

Terbit online pada laman web jurnal: <https://jurnal.plb.ac.id/index.php/tematik/index>



TEMATIK

Jurnal Teknologi Informasi Komunikasi (e-Journal)

Vol. 11 No. 2 (2024) 155 - 163

ISSN Media Elektronik: 2443-3640

Kombinasi Metode SVM Dengan Optimasi SMOTE Terhadap Ulasan Pengguna Layanan Streaming

Ni Putu Anik Juniantini^{1*}, Christina Purnama Yanti², Ni Ketut Utami Nilawati³

^{1,2,3}Program Studi Informatika, Fakultas Teknologi dan Informatika, Institut Bisnis dan Teknologi Indonesia

¹anikjunia26@gmail.com, ²christinapy@instiki.ac.id, ³utami.nilawati@instiki.ac.id

Abstract

The development of digital technology has changed media consumption patterns in Indonesia, with streaming services like Disney+ Hotstar becoming increasingly popular. Since its launch in 2020, Disney+ Hotstar has offered exclusive content from Marvel, Pixar, Disney, National Geographic and others, quickly capturing the market in Indonesia. However, this service is not without controversy, particularly regarding certain content deemed to conflict with social values in Indonesia. Additionally, service quality, subscription prices, and content availability are also concerns for users. The difference in ratings on the Play Store (1.7) and the App Store (4.8) indicates a disparity in user satisfaction between the two platforms. This study aims to analyze user sentiment towards Disney+ Hotstar, particularly regarding reviews on the Play Store and App Store. Using a classification model with Support Vector Machine (SVM), optimized with the Synthetic Minority Over-Sampling Technique (SMOTE) to address data imbalance. Based on the analysis of 1,650 datasets, user sentiment tends to be neutral, as measured using the Vader Lexicon. The method testing results show that SMOTE optimization can improve the performance of the SVM model, with an accuracy increase of +0.7 on Play Store reviews from 0.67 to 0.74, and an accuracy increase of +0.11 on App Store reviews from 0.72 to 0.83. In conclusion, the SVM method optimized with SMOTE has proven effective in improving the accuracy of sentiment classification in user reviews of Disney+ Hotstar.

Keywords: Sentiment, Disney+ Hotstar, SVM, SMOTE, user reviews

Abstrak

Perkembangan teknologi digital telah mengubah pola konsumsi media di Indonesia, dengan munculnya layanan *streaming* seperti *Disney+ Hotstar* menjadi semakin populer. Sejak diluncurkan pada tahun 2020, *Disney+ Hotstar* menghadirkan berbagai konten eksklusif dari *Marvel*, *Pixar*, *Disney*, *National Geographic* dan lainnya, dengan cepat meraih pasar di Indonesia. Meskipun demikian, layanan ini tidak lepas dari adanya kontroversi, terutama terkait beberapa konten yang dianggap bertentangan dengan nilai sosial di Indonesia. Selain itu, kualitas layanan, harga berlangganan, dan ketersediaan konten turut menjadi perhatian pengguna. Adanya perbedaan rating pada *Play Store* (1.7) dan *App Store* (4.8) mengindikasikan adanya perbedaan dalam kepuasan pengguna di kedua platform tersebut. Penelitian ini bertujuan untuk melakukan analisis sentimen pengguna terhadap *Disney+ Hotstar*, khususnya terhadap ulasan *Play Store* dan *App Store*. Dengan menggunakan model klasifikasi *Support Vector Machine (SVM)* yang dioptimalkan dengan teknik *Synthetic Minority Over-Sampling Technique (SMOTE)* untuk mengatasi ketidakseimbangan data. Berdasarkan analisis menggunakan 1.650 dataset, sentimen pengguna cenderung memiliki nilai Netral, yang telah diukur menggunakan *Vader Lexicon*. Hasil pengujian metode menunjukkan bahwa optimasi *SMOTE* mampu meningkatkan performa model *SVM*, dengan peningkatan akurasi sebesar +0,7 pada ulasan *Play Store* dari 0,67 menjadi 0,74, serta peningkatan akurasi sebesar +0,11 pada ulasan *App Store* dari 0,72 menjadi 0,83. Kesimpulannya, metode *SVM* yang dioptimalkan dengan *SMOTE* terbukti efektif dalam meningkatkan akurasi klasifikasi sentimen ulasan pengguna terhadap layanan *Disney+ Hotstar*.

Kata kunci: : Sentimen, *Disney+ Hotstar*, *SVM*, *SMOTE*, ulasan pengguna.

1. Pendahuluan

Perkembangan teknologi digital di Indonesia telah mengubah kebiasaan masyarakat dalam mengonsumsi media, beralih dari televisi konvensional ke layanan

streaming seperti *Netflix*, *Viu*, *WeTV*, *Iflix*, dan *Disney+ Hotstar* [1]. Ditengah maraknya layanan *streaming* di Indonesia, *Disney+ Hotstar* hadir pada September 2020 dengan cepat menarik perhatian masyarakat khususnya pecinta serial *Disney*. *Disney+ Hotstar* menawarkan

konten eksklusif dari *Marvel*, *Disney*, *Pixar*, *National Geographic*, dan berbagai film internasional serta lokal [2]. Berdasarkan riset Media Partners Asia, Belum genap 6 bulan dirilis di Indonesia *Disney+ Hotstar* mendominasi pasar layanan *streaming* berbayar dengan 2,5 juta pelanggan, jauh dari *Netflix* yang memiliki 850.000 pelanggan, kemudian *Viu* dengan memiliki 1,5 juta pelanggan, dan *Vidio* yang memiliki 1,1 pelanggan berbayar. Dengan kepopuleran *Disney+ Hotstar*, layanan ini pun tidak luput dari adanya kontroversi, terutama terkait dengan adanya konten yang mengandung unsur yang bertentangan dengan nilai-nilai sosial dan peraturan pemerintah, seperti yang tercantum pada Permendikbud Nomor 14 Tahun 2019 [3]. Selain dengan adanya kontroversi tersebut, kualitas layanan, pilihan konten dan biaya berlangganan juga dapat mempengaruhi tingkat kepopuleran dan minat calon pelanggan. Hal ini terlihat melalui ulasan pengguna serta rating *Disney+ Hotstar*, di mana *Disney+ Hotstar* memperoleh rating 1.7 pada *Play Store* dan 4.8 pada *App Store*. Ulasan dan rating merupakan indikator penting untuk mengukur kepuasan pengguna dan dapat mempengaruhi keputusan calon pelanggan. Oleh karena itu, analisis sentimen terhadap ulasan dapat digunakan dalam memahami bagaimana pandangan pengguna kepada layanan dan dengan adanya umpan balik pengguna dapat dijadikan sebagai pertimbangan evaluasi dan pengembangan layanan *Disney+ Hotstar* kedepannya [4].

Sudah ada penelitian terdahulu terkait analisis sentimen yang dilakukan, diantaranya Implementasi *K-Nearest Neighbor* (*K-NN*) terhadap Aplikasi *FLIP* menghasilkan 77,67% data materi yang diuji masuk ke dalam ulasan positif [5]. Kemudian, Penerapan *Naïve Bayes Classifier* terhadap Aplikasi *Streaming Film Online* menghasilkan sentimen positif sebanyak 885 dan sentimen negatif sebanyak 773 dengan hasil akurasi sebesar 75,25% [6]. Dalam analisis sentimen terdapat banyak model klasifikasi seperti pada penelitian Perbandingan *Naïve Bayes*, *SVM*, dan *k-NN* menyebutkan bahwa metode *SVM* dapat menunjukkan hasil terbaik, dengan rata-rata *accuracy* *SVM* sebesar 96.43%, *Naïve Bayes* 83,54%, dan *k-NN* 59,68% [7]. Maka dari hasil tersebut, penelitian ini akan mengimplementasikan metode *SVM* yang merupakan metode efektif dalam analisis sentimen, khususnya dalam data yang berdimensi tinggi dan dapat menghasilkan akurasi yang baik. Namun, *SVM* juga memiliki kelemahan dalam menangani data yang tidak seimbang.

