Perbandingan Hasil Penerapan Algoritma Klasifikasi dan *Natural Language Processing* Terhadap Data Kepuasan Pengguna Layanan Transportasi Umum MRT Jakarta

Muhammad Nabil Nufail Pribadi[1], Iin Ernawati[2]
Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jakarta
Jl. Rs. Fatmawati, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12450, Indonesia
email: muhammadnnp@upnvj.ac.id[1], iinernawati@upnvj.ac.id[2]

Abstrak. MRT Jakarta merupakan salah satu upaya pemerintah DKI Jakarta untuk mengatasi kemacetan. Selama masa operasinya, MRT Jakarta memberikan berbagai macam kesan bagi penggunanya. Oleh karena itu, dilakukan penelitian klasifikasi terhadap kepuasan pengguna layanan transportasi umum MRT Jakarta untuk mengetahui alasan yang menyebabkan masyarakat bersedia atau enggan memilih untuk memanfaatkan sarana transportasi MRT Jakarta, melalui media sosial X, memanfaatkan natural language processing dan algoritma Support Vector Machine, algoritma Random Forest Classifier, serta algoritma Logistic Regression multinomial. Data berjumlah sebanyak 525 post, dengan kategori 'positif' sebanyak 222 data, kategori 'negatif' sebanyak 185 data, dan kategori 'netral' sebanyak 118 data. Dengan pembagian dataset berdasarkan perbandingan 80:20, model klasifikasi dengan hasil paling akurat pada penelitian ini, dengan algoritma Random Forest Classifier, menggunakan parameter terbaik yang diperoleh melalui teknik hyperparameter tuning, dengan nilai class_weight='balanced', max_depth=350, min_samples_split=5, serta n_estimators=200, menghasilkan nilai akurasi sebesar 81%, serta berhasil secara akurat memprediksi seluruh sampel data baru berdasarkan target kelasnya.
Kata kunci: *Natural Language Processing*, Klasifikasi, MRT Jakarta

1 Pendahuluan

Seiring dengan bertambahnya jumlah penduduk dan terbatasnya lahan untuk tempat tinggal di DKI Jakarta, sudah tidak memungkinkan bagi setiap masyarakat untuk memiliki transportasi pribadi masing-masing. Hal ini dapat dibuktikan dengan kemacetan pada DKI Jakarta yang sering ditemukan pada kehidupan sehari-hari masyarakat Jakarta, dengan penyebab terbesar berasal dari transportasi pribadi yang terdiri dari kendaraan roda empat dan roda dua pada lalu lintas, serta infrastruktur yang sudah tidak memungkinkan lagi untuk dikembangkan atau diperluas (Sitanggang and Saribanon, 2018).

MRT Jakarta merupakan salah satu solusi pemerintah DKI Jakarta dalam upaya mengatasi kemacetan serta menambahkan variasi moda transportasi yang efisien dengan biaya yang mudah dijangkau. MRT Jakarta dibuka pada 24 Maret tahun 2019, dengan stasiun yang beroperasional berjumlah 13 pada satu jalur. Semenjak pandemi COVID-19, pengguna MRT Jakarta meningkat dari tahun ke tahun, menurut direktur utama PT MRT Jakarta, dengan jumlah rata-rata penumpang per harinya pada bulan Maret 2023 sebanyak 92 ribu penumpang, yang merupakan kenaikan dari rata-rata pada bulan Februari 2023 sebanyak 85 ribu penumpang. Dengan peranan sarana transportasi umum yang memadai, peringkat Jakarta turun dari urutan ke-31 menjadi urutan ke-46 dari 404 kota yang diukur, dengan tingkat kemacetan tertinggi di dunia, dengan indeks kemacetan yang menurun dari 36% di tahun 2020 menjadi 34% di tahun 2021, menurut Tomtom Traffic Index 2021.

Namun pada tahun 2022, posisi Indonesia kembali naik menjadi urutan ke-29 menurut Tomtom Traffic Index bulan Februari pada tahun 2022. Penyebab utamanya adalah kehidupan masyarakat yang kembali normal setelah pandemi COVID-19, yang menyebabkan bertambahnya kendaraan pribadi pada lalu lintas. Untuk itu, diperlukan analisis terhadap opini masyarakat yang kemudian dapat digunakan untuk mengevaluasi hasil dari kinerja sarana transportasi publik yang sudah beroperasi, salah satunya

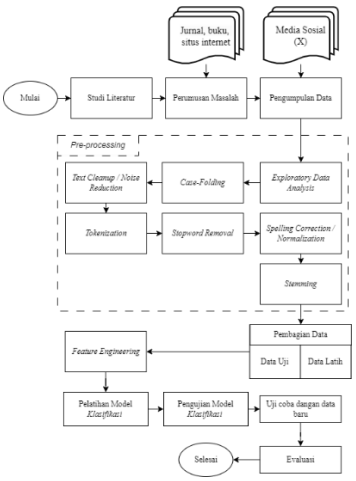
mengenai MRT Jakarta. Data opini yang dikumpulkan terdiri dari opini positif dan opini negatif (Kurniawan *et al.*, 2023). Hal ini dapat digunakan untuk mencari kelebihan yang harus dipertahankan, serta kekurangan yang dapat diperbaiki, untuk menarik lebih banyak pengguna pada sarana transportasi umum MRT Jakarta.

