

Perbandingan Algoritma SVM dan Decision Tree Dalam Klasifikasi Kepuasan Pengguna Aplikasi Migo E-Bike di Playstore

Dina Sakinah Al Azkiah, Erizal*, Fitri Nur Hikmah

Fakultas Teknologi Industri dan Informatika, Teknik Informatika, Universitas Muhammadiyah Prof. Dr. Hamka, Jakarta, Indonesia

Email: ¹dinasakinah2142002@gmail.com, ^{2,*}erizal@uhamka.ac.id, ³Nhfifi1205@gmail.com

Email Penulis Korespondensi: erizal@uhamka.ac.id

Submitted: 11/06/2024; Accepted: 25/06/2024; Published: 26/06/2024

Abstrak—Saat ini transportasi menjadi kebutuhan wajib dalam kehidupan sehari-hari, dan lajuperkembangan digital sudah memberikan dampak besar dalam penggunaan layanan dan interaksi dengan aplikasi mobile, termasuk di bidang transportasi. Aplikasi Migo E-Bike adalah aplikasi sepeda listrik pertama di Indonesia dengan sistem layanan sewa yang menghadirkan pelayanan ramah terhadap lingkungan untuk mengurangi pencemaran udara. Penelitian ini bertujuan untuk mencari efektivitas dari 2 algoritma data mining yaitu algoritma SVM dan algoritma Decision Tree saat mengklasifikasikan kepuasan pengguna aplikasi Migo E-Bike berdasarkan ulasan dan rating di Playstore. Hasil penelitian menunjukkan *Decision Tree* menghasilkan kinerja lebih baik dibanding dengan SVM. Didapatkan nilai akurat 76.39% dimana presisi dan recall yang seimbang untuk kedua kategori kepuasan dari *Decision Tree*. Sebaliknya, SVM menunjukkan ketidakseimbangan yang signifikan dengan akurasi keseluruhan hanya 51.25%. Oleh karena itu, Decision Tree lebih efektif dalam menangani dataset rating pengguna Migo E-Bike.

Kata Kunci: Transportasi; Migo E-Bike; Kepuasan pengguna; Decision Tree; Support Vector Machine

Abstract—Currently, transportation has become an essential need in daily life, and the rapid development of digital technology has had a significant impact on the use of services and interaction with mobile applications, including in the transportation sector. The Migo E-Bike app is the first electric bike rental service application in Indonesia, offering environmentally friendly services to reduce air pollution. This research aims to assess the effectiveness of two data mining algorithms, SVM and Decision Tree, in classifying user satisfaction of the Migo E-Bike app based on reviews and ratings on the Playstore. The research findings indicate that the Decision Tree algorithm performs better than SVM. The Decision Tree achieved an accuracy of 76.39%, with balanced precision and recall for both satisfaction categories. In contrast, SVM exhibited significant imbalance with an overall accuracy of only 51.25%. Therefore, the Decision Tree algorithm is more effective in handling the user rating dataset for the Migo E-Bike app.

Keywords: Transportation; Migo E-Bike; User Satisfaction; Decision Tree; Support Vector Machine

1. PENDAHULUAN

Saat ini transportasi menjadi kebutuhan wajib dalam kehidupan sehari-hari. Kita menggunakan transportasi untuk memudahkan perpindahan tempat dari satu lokasi ke lokasi lainnya, seperti pergi ke kantor dan sekolah, mengunjungi tempat wisata serta area publik lainnya. Dalam hal Kemajuan teknologi digital telah menghadirkan perubahan yang signifikan terhadap perilaku kita dalam menggunakan layanan dan berinteraksi dengan berbagai aplikasi mobile. Pengaruh perkembangan ini pun juga dialami oleh bidang transportasi. Dimana perusahaan penyedia aplikasi transportasi mulai membangun sebuah aplikasi berbasis internet yang mudah untuk diunduh melalui Google Playstore. Keberadaan transportasi yang dikembangkan dalam sebuah aplikasi kini juga sudah merambah dalam aplikasi layanan sewa sepeda listrik sehingga bisa mengurangi polusi udara. Penggunaan sepeda motor listrik dianggap tepat karena sebagai sumber energi memiliki kerangka ketahanan energi yaitu affordability and acceptability yang lebih baik dibandingkan BBM[1].

Terdapat salah satu penyedia jasa layanan rental sepeda listrik yang mampu meminimalisir timbulnya pencemaran udara di lingkungan yang ditimbulkan oleh kendaraan yang menggunakan BBM. Kendaraan ramah lingkungan ini adalah Migo *Electric Bike*/E-Bike, sebuah aplikasi berbasis *mobile* yang menyediakan layanan penyewaan sepeda listrik pertama di Indonesia. Dengan semakin banyaknya ulasan dan *feedback* dari pengguna di platform seperti *Playstore*, muncul kebutuhan untuk menganalisis data ini guna memahami tingkat kepuasan pengguna. Layanan dikatakan baik jika apa yang diperoleh konsumen sesuai dengan apa yang diharapkan dan diinginkan, konsumen akan membandingkannya[2]. Kepuasan pengguna termasuk kedalam faktor penentu keberhasilan suatu aplikasi mobile. Tingkat kepuasan ini seringkali diukur melalui ulasan dan rating yang diberikan oleh pengguna di berbagai platform. Hal ini juga dapat berguna untuk perbaikan dan pengembangan selanjutnya pada jasa yang disediakan[3].

Studi yang sudah dilakukan oleh Indi Nurul Hassanah dkk memberikan hasil penelitian yang menunjukkan bahwa algoritma SVM memiliki akurasi sebesar 84.2%, sedangkan algoritma *Decision Tree* memiliki akurasi sebesar 70%[4]. Selain penelitian tersebut, dalam penelitian lain oleh Yerik Afrianto Singgalen, dilakukan analisis dan pembahasan konteks Top 10 Hotel rekomendasi Tripadvisor di kota Makassar. Pada tahap *data preparation*, jumlah data teks yang telah diproses menggunakan algoritma *Decision Tree (DT)* dan *Support Vector Machine (SVM)* sebesar 1.138 melalui *pre-processing* data (*tokenize, transform cases, filter tokens by length, stopwords, stemming*). Pada tahap *modeling*, algoritma yang menunjukkan performa terbaik ialah SVM dengan nilai akurasi 98,98%, nilai presisi 100%, nilai *recall* 97,96%, nilai *f-measure* 98,97%, dan nilai AUC 100%. Pada tahap evaluasi, dapat diketahui bahwa klasifikasi ulasan dominan pada sentimen positif dibandingkan sentimen negatif[5].

