



PENERAPAN METODE NAÏVE BAYES UNTUK MENGIDENTIFIKASI SENTIMEN PENGGUNA PADA ULASAN APLIKASI REELSHORT DI GOOGLE PLAY STORE

Susilawati¹, Muhammad Iqbal²

¹susilawati@gmail.com, ²muhammadiqbal@dosen.pancabudi.ac.id

^{1,2}Magister Teknik Informatika, Universitas Pembangunan Pancabudi

Abstrak

Penelitian ini bertujuan untuk mengetahui pendapat pengguna aplikasi *ReelShort* berdasarkan ulasan mereka di *Google Play Store*. Metode yang digunakan adalah algoritma *Multinomial Naive Bayes*. Data ulasan pengguna dikumpulkan dengan teknik *scrapping* dari *Google Play Store*, selanjutnya dilakukan *pre-processing* data untuk membersihkan dan menyederhanakan teks. Setelah itu, sentimen dari setiap ulasan diklasifikasikan menjadi dua kategori yaitu positif dan negatif. Model klasifikasi dibangun menggunakan algoritma *Multinomial Naive Bayes* dan dievaluasi menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. Hasil penelitian menunjukkan bahwa model mampu mengklasifikasikan sentimen dengan akurasi sebesar 77%. Nilai presisi sebesar 83% mengindikasikan bahwa model cukup baik dalam mengidentifikasi ulasan yang benar-benar positif atau negatif. Namun, nilai *recall* yang sebesar 77% menunjukkan bahwa model masih melewatkan beberapa ulasan yang seharusnya terklasifikasi sebagai positif atau negatif. Analisis lebih lanjut menunjukkan bahwa model lebih baik dalam mengklasifikasikan ulasan negatif dibandingkan dengan ulasan positif. Hasil penelitian ini memberikan pemahaman yang lebih baik mengenai persepsi pengguna terhadap aplikasi *ReelShort* dan dapat dimanfaatkan untuk meningkatkan kualitas aplikasi di masa mendatang.

Kata Kunci: Analisis Sentimen, Naïve Bayes, Google Play Store, ReelShort, Klasifikasi Teks

Abstract

This study aims to determine users' opinions about the ReelShort application based on their reviews on the Google Play Store. The multinomial Naive Bayes algorithm was used as the method. User review data was collected using a scraping technique from the Google Play Store, followed by data pre-processing to clean and simplify the text. Afterward, the sentiment of each review was classified into two categories: positive and negative. A classification model was built using the multinomial Naive Bayes algorithm and evaluated using accuracy, precision, recall, and F1-score metrics. The results showed that the model could classify sentiment with an accuracy of 77%. A precision value of 83% indicates that the model is quite good at identifying reviews that are truly positive or negative. However, a recall value of 77% shows that the model still misses some reviews that should be classified as positive or negative. Further analysis showed that the model is better at classifying negative reviews than positive ones. The results of this study provide a better understanding of user perceptions of the ReelShort application and can be used to improve the application's quality in the future.

Keywords: Sentimen Analysis, Naïve Bayes, Google Play Store, ReelShort, Text Classification

1. Pendahuluan

Perkembangan Teknologi di era digital ini membawa banyak perubahan signifikan dalam berbagai aspek kehidupan salah satunya dalam dunia hiburan. Salah satunya adalah munculnya banyak platform hiburan digital yang menawarkan konten hiburan mulai dari film, serial TV dan music. Popularitas layanan hiburan digital ini didorong oleh kehadiran aplikasi *streaming* berbasis mobile yang memungkinkan masyarakat menikmati tayangan visual secara fleksibel. Fenomena ini semakin mencuat di Indonesia setelah pandemi Covid-19 mendorong masyarakat untuk berlangganan aplikasi streaming video [1].

ReelShort, merupakan salah satu *Platform streaming* video generasi baru yang telah berhasil menarik perhatian pengguna dengan konsep videonya yang unik dan inovatif. *Platform* ini menawarkan pengalaman menonton yang lebih eksklusif, personal dan interaktif. *Platform* ini sudah diunduh sebanyak 10 juta kali dengan rating 4,8 dan 413 ribu review atau ulasan dari pengguna di *Google Playstore*.

Analisis sentimen atau *opinion mining* menggambarkan bidang yang luas terkait pemrosesan bahasa alami, linguistik, komputasi, dan penambangan teks dengan tujuan untuk menganalisis pendapat, perasaan, evaluasi, sikap, penilaian, dan emosi seseorang atau penulis tentang suatu topik, produk, layanan, atau yang lainnya [2]. Dalam konteks platform digital seperti *ReelShort*, analisis sentimen digunakan untuk mengukur tingkat kepuasan pengguna, aspek-aspek yang disukai dan tidak disukai, serta untuk mengidentifikasi trend dan bentuk opini pengguna aplikasi ini.