Untuk itu, penulis akan melakukan penelitian klasifikasi kepuasan melalui data post berupa opini masyarakat pengguna MRT Jakarta yang meliputi opini positif, opini negatif, serta opini netral melalui sosial media X. Kumpulan data opini tersebut akan dilakukan pembersihan melalui teknik praproses dan diolah sesuai dengan kaidah pemanfaatan NLP atau natural language processing, penentuan label positif, negatif, dan netral, kemudian digunakan model *machine learning* menggunakan algoritma *support vector machine*, algoritma *Random Forest Classifier*, serta algoritma *Multinomial Logistic Regression*, yang kemudian akan dilakukan analisis terhadap luaran dan evaluasi confusion matrix dan uji coba terhadap sampel data baru pada ketiga model klasifikasi tersebut untuk menentukan model paling kompatibel yang diimplementasikan pada kasus klasifikasi dengan data *post* media sosial X mengenai kepuasan masyarakat terhadap transportasi umum MRT Jakarta.

Penelitian terdahulu pada jurnal pertama yang dilakukan terhadap *review* aplikasi transportasi online menggunakan metode *Support Vector Machine* dan *Decision Tree*, diperoleh hasil paling akurat dengan metode SVM dengan nilai akurasi sebesar 90.20% pada nilai k-fold 3 (Rokhman, Berlilana and Arsi, 2021). Pada jurnal kedua dilakukan penelitian terhadap data media sosial Twitter mengenai transportasi umum CommuterLine menggunakan metode *Support Vector Machine, Support Vector Machine* dengan *Particle Swarm Optimization, Naïve Bayes, dan Naïve Bayes* dengan *Adaboost*, diperoleh hasil paling akurat dengan metode SVM, diperoleh nilai akurasi sebesar 78.15%, yang kemudian memperoleh nilai akurasi yang lebih besar jika ditambahkan dengan PSO, yaitu sebesar 79.47% (Novaneliza *et al.*, 2023). Selanjutnya pada jurnal ketiga dilakukan penelitian terhadap *review* aplikasi Gojek dengan *Naïve Bayes* dan *Support Vector Machine*, diperoleh hasil paling akurat dengan metode SVM dengan rata-rata nilai akurasi sebesar 99.43% (Salsabilah Ramadinah, 2021). Kemudian pada jurnal keempat dilakukan penelitian terhadap *review* pelayanan transportasi GoRide pada Twitter menggunakan metode *Naïve Bayes* dan *Support Vector Machine*, yang di mana hasil paling akurat diperoleh menggunakan algoritma SVM dengan nilai akurasi sebesar 81,48% dan nilai AUC 0.832 (Utari Maharani, 2023). Terakhir, pada jurnal kelima dilakukan penelitian terhadap *review* KAI Access menggunakan *Naïve Bayes* dan *Support Vector Machine*, model terbaik dari dihasilkan algoritma SVM dengan *hyperparameter tuning* dengan rata-rata skor akurasi 91,63% (Mustakim and Priyanta, 2022).

SVM merupakan metode yang menghasilkan luaran terbaik dari penelitian-penelitian tersebut. Sehingga penulis memutuskan untuk membandingkan algoritma *support vector machine* dengan algoritma klasifikasi lainnya seperti *Random Forest Classifier* dan *Logistic Regression* untuk penelitian kali ini.

2 Metodologi Penelitian



Gambar 1. Tahapan Metode Penelitian

2.1 Studi Literatur

Sebelum memulai proses inti dari penelitian, penulis melakukan pembelajaran terhadap penelitian-penelitian sebelumnya. Berdasarkan hasil penelitian dari jurnal yang penulis bandingkan, penulis memutuskan untuk menggunakan algoritma support vector machine untuk penelitian ini, serta membandingkannya dengan algoritma klasifikasi lain seperti Logistic Regression, Random Forest Classifier. Selain itu, dilakukan pembelajaran terhadap artikel ilmiah yang memuat teori mengenai komponen yang diteliti, serta situs internet resmi yang menyediakan pengertian dan cara pakai dari metode yang akan digunakan dalam cakupan penelitian ini.

