

# Penerapan Naïve Bayes Untuk Analisis Sentimen Pada Ulasan Aplikasi Mobile Legends

Attila Elang Perkasa, Astrid Novita Putri\*

Fakultas Teknik Informasi dan Komunikasi, Teknik Informatika, Universitas Semarang, Semarang, Indonesia

Email: <sup>1</sup>attilaelangperkasa0@gmail.com, <sup>2,\*</sup>astrid@usm.ac.id

Email Penulis Korespondensi: astrid@usm.ac.id

Submitted: 23/12/2024; Accepted: 26/02/2025; Published: 01/03/2025

**Abstrak**—Indonesia menjadi salah satu pasar potensial bagi industri game terus meningkatkan jumlah pemain game atau *gamer*. Salah satu *mobile game* terpopuler di Indonesia adalah *Mobile Legends*. *Mobile Legends* adalah permainan video online bergenre *MOBA* yang dikembangkan oleh Moonton, sebuah perusahaan pengembang permainan asal Tiongkok. Pengaruh ulasan terhadap reputasi sebuah aplikasi juga akan berpengaruh kepada calon pengguna baru, baik ulasan positif maupun ulasan negatif. Penelitian yang telah dilakukan menunjukkan bahwa model Naïve Bayes menunjukkan akurasi yang baik untuk digunakan dalam analisis sentimen. Penelitian ini diharapkan dapat membantu dalam memahami persepsi dan pengalaman pemain permainan tersebut. Pada penelitian ini menggunakan metode *KDD (Knowledge Discovery in Data)* karena memiliki keunggulan proses pengidentifikasian pola yang terorganisir dari sekumpulan data yang kompleks, sehingga datanya mudah untuk dipahami. Pada proses penelitian didapatkan 320513 ulasan positif, 185777 ulasan negatif dan 20210 ulasan netral. Nilai akurasi konstan pada angka 87% pada skenario pembagian data 80:20. Performa pada kelas negatif menunjukkan presisi yang tinggi bernilai 88%. Pada *recall* negatif menunjukkan angka 92%, sehingga model dapat menangkap ulasan yang benar benar negatif. Nilai stabil 82% *F1-Score* menandakan keseimbangan model yang baik antara presisi dan *recall* untuk kelas negatif. Performa pada kelas positif di angka 87% dan 83% *recall*. *F1-score* antara keduanya hampir seimbang menunjukkan bahwa model memiliki performa yang serupa untuk kedua label secara keseluruhan.

**Kata Kunci:** Mobile Legend; Analisis Sentimen; Model Naive Bayes; Knowledge Discovery in Data; Ulasan

**Abstract**—Indonesia has become one of the potential markets for the gaming industry, continually increasing the number of gamers. One of the most popular mobile games in Indonesia is *Mobile Legends*. *Mobile Legends* is an online multiplayer battle arena (*MOBA*) video game developed by Moonton, a game development company based in China. The influence of reviews on the reputation of an app also affects potential new users, whether the reviews are positive or negative. Research has shown that the Naïve Bayes model provides good accuracy for sentiment analysis. This study is expected to help understand the perceptions and experiences of players of the game. The study uses the *KDD (Knowledge Discovery in Data)* method due to its advantages in identifying organized patterns from a complex dataset, making the data easier to understand. During the research process, 320,513 positive reviews, 185,777 negative reviews, and 20,210 neutral reviews were obtained. The accuracy value remained constant at 87% for the 80:20 data split scenario. Performance on the negative class showed high precision at 88%. The negative recall was 92%, indicating that the model could accurately capture truly negative reviews. A stable *F1-Score* of 82% signifies a good balance between precision and recall for the negative class. Performance on the positive class showed 87% precision and 83% recall. The *F1-Score* between the two was nearly balanced, indicating that the model performed similarly for both labels overall.

**Keywords:** Mobile Legend; sentiment analysis; Naïve Bayes Model; Knowledge Discovery in Data; Review

## 1. PENDAHULUAN

Indonesia menjadi salah satu pasar potensial bagi industri game seiring dengan terus meningkatkan jumlah pemain game atau *gamer*. Diperkirakan pada 2025, jumlah pemain game mencapai 192,1 juta orang yang tersebar di berbagai platform[1]. Dari angka tersebut, *smartphone* menjadi platform terpopuler. Salah satu *mobile game* terpopuler di Indonesia adalah *Mobile Legends*. *Mobile Legends* adalah permainan video online bergenre *MOBA* yang dikembangkan oleh Moonton, sebuah perusahaan pengembang permainan asal Tiongkok[2]. Game *Mobile Legends* sekarang menjadi salah satu game yang sedang naik daun, bersaing dengan *Arena of Valor* dan game *MOBA* Android lainnya[3]. *Mobile Legends* dirilis di server Indonesia pada tanggal 11 Juli 2016 dan menjadi salah satu game yang masuk dalam cabang *E-Sport*[4]. Menurut Craig Chapple dalam *Sensor Tower*, Negara dengan unduhan terbanyak adalah Indonesia, dengan 100,1 juta unduhan atau 36 persen dari total instalasi unik. Filipina berada di peringkat kedua dengan 41,2 juta unduhan atau 15 persen, sementara Vietnam berada di peringkat ketiga dengan 21,3 juta instalasi atau 8 persen dari total instalasi[5].

Memiliki jutaan pemain aktif tiap harinya, permainan ini telah menjadi budaya pada kalangan anak muda Asia khususnya pada Indonesia, dengan adanya kompetisi / turnamen skala regional, nasional maupun internasional yang diselenggarakan di Indonesia, menambah pentingnya memahami opini / sentimen pengguna dari Indonesia untuk menambah pengetahuan pengembang dalam kepuasan atau kekurangan permainan mereka. Pengaruh ulasan terhadap reputasi sebuah aplikasi juga akan berpengaruh kepada calon pengguna baru, baik ulasan positif maupun ulasan negatif. Dengan memahami sentimen tersebut, pengembang dapat merespon lebih cepat tentang umpan balik sehingga dapat meningkatkan kualitas layanan dan mempertahankan pengguna. Selain itu, analisis sentimen juga dapat digunakan untuk penyesuaian promosi aplikasi pada negara Indonesia, karena perbedaan kebutuhan dan preferensi pengguna pada setiap region. Secara keseluruhan, analisis sentimen ulasan pengguna di Indonesia tidak hanya memberikan masukan berharga bagi pengembang dalam menyelesaikan masalah teknis dan non-teknis, tetapi juga

membantu meningkatkan pengalaman bermain yang lebih baik bagi komunitas pemain yang sangat besar dan berpengaruh di Indonesia.