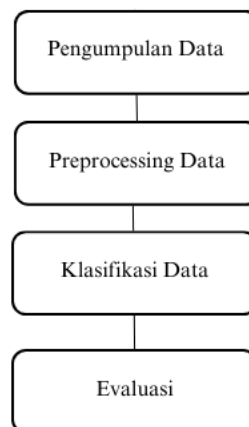
Dalam penelitian yang dilakukan oleh Khamim Surya Hadi Kusuma Al Atros dkk, bahwa pada proses penelitian ini adalah gabungan dari 2 model, yaitu model K-Means dan model Decision Tree. Sehingga penelitian ini bukanlah penelitian untuk membandingkan 2 metode. Berdasarkan hasil penelitian yang dilakukan untuk analisis kepuasan pengguna perpustakaan digital menggunakan algoritma K-Means dan decision tree, dapat disimpulkan bahwa dengan dataset berjumlah 150 data menghasilkan persentase akurasi sebesar 95,33% [6]. Namun, masih terdapat beberapa kesenjangan dalam penelitian yang telah dilakukan. Pertama, belum ada penelitian sebelumnya yang secara khusus fokus pada aplikasi Migo E-Bike dalam konteks transportasi ramah lingkungan. Selanjutnya penelitian ini akan menggunakan data ulasan dan rating dari Playstore yang relevan dengan aplikasi Migo E-Bike, yang belum banyak dijadikan fokus dalam penelitian sebelumnya. Terakhir Pada penelitian ini memiliki tujuan untuk mengevaluasi efektivitas dua algoritma tersebut dalam mengklasifikasikan kepuasan pengguna aplikasi Migo E-Bike melalui perbandingan performa kedua algoritma tersebut.

SVM dan Decision Tree sudah banyak digunakan dalam tugas-tugas klasifikasi, termasuk klasifikasi sentimen. SVM adalah suatu model klasifikasi dengan kemampuan dapat mengenali pola dari data yang diberikan. Karena SVM adalah classifier, maka diberikan pelatihan set, masing-masing ditandai sebagai milik salah satu dari dua kategori, konstruksi algoritma pelatihan SVM model yang memprediksi apakah data baru termasuk dalam kategori atau lainnya[7]. Di sisi lain, Decision Tree adalah algoritma klasifikasi data yang menggunakan pohon Keputusan[8]. Decision Tree merupakan algoritma yang membangun rangkaian yang menggambarkan pertanyaan serta jawaban dari sebuah dataset sehingga dapat digunakan untuk mengklasifikasikan data. Teknik ini terdiri dari kumpulan decision node, dan dihubungkan oleh cabang, bergerak ke bawah dari root node sampai berakhir di leaf node[9]. Meskipun keduanya memiliki kelebihan masing-masing, pertanyaan yang muncul adalah algoritma mana yang lebih efektif untuk mengklasifikasikan kepuasan pengguna aplikasi Migo E-Bike berdasarkan rating di *Playstore*.

2. METODOLOGI PENELITIAN

2.1 Alur Penelitian

Peneliti menggunakan *dataset* yang diperoleh dari aplikasi *Google Playstore* dengan menggunakan *tools Google Collab* untuk melakukan penarikan data. Setelah diperoleh dataset yang diperlukan, maka dilakukan proses *preprocessing* data. Selanjutnya akan dilakukan klasifikasi menggunakan dua algoritma yaitu *Decision Tree* dan *SVM* serta tahapan terakhir yaitu evaluasi dengan penerapan alur penelitian yang terdapat pada gambar 1.



Gambar 1. Alur Penelitian

Dibawah ini penjelasan dari alur atau rangkaian penelitian yang ada pada gambar 1:

a. Pengumpulan Data

Peneliti menggunakan *Google Collab* sebagai alat untuk menarik data pada tahapan pertama. Dalam tahapan ini, data yang digunakan berasal dari rating penggunaan aplikasi Migo Ebike pada *Playstore*.

b. *Preprocessing* Data

Preprocessing data merupakan tahapan mengubah data mentah menjadi data yang siap diolah. Pada tahapan ini, data diolah menggunakan beberapa proses diantaranya *case folding*, *tokenizing*, *stopword* dan *stemming*. Selain itu *preprocessing* data dilakukan untuk melakukan pembersihan data agar data tersebut siap untuk dilakukan pemodelan data[4].

c. Klasifikasi Data

Data yang sudah melewati proses *preprocessing* maka sudah bisa untuk dilakukan pemodelan data. Dalam tahap ini, dataset dipartisi menjadi dua bagian, yakni data latih dan data uji, menggunakan suatu operator. Data latih kemudian dimanfaatkan untuk proses pelatihan. data menggunakan algoritma SVM dan *Decision Tree*, kemudian

model akan mencari pola dan hubungan dalam data latih untuk membuat prediksi. Hasil prediksi model dari 2 algoritma yang digunakan pada data latih dibandingkan dengan nilai sebenarnya dari data uji. Matrix evaluasi yang digunakan *accuracy*, *precision*, dan *recall*.

d. Evaluasi

Dalam proses ini dilakukan perbandingan dari kedua algoritma untuk mengukur efektivitas dengan membandingkan hasil *accuracy*, *precision*, dan terhadap kepuasan pengguna aplikasi Migo E-Bike.