2.2 Perumusan Masalah

Tahapan identifikasi dan perumusan masalah dilakukan untuk menemukan permasalahan yang terdapat dalam lingkup penelitian, sebagai landasan utama atau penegas dilakukannya penelitian. Berdasarkan permasalahan yang penulis temukan, yaitu mengenai bertambahnya jumlah penduduk dan terbatasnya lahan untuk tempat tinggal di DKI Jakarta, yang menyebabkan kemacetan pada DKI Jakarta dengan penyebab terbesar berasal dari transportasi pribadi.

Selain itu, terdapat beberapa algoritma klasifikasi yang digunakan terhadap data berupa teks dengan tata bahasa (Hasydna and Dinata, 2020). Penulis memutuskan untuk melakukan penelitian terhadap klasifikasi opini masyarakat untuk mengetahui alasan masyarakat memilih atau enggan memilih untuk memanfaatkan sarana transportasi umum, khususnya MRT Jakarta, serta untuk mengetahui algoritma yang memiliki hasil terbaik dalam proses klasifikasi terhadap data yang telah penulis kumpulkan.

2.3 Pengumpulan Data

Untuk mengumpulkan data post yang diunggah oleh masyarakat mengenai transportasi umum MRT Jakarta, penulis memanfaatkan media sosial X atau Twitter, menggunakan fitur advanced search yang disediakan, dalam rentang waktu bulan Oktober tahun 2023, sampai dengan bulan Februari tahun 2024. Adapun format data yang akan diperoleh dari pengumpulan ini adalah bentuk csv.

2.4 Pre-processing

Tahap pre-processing dilakukan untuk mempersiapkan data sebelum diproses pada tahap selanjutnya (Mustikananda, Ratnawati and Rahayudi, 2022). Hal ini dilakukan sebab data yang diperoleh bersifat tidak konsisten, dikarenakan gaya penulisan dari setiap responden yang berbeda (Raharjo, Sunarya and Divayana, 2022). Berikut ini tahap pre-processing yang akan dilakukan:

2.4.1 Exploratory Data Analysis

Pada tahap EDA, penulis akan melakukan visualisasi data untuk mempelajari data yang diperoleh secara lebih mendalam, sehingga penulis dapat lebih memahami data yang akan diolah. Pada tahap ini biasanya ditemukan hambatan yang dapat mengganggu proses klasifikasi, atau membuat hasil prediksi menjadi tidak maksimal, seperti imbalance atau ketidakseimbangan data, duplikasi data dan missing value (Larasati, Ratnawati and Hanggara, 2022).

2.4.2 Case-Folding

Pada tahap case folding dilakukan proses pengubahan huruf kapital menjadi bentuk kecilnya terhadap seluruh karakter alfabetik pada teks dataset (Raharjo, Sunarya and Divayana, 2022). Hal ini bertujuan agar kata yang memiliki bentuk pengejaan yang sama, namun memiliki ukuran huruf yang berbeda, tidak diidentifikasi sebagai kata lain atau memiliki makna yang berbeda.

2.4.3 Noise Reduction/Text Cleanup

Tahap ini dilakukan untuk menghilangkan tanda baca, simbol non-ASCII, elemen URL, angka atau karakter numerik, serta whitespace berlebih pada kalimat-kalimat dataset, dikarenakan karakter-karakter ini tidak relevan terhadap penentuan kategori teks (Munasatya and Novianto, 2020).

2.4.4 Tokenization

Tahap selanjutnya yaitu proses tokenizing. Pada tahap ini dilakukan pemecahan kalimat menjadi potongan kata-kata individual dalam bentuk "token", sehingga data dapat lebih mudah untuk dilakukan pengolahan terhadap masing-masing kata atau substring (Wicaksono, 2023). Proses ini akan memudahkan perhitungan frekuensi kemunculan kata dari tiap kata dalam bentuk token tersebut, serta penghapusan imbuhan yang dapat dimiliki pada suatu kata tersebut (Raharjo, Sunarya and Divayana, 2022).

2.4.5 Stopword Removal

Pada proses stopword removal, dilakukan penghapusan kata yang tidak relevan dengan kata-kata utama penentu kategori atau kelas data. Kata-kata tersebut dapat mengurangi performa model dalam mengategorikan sentimen, seperti kata depan, kata sambung, kata ganti, dan kata sifat (Wicaksono, 2023).

2.4.6 Normalization/Spelling Correction

Pada tahap ini dilakukan perbaikan terhadap pengejaan kata-kata yang ditulis secara tidak baku. Hal ini dilakukan untuk membuat dataset menjadi lebih konsisten, serta mempermudah proses stemming pada tahap selanjutnya.

