

ANALISIS SENTIMEN ULASAN APLIKASI M-PASPOR MENGUNAKAN *NAIVE BAYES* DAN *SUPPORT VECTOR MACHINE*

Rahman Maheri*¹⁾, Febi Nur Salisah²⁾, Fitriani Muttakin³⁾, Megawati⁴⁾

1. Universitas Islam Negeri Sultan Syarif Kasim Riau, Indonesia
2. Universitas Islam Negeri Sultan Syarif Kasim Riau, Indonesia
3. Universitas Islam Negeri Sultan Syarif Kasim Riau, Indonesia
4. Universitas Islam Negeri Sultan Syarif Kasim Riau, Indonesia

Article Info

Kata Kunci: Analisis Sentimen; M-Paspor;
Naive Bayes; *Support Vector Machine*

Keywords: M-Paspor; *Naive Bayes*; *Sentiment
Analysis*; *Support Vector Machine*

Article history:

Received 21 October 2024
Revised 12 November 2024
Accepted 4 December 2025
Available online 1 March 2025

DOI :

<https://doi.org/10.29100/jupi.v10i1.5826>

* Corresponding author.

Corresponding Author

E-mail address:

12050317617@students.uin-suska.ac.id

ABSTRAK

M-Paspor merupakan sebuah aplikasi pelayanan publik yang dirilis pada awal tahun 2022 lalu. Telah lebih dari satu juta unduhan aplikasi ini di *Google Play Store*. Dalam beberapa kasus, konten ulasan dan ulasan produk atau layanan dapat berbeda. Analisis umpan balik pengguna harus menjadi prioritas utama saat mengembangkan aplikasi. Penelitian ini dilakukan dengan tujuan untuk memahami persepsi pengguna aplikasi M-Paspor di *Google Play Store*, dengan cara mengidentifikasi sentimen positif dan negatif dalam ulasan mereka, serta Membandingkan kemampuan algoritma *Support Vector Machine* dan *Naive Bayes* dalam menghasilkan klasifikasi yang akurat dalam menganalisis sentimen tersebut. Metode penelitian melibatkan pengumpulan, preprocessing, pelabelan, dan klasifikasi 5.934 ulasan aplikasi M-Paspor di *Google Play Store*, dan penerapan algoritma *Support Vector Machine* dan *Naive Bayes* untuk mengklasifikasikan sentimen ulasan. Setelah menerapkan algoritma, tahap terakhir adalah pengujian evaluasi. Hasil penerapan algoritma klasifikasi, menunjukkan bahwa algoritma *Support Vector Machine* dapat mengklasifikasikan ulasan aplikasi M-Paspor dengan lebih akurat, dengan rata-rata 80,76% dibandingkan dengan 78,12% dari *Naive Bayes*. Temuan ini menunjukkan potensi *Support Vector Machine* dalam membantu pengembang M-Paspor meningkatkan kualitas layanan berdasarkan umpan balik pengguna.

ABSTRACT

M-Paspor is a public service application that was released in early 2022. There have been more than one million downloads of this application on the Google Play Store. In some cases, review content and product or service reviews may differ. Analyzing user feedback should be a top priority when developing apps. This research was conducted with the aim of understanding the perceptions of users of the M-Paspor application on the Google Play Store, by identifying positive and negative sentiments in their reviews, and comparing the ability of the Support Vector Machine and Naive Bayes algorithms to produce accurate classifications in analyzing these sentiments. The research method involves collecting, preprocessing, labeling, and classifying 5,934 reviews of the M-Paspor app on the Google Play Store, and applying the Support Vector Machine and Naive Bayes algorithms to classify the sentiment of the reviews. After applying the algorithm, the last stage is evaluation testing. The results of applying the classification algorithms, show that the Support Vector Machine algorithm can classify M-Paspor app reviews more accurately, with an average of 80.76% compared to 78.12% from Naive Bayes. These findings demonstrate the potential of Support Vector Machine in helping M-Paspor developers improve service quality based on user feedback.

I. PENDAHULUAN

Direktorat Jenderal Imigrasi merupakan bagian dari struktur pemerintahan yang bertanggung jawab dalam memberikan layanan publik di bidang keimigrasian. Pelayanan ini mencakup Warga Negara Indonesia serta Warga Negara Asing. Dalam upaya Memperkuat komitmen untuk meningkatkan kualitas pelayanan keimigrasian demi memberikan pengalaman terbaik bagi masyarakat., Direktorat Jenderal Imigrasi terus mengembangkan dan meningkatkan sistem keimigrasian yang ada. Salah satunya adalah memperkenalkan sebuah aplikasi untuk memudahkan masyarakat dalam mengajukan permohonan paspor. Inovasi terbaru dari Direktorat Jenderal Imigrasi itu adalah Aplikasi M-Paspor, dirancang untuk memudahkan dalam pembuatan paspor. Melalui aplikasi ini, masyarakat dapat mendaftar untuk permohonan paspor baru dan juga untuk permohonan penggantian paspor. M-Paspor tersedia pada platform Pengguna smartphone dapat mengakses dan mengunduh sistem operasi Android pada platform *Google Play Store* dan iOS pada platform *App Store* [1].

