

Klasifikasi Sentimen Terhadap Pengangkatan Kaesang Sebagai Ketua Umum Partai PSI Menggunakan Metode Support Vector Machine

Safrizal, Surya Agustian*, Alwis Nazir, Yusra

¹Fakultas Sains dan Teknologi, Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru, Indonesia

Email: ¹12050110516@students.uin-suska.ac.id, ^{2,*} surya.agustian@uin-suska.ac.id,

³alwis.nazir@uin-suska.ac.id, ⁴yusra@uin-suska.ac.id

Email Penulis Korespondensi: surya.agustian@uin-suska.ac.id

Submitted: 13/06/2024; Accepted: 25/06/2024; Published: 26/06/2024

Abstrak—Penunjukan Kaesang Pangarep sebagai Ketua Umum Partai Solidaritas Indonesia (PSI) telah memicu berbagai tanggapan di media sosial, khususnya Twitter. Penelitian ini bertujuan untuk melakukan klasifikasi sentimen terhadap tanggapan masyarakat mengenai penunjukan tersebut menggunakan algoritma Support Vector Machine (SVM) dengan representasi fitur FastText. Data yang digunakan untuk klasifikasi dengan data training yang kecil. Proses text preprocessing mencakup cleaning, case folding, tokenizing, normalisasi, stopword removal, dan stemming. Fasttext word embedding digunakan untuk merepresentasikan kata menjadi kedalam bentuk vector selanjutnya membuat model SVM dengan Grid Search digunakan untuk penyetelan parameter guna mendapatkan model optimal. Penggunaan dataset eksternal untuk memperluas dataset pelatihan yang awalnya terbatas, meningkatkan representasi data, dan meningkatkan kinerja model dalam mengklasifikasikan sentimen. Dataset Covid diperluas dengan menambahkan tweet sebanyak 100, 200, dan 300 untuk setiap label negatif, positif dan netral. Dari eksperimen yang dilakukan model pada data uji terlihat akurasi terbaik ditemukan pada ID eksperimen C2 dengan nilai F1-Score sebesar 53.59% dan akurasi 62.73%. Selanjutnya pada ID eksperimen C3 dengan dataset yang sama, nilai F1-Score adalah 50.46% dan akurasi 60.46%. Terakhir, pada ID eksperimen C7 dengan dataset yang sama, nilai F1-Score adalah 47.19% dan akurasi 53.09%.

Kata Kunci: Kaesang Pangarep; klasifikasi sentimen; SVM; FastText; media sosial

Abstract—The appointment of Kaesang Pangarep as the Chairman of the Indonesian Solidarity Party (PSI) has sparked various responses on social media, particularly on Twitter. This research aims to classify public sentiment regarding the appointment using the Support Vector Machine (SVM) algorithm with FastText feature representation. The data used for classification involves a small training dataset. The text preprocessing process includes cleaning, case folding, tokenizing, normalization, stopword removal, and stemming. FastText word embedding is used to convert words into vectors, and an SVM model with Grid Search is used for parameter tuning to obtain the optimal model. The use of external datasets to expand the initially limited training dataset enhances data representation and improves the model's performance in sentiment classification. The Covid dataset was expanded by adding 100, 200, and 300 tweets for each negative, positive, and neutral label. From the experiments conducted, the best accuracy on the test data was found in experiment ID C2 with an F1-Score of 53.59% and an accuracy of 62.73%. In experiment ID C3 with the same dataset, the F1-Score was 50.46% and the accuracy was 60.46%. Finally, in experiment ID C7 with the same dataset, the F1-Score was 47.19% and the accuracy was 53.09%.

Keywords: Kaesang Pangarep; Sentiment Classification; SVM; FastText; Social Media

1. PENDAHULUAN

Partai Solidaritas Indonesia (PSI) merupakan salah satu partai politik yang relatif baru di Indonesia, berdiri pada akhir tahun 2014 dengan visi untuk mewujudkan Indonesia yang memiliki karakter kerakyatan, kemanusiaan, keragaman, keadilan, kemajuan, dan martabat yang tinggi[1]. Belakangan ini, aktivitas politik mengalami peningkatan yang signifikan, terutama ketika PSI menunjuk Kaesang Pangarep, putra dari Presiden Jokowi, sebagai Ketua Umum. Tentu saja, keputusan tersebut menjadi perbincangan hangat di kalangan masyarakat, dengan beberapa mendukungnya sebagai perwakilan generasi muda yang memahami politik[2], sementara sejumlah pihak menilai PSI di bawah kepemimpinan Kaesang hanya berorientasi pada kekuasaan belaka[3].