2.2 Pengumpulan Data

Dataset yang dikumpulkan dari ulasan pengguna pada aplikasi Migo E-Bike diantaranya *review Id*, *user Name*, *user Image*, *content*, *score*. Pengambilan data ini dilakukan pada tanggal 29 November 2023, dengan memanfaatkan *tools Google Collab* dan menggunakan bahasa pemrograman *python*. Lalu dataset yang sudah diperoleh sejumlah 1114 di *export* dengan format *csv* dan pada Gambar 2 diberikan alur visualisasi terhadap proses yang terjadi saat tahapan pengumpulan data.



Gambar 2. Proses Pengumpulan Data

2.3 Preprocessing Data

Setelah mendapatkan dataset yang diperlukan maka tahapan selanjutnya, yaitu *Preprocessing* data. Dibawah ini merupakan tahapan yang dilakukan dalam *Preprocessing* data:

a. Case Folding

Pada tahap *case folding* ini salah satu fungsinya adalah untuk merubah huruf kapital atau huruf kecil dalam sebuah kata atau kalimat. Jika terdapat huruf kapital setiap kata maka akan diubah menjadi huruf kecil, proses tersebut bertujuan agar data mempunyai suatu format yang sama [10].

b. Tokenizing

Tahapan ini bertujuan untuk membagi sebuah kalimat menjadi per kata. Tujuan dari proses ini untuk menyortir kata serta menyeleksi [11].

c. Filtering

Proses *Filtering* merupakan proses untuk menghilangkan kata-kata penghubung yang kurang memiliki makna yang sesuai dengan topik pembahasan. Proses *filtering* ini biasa disebut juga dengan *stopword removal* [12].

d. Stemming

Merupakan proses mengembalikan kata yang berimbuhan ke bentuk asalnya, namun bentuk asal ini tidak sama dengan akar kata. Sebagai contoh, pada kata "membaca", setelah melalui proses *stemming*, kata tersebut akan menjadi "baca" [13].

2.4 Klasifikasi Data

Proses klasifikasi ini menggunakan *tools Rapidminer*. Pada tahapan klasifikasi ini, diawali dengan membagi dataset menjadi data latih dan data uji. Dataset dibagi menggunakan perbandingan dengan rasio 80 banding 20 dimana data latih sebanyak 289 data dan data uji sebanyak 72 data. Selanjutnya dilakukan pengujian data secara terpisah dengan menggunakan algoritma *Decision Tree* dan SVM. Pada klasifikasi ini menggunakan 2 parameter penilaian *score*, yaitu *score 1* dan *score 5* dimana ditunjukkan sebagai nilai "sangat puas" dan "tidak puas".

2.4.1 Decision Tree

Decision Tree adalah algoritma pembelajaran mesin yang digunakan untuk klasifikasi dan regresi. Algoritma ini membagi dataset menjadi subset berdasarkan fitur yang paling signifikan untuk memprediksi label target. Setiap node internal dalam pohon mewakili fitur pada dataset, setiap cabang mewakili aturan keputusan, dan setiap daun mewakili hasil atau label. Decision Tree sendiri juga sudah banyak digunakan dalam berbagai sektor diantaranya sektor medis seperti diagnosa penyakit berdasarkan gejala yang diberikan oleh pasien. Selain itu *Decision Tree* juga sudah digunakan pada sektor teknologi seperti mendeteksi intrusi dan keamanan jaringan.

Pada penelitian sebelumnya yang dilakukan oleh Indi Nurul Hassanah, Sutan Faisal dan Amril Mutoi Siregar dalam judul "Perbandingan Algoritma *Support Vector Machine* dengan *Decision Tree* pada Aplikasi Ruang Guru", hasil penelitian menunjukkan bahwa algoritma SVM memiliki akurasi sebesar 84.2%, sedangkan algoritma *Decision Tree* memiliki akurasi sebesar 70% [4].

2.4.2 Support Vector Machine

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang digunakan untuk klasifikasi dan regresi. Prinsip kerja SVM adalah mencari hyperplane terbaik yang memisahkan dua kelas dalam ruang fitur dengan margin terbesar. SVM bekerja baik dalam ruang berdimensi tinggi dan sangat efektif untuk kasus klasifikasi biner. SVM ini telah diaplikasikan dalam beberapa bidang diantaranya, bidang BioInformatika seperti klasifikasi gen dan analisis urutan DNA. Selain itu juga sudah diterapkan di bidang keuangan seperti memprediksi harga saham dan deteksi penipuan.

2.5 Evaluasi

Dalam penelitian ini, peneliti menerapkan algoritma *Decision Tree* dan *Support Vector Machine* (SVM) untuk klasifikasi kepuasan pengguna aplikasi Migo E-Bike di Playstore, dengan tahapan mulai dari pengumpulan data menggunakan *Google Colab*, *preprocessing* data melalui *case folding*, *tokenizing*, *filtering*, dan *stemming*, hingga klasifikasi data dengan *Rapidminer*, serta evaluasi menggunakan metode *confusion matrix* untuk mengukur efektivitas model berdasarkan *accuracy*, *precision*, dan *recall*. *Confusion matrix* merupakan metode prediksi perhitungan matrix pada akurasi data mining[14]. Perhitungan performa klasifikasi yang digunakan pada penelitian ini adalah *confusion matrix* yang berisi perhitungan hasil klasifikasi aktual yang dapat diprediksi[15]. Pada Tabel 1, diberikan rumus *confusion matrix* yang akan digunakann dalam menentukan nilai *accuracy*, *precision*, *recall*.