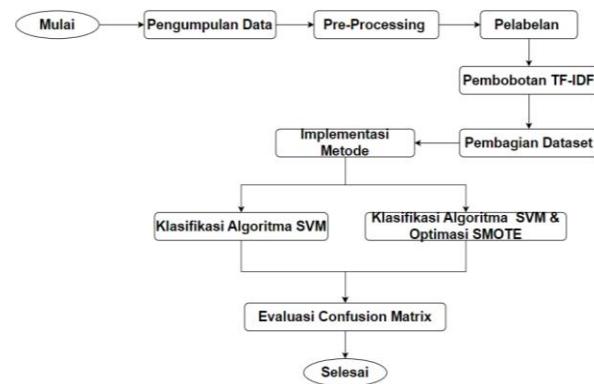
Teknik *Synthetic Minority Over-Sampling Technique* (*SMOTE*) dapat digunakan dalam mengatasi ketidaksetaraan data dengan melakukan *oversampling* pada kelas minoritas, sehingga model dapat mempelajari kelas data dengan lebih baik dan dapat meningkatkan akurasi. Dari hasil penelitian Optimasi *SVM* dan *Decision Tree* Menggunakan *SMOTE* menghasilkan bahwa Optimasi *SMOTE* menghasilkan metode *SVM* lebih unggul dari *Decision Tree* dengan

hasil akurasi setelah *SMOTE*, *SVM* sebesar 99% dan *Decision Tree* sebesar 97% [8]. Dari pemaparan diatas maka dapat dikatakan bahwa metode *SVM* dan Optimasi *SMOTE* merupakan metode yang baik untuk digunakan dalam klasifikasi data.

Berdasarkan latar belakang tersebut, penelitian ini memiliki tujuan untuk memberikan pemahaman yang lebih baik serta mendalam mengenai sentimen atau ulasan pengguna terhadap layanan *Disney+ Hotstar* pada *Play Store* dan *App Store* serta mengevaluasi performa metode *SVM* yang telah dioptimalkan dengan Optimasi *SMOTE*.

2. Metode Penelitian

Agar penelitian ini dapat dilaksanakan, terdapat beberapa tahapan atau alur yang akan dilakukan. Rancangan tahapan penelitian dilihat pada Gambar 1.

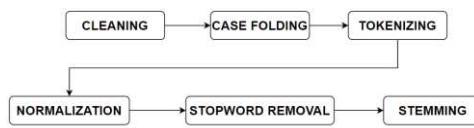


Gambar 1. Metode Penelitian

Penelitian ini dimulai dengan pengumpulan data ulasan pengguna *Disney+ Hotstar* dari *Play Store* dan *App Store* menggunakan teknik *scraping* data. Selanjutnya, dilakukan tahap *Pre-Processing* data untuk menghilangkan *noise*. Setelah data bersih, dilanjutkan dengan tahap pelabelan menggunakan *Vader Lexicon*, untuk mengelompokkan data ulasan ke dalam tiga kelas positif, netral, dan negatif. Setelah itu, data dibagi menjadi dataset pelatihan dan pengujian, serta dilakukan pembobotan menggunakan *TF-IDF*. Kemudian, melakukan implementasi metode *SVM* dan melakukan optimasi metode *SVM* menggunakan *SMOTE*. Langkah terakhir adalah mengevaluasi hasil dengan menggunakan *Confusion Matrix* untuk membandingkan kinerja *SVM* sebelum dan sesudah optimasi dengan *SMOTE*.

Pengumpulan Data: Data yang akan digunakan dalam penelitian ini berasal dari ulasan pengguna aplikasi *Disney+ Hotstar* pada platform *Play Store* dan *App Store*. Pengumpulan data dilakukan menggunakan bahasa pemrograman *python* pada *Google Colab*, tanggal 10 Mei 2024 dan mendapatkan sebanyak 1.650 ulasan yang bersumber dari *Play Store* dan *App Store* dengan masing-masing data yang digunakan berjumlah 825 ulasan.

Pre-processing adalah proses awal untuk mengubah data mentah menjadi data yang sesuai dengan prosedur data *mining* dan merupakan langkah terpenting dalam data *mining*. Pada tahap ini akan melakukan proses pembersihan terhadap data yang masih kotor dengan tujuan untuk menghilangkan *noise* data, menangani informasi yang hilang atau tidak lengkap [9]. *Pre-Processing* mencakup beberapa langkah, dapat dilihat pada Gambar 2.



Gambar 2. *Pre-Processing*

Pelabelan *Vader Lexicon: Valence Aware Dictionary Sentiment Reasoner (Vader)* merupakan sebuah tools analisis sentimen yang dapat digunakan secara khusus dalam pemrosesan bahasa alami untuk mengukur sentimen dalam teks. *Vader* merupakan salah satu alat analisis sentimen berbasis kosakata dan aturan yang dirancang khusus untuk sentimen yang diungkapkan di jejaring sosial. *Vader* akan menghasilkan skor yang dinyatakan secara komposit, yaitu kombinasi masing-masing skor positif, negatif, dan netral. Nilai yang lebih besar dari nol (>0) cenderung menunjukkan sentimen positif, nilai yang kurang dari nol (<0) cenderung menunjukkan sentimen negatif, dan nilai yang mendekati nol ($=0$) menunjukkan sentimen netral [10].

Pembobotan TF-IDF: *TF-IDF (Term Frequency-Inverse Document Frequency)* merupakan salah satu metode pembobotan kata yang sering digunakan dalam pemrosesan teks. Prinsip dasar dari *TF-IDF* adalah memberikan bobot jika sebuah kata sering muncul di satu dokumen tertentu, bobotnya akan meningkat, namun jika kata tersebut juga sering muncul di banyak dokumen lain, bobotnya justru akan berkurang karena dianggap kurang signifikan dalam membedakan dokumen tersebut [11]. Tujuan utama dari metode pembobotan ini adalah mengubah data teks yang tidak terstruktur menjadi data numerik yang dapat dipahami oleh algoritma pembelajaran mesin. Algoritma pembelajaran mesin bekerja secara optimal ketika input data dalam bentuk angka, sehingga fitur seperti *TF-IDF* berperan penting dalam proses pra-pemrosesan data teks sebelum digunakan dalam model pembelajaran mesin [12].

Pada tahap pembagian dataset, akan menggunakan rasio 80:20 yaitu dataset dibagi menjadi 80% data pelatihan dan 20% data uji. Data pelatihan digunakan dalam memberikan pelatihan terhadap model dalam tahap klasifikasi, sedangkan data uji digunakan dalam melakukan evaluasi model dalam mengklasifikasikan data baru.

Support Vector Machine (SVM) merupakan model yang dapat digunakan dalam mengklasifikasikan data dalam bentuk teks untuk menghasilkan tingkat akurasi yang lebih tinggi. *SVM* merupakan metode data *mining* yang

masuk dalam kelas *supervised learning*. *SVM* bekerja dengan menemukan *hyperplane* terbaik di antara beberapa *hyperplane* yang tersedia. *Hyperplane* terbaik ialah *hyperplane* yang terletak di antara kumpulan data milik dua kelas berbeda. Cara mencari *hyperplane* terbaik adalah dengan memaksimalkan *margin*. *Margin* merupakan jarak tegak lurus yang ada diantara *hyperplane* dengan obyek terdekat atau dapat disebut juga dengan *support vector* [13].