Tanggapan dan pandangan masyarakat mengenai penunjukan Kaesang sebagai Ketua Umum partai ini tengah mengalami perkembangan yang signifikan di berbagai platform media sosial, termasuk Twitter. Platform tersebut menjadi sarana utama bagi banyak individu untuk menyampaikan respons dan opini mereka terhadap peran Kaesang sebagai Ketua Umum partai PSI. Opini yang diungkapkan seringkali bersifat spontan dan emosional, mencakup berbagai pandangan dari positif, negatif, hingga netral. Pada penelitian ini, dilakukan klasifikasi sentimen untuk mengevaluasi dan mengidentifikasi polaritas opini tersebut. Sentimen merupakan ekspresi dari pendapat dan emosi seseorang terhadap subjek atau topik tertentu[4]. Bidang klasifikasi sentimen menggunakan teknologi seperti pengolahan bahasa alami, komputasi linguistik, dan text mining, dengan tujuan untuk memahami pandangan, sentimen, evaluasi, dan emosi terhadap beragam aspek seperti topik, produk, layanan, organisasi, individu, maupun kegiatan. Salah satu teknik yang umum digunakan dalam analisis sentimen adalah dengan menggunakan machine learning.

Machine learning merupakan sistem yang menggunakan data yang telah dipelajarinya untuk melakukan klasifikasi sentimen menggunakan algoritma tertentu[5]. Beberapa algoritma machine learning yang umum digunakan dalam analisis sentimen meliputi Support Vector Machine (SVM), Naïve Bayes (NB), K-Nearest Neighbor (KNN), dan Decision Tree. Penelitian ini akan melakukan klasifikasi pada tiga kelas, yaitu positif, negatif, dan netral. Dalam melakukan klasifikasi pada tiga kelas tersebut, diperlukan algoritma yang mampu mengklasifikasi data dengan dimensi yang tinggi. Algoritma SVM cocok digunakan dalam kasus ini karena SVM memiliki keunggulan dalam

menyelesaikan masalah klasifikasi pada data berdimensi tinggi, terutama ketika terdapat keterbatasan pada jumlah sampel data yang tersedia.