Tepat pada 26 Januari 2022, aplikasi M-paspor yang merupakan bagian dari proses permohonan paspor, secara resmi dirilis oleh Direktorat Jenderal Imigrasi. Sejak perilisan pertama kali di *Google Play Store*, Satu juta lebih pengguna smartphone Android telah mengunduh aplikasi M-Paspor. Setiap aplikasi tidak luput dari kelebihan dan kekurangan, yang berdampak pada reaksi pengguna, termasuk tingkat kepuasan dan ketidakpuasan yang ditunjukkan dalam ulasan yang diposting di *Google Play Store* [2].

Mengajukan permohonan dan penggantian paspor baru dapat dilakukan masyarakat menggunakan aplikasi M-Paspor [3]. Dengan memasukkan data pribadi dan mengunggah dokumen yang diperlukan secara online. M-Paspor membuat proses pengajuan paspor lebih mudah. Pertama, unduh dan instal aplikasi di smartphone atau tablet. Daftarkan akun pengguna baru atau login pada akun yang telah ada, lalu ajukan permohonan paspor dengan mengisi formulir yang tersedia. Pilih lokasi kantor imigrasi terdekat dan jadwal kedatangan, lalu lakukan pembayaran sesuai instruksi. Jika perlu, ubah jadwal kedatangan. Pada hari yang ditentukan, datang ke kantor imigrasi untuk wawancara dan verifikasi dokumen.

Google Play Store adalah layanan Android dalam bentuk perangkat lunak yang menyediakan tempat bagi pengguna untuk mengunduh aplikasi, game, dan lainnya untuk sistem operasi Android. Pengguna dapat memberikan pendapat atau tanggapan terhadap kinerja aplikasi yang digunakan melalui rating dan ulasan yang dapat diunduh pada *Google Play Store*. Nilai rating dan ulasan sangat berpengaruh terhadap aplikasi di *Google Play Store*. Sebuah penelitian lain menunjukkan bahwa ulasan pengguna adalah komponen penting dari pasar aplikasi seluler, dan analitik ulasan pengguna dapat memberikan wawasan tentang penggunaan dan dampak aplikasi tersebut [4]. Pada *Google Play Store*, pengguna memiliki kemampuan untuk memberi penilaian dan memberikan ulasan terhadap aplikasi. Namun, sering kali terdapat perbedaan antara penilaian yang diberikan dan konten ulasan, sehingga tidak selalu dapat mencerminkan secara tepat kualitas aplikasi. Dengan demikian, analisis sentimen bisa berperan sebagai pendekatan yang tepat dalam menentukan arah opini pengguna serta bagaimana kualitas dari aplikasi M-Paspor yang ada tersedia untuk perangkat Android.

Analisis sentimen adalah proses pengumpulan, pengolahan, dan analisis data untuk mengidentifikasi opini atau sentimen yang dimuat dalam teks, seperti ulasan, postingan media sosial, ataupun dari tweet [5]. Analisis sentimen bertujuan untuk menangkap pandangan dan emosi yang dimiliki orang terhadap suatu topik atau produk [6]. Analisis sentimen memiliki penerapan luas, termasuk dalam bidang bisnis, pemasaran, dan politik [7]. Analisis sentimen dapat dilakukan menggunakan beberapa metode seperti *Random Forest*, *Support Vector Machine* (SVM), dan *Naive Bayes*.

Naive Bayes dan *Support Vector Machine* (SVM) adalah dua metode yang dapat digunakan untuk melakukan klasifikasi atau prediksi pada data. Algoritma *Naive Bayes* adalah metode *machine learning* untuk menghitung probabilitas. Dengan kata lain, *Naive Bayes* mampu mengklasifikasikan teks dengan cepat, efisien, dan efektif, bahkan ketika digunakan untuk menganalisis data berskala besar [8]. Sedangkan *Support Vector Machine* adalah metode yang difungsikan untuk klasifikasi dengan menemukan *hyperplane* terbaik yang memilah dua kelas data. *Support Vector Machine* bisa dimanfaatkan untuk melakukan klasifikasi biner maupun multikelas [9]. Penelitian ini memilih algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) untuk analisis sentimen karena keduanya memiliki rekam jejak yang kuat dalam kinerja yang efektif dan efisien dalam tugas klasifikasi teks. *Naive Bayes* dikenal karena kesederhanaan dan kecepatan komputasinya, serta efektivitasnya dalam situasi di mana fitur-fitur bersifat independen. Sementara itu, SVM terkenal dengan kemampuannya untuk menghasilkan margin klasifikasi yang maksimal antara kelas-kelas yang berbeda, serta kinerjanya yang baik dalam menangani data berdimensi tinggi.