Penelitian sebelumnya yang dilakukan oleh [3] menunjukkan bahwa tingkat *Accuracy* 77%, presisi 26%, *recall* 33%, dan *f1-score* 29%, data testing yang digunakan sebanyak 300 data atau 10% dari 3000 data, dari jumlah data yang digunakan dengan metode random pada saat testing. Selanjutnya pada penelitian yang lalu dikemukakan oleh [4] yang membandingkan dua algoritma untuk klasifikasi teks yaitu *Naïve Bayes* dan *K-Nearest Neighbor* menunjukkan algoritma *Naïve Bayes* menghasilkan *accuracy* sebesar 85%, *precision* 88,5%, *recall* 88,2% dan *f-measure* 88,2%, sementara *K-Nearest accuracy* diangka 84,8 %, *precision* 85,4%, *recall* 84,6% dan *f-measure* 84,1% *precision* 85.4%, *f-measure* 88.2% sedangkan *K-Nearest Neighbor* menghasilkan *accuracy* 84.8%, *precision* 85.4%, *recall* 84.6% dan *f-measure* 84.1% . Pada penelitian yang lain oleh [5] menggunakan 3948 data ulasan menghasilkan nilai akurasi sebesar 91%, *Precision* 92% dan *recall*nya sebesar 100%. Dengan kesimpulan menunjukkan cenderung negatif sebesar 91% untuk aplikasi tersebut.

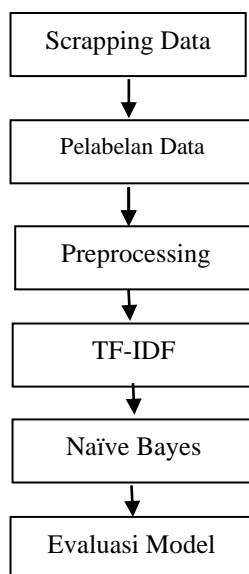
Berdasarkan referensi jurnal penelitian diatas, algoritma *Naïve Bayes* memiliki akurasi yang cukup tinggi dan mudah digunakan untuk analisis sentimen dibandingkan algoritma klasifikasi yang lainnya. *Naive Bayes* juga terbukti mempunyai tingkat kecepatan dan akurasi yang tinggi saat diaplikasikan kedalam basis data dengan jumlah yang besar [6].

Penelitian ini fokus pada analisis sentimen pengguna aplikasi *ReelShort* menggunakan metode *Naive bayes* yang bertujuan untuk menganalisis sentimen pengguna yang datanya bersumber dari ulasan pengguna aplikasi diambil dari *Google Playstore* menerapkan metode *Naive bayes* menggunakan Bahasa pemograman *Python* untuk mendapatka prediksi terhadap ulasan yang sudah ada dan memasukkannya kedalam 2 kategori positif dan negatif.

2. Metode

2.1 Kerangka Penelitian

Penelitian ini melalui beberapa tahapan atau proses yang dilakukan untuk mendapatkan hasil yang terbaik. Proses penelitian diawali dengan melakukan pengumpulan data ulasan pada aplikasi *google playstore* melalui aplikasi *google colab* dan Bahasa pemograman *Python* disimpan kedalam dataset dengan format *csv*. Kemudian dilanjut dengan *preprocessing text*, *TF-IDF*, algoritma *Naïve Bayes*, kemudian tahap evaluasi model menggunakan *Confusion Matrix*, *classification report*, dan visualisasi menggunakan *wordcloud*. Tahapan yang dilakukan bisa dilihat pada Gambar 1.



Gambar 1. Kerangka Kerja Penelitian

2.2 Scrapping Data

Ini adalah tahap pengumpulan data atau scrapping data yang dilakukan menggunakan bahasa pemrograman *Python* pada aplikasi *Google Caloboratory* dan *library google play scraper*. Data ulasan dan data ulasan negative [7]. Label positif dan negatif berdasarkan skor yang diberikan pada ulasan tersebut.

2.3 Pelabelan Data

Ulasan yang telah dikumpulkan akan dikategorikan menjadi dua kelas, yaitu positif dan negatif. Proses pengkategorian ini dilakukan secara manual. Dari hasil pelabelan didapatkan data ulasan positif dan data ulasan negative [7]. Label positif dan negatif berdasarkan skor yang diberikan pada ulasan tersebut.

