PERBANDINGAN MODEL *MACHINE LEARNING* DALAM ANALISIS SENTIMEN ULASAN PENGGUNA APLIKASI *E-COMMERCE*

[1*]Budi Utami Fahnun, [2]Sy Aifero Sayyidina Muhammad
[1,2]Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
[1]bufahnun@staff.gunadarma.ac.id, [2]aifero@student.gunadarma.ac.id

Abstrak

Fitur rating dan ulasan di Google Play Store memiliki dampak besar terhadap keputusan pengguna dalam mengunduh aplikasi, karena ulasan dari pengguna sebelumnya sering menjadi faktor utama dalam menilai kualitas suatu produk. Penelitian ini melakukan analisis sentimen terhadap ulasan aplikasi e commerce menggunakan pendekatan pelabelan ulasan berdasarkan skor dan membandingkan kinerja algoritma klasifikasi Multinomial Naïve Bayes, Random Forest, Extra Trees Classifier, dan Support Vector Machine (SVM). Dataset yang digunakan terdiri dari 2000-2200 baris data, dengan dua skema pembagian rasio data latih dan data uji yaitu 6:4 dan 8:2, untuk menemukan model dengan kinerja terbaik. Hasil pelabelan menggunakan pendekatan score-based labeling menunjukkan bahwa ulasan negatif paling dominan pada aplikasi Shopee dan Tokopedia, sementara ulasan positif mendominasi aplikasi Blibli. Eksperimen menunjukkan bahwa SVM secara konsisten mencapai nilai akurasi tertinggi pada dataset dari ketiga aplikasi tersebut. Secara spesifik, akurasi yang dicapai pada dataset Shopee adalah 90% dengan rasio data latih dan data uji 8:2, pada dataset Blibli adalah 91% dengan rasio 6:4, dan pada dataset Tokopedia adalah 83% dengan rasio 8:2. Capaian didapat dengan kombinasi hyperparameter kernel linear, Radial Basis Function (RBF), polinomial, dan sigmoid serta tingkat penalti terhadap kesalahan klasifikasi (c) dengan nilai 0.1, 1, dan 10. Temuan ini membuktikan algoritma SVM menunjukkan performa terbaik.

Kata kunci: *analisis sentimen, Extra Trees Classifier, machine learning, Multinomial Naïve Bayes, Random Forest, SVM, ulasan e-commerce*

Abstract

The rating and review features on Google Play Store have a significant impact on users' decisions to download applications, as previous user reviews are often a primary factor in assessing product quality. This study conducts sentiment analysis on e-commerce application reviews using a score-based review labeling approach and compares the performance of the Multinomial Naïve Bayes, Random Forest, Extra Trees Classifier, and Support Vector Machine (SVM) classification algorithms. The dataset used consists of 2000-2200 rows of data, with two train-test data split ratios (6:4 and 8:2) to identify the best-performing model. The results of score-based labeling indicate that negative reviews are most dominant for Shopee and Tokopedia applications, while positive reviews dominate Blibli's application. Experiments show that SVM consistently achieves the highest accuracy across datasets from all three applications. Specifically, the accuracy achieved on the Shopee dataset is 90% with an 8:2 train-test ratio, 91% on the Blibli dataset with a 6:4 ratio, and 83% on the Tokopedia dataset with an 8:2 ratio. These results were obtained using a combination of hyperparameters, including linear, Radial Basis Function (RBF), polynomial, and sigmoid kernels, and penalty parameters (c) with values of 0.1, 1, and 10. These findings demonstrate that the SVM algorithm exhibits the best performance.

Keywords: *e-commerce reviews, Extra Trees Classifier, machine learning, Multinomial Naïve Bayes, Random Forest, sentiment analysis, SVM*

PENDAHULUAN

Transaksi jual beli secara elektronik melalui *e-commerce* terus berkembang dengan cepat setiap tahunnya. Konsep pasar dalam bisnis ini mirip dengan pasar tradisional, di mana penjual dapat menawarkan produk mereka secara daring melalui internet. Hal ini memungkinkan bagi para pedagang untuk memperluas jangkauan pasar mereka tanpa harus memiliki toko fisik [1], [2], [3]. Pertumbuhan yang signifikan dalam industri ini telah menyebabkan munculnya berbagai media jual-beli *online* berbasis aplikasi misalnya Shopee, Blibli, dan Tokopedia yang bisa diunduh di Google Play Store [4], [5], [6].

Fitur rating dan ulasan di Google Play Store memiliki pengaruh yang signifikan ter-hadap keputusan pengguna lain dalam meng-unduh aplikasi, karena ulasan dari pengguna sebelumnya seringkali menjadi pertimbangan utama dalam menentukan kualitas produk [7], [8]. [9] sehingga berperan sangat penting dalam menentukan popularitas suatu aplikasi. Pendapat dan ulasan pengguna memegang peranan penting dalam membentuk citra suatu perusahaan. Ulasan yang ditampilkan tersebut menampung keluhan ataupun apresiasi yang dirasakan oleh para masing-masing pengguna aplikasi di waktu yang bersamaan [10].

Pada *marketplace*, pembeli dapat memberikan ulasan produk yang dibeli yang berisi sumber informasi tentang kualitas produk dan akan sangat berpengaruh pada konsumen dan produsen. Pelanggan yang merasa puas ataupun tidak dengan produk yang ditawarkan sering kali menuliskannya di media sosial. Ulasan pembelian produk terdiri dari bintang dan ulasan yang berisi tanggapan, apresiasi maupun kritik pada produk yang telah dibeli. Opini pelanggan yang dituliskan di media sosial dalam bentuk ulasan akan memberikan pengaruh pada calon pelanggan lain dalam penggunaan aplikasi *e-commerce* [9,10], maka kepuasan pelanggan merupakan hal penting yang menjadi tujuan perusahaan. Opini dapat berisi informasi yang tidak lengkap, informasi yang tidak konsisten serta jumlah data ulasan produk yang besar dan sangat banyak untuk setiap produk pada *e-commerce* serta informasi yang ada dalam opini akan membutuhkan analisis berbasis teks yang kompleks. Cara untuk mengekstrak informasi penting dan membangun sistem yang dapat menentukan kualitas produk secara objektif dan secara otomatis untuk menangani informasi teks yang sangat besar dan banyak adalah dengan menggunakan sistem analisis sentimen [11].

Analisis sentimen adalah suatu teknik untuk mengolah berbagai pendapat yang diekspresikan dalam ruang lingkup teks dengan memanfaatkan media yang berkaitan erat pada suatu hal. Sentimen berisi informasi tekstual serta mempunyai polaritas yang dapat bersifat positif, netral, dan negatif. Polaritas tersebut dapat dijadikan sebuah acuan untuk melakukan penentuan sebuah keputusan dalam pengembangan aplikasi dan peningkatan kepuasan pengguna [11,[12]

Aplikasi *e-commerce* telah tersedia Google Playstore. Aplikasi *e-commerce* yang paling banyak diunduh adalah Shopee yang telah dirilis pada tahun 2015. Shopee memiliki 295 juta pengguna aktif pada tahun 2023 dengan pasar terbesar di Indonesia yang memiliki 103 juta pengguna [13], [14]. Berdasarkan data pada Google Play Store hingga bulan Mei 2024, aplikasi Shopee telah diunduh sebanyak lebih dari 100 juta kali dengan rating 4.6 dari 5 dan memiliki 14 juta ulasan dari pengguna yang telah memakai aplikasi ini. Aplikasi Blibli sudah ada sejak tahun 2013 telah diunduh sebanyak lebih dari 10 juta kali dengan rating 4.7 dari 5 dan memiliki 600 ribu ulasan dari pengguna yang telah memakai aplikasi ini. Aplikasi Tokopedia juga merupakan salah satu pemain utama di pasar *e-commerce* Indonesia. Diluncurkan pada tahun 2014, Tokopedia telah berkembang pesat dan menjadi media yang sangat populer di kalangan konsumen Indonesia. Hingga Mei 2024, aplikasi Tokopedia di Google Play Store telah diunduh lebih dari 100 juta kali. Aplikasi ini memiliki rating 4.6 dari 5 berdasarkan 6 juta ulasan dari pengguna yang telah memakai aplikasi ini.

Penelitian terkait analisis sentimen telah dilakukan [13] yang memfokuskan pada aplikasi Shopee ke dalam tiga kategori yaitu negatif, netral, dan positif. Penelitian tersebut menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF) sebagai teknik vektorisasi teks dan ekstraksi fiturnya serta mengaplikasikan *Random Forest* sebagai model klasifikasi. Hasil pengujian model *Random Forest* menunjukkan akurasi 94%, presisi 91%, *recall* 91%, dan F1-*score* 93% [13]. *Multinomial Naïve Bayes* juga digunakan untuk mengklasifikasikan teks ke dalam kelas positif dan negatif dengan akurasi 88,89% berdasarkan 536 *dataset* dengan rincian 439 data positif dan 97 data negatif [14]. Penelitian dengan metode *Extra Trees Classifier* telah dilakukan [15] serta menambahkan tahap optimasi menggunakan *hyperparameter* dengan *Grid Search Cross-Validation* (CV). Sebelum optimasi, akurasi mencapai 95.85%, setelahnya meningkat menjadi 96.26%, menunjukkan peningkatan sebesar 0.41%.