Tabel 1. *Confusion Matrix*

	Prediksi Sangat Puas	Prediksi Tidak Puas
Aktual Sangat Puas	TP	FN
Aktual Tidak Puas	FP	TN

Keterangan :

TP (True Positive) = Data positif diklasifikasikan secara benar

TN (True Negative) = Data negatif diklasifikasikan secara benar

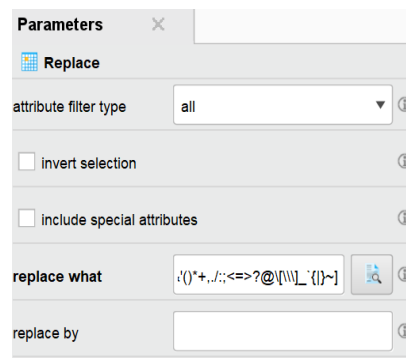
FP (False Positive) = Data negatif diklasifikasikan menjadi positif

FN (False Negative) = Data positif diklasifikasikan menjadi negatif

3. HASIL DAN PEMBAHASAN

3.1 Preprocessing Data

Pada tahap preprocessing, sebanyak 361 data skor penilaian akan diproses lebih lanjut. Selain itu *preprocessing* data dilakukan untuk melakukan pembersihan data agar data tersebut siap untuk pemodelan data[4]. Proses preprocessing melibatkan empat tahapan: tokenisasi, transformasi huruf, penyaringan kata-kata umum (stopword), dan penyaringan token berdasarkan panjang. Visualisasi dari preprocessing data dapat dilihat pada Gambar 3 di bawah ini.



Gambar 3. *Preprocessing* Data

3.4.1 Cleansing

Tahap *cleansing* akan dilakukan penghapusan karakter yang bukan termasuk alfabet dan mengurangi karakter yang tidak memiliki arti pada analisis sentimen. Contohnya penghapusan seperti @, #, %, *, dan emoji[10]. Pada tahapan ini digunakan operator *replace* dengan memilih *regular expression* nya dengan *punctuation character*. Dalam Gambar 4, diberikan gambar penggunaan atribut *replace* untuk menghapus karakter yang tidak digunakan.



Gambar 4. Tahap *cleansing* menggunakan operator *replace*

Tabel 2. Visualisasi proses *cleansing*

Ulasan	Cleaning	Rating
Keren,kendaraan kekinian, ramah lingkungan desain oke, buat jalan2 mantep betoolll 😊	Keren,kendaraan kekinian, ramah lingkungan desain oke, buat jalan2 mantep betoolll	Sangat Puas
Aplikasi ini sangat bgs krn mempermudah kita mengendarai motor,dan bisa untuk bljr motor 😊😊	Aplikasi ini sangat bgs krn mempermudah kita mengendarai motor,dan bisa untuk bljr motor	Sangat Puas
Gk bisa buat login dah di login masih aja disuruh login yg tau caranya kasih tau dong buat loginnya sudah berkali kali masih aja gk bisa 😞😞😞😞😞	Gk bisa buat login dah di login masih aja disuruh login yg tau caranya kasih tau dong buat loginnya sudah berkali kali masih aja gk bisa	Tidak Puas

Pada Tabel 2 diberikan gambar perubahan data saat proses *cleansing* untuk mendapatkan data yang sesuai.

3.4.2 Transform Case

Transform Case atau yang disebut sebagai *Case Folding* atau mengubah huruf menjadi huruf kecil[11]. merupakan langkah untuk mengubah semua huruf dalam teks menjadi huruf kecil ataupun sebaliknya. Hal ini dilakukan untuk memastikan konsistensi dataset dan menghindari perbedaan tujuan yang disebabkan dari kata yang berawal huruf besar maupun kecil. Pada Tabel3, diberikan gambaran berupa perubahan pada kalimat sebelum dan sesudah setelah dilakukan proses *transform case* dimana awalan kalimat yang menggunakan huruf besar akan diubah menjadi huruf kecil.

Tabel 3. Proses *transform case*

Ulasan	Transform Case	Rating
Keren,kendaraan kekinian, ramah lingkungan desain oke, buat jalan2 mantep betoolll	keren,kendaraan kekinian, ramah lingkungan desain oke, buat jalan2 mantep betoolll	Sangat Puas
Aplikasi ini sangat bgs krn mempermudah kita mengendarai motor,dan bisa untuk bljr motor	aplikasi ini sangat bgs krn mempermudah kita mengendarai motor,dan bisa untuk bljr motor	Sangat Puas
Gk bisa buat login dah di login masih aja disuruh login yg tau caranya kasih tau dong buat loginnya sudah berkali kali masih aja gk bisa	gk bisa buat login dah di login masih aja disuruh login yg tau caranya kasih tau dong buat loginnya sudah berkali kali masih aja gk bisa	Tidak Puas

3.4.3 Tokenize

Tokenize adalah proses memisahkan setiap kata atau teks ulasan serta menghapus tanda baca. Pada Tabel4, diberikan hasil dari proses tokenisasi:

Tabel 4. Proses *tokenize*

Ulasan	Tokenizing	Rating
keren,kendaraan kekinian, ramah lingkungan desain oke, buat jalan2 mantep betoolll	['keren'],['kendaraan'], ['kekinian'],['ramah'] ['lingkungan'],['desain'] ['oke'],['buat'],['jalan2'], ['mantep'],['betoolll']	Sangat Puas
aplikasi ini sangat bgs krn mempermudah kita mengendarai motor,dan bisa untuk bljr motor	['aplikasi'],['ini'], ['sangat'],['bgs'],['krn'], ['mempermudah'],['kita'], ['mengendarai'], ['motor'],['dan'],['bisa'], ['untuk'],['bljr'],['motor']	Sangat Puas
gk bisa buat login dah di login masih aja disuruh login yg tau caranya kasih tau dong buat loginnya sudah berkali kali masih aja gk bisa	['gk'],['bisa'],['buat'], ['login'],['dah'],['di'], ['login'],['masih'],['aja'], ['disuruh'],['login'],['yg'], ['tau'],['caranya'],['kasih'], ['tau'],['dong'],['buat'], ['loginnya'],['sudah'], ['berkali'],['kali'],['masih'], ['aja'],['gk'],['bisa']	Tidak Puas

3.4.4 Filter Tokens

Filter tokens merupakan proses penyaringan kata dengan kriteria tertentu. Dalam penelitian ini, peneliti menentukan minimum kata sebanyak 4 huruf dan maximum kata sebanyak 25. Maka, untuk kata yang berada dibawah minimum penentuan akan dihapus, begitu juga yang melewati batas maximum maka akan dihilangkan. Hal ini digunakan agar data yang didapat semakin jelas.