Pada klasifikasi menggunakan *SVM*, sering ditemukan situasi dimana *kernel linear* tidak bekerja secara maksimal sehingga hal ini menyebabkan hasil pengelompokan data yang buruk. Hal ini dapat diatasi dengan *kernel non-linear* menggunakan *kernel trick* [14]. Persamaan masing-masing *Kernel SVM*, dapat dilihat pada Tabel 1.

Tabel 1. Persamaan *Kernel SVM*

| Kernel | Persamaan | |
|------------|--|-----|
| Linear | $K(x_i, x) = x_i^T x$ | (1) |
| Polynomial | $K(x_i, x) = (y_i x_i^T x + r)^p, y > 0$ | (2) |
| RBF | $K(x_i, x) = \exp(-\gamma x_i - x ^2), y > 0$ | (3) |
| Sigmoid | $K(x_i, x) = \tanh(y_i x_i^T x + r)$ | (4) |

Synthetic Minority Oversampling Technique (SMOTE) merupakan metode *oversampling* untuk mengatasi ketidaksetaraan kelas. Teknik ini bekerja dengan membuat data sintetis dari kelas minoritas dengan mengambil beberapa contoh dari kelas minoritas dan menggabungkan karakteristiknya untuk menciptakan sampel baru, sehingga jumlah data dari kelas minoritas dan kelas mayoritas seimbang [15]. Kelebihan metode *SMOTE* yaitu tidak ada data yang hilang dalam suatu kelas karena tidak adanya pengurangan data. *SMOTE* juga memberikan kesempatan pada kelas minoritas untuk dipelajari lebih baik, sehingga bisa menghasilkan peningkatan akurasi pada kelas minoritas [16].

Tahap akhir dari penelitian ini adalah melakukan evaluasi menggunakan *Confusion Matrix*. *Confusion Matrix* merupakan alat yang sering digunakan untuk mengukur kinerja model prediksi, terutama dalam proses klasifikasi. Alat ini membantu mengidentifikasi tingkat akurasi model dengan menganalisis seberapa baik model tersebut mampu membedakan antara berbagai kelas yang ada dalam data. *Confusion Matrix* memberikan gambaran detail mengenai hasil klasifikasi, termasuk jumlah prediksi yang benar dan salah untuk setiap kelas [17].

3. Hasil dan Pembahasan

Penelitian ini memiliki tujuan untuk melakukan analisis sentimen dari ulasan pengguna aplikasi *Disney+ Hotstar* dengan mengklasifikasikan opini pengguna pada ulasan *Play Store* dan *App Store*. Dataset akan diuji menggunakan metode *Support Vector Machine (SVM)* dan dioptimalkan dengan algoritma *Synthetic Minority Oversampling Technique (SMOTE)*. Hasil dari analisis tersebut dapat digunakan sebagai dasar untuk

memberikan rekomendasi yang tepat, yang dapat dijadikan panduan dalam perbaikan dan peningkatan kualitas aplikasi di masa mendatang.

Pengumpulan data dalam penelitian ini dilakukan dengan menggunakan bahasa pemrograman *python* pada *Google Colab*, tanggal 10 Mei 2024 dan mendapatkan ulasan dari 4 September 2020 sampai 8 Mei 2024 sebanyak 1.650 ulasan yang bersumber dari *Play Store* dan *App Store* dengan masing-masing data yang digunakan berjumlah 825 ulasan. Berikut merupakan data mentah yang akan digunakan, dapat dilihat pada Gambar 3 dan Gambar 4.

| score | content | tanggal |
|-------|--|------------|
| 1 | Menurut saya kenaikan harga ini tidak wajar untuk market indonesia. Mungkin alas | 10/12/2023 |
| 2 | Dari UI nya sih saya suka, tapi masih ada bug dimana subtittlenya ga ada sama sej | 16/06/2023 |
| 1 | berbaray | 16/02/2024 |
| 1 | Gabut | 28/02/2024 |
| 2 | Ini kenapa pas mau nonton layar hitam cuman ada suaranya doang | 12/04/2024 |
| 1 | saat nonton jaringan tidak stabil dan terjeda jeda. Semoga ada perbaikan setelah i | 07/12/2023 |
| 5 | mantap dengan berlangganan pakai promo PO tsel bisa dpt harga murah dan film' | 13/09/2020 |
| 2 | Pengen nonton film Disney dalam Bahasa Spanyol, tapi cuma ada bahasa Indonesia | 26/01/2024 |
| 5 | Aplikasi nonton film termantabek dah pokoknya. Ga nyesel berlangganan setahun. | 08/02/2021 |
| 1 | Jelek banget sekali aplikasinya. Masa ia ada suara gk ada gambar. Emang mau non | 12/12/2023 |
| 1 | Tolong diperbaiki punya saya udah pembayaran tapi masih muter aplikasi ini mem | 07/11/2023 |
| 1 | tidak bisa masuk | 20/03/2024 |
| 5 | Drakornya bagus bagus dan downloadnya sangat mudah | 28/01/2024 |
| 2 | Ini kenapa setelah keluar akun, mau masuk lagi malah susah ya dan ga bisa login. ; | 13/04/2024 |
| 1 | Sangat mengecewakan. saat mau mengubah resolusi grafis nya ke yang paling ting | 11/10/2021 |

Gambar 3. Data Mentah *Play Store*

| rating | review_description | tanggal |
|--------|--|------------|
| 1 | Tidak support Airplay dan aplikasi tidak ada di Apple TV, hanya dari H | 04/09/2020 |
| 5 | Terima kasih sudah mewujudkan disney+ di Indonesia | 04/09/2020 |
| 4 | Kode verifikasi yg dimasukkan salah terus, pdhl udh bener:ð€ | 04/09/2020 |
| 4 | saya pre order tapi paket subscription-nya belum jalan | 04/09/2020 |
| 1 | Register pake nomer indosat gk bisa2 ya.. codenya gk masuk2.. | 04/09/2020 |
| 5 | Bakal jadi saingan netflix | 04/09/2020 |
| 2 | Tidak bisa casting melalui chromecast. sms OTP tidak diterima | 04/09/2020 |
| 1 | Kode verifikasi selalu error saat dinput. Mohon diperbaiki! | 04/09/2020 |
| 2 | Katanya aplikasi buat keluarga, anggota keluarga kan preferensi nont | 04/09/2020 |
| 5 | Per 4 September masih belum bisa di streaming, kemungkinan besar l | 04/09/2020 |
| 2 | Untuk menonton di HP sangat bagus. Tidak kalah dari Netflix atau You | 04/09/2020 |
| 3 | 3 4 digit code salah terus | 04/09/2020 |
| 3 | Tambahkan dark mode bro silau banget background warna putih | 04/09/2020 |
| 1 | tolong perbaiki!!!! | 04/09/2020 |
| 5 | Akhirnya ada pilihan Streaming Video baru kebetulan Netflix besok langganan-nya habi | |

Gambar 4. Data Mentah *App Store*

Fase awal yang dilakukan untuk mengolah data adalah *Pre-Processing*. *Pre-Processing* berfungsi untuk mentransformasi data mentah yang tidak terstruktur dan memiliki banyak *noise* menjadi data yang terstruktur sehingga dapat menghasilkan informasi dan siap digunakan dalam pemrosesan data selanjutnya. Terdapat 6 tahap dalam *Pre-Processing* yang akan dilakukan.

Cleaning: Tahap ini merupakan proses untuk membersihkan data ulasan pada *Play Store* dan *App Store* dari *noise* seperti emoji, tanda baca '!"#\$%&\'()*+,-./;:<=>?@[\]^_`{}~', angka dan lain sebagainya. Hasil *Cleaning* dapat dilihat pada Tabel 2.