Support Vector Machine (SVM) juga digunakan untuk mengklasifikasikan ulasan dalam dua kategori sentiment yaitu positif dan negatif dari pengguna aplikasi belanja daring di Google Play Store. Penelitian ini menggunakan evaluasi 10-*fold cross validation* dengan 3000 ulasan, dengan hasil akurasi Tokopedia sebesar 90.67%, JD.ID 75.33%, Blibli 74.00%, Shopee 70.00%, dan Lazada 69.00% [16]. Selain akurasi yang tinggi, kelebihan SVM juga memiliki efektifitas terkait dataset dengan dimensi tinggi dan juga memori, mampu meminimalkan konvergensi pada minimum lokal dan tinggi saat memecahkan masalah yang kompleks [17]. SVM juga merupakan alat berbasis kernel yang membangun *hyperplane* optimal antara dua *hyperplane support* paralel, yang menetapkan jarak optimal antara dua

hyperplane paralel. Efektivitasnya dalam masalah multikelas telah menjadikannya pilihan untuk beberapa aplikasi.

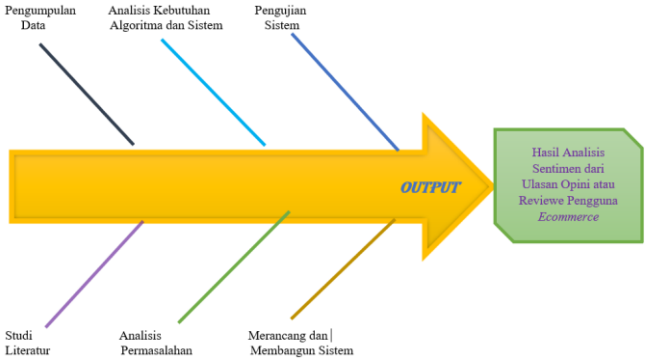
Penelitian [18] mengeksplorasi analisis sentimen ulasan pengguna aplikasi *e-commerce* di Google Play Store dengan menggunakan dua algoritma *machine learning*. Penelitian tersebut membagi *dataset* menjadi dua kondisi, menggunakan dua label (positif dan negatif) dan tiga label (positif, netral, negatif), yang diterapkan pada kedua algoritma tersebut. Hasil menunjukkan bahwa *Logistic Regression* dengan dua label pada *dataset* Shopee mencapai akurasi tertinggi, yaitu 84.58% dan 84.33% [18]. Penelitian ini relevan karena mempertimbangkan tiga kategori sentimen, namun peneliti menggunakan pendekatan yang berbeda dengan mengaplikasikan varian khusus dari *Naïve* Bayes, yaitu *Multinomial Naïve* Bayes, yang paling sesuai untuk data diskrit seperti teks, di mana fitur-fitur berupa frekuensi kata atau yang sejenisnya, sehingga cocok dengan penelitian ini yang menerapkan vektorisasi data TF-IDF. Penelitian ini menggunakan metode TF-IDF sebagai teknik statistik dalam sistem temu kembali informasi. TF-IDF berfungsi untuk mengukur pentingnya suatu *term* dalam sebuah dokumen relatif terhadap seluruh koleksi dokumen. Nilai TF-IDF yang dihasilkan memungkinkan pemeringkatan dokumen, pencarian informasi yang relevan, pemfilteran dokumen, dan ekstraksi ringkasan. Dengan demikian, sistem dapat membandingkan secara kuantitatif tingkat relevansi antara dokumen dengan *query* atau permintaan pengguna [19].

Penelitian sebelumnya memberikan fondasi yang kuat dan justifikasi untuk penelitian ini, yang memperluas cakupan dengan menguji berbagai algoritma dan pengaturan data yang lebih beragam. Tujuan penelitian ini adalah membandingkan empat algoritma *machine learning* yaitu *Multinomial Naïve Bayes, Random Forest, Extra Trees Classifier,* dan SVM untuk analisis sentimen ulasan pengguna aplikasi *e-commerse* Shopee, Blibli, dan Tokopedia ke dalam tiga kategori sentimen (negatif, netral, dan positif). Pada pemcagian dataset digunakan pembagian data yang berbeda (6:4 dan 8:2) dan menerapkan TF-IDF. Penelitian ini diharapkan dapat digunakan untuk mengevaluasi layanan aplikasi *e-commerce* dengan lebih efektif. Selain itu, penelitian ini bertujuan untuk mengisi kesenjangan pengetahuan dengan menganalisis ulasan pengguna aplikasi pada *e-commerce* menggunakan pendekatan yang sama. Penelitian ini diharapkan dapat memberikan wawasan yang berharga bagi pemilik bisnis dan pengembang aplikasi dalam meningkatkan kualitas layanan mereka dan calon pengguna aplikasi dalam menentukan aplikasi yang sesuai.

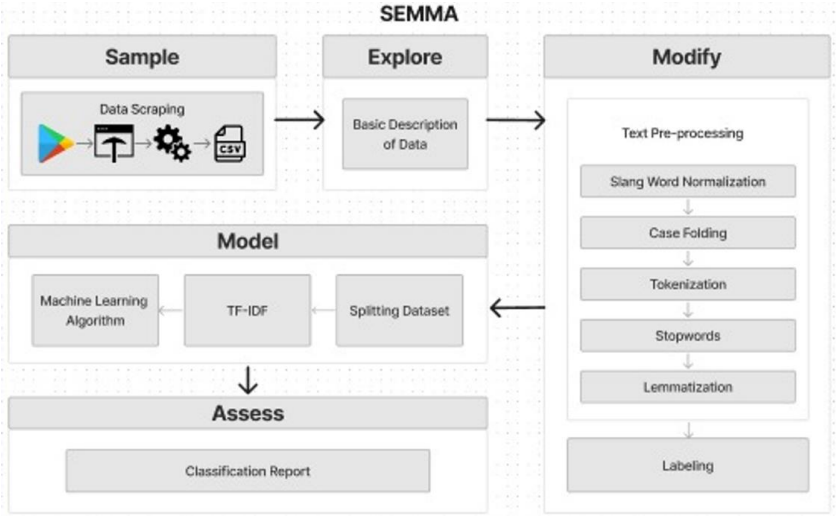
METODE PENELITIAN

Kerangka penelitian disajikan pada Gambar 1. Pada tahapan Studi literatur yang ditelaah terkait dengan ulasan produk, sistem

analisis sentimen, algoritma dan konsep lainnya yang terkait dengan masalah penelitian ini. Pengumpulan data adalah terkait ulasan produk aplikasi *e-commerce*. Analisis permasalahan adalah menganalisis permasalahan terkait ulasan produk oleh pengguna. Analisis kebutuhan algoritma dan sistem bertujuan untuk menganalisis kebutuhan penerapan algoritma dan pembuatan sistem analisis sentimen yang akan digunakan untuk penyelesaian masalah pada penelitian ini. Merancang dan membangun sistem analisis sentimen dengan menerapkan algoritma ke dalam sistem untuk mengklasifikasikan sentimen ke dalam kriteria positif, netral, dan negatif. Pengujian sistem dilakukan dengan menggabungkan modul-modul yang sudah dibuat untuk memverifikasi kesesuaian sistem dengan desain awal. Pengujian yang dilakukan dengan memasukkan data ulasan pelanggan untuk mengevaluasi akurasi klasifikasi sentimen.



Gambar 1. Kerangka Penelitian



Gambar 2. Diagram Alir Proses Penelitian

Pada Gambar 2 alur penelitian diawali dengan mengumpulan data penelitian yang dikumpulkan melalui teknik scraping data menggunakan API yang tersedia dari Google Play Store, dengan implementasi menggunakan pustaka Python yang disebut Google-Play-Scraper. Proses scraping berlangsung dari tanggal 10 Mei hingga 15 Mei 2024, dilakukan secara terpisah untuk setiap aplikasi dan diulang beberapa kali. Proses ini berlanjut hingga jumlah data yang diperlukan tercapai, karena terdapat ketidaksesuaian antara jumlah data yang diekstraksi dan yang diminta oleh kode program. Setelah itu, dataset dari masing-masing ulasan aplikasi digabungkan, dan diakhiri dengan penghapusan data duplikat. Ulasan yang dikumpulkan diurutkan berdasarkan relevansi tertinggi (most relevant). Setelah proses scraping selesai, hasilnya disimpan dalam format file .csv di Google Drive, yang dipilih untuk penyimpanan data karena memudahkan akses saat menggunakan Google Colaboratory.