2.4.7 Stemming

Tahap terakhir pada bagian pre-processing pada penelitian ini. Pada tahap ini dilakukan pengubahan setiap kata yang memiliki imbuhan menjadi bentuk dasarnya, sehingga dapat dengan mudah dihitung frekuensi kemunculannya untuk tahap selanjutnya (Wicaksono, 2023).

2.5 Pembagian Data

Pada tahap ini dilakukan pembagian dataset utama menjadi data latih dan data uji dengan persentase jumlah 80% untuk data latih, dan 20% untuk data uji berdasarkan kumpulan penelitian terdahulu yang menandakan perbandingan proporsi pembagian dataset terbaik. Data latih digunakan untuk melatih model klasifikasi pada tahap selanjutnya, kemudian data uji digunakan sebagai pengujian performa dari model klasifikasi tersebut.

2.6 Feature Engineering

Pada tahap ini dilakukan transformasi data melalui pembuatan word vector, yaitu dengan representasi fitur teks dari dataset ke dalam bentuk vektor, serta dilakukan pembobotan kata dengan menggunakan TF-IDF dengan menghitung nilai *term frequency* (TF) atau nilai kemunculan kata pada dokumen, selanjutnya dilakukan pencarian inverse document frequency atau IDF, kemudian dilakukan perhitungan terhadap *term frequency inverse document* atau TF-IDF sehingga dapat diproses pada model klasifikasi di tahap selanjutnya (Mustikananda, Ratnawati and Rahayudi, 2022).

2.7 Perancangan Model Klasifikasi

Setelah data sudah dilakukan pre-processing, pembobotan, dan pembagian, dilakukan perancangan model klasifikasi terhadap data post pengguna MRT Jakarta menggunakan algoritma *support vector machine*, *logistic regression multinomial*, dan *random forest classifier* yang diperoleh melalui *library* Scikit-Learn. Model tersebut akan dilatih menggunakan perbandingan data latih dan data uji sebanyak 70:30 dan 80:20 dari total data. Adapun parameter yang digunakan pada setiap model akan diperoleh melalui teknik hyperparameter tuning untuk memperoleh parameter terbaik dari segala kemungkinan parameter yang disediakan dalam dictionary.

2.8 Uji Coba Dengan Data Baru

Pada tahap ini, dilakukan pengujian terhadap semua model klasifikasi yang telah dirancang, menggunakan sampel data baru yang juga dibersihkan dengan teknik pra-proses data seperti yang dijelaskan sebelumnya.

2.9 Evaluasi

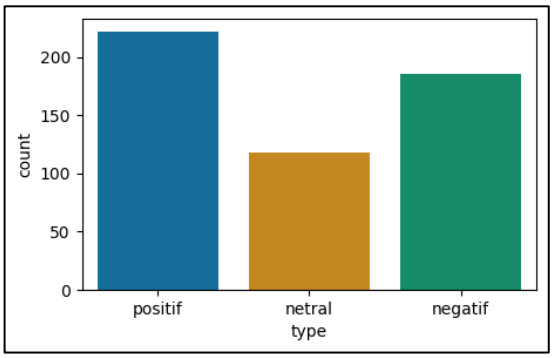
Tahap terakhir yang dilakukan adalah melakukan evaluasi terhadap model klasifikasi support vector machine, logistic regression, dan random forest classifier. Proses evaluasi dilakukan melalui analisis terhadap perbandingan confusion matrix yang dihasilkan oleh ketiga model klasifikasi. Melalui confusion matrix atau classification report, dapat diperoleh nilai akurasi, precision, recall, dan F1-Score yang menentukan performa dari model klasifikasi (Iskandar and Nataliani, 2021).

3 Hasil Dan Pembahasan

Data yang digunakan untuk penelitian ini diperoleh dari media sosial X dengan memanfaatkan fitur *Advanced search* dengan kata kunci MRT dan filter bahasa Indonesia. Data yang diperoleh berjumlah sebanyak 525 *posts* yang berada di dalam rentang waktu 5 bulan, yaitu dari bulan Oktober 2023 sampai dengan bulan Februari 2024. Adapun bentuk data yang dikumpulkan berupa *post* pengguna media sosial X mengenai MRT Jakarta dengan sentimen yang bervariasi, yang penulis masukkan ke dalam kategori positif, negatif, dan netral secara manual satu per satu untuk setiap *post*. Dalam proses pengumpulan, penulis tidak memasukkan data yang dapat bersifat sensitif, seperti *username* dan *display name* dari pemilik *post* yang bersangkutan. Data-data tersebut disimpan di dalam file dengan format *comma-separated values* (.csv).

3.1 Pre-processing

Tahapan yang dilakukan untuk mempersiapkan data sebelum dilakukan proses klasifikasi. Membersihkan data dari aspek-aspek yang tidak dibutuhkan dalam proses klasifikasi, namun dapat mempengaruhi performa atau hasil akhir model jika dibiarkan, dengan menggunakan *library* python.