Tabel 2. Hasil *Cleaning*

| Data Mentah | Cleaning |
|--|---|
| Baru saja mncoba berlangganan, saldo sudah terpotong tapi tidak aktif, sangat mengecewakan, mncoba email tapi tidak di balas | Baru saja mncoba berlangganan saldo sudah terpotong tapi tidak aktif sangat mengecewakan mncoba email tapi tidak di balas |
| :(| |

Case Folding: Tahap ini merupakan proses untuk mengubah huruf kapital menjadi huruf kecil, sehingga data yang dihasilkan lebih mudah untuk diklasifikasikan. Hasil *Case Folding* dapat dilihat pada Tabel 3.

Tabel 3. Hasil *Case Folding*

| Cleaning | Case Folding |
|---|---|
| Baru saja mncoba berlangganan saldo sudah terpotong tapi tidak aktif sangat mengecewakan mncoba email tapi tidak di balas | baru saja mncoba berlangganan saldo sudah terpotong tapi tidak aktif sangat mengecewakan mncoba email tapi tidak di balas |
| | |

Tokenizing: Tahap ini merupakan proses yang digunakan untuk memisahkan ulasan pengguna yang mulanya berupa kalimat menjadi satuan kata, yang dipisahkan dengan menggunakan tanda koma (,) dan petik (''). Hasil *Tokenizing* dapat dilihat pada Tabel 4.

Tabel 4. Hasil *Tokenizing*

| Case Folding | Tokenizing |
|---|--|
| baru saja mncoba berlangganan saldo sudah terpotong tapi tidak aktif sangat mengecewakan mncoba email tapi tidak di balas | ['baru', 'saja', 'mncoba', 'berlangganan', 'saldo', 'sudah', 'terpotong', 'tapi', 'tidak', 'aktif', 'sangat', 'mengecewakan', 'mncoba', 'email', 'tapi', 'tidak', 'di', 'balas'] |
| | |

Normalization: Tahap ini merupakan proses untuk mengubah kata yang tidak baku menjadi kata baku menurut Kamus Besar Bahasa Indonesia (KBBI). Hasil *Normalization* dapat dilihat pada Tabel 5.

Tabel 5. Hasil *Normalization*

| Tokenizing | Normalization |
|--|--|
| ['baru', 'saja', 'mncoba', 'berlangganan', 'saldo', 'sudah', 'terpotong', 'tapi', 'tidak', 'aktif', 'sangat', 'mengecewakan', 'mncoba', 'email', 'tapi', 'tidak', 'di', 'balas'] | ['baru', 'saja', 'mencoba', 'berlangganan', 'saldo', 'sudah', 'terpotong', 'tapi', 'tidak', 'aktif', 'sangat', 'mengecewakan', 'mencoba', 'email', 'tapi', 'tidak', 'di', 'balas'] |
| | |

Stopword Removal : Tahap ini merupakan proses menghilangkan atau menghapus kata yang tidak mempunyai arti penting dan kata berulang. Tahapan *Stopword Removal* bertujuan agar dapat lebih fokus pada data yang memiliki nilai. Hasil *Stopword Removal* dilihat pada Tabel 6.

Tabel 6. Hasil *Stopword Removal*

| Normalization | Stopword Removal |
|--|---|
| ['baru', 'saja', 'mencoba', 'berlangganan', 'saldo', 'sudah', 'terpotong', 'tapi', 'tidak', 'aktif', 'sangat', 'mengecewakan', 'mencoba', 'email', 'tapi', 'tidak', 'di', 'balas'] | ['mencoba', 'berlangganan', 'saldo', 'terpotong', 'aktif', 'sangat', 'mengecewakan', 'mencoba', 'email', 'balas'] |
| | |

Stemming: Tahap ini berfungsi untuk menghilangkan kata yang memiliki imbuhan baik yang terdapat pada awalan kata maupun akhiran kata dan mengubahnya ke

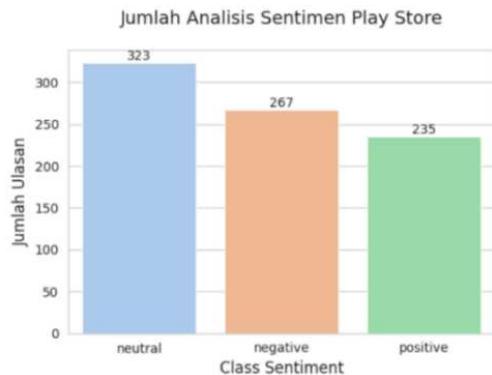
dalam bentuk dasar. Hasil *Stemming* dapat dilihat pada Tabel 7.

Tabel 7. Hasil *Stemming*

| Stopword Removal | Stemming |
|--|---|
| ['mencoba', 'berlangganan', 'saldo', 'terpotong', 'aktif', 'mengecewakan', 'email', 'balas'] | ['coba', 'langgan', 'saldo', 'potong', 'aktif', 'kecewa', 'email', 'balas'] |

Pelabelan *Vader Lexicon*: Pada tahapan pelabelan kata menggunakan *Vader Lexicon*, data akan diolah melalui bahasa pemrograman *Python* pada *Google Colab* dengan kamus *Vader* yang digunakan sebagai acuan untuk memberikan nilai berupa sentimen pada setiap data. Apabila sebuah data mendapatkan nilai sentimen atau $S > 0$ maka data itu akan dilabeli dengan sentimen positif. Kemudian, jika sebuah data mendapatkan nilai $S < 0$ maka akan dilabeli dengan sentimen negatif. Dan apabila, bila $S = 0$ dalam sebuah data maka akan dilabeli dengan sentimen netral.

Berdasarkan pelabelan kata menggunakan *Vader Lexicon* terhadap ulasan pengguna aplikasi *Disney+ Hotstar* pada *Play Store* menunjukkan terdapat sejumlah 323 data dikategorikan sebagai "Netral", sejumlah 267 data dikategorikan sebagai "Negatif" dan sejumlah 235 data dikategorikan sebagai "Positif". Hasil pelabelan kata pada Ulasan *Play Store* dapat dilihat pada Gambar 5.

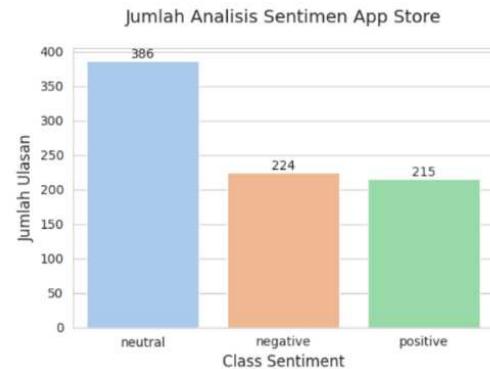


Gambar 5. Visualisasi Sentimen ulasan *Play Store*

Kemudian, Berdasarkan pelabelan kata menggunakan *Vader Lexicon* terhadap ulasan pengguna aplikasi *Disney+ Hotstar* pada *App Store* menunjukkan terdapat sejumlah 386 data dikategorikan sebagai "Netral", sejumlah 224 data dikategorikan sebagai "Negatif" dan sejumlah 215 data dikategorikan sebagai "Positif". Hasil pelabelan kata pada Ulasan *App Store* dapat dilihat pada Gambar 6.

