

OPTIMALISASI ALGORITMA SUPPORT VECTOR MACHINE PADA ASPECT-BASED SENTIMENT ANALYSIS MENGGUNAKAN GRIDSEARCHCV

Eko Ikhwan Saputra¹, M. Khairul Anam², Helda Yenni³, Hamdani⁴, Ahmad Zamsuri⁵

^{1,2,3,4}Program Studi Teknik Informatika STMIK Amik Riau
(Jl. Purwodadi Indah KM.10, Pekanbaru, Riau 28294)

⁵Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Lancang Kuning
e-mail: ¹2010031802036@sar.ac.id, ²khairulanam@sar.ac.id, ³heldayenni@sar.ac.id,
⁴hamdani@sar.ac.id, ⁵ahmadzamsuri@unilak.ac.id

Abstrak

Support Vector Machine (SVM) merupakan salah satu algoritma Machine Learning yang umum digunakan untuk menyelesaikan permasalahan klasifikasi. SVM sangat baik digunakan untuk klasifikasi biner, tetapi masih kurang optimal dalam melakukan klasifikasi multi-class. SVM masih mendapatkan akurasi yang rendah dalam melakukan klasifikasi multi-class, rentang akurasi yang didapatkan sekitar 52% - 62%. Ketidak konsistenan SVM dalam melakukan klasifikasi multi-class perlu dilakukan perbaruan agar lebih baik lagi. Penelitian ini menggunakan dataset opini masyarakat tentang Pariwisata Yogyakarta, label dari dataset adalah representasi dari wisatawan. Jumlah data yang diperoleh adalah 4121 baris, memiliki 5 label dalam hal ini pelabelan Aspect-Based Sentiment Analysis. Penelitian ini memiliki tahapan data acquisition, pre-processing data, feature extraction, feature selection, modelling dan evaluasi. Penelitian ini melakukan percobaan pada 2 kernel SVM yang berbeda, yaitu linear dan rbf. Kemudian dilakukan hyperparameter tuning menggunakan GridSearchCV untuk mendapatkan parameter terbaik dari algoritma SVM. GridSearchCV dapat meningkatkan akurasi SVM dengan kernel linear dengan nilai peningkatan tertinggi nya mencapai 7%. Akurasi pada kernel Rbf lebih tinggi yaitu 67,4% setelah dilakukan hyperparameter tuning.

Kata kunci: Aspect-Based Sentiment Analysis, GridSearchCV, Machine Learning, Pariwisata Yogyakarta, Support Vector Machine.

Abstract

Support Vector Machine (SVM) is a commonly used machine learning algorithm for solving classification problems. While SVM excels in binary classification, its performance in multi-class classification is suboptimal. SVM still achieves low accuracy in multi-class classification, with accuracy ranging from approximately 52% to 62%. The inconsistency of SVM in multi-class classification calls for improvements to enhance its performance. This research utilizes a dataset containing public opinions about tourism in Yogyakarta, where the dataset labels represent various aspects of tourists. The dataset comprises 4121 rows and is labeled for Aspect-Based Sentiment Analysis. The research involves data acquisition, data pre-processing, feature extraction, feature selection, modeling, and evaluation stages. The study experiments with two different SVM kernels: linear and rbf. Hyperparameter tuning is then performed using GridSearchCV to obtain the optimal parameters for the SVM algorithm. GridSearchCV improves the accuracy of the linear kernel SVM, with the highest increase reaching 7%. The accuracy for the rbf kernel is higher, reaching 67.4% after hyperparameter tuning.

Keywords: Aspect-Based Sentiment Analysis, GridSearchCV, Machine Learning, Yogyakarta Tourism, Support Vector Machine.