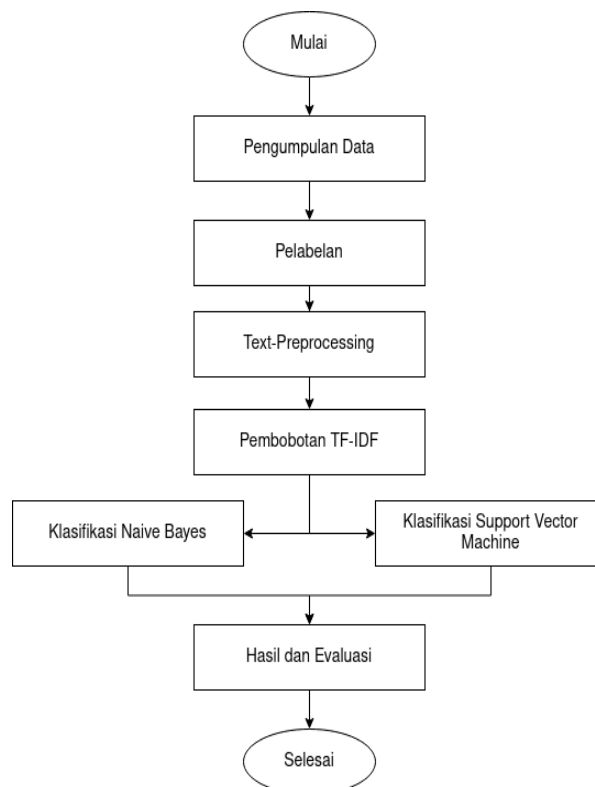
Penelitian sebelumnya [10] mengenai analisis sentimen dengan menggunakan algoritma *Random Forest* dan

Support Vector Machine, dengan judul “*Sentiment Analysis of Shopee App Reviews Using Random Forest and Support Vector Machine*”, didapatkan bahwa klasifikasi menggunakan algoritma *Support Vector Machine* lebih baik dengan akurasi sebesar 84.71%, sedangkan hasil nilai akurasi dari metode *Random Forest* didapatkan sebesar 82.21%. Oleh karena itu, model klasifikasi analisis sentimen untuk ulasan aplikasi Shopee menggunakan metode *Support Vector Machine* dapat dimanfaatkan oleh perusahaan marketplace (Shopee) untuk mengetahui apa yang dipikirkan masyarakat terhadap aplikasi Shopee di *Google Play Store*. Penelitian lainnya [11] mengenai analisis sentimen dengan judul “*Komparasi Metode KNN dan Naive Bayes Terhadap Analisis Sentimen Pengguna Aplikasi Shopee*” mendapat temuan bahwa Klasifikasi *Naive Bayes* memiliki skor akurasi 71,00% yang lebih unggul dibanding *k-Nearest Neighbors*, sementara *k-Nearest Neighbors* memiliki skor akurasi 70,00%.

Oleh karena itu, akan diuji klasifikasi menggunakan metode *Support Vector Machine* dan *Naive Bayes* untuk menemukan algoritma mana yang mempunyai tingkat keakuratan yang lebih baik dengan dataset ulasan aplikasi M-Paspor. Penelitian ini juga menangani ketidakseimbangan data ulasan dengan menggunakan teknik *Random Sampling*, memberikan wawasan baru dalam mengelola data tidak seimbang. Temuan penelitian ini dapat membantu pengembang aplikasi M-Paspor dalam memahami persepsi pengguna dan mengidentifikasi area yang perlu ditingkatkan. Hasil penelitian ini juga dapat menjadi dasar bagi pengembangan lebih lanjut untuk analisis sentimen pada aplikasi layanan publik lainnya.

II. METODE PENELITIAN

Alur dan skema penelitian ditunjukkan pada Gambar 1. Meliputi langkah-langkah atau rute yang diambil pada penelitian ini.



Gambar 1. Alur Skema Penelitian

A. Pengumpulan Data

Data yang diambil adalah data 1 tahun yang dimulai dari 14 Maret 2023 hingga 13 Maret 2024, periode ini cukup untuk mengumpulkan data yang cukup dari berbagai pengguna dengan beragam pengalaman dan melihat bagaimana sentimen pengguna terhadap aplikasi M-Paspor, memungkinkan kita untuk mendapatkan gambaran menyeluruh tentang sentimen pengguna terhadap aplikasi paspor. 5934 data didapatkan melalui teknik scrapping data menggunakan *library* bahasa pemrograman python *google-play-scrapper* di *Google colab*. Data ulasan pengguna tentang aplikasi M-Paspor kemudian dikumpulkan dan disimpan dalam file CSV. Ulasan pengguna pada *Google Play Store* dapat menunjukkan kinerja aplikasi.