Gambar 2. Visualisasi Persebaran Jumlah Data

Selisih jumlah dari setiap kategori data tersebut menunjukkan terdapatnya *data imbalance*, namun tidak ekstrim karena rasio perbedaan jumlah tidak terlalu besar, sehingga tidak perlu dilakukan undersampling maupun oversampling.

Selanjutnya data akan dibersihkan dengan *Case-Folding*, *Text Cleanup*, *Tokenization*, *Stopword Removal*, *Normalization*, dan *Stemming*. Adapun perbedaan sebelum dan sesudah proses-proses tersebut ditunjukkan pada tabel 1 di bawah ini.

Tabel 1. Perbandingan sebelum dan sesudah teknik *Pre-processing* pada data

Sebelum teknik Pre-processing	Setelah teknik Pre-processing
So far, MRT itu transportasi umum yang bagus sama paling bisa diandelin. Minim error karena lintasan eksklusif sendiri juga. Buat cewek, malem-malem pulang sendiri pun gak takut. Jadwalnya banyak, kereta bagus, aman hingga di luar stasiun pun nyaman, keren pokoknya.	['mrt', 'transportasi', 'bagus', 'andal', 'minim', 'error', 'lintas', 'eksklusif', 'cewek', 'malam', 'malam', 'pulang', 'takut', 'jadwal', 'kereta', 'bagus', 'aman', 'stasiun', 'nyaman', 'keren', 'pokok']
Kerasa bgt sih kenyamanan pelayanannya, apalagi sebagai pengguna moda	['rasa', 'sangat', 'nyaman', 'layan', 'guna', 'moda', 'transportasi', 'publik', 'malas',

transportasi publik yg tadinya males”an buat explore Jakarta kl weekend tapi skrg jadi semangat kl dijakin muterin Jakarta ver low budget by tj & mrt. Thank you for making Jakarta more beautiful and colorful	'explore', 'jakarta', 'weekend', 'sekarang', 'semangat', 'ajak', 'putar', 'jakarta', 'low', 'budget', 'tj', 'mrt', 'thank', 'you', 'making', 'jakarta', 'more', 'beautiful', 'colorful']
BINGUNG BGT KOK MALAH GA BISA DIPAKE LAGI?!?!?! kemunduran yg signifikan.. Pdhl kartu MTT enak sat set gak bikin repot	['bingung', 'sangat', 'pakai', 'mundur', 'signifikan', 'padahal', 'kartu', 'mtt', 'enak', 'repot']

3.2 Pembagian Data

Data akan dibagi menjadi data latih dan data uji dengan perbandingan jumlah sebesar 70:30 dan 80:20, dengan 80% dari total dataset untuk data latih, dan 20% dari total dataset digunakan untuk data uji. Proses ini memanfaatkan *library* scikit-*learn* dengan menggunakan fungsi “train_test_split()”. Berikut ini merupakan dimensi atau *shape* data yang telah dilakukan pembagian yang diuraikan pada tabel 2 di bawah ini.

Tabel 2. Dimensi data setelah proses pembagian

ratio	train data	test data
80:20	420	105
70:30	367	158

3.3 Feature Engineering

Data yang telah dilakukan pembagian akan divektorisasikan menggunakan teknik *Term Frequency-Inverse Document Frequency*. Teknik ini dilakukan untuk tingkat kepentingan atau bobot suatu kata dalam satu teks atau dokumen terhadap kata-kata lainnya yang juga memiliki kemunculan pada dokumen.

Berikut ini merupakan contoh perhitungan *TF-IDF*. Untuk percobaan ini, akan digunakan tiga pratinjau data yang sudah dibersihkan dari tahap *pre-processing* sebelumnya yang ditunjukkan pada tabel 3.

Tabel 3. Sampel data untuk perhitungan teknik *TF-IDF*

Doc	*sample*	*len()*	*type*
1	['mrt', 'transportasi', 'bagus', 'andal', 'minim', 'error', 'lintas', 'eksklusif', 'cewek', 'malam', 'malam', 'pulang', 'takut', 'jadwal', 'kereta', 'bagus', 'aman', 'stasiun', 'nyaman', 'keren', 'pokok']	21	positif
2	['rasa', 'sangat', 'nyaman', 'layan', 'guna', 'moda', 'transportasi', 'publik', 'malas', 'explore', 'jakarta', 'weekend', 'sekarang', 'semangat', 'ajak', 'putar', 'jakarta', 'low', 'budget', 'tj', 'mrt', 'thank', 'you', 'making', 'jakarta', 'more', 'beautiful', 'colorful']	28	positif
3	['bingung', 'sangat', 'pakai', 'mundur', 'signifikan', 'padahal', 'kartu', 'mtt', 'enak', 'repot']	10	negatif

Berikut ini merupakan perhitungan *TF-IDF* terhadap 3 sampel data dengan rumus yang telah dijelaskan sebelumnya yang ditunjukkan pada tabel 4.