1. PENDAHULUAN

Support Vector Machine (SVM) dikenal sebagai salah satu algoritma klasifikasi yang paling kuat dan akurat[1]. SVM satu level dengan Artificial Neural Networks (ANN) jika di komparasi dalam

hal fungsi serta konteks permasalahan yang diselesaikan[1]. Algoritma ini merupakan model *Supervised Learning* dan masih memiliki kemiripan dengan *Algorithm Learning* yang digunakan untuk menganalisa data dalam kasus klasifikasi dan regresi, sejauh ini SVM banyak digunakan dalam kasus klasifikasi[2]. Kelebihan dari SVM yaitu memiliki beberapa *kernel* yang digunakan untuk melakukan pemisahan pada masukan *non-linear* berdimensi tinggi, *kernel-kernel* tersebut adalah *Radial Basis Function* (Rbf), *polynomial*, dan *linear*[2].

SVM dalam melakukan klasifikasi untuk kasus *Aspect-Based Setiment Anlysis* terhadap 3 kelas atau lebih masih mendapatkan hasil yang kurang maksimal, dimana akurasi yang dihasilkan oleh model ketika dilakukan pengujian dapat dikategorikan rendah. Penelitian sebelumnya dengan oleh Khan, et al [3] menunjukkan akurasi dari SVM dalam memprediksi 7 kelas adalah 62%. Penelitian lain yang dilakukan oleh Sivabharathi dan Chitra [4] menunjukkan akurasi pada SVM direntang angka 50%. Untuk meningkatkan akurasi pada model kita bisa memanfaatkan *hyperparameter tuning*.

Hyperparameter tuning sangat berperan penting untuk menghasilkan algoritma *machine learning* yang optimal[5]. Tujuan dari *hyperparameter tuning* adalah menemukan *hyperparameter set* yang paling baik[6]. Salah satu teknik *hyperparameter tuning* yang bisa digunakan adalah *GridSearch CrossValidation* (*GridSearchCV*). *GridSearchCV* merupakan teknik untuk memilih kombinasi algoritma beserta *hyperparameter* dimana kombinasi-kombinasi tersebut diuji dan divalidasi[7]. Pengujian dan validasi akan dilakukan satu persatu secara otomatis pada setiap kombinasi[5], [7]. Di setiap kombinasi akan dilakukan pengukuran kinerja seperti akurasi[8]. Output yang akan dihasilkan adalah kombinasi terbaik berdasarkan pengujian dan validasi yang telah dilakukan.

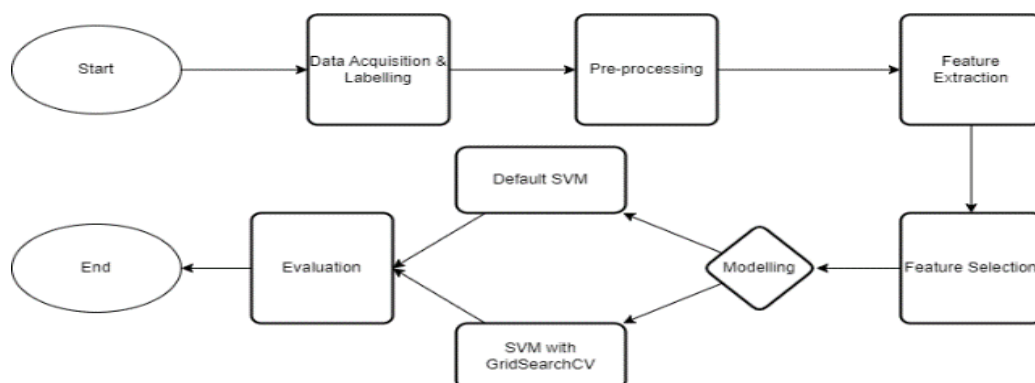
GridSearchCV dapat mengoptimalkan model *machine learning*, pada penelitian yang dilakukan oleh Malik[6] *GridSearchCV* dapat meningkatkan akurasi model 0,19%, algoritma *machine learning* yang digunakan adalah *RandomForest* dengan tugas klasifikasi 2 kelas pada data tabular (terstruktur). Penelitian serupa yang dilakukan oleh Toba, dkk[9] berhasil meningkatkan akurasi dari algoritma SVM sebesar 21,49% dengan tugas klasifikasi 3 kelas pada data tabular (terstruktur). Pada penelitian yang dilakukan oleh Abas[10] mendapatkan optimalisasi yang tinggi pada *tuning* algoritma SVM dengan akurasi mencapai 100% pada tugas klasifikasi 2 kelas di data tabular (terstruktur).