B. Pelabelan Data

Proses penetapan kelas pada data yang dihasilkan melalui proses *scrapping* [12]. Komentar tersebut akan dibagi menjadi kelas positif, netral dan negatif pada teks. Pelabelan dilakukan secara otomatis menggunakan script pada pemrograman python, memanfaatkan kamus *Lexicon* bahasa indonesia, kamus ini berisi lebih dari 13.000 kata dan frasa yang diklasifikasikan ke dalam kategori sentimen positif, negatif, dan netral, yang mana memiliki bobot sentimen yang berbeda-beda. Bobot ini mencerminkan kekuatan sentimen yang terkandung dalam kata atau frasa tersebut. Namun, perlu diingat bahwa kamus *Lexicon* tidak sempurna dan mungkin terdapat kata atau frasa yang belum terdaftar atau diklasifikasikan secara tidak tepat.

C. Text Preprocessing

Tahapan *preprocessing* data merupakan proses pengolahan data yang sudah didapat untuk meningkatkan kualitas data agar data yang dimiliki lebih terstruktur dan bersih sehingga menambah keakuratan data yang akan digunakan selama penelitian [13]. Tahapan ini meliputi: *cleaning*, *filtering*, *tokenizing* [14], dan *stemming*.

D. Pembobotan TF-IDF

Perhitungan untuk menghasilkan bobot pada setiap kata adalah langkah awal dalam metode klasifikasi, metode TF-IDF bisa diimplementasikan untuk pembobotan fitur tiap-tiap kata [15]. Pertama, perhitungan frekuensi kata (TF) dilakukan, yang mengukur frekuensi kata yang muncul dalam suatu dokumen. Selanjutnya, Bobot signifikansi kata dihitung menggunakan metode frekuensi dokumen invers (IDF). Metode ini mempertimbangkan seberapa sering kata muncul di seluruh dokumen untuk menentukan relevansinya. Kata yang jarang muncul dianggap lebih penting karena menunjukkan kekhususan topik. Dengan menyatukan kalkulasi TF dan IDF, diperoleh bobot, yang menunjukkan seberapa penting setiap kata dalam konteks dokumen dan secara keseluruhan. Contoh perhitungan TF-IDF.

TABEL I
 CONTOH ULASAN YANG AKAN DIHITUNG TF-IDF

No.	Ulasan
1	tanggal,klik,gimana,jadwal,mudah,sulit
2	tolong,baik,aplikasi,mudah,pakai

Rumus menghitung TF:

$$TF(t, d) = \frac{\text{jumlah kemunculan kata } t \text{ dalam dokumen } d}{\text{total jumlah kata dalam dokumen } d}$$

Rumus menghitung IDF:

$$IDF(t) = \log \left(\frac{\text{total jumlah dokumen dalam korpus}}{\text{jumlah dokumen dalam korpus yang mengandung kata } t} \right)$$

Rumus menghitung TF-IDF:

$$TF - IDF(t, d) = TF(t, d) \times IDF(t)$$

Keterangan:

- t adalah kata tertentu.
- d adalah dokumen tertentu

TABEL II
 PERHITUNGAN TF-IDF

Kata	Kalimat 1 (TF)	Kalimat 2 (TF)	IDF	Kalimat 1 (TF-IDF)	Kalimat 2 (TF-IDF)
tanggal	1/6 ≈ 0.1667	0	log(2) ≈ 0.3010	0.0502	0
klik	1/6 ≈ 0.1667	0	log(2) ≈ 0.3010	0.0502	0
gimana	1/6 ≈ 0.1667	0	log(2) ≈ 0.3010	0.0502	0
jadwal	1/6 ≈ 0.1667	0	log(2) ≈ 0.3010	0.0502	0

mudah	$1/6 \approx 0.1667$	$1/5 = 0.2$	$\log(2) \approx 0.3010$	0	0.0602
sulit	$1/6 \approx 0.1667$	0	$\log(2) \approx 0.3010$	0.0502	0
tolong	0	$1/5 = 0.2$	$\log(2) \approx 0.3010$	0	0.0602
baik	0	$1/5 = 0.2$	$\log(2) \approx 0.3010$	0	0.0602
aplikasi	0	$1/5 = 0.2$	$\log(2) \approx 0.3010$	0	0.0602
pakai	0	$1/5 = 0.2$	$\log(2) \approx 0.3010$	0	0.0602

E. K-Fold Cross-Validation

K-fold cross-validation adalah metode evaluasi model yang membagi dataset menjadi k subset untuk digunakan sebagai data latih dan uji secara bergantian. Setiap subset berfungsi sebagai data uji satu kali dan sebagai data latih sebanyak k-1 kali. Proses ini diulang hingga semua subset telah digunakan sebagai data uji, yang menghasilkan estimasi kinerja model yang lebih stabil dan akurat [16]. Penggunaan *k-fold cross-validation* dapat mengurangi bias dan varians dalam evaluasi model serta mengoptimalkan pemanfaatan data. Pada penelitian ini k-10 kali yang digunakan.