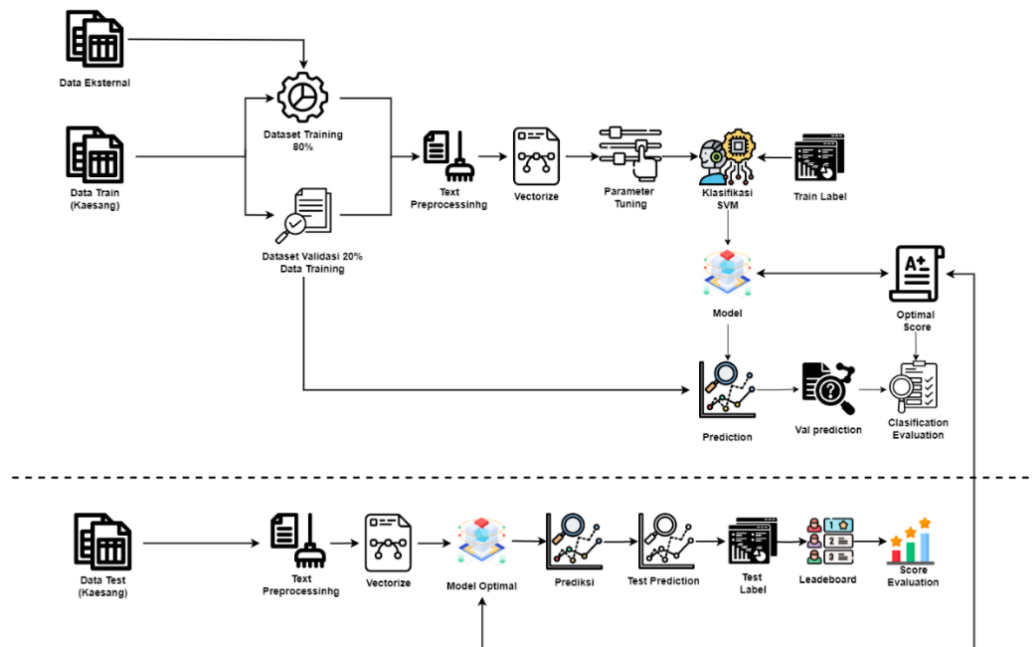
Penelitian yang dilakukan Abdul Mohaimin Rahat pada tahun 2019, menggunakan metode SVM dan Naïve Bayes untuk mengetahui Perbandingan Algoritma Naive Bayes dan SVM berdasarkan Analisis Sentimen Menggunakan Review Himpunan data[6]. Hasil akurasi pada perbandingan metode ini adalah Akurasi SVM adalah 83% dan akurasi Naïve Bayes adalah 77%. Nilai Presisi dan Recall untuk SVM adalah 91% dan 82% serta 89% dan 84% untuk Naïve Bayes. Selanjutnya pada Penelitian yang dilakukan Md Deloar Hossan Jasy pada tahun 2021, melakukan evaluasi kinerja sentimen klasifikasi menerapkan SVM, KNN, dan Naïve Bayes. Hasil di peroleh algoritma Support Vector Machine memiliki akurasi tertinggi 92%, disusul KNN dan Naive Bayes dengan Akurasi masing-masing 88% dan 85%[7]. Pada penelitian Aina Damayunita pada tahun 2022, membandingkan Naive Bayes, KNN, dan SVM untuk klasifikasi penyakit. SVM memiliki akurasi tertinggi 92%, diikuti KNN 88% dan Naive Bayes 85% [8]. Selanjutnya penelitian yang dilakukan oleh Khafifah Munawaroh pada tahun 2022, membandingkan kinerja algoritma SVM, Naïve Bayes, dan KNN dalam menganalisis sentimen opini publik terhadap vaksinasi COVID-19 di Twitter. SVM menonjol dengan akurasi tertinggi, mencapai 96,3%, dibandingkan dengan Naïve Bayes 94% dan KNN 91%[9]. Sementara itu penelitian yang dilakukan oleh Rani Yunita dan Mia Kamayani pada tahun 2023, membandingkan SVM dan Naïve Bayes dalam analisis sentimen terkait kebijakan penghapusan kewajiban skripsi. Hasilnya menunjukkan bahwa SVM menjadi algoritma terbaik dengan akurasi 80%, recall 83%, presisi 76%, dan F1-Score 79%, dengan total 331 sentimen positif dan 369 sentimen negatif [10].

Dalam proses klasifikasi teks, vektorisasi data merupakan langkah krusial. Salah satu metode word embedding yang menjadi fokus adalah FastText. Penelitian yang dilakukan oleh Arliyanti Nurdin pada tahun 2020, membandingkan kinerja word embedding seperti word2vec, GloVe, dan FastText dalam konteks klasifikasi teks. Hasil eksperimen menunjukkan bahwa FastText memiliki kinerja terbaik dengan F-Measure untuk dataset 20 newsgroup (0.979) dan Reuters News (0.715). Keunggulan FastText terlihat dalam representasi vektor kata yang tidak ada dalam korpus (out of vocabulary)[11].

Penelitian shared task mengenai klasifikasi sentimen sebagaimana di dalam penelitian[12], yang mengangkat problem klasifikasi dengan data training yang kecil. Jumlah data training yang tersedia untuk pengembangan model machine learning adalah 300 tweet, yang kurang memenuhi syarat untuk mendapatkan hasil yang baik. penelitian ini akan melakukan klasifikasi sentimen dari tiga kelas yang meliputi positif, negatif, dan netral. Metode yang digunakan adalah SVM dengan penyetulan parameter menggunakan Gridsearch untuk mencapai model optimal. Eksperimen akan dilakukan dengan variasi parameter SVM dan menggunakan ekstraksi fitur dari FastText language model sebagai kamus vektor. Selain itu, penelitian ini juga menyelidiki potensi penggunaan data eksternal dengan topik yang berbeda dari isu yang sedang diteliti untuk meningkatkan jumlah training yang terbatas. Proses klasifikasi sentimen akan dilakukan menggunakan bahasa pemrograman Python dengan berbagai library yang diperlukan untuk analisis data. Evaluasi dan pengujian model akan menggunakan confusion matrix.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian



Gambar 1. Design Tahapan Penelitian

Penelitian ini mencakup beberapa langkah, seperti Dataset, *Text preprocessing*, Word Embedding Fastext, penerapan klasifikasi menggunakan Support Vector Machine (SVM), dilanjutkan dengan tahap evaluasi menggunakan Confusion Matriks, Paragraf berikutnya akan memberikan penjelasan yang lengkap dan rinci tentang setiap langkah tersebut.

2.2 Dataset

Dataset yang digunakan dalam penelitian ini berasal dari dua sumber yang diperoleh melalui teknik crawling Twitter. Data pertama adalah dataset yang mencerminkan sentimen terhadap Kaesang yang menjadi ketua PSI (Data Kaesang). Data kedua adalah sentimen terhadap program vaksin Covid-19 (Data Covid). Dataset Kaesang menjadi fokus utama penelitian ini karena dikumpulkan secara khusus untuk menginvestigasi penggunaan data yang terbatas dalam proses pelatihan machine learning.

Dataset Kaesang dikumpulkan khusus untuk shared task penelitian ini, dengan fokus pada penyelidikan penggunaan data terbatas dalam proses training machine learning. Data diperoleh melalui crawling Twitter dari rentang waktu 25 September 2023 hingga 03 Oktober 2023 dengan kata kunci utama "Kaesang PSI". Untuk memberikan label sentimen pada setiap tweet, dilakukan proses crowd sourcing dengan melibatkan minimal empat anotator untuk setiap tweet. Label positif, netral, atau negatif ditetapkan berdasarkan majority vote. Dalam kasus ketidaksepakatan label, tweet yang tidak memiliki label dominan dihapus dan dianggap tidak valid.

Untuk data training, tersedia dua dataset yang masing-masing jumlahnya sebanyak 300 tweet yang dilabeli sebagai positif, negatif, dan netral. Dataset train pertama yaitu Kaesang V1 dan dataset train kedua yaitu Kaesang V2. Sementara untuk testing, tersedia 924 tweet yang label gold standard-nya disimpan pada server leaderboard[12]. Model paling optimal dari hasil penelitian ini diterapkan pada data testing tersebut, dan hasil prediksinya disubmit pada sistem leaderboard untuk mendapatkan skor evaluasi.

Selain itu, Data Covid yang digunakan merupakan data eksternal yang telah dikumpulkan oleh penelitian sebelumnya[13], [14], [15] Data ini terdiri dari 8.000 tweet yang telah diberi label positif, negatif, dan netral. Data Covid akan digunakan sebagai tambahan dataset pelatihan untuk data Kaesang.

Tabel 1. Rincian dataset dan jumlah tweet

Dataset	Jumlah Tweet
Train Kesang V1	300
Train Kaesang V2	300
Test Kaesang	924
Data Covid	8.000

Dari rincian dataset yang ditunjukkan dalam Tabel 1, penelitian ini menggunakan satu dataset train saja, yaitu dataset train Kaesang V2 sebagai data pelatihan dan validasi. Selanjutnya, dataset train Kaesang V2 dan Data Covid digabungkan untuk pelatihan, dan data test Kaesang digunakan sebagai data testing untuk mendapatkan skor evaluasi pada sistem leaderboard.

2.3 Text Preprocessing

Text Preprocessing adalah langkah penting yang bertujuan untuk membersihkan dan menstandarisasi data sebelum melanjutkan pemrosesan selanjutnya. Beberapa tahapan yang umum dilakukan dalam proses pra-pemrosesan teks adalah sebagai berikut:

- Cleaning*: Menyederhanakan teks dengan menghapus tanda baca, tautan (link), dan karakter khusus yang tidak relevan.
- Case Folding*: Membuat konsistensi dalam pemrosesan dengan mengubah seluruh teks menjadi huruf kecil
- Tokenizing*: Memecah teks menjadi unit-unit kecil seperti kata-kata atau token untuk analisis lebih lanjut.
- Normalisasi: Mengubah kata-kata ke dalam bentuk dasar atau normal, seperti mengoreksi kata-kata yang typo atau mengubah singkatan tidak baku menjadi kata baku.
- Stopword*: Menghapus kata-kata umum yang kurang informatif dalam konteks tertentu, seperti "dan", "di", atau "untuk".
- Stemming*: Mengubah kata-kata ke bentuk dasarnya dengan menghilangkan akhiran atau awalan khusus.