Tingginya popularitas aplikasi ini mendorong munculnya banyak ulasan atau *review* dari pengguna di berbagai platform, seperti *Google Play Store*. Platform tersebut merupakan layanan android market dalam bentuk software sebagai tempat bagi pengguna (user) untuk mengunduh aplikasi, *game*, dan sebagainya untuk sistem operasi *Android*. *Play store* menyediakan fitur ulasan berupa teks sebagai tempat bagi para pengguna aplikasi untuk memberikan ulasan atau opini mereka terhadap kinerja aplikasi yang digunakan. Ulasan tersebut berperan signifikan dalam menilai tingkat kepuasan pengguna sekaligus memberikan masukan berharga bagi pengembang untuk menyempurnakan kualitas aplikasi. Ulasan yang diberikan oleh pengguna umumnya merupakan cerminan pengalaman saat menggunakan aplikasi. Dalam konteks aplikasi populer seperti *Mobile Legends*, ulasan yang diberikan pengguna dapat memberikan gambaran lebih mendalam mengenai sentimen pengguna dalam skala besar. Dengan tingginya jumlah ulasan yang tersedia, diperlukan metode yang tepat untuk menganalisis opini tersebut secara sistematis. Analisis sentimen adalah proses memahami dan mengolah data tekstual secara otomatis untuk mendapatkan informasi. Analisis sentimen dilakukan untuk mendeteksi opini terhadap suatu subjek dan objek (misalnya individu, organisasi ataupun produk) dalam sebuah kumpulan data [6]. *Sentiment Analysis* merupakan sebuah penggambaran polaritas pada suatu teks atau kata. Jadi bisa disimpulkan bahwa sentimen analisis adalah pengolahan bahasa dari suatu teks atau kata yang dapat dianalisa dari pendapat, sikap atau penilaian dari seseorang terhadap suatu topik atau pembicaraan[7].

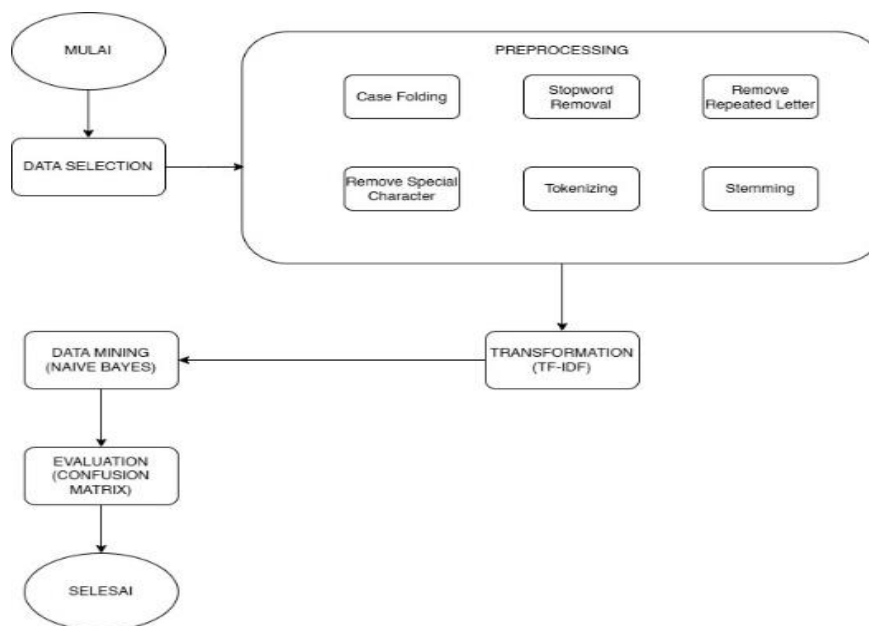
Penelitian sebelumnya yang mengangkat analisis sentimen mobile legends belum ada yang menggunakan ulasan *google playstore* melainkan menggunakan komentar *twitter* seperti yang dilakukan oleh Deden Kusnanda dan Angga Permana pada penelitian yang berjudul "*Implementation of Naive Bayes Classifier (NBC) for Sentiment Analysis on Twitter in Mobile Legends*" dimana dalam penelitian tersebut menggunakan komentar yang dimana dapat dianalisis dengan baik menggunakan metode Naive Bayes[8]. Pada penelitian yang dilakukan Putra Aditya, Afifah Azzahra dan Andri Wijaya yang berjudul "Analisis Sentimen Pemain Subway Surf Melalui Metode Naive Bayes Menurut Ulasan Play Store" menyimpulkan bahwa hasil akhir analisis sentimen menggunakan naive bayes berada pada angka 82% [9]. Penelitian yang dilakukan oleh Aluisius Dwiki Adhi Putra dan Safitri Juanita yang berjudul "Analisis Sentimen Pada Ulasan Pengguna Aplikasi Bibit Dan Bareksa Dengan Algoritma KNN" menunjukkan bahwa berdasarkan proses klasifikasi model dengan algoritma k-nearest neighbors, menggunakan model split data dengan perbandingan 60:40 terhadap dataset dengan jumlah sample 2.061 ulasan pengguna pada aplikasi bibit dan bareksa, maka nilai *accuracy*, *precision*, dan *recall* yang dihasilkan sebesar 85,14% , 91,91%, dan 76,44% untuk bibit sedangkan untuk bareksa yaitu 81,70% , 87,15%, 75,73%[10]. Pada penelitian yang dilakukan oleh Andriani Nurian dan Betha Nurina Sari yang berjudul "Analisis Sentimen Ulasan Pengguna Aplikasi Google Play Menggunakan Naive Bayes" menunjukkan bahwa Hasil *accuracy* tertinggi adalah klasifikasi menggunakan *Naive Bayes Classifier* dengan seleksi fitur *TF-IDF* yaitu sebesar akurasi 85%, presisi 79%, *recall* 85%, dan *F1-Score* 80%, data testing yang digunakan sebanyak 200 data atau 20% dari 3000 data, dari jumlah data yang digunakan dengan metode random pada saat testing[11]. Pada penelitian yang dilakukan oleh Hilmy Zhafran Muflih, Allif Rizki Abdillah dan Firman Noor Hasan menunjukkan bahwa hasil akurasi pengujian algoritma *naive bayes* yang didapatkan dari proses evaluasi dengan *confusion matrix* menunjukkan bahwa akurasi algoritma *naive bayes* adalah sebesar 76,67%. Pada proses evaluasi sentimen positif didapatkan nilai *precision* mencapai 75,41% dengan *recall* sebesar 77,97%, sementara untuk sentimen negatif mendapatkan nilai *precision* 77,97 % dengan *recall* sebesar 75,41%[12]. Pada Penelitian yang dilakukan oleh Kairil Anwar yang berjudul "Analisa sentimen Pengguna Instagram Di Indonesia Pada Review Smartphone Menggunakan Naive Bayes" menunjukkan bahwa hasil pengujian analisa sentimen pada komentar menggunakan *Gaussian Naive Bayes* bisa menghasilkan akurasi sebesar 73% sementara dengan menggunakan *Multinomial Naive Bayes* menghasilkan akurasi sebesar 81%[13].