Penelitian ini mencoba meningkatkan akurasi algoritma SVM dalam melakukan klasifikasi *multi-class* dengan kasus *Aspect-Based Sentiment Analysis* Pariwisata Yogyakarta. Data yang dikumpulkan merupakan *tweet* dari platform sosial media *twitter* sejak tanggal 2 september 2021 hingga 31 desember 2022 dengan jumlah 4121 baris. Data yang telah dikumpulkan kemudian diberi label, setelah pemberian label, ternyata distribusi antarkelas tidak seimbang atau yang dikenal sebagai *imbalance data*.

Tujuan penelitian ini yaitu mengoptimalkan algoritma SVM dengan *GridSearchCV* dalam melakukan *Aspect-Based Sentiment Anlysis* pada data teks yang diperoleh dari *twitter* dengan jumlah kelas sebanyak 5.

2. METODE PENELITIAN

Penelitian ini menggunakan metode *Support Vector Machine* (SVM) untuk klasifikasi sentiment dan *GridSearchCV* sebagai *hyperparameter tuning*. Rangkaian proses penelitian yang dilakukan dapat dilihat pada gambar 1.



Gambar 1. Alur Penelitian

2.1. Data Acquisition & Labelling

Dataset yang digunakan merupakan data *tweet* pengguna *twitter* di Indonesia yang terkait dengan kata kunci Pariwisata Yogyakarta. Dataset ini pada awalnya berjumlah 4121 baris, lalu dilakukan operasi *remove duplicates* menggunakan *Microsoft Excel* sehingga tersisa 2128 baris. Kemudian data diberi label dengan jumlah label adalah 5, yaitu Kebijakan, Fasilitas, Layanan, Kuliner dan Other. Hal ini dilakukan karena SVM merupakan algoritma *supervised* yang membutuhkan label pada dataset dalam proses *traininig*. Proses pelabelan dilakukan secara manual menggunakan *Microsoft Excel*.

2.2. Text Pre-processing

Di tahap ini dilakukan pengolahan pada data teks supaya siap untuk digunakan sebagai data latih pada model, karakter atau kata yang tidak penting dihilangkan pada proses ini. Proses yang dilakukan antara lain adalah *case folding*, yaitu proses dimana seluruh huruf pada teks diubah menjadi huruf kecil[11], [12]. Selanjutnya adalah *Stopwords removal*, pada tahap ini kata tidak penting yang masuk pada *stopwords list* pada teks hasil proses sebelumnya akan dibuang[11], [13]. Contoh bentuk kata yang akan dibuang seperti “di”, “oleh”, “pada”, “sebuah”, “karena”, dan lainnya[14]. *Stopwords list* yang digunakan berbahasa Indonesia dari *nlTK*. Tahap terakhir pada proses ini adalah *stemming*, yaitu teknik untuk menguraikan kata sehingga menjadi bentuk dasar berdasarkan aturan-aturan tertentu[11], [12]. Tujuan dari *stemming* adalah menghilangkan segala jenis imbuhan pada setiap kata[14]. Pada penelitian ini proses *stemming* menggunakan *library sastrawi*. Teks yang telah melalui proses-proses sebelumnya akan di record dalam kolom baru dengan dan kolom data teks original dihapus. Selanjutnya operasi *remove duplicates* dilakukan kembali sehingga menyisakan 1935 baris saja.

2.3. Feature Extraction

Feature Extraction pada teks yaitu proses mengekstrak fitur asli menjadi fitur baru dalam bentuk kecil yang lebih sederhana dengan cara pemetaan fungsional[15], [16]. Teknik *feature extraction* yang dipakai pada penelitian ini adalah *Term Frequency – Invers Document Frequency* (TF-IDF) dengan N-gram model unigram. TF-IDF merupakan teknik memberikan bobot pada kata[12] untuk mengetahui pentingnya kata pada dokumen[17].

