

Analisis Sentimen Aplikasi *Get Contact* di *App Store* Menggunakan Metode SVM (Support Vector Machine)

N. Aulia¹, S. N. Sari², N. Wakhidah³

^{1,2,3} Universitas Semarang, Jl. Soekarno-Hatta Tlogosari Semarang, 50196, Indonesia

Info Artikel

Riwayat Artikel:

Received 2024-12-11

Revised 2025-02-11

Accepted 2025-01-26

Abstract – Current technological developments have led to a variety of new innovations to create applications that make it easier for users to manage phone calls, one of which is the *Get Contact* application. By managing phone calls, it is hoped that it will help users to minimize the occurrence of fraud or the like. The goal is to analyze user sentiment towards the *Get Contact* application by classifying user reviews into positive and negative categories through sentiment analysis. The Support Vector Machine method is used in this analysis process with a linear kernel approach to determine the accuracy of the *Get Contact* application review classification. The stages used in this research include data collection, preprocessing, labeling, split data, SVM model training, and model evaluation. This study shows that the Support Vector Machine (SVM) method classification of sentiment analysis of *Get Contact* application reviews on the *App Store* produces an accuracy value of 95.50%, negative precision 0.96, positive precision 0.95, negative recall 0.95, positive recall 0.96, positive and negative f-1 scores are the same, amounting to 0.95. As for the results of the most reviews are negative reviews with a negative review percentage of 94.8%, while for positive reviews it is 5.2%.

Keywords: Sentimen; SVM; *Get Contact*; Spam; *App Store*

Corresponding Author:

S. N. Sari

Email: seftina808@gmail.com



This is an open access article under the [CC BY 4.0](https://creativecommons.org/licenses/by/4.0/) license.

Abstrak– Perkembangan teknologi saat ini memunculkan beragam inovasi baru untuk menciptakan aplikasi yang memudahkan pengguna mengelola panggilan telepon, salah satunya adalah aplikasi *Get Contact*. Dengan mengelola panggilan telepon, diharapkan membantu pengguna untuk meminimalisir terjadinya penipuan atau semacamnya. Adapun tujuannya yaitu untuk menganalisis sentimen pengguna terhadap aplikasi *Get Contact* dengan mengklasifikasikan ulasan pengguna kedalam kategori positif dan negatif melalui analisis sentimen. Metode Support Vector Machine digunakan pada proses analisis ini dengan pendekatan kernel linear guna mengetahui tingkat akurasi dari pengklasifikasian ulasan aplikasi *Get Contact* tersebut. Tahapan yang digunakan pada penelitian ini diantaranya pengumpulan data, preprocessing, pelabelan, split data, pelatihan model SVM, dan evaluasi model. Penelitian ini menunjukkan bahwa pengklasifikasian metode Support Vector Machine (SVM) terhadap analisis sentimen ulasan aplikasi *Get Contact* di *App Store* menghasilkan nilai akurasi 95.50%, precision negatif 0.96, precision positif 0.95, recall negatif 0.95, recall positif 0.96, f-1 score positif dan negatif sama, sebesar 0.95. Sedangkan untuk hasil ulasan terbanyak ialah ulasan berkategori negatif dengan presentase ulasan negatif sebesar 94.8%, sedangkan untuk ulasan positif sebesar 5.2%.

Kata Kunci: Sentimen; SVM; *Get Contact*; Spam; *App Store*

I. PENDAHULUAN

Di era digital yang serba canggih seperti sekarang ini, telah muncul beragam sistem operasi yang canggih untuk gawai. Salah satu sistem operasi yang cukup dikenal di khalayak umum adalah sistem iOS. Di Indonesia, perangkat iOS cukup banyak di minati khalayak umum karena salah satu sistem terbaik di Indonesia adalah sistem operasi iOS yang dikembangkan oleh Apple Inc untuk perangkatnya[1]. Di sisi lain, kejahatan siber juga semakin meningkat dengan cepat. Indonesia mendapatkan peringkat ke-2 dunia pada kasus kejahatan siber. Pada kejahatan melalui telepon atau panggilan spam, Indonesia dikenal sebagai negara dengan volume panggilan spam yang tinggi, menempati posisi teratas di antara 20 negara yang menghadapi masalah serupa. Secara rata-rata, warga Indonesia mendapatkan 14 panggilan spam per hari, yang kebanyakan berasal dari nomor yang tidak dikenal dalam daftar kontak mereka. Salah satu aplikasi yang dapat digunakan untuk mengidentifikasi panggilan spam adalah aplikasi *Get Contact*. Aplikasi ini cukup populer di khalayak umum, karena menawarkan fitur yang beragam dan mudah digunakan untuk membatasi interaksi panggilan telepon pengguna. Dilansir dari *website official Get Contact*, selain memungkinkan untuk mengetahui bagaimana orang lain memberi nama kontak pengguna, ada beberapa fitur yang bisa dinikmati diantaranya yaitu dapat mengidentifikasi penelpon berdasarkan nomor telepon yang terdapat di kontak pengguna, mendeteksi nomor yang terindikasi spam atau penipuan[2]. Selain itu aplikasi ini juga memiliki fitur *messenger*, fitur skor kepercayaan pada nomor kontak pengguna lain, memiliki fitur tagar yang memungkinkan pengguna lain menambahkan tag yang akan muncul pada profil pengguna. *Get Contact* juga dikenal sebagai aplikasi *support multiplatform*, yang berarti aplikasi ini mendukung berbagai perangkat. Namun dengan adanya aplikasi *Get Contact* ini, muncul beragam reaksi serta beragam persepsi dari khalayak umum. Tentunya ada yang merespon dengan positif dan ada juga yang merespon negatif.