Penelitian yang telah dilakukan menunjukkan bahwa model *Naive Bayes* menunjukkan akurasi yang baik untuk digunakan dalam analisis sentimen. Algoritma ini mengasumsikan bahwa fitur-fitur dalam ulasan (seperti kata-kata) adalah independen satu sama lain, sehingga memudahkan dalam perhitungan probabilitas. Meskipun asumsi ini mungkin tidak selalu akurat, *Naive Bayes* tetap menjadi salah satu algoritma yang efektif dan efisien dalam analisis sentiment. Algoritma *Naive Bayes* memiliki kecenderungan untuk memperlakukan kata-kata secara terpisah tanpa mempertimbangkan konteks dengan kedalaman yang cukup, sehingga berpotensi menghasilkan hasil yang tidak selalu akurat ketika mengidentifikasi sentimen yang bersifat kompleks, seperti ironi, sarkasme, atau makna ganda yang sering muncul dalam ulasan[14]. Tujuan dari penelitian ini adalah untuk menambah wawasan mengenai aplikasi *Mobile Legends*, dimana pada penelitian sebelumnya belum ada yang menggunakan ulasan pengguna melalui *Google Play Store*. Penelitian ini diharapkan dapat membantu dalam memahami persepsi dan pengalaman pemain permainan tersebut, serta memberikan wawasan untuk pengembang game dan komunitas untuk meningkatkan pengalaman permainan serta mengatasi permasalahan yang terdapat dalam aplikasi.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Pada penelitian ini menggunakan metode *KDD* (*Knowledge Discovery in Data*) karena memiliki keunggulan proses pengidentifikasian pola yang terorganisir dari sekumpulan data yang kompleks, sehingga datanya mudah untuk dipahami[15]. Gambar 1 menunjukkan alur penelitian yang menerapkan metode *KDD*.



Gambar 1 Alur Penelitian

Alur Penelitian pada Gambar 1 dijelaskan sebagai berikut

#### 2.1.1 Data Selection

Data selection merupakan tahapan pertama dalam *KDD*, pada tahap ini akan dilakukan pengumpulan dan pelabelan data. Data dikumpulkan dari situs *Google Play Store*.

#### 2.1.2 Preprocessing

Pada tahap preprocessing ini, data mentah yang telah terkumpul akan diolah menjadi data yang dapat digunakan pada tahap selanjutnya. Tahap preprocessing terdiri dari 6 proses yaitu:

- a. *Case Folding*  
Proses mengubah semua karakter huruf menjadi huruf kecil[16].
- b. *Stopword Removal*  
*Stopword* atau kata-kata penghenti biasanya memiliki arti yang kecil dan tidak menambah banyak nilai dalam tugas klasifikasi[17]. Kata-kata penghenti dapat mencakup kata-kata, seperti “yang”, “dan”, “dari”, dan lain-lain.
- c. *Remove Repeated Character*  
Bertujuan untuk mendeteksi karakter berulang dalam sebuah string dan mereduksinya menjadi satu karakter saja.
- d. *Removing Punctuation and Special Character*  
Tanda baca membantu menyampaikan makna, namun terlalu banyak tanda baca dapat membingungkan pembaca dan mesin. Karakter khusus (seperti #, \$, %, dll.) sering kali tidak memberikan nilai tambah pada analisis kami dan dapat memberikan hasil yang menyesatkan. Semakin banyak karakter yang tidak perlu (atau “noise”) yang kita miliki dalam data, semakin sulit bagi algoritma untuk mempelajari pola. Membersihkan teks membuat model kami berjalan lebih cepat dan akurat[18].
- e. *Tokenizing*  
Tokenisasi adalah teknik praproses dalam pemrosesan bahasa alami (NLP). *Tokenizing* memecah data teks tidak terstruktur menjadi unit-unit lebih kecil yang disebut token. Token tunggal dapat berkisar dari satu karakter atau satu kata hingga unit tekstual yang jauh lebih besar[19].
- f. *Stemming*  
Stemming adalah teknik pra-pemrosesan teks dalam pemrosesan bahasa alami (NLP). Secara khusus, ini adalah proses mereduksi bentuk kata yang disesuaikan menjadi apa yang disebut “*stem*”, atau bentuk akar, yang juga dikenal sebagai “*lemma*” dalam linguistik.



### 2.1.3 Transforming

Tahap transformation adalah tahap mengubah data menjadi bentuk yang dapat diolah pada tahapan machine learning[20]. Pada tahap ini akan dilakukan pembagian data ke dalam 3 skenario terlebih dahulu yaitu skenario 1 (90% data training dan 10% data testing), skenario 2 (80% data training dan 10% data testing), skenario 3 (70% data training dan 30% data testing). Selanjutnya, proses pembobotan kata akan dilakukan menggunakan metode TF-IDF, yang berfungsi untuk mengubah data teks menjadi vektor dengan bobot. Perhitungan metode TF-IDF dapat dijelaskan melalui rumus berikut.

$$TF\ IDF_{t,d} = 1 + \log(FT_{t,d}) * \log \frac{N}{DF_T} \tag{1}$$

### 2.1.4 Data Mining

Dalam manajemen data, data mining menjadi contoh kongret utama dalam menjalankan suatu aplikasi, dengan data mining dapat dilakukan, analisis terhadap data yang telah terkumpul. Data mining sangat berhubungan erat dengan analisis data untuk mencari pola dan kesamaan dalam sekumpulan data dengan teknik atau metode tertentu. pada penelitian ini metode yang digunakan adalah metode naive bayes karena performanya yang kompetitif dalam proses klasifikasi walaupun menggunakan atribut independent. Naive bayes merupakan metode statistik yang sederhana dan memiliki akurasi yang baik serta error rate yang minimum dalam proses pengklasifikasian[21] Naïve bayes disebut juga multinomial naïve bayes merupakan metode klasifikasi yang memanfaatkan metode probabilitas dan statistik. Metode ini merupakan model penyederhanaan dari algoritma bayes yang cocok dalam pengklasifikasian teks atau dokumen[22]. Pada tahap data mining akan dilakukan proses klasifikasi sentimen pada data ulasan dengan menggunakan algoritma *Naive Bayes*. Berikut adalah rumus algoritma *Naive Bayes*.

$$P(A|B) = \frac{P(B|A)*P(A)}{P(B)} \tag{2}$$

Dimana  $P(A|B)$  = Probabilitas bersyarat A yang diberikan oleh B,  $P(B|A)$  = Probabilitas bersyarat B yang diberikan oleh A,  $P(A)$  = Probabilitas kejadian A,  $P(B)$  = Probabilitas kejadian B.  $P(A|B)$  adalah probabilitas bersyarat dari kejadian A yang terjadi, diberikan bahwa kejadian B sudah terjadi, sedangkan  $P(B|A)$  adalah probabilitas bersyarat dari kejadian B yang terjadi, dengan asumsi bahwa kejadian A sudah terjadi.  $P(A)$  adalah probabilitas terjadinya kejadian A secara keseluruhan, tanpa mempertimbangkan kejadian B, dan  $P(B)$  adalah probabilitas terjadinya kejadian B secara keseluruhan, tanpa mempertimbangkan kejadian A.

### 2.1.5 Evaluation

Tahap evaluasi akan menggunakan *confusion matrix*. Metrik yang diekstraksi dari confusion matrix seperti *precision*, *recall*, dan *F1 score* untuk setiap kelas dan data mikro, makro, dan beban rata-rata dari semua kelas[23], digunakan untuk mengukur kinerja keseluruhan skenario.