Explore

Eksplorasi *dataset* melibatkan deskripsi kondisi data yang diperoleh melalui proses *scraping*, serta penentuan kolom-kolom yang akan digunakan dalam penelitian. Selanjutnya, data divisualisasikan pada tahap awal penelitian untuk menunjukkan distribusi berdasarkan skor dan label dalam bentuk diagram serta *wordcloud* kata mayoritas.

Modify

Pada tahap *modify* dilakukan seleksi dan transformasi elemen penyusun *dataset* sehingga menghasilkan data yang ternormalisasi, bertujuan untuk mengubah *dataset* ke dalam bentuk data terstruktur untuk kebutuhan pemrosesan lebih lanjut seperti analisis sentimen dan pemodelan. Tahap *text preprocessing* terdiri dari beberapa langkah yaitu *slang word normalization, case folding*, tokenisasi, *stopwords*, dan *lemmatization* yang dilanjutkan dengan proses pelabelan sentimen.

Pada tahap *slang word normalization*, bahasa tidak resmi diubah menjadi kata baku menggunakan *SymSpell dan Verbosity* dari pustaka *symspellpy*, yang digunakan untuk koreksi kesalahan ketik dan saran kata. Pada tahap *case folding* semua huruf pada dokumen diubah menjadi huruf kecil dan melakukan eliminasi karakter selain huruf “a” sampai dengan “z” dan angka “1” sampai dengan “9”. Pada awalnya tipe data dari ulasan adalah *string*, karena ulasan terdiri dari kumpulan kata dan kata terdiri dari karakter huruf dan angka. Pada tahap tokenisasi setiap kata penyusun ulasan pada tahap *case folding* dijadikan token penyusun *array*. Tipe data *string* diubah menjadi *array* pada tahap ini bertujuan untuk pemrosesan data lebih lanjut khususnya pada tahap pelabelan sentimen. Pada tahap *stopwords* dilakukan pembersihan dan penyederhanaan teks agar lebih mudah dipahami oleh model pembelajaran mesin atau algoritma lainnya. *Stopwords* adalah kata-kata

umum yang sering muncul dalam sebuah teks dan umumnya tidak memberikan nilai tambah dalam pemrosesan atau analisis teks. Contohnya adalah "the", "is", "at", dan "which". *Lemmatization* adalah proses untuk mengubah kata-kata dalam bentuk berimbuhan (*inflected forms*) menjadi bentuk dasarnya (*lemma*). Tahap ini bertujuan untuk membantu dalam mengurangi variasi kata yang sama menjadi bentuk dasarnya sehingga memudahkan pemrosesan dan analisis teks. Misalnya, kata-kata "running", "ran", dan "runs" akan dilematkan menjadi "run".

Pada tahap *labelling*, *dataset* diberi label positif, netral, atau negatif. Proses labelisasi pada penelitian ini menggunakan teknik *score based labelling*. Hasil labelisasi akan disebut sentimen aktual pada penelitian ini. Pada tahap ini *dataset* diberi label untuk membagi *dataset* menjadi tiga kelas sentimen berdasarkan nilai rating bintang pada ulasan di Google Play Store. Bintang 1 sampai 2 dikategorikan negatif, bintang 3 dikategorikan netral, dan bintang 4 sampai 5 dikategorikan positif.

Pembagian Dataset

Setelah *dataset* telah diidentifikasi orientasi sentimennya, langkah berikutnya adalah membagi *dataset* menjadi data latih dan data uji. Data latih berfungsi untuk melatih model dan membangun pengetahuan, sedangkan data uji digunakan untuk meng-evaluasi akurasi model dalam meng-klasifikasikan sentimen suatu ulasan. Pada penelitian ini, data latih dan data uji dibagi dengan dua rasio yang berbeda, yaitu 8:2 dan 6:4. Tabel 1 adalah rincian pembagian data latih dan data uji untuk setiap *dataset* aplikasi yang digunakan.

Tabel 1. Pembagian Dataset Penelitian

Rasio Pembagian Data	**Aplikasi**	**Jenis Data**	***Banyak Data Tiap Kelas***			**Total**
			Positive	***Neutral***	***Negative***	
8:2	Shopee	Train	374	248	1134	1756
		Test	84	67	289	440
	Blibli	Train	1228	36	336	1600
		Test	286	18	96	400
	Tokopedia	Train	473	123	1004	1600
		Test	126	27	247	400
6:4	Shopee	Train	290	172	855	1317
		Test	168	143	568	879
	Blibli	Train	909	31	260	1200
		Test	605	23	172	800
	Tokopedia	Train	361	85	754	1200
		Test	238	65	497	800

TF-IDF

Metode TF-IDF adalah sebuah metode yang digunakan untuk menentukan bobot hubungan antara suatu kata (*term*) dan dokumen. Dalam konteks ini, setiap kalimat dalam dokumen dianggap sebagai entitas tersendiri. Metode ini mengintegrasikan dua konsep utama dalam perhitungannya. *Term Frequency* (TF) mencerminkan frekuensi kemunculan kata (*t*) dalam kalimat (*d*), sedangkan *Document Frequency* (DF) mengindikasikan jumlah kalimat dimana kata (*t*) muncul.

Frekuensi kemunculan suatu kata dalam dokumen menunjukkan tingkat pentingnya kata tersebut dalam konteks dokumen tersebut. Sementara itu, frekuensi dokumen yang mengandung kata tersebut mencerminkan seberapa umumnya kata tersebut dalam keseluruhan dokumen. Bobot kata dalam algoritma TF-IDF diperhitungkan dengan menggunakan Persamaan (3) untuk menghitung bobot (*W*) dari setiap dokumen terhadap kata kunci dengan Persamaan (1) dan (2) merupakan rumus untuk menghitung TF dan IDF.

$$tf = \frac{ft,d}{max\ (ft,d)} \qquad (1)$$

$$idf = log\ (\frac{N}{df_t}) \qquad (2)$$

$$W = tf \times idf \qquad (3)$$

di mana *d* adalah dokumen, *t* adalah kata pada dokumen, ft,d adalah frekuensi kata pada dokumen, *N* adalah total jumlah dokumen, dan df_t adalah jumlah dokumen yang mengandung kata-*t*.

Pelatihan Algoritma *Machine Learning*

Pada tahap ini, pelatihan dilakukan dengan menerapkan empat model algoritma *machine learning* serta optimasi *hyperparameter* untuk memperoleh model dengan kinerja optimal. Proses pelatihan ini diterapkan pada data latih guna menghasilkan model yang berkualitas tinggi. Keempat model algoritma tersebut terdiri dari *Multinomial Naïve Bayes, Random Forest, Extra Trees Classifier*, dan SVM.

Multinomial Naïve Bayes

Multinomial Naïve Bayes atau *Multinomial* NB adalah sebuah pengembangan dari algoritma Bayesian yang ideal untuk mengklasifikasikan teks atau dokumen. Dalam rumus *Multinomial Naïve Bayes*, kelas tidak hanya ditentukan oleh kehadiran kata, melainkan juga oleh frekuensi kemunculannya. Model multinomial ini mempertimbangkan jumlah kata yang muncul dalam dokumen, dengan mengasumsikan bahwa panjang dokumen tidak mempengaruhi kelasnya. Secara dasar Bayes, setiap kemunculan kata dalam dokumen diasumsikan *independent* dari konteks dan posisi kata tersebut dalam dokumen. Saat melakukan klasifikasi, algoritma mencari probabilitas tertinggi dari kelas yang diuji atau VMAP (*Variational Maximum a Posteriori*). Langkah awal dalam kategorisasi dokumen adalah menghitung probabilitas dengan menggunakan Persamaan (4).

$$P(c) = \frac{Nc}{N} \qquad (4)$$

di mana $P(ci)$ adalah probabilitas kelas, Nc merupakan jumlah kelas *c* pada seluruh

dokumen, dan *N* merupakan jumlah seluruh dokumen. Selanjutnya untuk memperkirakan *conditional probability* digunakan Persamaan (5).

$$P(w|c) = \frac{Wct + 1}{(\sum w' \in VWct') + B'} \quad (5)$$

di mana $P(w|c)$ adalah *conditional probability*, Wct adalah *W* dari *term t* di kategori *c*, $(\sum w' \in VWct')$ adalah jumlah total *W* dari keseluruhan *term* yang berada di kategori *c*, dan B' adalah nilai idf pada seluruh dokumen. Selanjutnya untuk mengatasi masalah nilai nol, digunakan metode *smoothing* seperti *add-one* atau *Laplace smoothing*. Dalam proses ini, nilai satu (1) ditambahkan pada setiap nilai Wct dari perhitungan *conditional probability*. Setelahnya, VMAP dari sebuah kalimat dapat dihitung menggunakan Persamaan (6).

$$VMAP = argmax\, P(x_1, x_2, x_3, \dots, x_n)P(c) \quad (6)$$

Dimana $P(x_1)$ adalah probabilitas pada kata dan $P(c)$ adalah probabilitas pada kelas. Kemudian setelah mendapatkan nilai VMAP, langkah terakhir adalah menentukan kelas kalimat tersebut. Jika nilai VMAP positif lebih tinggi dibandingkan dengan nilai VMAP negatif, maka kalimat tersebut diklasifikasikan sebagai positif. Sebaliknya, jika nilai VMAP negatif lebih tinggi, maka kalimat diklasifikasikan sebagai negatif.