Di dalam aplikasi *Get Contact*, selain menyediakan fitur yang menarik tentunya terdapat beberapa kekurangan yang menyebabkan penilaian buruk *user experience* terhadap aplikasi tersebut. Maka dari itu, diperlukan tanggapan dari para pengguna terkait tingkat kepuasannya terhadap aplikasi *Get Contact* sebagai dasar untuk evaluasi perusahaan dalam mengukur kualitas produk yang ditawarkan. Ulasan yang telah di ambil dan di proses akan menghasilkan sebuah analisis. Analisis sentimen merupakan suatu proses mengidentifikasi emosi atau perasaan pengguna yang berupa teks, kemudian teks tersebut akan di analisis untuk memahami sentimen yang terkandung didalamnya. Analisis sentimen dilakukan karena adanya beragam emosi atau ekspresi dari khalayak umum.

Penelitian ini bertujuan untuk menganalisa bagaimana tingkatan sentimen dalam penggunaan *Get Contact* di *App Store* dengan menggunakan ulasan dari pengguna. Peneliti menggunakan *machine learning* untuk menganalisis sentimen pengguna aplikasi *Get Contact*. SVM (*Support Vector Machine*) akan digunakan dalam penelitian ini karena SVM merupakan salah satu metode klasifikasi menggunakan metode *machine learning* (*supervised learning*) yang memprediksi kelas berdasarkan pola dari hasil proses training, kemudian dipisahkan dengan garis pembatas (*hyperlane*)[3]. Pada penelitian ini, ulasan akan dibagi menjadi ulasan positif dan ulasan negatif. Adapun pengumpulan data, *preprocessing*, kemudian pelatihan model SVM menjadi *metodologi* yang akan dipergunakan saat menganalisa data. Beberapa kelebihan SVM diantaranya adalah kemampuan kernel guna menerapkan *input hyperplane* non-linear dalam dimensi yang lebih tinggi sebagai aspek dalam memperbaiki performanya. SVM menawarkan berbagai fungsi yang berguna melalui kernel, yang dibuat khusus untuk menangani masalah yang kompleks. Hal tersebut menyediakan fungsi-fungsi yang dapat mengatasi keterbatasan linearitas dan beroperasi pada data dengan struktur yang kompleks.