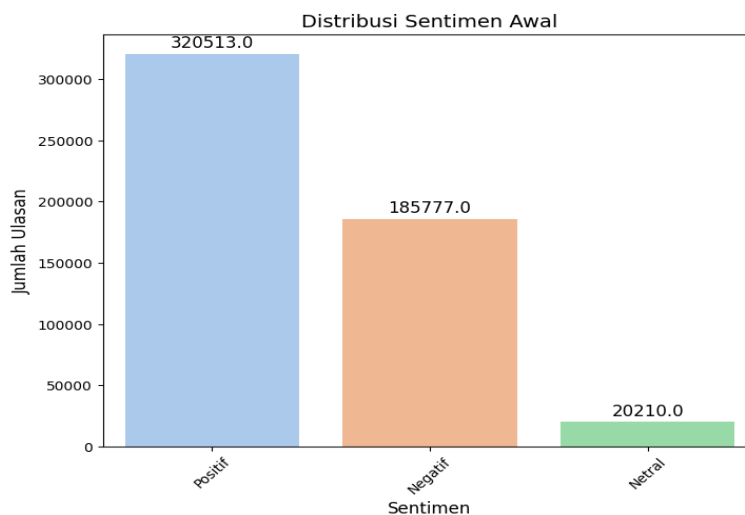
## 3. HASIL DAN PEMBAHASAN

### 3.1 Pengumpulan Data

Menggunakan alat *Google Play Scraper* untuk secara otomatis mengumpulkan data dari *Google Play Store* adalah cara yang efektif dan efisien untuk mendapatkan berbagai informasi aplikasi, seperti detail aplikasi, ulasan pengguna, penilaian, jumlah unduhan, dan informasi penting lainnya. Dengan alat ini, data dalam jumlah besar dapat diakses dan diekstrak secara otomatis, lalu dianalisis untuk mendapatkan wawasan berharga tentang kinerja aplikasi, tren pengguna, dan umpan balik konsumen. Pada proses scraping kali ini, berhasil dikumpulkan sebanyak 220.500 data. Data yang digunakan dalam penelitian berupa ulasan dan rating, dengan kondisi jika rating 1-2 maka ulasan akan mendapat label negatif. Jika rating 3 maka ulasan mendapat label netral. Jika rating 4-5 maka ulasan akan mendapatkan label positif. Proses pelabelan dilakukan secara otomatis menggunakan fungsi bawaan pada *python*. Pada proses ini didapatkan 132393 ulasan positif, 79540 ulasan negatif dan 8567 ulasan netral. Hasil Sampel Dataset akan dijabarkan pada Tabel 1.

Tabel 1. Sampel Dataset

Rating	Ulasan	Sentimen
2	Game ini sangat buruk	Negatif
1	Kurang enak kalo main solo	Negatif
1	Maaf ya saya kasih bintang 1, masih banyak kurangnya	Negatif
5	moonton gantengggg	Positif



Gambar 2 Distribusi Sentimen

Pada Gambar 2, terlihat adanya class imbalance, dengan jarak yang sangat jauh antara kelas Netral dan kelas Mayoritas (Positif dan Negatif). Hal ini mencerminkan bahwa jumlah ulasan dengan sentimen Netral jauh lebih sedikit dibandingkan dengan ulasan Positif dan Negatif. Ketidakseimbangan ini berpotensi mempengaruhi performa model, terutama dalam mendeteksi ulasan netral, yang sudah terbukti memiliki performa buruk berdasarkan hasil evaluasi model sebelumnya. Dalam kondisi ini, kelas Netral sengaja dihapus dari dataset untuk memfokuskan analisis hanya pada kelas Positif dan Negatif. Setelah itu, dilakukan undersampling pada kelas mayoritas (Positif dan Negatif) untuk menyetarakan jumlah kedua kelas tersebut. Langkah ini bertujuan untuk mengurangi ketidakseimbangan yang ada dan memberikan kesempatan yang lebih adil bagi model untuk mempelajari pola-pola pada kedua kelas tersebut.

### 3.2 Text Preprocessing

Langkah pertama sebelum memulai pengolahan data adalah mengidentifikasi dan menghapus kolom-kolom yang tidak relevan atau tidak diperlukan dalam dataset. Hal ini dilakukan untuk membersihkan dataset dari informasi yang tidak berkontribusi pada analisis atau pemodelan yang akan dilakukan. Tujuan utamanya adalah memastikan bahwa data yang diproses hanya berisi informasi yang relevan dan sesuai dengan kebutuhan penelitian.

#### 3.2.1 Case Folding

Pada langkah selanjutnya, semua data komentar yang masih menggunakan huruf kapital akan diubah menjadi huruf kecil guna memastikan konsistensi dalam data yang akan diproses. Implementasi *case folding* pada kali ini digunakan fungsi bawaan *python* yaitu *lower*, sehingga tidak memerlukan *library* tambahan. Pada Tabel 2 ulasan telah dilakukan *case folding*, yaitu membuat semua huruf dalam kalimat menjadi huruf kecil.

Tabel 2. Sampel Dataset Case Folding

Ulasan	Case Folding
Game ini sangat buruk	game ini sangat buruk
Kurang enak kalo main solo	kurang enak kalo main solo
Maaf ya saya kasih bintang 1, masih banyak kurangnya moonton gantengggg	maaf ya saya kasih bintang, 1 masih banyak kurangnya moonton gantengggg

#### 3.2.2 Tokenizing

Proses tokenizing adalah proses untuk memecah suatu kalimat menjadi kata berdasarkan suatu karakter tertentu. Proses tokenizing sangat penting peranannya karena sebagai dasar penggunaan algoritma TF-IDF dimana dalam proses TF dilakukan pengelompokan dan perhitungan kata. Karakter biasa dijadikan pemisah yaitu spasi, titik (.), koma (,), semicolon(;), dan titik dua (:)[24]. Pada tabel 3 adalah proses tokenisasi dari penerapan naive bayes pada aplikasi mobile legend.

Tabel 3. Sampel Dataset Tokenizing

Ulasan	Tokenizing
game ini sangat buruk	“game”, “ini”, “sangat”, “burik”
kurang enak kalo main solo	“kurang”, “enak”, “kalo”, “main”, “solo”
maaf ya saya kasih bintang, 1 masih banyak kurangnya	“maaf”, “ya”, “saya”, “kasih”, “bintang”, “1” “masih”, “banyak”, “kurangnya”
moonton gantengggg	“moonton”, “gantengggg”



Tokenisasi adalah proses untuk membagi teks yang dapat berupa kalimat, paragraf atau dokumen, menjadi token-token/bagian-bagian tertentu. Pada penelitian ini menggunakan fungsi tokenisasi yang berasal dari *library* NLTK (Natural Language Toolkit). Library ini populer untuk pengolahan bahasa alami (NLP) di *Python*.