Setelah melakukan tahap *Pre-Processing* dan Pelabelan Kata, data selanjutnya akan memasuki tahap pembobotan. Pada penelitian ini, melakukan pembobotan kata menggunakan *TF-IDF*. *TF-IDF* adalah sebuah proses yang dapat menghitung jumlah kata lalu akan diakumulasikan pada dokumen seberapa banyak kata tersebut muncul. Tahap pembobotan ini digunakan untuk mengubah data ulasan menjadi angka atau berupa numerik agar lebih mudah diolah oleh

model klasifikasi. Dengan mengubah data teks menjadi numerik, model dapat memahami pola dan informasi dari teks sehingga dapat membuat prediksi atau analisis dengan lebih baik.



Gambar 6. Visualisasi Sentimen ulasan App Store

Penelitian ini menggunakan 1.650 dataset dengan pembagian data latih dan data uji menggunakan rasio 80:20 atau menjadi 80% data pelatihan dan 20% data uji. Pembagian dataset dapat dilihat pada Tabel 8.

Tabel 8. Pembagian Dataset

| Data Latih (80%) | Data Uji (20%) |
|------------------|----------------|
| 659 Dataset | 165 Dataset |

Pada tahap implementasi model akan menggunakan *kernel linear* berdasarkan klasifikasi *SVM* untuk memprediksi data yang telah diketahui kelasnya. Dalam *SVM*, *kernel linear* baik digunakan ketika data dapat dipisahkan secara *linear* atau secara garis lurus seperti klasifikasi teks. *Kernel linear* dapat memberikan kinerja yang kompetitif atau bahkan lebih baik dibandingkan *kernel non-linear* yang lebih kompleks. Hal ini dikarenakan data teks yang telah diubah menjadi vektor fitur sudah berada dalam ruang yang dapat dipisahkan secara *linear*.

Dengan menggunakan *kernel linear* akan membuat *SVM* mencari *hyperplane* dan memisahkan data berdasarkan kelasnya. Kemudian setelah menentukan *kernel*, langkah selanjutnya adalah melatih model dengan data latih yang telah dibagi pada tahap sebelumnya. Pada paramater model, nilai c akan digunakan untuk menentukan nilai paramater, *intercept* digunakan untuk menentukan posisi relatif dari batas *hyperplane* dalam ruang model, dan *support vectors* digunakan untuk mencetak titik-titik dari dataset pelatihan yang kemudian akan membangun batas atau *hyperplane* dalam ruang pada model *SVM*.

Tahap klasifikasi *SVM* ulasan *Play Store* ini merupakan implementasi metode *SVM* terhadap ulasan pengguna *Disney+ Hotstar* pada *Play Store*. Hasil inisiasi model menunjukkan nilai parameter yang digunakan atau $c = 1$, kemudian hasil *intercept* atau posisi *hyperplane* dari 3 kelas data terletak pada nilai $[-0.46373915, 0.19125302, 0.65293671]$, dan *support vectors* merupakan titik-titik data dari dataset pelatihan dalam

ruang pada model. Hasil inisiasi *SVM* ulasan *Play Store* dapat dilihat pada Gambar 7.

```
Parameter Model SVM:  
Kernel: linear  
C: 1.0  
Intercept: [-0.46373915 0.19125302 0.65293671]  
Support Vectors: (0, 129) 0.35319960811099077  
(0, 146) 0.48747239326480635  
(0, 334) 0.5154902954891853  
(0, 429) 0.36269128311193344  
(0, 492) 0.3862117038198292  
(0, 815) 0.1586881479453731  
(0, 1570) 0.2001037874284674  
(0, 1707) 0.16112900681457382  
(1, 164) 0.3070743599887914  
(1, 359) 0.14568449660559316
```

Gambar 7. Hasil Inisiasi *SVM* ulasan *Play Store*

Tahap selanjutnya adalah klasifikasi *SVM* Ulasan App Store yang merupakan implementasi metode *SVM* terhadap ulasan pengguna *Disney+ Hotstar* pada *App Store*. Hasil inisiasi model menunjukkan nilai parameter yang digunakan atau $c = 1$, kemudian hasil *intercept* atau posisi *hyperplane* dari 3 kelas data terletak pada nilai $[-0.60856586, 0.08108848, 0.61349401]$, dan *support vectors* merupakan titik-titik data dari dataset pelatihan dalam ruang pada model. Hasil inisiasi *SVM* ulasan *App Store* dapat dilihat pada Gambar 8.

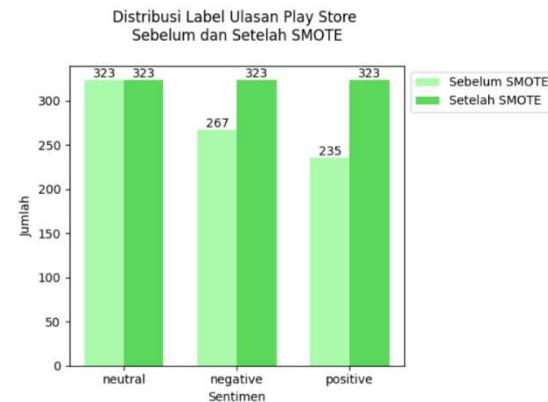
```
Parameter Model SVM:  
Kernel: linear  
C: 1.0  
Intercept: [-0.60856586 0.08108848 0.61349401]  
Support Vectors: (0, 72) 0.36162171643371205  
(0, 124) 0.17870525002503224  
(0, 158) 0.3746652310025807  
(0, 179) 0.2665001747223109  
(0, 220) 0.2274446492386177  
(0, 324) 0.3746652310025807  
(0, 362) 0.2985275210941251  
(0, 791) 0.3746652310025807  
(0, 1087) 0.3041126776072938  
(0, 1455) 0.18225784012707172
```

Gambar 8. Hasil Inisiasi *SVM* ulasan *App Store*

Klasifikasi SVM dan SMOTE: Setelah melakukan klasifikasi dengan metode *SVM* tahap selanjutnya adalah melakukan Optimasi menggunakan *SMOTE*. *SMOTE* merupakan teknik untuk mengatasi ketidakseimbangan kelas. Ketidakseimbangan kelas terjadi pada saat suatu kelas dalam dataset memiliki lebih banyak sampel dibandingkan dengan kelas lainnya, yang bisa menyebabkan metode pembelajaran mesin cenderung mengabaikan kelas minoritas. *SMOTE* bekerja dengan membuat sampel atau data sintetis dari kelas minoritas, sehingga dapat menyeimbangkan kelas. Optimasi *SMOTE* dapat memperbaiki kekurangan yang terdapat pada metode *SVM* dengan melakukan teknik *oversampling* dan menyeimbangkan data sehingga mendapatkan akurasi yang lebih baik.

Klasifikasi SVM dan SMOTE Ulasan Play Store: Berikut ini adalah hasil dari implementasi Optimasi *SMOTE* terhadap ulasan pengguna aplikasi *Disney+ Hotstar* pada *Play Store*. Dari hasil optimasi ini, terlihat

adanya peningkatan jumlah data pada setiap kelas sentimen. Sebelum penerapan *SMOTE*, terdapat ketidakseimbangan jumlah ulasan untuk masing-masing kelas, yaitu 323 untuk "neutral," 267 untuk "negative," dan 235 untuk "positive." Setelah diterapkan *SMOTE*, seluruh kelas menjadi seimbang dengan jumlah data yang sama, yaitu 323 ulasan untuk masing-masing kelas. Hasil implementasi Optimasi *SMOTE* pada ulasan *Play Store* dapat dilihat pada Gambar 9.