F. Klasifikasi Naive Bayes

Naive Bayes adalah algoritma klasifikasi dalam *machine learning* yang didasarkan pada teorema Bayes. Algoritma ini mengasumsikan bahwa fitur-fitur dalam dataset bersifat independen satu sama lain. Dengan menggunakan teorema Bayes, *Naive Bayes* menghitung probabilitas bahwa suatu data termasuk dalam kelas tertentu [17]. Meskipun asumsi independensi seringkali tidak sepenuhnya benar, algoritma ini tetap efektif dan efisien untuk banyak tugas klasifikasi, seperti teks dan analisis sentimen. Keunggulan utama *Naive Bayes* adalah kesederhanaan dan kecepatan dalam pelatihan serta prediksi, meskipun memiliki keterbatasan dalam menghadapi fitur yang saling bergantung [18].

G. Klasifikasi Support Vector Machine

Support Vector Machine (SVM) digunakan dalam analisis sentimen untuk memisahkan teks menjadi kategori sentimen [19]. SVM dengan kernel linear dipilih karena efektif dalam memisahkan data dengan pola linier, seperti ulasan Play Store yang mungkin memiliki perbedaan sentimen yang jelas antara positif dan negatif. Kernel linear mencari hyperplane optimal untuk memaksimalkan margin antara kelas-kelas data, sehingga cocok untuk kasus data dengan pemisahan linier.

H. Hasil dan Evaluasi

Setelah semua hasil dari metode-metode sudah didapat, Langkah terakhir adalah melakukan pengukuran untuk mencari metode mana yang memiliki tingkat skurasi yang paling besar berdasarkan data yang telah diteliti kemudian melakukan visualisasi hasil pengukuran antar metode menggunakan *Confusion Matrix*.

III. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Data yang dikumpulkan untuk penelitian ini didapatkan pada ulasan M-Passpor di *Google Play Store*. Ini dilakukan memanfaatkan *library python google-play-scraper* di *Google Colaboratory*. Adapun hasil dari *scrapping* tersebut mengumpulkan sebanyak 5934 komentar terhitung dari tanggal 14 Maret 2023 sampai dengan tanggal 13 Maret 2024, dan data disimpan ke dalam file CSV demi memudahkan untuk penggunaan penelitian analisis sentimen.

B. Pelabelan Data

Data yang telah didapat, dilabelkan berdasarkan sentimennya, berupa sentimen negatif, netral dan sentimen positif dengan menggunakan kamus *lexicon* bahasa indonesia. Lalu data yang bersentimen netral akan difilter, karena kita hanya menggunakan sentimen positif dan negatif pada penelitian ini.

TABEL III
 CONTOH ULASAN YANG TELAH DIBERI LABEL

No.	Ulasan	Label
1	Gak jelas, tanggal tdk bisa di klik, gimana mau buat jadwalnya... bukannya mempermudah jd mempersulit	Negatif
2	Tolong perbaiki aplikasinya agar mudah dan bisa dipakai	Negatif
3	App cacat	Negatif
4	Ok Banget	Positif

C. Preprocessing Data

Tahap pra-pemrosesan data adalah langkah yang dilakukan untuk menyiapkan data mentah sebelum memasukkannya ke tahap-tahap pemrosesan selanjutnya [20]. Pada dasarnya, tahap pra-pemrosesan data mencakup menghapus data yang tidak penting atau mengubah format data sehingga sistem dapat memprosesnya dengan lebih mudah. Dalam konteks ini, pra-pemrosesan data dilakukan menggunakan bahasa pemrograman *Python* dan platform *Google Colaboratory*. Tahap pra-pemrosesan data dalam penelitian ini meliputi proses (*cleaning, filtering, dan stemming*).

1) Cleaning Data

Data yang diolah berupa teks akan dibersihkan terlebih dahulu untuk membersihkan kata *symbol, hashtag, username, url*, komentar yang sama dan mengkonversi setiap kata kapital pada teks menjadi huruf kecil (*lower-case*), setelah data dilakukan pembersihan maka data berkurang menjadi 4830 data.