2.4. Feature Selection

Tahap ini dilakukan supaya fitur-fitur yang tidak relevan untuk proses *training* model *machine learning* direduksi[14]. Ini bertujuan untuk menghilangkan fitur yang dianggap mengganggu dalam proses klasifikasi[14]. Seleksi fitur yang digunakan pada penelitian ini adalah teknik *Chi-Square*, teknik ini bertujuan untuk menghilangkan istilah yang tidak mencapai ambang batas tertentu, sehingga berdampak pada biaya dan akurasi suatu metode[18]. *Chi-Square* merupakan algoritma *supervised learning* dan mampu menghilangkan fitur yang tidak relevan dengan tanpa mengurangi tingkat akurasi dan sudah umum digunakan jika ingin menyeleksi fitur dengan keakuratan yang baik[19].

2.5. Modelling

Tahap ini adalah proses membuat model dengan algoritma SVM, model yang dibuat lebih dari satu untuk dilakukan evaluasi dan perbandingan. Pada penelitian ini, model SVM dilatih dengan parameter $C=1$, $\gamma=1$ dengan dua kernel yaitu *Rbf* dan *linear*. Lalu dilakukan *hyperparameter tuning* menggunakan teknik *GridSearchCV* untuk mendapatkan parameter terbaik dalam melatih dataset yang digunakan. Parameter yang di tuning adalah C dan γ dengan proses dua kali *tuning*, yaitu untuk kernel *Rbf* dan kernel *linear*. Berikut adalah list parameter yang di *tuning*.

- C : 0.1, 1, 10, 100, 1000

- γ : 1, 0.1, 0.01, 0.001, 0.0001

2.6. Evaluasi

Evaluasi dilakukan untuk melihat dan mengukur performa dari model yang telah dilatih. Sebagai acuan untuk mengukur performa model di penelitian ini adalah *confusion matrix*. *Confusion matrix* adalah tabel yang berisi informasi tentang bagaimana sistem klasifikasi berhasil

mengklasifikasikan data uji dengan benar[20], [21]. Pada tahap ini juga digunakan teknik *cross-validation* sebagai cara untuk memvalidasi akurasi model agar lebih akurat. Nilai K yang digunakan saat *cross-validation* yaitu 10.

3. HASIL DAN PEMBAHASAN

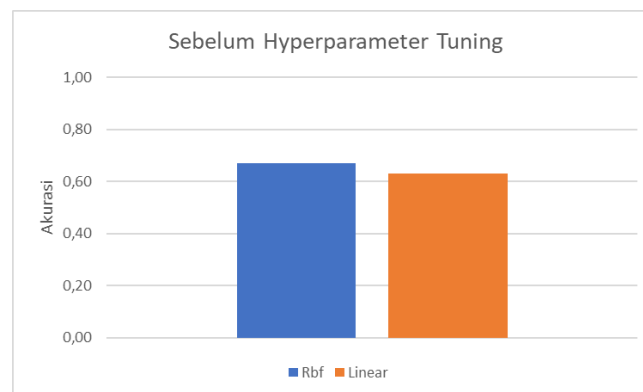
3.1. Hasil

Sebelum proses *modelling*, dataset dibagi menjadi dua, yaitu data *training* dan data *testing*, proses ini biasa disebut dengan *split data*.. Data *traininig* digunakan untuk melatih model dan data *testing* untuk menguji performa model. Data dibagi dengan rasio data *training* 70% dan data *testing* 30%. Kemudian label pada dataset yang berupa string diubah menjadi bentuk numerik seperti pada tabel dibawah.