Random Forest

Random Forest merupakan sebuah algoritma klasifikasi yang terdiri dari beberapa pohon keputusan atau *Decision Tree* yang dibangun dengan menggunakan vektor acak [8]. Metode *Random Forest* melibatkan dua tahap utama. Tahap pertama adalah pembentukan *forest*, di mana sejumlah *decision tree* dibangun berdasarkan sampel-sampel vektor acak. Tahap kedua adalah melakukan voting untuk hasil klasifikasi, di mana prediksi akhir diperoleh dengan mempertimbangkan mayoritas suara dari semua *decision tree* yang terlibat [9].

$$forest = \{h(x, \theta_k), k = 1, \dots\} \quad (7)$$

di mana *h* adalah hipotesa atau klasifikasi, *x* adalah *input vector*, dan θ_k adalah *independent and identically distributed random vectors*. Persamaan (7) menyatakan bahwa setiap *forest* terdiri dari sekumpulan klasifikasi atau hipotesis *k*. Setiap hipotesis ini mengambil *input x* yang kemudian di-*resampling* menggunakan vektor acak dari *x* itu sendiri [9].

$$C_{rf} = majority\ vote\ \{C_n(x)\} = 1 \quad (8)$$

di mana C_{rf} adalah *class* hasil klasifikasi Random Forest, *x* adalah *input vector*, dan C_n adalah *class* prediksi dari *tree* ke-*n* pada *Random Forest*. Kemudian setelah pembuatan *forest*, langkah selanjutnya adalah melakukan voting untuk klasifikasi dan mengukur performa *Random Forest* menggunakan Persamaan (8) [9].

Extra Trees Classifier

Extra Tree, atau juga dikenal sebagai *Extremely Randomized Trees*, merupakan suatu jenis teknik pembelajaran mesin

ansambel yang mengintegrasikan hasil dari beberapa pohon keputusan yang tidak saling berkorelasi, yang dikumpulkan dalam suatu "*forest*" untuk menghasilkan klasifikasi [10]. Tiap pohon keputusan dalam hutan *Extra Trees* dibangun menggunakan seluruh sampel pelatihan asli tanpa melakukan replikasi *bootstrap*, sehingga tidak ada proses *resampling*. Algoritma ini menggunakan tipe pohon CART (*Classification and Regression Tree*) untuk membangun setiap pohon. Mekanisme dari pohon keputusan melibatkan struktur tertentu, di mana setiap node internal menguji atribut tertentu, setiap cabang menunjukkan hasil dari pengujian tersebut, dan node daun (*leaf node*) mewakili kelas atau kategori.

Di setiap node pengujian, setiap pohon dalam *Extra Trees* menerima sampel acak dari *k* fitur dari *set* fitur yang tersedia. Tiap pohon keputusan harus memilih fitur terbaik untuk memisahkan data berdasarkan kriteria matematis seperti Indeks Gini. Penggunaan fitur acak ini memastikan bahwa setiap pohon keputusan dalam hutan tidak saling berkorelasi. Secara *default*, sebuah hutan *Extra Trees* memiliki sekitar 100 pohon keputusan, yang dapat disesuaikan jumlahnya dengan parameter n_estimators. Setelah dibangun 100 pohon keputusan, hasil klasifikasi diambil berdasarkan mayoritas suara (*majority vote*) dari seluruh pohon tersebut. Rumus untuk menghitung Indeks Gini digunakan seperti yang tercantum dalam Persamaan (9).

$$Gini(D) = 1 - \sum_{i=1}^{m} (p_i)^2 \qquad (9)$$

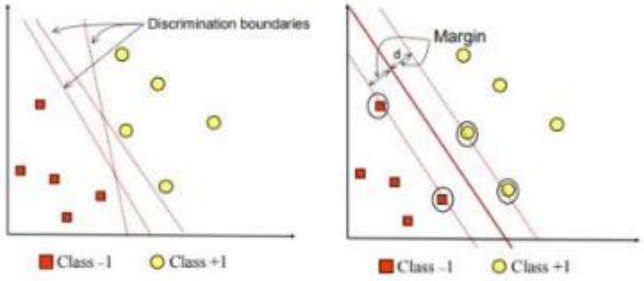
di mana *D* adalah *dataset* dari kelas, *m* adalah jumlah kelas variabel atribut, *i* adalah kelas atribut, dan p_i adalah proporsi jumlah kelas dalam atribut *i* terhadap jumlah kelas *c* dalam atribut. Selanjutnya setelah menghitung Indeks Gini, langkah selanjutnya adalah menentukan atribut mana yang akan digunakan dalam pembentukan node. Hal ini dilakukan dengan menggunakan rumus seperti yang tercantum dalam Persamaan (10).

$$Gini_A(D) = \frac{|D_1|}{D} gini(D_1) + \frac{|D_2|}{D} gini(D_2) \qquad (10)$$

di mana *A* adalah jumlah bobot indeks gini untuk atribut dan D_{12} adalah jumlah *dataset D* jika dibagi ke *A* untuk menjadi dua *subset* D_1 dan D_2. Kemudian dari persamaan tersebut, dihitung nilai bobot Indeks Gini untuk setiap fitur *A* dan setiap fitur lainnya. Setelah perhitungan semua fitur untuk Indeks Gini tersebut, langkah selanjutnya adalah memilih nilai fitur terkecil sebagai keputusan pertama dalam pembentukan node pada pohon.

SVM

Konsep dasar SVM terfokus pada penciptaan *hyperplane* optimal, yang juga dikenal sebagai batas keputusan atau batas optimal. *Hyperplane* ini dirancang untuk memaksimalkan jarak antara titik data terdekat (*support vector*) dan secara efisien memisahkan kelas-kelas yang berbeda [11].



Gambar 3. Ilustrasi *Hyperplane*

Gambar 3 menggambarkan sepasang *hyperplane* yang memisahkan dua kelas, dengan titik-titik kuning dan merah yang berada di dalam lingkaran hitam disebut sebagai *support vector*. Pada Gambar 4 juga terlihat *hyperplane* optimal yang dipilih karena menghasilkan margin yang maksimal. Fungsi *hyperplane* yang memisahkan dua kelas dapat dinyatakan dalam Persamaan (11) [12].

$$w_i x_i + b = 0 \qquad (11)$$

Gambar 4 menjelaskan bagaimana proses pemisahan *hyperplane*. Untuk menghitung margin terbesar dapat dilakukan dengan cara memaksimalkan nilai jarak antara titik terdekat yang ada dengan *hyperplane* yang dapat dirumuskan sebagai fungsi *Quadratic Programming* (QP) *problem*, yaitu mencari titik minimum dengan menggunakan Persamaan (12) [13].

$$\min \tau(w) = \frac{1}{2}||w||^2 \qquad (12)$$

Dengan syarat memperhatikan kendala yang dirumuskan ke dalam Persamaan (13).

$$y_i(x_i w + b) - 1 \geq 0, \forall I \qquad (13)$$

Untuk melakukan optimasi atau mencari nilai optimal, metode yang sering digunakan melibatkan penggunaan fungsi *Lagrange Multiplier*, yang direpresentasikan dalam bentuk Persamaan (14).

$$L = (w, b, \alpha) = \frac{1}{2}||w|| - \sum_{i=1}^{i} \alpha_i (y_i((x_i . w + b) - 1)) \qquad (14)$$

Variabel α_i dinyatakan sebagai *Lagrange Multiplier* yang bernilai positif atau nol. Untuk mencapai nilai optimal dari persamaan tersebut, pendekatan dilakukan dengan meminimalkan nilai *L* terhadap *w* dan *b*. Dalam konteks optimasi, strategi lain dapat diterapkan dengan memaksimalkan *L* terhadap α_i melalui penyesuaian Persamaan (15) untuk mencapai solusi maksimal yang hanya mempertimbangkan nilai α_i [13].

$$\sum_{i=1}^{i} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{i} \alpha_i \alpha_j y_i y_j x_i x_j \qquad (15)$$

dengan,

$$\alpha_i \geq 0 \, (i = 1,2, \ldots, l) \sum_{i=1}^{i} \alpha_i y_i = 0 \qquad (16)$$

Hasil perhitungan menunjukkan bahwa sebagian besar nilai α_i positif. Data yang terkait dengan α_i dan memiliki nilai positif dikenal sebagai *support vector*. SVM memiliki keunggulan dalam penggunaannya di ruang dimensi tinggi serta efisiensi memori yang baik karena menggunakan titik pelatihan dari fungsi keputusan (*support vector*).

https://doi.org/10.35760/tr.2025.v30i1.13530

Assess

Hasil pengujian model disebut sebagai prediksi sentimen dan direpresentasikan dalam bentuk *confusion matrix*. Performa suatu model dapat dievaluasi menggunakan *confusion matrix*. Secara sederhana, *confusion matrix* menampilkan jumlah prediksi yang benar dan salah. Dari *confusion matrix*, beberapa parameter turunan dapat dihitung, yaitu *accuracy*, *precision*, *recall*, dan F1-*score*. Selanjutnya disajikan rangkuman dari hasil perbandingan efektivitas beberapa algoritma *machine learning* dalam analisis sentimen. Analisis ini mengidentifikasi algoritma-algoritma yang menunjukkan kinerja tertinggi berdasarkan metrik evaluasi yang digunakan serta merangkum faktor-faktor yang berkontribusi terhadap performa masing-masing algoritma tersebut. Selain itu, diberikan rekomendasi untuk penelitian lanjutan di masa depan serta penerapan praktis dari temuan-temuan ini.