Tabel 5. Proses *filter tokens*

Ulasan	Tokenizing	Rating
['keren'], ['kendaraan'], ['kekinian'], ['ramah']	['keren'], ['kendaraan'], ['kekinian'], ['ramah']	Sangat Puas
['lingkungan'], ['desain'] ['oke'], ['buat'], ['jalan2'], ['mantep'], ['betooilll']	['lingkungan'], ['desain'] ['buat'], ['jalan2'], ['mantep'], ['betooilll']	Sangat Puas
['aplikasi'], ['ini'], ['sangat'], ['bgs'], ['krn'], ['mempermudah'], ['kita'], ['mengendarai'], ['motor'], ['dan'], ['bisa'], ['untuk'], ['bljr'], ['motor']	['aplikasi'], ['sangat'], ['mempermudah'], ['kita'], ['mengendarai'], ['motor'], ['dan'], ['bisa'], ['untuk'], ['bljr'], ['motor']	Sangat Puas
['gk'], ['bisa'], ['buat'], ['login'], ['dah'], ['di'], ['login'], ['masih'], ['aja'], ['disuruh'], ['login'], ['yg'], ['tau'], ['caranya'], ['kasih'], ['tau'], ['dong'], ['buat'], ['loginnya'], ['sudah'], ['berkali'], ['kali'], ['masih'], ['aja'], ['gk'], ['bisa']	['bisa'], ['buat'], ['login'], ['login'], ['masih'], ['disuruh'], ['login'], ['caranya'], ['kasih'], ['dong'], ['buat'], ['loginnya'], ['sudah'], ['berkali'], ['kali'], ['masih'], ['bisa']	Tidak Puas

Pada Tabel5, bisa dilihat bahwa terdapat pengurangan kata seperti 'oke', 'ini', 'bgs', 'krn', 'dan', 'gk', 'dah', 'di', 'aja', 'yg', 'tau', 'tau' merupakan hasil filter token yang tidak sesuai dengan ketentuan minimum yang dibuat.

3.4.5 Filter Stopwords

Langkah terakhir dalam preprocessing adalah *Filter Stopword*. Ini adalah proses mengembalikan kata ke bentuk dasarnya, meskipun bentuk dasar tersebut tidak selalu sama dengan akar kata. *Filter Stopword* atau *stemming* juga menghapus kata-kata imbuhan seperti "ke", "dan", atau "yang" sehingga hanya kata-kata yang memiliki makna penting yang dipertahankan. Tabel 6, memperlihatkan proses dari *Filter Stopword*:

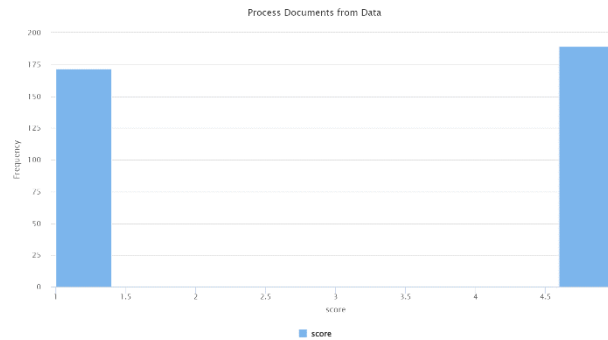
Tabel 6. Proses *filter stopwords*

Ulasan	Tokenizing	Rating
['keren'], ['kendaraan'], ['kekinian'], ['ramah']	['keren'], ['kendaraan'], ['kekinian'], ['ramah']	Sangat Puas
['lingkungan'], ['desain'] ['buat'], ['jalan2'], ['mantep'], ['betooilll']	['lingkungan'], ['desain'], ['jalan2'], ['mantep'], ['betooilll']	Sangat Puas
['aplikasi'], ['sangat'], ['mempermudah'] ['kita'], ['mengendarai'], ['motor'], ['dan'], ['bisa'], ['untuk'], ['bljr'], ['motor']	['aplikasi'], ['mempermudah'], ['mengendarai'], ['motor'], ['bljr'], ['motor']	Sangat Puas
['bisa'], ['buat'], ['login'], ['login'], ['masih'], ['disuruh'], ['login'], ['caranya'], ['kasih'], ['dong'], ['buat'], ['loginnya'], ['sudah'], ['berkali'], ['kali'], ['masih'], ['bisa']	['login'], ['login'], ['disuruh'], ['login'], ['caranya'], ['kasih'], ['dong'], ['loginnya'], ['sudah'], ['berkali'], ['kali']	Tidak Puas

Beberapa kata seperti 'buat', 'sangat', 'kita', 'dan', 'bisa', 'untuk', 'masih' dihapus karena merupakan kata imbuhan yang tidak relevan. Dengan demikian, hanya kata-kata yang memiliki makna penting yang disimpan.

3.2 Pelabelan Data

Penentuan label suatu ulasan didasarkan pada nilai penjumlahan secara keseluruhan kata [16]. Pelabelan Data merupakan proses dimana memberikan label berupa positif dan negatif. Label positif dan negatif diperoleh dari rating yang disematkan pada komentar [17]. Pada tahapan ini, dataset yang digunakan sebanyak 361 data. Hal ini dikarenakan dataset yang diambil hanyalah ulasan dan rating yang diberikan di *score* 1 dan 5. Dimana nilai 1 dianggap sebagai label negatif dan nilai 5 sebagai label positif. Dalam penelitian ini akan digunakan penilaian berupa "sangat puas" dan "tidak puas". Hasil pengumpulan data berbentuk csv akan dilakukan pelabelan secara manual. Gambar5, memberikan gambaran hasil visualisasi perbandingan dari kedua nilai tersebut:



Gambar 5. Hasil visualisasi perbandingan rating

3.3 Proses TF-IDF

Setelah melewati proses pelabelan data, maka selanjutnya tahapan pembobotan TF-IDF. Proses ini bertujuan untuk mendapatkan nilai dari kata dasar yang berhasil diekstrak, kemudian kata-kata dasar tersebut dikonversi menjadi sebuah *vector* yang mewakili kata yang bersangkutan[4]. Pada gambar 6. menunjukkan hasil proses TF-IDF yang dilakukan melalui operator *Process Documents From Data* pada proses *processing* yang dilakukan menggunakan Rapidminer. Gambar6, memberikan hasil dari proses TF-IDF.