Tabel 4. Perhitungan teknik TF-IDF terhadap tiga sampel data

Term	*Term Occurrences*			*Term Frequency (TF)*			**df**	**IDF**	*Term Weight (TF-IDF)*		
	Document			*Document*					*Document*		
	1	**2**	**3**	**1**	**2**	**3**			**1**	**2**	**3**
ajak	0	1	0	0	0.036	0	1	0.477	0	0.017	0
aman	1	0	0	0.048	0	0	1	0.477	0.023	0	0
andal	1	0	0	0.048	0	0	1	0.477	0.023	0	0
bagus	2	0	0	0.095	0	0	1	0.477	0.045	0	0
					…						
transportasi	1	1	0	0.048	0.036	0	2	0.176	0.008	0.006	0

3.4 Perancangan Model Klasifikasi

3.4.1 Model *Support Vector Machine*

Model klasifikasi pertama pada penelitian ini menggunakan algoritma *support vector machine*. SVM diperoleh dari *library scikit-learn* melalui *class* "SVC". Untuk parameter yang akan diujicobakan di antaranya adalah jenis *kernel*, nilai *cost*, serta nilai *gamma*. Selanjutnya akan dilakukan *hyperparameter tuning* untuk menentukan parameter terbaik yang dapat memaksimalkan hasil dari model klasifikasi SVM, dengan menggunakan "*GridSearchCV*" dari *library scikit-learn*.

Berdasarkan metode tersebut, *grid search* mampu menemukan parameter yang mampu menghasilkan luaran terbaik untuk klasifikasi terhadap data yang penulis kumpulkan berdasarkan rasio pembagian datanya. Adapun parameter tersebut adalah sebagai berikut.

Tabel 5. Parameter model klasifikasi algoritma SVM

parameter	*value*	
	Pembagian Data	
	70:30	**80:20**
C	1	
gamma	'scale'	
kernel	'linear'	
	SVC(C=1, kernel='linear')	

3.4.2 Model *Random Forest Classifier*

Algoritma ini diperoleh dari modul *ensemble* pada *library scikit-learn*, yaitu "RandomForestClassifier". Parameter yang akan dilakukan pengujian terdiri dari *max_depth*, *min_samples_leaf*, *min_samples_split*, *n_estimators*, serta *class_weight* dengan nilai "balanced". Selain *class_weight*, parameter lainnya akan dilakukan pencarian nilai terbaik untuk hasil klasifikasi yang lebih akurat dengan teknik *hyperparameter tuning* menggunakan *grid search*.

Melalui metode tersebut, telah ditentukan nilai parameter terbaik yang memberikan hasil klasifikasi paling akurat untuk masing-masing perbandingan pembagian dataset, yang tertulis pada tabel 6 di bawah ini.

Tabel 6. Parameter model klasifikasi algoritma *Random Forest Classifier*

parameter	value	
	Pembagian Data	
	70:30	80:20
class_weight	'balanced'	'balanced'
max_depth	550	350
min_samples_leaf	1	1
min_samples_split	5	5
n_estimators	200	200
	RandomForestClassifier(class_weight='balanced', max_depth=550, min_samples_split=5, n_estimators=200)	RandomForestClassifier(class_weight='balanced', max_depth=350, min_samples_split=5, n_estimators=200)

3.4.3 Model *Logistic Regression*

Logistic Regression diperoleh dari modul *linear_model* dari *library scikit-learn* dan digunakan dengan mendeklarasikan objek "LogisticRegression". Adapun parameter yang digunakan pada model ini akan dilakukan pencarian nilai terbaiknya untuk hasil model klasifikasi yang lebih optimal dengan teknik *hyperparameter tuning*. Parameter tersebut terdiri dari nilai *hyperparameter* (*C*) dan penalty.

Dengan teknik *hyperparameter tuning*, nilai terbaik dari kedua parameter tersebut berdasarkan daftar nilai parameter pada tabel di atas sudah berhasil ditemukan. Adapun nilai-nilai tersebut dicantumkan pada tabel 7 di bawah ini.