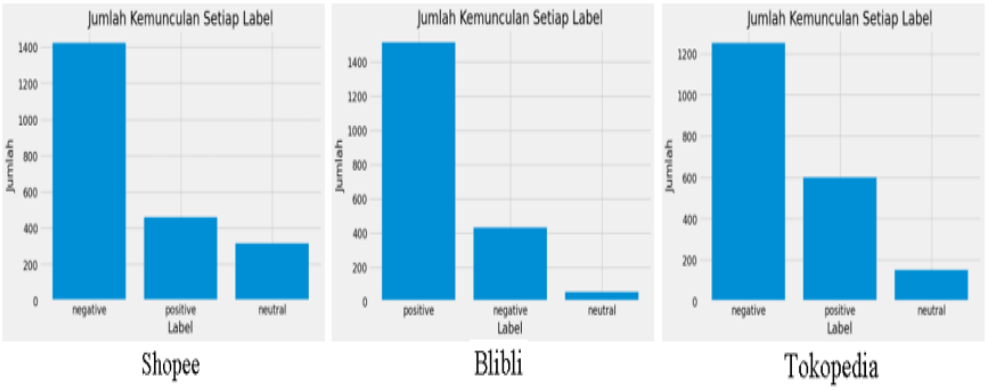
HASIL DAN PEMBAHASAN

Pada Tabel 2 ditunjukkan hasil *slang word normalization*, *case folding*, *tokenisasi*, *stopwords*, dan *lemmatization* untuk satu ulasan.

Tabel 2. Hasil Preprocessing Data Untuk Satu Ulasan

Ulasan	**Hasil Normalisasi *Slang Word***	**Hasil *Case Folding***	**Hasil Tokenisasi**	**Hasil *Stopwords***	**Hasil *Lemmatization***
Well this is very helpful app, I think **I've** been actively using it since 2020, it will be very helpful especially for people who are busy and **don't** have much time to shop directly, you can set the choice of store and budget range that you will allocate yourself, and the options pretty much.. eficient **n** simple	Well this is very helpful app, I think **I have** been actively using it since 2020, it will be very helpful especially for people who are busy and **do not** have much time to shop directly, you can set the choice of store and budget range that you will allocate yourself, and the options pretty much.. eficient **and** simple	**w**ell this is very helpful app, **i** think **i** have been actively using it since 2020, it will be very helpful especially for people who are busy and do not have much time to shop directly, you can set the choice of store and budget range that you will allocate yourself, and the options pretty much.. eficient and simple	['well', **'this'**, **'is'**, **'very'**, 'helpful', 'app', **'i'**, 'think', **'i'**, **'have'**, **'been'**, 'actively', 'using', **'it'**, 'since', '2020', **'it'**, **'will'**, **'be'**, **'very'**, 'helpful', 'especially', **'for'**, 'people', **'who'**, **'are'**, 'busy', **'and'**, **'do'**, **'not'**, **'have'**, **'much'**, 'time', '**to**', 'shop', 'directly', **'you'**, **'can'**, 'set', **'the'**, 'choice', **'of'**, 'store', **'and'**, 'budget', 'range', **'that'**, **'you'**, **'will'**, 'allocate', **'yourself'**, **'and'**, **'the'**, 'options', 'pretty', **'much'**, 'efficient', **'and'**, 'simple']	['well', 'helpful', 'app', 'think', 'actively', 'using', 'since', '2020', 'helpful', 'especially', 'people', 'busy', 'time', 'shop', 'directly', 'set', 'choice', 'store', 'budget', 'range', 'allocate', 'options', 'pretty', 'efficient', 'simple']	['well', 'helpful', 'app', 'think', 'actively', **'use'**, 'since', '2020', 'helpful', 'especially', 'people', 'busy', 'time', 'shop', 'directly', 'set', 'choice', 'store', 'budget', 'range', 'allocate', **'option'**, 'pretty', 'efficient', 'simple']

Hasil Pelabelan Data



Gambar 4. Distribusi Label Sentimen Aplikasi

Pada Gambar 4 ditunjukkan jumlah ulasan negatif menjadi sentimen yang paling dominan pada aplikasi Shopee dan Tokopedia, disusul dengan ulasan positif. Namun pada aplikasi Blibli sentimen positif menjadi yang terbanyak kemudian disusul dengan ulasan negatif. Pada ketiga aplikasi ulasan netral menjadi yang paling sedikit.

Hasil Performa Model

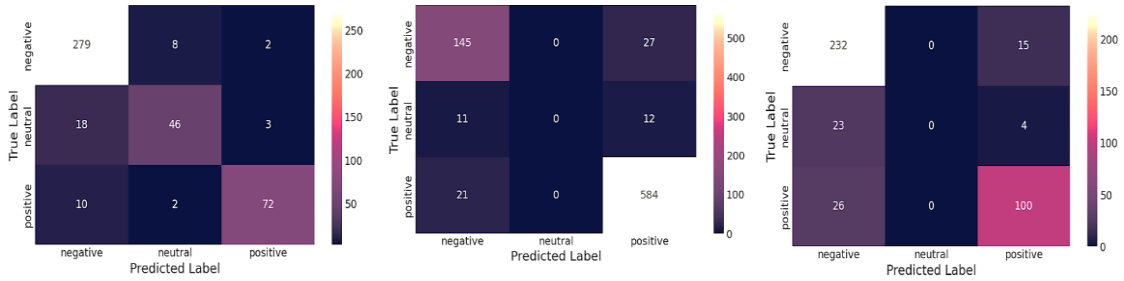
Evaluasi model *machine learning* dari hasil pelatihan menggunakan *confusion matrix* sebagai instrumen evaluasi yang terdiri dari beberapa variabel yaitu *accuracy, precision, recall, dan F1 score*. Tabel 3 adalah hasil percobaan yang menunjukkan kinerja terbaik dari beberapa algoritma *machine learning* terhadap aplikasi Shopee, Blibli, dan Tokopedia.

Berdasarkan Tabel 3, model dengan tingkat akurasi tertinggi untuk *dataset* ulasan aplikasi Shopee adalah *Random Forest, Extra Trees Classifier*, dan SVM dengan pembagian rasio data 8:2, yang mencapai akurasi sebesar 90%. Sementara itu, untuk *dataset* ulasan aplikasi Blibli, model *Multinomial Naïve Bayes* dan SVM berhasil meraih akurasi terbaik sebesar 91%. Meskipun demikian, model lain juga menunjukkan hasil akurasi yang cukup tinggi dengan rata-rata 89%. Untuk *dataset* ulasan aplikasi Tokopedia, model *Extra Trees Classifier* dan SVM mencatat akurasi terbaik dengan nilai sebesar 83%. Tingginya nilai akurasi ini mengindikasikan bahwa algoritma klasifikasi SVM mampu konsisten mengklasifikasikan sentimen dari setiap *dataset* ulasan aplikasi dengan baik.

Pengujian model menggunakan data uji menghasilkan *confusion matrix* yang bermanfaat untuk mengevaluasi ketepatan model dalam memprediksi ulasan sesuai dengan label sentimennya, yaitu negatif, netral, dan positif. Berikut adalah Gambar 5 yaitu *confusion matrix* dari model terbaik, yaitu SVM terhadap *dataset* ulasan aplikasi Shopee, Blibli, dan Tokopedia.