text	score	abangnya	abis	abistu	about	ar
puas senang bermain memudahkan keliling monas abangnya terimakasih	5	0.361	0	0	0	0
puas belum dowload disuruh kasih alasan mengenali gara gara teman make ...	5	0	0	0	0	0
puas cuman diskon doang kesananya login doang kesana promo	5	0	0	0	0	0
puas nunggu otentikasi gagal tolong diperbaharui lebih baik kenyamanan ter...	1	0	0	0	0	0
puas verifikasi banget pelayanan gimana migo verifikasi tolong perbaiki verifi...	5	0	0	0	0	0
puas migo ngebantu banget kalo sekedar belanja makan jalan komplek ngab...	5	0	0	0	0	0
puas nggak dipake tulisan sorry this application wrong about exit abistu lang...	1	0	0	0.354	0.284	0
puas bintang upload foto station stuck website	1	0	0	0	0	0
puas aplikasi toleransi sumpah kecewa kebijakan solusi berulang kali mema...	1	0	0	0	0	0

Gambar 6. Hasil proses TF-IDF

3.4 Klasifikasi Data

Setelah data selesai pada tahap pelabelan data, maka selanjutnya dilakukan model klasifikasi dengan menggunakan algoritma *Decision Tree* dan *Support Vector Machine (SVM)*. Klasifikasi ini dilakukan dengan menggunakan data rating yang diberikan oleh pengguna aplikasi Migo *E-Bike* pada layanan *Google Playstore* terhadap kepuasan setelah pemakaian sepeda listrik Migo *E-Bike*. Rekapitulasi rating terhadap kepuasan pengguna terdapat dalam Tabel 7.

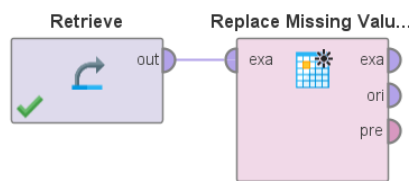
Tabel 7. Rekapitulasi rating terhadap kepuasan pengguna

Rating	Tidak Puas	Sangat Puas	Jumlah
1	134	38	172
5	47	142	189
Jumlah	181	180	361

Berdasarkan data yang terdapat pada tabel 7, pemberian nilai rating 1 masuk kedalam kategori tidak puas. Dan rating 5 termasuk dalam kategori sangat puas terhadap layanan yang diberikan oleh Migo *E-Bike*.

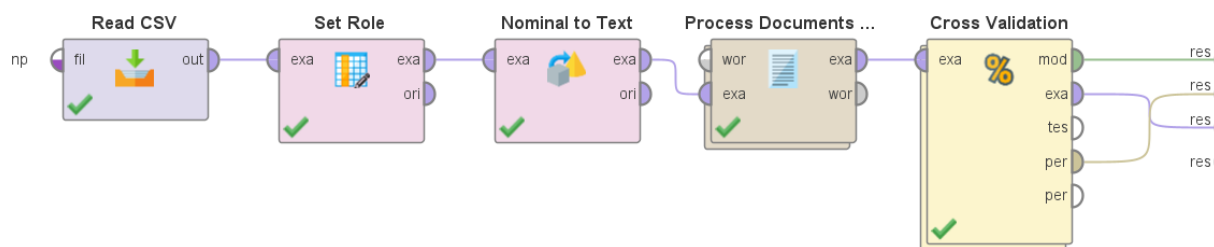
3.4.6 Decision Tree

Decision Tree merupakan algoritma yang membangun struktur pohon yang terdiri dari simpul dan cabang yang merepresentasikan pertanyaan dan jawaban yang dapat digunakan untuk mengklasifikasikan data. Teknik ini terdiri dari kumpulan *decision node*, dan dihubungkan oleh cabang, bergerak ke bawah dari root node sampai berakhir di leaf node[9]. *Decision Tree* dapat dibangun relatif cepat dibandingkan dengan metode klasifikasi lainnya[18]. Pada gambar 7 berikut, merupakan tahapan awal dimana akan dilakukan pengisian data yang hilang sebelum memasuki proses klasifikasi. Karena jika terdapat nilai yang hilang pada dataset, algoritma tidak dapat memproses klasifikasi sampai dataset yang digunakan berhasil terisi.



Gambar 7. Proses Replace Missing Values

Setelah dilakukan proses pengisian data yang hilang melalui operator *Replace Missing Values* tentunya barulah dapat dilakukan proses klasifikasi menggunakan algoritma *Decision Tree*. Selanjutnya dengan menggunakan beberapa operator pendukung seperti *split data*, yang dimana akan membagi data menjadi data training dan data testing. Lalu dilanjut dengan operator *Decision Tree* itu sendiri yang dibantu juga dengan operator *apply model* dan operator *performance* untuk dapat mendapatkan hasil *confusion matrix*. Dalam gambar 8 dibawah ini dijelaskan secara visualisasi dari proses yang dilakukan dalam penggunaan algoritma *Decision Tree* pada tools rapidminer.



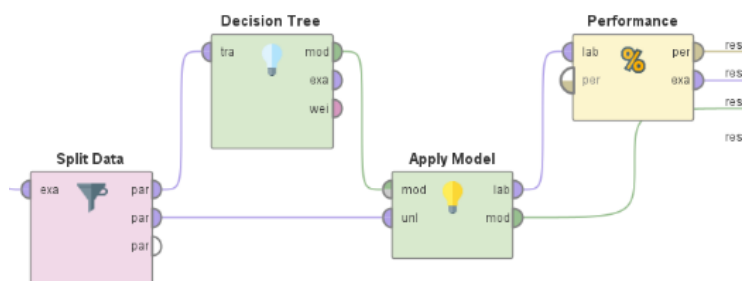
Gambar 8. Implementasi Algoritma Decision Tree

Dalam operator *split data*, digunakan ratio perbandingan dengan nilai 80% untuk pengujian data training dan 20% untuk data testing. Data training digunakan sebagai data latih yang akan digunakan untuk pengujian dengan data testing.