2.4 Fasttext

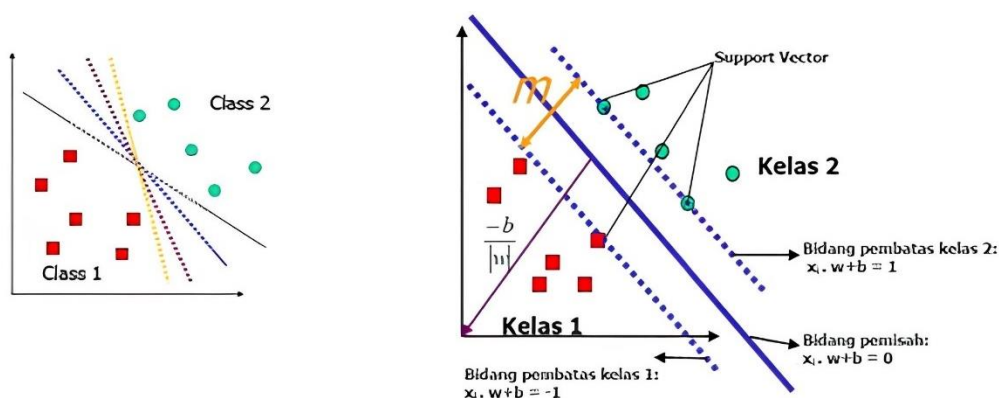
FastText[16] adalah sebuah teknik word embedding yang merupakan pengembangan dari word2vec. Teknik ini memperoleh representasi kata dengan memperhitungkan informasi subkata. Ini membantu dalam memahami makna kata-kata yang lebih singkat dan dapat meningkatkan kemampuan model dalam mengatasi kata-kata jarang muncul atau kesalahan penulisan[17]. Dalam FastText, setiap kata direpresentasikan sebagai kumpulan karakter n-gram. Misalnya, untuk kata "representasi" dengan n-gram = 3, FastText akan memprosesnya menjadi ("rep", "epr", "pre", "res", "ese", "sen", "ent", "nta", "asi"). Keunggulan FastText terletak pada kecepatan pemrosesannya yang relatif cepat. FastText dapat mengatasi kata-kata yang belum pernah muncul dalam kamus, sedangkan pendekatan lain seperti Word2Vec cenderung mengalami kesalahan dalam situasi ini.

2.5 Parameter Tuning

Parameter tuning adalah suatu pengaturan atau konfigurasi dari metode atau model tertentu yang dapat dipilih secara bebas dalam suatu rentang nilai tertentu[18]. Pengaturan ini memiliki dampak langsung terhadap kinerja atau kualitas suatu model. Metode hyperparameter yang digunakan adalah metode grid search yaitu salah satu metode yang bisa dimanfaatkan untuk mencari parameter optimal dalam suatu model[19]. Algoritma ini bekerja dengan memilih kombinasi model dan hyperparameter dengan menguji setiap kombinasi secara berurutan dan melakukan validasi pada masing-masing kombinasi tersebut[20].

2.6 Klasifikasi Support Vector Machine (SVM)

Algoritma Support Vector Machine (SVM) merupakan metode klasifikasi supervised learning yang didasarkan pada konsep yang meliputi margin hyperplane kernel yang diperkenalkan oleh Aronszajn pada tahun 1950, serta konsep-konsep pendukung lainnya[21]. SVM memiliki kemampuan untuk menangani permasalahan klasifikasi baik yang bersifat linier maupun non-linier. Dalam menangani permasalahan non-linier, SVM menggunakan konsep kernel pada ruang kerja berdimensi tinggi untuk mencari hyperplane yang dapat memaksimalkan margin antar label data. Konsep klasifikasi dengan Support Vector Machine adalah mencari hyperplane terbaik yang berfungsi sebagai pemisah antara dua kelas dalam ruang input[22].