Tabel 1. Label numerik

Label	Kode numerik
Kebijakan	1
Fasilitas	2
Layanan	3
Kuliner	4
Other	5

Penelitian ini ingin mengetahui pengaruh *hyperparameter tuning* menggunakan GridSearchCV terhadap performa model SVM. Di awal proses *modelling*, peneliti mencoba mengisi *value* parameter pada model yang akan dibangun berdasarkan intuisi dengan *value* yang sudah dijelaskan pada sub-bab 3.5. Hasil nya dapat dilihat pada tabel gambar 2 dan tabel 5.



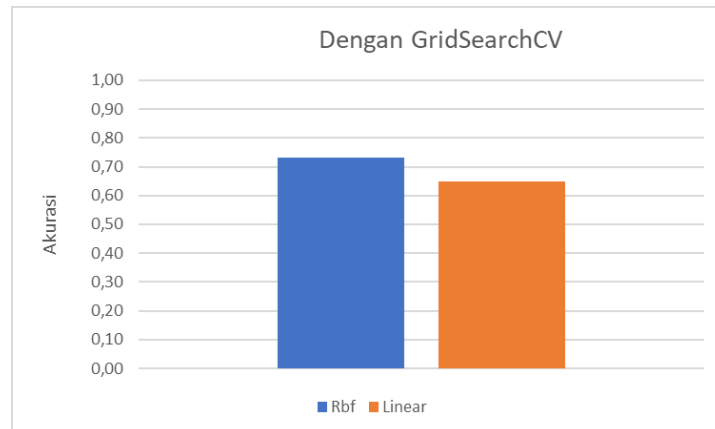
Gambar 2. Performa model sebelum hperparameter tuning

Tabel 2. Evaluasi sebelum *hyperparameter tuning*

Cross-validation	Akurasi	
	Rbf	Linear
Fold 1	0.645	0.552
Fold 2	0.626	0.578
Fold 3	0.655	0.612
Fold 4	0.666	0.604
Fold 5	0.654	0.597
Fold 6	0.631	0.598
Fold 7	0.652	0.635
Fold 8	0.647	0.633
Fold 9	0.621	0.590
Fold 10	0.669	0.586
Rata-rata	0.646	0.598

Penelitian ini mengukur performa model menggunakan akurasi. Pada percobaan pertama dapat dilihat bahwa performa model masih termasuk kedalam kategori rendah dengan akurasi pada kernel *Rbf* adalah 67,64% dan kernel *linear* 63,33%. Sedangkan pengujian menggunakan *cross-validation* akurasi tertinggi kernel *Rbf* di angka 66,9% pada *fold* ke 10 dengan rata-rata akurasi adalah 64,6% dan kernel *linear* di angka 63,5% pada *fold* ke 7 dengan rata-rata akurasi 59,8%.

Kemudian dilakukan percobaan kedua yaitu *hyperparameter tuning* dengan GridSearchCV. Parameter dan *value* yang dijadikan *input* pada GridSearchCV sesuai dengan yang sudah dijabarkan pada tabel di sub-bab 2.5. Parameter terbaik hasil GridSearchCV untuk kedua kernel adalah $C = 10$ dan $\gamma = 1$. Setelah mendapatkan parameter terbaik, maka dilakukan training model kembali dengan hasil pada gambar 3 dan tabel 6.

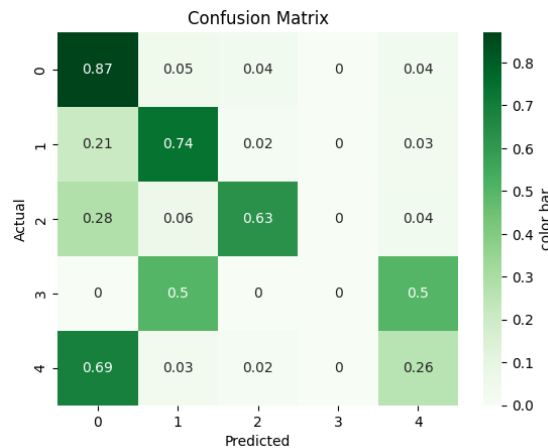


Gambar 3. Performa model setelah optimalisasi hyperparameter tuning

Tabel 3. Evaluasi setelah hyperparameter tuning

Cross-validation	Accuracy	
	Rbf	Linear
Fold 1	0.697	0.660
Fold 2	0.695	0.669
Fold 3	0.674	0.679
Fold 4	0.678	0.666
Fold 5	0.679	0.667
Fold 6	0.698	0.676
Fold 7	0.700	0.657
Fold 8	0.669	0.679
Fold 9	0.674	0.638
Fold 10	0.709	0.686
Rata-rata	0.687	0.668