3.4.2 Support Vector Machine

SVM (*Support Vector Machine*) adalah sebuah metode klasifikasi yang dapat mengenali pola dari data yang diberikan. Teknik *Support Vector Machine* (SVM) bertujuan untuk menemukan fungsi pemisah terbaik di antara fungsi yang ada untuk memisahkan dua macam obyek[19]. Dalam penelitian ini dilakukan proses klasifikasi menggunakan algoritma *Support Vector Machine* yang dapat dilihat pada gambar 9 dibawah ini.

Pada proses ini, menggunakan operator diantaranya “*read csv*” untuk membaca file dataset dengan jenis csv, selanjutnya menggunakan operator *set role*. Operator ini digunakan untuk memberikan keterangan pada atribut. Untuk *set role* ini digunakan atribut *label* sesuai dengan nama pada dataset yang digunakan dan *target role* dipilih sebagai *label*. Selain *set role* juga ditambahkan operator *nominal to text*, *process documents from data* serta *cross validation* untuk mendapatkan hasil *confusion matrix*.



Gambar 9. Proses klasifikasi algoritma svm

3.5 Confusion Matrix

Dalam tahapan ini model algoritma yang telah digunakan pada metode klasifikasi dilakukan perhitungan performa model yakni pada algoritma *Decision Tree* dan *Support Vector Machine* (SVM). Dan nilai matrix yang dihasilkan dari setiap algoritma akan dilakukan perbandingan dengan tujuan untuk menentukan algoritma yang lebih baik. Perhitungan performa model klasifikasi didasarkan pada pengujian objek yang benar dan objek yang salah[20]. Tabel dibawah ini menunjukkan hasil model klasifikasi menggunakan metode algoritma *Decision Tree* dan *Support Vector Machine* (SVM). Tabel 8, berisikan hasil *confusion matrix* dari hasil penggunaan algoritma *decision tree*.

Tabel 8. *confusion matrix decision tree*

Performa	True Tidak Puas	True Sangat Puas	Class Precision
Pred. Tidak Puas	25	6	80.65%
Pred. Sangat Puas	11	30	73.17%
Class Recall	69.44%	83.33%	

Tabel 9, memberikan hasil *confusion matrix* dalam penggunaan algoritma SVM setelah pengolahan data.

Tabel 9. *Confusion matrix support vector machine*

Performa	True Tidak Puas	True Sangat Puas	Class Precision
Pred. Tidak Puas	7	2	77.78%
Pred. Sangat Puas	174	178	50.57%
Class Recall	3.87%	98.89%	

Pada tabel 10 menunjukkan perbandingan performa dari dua algoritma klasifikasi yang digunakan pada dataset.

Tabel 10. Perbandingan performa dari 2 model algoritma

Algoritma	Accuracy	Precision	Recall
Decision Tree	76.39%	80.65%	69.44%
Support Vector Machine	51.25%	77.78%	3.87%

Berdasarkan pada tabel perbandingan *confusion matrix* dari dua algoritma klasifikasi menunjukkan bahwa algoritma *Decision Tree* merupakan algoritma yang lebih baik dibanding dengan algoritma SVM.

4. KESIMPULAN

Berdasarkan analisis terhadap dataset rating pengguna aplikasi Migo E-Bike, dapat disimpulkan performa dari dua algoritma klasifikasi yang digunakan, yaitu Decision Tree dan Support Vector Machine (SVM). Dataset ini terdiri dari dua kategori utama: "tidak puas" untuk skor 1 dan "sangat puas" untuk skor 5. Dari analisis performa model klasifikasi, ditemukan bahwa algoritma Decision Tree menunjukkan kinerja yang lebih baik dibandingkan dengan SVM dalam mengklasifikasikan kepuasan pengguna aplikasi Migo E-Bike. Pertama, hasil klasifikasi menggunakan algoritma Decision Tree menunjukkan performa yang cukup baik. Berdasarkan confusion matrix yang dihasilkan, Decision Tree memiliki precision untuk kategori "tidak puas" sebesar 80,65% dan recall sebesar 69,44%. Untuk kategori "sangat puas", precision mencapai 73,17% dan recall sebesar 83,33%. Secara keseluruhan, Decision Tree mencapai tingkat akurasi sebesar 76,39%. Sebaliknya, hasil klasifikasi menggunakan algoritma SVM menunjukkan performa yang kurang memuaskan. Precision untuk kategori "tidak puas" tercatat sebesar 77,78%, namun recall-nya sangat rendah, yaitu hanya 3,87%. Untuk kategori "sangat puas", precision hanya sebesar 50,57% meskipun recall-nya sangat tinggi, yaitu 98,89%. Akurasi keseluruhan untuk SVM hanya mencapai 51,25%. Dari perbandingan ini, terlihat jelas bahwa algoritma Decision Tree lebih unggul dalam menangani dataset rating pengguna Migo E-Bike dibandingkan dengan SVM. Decision Tree mampu memberikan hasil yang lebih konsisten dan seimbang antara precision dan recall untuk kedua kategori. Sebaliknya, SVM menunjukkan ketidakseimbangan yang signifikan dengan recall yang sangat rendah untuk kategori "tidak puas", meskipun recall untuk kategori "sangat puas" sangat tinggi.