2.4 Preprocessing Data

Preprocessing adalah tahap pembersihan dan perbaikan data. Pada saat pengumpulan data yang didapat biasanya merupakan data yang tidak terstruktur dan mengandung banyak karakter [8]. Tahap-tahap dari preprocessing ini adalah:

1. *Cleaning*, ini adalah tahap pembersihan data yang bertujuan untuk menghilangkan elemen-elemen, seperti karakter khusus, tanda baca, angka, dan simbol. Proses ini penting untuk memastikan bahwa hanya kata-kata yang terdapat pada ulasan bermakna dan bisa digunakan dalam analisis sentimen.
2. *Case folding*, adalah proses mengubah kata atau huruf besar menjadi huruf kecil semua atau huruf besar semua. Dalam penerapan case folding untuk penelitian biasanya, semua teks diubah menjadi huruf kecil. Misalnya, 'Aplikasi' dan 'aplikasi' akan diperlakukan sebagai dua kata terpisah dalam kalimat apa pun; oleh karena itu, harus membuat semua kata dalam teks menjadi huruf kecil untuk menghindari duplikasi [9].
3. *Stopword removal*, Penghapusan kata-kata kosong (*stopword*) adalah proses menghilangkan kata-kata umum yang sering muncul dalam bahasa, seperti 'yang', 'dan', 'atau'. Kata-kata ini dianggap tidak membawa makna signifikan dalam analisis sentimen. Dengan menghapus *stopword*, model dapat lebih fokus pada kata-kata bermakna yang mampu menunjukkan sentimen positif atau negatif.
4. *Tokenizing*, merupakan proses awal dalam analisis teks yang bertujuan untuk membagi teks menjadi unit-unit terkecil yang disebut token atau kata. Dengan memecah teks menjadi kata-kata individu, kita dapat menganalisis kata per kata dan mengidentifikasi pola atau tren yang terkandung di dalamnya.

5. *Stemming*, proses perubahan kata menjadi bentuk akarnya yang paling sederhana. Misalnya, kata 'membeli' akan diubah menjadi 'beli'. Tujuan dari proses ini adalah untuk mengurangi variasi bentuk kata yang memiliki makna dasar yang sama, sehingga algoritma dapat lebih mudah mengidentifikasi kata-kata yang terkait.

2.5 Term Frequency Inverse Document Frequency (TF-IDF)

TF-IDF (*Term Frequency Inverse Document Frequency*) adalah teknik ekstraksi fitur yang mengubah teks menjadi representasi numerik. Dengan melakukan perhitungan nilai TF-IDF untuk setiap kata dalam dokumen, kita dapat menghasilkan representasi numerik yang sangat berguna dalam berbagai *algoritma machine learning*. Perhitungan bobot tiap *term* dicari keseluruhan pada setiap dokumen bertujuan untuk mengetahui dan mendapat jumlah kata yang sering muncul pada dokumen dan semakin sering kata tersebut muncul maka semakin tinggi pula bobot atau nilai term tersebut [5].

2.6 Naïve Bayes

Naive Bayes merupakan sebuah pengklasifikasian probabilistik sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan. Algoritma menggunakan teorema Bayes dan mengasumsikan semua atribut independen atau tidak saling ketergantungan yang diberikan oleh nilai pada variabel kelas. *Naive Bayes* didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara kondisional saling bebas jika diberikan nilai output. Keuntungan penggunaan *Naive Bayes* adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (*Training Data*) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian [10].

2.7 Evaluasi Model

Tahap evaluasi dilakukan untuk mengukur tingkat ketepatan dan kinerja model yang telah dibangun. Evaluasi ini dilakukan dengan menggunakan tiga metrik utama yaitu *Confusion matrik*, *accuracy* dan *classification report*. Setelah itu dilanjutkan Pada tahap menampilkan banyak nya kata atau komentar yang sering diperbincangkan oleh para pengguna aplikasi ReelShot hingga dihasilkan suatu visualisasi data dalam bentuk *wordcloud*. Besar kecilnya ukuran kata dalam *wordcloud* berpengaruh dalam opini public.[11] *Word cloud* merupakan salah satu teknik untuk memvisualisasikan dokumen teks yang sering digunakan sebagai representasi grafis dari suatu dokumen yang dilakukan dengan memplotting kata – kata yang sering muncul dari suatu dokumen pada ruang dua dimensi [12].

3. Hasil dan Pembahasan

3.1 Hasil Scraping Data

Proses pengumpulan data (*scraping*) dilakukan dengan memanfaatkan *platform Google Colaboratory*. Untuk mengambil data ulasan aplikasi *ReelShot* dari *Google Play Store*, digunakan *library google-play-scraper*. Sebanyak 3000 data ulasan berhasil dikumpulkan. Hasil dari proses *scraping* ini dapat dilihat pada visualisasi yang disajikan pada Gambar 2.

	reviewId	userName	userImage	content	score	thumbsUpCount	reviewCreatedVersion
0	2af1fd98-92e8-4088-b249-d9ea73827fb2	A Google user	lh.googleusercontent.com/EGemol2N...	If you like stuff like CWV shows, Hallmark, and...	5	27	2.1.00
1	befb0221-5d13-4507-8708-628713267d38	A Google user	lh.googleusercontent.com/EGemol2N...	DO NOT WASTE YOUR TIME!!! Firstly, the member...	1	4	2.1.00
2	62951dea-89bf-4566-8cf8-2c00f0f1a2fb	A Google user	lh.googleusercontent.com/EGemol2N...	I don't normally leave reviews but this one wa...	1	39	2.1.00