Tabel 7. Parameter model klasifikasi algoritma *Logistic Regression*

parameter	value	
	Pembagian Data	
	70:30	80:20
C	1.6238	
penalty	'l2'	
solver	'lbfgs'	
LogisticRegression(C=1.6238, penalty='l2')		

3.5 Uji Coba Dengan Data Baru

Dilakukan uji coba ketiga model klasifikasi dengan menggunakan empat sampel data teks baru yang diperoleh dari *post* pengguna media sosial X. Berikut ini adalah hasil prediksi terhadap empat sampel data tersebut berdasarkan perbandingan pembagian data 70:30 dan 80:20 menggunakan ketiga model klasifikasi yang sebelumnya telah dirancang yang diuraikan pada tabel 8 berikut.

Tabel 8. Uji coba model klasifikasi dengan data baru

model	ratio	T_1	T_2	T_3	T_4
Support Vector Machine	70:30	netral	netral	negatif	positif
	80:20	netral	netral	negatif	positif
Random Forest Classifier	70:30	netral	negatif	negatif	netral
	80:20	netral	negatif	negatif	positif
Multinomial Logistic Regression	70:30	netral	negatif	negatif	positif
	80:20	netral	negatif	negatif	positif
Target		netral	negatif	negatif	positif

3.6 Evaluasi

Tabel berikut ini berisi perbandingan nilai *accuracy*, nilai *precision*, nilai *recall*, dan nilai *f1-score* dari ketiga model tersebut.

Tabel 9. Perbandingan hasil model klasifikasi

model	ratio	accuracy	precision	recall	f1-score
Support Vector Machine	70:30	0.759493	0.758068	0.759493	0.756846
	80:20	0.780952	0.778655	0.780952	0.777637
Random Forest Classifier	70:30	0.803797	0.812482	0.803797	0.806766
	80:20	0.809523	0.823733	0.809523	0.815236
Multinomial Logistic Regression	70:30	0.740506	0.726648	0.740506	0.727777
	80:20	0.742857	0.728415	0.742857	0.726166

Dikarenakan perbandingan atau *imbalance data* dari jumlah data ketiga kelas pada dataset memiliki rasio yang tidak tinggi, maka untuk nilai *accuracy*, nilai *precision*, nilai *recall*, dan nilai *f1-score* akan menggunakan *weighted average*. Berdasarkan perbandingan tersebut, algoritma yang memiliki hasil paling akurat untuk pemrosesan terhadap dataset yang penulis kumpulkan adalah model klasifikasi dengan algoritma *Random Forest Classifier* dengan perbandingan pembagian dataset sebesar 80:20. Hasil akurasi ini diperoleh dari banyaknya penggunaan pohon keputusan pada algoritma *Random Forest Classifier*, yang di mana semakin banyak pohon keputusan yang dibuat, semakin akurat pula hasil klasifikasinya dan dapat mencegah terjadinya *overfitting*, namun akan memakan waktu yang lebih lama yang disebabkan keperluan tenaga komputasi yang lebih besar.

4 Kesimpulan

4.1 Kesimpulan

Model klasifikasi dengan hasil paling akurat diperoleh melalui rasio pembagian dataset sebesar 80:20 dan dengan menggunakan algoritma *Random Forest Classifier*. Model klasifikasi *Support Vector Machine* dengan rasio pembagian dataset terbaik sebesar 80:20, dan parameter terbaik yang diperoleh menggunakan *hyperparameter tuning* metode *grid search*, yaitu nilai *cost* (*C*) = 1, nilai *gamma* = ‘scale’, dan penggunaan *kernel* ‘*linear*’, menghasilkan nilai *accuracy* sebesar 78%, nilai *precision* sebesar 77.8%, nilai *recall* sebesar

78%, dan nilai *f1-score* sebesar 77.7%, serta berhasil memprediksi 3 dari 4 sampel data baru berdasarkan target kelasnya. Untuk model *Logistic Regression* multinomial dengan rasio pembagian dataset terbaik sebesar 80:20, dan parameter terbaik menurut hasil *hyperparameter tuning*, yaitu nilai *hyperparameter* (*C*) = 1.6238 dan *penalty* L2, menghasilkan nilai *accuracy* sebesar 74%, nilai *precision* sebesar 72.8%, nilai *recall* sebesar 74%, dan nilai *f1-score* sebesar 72.6%, dan berhasil dengan sesuai memprediksi 4 dari 4 sampel data baru sesuai dengan target kelasnya. Model klasifikasi dengan nilai akurasi paling tinggi pada penelitian ini, yaitu model dengan algoritma *Random Forest Classifier*, menggunakan rasio pembagian dataset terbaik sebesar 80:20, dan parameter terbaik yang diperoleh melalui teknik *hyperparameter tuning* dengan nilai *class_weight='balanced'*, nilai *max_depth*=350, nilai *min_samples_split*=5, serta nilai *n_estimators*=200, menghasilkan nilai *accuracy* sebesar 81%, nilai *precision* sebesar 82.3%, nilai *recall* sebesar 81%, dan nilai *f1-score* sebesar 81.5%, serta berhasil dengan akurat memprediksi 4 dari 4 sampel data baru berdasarkan target kelasnya.