Hasil dari percobaan kedua yaitu terdapat peningkatan akurasi pada model yang dibangun dengan kernel *linear*, model yang sebelumnya 63,33% meningkat 2,59% ke angka 65,92%. Sedangkan model yang dibangun dengan kernel *Rbf* mengalami sedikit penurunan sebesar 0,12% dari 67,64% ke angka 67,12%. Pengujian dengan *cross-validation* juga menunjukkan peningkatan sebesar 7% pada model kernel *linear* dimana akurasi rata-rata awal adalah 59,8% menjadi 66,8% dengan akurasi tertinggi 68,6% pada *fold* ke 10. Sedangkan model dengan kernel *Rbf* juga mengalami peningkatan ketika diuji dengan *cross-validation*, akurasi rata-rata sebelumnya yang sebelumnya 64,6% meningkat sebesar 4,1% menjadi 68,7% dengan nilai akurasi tertinggi yaitu 70,9% pada *fold* ke 10. Gambar 4 merupakan plot dari *confusion matrix* dari model terbaik.



Gambar 4. Confusion Matrix model terbaik

Tabel 4. Classification report

Model	Akurasi Tertinggi	Precision	Recall	F-1 Score
Linear	66,9	0.66	0.43	0.42
Rbf	59,9	0.75	0.54	0.60
Linear	68,8	0.57	0.48	0.50
GSCV				
Rbf	70,9	0.56	0.50	0.51
GSCV				

3.2. Pembahasan

Berdasarkan percobaan pertama dan percobaan kedua, *hyperparameter tuning* dengan GridSearchCV bisa mengoptimalkan proses pemodelan *machine learning* algoritma SVM untuk ABSA dengan kasus *multi-class*. Hasil dari penelitian ini dibandingkan dengan penelitian sebelumnya yang dilakukan oleh Khan, et al.[3] dan Sivabharathi bersama Chitra[4]. Perbandingan yang digunakan adalah akurasi dari model yang dibangun. Dapat dilihat pada tabel bahwa GridSearchCV dapat meningkatkan akurasi dari model dengan algoritma SVM yang ditugaskan untuk klasifikasi *multi-class*. Output dari GridSearchCV merupakan *best params* (parameter terbaik) dari algoritma dan dataset yang kita jadikan sebagai *input*. Penelitian ini mendapatkan akurasi yang lebih tinggi dari sebelumnya, dimana Khan, et al[3] dengan akurasi 62% dan Sivabharathi bersama Chitra[4] dengan akurasi 50%. Sedangkan penelitian ini dengan akurasi paling tinggi adalah 70.9%.

Penelitian sebelumnya menyatakan GridSearchCV dapat mempermudah proses pemilihan model *machine learning*[5]. Penelitian ini pun menunjukkan hal yang sama, dimana dengan menggunakan GridSearchCV kita dapat mengoptimalkan model dengan kenaikan akurasi terbesar adalah 7% pada percobaan dengan kernel *linear*. Meskipun demikian, peningkatan yang didapatkan masih tidak dapat dikatakan optimal, karena performa model masih dalam kategori rendah. Peneliti menduga salah satu penyebabnya adalah *imbalance* data yang ada pada dataset, dimana distribusi setiap kelas pada dataset berbeda jauh. Penelitian lain menyampaikan bahwa *imbalance* data pada dataset bisa mempengaruhi model dalam mengklasifikasi[13].