Gambar 9. Distribusi Data *Play Store* Sebelum dan Setelah *SMOTE*

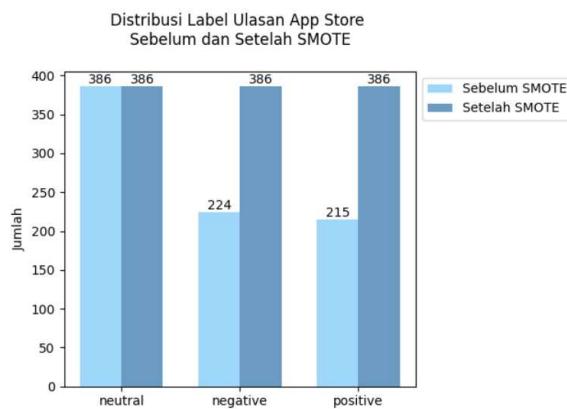
Setelah melakukan optimasi menggunakan *SMOTE*, tahapan selanjutnya adalah mengimplementasikan kembali model *SVM* terhadap data yang telah diseimbangkan. Dari hasil inisiasi *SVM* setelah *SMOTE* menunjukkan nilai parameter atau $c = 1$, kemudian *intercept* atau posisi *hyperplane* dari 3 kelas data terletak pada nilai $[-0.49120348, 0.20134219, 0.68378538]$, dan hasil dari *support vectors* merupakan titik-titik data dari dataset pelatihan dalam ruang model. Hasil inisiasi *SVM* setelah *SMOTE* dapat dilihat pada Gambar 10.

```
Parameter Model SVM dan SMOTE:  
Kernel: linear  
C: 1.0  
Intercept: [-0.49120348 0.20134219 0.68378538]  
Support Vectors: (0, 154) 0.2064649968570349  
(0, 302) 0.24456193599515896  
(0, 339) 0.3534312997260467  
(0, 598) 0.4584968180360541  
(0, 1014) 0.24580541886703897  
(0, 1030) 0.43202770020871334  
(0, 1243) 0.19809244136222784  
(0, 1442) 0.25671085601086424  
(0, 1450) 0.4584968180360541  
(1, 51) 0.2282410659112449
```

Gambar 10. Hasil Inisiasi *SVM* dan *SMOTE* ulasan *Play Store*

Klasifikasi SVM dan SMOTE Ulasan App Store: Tahap berikut ini adalah hasil dari implementasi Optimasi *SMOTE* terhadap ulasan pengguna aplikasi *Disney+ Hotstar* pada *App Store*. Dari hasil optimasi ini, terlihat adanya peningkatan jumlah data pada setiap kelas sentimen. Sebelum penerapan *SMOTE*, terdapat ketidakseimbangan jumlah ulasan untuk masing-masing kelas, yaitu 386 untuk "neutral," 224 untuk "negative," dan 215 untuk "positive." Setelah diterapkan *SMOTE*, seluruh kelas menjadi seimbang

dengan jumlah data yang sama, yaitu 386 ulasan untuk masing-masing kelas. Hasil implementasi Optimasi *SMOTE* pada ulasan *App Store* dapat dilihat pada Gambar 11.



Gambar 11. Distribusi Data *App Store* Sebelum dan Setelah *SMOTE*

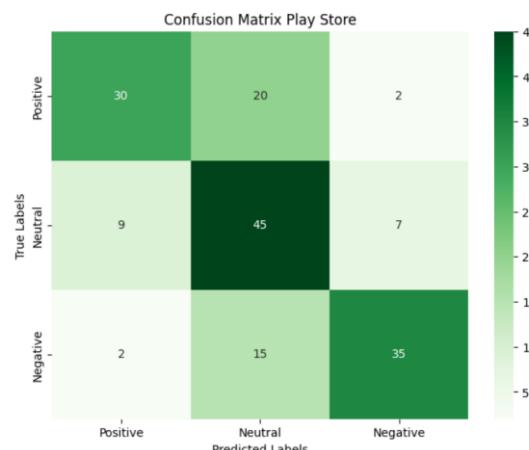
Kemudian, pada ulasan *App Store* setelah melakukan optimasi *SMOTE*, dan mengimplementasikan kembali model *SVM* terhadap data yang telah diseimbangkan. Dari hasil inisiasi *SVM* setelah *SMOTE* menunjukkan nilai parameter atau $c = 1$, kemudian *intercept* atau posisi *hyperplane* dari 3 kelas data terletak pada nilai $[-0.48978136, 0.20655966, 0.64974361]$, dan *support vectors* merupakan titik-titik data dari dataset pelatihan dalam ruang model. Hasil inisiasi *SVM* setelah *SMOTE* dapat dilihat pada Gambar 12.

```
Parameter Model SVM dan SMOTE:
Kernel: linear
C: 1.0
Intercept: [-0.48978136 0.20655966 0.64974361]
Support Vectors: (0, 154) 0.2064649968570349
(0, 302) 0.24456193599515896
(0, 339) 0.3534312997260467
(0, 598) 0.4584968180360541
(0, 1014) 0.24580541886703897
(0, 1030) 0.43202770020871334
(0, 1243) 0.19809244136222784
(0, 1442) 0.25671085601086424
(0, 1450) 0.4584968180360541
(1, 83) 0.11755106087846824
```

Gambar 12. Hasil Inisiasi *SVM* dan *SMOTE* ulasan *App Store*

Setelah melakukan klasifikasi dengan model *SVM* dan optimasi dengan *SMOTE*, tahap berikutnya merupakan evaluasi menggunakan *Confusion Matrix*. *Confusion Matrix* digunakan dalam melakukan evaluasi kinerja model dengan melihat distribusi prediksi terhadap kelas yang sebenarnya. *Confusion Matrix* akan menghitung beberapa indikator performa seperti akurasi, presisi, recall dan *F1-score*, yang memberikan gambaran menyeluruh mengenai seberapa baik model dalam mengelompokkan data ke dalam masing-masing kelas sentimen.

Evaluasi *SVM*: Tahap ini merupakan hasil Evaluasi menggunakan *Confusion Matrix* terhadap model *SVM* dengan data ulasan pengguna *Disney+ Hotstar* pada *Play Store*. Hasil *Confusion Matrix* dapat dilihat pada Gambar 13.



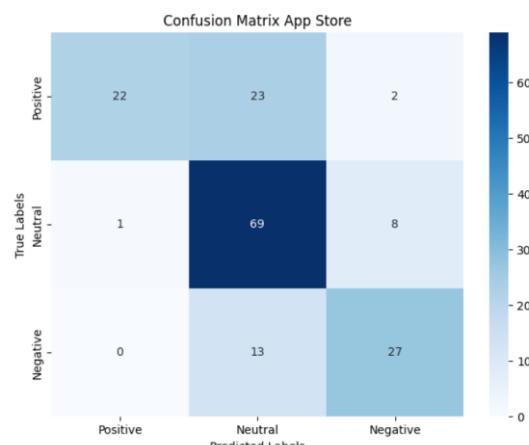
Gambar 13. *Confusion Matrix SVM* pada Ulasan *Play Store*

Kemudian, pada *Classification Report* dari model klasifikasi *SVM* menghasilkan *accuracy* keseluruhan sebesar 0.67 atau sebesar 67%. Kelas *Negative* memiliki *presisi* 0.73, *recall* 0.58, dan *f1-score* 0.65, Kelas *Neutral* memiliki *presisi* 0.56, *recall* 0.74, dan *f1-score* 0.64, dan Kelas *Positive* memiliki *presisi* 0.80, *recall* 0.67, dan *f1-score* 0.73. *Classification Report* pada ulasan *Play Store* dapat dilihat pada Gambar 14.

| | Classification Report: | | | |
|--------------|------------------------|--------|----------|---------|
| | precision | recall | f1-score | support |
| negative | 0.73 | 0.58 | 0.65 | 52 |
| neutral | 0.56 | 0.74 | 0.64 | 61 |
| positive | 0.80 | 0.67 | 0.73 | 52 |
| accuracy | | | 0.67 | 165 |
| macro avg | 0.70 | 0.66 | 0.67 | 165 |
| weighted avg | 0.69 | 0.67 | 0.67 | 165 |

Gambar 14. *Classification Report SVM* pada Ulasan *Play Store*

Pada tahap selanjutnya merupakan hasil Evaluasi menggunakan *Confusion Matrix* terhadap model *SVM* dengan data ulasan pengguna *Disney+ Hotstar* pada *App Store*. Hasil *Confusion Matrix* dapat dilihat pada Gambar 15.