TABEL IV
 CLEANING DATA

Ulasan	Clean Data
Gak jelas, tanggal tdk bisa di klik, gimana mau buat jadwalnya... bukannya mempermudah jd mempersulit	jelas tanggal bisa klik gimana buat jadwalnya bukannya mempermudah mempersulit
Tolong perbaiki aplikasinya agar mudah dan bisa dipakai	tolong perbaiki aplikasi agar mudah bisa dipakai
Kenapa pilih tanggal kedatangan, selalu tidak ada kuota tersedia, dan besok hari nya udah berubah jadi kuota penuh, ada apa ya.....???? Daerah saya MEDAN	kenapa pilih tanggal kedatangan selalu tidak kuota tersedia besok hari udah berubah jadi kuota penuh daerah saya medan

2) Filtering Data

Pada tahapan filtering akan dilakukan pengambilan kata-kata penting pada sebuah kalimat guna membuang kalimat atau kata-kata yang dianggap tidak penting seperti *stopword*. Proses ini juga melibatkan *tokenizing*, yaitu membagi kalimat menjadi token-token yang mewakili unit-unit makna seperti kata-kata.

TABEL V
 TOKENIZING DATA

Clean Data	Token Data
jelas tanggal bisa klik gimana buat jadwalnya bukannya mempermudah mempersulit	jelas,tanggal,bisa,klik,gimana,buat,jadwalnya,bukannya,mempermudah,mempersulit
tolong perbaiki aplikasi agar mudah bisa dipakai	tolong,perbaiki,aplikasi,agar,mudah,bisa,dipakai
kenapa pilih tanggal kedatangan selalu tidak kuota tersedia besok hari udah berubah jadi kuota penuh daerah saya medan	kenapa,pilih,tanggal,kedatangan,selalu,tidak,kuota,tersedia,besok,hari,udah,berubah,jadi,kuota,penuh,daerah,saya,medan

Setelah tokenizing selesai, langkah selanjutnya adalah menghilangkan setiap kata yang merupakan *stopwords* atau setiap umum yang tidak memiliki arti khusus dalam situasi tertentu. Penghapusan *stopwords* ini memudahkan analisis teks dengan berkonsentrasi pada kata-kata penting.

TABEL VI
 STOPWORDS DATA

Token Data	Stop Data
------------	-----------

jelas,tanggal,bisa,klik,gimana,buat,jadwal- nya,bukannya,mempermudah,mempersulit	tanggal,klik,gimana,jadwalnya,mempermudah,mem- persulit
tolong,perbaiki,aplikasi,agar,mudah,bisa,dipakai	tolong,perbaiki,aplikasi,mudah,dipakai
kenapa,pilih,tanggal,kedatangan,se- lalu,tidak,kuota,tersedia,besok,hari,udah,beru- bah,jadi,kuota,penuh,daerah,saya,medan	pilih,tanggal,kedatangan,kuota,tersedia,besok,beru- bah,kuota,penuh,daerah,medan

3) Stemming Data

Pengubahan kata yang memiliki imbuhan menjadi bentuk dasarnya. Proses ini melibatkan pencarian kata dasar untuk setiap kata dengan menggunakan modul Sastrawi dalam bahasa pemrograman *Python*.

TABEL VII
STEMMING DATA

Stop Data	Stem Data
tanggal,klik,gimana,jadwalnya,mempermudah,mem- persulit	tanggal,klik,gimana,jadwal,mudah,sulit
tolong,perbaiki,aplikasi,mudah,dipakai	tolong,baik,aplikasi,mudah,pakai
pilih,tanggal,kedatangan,kuota,tersedia,besok,beru- bah,kuota,penuh,daerah,medan	pilih,tanggal,datang,kuota,se- dia,besok,ubah,kuota,penuh,daerah,medan

D. Pembobotan TF-IDF

TF-IDF merupakan sebuah algoritma yang berguna dalam menentukan bobot relatif dari setiap kata yang sering muncul dalam sebuah dokumen atau kumpulan data. Metode TF-IDF membantu dalam menentukan seberapa penting setiap kata dalam dokumen atau dataset yang diselidiki dengan mempertimbangkan seberapa sering kata-kata tersebut muncul dalam dokumen dibandingkan dengan dataset secara keseluruhan.

E. Model Klasifikasi

1) Menyeimbangkan Dataset

Dalam penelitian ini, data yang awalnya memuat 416 ulasan positif dan 4030 ulasan negatif dikonversi menjadi data seimbang dengan rasio data 416:416 di mana penyeimbangan dilakukan menggunakan metode *Random Sampling* sehingga jumlah ulasan positif sama dengan ulasan negatif.