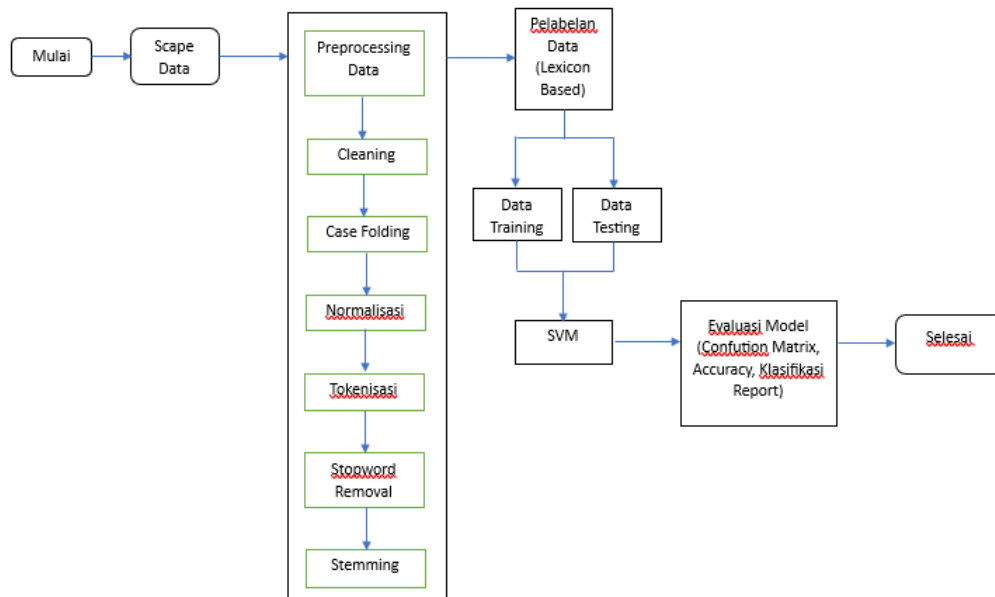
Dalam temuan penelitian sebelumnya, sejumlah peneliti telah memanfaatkan metode SVM guna menganalisis sentimen. Sebagai contoh, dalam analisis sentimen terhadap aplikasi Jamsostek Mobile, metode SVM menunjukkan kinerja optimal dengan tingkat akurasi mencapai 0,96 [4]. Kemudian penelitian analisis sentimen publik pada program kartu prakerja di *platform* Twitter dengan diterapkannya SVM berhasil memperoleh hasil terbaik dengan tingkat akurasi 98,67% [5]. Selain itu, penelitian mengenai pengimplementasian SVM pada data akreditasi sekolah dasar di Kabupaten Magelang mendapatkan nilai terbaik dengan *accuracy* 93. 902%[6]. Selain itu terdapat juga penelitian analisa sentimen masyarakat atas paylater dengan diterapkannya *naive bayes classifier* mendapatkan nilai terbaik dengan *accuracy* 91% untuk algoritmanya memperoleh nilai *accuracy* 61% dengan TextBlob. Kemudian ada juga penelitian tentang *comparison of the naive bayes method and SVM on twitter sentiment analysis* mendapatkan nilai terbaik yaitu sebesar 73.65% dari metode *naive bayes*[7]. Mengacu pada pernyataan tersebut, peneliti terdorong untuk mengeksplorasi lebih lanjut analisis sentimen dengan diterapkannya metode SVM pada aplikasi *Get Contact* yang tersedia di *App Store*. Tujuan dari analisis ini adalah untuk menganalisis kualitas dari aplikasi penyekat telepon tersebut, berdasarkan ulasan pengguna yang terdiri atas dua bagian yaitu ulasan positif dan ulasan negatif.

II. METODE

Penelitian ini menggunakan metode *platform* Google Colab. Google colab sebagai sebuah *platform* yang beropersi dengan basis teknologi *cloud* menggunakan bahasa pemrograman *Python* melalui *web browser*. Aplikasi tersebut memiliki *notebook* yang memungkinkan penggabungan kode yang dapat di eksekusi dan teks dalam satu dokumen, bersama dengan gambar, HTML, LaTeX, dan masih banyak lagi [7]. Program yang akan dijalankan pada google colab di antara nya berupa :

1. Pengumpulan Data: Mengumpulkan ulasan pengguna aplikasi *Get Contact* dari *Google Play Store* sebagai dataset.
2. *Preprocessing* Data: Proses ini berisi *cleaning*, *case holding*, normalisasi, tokenisasi, *stopword removal / filtering*, dan *stemming*. *Preprocessing* berfungsi untuk melakukan pembersihan data, termasuk penghapusan tanda baca, konversi teks menjadi huruf kecil, dan penghilangan *stop words* untuk meningkatkan kualitas analisis.
3. Pelabelan Data: Proses ini berisi *lexicon based*. *Lexicon based* ini merupakan kamus dengan kumpulan kata bersentimen positif dan negatif untuk menjadi *wordlist*.
4. *Split Data*: Proses ini terdiri dari data *training* dan data *testing*.
5. Pelatihan Model SVM: Membangun model klasifikasi dengan dataset yang telah diproses menggunakan algoritma *Support Vector Machine*.
6. Evaluasi Model: Proses ini terdiri dari *confusion matrix*, *accuracy*, dan klasifikasi report.