4. KESIMPULAN

Penelitian ini menyimpulkan bahwa algoritma SVM masih belum optimal dalam menyelesaikan kasus ABSA, ini dilihat dari akurasi yang dihasilkan setiap percobaan dibawah 75%. Tetapi GridSearchCV dapat meningkatkan performa model SVM serta memudahkan dan mempercepat proses menemukan parameter yang paling optimal.

Terlepas dari hasil penelitian yang telah disampaikan, tentunya ada keterbatasan dalam proses penelitian ini. Salah satu keterbatasan adalah pelabelan data secara manual yang memakan waktu cukup

lama. Untuk penelitian yang akan dilakukan lebih lanjut disarankan untuk melakukan pelabelan bersama, membayar jasa pelabelan atau menggunakan cara pelabelan otomatis.

Saran untuk penelitian selanjutnya adalah menggunakan dataset dengan distribusi setiap kelas yang seimbang, menggunakan teknik atau algoritma tertentu untuk menangani imbalance data seperti *Synthetic Minority Over-sampling (SMOTE)* dan juga bisa mencoba teknik hyperparameter tuning lain seperti *Random Search Cross Validation*.

Daftar Pustaka

- [1] A. S. Ritonga, E. S. Purwaningsih, dan U. W. Putra, “PENERAPAN METODE SUPPORT VECTOR MACHINE (SVM) DALAM KLASIFIKASI KUALITAS PENGELASAN SMAW (SHIELD METAL ARC WELDING),” 2018, doi: 10.21107/edutic.v5i1.4382.
- [2] A. Ahmad dan W. Gata, “Sentimen Analisis Masyarakat Indonesia di Twitter Terkait Metaverse dengan Algoritma Support Vector Machine,” *jtik*, vol. 6, no. 4, hlm. 548–555, Mar 2022, doi: 10.35870/jtik.v6i4.569.
- [3] Md. S. S. Khan, S. R. Rafa, A. E. H. Abir, dan A. K. Das, “Sentiment Analysis on Bengali Facebook Comments To Predict Fan’s Emotions Towards a Celebrity,” *J. eng. adv.*, hlm. 118–124, Jul 2021, doi: 10.38032/jea.2021.03.001.
- [4] G. Sivabharathi dan K. Chitra, “Aspect-based sentiment analysis on current state of the market by classifying customer product reviews using support vector machine with several kernel learning,” *ijhs*, hlm. 3168–3180, Apr 2022, doi: 10.53730/ijhs.v6nS1.5371.
- [5] W. Nugraha dan A. Sasongko, “Hyperparameter Tuning on Classification Algorithm with Grid Search,” *SISTEMASI*, vol. 11, no. 2, hlm. 391, Mei 2022, doi: 10.32520/stmsi.v11i2.1750.
- [6] “Hyperparameter Tuning Menggunakan GridsearchCV pada Random Forest untuk Deteksi Malware,” *JURNAL MULTIMEDIA NETWORKING INFORMATICS*, vol. 9, no. 1, hlm. 43–50, Mei 2023, doi: 10.32722/multinetics.v9i1.5578.
- [7] A. W. Ishlah, S. Sudarno, dan P. Kartikasari, “IMPLEMENTASI GRIDSEARCHCV PADA SUPPORT VECTOR REGRESSION (SVR) UNTUK PERAMALAN HARGA SAHAM,” *J.Gauss*, vol. 12, no. 2, hlm. 276–286, Jul 2023, doi: 10.14710/j.gauss.12.2.276-286.
- [8] “Optimalisasi Deteksi Serangan DDoS Menggunakan Algoritma Random Forest, SVM, KNN dan MLP pada Jaringan Komputer | Maulana | Indonesian Journal of Mathematics and Natural Sciences.” Diakses: 24 Oktober 2023. [Daring]. Tersedia pada: <https://journal.unnes.ac.id/nju/index.php/JM/article/view/48231>
- [9] A. Toha, P. Purwono, dan W. Gata, “Model Prediksi Kualitas Udara dengan Support Vector Machines dengan Optimasi Hyperparameter GridSearch CV,” *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 4, no. 1, hlm. 12–21, Mei 2022, doi: 10.12928/biste.v4i1.6079.
- [10] M. A. Haqmi Abas, “Agarwood Oil Quality Classification using Support Vector Classifier and Grid Search Cross Validation Hyperparameter Tuning,” *IJETER*, vol. 8, no. 6, hlm. 2551–2556, Jun 2020, doi: 10.30534/ijeter/2020/55862020.
- [11] L. A. Andika, P. A. N. Azizah, dan R. Respatiwulan, “Analisis Sentimen Masyarakat terhadap Hasil Quick Count Pemilihan Presiden Indonesia 2019 pada Media Sosial Twitter Menggunakan Metode Naive Bayes Classifier,” *IJAS*, vol. 2, no. 1, hlm. 34, Jul 2019, doi: 10.13057/ijas.v2i1.29998.
- [12] P. Arsi dan R. Waluyo, “Analisis Sentimen Wacana Pemindahan Ibu Kota Indonesia Menggunakan Algoritma Support Vector Machine (SVM),” *JTIK*, vol. 8, no. 1, hlm. 147, Feb 2021, doi: 10.25126/jtik.0813944.
- [13] S. Khomsah dan Agus Sasmito Aribowo, “Text-Preprocessing Model Youtube Comments in Indonesian,” *RESTI*, vol. 4, no. 4, hlm. 648–654, Agu 2020, doi: 10.29207/resti.v4i4.2035.
- [14] A. Z. Amrullah, A. S. Anas, dan M. A. J. Hidayat, “Analisis Sentimen Movie Review Menggunakan Naive Bayes Classifier Dengan Seleksi Fitur Chi Square,” 2020, doi: 10.30812/bite.v2i1.804.
- [15] D. P. Langgeni dan Z. A. Baizal, “CLUSTERING ARTIKEL BERITA BERBAHASA INDONESIA MENGGUNAKAN UNSUPERVISED FEATURE SELECTION,” 2010.
- [16] Y. Puspitarani, S. Sukenda, A. P. Wahyu, B. Yustim, dan S. Sunjana, “MODELING OF DIGITALIZATION AND VISUALIZATION OF LABOR COMPLAINTS USING OCR,