Gambar 2. Hasil *Scraping* Data

Kemudian data disortir dan diambil hanya 'userName', 'score', 'at', 'content' saja menggunakan *code python* seperti yang terlihat pada Gambar 3. Selanjutnya dipilah lagi hanya diambil *content* dan *score* untuk kemudian dilanjutkan ketahap pelabelan data.

```
new_df = df_busu[['userName', 'score', 'at', 'content']]
sorted_df = new_df.sort_values(by='at', ascending=False)
sorted_df.head(-5)

Show hidden output

t steps:  

my_df = sorted_df[['userName', 'score', 'at', 'content']]
my_df=my_df[['content', 'score']]
my_df.head()
```

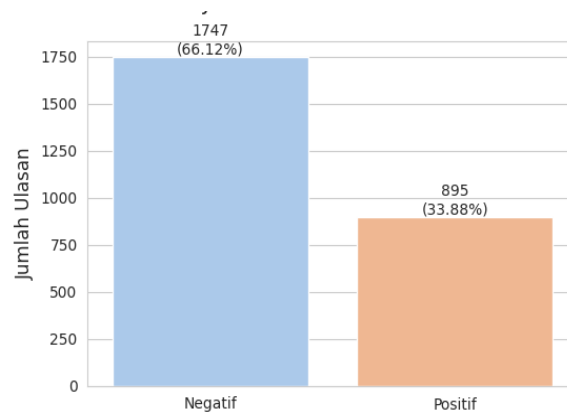
Gambar 3. Code Python untuk Sortir Data Frame

3.2 Pelabelan Data

Setelah proses scrapping data, selanjutnya dilakukan proses labeling data. Labeling ini dilakukan dengan mengacu pada nilai score pada ulasan yang diberikan oleh pengguna. Dimana score < 3 maka labelnya negative, score lebih =4 dan =5 maka labelnya positif. Sementara untuk score = 3 itu labelnya netral atau none. Kemudian dilakukan pembersihan untuk yang nilainya netral atau none sehingga data hanya tinggal yang memiliki dua kategori saja yaitu positif dan Negatif. Setelah proses pembersihan nilai none maka data tersisa sebanyak 2642 ulasan. Hasil dari proses labeling ini dapat dilihat pada Gambar 4.

	content	score	Label
0	It's way too stressful having to watch so many...	1	Negatif
1	Don't like that the reels get shorter on the o...	1	Negatif
2	I wouldn't recommend this it's honestly so stu...	1	Negatif
3	It could be a good app, but you can only watch...	2	Negatif
4	It's so good to watch, even if I need to watch...	5	Positif
...
2990	I would give it 0 stars if I could. Storyline ...	1	Negatif
2991	Cool concept, poor execution. The ads you have...	1	Negatif
2992	The story I listen to was nice but the length ...	2	Negatif
2993	I like the concept and the fact that I don't h...	2	Negatif
2994	I have it 2 stars because they're is 1 min. vi...	2	Negatif

Gambar 4. Hasil Labeling Data



Gambar 5. Perbandingan Jumlah Ulasan Positif dan Negatif

Gambar 5 diatas menunjukkan perbandingan ulasan pengguna aplikasi *ReelShort* setelah proses *labeling* yang menunjukkan hasil yang dominan negatif. Dari total ulasan yang dikumpulkan, 66,12% memberikan penilaian negatif, sementara hanya 33,88% yang bernada positif.

3.3 Hasil Preprocessing Data

Proses awal dalam pengolahan data teks adalah pembersihan data. Tahap ini bertujuan untuk menghilangkan karakter-karakter yang tidak relevan seperti emoji, simbol, dan karakter khusus yang tidak diperlukan dalam analisis sentimen. Setelah proses pembersihan, dilakukan proses *case folding* yaitu mengubah semua huruf menjadi huruf kecil. Tujuannya adalah untuk menyatukan bentuk kata yang sama namun berbeda kapitalisasi. Hasil dari proses *case folding* dapat dilihat pada Tabel 1.

Tabel 1. Hasil *Case Folding*

Hasil <i>Cleaning</i>	Hasil <i>Case Folding</i>
It's way too stressful having to watch so many..	its way too stressful having to watch so many
Don't like that the reels get shorter on the..	dont like that the reels get shorter on the on..
It's so good to watch, even if I need to watch...	its so good to watch even if i need to watch t...

Proses berikutnya adalah menghapus kata-kata yang tidak penting (*stopword removal*). *Stopword* adalah kata-kata umum yang sering muncul dalam bahasa, seperti kata hubung dan kata depan. Contoh *stopword* dalam bahasa Inggris adalah "in", "on", "at", "of", "to", "for". Kata-kata ini dianggap memiliki informasi yang rendah sehingga tidak terlalu berguna dalam analisis teks. Dengan menghapus kata-kata ini, kita bisa lebih fokus pada kata-kata kunci yang mengandung informasi berharga.