2) Uji Model Klasifikasi

Metode yang diusulkan dalam penelitian ini adalah dengan membandingkan kinerja metode klasifikasi *Naive Bayes* dan *Support Vector Machine*, skor akurasi sebagai indikator utama untuk membandingkan kinerja kedua model klasifikasi ini.

Pengujian klasifikasi *Naive Bayes* dan *Support Vector Machine* (SVM) dilakukan menggunakan teknik *k-fold cross validation* dengan nilai $k=10$. *K-fold cross validation* adalah metode evaluasi model yang membagi dataset menjadi k subset (*fold*). Pada setiap iterasi, satu subset digunakan sebagai data uji dan $k-1$ subset lainnya sebagai data latih. Proses ini diulang sebanyak k kali, sehingga setiap subset digunakan sebagai data uji sekali. Dengan demikian, setiap data mendapat kesempatan untuk berperan sebagai data uji, yang meningkatkan akurasi dan generalisasi model.

Untuk pengujian SVM, parameter yang digunakan adalah *kernel linear*. *Kernel linear* dipilih karena sifatnya yang sederhana dan efektif dalam memisahkan data yang bersifat *linier*. *Kernel* ini bekerja dengan mencari *hyper-plane* optimal yang memaksimalkan *margin* antara kelas-kelas dalam data latih.

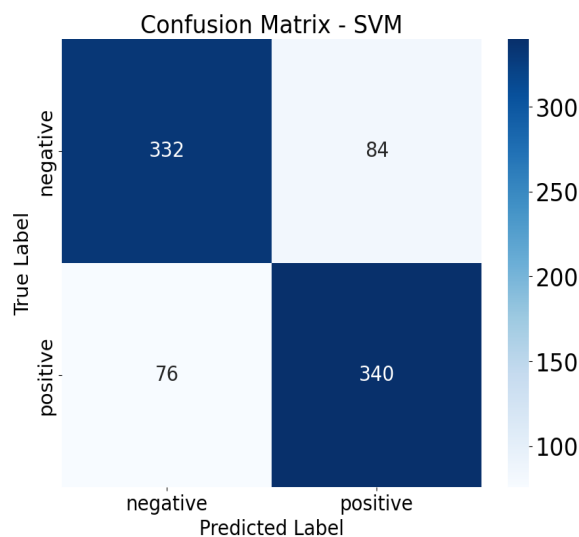
Hasil perbandingan ini menunjukkan bahwa model terbaik ditemukan menggunakan metode *Support Vector Machine*, dengan tingkat akurasi rata-rata sebesar 80,76%. Penelitian ini juga mencakup perbandingan nilai akurasi dari eksperimen *cross-validation* antara *Naive Bayes* dan *Support Vector Machine*, yang disajikan dalam Tabel VI sebagai indikator kualitatif yang membantu dalam mengevaluasi kinerja kedua metode klasifikasi tersebut.

TABEL VIII
HASIL UJI AKURASI MEMNGGUNAKAN 10 FOLD CROSS VALIDATION

Model	<i>Naive Bayes</i>	<i>Support Vector Machine</i>
1	0,7857	0,8214
2	0,7857	0,8571
3	0,7952	0,8675

4	0,7108	0,7831
5	0,7831	0,7711
6	0,7952	0,8313
7	0,7711	0,7590
8	0,7952	0,8434
9	0,8193	0,8193
10	0,7711	0,7229
Mean	0,7812	0,8076

Dari Tabel VI, diketahui bahwa nilai akurasi model dengan *10-level-cross-validation*. Berdasarkan rata-rata akurasi untuk setiap model, diperoleh bahwa model *Naive Bayes* memiliki nilai akurasi rata-rata sebesar 0,7812 atau 78,12%. Sedangkan model *Support Vector Machine* menghasilkan nilai akurasi rata-rata yang lebih tinggi sebesar 0,8076 atau 80,76%. Dari hasil perhitungan tingkat akurasi, model *Support Vector Machine* adalah model terbaik dengan akurasi rata-rata sebesar 80,76%, Akurasi sebesar 80,76% dari SVM dianggap baik dalam konteks analisis sentimen aplikasi layanan publik, melebihi benchmark dari beberapa studi sebelumnya yang sering kali mencapai sekitar 75-80%. Ini menunjukkan bahwa model yang digunakan efektif dalam mengklasifikasikan sentimen ulasan pengguna, ulasan diklasifikasikan dengan benar seperti dalam *Confusion Matrix Support Vector Machine* yang ditunjukkan dalam Gambar 2.



Gambar 2. *Confusion Matrix Support Vector Machine*

Dari Gambar 2, didapat 340 ulasan positif diklasifikasikan dengan tepat dan 76 ulasan positif tidak diklasifikasikan dengan tepat. Selain itu, untuk ulasan negatif, 332 ulasan diklasifikasikan dengan tepat dan 84 ulasan tidak diklasifikasikan dengan tepat. Ini menunjukkan bahwa model *Support Vector Machine* sudah cukup baik dalam mengklasifikasikan ulasan aplikasi M-Paspor.