### 3.2.2 Stopwords Removal

Proses Stopwords Removal (penghapusan stopwords) merupakan salah satu tahapan penting dalam pengolahan teks atau text preprocessing. Hal ini sejalan dengan Stopword removal adalah proses penghapusan kata-kata yang tidak memiliki makna penting pada tweet, seperti kata ‘yang’, ‘di’, ‘ke’, ‘dengan’, dan lain-lain agar kata-kata yang tersisa dalam kumpulan tweet hanyalah kata yang memiliki makna penting saja[25]. Tujuan utama dari langkah ini adalah untuk menghilangkan kata-kata yang sering muncul tetapi tidak memberikan kontribusi signifikan terhadap pemahaman konteks atau analisis teks, seperti kata sambung atau kata umum lainnya, Hal ini bertujuan untuk menyederhanakan kalimat dan meningkatkan pesan yang disampaikan[26]. Berikut adalah contoh sampel dataset pada Tabel 4 sebelum dan setelah dilakukan penghapusan stopwords:

Tabel 4. Sampel Dataset *Stopwords Removal*

Ulasan	<i>Stopwords Removal</i>
“game”, “ini”, “sangat”, “burik”	“game” “burik”
“kurang”, “anak”, “kalo”, “main”, “solo”	“anak” “kalo” “main” “solo”
“maaf”, “ya”, “saya”, “kasih”, “bintang”, “1”	“maaf” “ya” “kasih” “bintang” “1”, “kurangnya”
“masih”, “banyak”, “kurangnya”	
“moonton”, “gantengggg”	“moonton” “gantengggg”

Proses *stopwords removal* adalah langkah dalam pemrosesan teks yang bertujuan untuk menghapus kata-kata umum yang tidak memiliki makna signifikan dalam analisis, seperti “dan”, “atau”, “yang”, dan “dengan”. Kata-kata ini biasanya tidak memberikan kontribusi penting dalam memahami konteks atau makna utama dari teks. Langkah ini akan menghilangkan seluruh kata-kata umum yang sering digunakan, tetapi tidak memiliki makna jika berdiri sendiri. Skrip untuk menghilangkan *stopwords* pada *python* menggunakan fungsi bawaan *python*, *list comprehension* untuk memfilter token dengan menggunakan *library* NLTK untuk mendeteksi *stopwords*.

### 3.2.3 Remove Repeated Character

Proses hapus karakter berulang adalah langkah penting dalam text preprocessing yang bertujuan untuk mengatasi masalah karakter yang diulang secara tidak perlu dalam teks, seperti dalam kata “gantengggg” yang seharusnya hanya “ganteng”. Tanpa penghapusan karakter berulang ini, analisis dapat menghasilkan token-token yang lebih panjang atau duplikat yang tidak memberikan informasi tambahan, dan bahkan dapat mengganggu performa model dalam pengolahan teks. Penghapusan stopwords ini bertujuan untuk menyederhanakan data teks dan meningkatkan kualitas analisis. Berikut adalah sampel dataset pada Tabel 5 sebelum dan setelah dilakukan penghapusan karakter berulang:

Tabel 5. Sampel Dataset *Remove Repeated Character*

Ulasan	Hapus Karakter Berulang
“game” “burik”	“game” “burik”
“anak” “kalo” “main” “solo”	“anak” “kalo” “main” “solo”
“maaf” “ya” “kasih”	“maaf” “ya” “kasih” “bintang” “1”, “kurangnya”
“bintang” “1”,	
“kurangnya”	“moonton” “ganteng”
“moonton” “gantengggg”	

Tanpa penghapusan ini, analisis bisa menghasilkan token-token yang lebih panjang atau tidak perlu. Seperti contoh pada tabel “gantengggg” dan “ganteng” akan dianggap dua karakter berbeda walaupun bermakna sama. langkah ini membantu menyederhanakan data teks dan memastikan bahwa model atau analisis dapat bekerja lebih efektif dan efisien. Skrip ini diterapkan pada setiap token, menghasilkan daftar token baru yang telah dihapus karakter berulang. Library yang digunakan adalah *regular expression*.

### 3.2.4 Removing Punctuation and Special Character

Proses menghapus tanda baca dan karakter khusus merupakan langkah penting dalam pembersihan data teks untuk memastikan bahwa analisis atau pemodelan machine learning dapat fokus pada informasi yang relevan. Tanda baca seperti koma, titik, atau karakter khusus lainnya sering kali tidak memberikan kontribusi langsung pada makna atau konteks teks. Oleh karena itu, penghapusan elemen-elemen ini akan mengurangi noise dalam data dan membantu model untuk lebih efektif memproses informasi yang lebih relevan. Sebagai contoh, pada tabel di bawah ini, tanda baca dan karakter khusus dihapus dari setiap token sehingga hanya kata-kata yang bermakna yang tersisa,

meningkatkan akurasi analisis teks atau model machine learning. Pada Tabel 6 adalah sampel dataset sebelum dan setelah dilakukan penghapusan tanda baca dan karakter khusus:

**Tabel 6.** Sampel Dataset *Removing Punctuation and Special Character*

Ulasan	Hapus <i>Punctuation and Special Character</i>
“game” “burik”	“game” “burik”
“enak” “kalo” “main” “solo”	“enak” “kalo” “main” “solo”
“maaf” “ya” “kasih”	“maaf” “ya” “kasih” “bintang” “kurangnya”
“bintang” “1”, “kurangnya”	“moonton” “ganteng”
“moonton” “ganteng”	

Untuk memastikan data lebih bersih dan relevan bagi model analisis teks atau pembelajaran mesin, dapat dilakukan penghapusan terhadap tanda baca dan karakter special. Dengan menghapus elemen-elemen yang tidak penting, dapat memastikan bahwa model akan fokus pada informasi meningkatkan akurasi analisis.

### 3.2.5 Stemming

*Stemming* adalah proses mengubah kata menjadi bentuk dasarnya atau akar kata, yang bertujuan untuk menyederhanakan variasi kata yang memiliki makna yang sama. Misalnya, kata "kurangnya" dapat diubah menjadi "kurang". Dalam pemrosesan teks, stemming sangat berguna untuk mengurangi kompleksitas data teks, sehingga model atau algoritma dapat lebih mudah memahami konteksnya.. Pada Tabel 7 adalah contoh dataset sebelum dan setelah dilakukan stemming:

**Tabel 7.** Sampel Dataset *Stemming*

Ulasan	<i>Stemming</i>
“game” “burik”	“game” “burik”
“enak” “kalo” “main” “solo”	“enak” “kalo” “main” “solo”
“maaf” “ya” “kasih”	“maaf” “ya” “kasih” “bintang” “kurang”
“bintang” “1”, “kurangnya”	“moonton” “ganteng”
“moonton” “ganteng”	“moonton” “ganteng”

Stemming adalah mengubah sebuah kata menjadi kata dasar, dengan menggunakan *library* Sastrawi. Sastrawi adalah perpustakaan NLP yang didedikasikan untuk bahasa Indonesia. Awalnya, Library dikembangkan dan ditujukan untuk bahasa pemrograman PHP, tetapi karena popularitasnya, perpustakaan ini juga dikembangkan untuk mendukung bahasa pemrograman Python [27].