Tabel 2. Hasil *Stopword Removal*

Hasil <i>Case Folding</i>	Hasil <i>Stopword Removal</i>
its way too stressful having to watch so many..	way stressful watch many ads finish min story ...
dont like that the reels get shorter on the on..	dont like reels get shorter one watch commerci..
its so good to watch even if i need to watch t...	good watch even need watch ads wait long time ...

Setelah melalui tahap *stopword removal*, tahap selanjutnya adalah *tokenizing* yaitu tahap menguraikan ulasan menjadi token-token atau kata-kata. Proses dilakukan menggunakan *library NLTK*. Hasil dari proses *tokenizing* dapat diamati pada tabel yang disajikan di bawah.

Tabel 3. Hasil Hasil *Tokenizing*

Hasil <i>Stopword Removal</i>	Hasil <i>Tokenizing</i>
way stressful watch many ads finish min story..	['way', 'stressful', 'watch', 'many', 'ads', '...]
dont like reels get shorter one watch commerci..	['dont', 'like', 'reels', 'get', 'shorter', 'o...]
good watch even need watch ads wait long time	['good', 'watch', 'even', 'need', 'watch', 'ad...]
...	

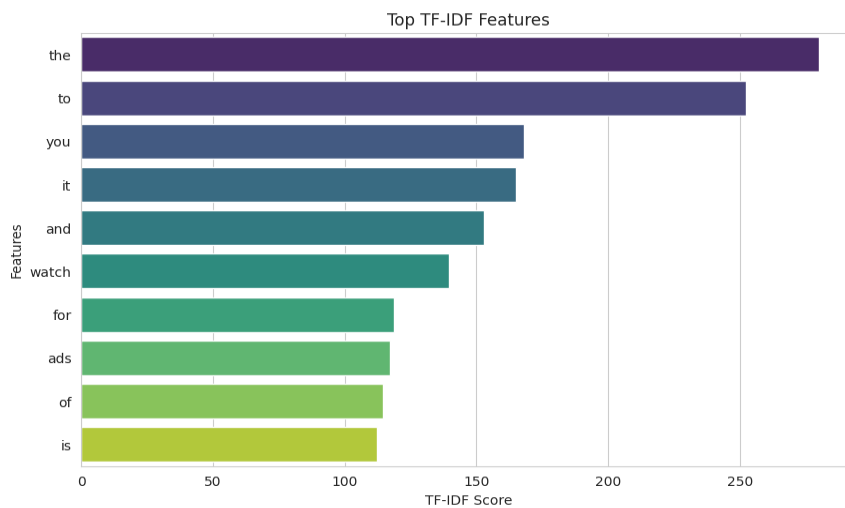
Tahapan berikutnya adalah proses *stemming* yaitu data yang sudah diproses pada tahap sebelumnya akan diubah kata-katanya menjadi kata dasar. Proses ini sangat krusial dalam berbagai tugas NLP (*Natural Language Processing*), termasuk analisis sentimen, pengelompokan dokumen, dan pencarian informasi. *Python*, sebagai bahasa pemrograman yang populer untuk NLP, menyediakan berbagai *library* yang mendukung *stemming*. Salah satu *library* yang paling komprehensif dan sering digunakan adalah NLTK (*Natural Language Toolkit*). Salah satu algoritma *stemming* yang paling terkenal dan tersedia dalam NLTK adalah *Porter Stemmer*. Algoritma ini bekerja dengan serangkaian aturan yang diterapkan secara berurutan pada suatu kata untuk menghilangkan imbuhan. Meskipun sederhana, *Porter Stemmer* cukup efektif dalam mengurangi kata ke bentuk dasarnya.

Tabel 4. Hasil *Stemming*

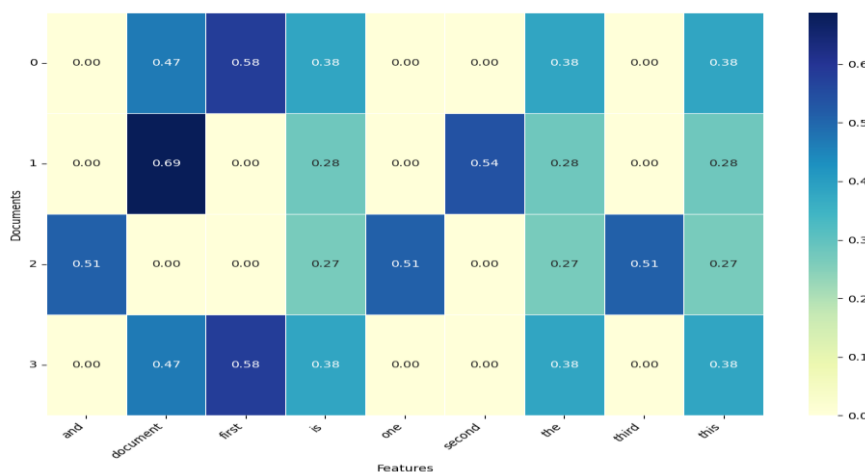
Hasil <i>Tokenizing</i>	Hasil <i>Stemming</i>
['way', 'stressful', 'watch', 'many', 'ads', '...']	way stress watch mani ad finish min stori like...
['dont', 'like', 'reels', 'get', 'shorter', 'o...']	dont like reel get shorter one watch commerci ...
['good', 'watch', 'even', 'need', 'watch', 'ad...']	good watch even need watch ad wait long time