Gambar 15. *Confusion Matrix SVM* pada Ulasan *App Store*

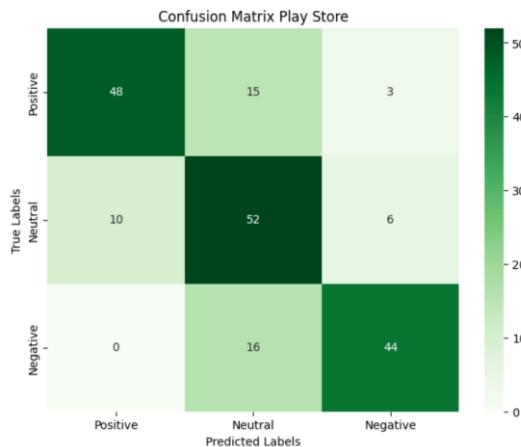
Selanjutnya, pada *Classification Report* dari model klasifikasi *SVM* ulasan *App Store* menghasilkan

accuracy keseluruhan sebesar 0.72 atau sebesar 72%. Kelas *Negative* memiliki *presisi* 0.96, *recall* 0.47, dan *f1-score* 0.63, Kelas *Neutral* memiliki *presisi* 0.66, *recall* 0.88, dan *f1-score* 0.75, dan Kelas *Positive* memiliki *presisi* 0.73, *recall* 0.68, dan *f1-score* 0.70. *Classification Report* pada ulasan *App Store* dapat dilihat pada Gambar 16.

| Classification Report: | | | | |
|------------------------|-----------|--------|----------|---------|
| | precision | recall | f1-score | support |
| negative | 0.96 | 0.47 | 0.63 | 47 |
| neutral | 0.66 | 0.88 | 0.75 | 78 |
| positive | 0.73 | 0.68 | 0.70 | 40 |
| accuracy | | | 0.72 | 165 |
| macro avg | 0.78 | 0.68 | 0.69 | 165 |
| weighted avg | 0.76 | 0.72 | 0.71 | 165 |

Gambar 16. *Classification Report SVM* pada Ulasan *App Store*

Evaluasi *SVM* dan *SMOTE*: Pada tahap ini merupakan hasil Evaluasi menggunakan *Confusion Matrix* terhadap model *SVM* setelah dilakukan *SMOTE* dengan data ulasan pengguna *Disney+ Hotstar* pada *Play Store*. Hasil *Confusion Matrix* dapat dilihat pada Gambar 17.



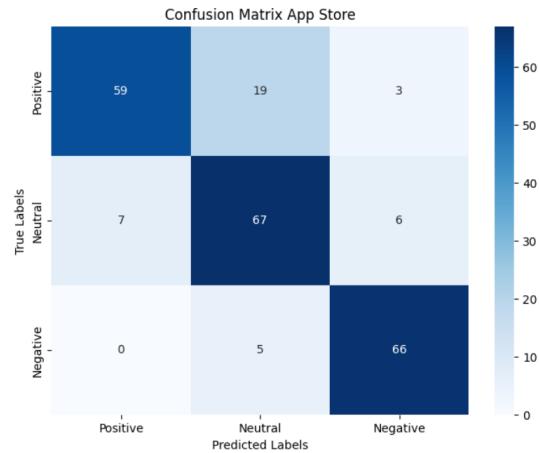
Gambar 17. *Confusion Matrix SVM* dan *SMOTE* pada Ulasan *Play Store*

Kemudian pada Hasil *Classification Report* dari model klasifikasi *SVM* setelah dilakukan *SMOTE* menghasilkan accuracy keseluruhan sebesar 0.74 atau sebesar 74%. Kelas *Negative* memiliki *presisi* 0.83, *recall* 0.73, dan *f1-score* 0.77, Kelas *Neutral* memiliki *presisi* 0.63, *recall* 0.76, dan *f1-score* 0.69, dan Kelas *Positive* memiliki *presisi* 0.83, *recall* 0.73, dan *f1-score* 0.78. *Classification Report* pada ulasan *Play Store* dapat dilihat pada Gambar 18.

| Classification Report: | | | | |
|------------------------|-----------|--------|----------|---------|
| | precision | recall | f1-score | support |
| negative | 0.83 | 0.73 | 0.77 | 66 |
| neutral | 0.63 | 0.76 | 0.69 | 68 |
| positive | 0.83 | 0.73 | 0.78 | 60 |
| accuracy | | | 0.74 | 194 |
| macro avg | 0.76 | 0.74 | 0.75 | 194 |
| weighted avg | 0.76 | 0.74 | 0.75 | 194 |

Gambar 18. *Classification Report SVM* dan *SMOTE* pada Ulasan *Play Store*

Pada tahap selanjutnya merupakan hasil Evaluasi menggunakan *Confusion Matrix* terhadap model *SVM* setelah dilakukan *SMOTE* dengan data ulasan pengguna *Disney+ Hotstar* pada *App Store*. Hasil *Confusion Matrix* dapat dilihat pada Gambar 19.



Gambar 19. *Confusion Matrix SVM* dan *SMOTE* pada Ulasan *App Store*

Selanjutnya, metode *SVM* setelah dilakukan *SMOTE* menghasilkan accuracy keseluruhan sebesar 0.83 atau sebesar 83%. Kelas *Negative* memiliki *presisi* 0.89, *recall* 0.73, dan *f1-score* 0.80, Kelas *Neutral* memiliki *presisi* 0.74, *recall* 0.84, dan *f1-score* 0.78, dan Kelas *Positive* memiliki *presisi* 0.88, *recall* 0.93, dan *f1-score* 0.90. *Classification Report* pada ulasan *App Store* dapat dilihat pada Gambar 20.

| Classification Report: | | | | |
|------------------------|-----------|--------|----------|---------|
| | precision | recall | f1-score | support |
| negative | 0.89 | 0.73 | 0.80 | 81 |
| neutral | 0.74 | 0.84 | 0.78 | 80 |
| positive | 0.88 | 0.93 | 0.90 | 71 |
| accuracy | | | 0.83 | 232 |
| macro avg | 0.84 | 0.83 | 0.83 | 232 |
| weighted avg | 0.84 | 0.83 | 0.83 | 232 |

Gambar 20. *Classification Report SVM* dan *SMOTE* pada Ulasan *App Store*

Perbandingan Performa: Pada tahap ini merupakan hasil dari perbandingan performa model *SVM* dan model *SVM* yang telah dioptimalkan menggunakan *SMOTE*, berdasarkan akurasi dalam mengklasifikasikan ulasan pengguna *Disney+ Hotstar* pada *Play Store* dan *App Store*. Perbandingan ini memiliki tujuan untuk mengetahui sejauh mana penyeimbangan data menggunakan *SMOTE* dapat meningkatkan performa model *SVM* dalam menangani ketidakseimbangan kelas, khususnya dalam mengidentifikasi sentimen positif, negatif, dan netral dari ulasan pengguna.