Penelitian ini memiliki beberapa keterbatasan, antara lain potensi bias yang disebabkan oleh pengumpulan data yang terbatas hanya dari *Google Play Store*, serta keterbatasan kamus leksikon yang mungkin tidak sepenuhnya mencakup semua variasi bahasa pengguna. Selain itu, dapat dilakukan penerapan parameter yang berbeda untuk algoritma *Support Vector Machine* dan *Naive Bayes* demi menghasilkan akurasi yang lebih mendekati 100%.

F. Visualisasi Data

Visualisasi data *wordcloud* adalah representasi visual dari kumpulan kata-kata dengan bobot visual yang lebih besar untuk setiap kata yang dominan ada dalam teks. Representasi grafis frekuensi kata umumnya menerapkan variasi ukuran teks, di mana kata yang sering muncul ditampilkan dengan ukuran font yang lebih besar, dan kata yang jarang muncul ditampilkan dengan ukuran font yang lebih kecil. Pendekatan ini memberikan gambaran visual tentang distribusi frekuensi kata dalam teks. Seperti analisis sentimen, ulasan pelanggan, atau pemrosesan teks lainnya, *Wordcloud* sering digunakan untuk memberikan gambaran besar mengenai topik yang sering digunakan dalam teks.

1) Visualisasi *Wordcloud* Sentimen Positif

Dalam visualisasi *wordcloud* sentimen positif, ditemukan bahwa tiga kata paling sering muncul selain kata kunci, yaitu "mudah", "aplikasi", dan "paspor" dengan masing-masing muncul sebanyak 73, 71, dan 53 kali. Ini menunjukkan bahwa "mudah" adalah salah satu kata kunci yang memicu sentimen positif.

- dan ..., vol. 6, no. 9, pp. 4305–4313, 2022, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [13] M. Nurmalasari, N. A. Temesvari, and ..., "Analisis Sentimen terhadap Opini Masyarakat dalam Penggunaan Mobile-JKN untuk Pelayanan BPJS Kesehatan Tahun 2019," *Indones. Heal. ...*, vol. 8, no. 1, pp. 35–44, 2020, [Online]. Available: <https://inohim.esaunggul.ac.id/index.php/INO/article/view/208>
- [14] R. Rismayani, H. SY, T. Darwansyah, and I. Mansyur, "Implementasi Algoritma Text Mining dan Cosine Similarity untuk Desain Sistem Aspirasi Publik Berbasis Mobile," *Komputika J. Sist. Komput.*, vol. 11, no. 2, pp. 169–176, 2022, doi: 10.34010/komputika.v11i2.6501.
- [15] O. I. Gifari, M. Adha, F. Freddy, and F. F. S. Durrand, "Analisis Sentimen Review Film Menggunakan TF-IDF dan Support Vector Machine," *J. Inf. Technol.*, vol. 2, no. 1, pp. 36–40, 2022, doi: 10.46229/jifotech.v2i1.330.
- [16] J. A. Septian, T. M. Fachrudin, and A. Nugroho, "Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor," *J. Intell. Syst. Comput.*, vol. 1, no. 1, pp. 43–49, 2019, doi: 10.52985/insyst.v1i1.36.
- [17] P. K. Sari and R. R. Suryono, "Komparasi Algoritma Support Vector Machine Dan Random Forest Untuk Analisis Sentimen Metaverse," *J. Mnemon.*, vol. 7, no. 1, pp. 31–39, 2024, doi: 10.36040/mnemonic.v7i1.8977.
- [18] A. Hendra and F. Fitriyani, "Analisis Sentimen Review Halodoc Menggunakan Naïve Bayes Classifier," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 6, no. 2, pp. 78–89, 2021, doi: 10.14421/jiska.2021.6.2.78-89.
- [19] M. Diki Hendriyanto, A. A. Ridha, and U. Enri, "Analisis Sentimen Ulasan Aplikasi Mola Pada Google Play Store Menggunakan Algoritma Support Vector Machine Sentiment Analysis of Mola Application Reviews on Google Play Store Using Support Vector Machine Algorithm," *J. Inf. Technol. Comput. Sci.*, vol. 5, no. 1, pp. 1–7, 2022.
- [20] C. F. Hasri and D. Alita, "Penerapan Metode Naïve Bayes Classifier Dan Support Vector Machine Pada Analisis Sentimen Terhadap Dampak Virus Corona Di Twitter," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 3, no. 2, pp. 145–160, 2022, doi: 10.33365/jatika.v3i2.2026.