3.4 Hasil Proses TF-IDF

Setelah tahap persiapan data, setiap kata dalam teks diberi skor untuk mengukur tingkat kepentingannya. Proses ini dinamakan *term weighting*. Tujuannya adalah untuk mengidentifikasi kata-kata yang paling relevan dan unik dalam setiap ulasan. Metode yang umum digunakan adalah TF-IDF, yaitu kombinasi antara frekuensi kemunculan kata dalam suatu dokumen dan tingkat kelangkaan kata tersebut di seluruh dokumen. Hasil perhitungan TF-IDF ditampilkan pada Gambar 6. Grafik ini menunjukkan bahwa kata-kata seperti "watch" dan "ads" memiliki skor yang sangat tinggi, mengindikasikan bahwa kata-kata tersebut sangat relevan dengan topik utama ulasan, yaitu pengalaman pengguna dalam menonton video dan interaksi dengan iklan di aplikasi ReelShort. Sebaliknya, kata-kata umum seperti "the", "to", dan "and" memiliki skor yang lebih rendah karena sering muncul dalam bahasa sehari-hari.



Gambar 6. Grafik Skor TF-IDF



Gambar 7. Grafik Heatmap Hasil Proses TF-IDF

Pada Gambar 7 terlihat Grafik heatmap TF-IDF yang memberikan visualisasi mengenai distribusi kata-kata kunci dalam ulasan pengguna aplikasi *ReelShort*. Setiap sel dalam heatmap merepresentasikan nilai TF-IDF dari suatu kata tertentu pada ulasan. Warna yang lebih gelap mengindikasikan nilai TF-IDF yang lebih tinggi, yang berarti kata tersebut sangat relevan dan khas dalam dokumen tersebut. Kata-kata seperti "document", "first", "second", dan "third" mungkin mengindikasikan adanya pola tertentu dalam ulasan, seperti ulasan yang membandingkan fitur-fitur aplikasi atau urutan penggunaan aplikasi. Terdapat variasi yang cukup besar dalam penggunaan kata-kata kunci antar dokumen. Hal ini menunjukkan keragaman topik yang dibahas dalam ulasan pengguna. Warna-warna yang serupa dalam baris atau kolom yang sama menunjukkan adanya korelasi antara kata-kata tersebut. Misalnya, jika kata "first" dan "second" sering muncul bersama-sama dengan nilai TF-IDF yang tinggi, maka dapat diasumsikan bahwa pengguna sering membandingkan fitur pertama dan kedua dari aplikasi.

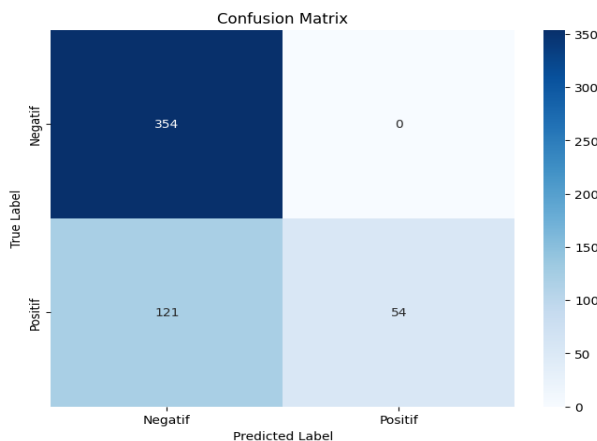
3.5 Hasil Klasifikasi Naïve Bayes

Tahap berikutnya dalam analisis ini adalah melakukan klasifikasi sentimen ulasan menggunakan algoritma *Multinomial Naive Bayes* yang diimplementasikan melalui modul *Scikit-learn*. Pemilihan algoritma ini didasarkan pada kemampuannya dalam menangani data teks dan kinerja yang telah terbukti dalam berbagai penelitian, seperti yang dilaporkan dalam [3], [4], dan [5] di mana algoritma ini mencapai akurasi di atas 77%. Untuk melatih dan menguji model, data dibagi menjadi data latih (80%) dan data uji (20%), menghasilkan masing-masing 2.270 dan 568 data. Hasil evaluasi menunjukkan bahwa model mencapai akurasi 77%, presisi 83%, *recall* 77%, dan *F1-score* 73%. Evaluasi kinerja model secara lengkap dapat dilihat pada Tabel 5.