- FEATURE EXTRACTION AND BUSINESS INTELLIGENCE,” *jitter*, vol. 7, no. 2, hlm. 166–171, Apr 2021, doi: 10.33197/jitter.vol7.iss2.2021.592.
- [17] M. P. Simatupang dan D. P. Utomo, “ANALISA TESTIMONIAL DENGAN MENGGUNAKAN ALGORITMA TEXT MINING DAN TERM FREQUENCY- INVERSE DOCUMENT FREQUENCY (TF-IDF) PADA TOKO ALLMEEART,” *KOMIK*, vol. 3, no. 1, Des 2019, doi: 10.30865/komik.v3i1.1697.
- [18] I. Listiowarni dan N. Puspa Dewi, “Pemanfaatan Klasifikasi Soal Biologi Cognitive Domain Bloom’s Taxonomy Menggunakan KNN Chi-Square Sebagai Penyusunan Naskah Soal,” *Digitalzone*, vol. 11, no. 2, hlm. 186–197, Nov 2020, doi: 10.31849/digitalzone.v11i2.4798.
- [19] M. Tsani, A. Rupaka, L. Asmoro, dan B. Pradana, “ANALISIS SENTIMEN REVIEW TRANSPORTASI MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE BERBASIS CHI SQUARE,” *smartcomp*, vol. 9, no. 1, hlm. 35–39, Jan 2020, doi: 10.30591/smartcomp.v9i1.1817.
- [20] B. Gunawan, H. S. Pratiwi, dan E. E. Pratama, “Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes,” *JEPIN*, vol. 4, no. 2, hlm. 113, Des 2018, doi: 10.26418/jp.v4i2.27526.
- [21] D. Normawati dan S. A. Prayogi, “Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter,” vol. 5, 2021, doi: 10.30645/j-sakti.v5i2.369.