Tabel 3. Hasil Akurasi Model

Model Accuracy (%) with Best Hyperparameter					
			Application		
			Shopee	Blibli	Tokopedia
Classifier	*Multinomial*	8:2	80	89	82
	Naïve Bayes	6:4	76	91	81
	Random	8:2	90	87	81
	Forest	6:4	85	90	81
	Extra Trees	8:2	90	87	83
	Classifier	6:4	85	90	81
	SVM	8:2	90	88	83
		6:4	87	91	81



a. Shopee b. Blibli c. Tokopedia

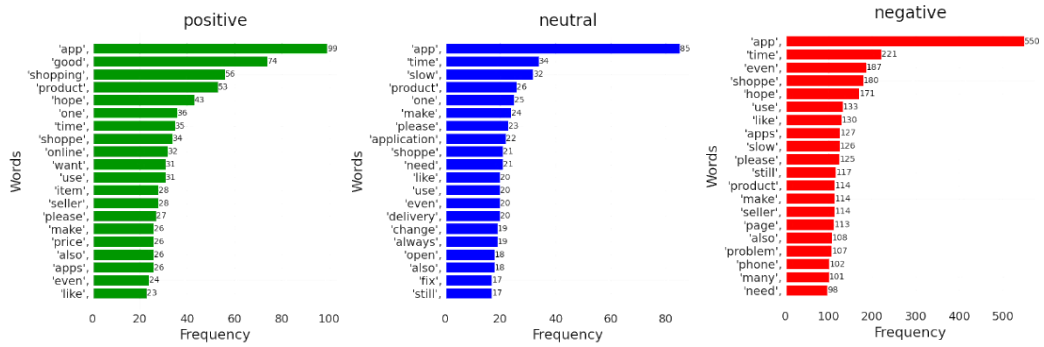
Gambar 5. *Confusion Matrix* Model SVM

Confusion matrix pada Gambar 5a menampilkan jumlah prediksi benar dan salah yang dihasilkan oleh model SVM terhadap *dataset* ulasan aplikasi Shopee. Model ini berhasil memprediksi 72 ulasan positif dari total 84 ulasan, berhasil memprediksi 46 untuk ulasan netral dari total 67 ulasan, dan 279 ulasan negatif dari total 289 ulasan dalam data uji. *Confusion matrix* pada Gambar 5b menampilkan jumlah prediksi benar dan salah yang dihasilkan oleh model SVM terhadap *dataset* ulasan aplikasi Blibli. Model ini berhasil memprediksi 584 ulasan positif dari total 605 ulasan, tidak ada prediksi untuk ulasan netral dari total 23 ulasan, dan 145 ulasan negatif dari total 172 ulasan dalam data uji.

Confusion matrix pada Gambar 5c menampilkan jumlah prediksi benar dan salah yang dihasilkan oleh model SVM terhadap *dataset* ulasan aplikasi Tokopedia. Model ini berhasil memprediksi 100 ulasan positif dari total 126 ulasan, tidak ada prediksi untuk ulasan netral dari total 27 ulasan, dan 232 ulasan negatif dari total 247 ulasan dalam data uji. Jumlah prediksi benar dan salah pada setiap kelas sentimen ini dapat diolah menjadi beberapa parameter evaluasi model. Parameter tersebut mencakup *accuracy*, *precision*, *recall*, dan F1-*score*, yang sering disebut sebagai *classification report*.

Tabel 4. *Classification Report* SVM

Ukuran	Kelas	Shopee	Blibli	Tokopedia
Accuracy (%)		90	91	83
Precision (%)	*Negative*	91	82	83
	Neutral	82	0	0
	Positive	94	94	84
Recall (%)	*Negative*	97	84	94
	Neutral	69	0	0
	Positive	86	97	79
F-1 Score (%)	*Negative*	94	83	88
	Neutral	75	0	0
	Positive	89	95	82



Gambar 6. 20 Kata Terbanyak di Aplikasi Shopee

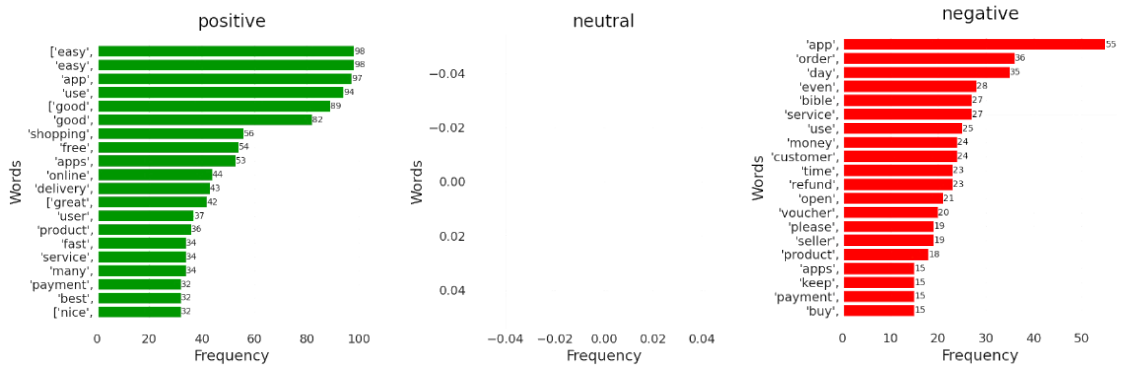
Tabel 4 menunjukkan tingkat akurasi model SVM terhadap tiga *dataset* ulasan aplikasi yang berbeda, yaitu 90% untuk Shopee, 91% untuk Blibli, dan 83% untuk Tokopedia. Hasil ini menunjukkan performa terbaik dibandingkan dengan algoritma *machine learning* lain yang dievaluasi. Model SVM yang dikembangkan mampu menghasilkan prediksi yang lebih akurat terhadap ulasan dengan sentimen positif dan negatif, seperti yang tercermin dari nilai *precision, recall*, dan F1-*score* yang lebih tinggi dibandingkan dengan sentimen netral. Perbedaan ini disebabkan karena jumlah ulasan yang berlabel netral yang jauh lebih sedikit dibandingkan dengan ulasan berlabel positif dan negatif dalam *dataset* yang digunakan, maka model tidak mendapatkan bahan latih yang cukup, khususnya untuk memprediksi sentimen netral.

Gambar 6 memperlihatkan diagram batang yang menggambarkan 20 kata yang paling sering muncul dalam ulasan aplikasi *e-commerce* Shopee, berdasarkan analisis sentimen menggunakan model *machine learning* SVM (SVM). Kata-kata ini dibagi ke dalam tiga kategori sentimen yang berbeda. Kata seperti 'app', 'good', 'shopping', 'product', dan 'hope' dalam kategori sentimen positif menunjukkan kepuasan pengguna terhadap aplikasi dan produk Shopee. 'Good' dan 'shopping' mengindikasikan pengalaman

berbelanja yang umumnya positif, sementara 'hope' dan 'please' menunjukkan harapan dan permintaan pengguna untuk peningkatan layanan di masa mendatang. Kata 'product' dan 'seller' mencerminkan perlunya perbaikan kualitas produk dan interaksi dengan penjual. Shopee harus memastikan bahwa produk sesuai dengan deskripsi dan bahwa penjual memberikan layanan yang memadai.

Pada kategori sentimen netral, kata 'app', 'time', dan 'slow' mendominasi, yang mungkin mencerminkan ketidakpuasan terkait kecepatan aplikasi tanpa menjadi keluhan serius. Kata 'please' dan 'still' menunjukkan bahwa pengguna sering meminta perbaikan tetapi belum melihat perubahan yang diharapkan. Shopee perlu lebih proaktif dalam berkomunikasi dengan pengguna dan memberikan pembaruan berkala tentang perbaikan yang dilakukan. Kata 'please', 'application', dan 'delivery' juga menunjukkan ekspektasi tertentu pengguna terhadap kinerja aplikasi dan proses pengiriman. Keluhan terkait 'time' dan 'delivery' menunjukkan bahwa proses ini perlu dioptimalkan. Shopee dapat meningkatkan logistik dan menyediakan informasi pengiriman yang lebih akurat.