Tabel 9 adalah tabel yang menampilkan perbandingan akurasi pada ulasan *Play Store*. Dari tabel tersebut

terlihat adanya peningkatan kinerja model setelah diterapkan teknik *oversampling* menggunakan *SMOTE*. Awalnya, akurasi model berada di angka 67%, namun setelah dilakukan *oversampling*, akurasi meningkat menjadi 74%. Ini menunjukkan bahwa penerapan *SMOTE* efektif dalam meningkatkan performa model, dengan peningkatan akurasi sebesar +7%.

Tabel 9. Perbandingan Performa Ulasan *Play Store*

| Performa | SVM | SVM & SMOTE | Peningkatan |
|----------|-------|-------------|-------------|
| Akurasi | 0,67% | 0,74% | + 0,7% |

Kemudian, Tabel 10 merupakan tabel perbandingan akurasi pada ulasan *App Store*. Terlihat peningkatan kinerja model setelah penerapan *SMOTE*, di mana akurasi awal 72% meningkat menjadi 83%, menunjukkan peningkatan sebesar +11%. Teknik ini membantu mengatasi ketidakseimbangan data, sehingga model dapat melakukan klasifikasi dengan lebih baik dan menghasilkan hasil yang lebih akurat.

Tabel 10. Perbandingan Performa Ulasan *App Store*

| Performa | SVM | SVM & SMOTE | Peningkatan |
|----------|-------|-------------|-------------|
| Akurasi | 0,72% | 0,83% | + 0,11% |

4. Kesimpulan

Berdasarkan hasil pengujian dan analisis yang telah dilakukan menggunakan 1.650 dataset menunjukkan bahwa sentimen pengguna aplikasi *Disney+ Hotstar* cenderung memiliki nilai netral pada *Play Store* maupun *App Store*, hal ini terbukti dari hasil pelabelan kata menggunakan *Vader Lexicon*. Kemudian, berdasarkan hasil klasifikasi model menunjukkan bahwa penggunaan optimasi *SMOTE* terbukti berhasil dapat meningkatkan performa model *SVM*. Dengan mengalami peningkatan *accuracy* sebesar +0,7 pada ulasan *Play Store* dari *accuracy* awal 0.67 menjadi 0.74 dan peningkatan *accuracy* sebesar +0.11 pada ulasan *App Store* dari *accuracy* awal 0.72 menjadi 0.83. Penelitian ini juga berupaya memberikan saran bagi penelitian di masa mendatang, seperti menggunakan algoritma yang berbeda dan tidak terbatas pada analisis sentimen mengenai layanan streaming, tetapi juga topik lainnya. Selain itu, disarankan untuk menerapkan metode seleksi fitur untuk menyaring fitur-fitur yang kurang relevan atau tidak berkontribusi signifikan. Seleksi fitur ini bertujuan untuk menghilangkan data yang tidak diperlukan, sehingga dapat membantu meningkatkan efisiensi dan akurasi model klasifikasi dalam analisis teks.

Daftar Rujukan

- [1] M. Y. Siregar *et al.*, “Analisis Sentimen Pada Ulasan Pengguna Aplikasi Streaming Vidio Menggunakan Metode Naïve Bayes,” vol. 4, no. 5, pp. 2419–2429, 2024, doi: 10.30865/klik.v4i5.1787.

- [2] A. Y. Permatasari, “BAB II Tinjauan Pustaka BAB II TINJAUAN PUSTAKA 2.1. 1–64,” *Gastron. ecuatoriana y Tur. local.*, vol. 1, no. 69, pp. 5–24, 2023.
- [3] A. U. Chasanah, “Tinjauan Hukum Islam Mengenai Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 14 Tahun 2019 BAB II Bagian Kedua Tentang Kriteria Penyensoran Film,” no. 18913053, pp. 1–66, 2020.
- [4] A. Faadilah, “Analisis Sentimen Pada Ulasan Aplikasi Tokopedia di Google Play Store Menggunakan Metode Long Short Term Memory,” pp. 1–46, 2020.
- [5] S. Rahayu, Y. MZ, J. E. Bororing, and R. Hadiyat, “Implementasi Metode K-Nearest Neighbor (K-NN) untuk Analisis Sentimen Kepuasan Pengguna Aplikasi Teknologi Finansial FLIP,” *Edumatic J. Pendidik. Inform.*, vol. 6, no. 1, pp. 98–106, 2022, doi: 10.29408/edumatic.v6i1.5433.
- [6] A. Sanjaya *et al.*, “PENERAPAN ALGORITMA NAÏVE BAYES CLASSIFIER MENGENAI SENTIMEN TERHADAP APLIKASI STREAMING FILM ONLINE NETFLIX,” pp. 1038–1044, 2024.
- [7] J. W. Iskandar and Y. Nataliani, “Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 6, pp. 1120–1126, 2021, doi: 10.29207/resti.v5i6.3588.
- [8] R. N. Ikhnsani and F. F. Abdulloh, “Optimasi SVM dan Decision Tree Menggunakan SMOTE Untuk Mengklasifikasi Sentimen Masyarakat Mengenai Pinjaman Online,” *J. Media Inform. Budidarma*, vol. 7, pp. 1667–1677, 2023, doi: 10.30865/mib.v7i4.6809.
- [9] M. I. Fikri, T. S. Sabrina, and Y. Azhar, “Comparison of Naïve Bayes and Support Vector Machine Methods in Twitter Sentiment Analysis,” *Smatika J.*, vol. 10, no. 02, pp. 71–76, 2020.
- [10] S. Ernawati and R. Wati, “Evaluasi Performa Kernel SVM dalam Analisis Sentimen Review Aplikasi ChatGPT Menggunakan Hyperparameter dan VADER Lexicon,” vol. 15, no. April, pp. 40–49, 2024.
- [11] C. H. Yutika and S. Al Faraby, “Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes,” vol. 5, no. April, pp. 422–430, 2021, doi: 10.30865/mib.v5i2.2845.
- [12] R. Wati, S. Ernawati, and H. Rachmi, “Pembobatan TF-IDF Menggunakan Naïve Bayes pada Sentimen Masyarakat Mengenai Isu Kenaikan BIPIH,” *J. Manaj. Inform.*, vol. 13, no. 1, pp. 84–93, 2023, doi: 10.34010/jamika.v13i1.9424.
- [13] Styawati, Andi Nurkholis, Zaenal Abidin, and Heni Sulistiani, “Optimasi Parameter Support Vector Machine Berbasis Algoritma Firefly Pada Data Opini Film,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 5, pp. 904–910, 2021, doi: 10.29207/resti.v5i5.3380.
- [14] H. C. Husada and A. S. Paramita, “Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM),” *Teknika*, vol. 10, no. 1, pp. 18–26, 2021, doi: 10.34148/teknika.v10i1.3111.
- [15] Q. N. A. Andreystha, “Analisa Sentimen Kicauan Twitter Tokopedia Dengan Optimalisasi Data Tidak Seimbang Menggunakan Algoritma SMOTE,” *Infotek J. Inform. dan Teknol.*, vol. 5, no. 1, pp. 108–116, 2022, doi: 10.29408/jit.v5i1.4581.
- [16] K. Pramayasa, I. M. D. Maysanjaya, and I. G. A. A. D. Indradewi, “Analisis Sentimen Program Mbkm Pada Media Sosial Twitter Menggunakan KNN Dan SMOTE,” *SINTECH (Science Inf. Technol. J.*, vol. 6, no. 2, pp. 89–98, 2023, doi: 10.31598/sintechjournal.v6i2.1372.
- [17] S. Probongrum and Aciyahma Sidauruk, “Sistem Pendukung Keputusan Pemilihan Supplier Kain Dengan Metode Moora,” *JSII (Jurnal Sist. Informasi)*, vol. 8, no. 1, pp. 43–48, 2021, doi: 10.30656/jsii.v8i1.3073.