### 3.2.6 M Joining Stemmed Token

Setelah melalui berbagai tahap pemrosesan teks, seperti penghapusan stopwords, penghapusan karakter berulang, dan stemming, data teks tersebut akan terdiri dari serangkaian token atau kata-kata yang telah diproses. Langkah selanjutnya adalah menggabungkan token-token tersebut menjadi teks kembali, agar kalimat dapat dianalisis dalam konteks yang lebih mudah dipahami. Pada Tabel 8 adalah contoh sampel dataset sebelum dan setelah penggabungan token menjadi teks:

**Tabel 8.** Sampel Dataset *Joining Stemmed Tokens*

Ulasan	Menggabungkan token menjadi teks
“game” “burik”	game burik
“enak” “kalo” “main” “solo”	enak kalo main solo
“maaf” “ya” “kasih”	maaf ya kasih bintang kurang
“bintang” “kurang”	
“moonton” “ganteng”	moonton ganteng

Setelah melalui semua tahap pemrosesan, token yang telah diproses hasil stemming akan menjadi daftar kata-kata lagi. Untuk evaluasi, token tersebut digabungkan kembali sebagai agar kalimat dapat dianalisis secara kontekstual. Tidak ada *library* eksternal pada skrip penggabungan token menjadi teks kembali, proses ini dapat dilakukan menggunakan fungsi bawaan *python*.

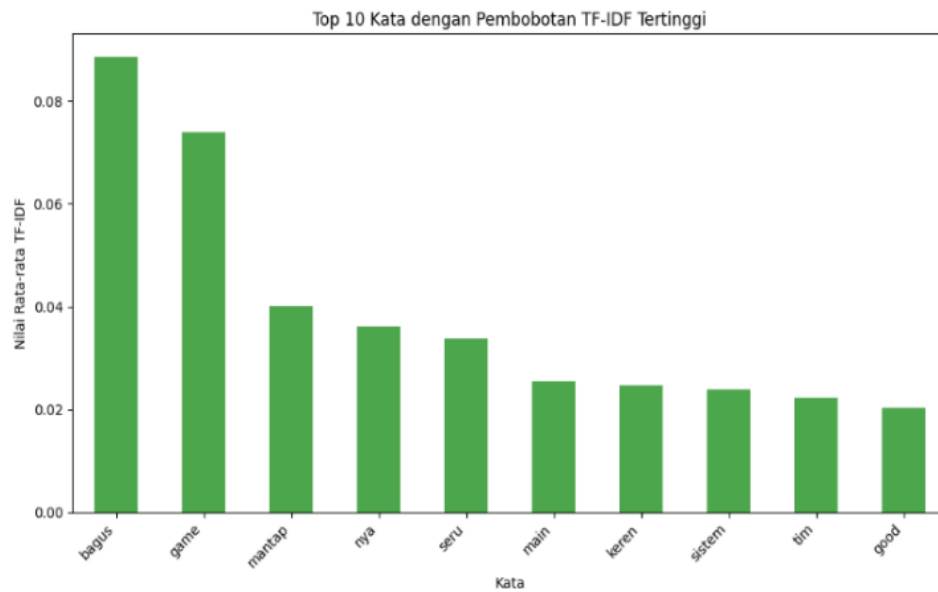
## 3.2 Transformation

Tahap transformation ini diawali dengan melakukan pembagian data menjadi data training dan data testing dalam 3 skenario. Pembagian data ini menggunakan fungsi train test split dengan parameter `random_state=42`.

**Tabel 9.** Pembagian Skenario Dataset

Skenario Rasio Perbandingan	Data Training	Data Testing
80:20	397059	99265

Tabel 9 menunjukkan Rasio perbagian data latih dan data uji. Setelah pembagian data menjadi data pelatihan dan data pengujian, langkah berikutnya adalah penerapan pembobotan kata menggunakan metode TF-IDF. Gambar 2 menunjukkan contoh hasil pembobotan kata tertinggi menggunakan TF-IDF pada data pelatihan untuk skenario 1 (80:20).



**Gambar 3** Pembobotan Kata menggunakan TF-IDF

Grafik pembobotan TF-IDF pada Gambar 2 menunjukkan kata-kata yang memiliki nilai TF-IDF tertinggi. Kata-kata ini dianggap lebih penting dalam konteks data yang dianalisis. Kata-kata seperti "bagus," "game," "mantap," hingga "good" berada di daftar teratas. Semakin tinggi nilai TF-IDF, semakin signifikan kata tersebut dalam membedakan data dari yang lain. Menurut grafik, kata "bagus" memiliki bobot TF-IDF tertinggi, menunjukkan bahwa kata tersebut sangat penting dalam konteks data ini, karena frekuensinya yang tinggi.

### 3.3 Data Mining

Pada tahap ini dilakukan visualisasi dan klasifikasi sentimen dengan tiga skenario pembagian data yang telah dilakukan menggunakan pada algoritma Naive Bayes. Hasil visualisasi frekuensi kata yang muncul terdapat pada Gambar 4 dan Gambar 5. Dan visualisasi confusion matrix setiap skenario akan dipaparkan pada Gambar 6.



**Gambar 4.** Visualisasi Frekuensi Kata Kelas Positif

Berdasarkan *wordcloud* yang terlihat pada Gambar 4, menggambarkan bahwa aplikasi *Mobile legend* mendapatkan apresiasi dari pemainnya, terutama terkait *gameplay*, kualitas dan pengalaman bermain



Pada Tabel 10, dipaparkan hasil confusion matrix dan classification report. Nilai akurasi model menunjukkan di angka 88%. Performa pada kelas Negatif menunjukkan precision yang tinggi dengan nilai 84%, yang menunjukkan model cukup baik dalam mengidentifikasi ulasan negatif. Pada recall, model dapat menangkap 92% ulasan negatif, sehingga dapat menangkap sebagian besar ulasan yang benar-benar negatif. Dengan F1-Score pada angka 0.88, model menunjukkan keseimbangan yang baik antara precision dan recall untuk kelas negatif. Pada kelas Positif, model menghasilkan precision sebesar 91%, yang menunjukkan prediksi positif cukup akurat. Model juga memiliki recall yang baik dengan angka 83%, dapat menangkap 83% dari semua ulasan positif. F1-Score pada kelas positif sebesar 0.87 menunjukkan keseimbangan yang baik antara precision dan recall.