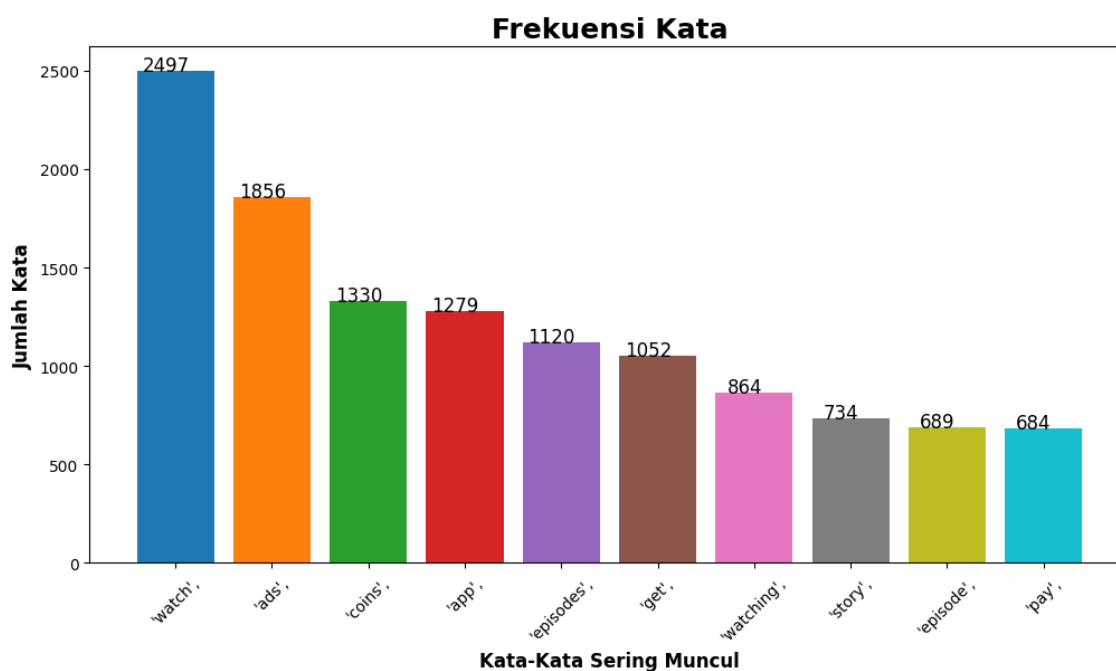
Tabel 5. *Classification Report*

Akurasi Model	: 0.77			
	: 77%			
<i>Classification Report</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Support</i>
Negatif	0.75	1.00	0.85	354
Positif	1.00	0.31	0.47	175
Akurasi				0.80
<i>Macro Avg</i>	0.87	0.65	0.66	529
<i>Weighted Avg</i>	0.83	0.77	0.73	529

Analisis *Confusion matrix* pada Gambar 8 menunjukkan bahwa model klasifikasi sentimen telah mencapai akurasi yang tinggi dalam mengidentifikasi ulasan negatif. Akan tetapi, model masih menghadapi tantangan dalam mengklasifikasikan ulasan positif dengan tingkat akurasi yang setara. Tingginya jumlah *false positive* pada kelas positif mengindikasikan adanya potensi perbaikan pada model, terutama dalam hal kemampuannya untuk mengidentifikasi fitur *linguistik* yang mengindikasikan sentimen positif.



Gambar 8. Confusion Matrik



Gambar 11. Frekuensi Kata yang Sering Muncul

4. Kesimpulan dan Saran

4.1 Kesimpulan

Analisis sentimen menggunakan algoritma *Multinomial Naive Bayes* pada ulasan pengguna ReelShort menghasilkan akurasi keseluruhan sebesar 77%. Model ini menunjukkan kinerja yang baik dalam mengklasifikasikan ulasan negatif, dengan presisi 75% dan recall 100%. Akan tetapi, model mengalami kesulitan dalam mengidentifikasi ulasan positif, sebagaimana ditunjukkan oleh nilai *recall* yang rendah (31%). Ketimpangan kinerja ini mengindikasikan bahwa model cenderung lebih baik dalam mendeteksi sentimen negatif dibandingkan positif. Hal ini dapat disebabkan oleh distribusi data yang tidak seimbang atau kompleksitas ekspresi sentimen positif yang lebih beragam.

Fitur-fitur menarik seperti konten video pendek yang beragam dan antarmuka yang *user-friendly* menjadi sorotan utama dalam ulasan positif. Namun, keluhan utama pengguna terfokus pada jumlah iklan yang berlebihan, kualitas konten yang tidak konsisten, dan masalah teknis yang sering ditemui. Hal ini tercermin dari kemunculan kata-kata kunci seperti "ads", "bug", dan "pay" dalam frekuensi yang tinggi pada ulasan negatif.