Berikut alur atau tahapan proses yang akan digunakan dalam penelitian ini di tunjukkan pada Gambar 1:



Gambar 1. Diagram alur proses

III. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Ada dua metode yang bisa dijalankan untuk pengumpulan data diantaranya yaitu, metode manual dan otomatis. Metode manual adalah cara menyalin data hanya dengan meng-copy-paste alamat aplikasi dari sebuah website. Sedangkan metode otomatis adalah metode yang dijalankan menggunakan coding, aplikasi, atau extension browser. Pada proses pengumpulan data ini, peneliti menggunakan metode scrape data. Scrape data atau web scrapping merupakan sebuah metode atau informasi tertentu dengan jumlah yang besar secara online. Web scrapping ini bertujuan untuk melakukan ekstraksi sejumlah data pada suatu website atau aplikasi[8]. Pada tahapan ini, peneliti mengimpor ulasan di App Store dengan jumlah 1000 ulasan dari tahun 2019 sampai tahun 2024 menggunakan metode otomatis, yaitu dengan menggunakan Google Colab. Hasil import ulasan menggunakan Google Colab ditunjukkan pada Gambar 2.

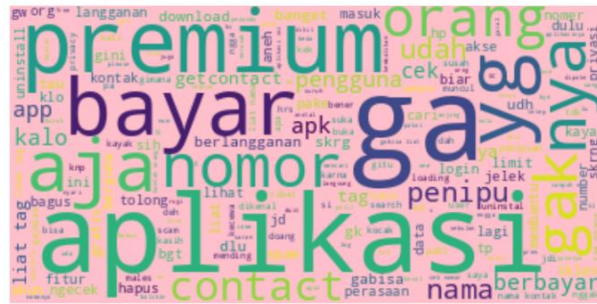
```

                                date                                review \
195 2024-12-30 16:35:21                                Buruk
196 2024-12-30 12:17:37  Pengen get contact kaya dlu, bisa cek nama pen...
197 2024-12-30 03:38:14                                Not good
198 2024-12-30 03:35:31  Aplikasi ribet, dan tidak membantu sama sekali...
199 2024-12-30 02:04:53  Sekarang ribet, mau lihat kontak aja musti jad...
.. ..
1   2020-02-28 06:40:35                                The code never show
13  2020-02-25 09:00:19                                Kok kode nya gk dikirim2 ya? Aneh
93  2019-10-25 09:57:41  Hello. I cannot open the app since i've update...
12  2019-05-30 17:31:30                                Udh get kode tapi gak dapet2 kodenya lewat sms
6   2019-05-15 21:43:38  Berkali-kali ngirim sms gabisa terus! ga dapet...

```

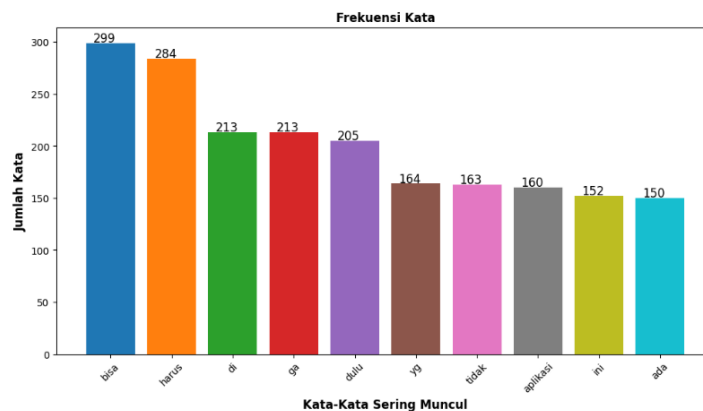
Gambar 2. Hasil import ulasan pada App Store

Dengan menggunakan Google Colab, peneliti dapat melihat visualisasi kata yang umumnya diperlihatkan pada Gambar 3. Kata yang sering muncul ini merupakan suatu sentimen atau perasaan pengguna. Dengan melihat wordcloud yang ada, peneliti dapat lebih mudah untuk memahami pandangan atau perasaan pengguna terhadap aplikasi Get Contact.



Gambar 3. Wordcloud sebelum *preprocessing*

Selain *wordcloud*, visualisasi kata yang sering muncul juga bisa berupa diagram frekuensi kata seperti yang ditunjukkan pada Gambar 4.



Gambar 4. Diagram frekuensi kata sebelum *preprocessing*

B. *Preprocessing Data*

Preprocessing data sebagai serangkaian proses yang dilakukan terhadap dataset sebelum diaplikasikan dalam analisis atau pemodelan. Tahapan ini dirancang untuk membersihkan, menormalkan, dan mengatur data sehingga dapat diproses dengan lebih efektif oleh algoritma analitik atau model pembelajaran mesin. Proses ini meliputi pembersihan, penghalusan, dan transformasi data, sembari mempertahankan validitas dan integritas informasi yang ada[9]. Langkah dari *preprocessing* diantaranya:

1) *Cleaning*

Proses eliminasi elemen-elemen yang tidak sesuai dengan aturan yang ditetapkan, seperti karakter di luar alfabet a-z (termasuk tanda baca), tautan atau *URL*, *hashtag*, hingga nama pengguna[10]. Tahapan ini dilakukan sebagai langkah awal sebelum analisis data, mengingat data mentah umumnya mengandung informasi yang kurang akurat, tidak terstruktur dengan baik, atau tidak lengkap. Gambar 5 menunjukkan hasil dari proses ini.

```

cleaning \
195          Buruk
196 Pngen get contact kaya dlu bisa cek nama peng...
197          Not good
198 Aplikasi ribet dan tidak membantu sama sekali ...
199 Sekarang ribet mau lihat kontak aja musti jadi...
    
```

Gambar 5. Hasil *cleaning*

2) *Case Folding*

Case folding menjadi tahapan penting dalam pra-pemrosesan teks yang mengubah semua huruf dalam sebuah dokumen ke bentuk huruf kecil. Karakter non-alfabetik juga dihilangkan dan dianggap sebagai delimitter. Hasil dari proses *case folding* dapat dilihat pada Gambar 6.

```
case_folding \  
195          buruk  
196 pengan get contact kaya dlu bisa cek nama peng...  
197          not good  
198 aplikasi ribet dan tidak membantu sama sekali ...  
199 sekarang ribet mau lihat kontak aja musti jadi...
```

Gambar 6. Hasil *case folding*

3) Normalisasi Kata

Proses normalisasi kata merupakan tahapan untuk melakukan pemrosesan data yang berfungsi menghilangkan redundansi data dan mengubah teks menjadi bentuk yang seragam atau standar. Tahapan ini berisi proses penghapusan karakter yang tidak relevan, penggantian kata tidak baku ke kata baku, memperbaiki kesalahan ejaan, mengubah singkatan atau akronim ke bentuk yang lengkap, serta penghilangan kata yang tidak diperlukan seperti pada Gambar 7. Dengan melakukan normalisasi kata, konsistensi data akan meningkat, mengurangi dimensi fitur, dan akan meningkatkan akurasi model.

```
normalisasi \  
195          buruk  
196 pengan get contact kayak dulu bisa cek nama pe...  
197          not good  
198 aplikasi ribet dan tidak membantu sama sekali ...  
199 sekarang ribet mau lihat kontak saja musti jad...
```

Gambar 7. Hasil normalisasi

4) Tokenisasi

Tahap ini merupakan tahapan untuk memecah teks menjadi unit yang terkecil, yang diistilahkan dengan token. Unit ini dapat meliputi karakter, frasa ataupun kata. Tokenisasi berfungsi mentransformasi teks mentah menjadi komponen yang lebih terorganisir sistematis maupun mudah diproses komputer. Proses tokenisasi mencakup metode *Rule-based tokenization*, *statistical tokenization*, dan tokenisasi *modern*. Dalam penelitian ini, peneliti menggunakan metode *Rule-based tokenization* yang kemudian menghasilkan tokenisasi seperti yang dirincikan Gambar 8.

```
tokenize  
195          [buruk]  
196 [pengan, get, contact, kayak, dulu, bisa, cek,...  
197          [not, good]  
198 [aplikasi, ribet, dan, tidak, membantu, sama, ...  
199 [sekarang, ribet, mau, lihat, kontak, saja, mu...
```

Gambar 8. Hasil tokenisasi

5) Stopword Removal

Setelah melakukan tokenisasi, peneliti melakukan tahap *stopword removal*. Tahapan ini berfungsi menghapuskan kata yang bersifat umum yang tidak menjelaskan informasi yang penting, seperti kata penghubung atau kata benda umum. *Stopword removal* dapat dijalankan dengan beberapa *tools* diantaranya NLTK, SpaCy, dan Sastrawi. Peneliti menggunakan NLTK untuk proses *stopword removal* ini. Adapun tahapan ini dirincikan oleh Gambar 9.

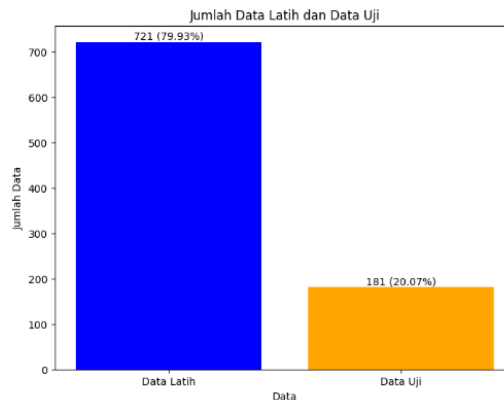
```
stopword_removal  
0 [pengguna, biar, akses, nama, kontaknya, ya, p...  
1          [the, code, never, show]  
2 [im, trying, find, out, the, identity, from, a...  
3 [hatihati, aplikasi, scam, data, rampas, downl...  
4 [biar, data, track, aplikasi, sih, kerja, apli...
```