4.2 Saran

Menggunakan dataset dengan jumlah data lebih banyak dan dengan kalimat atau kata yang lebih bervariasi. Serta pelabelan dataset secara otomatis agar mempermudah pengolahan terhadap data dengan skala yang lebih besar. Mengotomatisasi proses normalisasi pengejaan atau spelling correction dengan metode tertentu, seperti metode Peter Norvig, N-Gram, serta dapat juga menggunakan library SymSpell untuk mempermudah pemrosesan data baru atau kalimat dengan kata yang lebih bervariasi.

Referensi

[1] N. Hasydna and R. K. Dinata, Machine Learning. Unimal Press, 2020.

[2] J. W. Iskandar and Y. Nataliani, "Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek," Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), vol. 5, no. 6, pp. 1120–1126, 2021, doi: 10.29207/resti.v5i6.3588.

[3] I. Kurniawan et al., "Perbandingan Algoritma Naive Bayes Dan SVM Dalam Sentimen Analisis Marketplace Pada Twitter," JATISI (Jurnal Teknik Informatika dan Sistem Informasi), vol. 10, no. 1, pp. 731–740, 2023. [Online]. Available: https://jurnal.mdp.ac.id/index.php/jatisi/article/view/3582

[4] F. A. Larasati, D. E. Ratnawati, and B. T. Hanggara, "Analisis Sentimen Ulasan Aplikasi Dana dengan Metode Random Forest," … Teknologi Informasi dan …, vol. 6, no. 9, pp. 4305–4313, 2022. [Online]. Available: http://j-ptiik.ub.ac.id

[5] N. Munasatya and S. Novianto, "Natural Language Processing untuk Sentimen Analisis Presiden Jokowi Menggunakan Multi Layer Perceptron," Techno.Com, vol. 19, no. 3, pp. 237–244, 2020, doi: 10.33633/tc.v19i3.3630.

[6] H. Mustakim and S. Priyanta, "Aspect-Based Sentiment Analysis of KAI Access Reviews Using NBC and SVM," IJCCS (Indonesian Journal of Computing and Cybernetics Systems), vol. 16, no. 2, p. 113, 2022, doi: 10.22146/ijccs.68903.

[7] D. Mustikananda, D. E. Ratnawati, and B. Rahayudi, "Perbandingan Algoritma Naïve Bayes dan Support Vector Machine untuk Analisis Sentimen terhadap Review Produk Aster Kosmetik Malang Marketplace Shopee," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 6, no. 7, pp. 3137–3144, 2022. [Online]. Available: http://j-ptiik.ub.ac.id

[8] R. Novaneliza et al., "Perbandingan Algoritma Untuk Analisis Sentimen Pada Twitter Transportasi Umum Commuterline," Jurnal Sains Komputer & Informatika (J-SAKTI), vol. 7, no. 1, pp. 13–21, 2023.

[9] R. A. Raharjo, I. M. G. Sunarya, and D. G. H. Divayana, "Perbandingan Metode Naïve Bayes Classifier Dan Support Vector Machine Pada Kasus Analisis Sentimen Terhadap Data Vaksin Covid-19 Di Twitter," Elkom: Jurnal Elektronika dan Komputer, vol. 15, no. 2, pp. 456–464, 2022, doi: 10.51903/elkom.v15i2.918.

[10] K. A. Rokhman, B. Berlilana, and P. Arsi, "Perbandingan Metode Support Vector Machine Dan Decision Tree Untuk Analisis Sentimen Review Komentar Pada Aplikasi Transportasi Online," Journal of Information System Management (JOISM), vol. 3, no. 1, pp. 1–7, 2021, doi: 10.24076/joism.2021v3i1.341.

[11] S. R. Ramadinah, Perbandingan Metode Klasifikasi Naïve Bayes Dan Support Vector Machine Pada Analisis Sentimen Review Gojek, 2021.

[12] R. Sitanggang and E. Saribanon, "Faktor-Faktor Penyebab Kemacetan Di DKI Jakarta," Jurnal Manajemen Bisnis Transportasi dan Logistik (JMBTL), vol. 4, no. 3, pp. 289–296, 2018.

[13] P. U. Maharani, "Perbandingan Algoritma Naive Bayes dan Support Vector Machine Dalam Analisis Sentimen Terhadap Pelayanan Goride Pada Twitter," pp. 31–41, 2023.

[14] H. Wicaksono, "Perbandingan Algoritma Machine Learning untuk Analisis Sentimen Berbasis Aspek pada Review Female Daily," e-Proceeding of Engineering, vol. 10, no. 3, pp. 3591–3600, 2023.