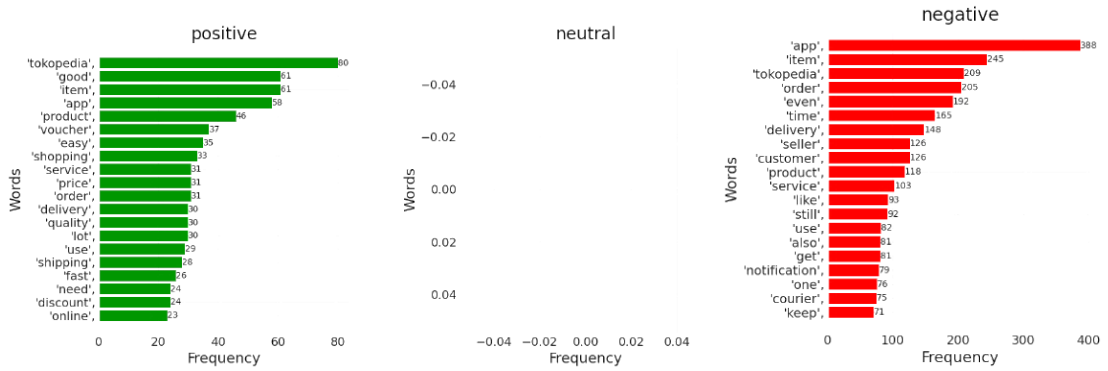
Pada kategori sentimen negatif, kata 'app', 'time', 'even', dan 'slow' menunjukkan keluhan pengguna tentang kinerja aplikasi yang lambat dan waktu yang diperlukan untuk menyelesaikan transaksi. Dominasi kata 'app' dalam sentimen negatif menekankan pentingnya peningkatan kecepatan dan stabilitas aplikasi. Pengembang perlu mengurangi waktu loading dan meningkatkan responsivitas aplikasi. Keluhan lain seperti 'problem', 'phone', dan 'many' mencerminkan berbagai masalah teknis dan kekecewaan umum yang dialami pengguna. Kata 'slow' dan 'problem' menunjukkan adanya masalah teknis yang perlu diselesaikan. Pengembang harus mengidentifikasi dan memperbaiki bug yang menyebabkan aplikasi lambat atau tidak responsif. Temuan ini memiliki beberapa implikasi untuk pengembangan dan pemeliharaan aplikasi Shopee, termasuk optimalisasi kinerja aplikasi, peningkatan proses pemesanan dan pengiriman, manajemen produk dan layanan, perbaikan masalah teknis, dan komunikasi dengan pengguna.



Gambar 7. 20 Kata Terbanyak di Aplikasi Blibli

Gambar 7 menampilkan hasil analisis sentimen dari ulasan aplikasi *e-commerce* Blibli menggunakan model klasifikasi *machine learning* SVM dalam bentuk diagram batang yang mengelompokkan kata-kata ke dalam tiga kategori sentimen. Pada kategori sentimen positif, kata-kata seperti 'easy', 'app', 'use', 'good', 'shopping', 'free', 'apps', '*online*', 'delivery', dan 'great' mendominasi. Frekuensi kemunculan kata 'easy' dan 'app' yang tinggi menunjukkan bahwa pengguna menganggap aplikasi ini mudah digunakan dan memberikan pengalaman yang memuaskan. Kata 'shopping', 'free', dan 'apps' menegaskan bahwa pengguna menikmati pengalaman berbelanja, terutama karena adanya fitur gratis yang ditawarkan. Kata 'delivery' juga menonjol, mengindikasikan kepuasan pengguna terhadap proses pengiriman. Kata-kata positif seperti 'easy', 'good', dan 'free' menegaskan bahwa pengguna menghargai kemudahan penggunaan dan fitur gratis. Pengembang perlu terus mengembangkan fitur-fitur yang memudahkan pengguna dan memberikan nilai tambah tanpa biaya tambahan. Kata 'order' dan 'delivery' menunjukkan bahwa ada ruang untuk meningkatkan efisiensi proses pemesanan dan pengiriman. Pengembang bisa mengoptimalkan sistem logistik dan menyediakan informasi pengiriman yang lebih akurat untuk mengurangi keluhan pengguna.

Pada kategori sentimen netral, tidak ada kata yang ditampilkan, yang mungkin disebabkan oleh model yang tidak menghasilkan prediksi ulasan bersentimen netral. Dalam kategori sentimen negatif, kata-kata seperti 'app', 'order', 'day', 'even', 'bible', 'service', 'use', 'money', 'customer', dan 'time' muncul. Kata 'app' dan 'order' sering disebut dalam konteks negatif, menunjukkan ketidakpuasan pengguna terhadap fungsi aplikasi dan proses pemesanan. Kata 'day' dan 'time' mungkin terkait dengan waktu yang diperlukan untuk menyelesaikan transaksi atau pengiriman, yang dianggap terlalu lama oleh pengguna. Keluhan utama seperti 'app' dan 'time' dalam sentimen negatif menunjukkan ketidakpuasan pengguna terhadap kecepatan dan kinerja aplikasi. Pengembang perlu fokus pada peningkatan kecepatan *loading*, responsivitas, dan stabilitas aplikasi untuk meningkatkan pengalaman pengguna. Selain itu, kata 'money' dan 'refund' menunjukkan adanya masalah dalam proses pengembalian uang atau transaksi keuangan lainnya. Pengembang perlu memastikan bahwa proses pembayaran dan pengembalian dana berjalan lancar dan transparan untuk meningkatkan kepercayaan pengguna. Berdasarkan temuan ini, implikasi untuk pengembangan dan pemeliharaan aplikasi Blibli adalah optimasi kinerja aplikasi, peningkatan proses pemesanan dan pengiriman, manajemen keuangan yang lebih baik, dan fokus pada pengalaman pengguna.



Gambar 8. 20 Kata Terbanyak di Aplikasi Tokopedia

Hasil analisis sentimen ulasan aplikasi *e-commerce* Tokopedia menggunakan model klasifikasi *machine learning* SVM ditampilkan pada Gambar 8 dalam bentuk diagram batang yang mengelompokkan kata-kata ke dalam tiga kategori sentimen. Pada kategori sentimen positif, kata-kata seperti 'tokopedia', 'good', 'item', 'app', 'product', 'voucher', 'easy', 'shopping', 'service', 'price', 'order', 'delivery', 'quality', 'lot', 'use', 'shipping', 'fast', 'need', 'discount', dan '*online*' mendominasi. Kata 'tokopedia' yang paling sering muncul mengindikasikan pandangan positif pengguna terhadap merek Tokopedia secara umum. Kata 'good', 'item', dan 'app' menunjukkan kepuasan pengguna terhadap aplikasi dan produk yang ditawarkan. Kata 'easy' dan 'shopping' menegaskan bahwa pengalaman berbelanja dianggap mudah dan menyenangkan oleh pengguna. Selain itu, kata 'voucher' dan 'discount' menunjukkan bahwa promosi dan penawaran spesial menarik bagi pengguna. Kata-kata positif seperti 'voucher' dan 'discount' menunjukkan bahwa promosi dan penawaran spesial sangat dihargai oleh pengguna. Pengembang harus terus menyediakan dan memperbarui penawaran-penawaran ini untuk menarik lebih banyak pengguna.

Pada kategori sentimen netral, tidak ada kata yang ditampilkan, mungkin karena model tidak menghasilkan prediksi ulasan bersentimen netral. Pada kategori sentimen negatif, kata-kata seperti 'app', 'item', 'tokopedia', 'order', 'even', 'time', 'delivery', 'seller', 'customer', 'product', 'service', 'like', 'still', 'use', 'also', 'get', 'notification', 'one', 'courier', dan 'keep' muncul. Kata 'app' dan 'item' sering disebut dalam konteks negatif, menunjukkan ketidakpuasan pengguna terhadap fungsi aplikasi dan produk yang dibeli. Kata 'order' dan 'time' yang sering muncul menunjukkan adanya keluhan terkait proses pemesanan dan waktu yang diperlukan untuk menyelesaikan transaksi. Kata 'delivery' dan 'courier' menunjukkan ketidakpuasan terhadap proses pengiriman. Keluhan utama seperti 'app' dan 'time' dalam sentimen negatif menunjukkan ketidakpuasan pengguna terhadap kecepatan dan kinerja aplikasi. Pengembang perlu fokus pada peningkatan kecepatan loading, responsivitas, dan stabilitas aplikasi

untuk meningkatkan pengalaman pengguna. Kata 'order' dan 'delivery' menunjuk-kan adanya ruang untuk meningkatkan efisiensi proses pemesanan dan pengiriman. Pengembang bisa mengoptimalkan sistem logistik dan mem-berikan informasi yang lebih akurat terkait pengiriman untuk mengurangi keluhan pengguna. Selain itu, kata 'item' dan 'product' menunjukkan adanya masalah dengan kualitas atau kesesuaian produk yang dijual. Pengembang perlu memastikan bahwa produk yang dijual sesuai dengan deskripsi dan standar kualitas yang dijanjikan. Selain itu, kata 'service' menunjukkan bahwa layanan pelanggan perlu ditingkatkan untuk menangani keluhan dan masalah pengguna secara efektif. Berdasarkan temuan ini, implikasinya untuk pengembangan dan pemeliharaan aplikasi Tokopedia adalah optimasi kinerja aplikasi, peningkatan proses pemesanan dan pengiriman, manajemen produk dan layanan, serta fokus pada promosi dan penawaran spesial.