Gambar 9. Hasil *stopword removal*

6) Stemming

Tahapan ini berfungsi untuk mengubah kata menjadi guna mengurangi variasi morfologis maupun bentuk dasar. *Stemming* memiliki tujuan untuk meningkatkan pemrosesan teks dalam pembelajaran mesin dan sistem pencarian informasi. Hasil proses ini berupa kata dasar dari kata turunan atau kata imbuhan (*prefix*, *suffix*, *infix*, atau kombinasi), yang ditampilkan pada Gambar 10.

pelatihan dan 20% untuk pengujian[11] yang kemudian menghasilkan 721 data latih dan 181 data uji seperti pada Gambar 13. Adapun prosesnya terbagi dari dua subset yang bertujuan guna melatih model maupun menguji data.



Gambar 13. Diagram jumlah data latih dan data uji

E. Support Vector Machine (SVM)

SVM sebagai metode pembelajaran mesin yang beroperasi berdasarkan prinsip SRM dengan tujuannya guna menemukan *hyperplane* optimal untuk memisahkan dua kelas dalam ruang input[12]. Pendekatan ini memanfaatkan ruang hipotesis berupa fungsi-fungsi linear dalam fitur berdimensi tinggi, yang dilatih menggunakan algoritma pembelajaran berbasis teori optimasi. Peneliti menggunakan SVM dengan kernel linear seperti yang di tunjukkan pada Gambar 14, di mana model dilatih dengan digunakannya data yang akan diubah menjadi TF-IDF untuk memprediksi sentimen. TF-IDF digunakan untuk mengubah representasi numerik dengan menghitung nilai TF-IDF (*Term Frequency-Inverse Document Frequency*) atas setiap kata dalam teks.

```
svm_model = SVC(kernel='linear')  
svm_model.fit(X_train_vectorized, y_train)
```

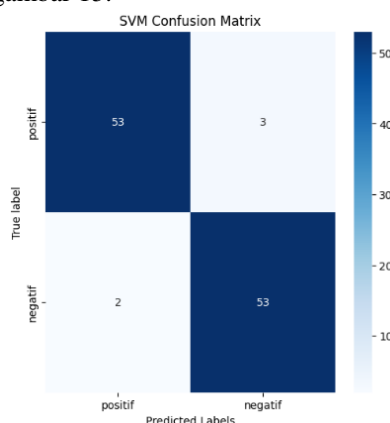
Gambar 14. SVM dengan kernel linear

F. Evaluasi Model

Tahapan ini bertujuan untuk menilai kinerja model yang telah dikembangkan, sekaligus menentukan nilai akurasi dan loss. Nilai akurasi merepresentasikan tingkat keberhasilan model dalam menjalankan tugasnya, menjadi indikator utama efektivitas model yang telah dirancang[13]. Evaluasi model menghasilkan *confusion matrix*, *accuracy*, dan klasifikasi report yang di dalamnya terdapat nilai *precision*, *recall*, *f1-score*, serta *support*, seperti sebagai berikut :

1) Confusion Matrix

Confusion matrix merupakan suatu alat evaluasi kinerja dalam klasifikasi data mining yang memberikan gambaran menyeluruh tentang hasil prediksi model. Hasil prediksi dari *confusion matrix* ini berbentuk diagram, seperti pada gambar 15.



Gambar 15. Diagram *confusion matrix*

Sedangkan vektorisasi yang telah dilakukan pada evaluasi model menghasilkan vektorisasi matriks untuk data latih seperti pada gambar 16.

```
Matriks Vektorisasi untuk Data Latih:  
[[0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 ...  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
  
Sebagian kecil Matriks Vektorisasi untuk Data Latih:  
[[0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]
```

Gambar 16. Matriks vektorisasi

2) Accuracy

Ukuran seberapa dekat data dengan nilai sebenarnya, atau seberapa tingkat kebenaran SVM informasi yang dikumpulkan merupakan *accuracy*. Pada penelitian ini, peneliti mendapatkan nilai SVM *accuracy* yang sangat bagus yakni 100% seperti yang di tujukkan pada Gambar 17.

```
SVM Accuracy: 0.954954954954955  
SVM Accuracy: 95.50%
```