Namun, pada dataset yang digunakan, merujuk pada visualisasi *wordcloud* pada Gambar 4 dan Gambar 5, terdapat kata yang muncul pada kedua kelas positif dan negatif. Pada dataset yang didapat, banyak ulasan negatif yang tidak berdasar terhadap pengalaman secara objektif, melainkan ulasan aplikasi tersebut digunakan untuk meluapkan kekesalan yang dihadapi oleh pemain, terutama jika ada lonjakan jumlah ulasan negatif yang tidak mencerminkan pengalaman pengguna yang biasa. Review bombing adalah praktik di mana sekelompok orang memberikan ulasan negatif secara masal untuk tujuan tertentu, yang seringkali tidak berhubungan dengan kualitas produk atau pengalaman pengguna yang sebenarnya. Ini bisa disebabkan oleh isu politik, kebijakan dalam permainan, atau perubahan lainnya yang memicu ketidakpuasan dalam komunitas. Pada kasus ini, ulasan negatif dapat mempengaruhi hasil pada penilaian sentimen ulasan. Algoritma analisis sentimen dapat menilai ulasan dengan cara yang tidak akurat jika banyak ulasan negatif yang tidak relevan. Jika model dilatih menggunakan dataset yang terdampak *review bombing*, model dapat mengalami kesalahan dalam mendeteksi sentimen. Serangan ulasan negatif juga dapat menyebabkan perubahan ketidak tetapan sentimen, membuat aplikasi terlihat tidak stabil dalam hal kepuasan pengguna, walaupun hal tersebut hanya serangan ulasan negatif yang tidak mencerminkan keseluruhan kualitas aplikasi. Ulasan sarkasme atau ironi juga dapat mempengaruhi kinerja model *machine learning*. Sarkasme bergantung kepada konteks, nada atau pengetahuan tambahan yang sulit untuk ditangkap oleh algoritma. Algoritma juga menggunakan pembobotan TF-IDF, hal ini membuat sarkasme sulit untuk dideteksi, karena makna positif atau negatif dapat dibalik dari makna sebenarnya. Jika dataset pelatihan model tidak banyak dengan sampel sarkasme atau ulasan ironis, akurasi analisis sentimen akan menurun. Variasi emoji juga ditemui dalam dataset ulasan, namun *emoji* dihapus dalam *preprocessing* untuk menyederhanakan proses evaluasi dan meningkatkan kinerja dataset non-*emoji*. Karena emoji dihapus, adapun dampak negatif seperti ulasan kehilangan makna sesungguhnya, karena *emoji* sering digunakan untuk menyampaikan emosi secara langsung. Menghapus emoji dapat membuat model kehilangan petunjuk penting tentang sentimen pengguna, terutama jika ulasan memiliki sedikit teks tetapi menggunakan emoji untuk menggambarkan sentimen. Jika model yang didapatkan dalam penelitian ini di gunakan pada data yang menggunakan emoji atau sebaliknya, model akan mengalami penurunan performa. Pada Gambar 8 telah dipaparkan perbandingan jumlah ulasan positif, negatif dan netral. Persebaran kelas ulasan yang tidak seimbang (*class imbalance*) dapat mempengaruhi kinerja model *machine learning*. Masalah yang muncul jika ulasan tidak seimbang adalah bias model terhadap kelas mayoritas. Model cenderung lebih akurat dalam memprediksi ulasan positif dan negatif, sementara ulasan netral sering terabaikan. Mengingat bahwa kelas netral dihilangkan dalam skenario ini, model akan difokuskan hanya pada dua kelas, positif dan negatif. Ketika jumlah ulasan positif dan negatif disetarakan, model dapat lebih fokus pada dua kelas tersebut, meningkatkan akurasi untuk kedua kelas ini, namun dengan risiko mengabaikan nuansa sentimen yang mungkin ada pada ulasan netral. Dengan menyeimbangkan jumlah data positif dan negatif, model dapat mempelajari pola-pola dengan lebih baik pada kedua kelas tersebut, sehingga dapat memprediksi dengan lebih akurat ulasan positif dan negatif. Dengan mengurangi jumlah data positif, dataset kehilangan informasi yang mungkin bermanfaat untuk model. Ini dapat mengurangi kekuatan model dalam memahami pola sentimen positif. Namun, hal ini juga berarti bahwa potensi informasi yang diberikan oleh ulasan netral, yang dapat menggambarkan ambiguitas atau ketidakpastian dalam opini, akan hilang. Dalam konteks ulasan game seperti *Mobile Legends*, ulasan netral bisa berarti pengalaman bermain yang "cukup memuaskan" tanpa memberikan kesan yang sangat mengesankan atau mengecewakan.

#### 4. KESIMPULAN

Berdasarkan penelitian yang dilakukan menggunakan implementasi model *machine learning naive bayes* pada ulasan aplikasi *Mobile Legends*, dapat disimpulkan bahwa Performa model *naive bayes* sangat baik dari ketiga skenario mendapatkan skor akurasi 87%. Kesalahan pada False Negative lebih besar dibandingkan False Positive, yang menunjukkan bahwa model lebih cenderung melewatkan data positif daripada salah mendeteksi data negatif sebagai positif. Setelah diterapkan undersampling, hasil precision dan recall untuk kedua kelas (Negatif dan Positif) menunjukkan hasil yang cukup baik, dengan precision masing-masing mencapai 84% untuk kelas Negatif dan 91% untuk kelas Positif. Hal ini menunjukkan bahwa model mampu memberikan prediksi yang cukup akurat setelah data disetarakan. Algoritma *machine learning* memiliki keterbatasan dalam mendeteksi ulasan yang bersifat sarkasme dan ironi karena model *naive bayes* mengabaikan konteks. Untuk mendeteksi ulasan yang bersifat ironi atau sarkasme dibutuhkan model berbasis *deep learning* yang lebih canggih. Karena tahap *preprocessing* menghilangkan emoji, maka hasil evaluasi akan kesulitan jika diterapkan pada dataset yang menggunakan emoji untuk mengungkapkan sentimen.