Penelitian ini memberikan kontribusi signifikan dalam pemahaman mengenai efektivitas algoritma *Multinomial Naive Bayes* dalam menganalisis sentimen pengguna aplikasi. Keberhasilan model dalam mengklasifikasikan sentimen negatif dengan akurasi yang tinggi mengindikasikan bahwa model dapat menjadi alat yang berharga bagi pengembang aplikasi untuk mengidentifikasi area yang perlu ditingkatkan. Selain itu, penelitian ini juga menegaskan pentingnya tahap pra-pemrosesan data yang cermat untuk menghasilkan model yang lebih akurat.

4.2 Saran

Walaupun algoritma *Multinomial Naive Bayes* telah menunjukkan performa yang baik, kualitas data masukan yang tidak merata dapat menjadi kendala. Untuk mengatasi hal ini, sebaiknya dilakukan pengumpulan data yang lebih *representatif* serta penerapan teknik augmentasi data guna memperkaya variasi data. Dengan demikian, model dapat dilatih pada data ulasan yang lebih seimbang dan kompleks. Sebagai alternatif, algoritma seperti *Support Vector Machine (SVM)* atau metode *Deep Learning* dapat dipertimbangkan untuk meningkatkan akurasi klasifikasi, terutama untuk kelas positif.

Daftar Pustaka

- [1] M. Yusuf Rismanda Gaja, I. Maulana, and O. Komarudin, “Analisis Sentimen Opini Pengguna Aplikasi Vidio Pada Ulasan Playstore Menggunakan Algoritma Naive Bayes,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 4, pp. 2767–2774, 2024, doi: 10.36040/jati.v7i4.7197.
- [2] D. P. Santoso and W. Wibowo, “Analisis Sentimen Ulasan Aplikasi Buzzbreak Menggunakan Metode Naive Bayes Classifier pada Situs Google Play Store,” *J. Sains dan Seni ITS*, vol. 11, no. 2, 2022, doi: 10.12962/j23373520.v11i2.72534.
- [3] A. Nurian, M. S. Ma’arif, I. N. Amalia, and C. Rozikin, “Analisis Sentimen Pengguna Aplikasi Shopee Pada Situs Google Play Menggunakan Naive Bayes Classifier,” *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3631.
- [4] R. T. S. A. Putri, D. E. Ratnawati, and D. W. Brata, “Perbandingan Naive Bayes dan K-Nearest Neighbor untuk Analisis Sentimen Aplikasi Gapura UB Berdasarkan Ulasan Pengguna pada Playstore,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 1, pp. 229–236, 2023, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [5] G. Darmawan, S. Alam, and M. I. Sulisty, “Analisis Sentimen Berdasarkan Ulasan Pengguna Aplikasi Mypertamina Pada Google Playstore Menggunakan Metode Naive Bayes,” *STORAGE – J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, pp. 100–108, 2023.
- [6] A. C. Fauzan and K. Hikmah, “Implementasi Algoritma Naive Bayes Dalam Analisis Polarisasi Opini Masyarakat Terkait Vaksin Covid-19,” *Rabit J. Teknol. dan Sist. Inf. Univrab*, vol. 7, no. 2, pp. 122–128, 2022, doi: 10.36341/rabit.v7i2.2403.
- [7] A. I. Tanggraeni and M. N. N. Sitokdana, “Analisis Sentimen Aplikasi E-Government pada Google Play Menggunakan Algoritma Naive Bayes,” *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 2, pp. 785–795, 2022, doi: 10.35957/jatisi.v9i2.1835.
- [8] M. K. Insan, U. Hayati, and O. Nurdiawan, “Analisis Sentimen Aplikasi Brimo Pada Ulasan Pengguna Di,” *J. Mhs. Tek. Inform.*, vol. 7, no. 1, pp. 478–483, 2023.
- [9] M. Tirta Nugraha, N. Nina Sulistiyowati, and U. Ultach Enri, “Analisis Sentimen Ulasan Aplikasi Satu Sehat Pada Google Play Store Menggunakan Naive Bayes Classifier,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 5, pp. 3593–3601, 2024, doi: 10.36040/jati.v7i5.7753.
- [10] A. Nurian, “Analisis Sentimen Ulasan Pengguna Aplikasi Google Play Menggunakan Naive Bayes,” *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3s1, pp. 829–835, 2023, doi: 10.23960/jitet.v11i3s1.3348.
- [11] K. D. Indarwati and H. Februariyanti, “Analisis Sentimen Terhadap Kualitas Pelayanan Aplikasi Gojek Menggunakan Metode Naive Bayes Classifier,” *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 10, no. 1, 2023, doi: 10.35957/jatisi.v10i1.2643.
- [12] Z. Annisa and B. S. S. Ulama, “Analisis Sentimen Data Ulasan Pengguna Aplikasi ‘PeduliLindungi’ pada Google Play Store Menggunakan Metode Naive Bayes Classifier Model Multinomial,” *J. Sains dan Seni ITS*, vol. 11, no. 6, 2023, doi: 10.12962/j23373520.v11i6.94064.