gaussian/>
- Fitri, V. A., Andreswari, R., & Hasibuan, M. A. (2019). Sentiment analysis of social media Twitter with case of Anti-LGBT campaign in Indonesia using Naïve Bayes, decision tree, and random forest algorithm. *Procedia Computer Science*, 161, 765–772. Elsevier B.V. <https://doi.org/10.1016/j.procs.2019.11.181>
- Han, J., Kamber, M., & Pei, J. (2012). *Data Mining. Concepts and Techniques*, 3rd Edition (The Morgan Kaufmann Series in Data Management Systems) (3rd ed.). 225 Wyman Street, Waltham, MA 02451, USA: Elsevier.
- Hananto, B. K., Pinandito, A., & Kharisma, A. P. (2018). Penerapan Maximum TF-IDF Normalization Terhadap Metode KNN Untuk Klasifikasi Dataset Multiclass Panichella Pada Review Aplikasi Mobile. *Jurnal Pengembangan Teknologii Informasi Dan Ilmu Komputer*, 2(12), 6812–6823. Retrieved from <http://j-ptiik.ub.ac.id>

- Indriyani, F. A., Fauzi, A., & Faisal, S. (2023). Analisis Sentimen Aplikasi TikTok Menggunakan Algoritma Naive Bayes dan Support Vector Machine. *TEKNOSAINS : Jurnal Sains, Teknologi Dan Informatika*, 10(2), 176–184. <https://doi.org/10.37373/tekno.v10i2.419>
- Irfani, F. F., Triyanto, M., Hartanto, A. D., & Kusnawi. (2020). Analisis Sentimen Review Aplikasi Ruangguru Menggunakan Algoritma Support Vector Machine. *Jurnal Bisnis, Manajemen Dan Informatika*, 16(3), 258–266.
- Isnan, M., Elwirehardja, G. N., & Pardamean, B. (2023). Sentiment Analysis for TikTok Review Using VADER Sentiment and SVM Model. *Procedia Computer Science*, 227, 168–175. <https://doi.org/10.1016/j.procs.2023.10.514>
- Mccallum, A., & Nigam, K. (2001). A Comparison of Event Models for Naive Bayes Text Classification. *Work Learn Text Categ*, 752.
- Mubaroroh, H. H., Yasin, H., & Rusgiyono, A. (2022). Analisis Sentimen Data Ulasan Aplikasi Ruangguru Pada Situs Google Play Menggunakan Algoritma Naive Bayes Classifier Dengan Normalisasi Kata Levenshtein Distance. *Jurnal Gaussian*, 11(2), 248–257. Retrieved from <https://ejournal3.undip.ac.id/index.php/gaussian/>
- Neogi, A. S., Garg, K. A., Mishra, R. K., & Dwivedi, Y. K. (2021). Sentiment analysis and classification of Indian farmers' protest using twitter data. *International Journal of Information Management Data Insights*, 1(2). <https://doi.org/10.1016/j.jjime.2021.100019>
- Novitasari, I., Kurniawan, T. B., Dewi, D. A., & Misinem. (2022). Analisis Sentimen Masyarakat Terhadap Tweet Ruang Guru Menggunakan Algoritma Naive Bayes Classifier (NBC). *Jurnal Mantik*, 6(3), 2685–4236.
- Pawar, A. B., Jawale, M. A., & Kyatanavar, D. N. (2016). Fundamentals of Sentiment Analysis: Concepts and Methodology. In *Studies in Computational Intelligence* (Vol. 639, pp. 25–48). Springer Verlag. [https://doi.org/10.1007/978-3-319-30319-2\\_2](https://doi.org/10.1007/978-3-319-30319-2_2)
- Pradhan, A., Senapati, M. R., & Sahu, P. K. (2022). Improving sentiment analysis with learning concepts from concept, patterns lexicons and negations: Improving sentiment analysis with learning concepts. *Ain Shams Engineering Journal*, 13(2). <https://doi.org/10.1016/j.asej.2021.08.004>
- Rahab, H., Zitouni, A., & Djoudi, M. (2021). SANA: Sentiment analysis on newspapers comments in Algeria. *Journal of King Saud University - Computer and Information Sciences*, 33(7), 899–907. <https://doi.org/10.1016/j.jksuci.2019.04.012>

- Roldós, I. (2020, March 2). Go-to Guide for Text Classification with Machine Learning.
- Samsir, Ambiyar, Verawardina, U., Edi, F., & Watrianthos, R. (2021). Analisis Sentimen Pembelajaran Daring Pada Twitter di Masa Pandemi COVID-19 Menggunakan Metode Naive Bayes. *Jurnal Media Informatika Budidarma*, 5(1), 157–163. <https://doi.org/10.30865/mib.v5i1.2604>
- Sudhir, P., & Suresh, V. D. (2021). Comparative study of various approaches, applications and classifiers for sentiment analysis. *Global Transitions Proceedings*, 2(2), 205–211. <https://doi.org/10.1016/j.glt.2021.08.004>
- Wenando, F. A., Hayami, R., & Anggrawan, A. J. (2020). Analisis Sentimen Pada Pemerintahan Terpilih Pada Pilpres 2019 di Twitter Menggunakan Algoritman Naive Bayes. *Jurnal Teknologi Dan Sistem Informasi*, 7(1), 101–106. <https://doi.org/10.33330/jurteksi.v7i1.851>
- Worth, D. (2010). Introduction to Modern Information Retrieval, 3rd Edition. *Australian Academic and Research Libraries*, 41(4), 305–306. <https://doi.org/10.1080/00048623.2010.10721488>
- Yulita, W., Nugroho, E. D., & Algifari, M. H. (2021). Analisis Sentimen Terhadap Opini Masyarakat Tentang Vaksin Covid-19 Menggunakan Algoritma Naive Bayes Classifier. *JDMSI*, 2(2), 1–9.
- Zulqornain, J. A., Indriati, & Adikara, P. P. (2021). Analisis Sentimen Tanggapan Masyarakat Aplikasi Tiktok Menggunakan Metode Naive Bayes dan Categorical Proportional Difference (CPD). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 5(7), 2886–2890. Retrieved from <http://j-ptiik.ub.ac.id>