KESIMPULAN DAN SARAN

Pada penelitian ini ulasan pengguna aplikasi Shopee, Blibli, dan Tokopedia pada Google Plays Store dianalisis menggunakan teknik *machine learning* serta pendekatan SEMMA untuk mengembangkan model klasifikasi yang akurat. Hasil evaluasi mengindikasikan bahwa algoritma klasifikasi yang diusulkan dalam studi ini yaitu SVM, menunjukkan nilai akurasi tertinggi secara konsisten pada *dataset* dari ketiga aplikasi tersebut. Secara spesifik nilai akurasi yang dicapai pada *dataset* Shopee adalah 90% dengan pembagian data latih dan data uji (8:2), pada *dataset* Blibli adalah 91% (6:4), dan pada *dataset* Tokopedia adalah 83% (8:2). Pencapaian ini diperoleh melalui kombinasi *hyperparameter* kernel linear, rbf, poly, dan sigmoid serta tingkat penalti terhadap kesalahan klasifikasi (*c*) dengan nilai 0.1, 1, dan 10. Temuan ini membuktikan bahwa algoritma SVM menunjukkan performa terbaik dibandingkan dengan algoritma *machine learning* lainnya seperti *Multinomial Naïve Bayes, Random Forest*, dan *Extra Trees Classifier*.

Berdasarkan hasil pelabelan meng-gunakan pendekatan *score based labeling*, ditemukan bahwa ulasan negatif merupakan sentimen yang paling dominan pada aplikasi Shopee dan Tokopedia, sedangkan pada aplikasi Blibli, sentimen positif adalah yang terbanyak. Kemudian hasil dari analisis mayoritas kata dalam setiap kelas sentimen, dihasilkan rekomendasi untuk pengembangan dan pemeliharaan ketiga aplikasi. Untuk aplikasi Shopee, diperlukan optimasi kinerja aplikasi, peningkatan proses *delivery* dan komunikasi dengan pengguna, serta perbaikan manajemen produk dan pemecahan masalah teknis. Pada aplikasi Blibli, diperlukan optimasi kinerja aplikasi, peningkatan proses *delivery* dan fitur peningkat kepuasan pengguna, serta perbaikan manajemen keuangan. Sementara itu, untuk aplikasi Tokopedia, diperlukan optimasi kinerja

aplikasi, peningkatan proses *delivery*, perbaikan manajemen produk, disertai dengan pembaharuan promosi atau penawaran spesial.

Penelitian selanjutnya dapat menggabungkan teknik *ensemble learning* atau *deep learning* seperti *Transformer-based models* (misalnya BERT) untuk meningkatkan akurasi klasifikasi sentimen, terutama pada ulasan yang bersifat ambigu atau netral. Selain itu, penelitian selanjutnya dapat memperluas *dataset* dengan mencakup *platform e-commerce* lain atau sumber ulasan tambahan seperti media sosial untuk memperoleh representasi sentimen yang lebih komprehensif. Penggunaan teknik *feature engineering* lain, seperti *word embeddings* atau *topic modeling*, juga dapat dieksplorasi untuk memahami konteks ulasan secara lebih mendalam. Penelitian selanjutnya dapat juga melakukan analisis temporal untuk mengetahui perubahan tren sentimen pengguna seiring waktu, sehingga dapat memberikan rekomendasi yang lebih dinamis bagi pengembang aplikasi *e-commerce*.

DAFTAR PUSTAKA

[1] B. Z. Ramadhan, R. I. Adam, and I. Maulana, "Analisis sentimen ulasan pada aplikasi e-commerce dengan menggunakan algoritma Naïve Bayes," *Journal of Applied Informatics and Computing*, vol. 6, no. 2, pp. 220-225, 2022, doi: 10.30871/jaic.v6i2.4725.

[2] I. A. Novianti, I. Purwanti, and V. Y. Pratama, "Dampak jual beli online terhadap pasar tradisional (studi kasus Pasar Kedungwuni)," *Sahmiyya: Jurnal Ekonomi dan Bisnis*, vol. 3, no. 1, pp. 131-141, 2024. [Online]. Available: https://e-journal.uingusdur.ac.id/sahmiyya/article/view/1846.

[3] S. Mulyana, J. Wulandari, and F. Liani, "Pengaruh pasar modern terhadap keberlangsungan pasar tradisional di Indonesia," *Jurnal Kolaboratif Sains*, vol. 7, no. 12, pp. 4689-4695, 2024, doi: 10.56338/jks.v7i12.6623.

[4] Musfiroh, A. Tholib, and Z. Abidin, "Analisis sentimen terhadap ulasan aplikasi Shopee di Google Play Store menggunakan metode TF-IDF dan long short-term memory," *Journal of Electrical Engineering and Computer (JEECOM)*, vol. 6, no. 2, pp. 371-381, 2024, doi: 10.33650/jeecom.v6i2.8713.

[5] Tukino, D. Abdullah, M. M. Amalia, Y. N. Supriadi, and T. Winarko, *Strategi Bisnis E-Commerce*. Medan: Yayasan Kita Menulis, May 2023, ISBN: 978-623-342-856-9.

[6] B. Harto, A. Y. Rukmana, R. Subekti, R. Tahir, E. Waty, A. C. Situru, and Sepriano, *Transformasi Bisnis di Era Digital*. Jambi: Sonpedia Publishing Indonesia, Aug. 2023, ISBN: 978-623-8345-30-4.

[7] A. Binoy, F. Safna, and M. David, “Factors influencing online shopping behaviour: An empirical study in Bangalore,” *International Journal of Statistics and Applied Mathematics*, vol. 1, no. 6, pp. 21-26, 2016.

[8] Y. Liu and S. Zheng, “Factors affecting consumers purchase intention for agriculture products omni-channel,” *Frontiers in Psychology*, vol. 13, Jan. 2023, doi: 10.3389/fpsyg.2022.948982.

[9] S. G. Zaato, N. R. Zainol, S. Khan, A. U. Rehman, M. R. Faridi, and A. A. Khan, “The mediating role of customer satisfaction between antecedent factors and brand loyalty for the Shopee application,” *Behavioral Sciences*, vol. 13, no. 7, p. 563, Jul. 2023, doi: 10.3390/bs13070563.

[10] R. F. Abdillah and A. N. Pramesti, “Dampak rating dan ulasan konsumen terhadap keputusan pembelian di e-commerce,” *Prosiding Seminar Nasional AMIKOM Surakarta (SEMNASA)*, Sukoharjo, pp. 1480-1494, Nov. 23, 2024.

[11] D. Purnamasari, A. B. Aji, D. W. A. Putri, F. A. Reza, M. S. Oktiana, N. Yanda, and U. Hidayati, *Pengantar metode analisis sentimen*. Jakarta: Gunadarma, 2023.

[12] F. K. Wardani, V. R. Hananto, and V. Nurcahyawati, “Analisis sentimen untuk pemeringkatan popularitas situs belanja online di Indonesia menggunakan metode Naive Bayes (studi kasus data sekunder),” *Jurnal Sistem Informasi Universitas Dinamika (JSIKA)*, vol. 8, no. 1, 2019.

[13] M. R. Rahman, A. F. Diansyah, and Hanafi, “Sentiment analysis on the Shopee application on Playstore using the Random Forest classification method,” *Inform: Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 9, no. 1, pp. 20-24, 2024. doi: 10.25139/inform.v9i1.5465.

[14] F. Hadaina and U. Budiyanto, “Implementasi metode Multinomial Naïve Bayes untuk sentiment analysis terhadap data ulasan produk Colearn pada Google Play Store,” *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)*, vol. 1, no. 1, pp. 660-666, Sep. 2022.

[15] L. N. Aina, V. R. S. Nastiti, and C. S. K. Aditya, "Implementasi Extra Trees Classifier dengan optimasi Grid Search CV pada prediksi tingkat adaptasi," *MIND (Multimedia Artificial Intelligent Networking Database) Journal*, vol. 9, no. 1, pp. 78-88, Jun. 2024, doi: 10.26760/mindjournal.v9i1.78-88.

[16] M. I. Ahmadi, F. Apriani, M. Kurniasari, S. Handayani, and D. Gustian, “Sentiment analysis online shop on the Play Store using method Support Vector Machine (SVM),”

Prosiding Seminar Nasional Informatika (SEMNASIF), vol. 1, no. 1, pp. 196-203, Dec. 2020.

[17] P. A. Adedeji, J. A. Oyewale, T. I. Ogedengbe, O. O. Olatunji, and N. Madushele, "Comparative assessment of soft computing and SVM architectures for multi-class automobile engine fault classification," *International Journal of System Assurance Engineering and Management* 2025, doi: 10.1007/s13198-025-02747-y.

[18] S. A. H. Bahtiar, C. K. Dewa, and A. Luthfi, "Comparison of Naïve Bayes and Logistic Regression in sentiment analysis on marketplace reviews using rating-based labeling," *Journal of Information Systems and Informatics*, vol. 5, no. 3, pp. 915-927, 2020. doi: 10.51519/journalisi.v5i3.539.

[19] F. D. Adhiatma and A. Qoiriah, "Penerapan metode TF-IDF dan deep neural network untuk analisa sentimen pada data ulasan hotel," *Journal of Informatics and Computer Science (JINACS)*, vol. 4, no. 2, pp. 183-193, 2022, doi: 10.26740/jinacs.v4n02.p183-193.