## REFERENCES

- [1] B. Adikara, “Jumlah Gamer di Indonesia Terus Meningkat, Diperkirakan Capai 192,1 Juta Orang di 2025, Salah Satu yang Paling Banyak Adalah Pemain Mobile Legends - Jawa Pos.” Accessed: Dec. 21, 2024. [Online]. Available: <https://www.jawapos.com/hobi-kesenangan/015230108/jumlah-gamer-di-indonesia-terus-meningkat-diperkirakan-capai-192-1-juta-orang-di-2025-salah-satu-yang-paling-banyak-adalah-pemain-mobile-legends>
- [2] G. A. Wijaya and F. Santi, “The Influence of the Mobile Legends E-Sport Phenomenon on Student Financial Behavior,” *Jurnal Informatika Ekonomi Bisnis*, pp. 280–285, 2024, doi: <https://doi.org/10.37034/infec.v6i1.852>.
- [3] A. R. Hidayat, “Tinjauan Ekonomi Islam Terhadap Jual Beli Online Account Game Mobile Legends: Bang Bang Dalam Tinjauan Fiqih Muamalah,” *Jurnal Syntax Admiration*, vol. 1, no. 1, pp. 13–22, 2020, doi: <https://doi.org/10.46799/jsa.v1i1.2>.
- [4] D. Kusnanda and A. Permana, “Implementation of Naive Bayes Classifier (NBC) for Sentiment Analysis on Twitter in Mobile Legends,” *International Journal of Science, Technology & Management*, vol. 4, no. 5, pp. 1132–1138, 2023, doi: <https://doi.org/10.46729/ijstm.v4i5.935>.
- [5] C. Chapple, “Mobile Legends Revenue Passes \$500 Million as Southeast Asia Powers Explosive Growth,” Sensor Tower. Accessed: Dec. 12, 2024. [Online]. Available: <https://sensortower.com/blog/mobile-legends-revenue-500-million>
- [6] C. F. Hasri and D. Alita, “Penerapan Metode Naïve Bayes Classifier Dan Support Vector Machine Pada Analisis Sentimen Terhadap Dampak Virus Corona Di Twitter,” *Jurnal informatika dan rekayasa perangkat lunak*, vol. 3, no. 2, pp. 145–160, 2022, doi: <https://doi.org/10.33365/jatika.v3i2.2026>.
- [7] W. Khofifah, D. N. Rahayu, and A. M. Yusuf, “Analisis Sentimen Menggunakan Naive Bayes Untuk Melihat Review Masyarakat Terhadap Tempat Wisata Pantai Di Kabupaten Karawang Pada Ulasan Google Maps,” *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 16, no. 4, pp. 171–180, 2022, doi: <https://doi.org/10.35969/interkom.v16i4.192>.
- [8] D. Kusnanda and A. Permana, “Implementation of Naive Bayes Classifier (NBC) for Sentiment Analysis on Twitter in Mobile Legends,” *International Journal of Science, Technology & Management*, vol. 4, no. 5, pp. 1132–1138, 2023, doi: <https://doi.org/10.46729/ijstm.v4i5.935>.
- [9] P. Aditya, A. Azzahra, and A. Wijaya, “Analisis Sentimen Pemain Subway Surf Melalui Metode Naive Bayes Menurut Ulasan Play Store,” *Jurnal Ilmiah Sistem Informasi*, vol. 3, no. 2, pp. 267–275, 2023.
- [10] A. D. A. Putra and S. Juanita, “Analisis Sentimen pada Ulasan pengguna Aplikasi Bibit Dan Bareksa dengan Algoritma KNN,” *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 8, no. 2, pp. 636–646, 2021, doi: <https://doi.org/10.35957/jatisi.v8i2.962>.
- [11] A. Nurian, “Analisis Sentimen Ulasan Pengguna Aplikasi Google Play Menggunakan Naïve Bayes,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 3s1, 2023, doi: <http://dx.doi.org/10.23960/jitet.v11i3s1.3348>.
- [12] H. Z. Muflih, A. R. Abdillah, and F. N. Hasan, “Analisis Sentimen Ulasan Pengguna Aplikasi Ajaib Menggunakan Metode Naïve Bayes,” *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 4, no. 3, pp. 1613–1621, 2023, doi: <https://doi.org/10.30865/klik.v4i3.1303>.
- [13] K. Anwar, “Analisa sentimen Pengguna Instagram Di Indonesia Pada Review Smartphone Menggunakan Naive Bayes,” *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 2, no. 4, pp. 148–155, 2022, doi: <https://doi.org/10.30865/klik.v2i4.315>.
- [14] A. Nuraini, A. Faqih, G. Dwilestari, N. D. Nuris, and R. Narasati, “Analisis Sentimen Terhadap Review Aplikasi Brimo Di Google Play Store Menggunakan Algoritma Naïve Bayes,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 6, pp. 3661–3666, 2023, doi: <https://doi.org/10.36040/jati.v7i6.8228>.
- [15] M. D. Hendriyanto, A. A. Ridha, and U. Enri, “Analisis Sentimen Ulasan Aplikasi Mola Pada Google Play Store Menggunakan Algoritma Support Vector Machine,” *INTECOMS: Journal of Information Technology and Computer Science*, vol. 5, no. 1, pp. 1–7, 2022, doi: <https://doi.org/10.31539/intecoms.v5i1.3708>.
- [16] N. L. P. M. Putu and A. Z. Amrullah, “Analisis Sentimen dan Pemodelan Topik Pariwisata Lombok Menggunakan Algoritma Naive Bayes dan Latent Dirichlet Allocation,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 1, pp. 123–131, 2021, doi: <https://doi.org/10.29207/resti.v5i1.2587>.
- [17] IBM, “Using CountVectorizer for NLP feature extraction | IBM.” Accessed: Dec. 21, 2024. [Online]. Available: <https://www.ibm.com/reference/python/countvectorizer>
- [18] V. Kumar, “Day 7: Handling Punctuation and Special Characters: Cleaning Up Text in NLP!” Accessed: Dec. 21, 2024. [Online]. Available: <https://www.linkedin.com/pulse/day-7-handling-punctuation-special-characters-up-text-vinod-kumar-gr-0xosf>
- [19] J. Murel, “Tokenizing text in Python - IBM Developer.” Accessed: Dec. 21, 2024. [Online]. Available: <https://developer.ibm.com/tutorials/awb-tokenizing-text-in-python/>
- [20] R. Slamet, W. Gata, A. Novtariany, K. Hilyati, and F. A. Jariyah, “Analisis Sentimen Twitter Terhadap Penggunaan Artis Korea Selatan Sebagai Brand Ambassador Produk Kecantikan Lokal,” *INTECOMS: Journal of Information Technology and Computer Science*, vol. 5, no. 1, pp. 145–153, 2022, doi: <https://doi.org/10.31539/intecoms.v5i1.3933>.
- [21] D. Alita, “Multiclass SVM Algorithm for Sarcasm Text in Twitter,” *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, vol. 8, no. 1, pp. 118–128, 2021, doi: <https://doi.org/10.35957/jatisi.v8i1.646>.
- [22] F. Juliyanto and P. Parjito, “Rekayasa Aplikasi Manajemen E-Filling Dokumen Surat Pada Pt Alp (Atosim Lampung Pelayaran),” *Jurnal Teknologi Dan Sistem Informasi*, vol. 2, no. 1, pp. 43–49, 2021, doi: <https://doi.org/10.33365/jtsi.v2i1.741>.
- [23] M. E. Effendi, I. Yuadi, and I. Puspitasari, “Prediksi Guru Kemungkinan Tetap Bekerja di Sekolah Al Uswah Surabaya Menggunakan Machine Learning,” *Jurnal Informasi dan Teknologi*, pp. 129–137, 2023, doi: <https://doi.org/10.37034/jidt.v5i2.361>.
- [24] T. A. L. Jaya and M. Ayub, “Pengembangan Knowledge Management System dengan Teknik Information Retrieval,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 7, no. 1, 2021, doi: <https://doi.org/10.28932/jutisi.v7i1.3316>.
- [25] T. S. Sabrila, V. R. Sari, and A. E. Minarno, “Analisis Sentimen Pada Tweet Tentang Penanganan Covid-19 Menggunakan Word Embedding Pada Algoritma Support Vector Machine Dan K-Nearest Neighbor,” *Fountain of Informatics Journal*, vol. 6, no. 2, p. 69, 2021, doi: <http://dx.doi.org/10.21111/fij.v6i2.5536>.



- [26] A. C. Kamilla, N. Priyani, R. Priskila, and V. H. Pranatawijaya, “Analisis Sentimen Film Agak Laen Dengan Kecerdasan Buatan: Text Mining Metode Naïve Bayes Classifier,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 3, pp. 2923–2928, 2024, doi: <https://doi.org/10.36040/jati.v8i3.9587>.
- [27] M. U. Albab and M. N. Fawaiq, “Optimization Of The Stemming Technique On Text Preprocessing President 3 Periods Topic,” *Jurnal Transformatika*, vol. 20, no. 2, pp. 1–12, 2023, doi: <http://dx.doi.org/10.26623/transformatika.v20i2.5374>.