Gambar 17. Hasil *accuracy* SVM

3) Klasifikasi Report

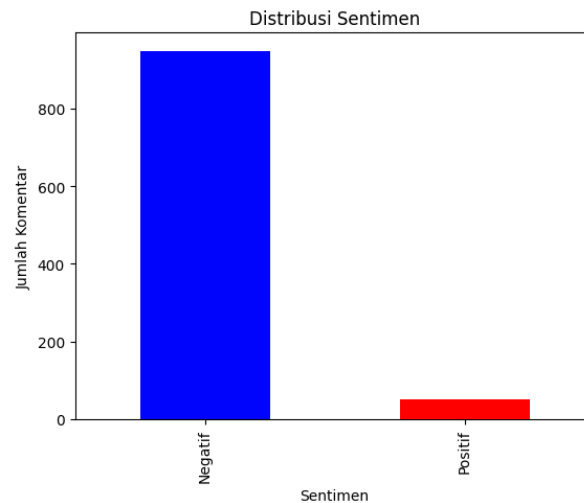
Klasifikasi report atau laporan klasifikasi yakni dokumen yang menampilkan metrik evaluasi performa suatu model klasifikasi, termasuk *precision*, *recall*, *F1-score*, dan *support*. *Precision* adalah rasio antara prediksi positif yang tepat terhadap keseluruhan prediksi positif. Metrik ini mengukur kemampuan model dalam mengidentifikasi data kelas positif dengan akurat, di mana nilai *precision* yang lebih tinggi menunjukkan bahwasanya model memiliki jumlah false positive yang rendah, yaitu situasi di mana model salah memprediksi suatu sampel sebagai positif padahal sebenarnya negatif[14]. *Recall*, di sisi lain sebagai metrik yang menghitung sejauh mana model mampu memprediksi semua kasus dalam kelas yang sama[15]. Sementara itu, *F1-score* sebagai rata-rata harmonis antara *precision* dan *recall*, memberikan gambaran keseimbangan antara keduanya dalam evaluasi model[16]. Pada penelitian ini, peneliti mendapatkan nilai klasifikasi report SVM yang berisi *precision*, *recall*, *f-1 score*, *support*, masing – masing nilai nya ditunjukkan pada Gambar 18.

```
SVM Classification Report:  
              precision    recall  f1-score   support  
  
negative       0.96      0.95      0.95         56  
positive       0.95      0.96      0.95         55  
  
accuracy                    0.95         111  
macro avg       0.96      0.96      0.95         111  
weighted avg    0.96      0.95      0.95         111
```

Gambar 18. Hasil klasifikasi report

Setelah melakukan klasifikasi report, peneliti melakukan visualisasi distribusi sentimenn yang berisi presentase sentimen negatif dan positif yang dihasilkan. Hal ini memungkinkan peneliti lebih efektif untuk mengetahui banyak masing – masing ulasan, seperti yang dirincikan pada Gambar 19.

```
Persentase Sentimen:  
sentiment  
Negatif 94.8  
Positif 5.2  
Name: count, dtype: float64
```



Gambar 19. Hasil distribusi sentimen

IV. SIMPULAN

Berdasarkan analisis penelitian di atas terhadap 1000 ulasan pada aplikasi *Get Contact* dari tahun 2019 sampai tahun 2024, maka dihasilkan ulasan negatif sebesar 94.8% dan positif 5.2%. Sedangkan berdasarkan penghitungan pada *confusion matrix*, nilai *accuracy* yang dihasilkan sebesar 95.50%. Dengan begitu, dari hasil penelitian ini, metode SVM sangat cocok digunakan untuk menganalisis sentimen pengguna pada aplikasi *Get Contact* di *App Store*. Melihat hasilnya, dapat dikatakan bahwa lebih banyak pengguna di *App Store* yang kurang suka dan tidak merekomendasikan menggunakan aplikasi *Get Contact*. Tentunya hal ini perlu lebih diperhatikan oleh pengembang, supaya aplikasi *Get Contact* dapat lebih berkembang lagi dan memiliki banyak peminat, khususnya pengguna di *App Store*.

UCAPAN TERIMAKASIH

Peneliti mengucapkan terima kasih terhadap Fakultas Teknologi Informasi dan Komunikasi Universitas Semarang atas dukungan dan kesempatan yang telah diberikan kepada peneliti untuk melakukan penelitian ini.