REFERENCES

- [1] M. Asti, I. Supriyadi, and P. Yusgiantoro, "Analisa Penggunaan Sepeda Motor Listrik Bagi Transportasi Online Terhadap Ketahanan Energi (Studi Pada Gojek)," *J. Ketahanan Energi*, vol. 6, no. 1, pp. 19–38, 2020, [Online]. Available: <https://jurnalprodi.idu.ac.id/index.php/KE/article/view/494>
- [2] S. Wulandari and N. Marlina, "Dengan Kepuasan Konsumen Sebagai Variabel Intervening Pada Pengguna Transportasi Migo Di Surabaya," *J. Pendidik. Tata Niaga*, vol. 8, no. 2, pp. 791–797, 2020.
- [3] D. Gunawan, M. Suyanto, and H. Henderi, "Pengukuran Kepuasan Pengguna Aplikasi Secure System Of Payment (SSP) Menggunakan Metode End-User Computing Satisfaction (EUCS)," *Respati*, vol. 15, no. 1, p. 50, 2020, doi: 10.35842/jtir.v15i1.331.
- [4] I. Nurul Hassanah, S. Faisal, A. Mutoi Siregar, U. Buana Perjuangan Karawang Ji HSRonggo Waluyo, T. Timur, and J. Barat, "Perbandingan Algoritma Support Vector Machine Dengan Decision Tree Pada Aplikasi Ruang Guru," *Kumpul. J. Ilmu Komput.*, vol. 10, no. 1, pp. 39–50, 2023.
- [5] Y. A. Singgalen, "Analisis Sentimen Top 10 Traveler Ranked Hotel di Kota Makassar Menggunakan Algoritma Decision Tree dan Support Vector Machine," *Media Online*, vol. 4, no. 1, pp. 323–332, 2023, doi: 10.30865/klik.v4i1.1153.
- [6] K. S. H. Kusuma Al Atros, A. R. Padri, O. Nurdiawan, A. Faqih, and S. Anwar, "Model Klasifikasi Analisis Kepuasan Pengguna Perpustakaan Online Menggunakan K-Means dan Decision Tree," *JURIKOM (Jurnal Ris. Komputer)*, vol. 8, no. 6, p. 323, 2021, doi: 10.30865/jurikom.v8i6.3680.
- [7] A. M. Siregar, S. Faisal, A. Puspabhuana, and M. S. H. Simarankir, "Comparison Study Of Term Weighting Optimally With SVM In Sentiment Analysis," 2019, doi: 10.4108/eai.18-7-2019.2288508.
- [8] J. Jtik, J. Teknologi, T. W. Putra, and A. Triayudi, "Analisis Sentimen Pembelajaran Daring menggunakan Metode Naïve



- Bayes , KNN , dan Decision Tree,” vol. 6, no. 1, 2022.
- [9] M. A. C, “KLASIFIKASI DIAGNOSA PENYAKIT DIABETES MELLITUS (DM),” no. Dm.
- [10] M. I. Fariz, D. Arifianto, and Y. D. Rahayu, “Optimasi Metode Multinomial Naïve Bayes Dengan Menggunakan Metode Levenshtein Distance Pada Ulasan Aplikasi KAI Access,” vol. 5, no. 1, pp. 19–27, 2023.
- [11] M. Izunnahdi, G. Aburrahman, and A. E. Wardoyo, “Jurnal Smart Teknologi Sentimen Analisis Pada Data Ulasan Aplikasi KAI Access Di Google PlayStore Menggunakan Metode Multinomial Naive Bayes Sentiment Analysis on KAI Access Application Review Data on Google PlayStore Using Multinomial Naive Bayes Method Jurnal Smart Teknologi,” vol. 4, no. 2, pp. 192–198, 2023.
- [12] H. Ashari *et al.*, “PERBANDINGAN KINERJA ALGORITMA MULTINOMIAL NAÏVE BAYES (MNB), MULTIVARIATE BERNOULLI DAN ROCCHIO ALGORITHM DALAM KLASIFIKASI KONTEN BERITA HOAX BERBAHASA INDONESIA PADA MEDIA SOSIAL,” pp. 1–12. 2023
- [13] F. R. Irawan *et al.*, “ANALISIS SENTIMEN TERHADAP PENGGUNA GOJEK MENGGUNAKAN METODE K-NEAREST NEIGHBORS SENTIMENT ANALYSIS OF GOJEK USERS USING K-NEAREST NEIGHBOR,” vol. 5, no. 1, pp. 62–68, 2022, doi: 10.33387/jjiko.
- [14] D. Sandi, E. Utami, M. T. Informatika, and U. Amikom, “Analisis Sentimen Publik Terhadap Elektabilitas Ganjar Pranowo di Tahun Politik 2024 di Twitter dengan Algoritma KNN dan Naïve Bayes,” vol. 7, pp. 1097–1108, 2024, doi: 10.30865/mib.v7i3.6298.
- [15] R. Sistem, “Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam,” vol. 1, no. 10, 2021.
- [16] I. Of, S. Vector, M. For, A. Comments, and D. Signature, “IMPLEMENTASI SUPPORT VECTOR MACHINE UNTUK ANALISIS SENTIMEN KOMENTAR APLIKASI TANDA TANGAN DIGITAL IMPLEMENTATION OF SUPPORT VECTOR MACHINE FOR SENTIMENT,” vol. 15, no. 1, pp. 10–19, 2022.
- [17] P. A. D. A. G. Oogle and P. L. A. Y. S. Tore, “T EXT M INING S ENTIMEN A NALISIS P ENGGUNA A PLIKASI M ARKETPLACE T OKOPEDIA B ERDASAR R ATING D AN K OMENTAR,” pp. 33–40, 2022.
- [18] J. J. Purnama *et al.*, “KLASIFIKASI MAHASISWA HER BERBASIS ALGORTIMA SVM DAN DECISION CLASSIFICATION OF HER STUDENTS BASED ON SVM ALGORITHM AND,” vol. 7, no. 6, pp. 1253–1260, 2020, doi: 10.25126/jtiik.202073080.
- [19] N. Hendrastuty *et al.*, “Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja Pada Twitter Dengan Metode Support Vector Machine,” vol. 6, no. 3, pp. 150–155, 2021.
- [20] L. M. Sotarjua and D. B. Santoso, “PERBANDINGAN ALGORITMA KNN , DECISION TREE ,* DAN RANDOM * FOREST PADA DATA IMBALANCED CLASS UNTUK KLASIFIKASI PROMOSI KARYAWAN,” vol. 7, pp. 192–200, 2022.