DAFTAR PUSTAKA

- [1] G. A. Sandag, "Prediksi Rating Aplikasi *App Store* Menggunakan Algoritma Random Forest," *CogITO Smart J.*, vol. 6, no. 2, pp. 167–178, 2020, doi: 10.31154/cogito.v6i2.270.167-178.
- [2] F. Hanifah and L. M. Wisudawati, "Analisis User Experience Pada Aplikasi *Get Contact* Menggunakan Metode Heuristic Evaluation," *J. Ilm. Inform. Komput.*, vol. 28, no. 3, pp. 186–202, 2023, doi: 10.35760/ik.2023.v28i3.9192.
- [3] I. Siti Aisah, B. Irawan, and T. Suprpti, "Algoritma Support Vector Machine (Svm) Untuk Analisis Sentimen Ulasan Aplikasi Al Qur'an Digital," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 6, pp. 3759–3765, 2024, doi: 10.36040/jati.v7i6.8263.
- [4] T. Tinaliah and T. Elizabeth, "Analisis Sentimen Ulasan Aplikasi PrimaKu Menggunakan Metode Support Vector Machine," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 4, pp. 3436–3442, 2022, doi: 10.35957/jatisi.v9i4.3586.
- [5] N. Hendrastuty, A. Rahman Isnain, and A. Yanti Rahmadhani, "Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja Pada Twitter Dengan Metode Support Vector Machine," *J. Inform. J. Pengemb. IT*, vol. 6, no. 3, pp. 150–155, 2021.
- [6] P. A. Octaviani, Yuciana Wilandari, and D. Ispriyanti, "Penerapan Metode Klasifikasi Support Vector Machine (SVM) pada Data Akreditasi Sekolah Dasar (SD) di Kabupaten Magelang," *J. Gaussian*, vol. 3, no. 8, pp. 811–820, 2014.
- [7] Alfandi Safira and F. N. Hasan, "Analisis Sentimen Masyarakat Terhadap Paylater Menggunakan Metode Naive Bayes Classifier," *Zo. J. Sist. Inf.*, vol. 5, no. 1, pp. 59–70, 2023, doi: 10.31849/zn.v5i1.12856.
- [8] K. Dwicahyo and C. Indah Ratnasari, "Perbandingan Metode Web Scraping Dalam Pengambilan Data: Kajian Literatur," *Automata*, vol. 4, 2023.
- [9] F. Putra, H. F. Tahiyat, R. M. Ihsan, R. Rahmaddeni, and L. Efrizoni, "Penerapan Algoritma K-Nearest Neighbor Menggunakan Wrapper Sebagai Preprocessing untuk Penentuan Keterangan Berat Badan Manusia," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 1, pp. 273–281, 2024, doi: 10.57152/malcom.v4i1.1085.
- [10] D. Oktavia, Y. R. Ramadhan, and Minarto, "Analisis Sentimen Terhadap Penerapan Sistem E-Tilang Pada Media Sosial Twitter Menggunakan Algoritma Support Vector Machine (SVM)," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 1, pp. 407–417, 2023, doi: 10.30865/klik.v4i1.1040.
- [11] Y. Astuti, I. R. Wulandari, A. R. Putra, and N. Kharomadhona, "Naïve Bayes untuk Prediksi Tingkat Pemahaman Kuliah Online Terhadap Mata Kuliah Algoritma Struktur Data," *J. Edukasi dan Penelit. Inform.*, vol. 8, no. 1, p. 28, 2022, doi: 10.26418/jp.v8i1.48848.

- [12] Y. X. Chu, X. G. Liu, and C. H. Gao, "Multiscale models on time series of silicon content in blast furnace hot metal based on Hilbert-Huang transform," *Proc. 2011 Chinese Control Decis. Conf. CCDC 2011*, pp. 842–847, 2011, doi: 10.1109/CCDC.2011.5968300.
- [13] N. S. Syam *et al.*, "Model Support Vector Machine untuk Prediksi pada Penggunaan Energi Listrik di Rumah Hemat Energi," *J. Inform.*, vol. 1, no. 2, pp. 56–59, 2022, [Online]. Available: <https://jurnal.uniraya.ac.id/index.php/JI>
- [14] A. Akram, K. Fayakun, and H. Ramza, "Klasifikasi Hama Serangga pada Pertanian Menggunakan Metode Convolutional Neural Network," *Build. Informatics, Technol. Sci.*, vol. 5, no. 2, pp. 397–406, 2023, doi: 10.47065/bits.v5i2.4063.
- [15] K. Citra, P. Daun, and T. Padi, "Klasifikasi Citra Penyakit Daun Tanaman Padi Menggunakan CNN dengan Arsitektur VGG-19," *J. Sains dan Inform.*, vol. 9, no. 1, pp. 37–45, 2023, doi: 10.22216/jsi.v9i1.2175.
- [16] Y. S. HARIYANI, S. HADIYOSO, and T. S. SIADARI, "Deteksi Penyakit Covid-19 Berdasarkan Citra X-Ray Menggunakan Deep Residual Network," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 8, no. 2, p. 443, 2020, doi: 10.26760/elkomika.v8i2.443.