Gambar 2. Ilustrasi metode Support Vector Machine

Pada Gambar 2 diatas menunjukkan Ilustrasi cara kerja metode Support Vector Machine mendapatkan garis hyperplane yang optimal dalam memisahkan data ke dua kelas, diperlukan perhitungan margin hyperplane dan penentuan titik maksimal. Dalam konteks mencari hyperplane pada SVM, dapat menggunakan persamaan (1).

$$(w \cdot x_i) + b = 0 \quad (1)$$

Data x_i , yang termasuk dalam kelas -1 dapat dijelaskan dengan rumus yang diberikan pada persamaan (2).

$$(w \cdot x_i + b) = 0 \leq 1, y_i = -1 \quad (2)$$

Data x_i yang masuk ke dalam kelas +1 dapat diungkapkan dengan persamaan (3) seperti berikut.

$$(w \cdot x_i + b) = 0 \geq 1, y_i = 1 \quad (3)$$

Dalam konteks klasifikasi menggunakan Support Vector Machine (SVM), seringkali terjadi situasi di mana kernel linear tidak memberikan hasil optimal dalam mengklasifikasikan data, yang mengakibatkan kinerja klasifikasi yang buruk. Untuk mengatasi hal ini, digunakan kernel non-linear dengan memanfaatkan teknik kernel trick[23]. Teknik ini melibatkan pemetaan data input ke dalam ruang fitur yang memiliki dimensi yang lebih tinggi, sehingga memungkinkan data input dipisahkan secara linear dan membentuk hyperplane yang optimal[24].

2.5 Evaluasi

Pada tahapan ini akan dilakukan evaluasi terhadap model Support Vector Machine. Evaluasi dilakukan setelah mendapatkan kombinasi terbaik dari fitur-fitur dan parameter dari pencarian optimal. Confusion Matrix memberikan representasi visual dari hasil klasifikasi dengan menunjukkan data yang diklasifikasikan secara benar maupun salah. Biasanya, Confusion Matrix digunakan untuk menghitung akurasi, presisi, dan recall. Namun, dalam penelitian ini, Hasil evaluasi akan dinilai berdasarkan nilai F1-score sebagai official score yang digunakan dalam shared task ini. rumus F1 score, seperti yang ditunjukkan pada persamaan (4) di bawah ini:

$$F_1 = 2 \times \left(\frac{\text{presisi} \times \text{recall}}{\text{presisi} + \text{recall}} \right) \quad (4)$$

F1 score digunakan sebagai skor resmi untuk menilai kinerja model dalam tugas ini. F1 score memberikan gambaran yang lebih holistik tentang kemampuan model dalam mengklasifikasikan data dengan baik dan menghindari kesalahan prediksi yang signifikan.

3. HASIL DAN PEMBAHASAN

3.1 Dataset

Data tweet Kaesang yang diperoleh dari crawling di Twitter melalui proses tahap seleksi data. Total data yang berhasil dihimpun adalah sebanyak 1.524 tweet. Data tersebut kemudian dibagi menjadi dua bagian, yaitu data Train sebanyak 300 Tweet data Test sebanyak 924 Tweet. Pembagian data ini bertujuan untuk keperluan data pelatihan dan pengujian. Data Testing sebagai data uji tidak pernah terlihat selama proses pelatihan. Selanjutnya, dalam proses pencarian model yang optimal, dari data Train yang berjumlah 300 tweet, akan dilakukan pembagian kembali dengan komposisi 80% data untuk pelatihan dan 20% data untuk validasi.

Tabel 2. Contoh hasil crawling dataset

No	Tweet	Label
1	resmi oleh PSI hari ini. Selamat kepada Kaesang Pangarep atas jabatan barunya! Semoga bisa membawa perubahan positif bagi partai dan masyarakat. #KaesangPangarep #PSI	Positif
2	Setelah @kaesangb diresmikan menjadi ketum, apakah pandanganmu terhadap @psi_id menjadi berubah? Saksikan #MetroTVKontroversi Kaesang Ketum PSI, Sinyal Dukungan Jokowi di link https://t.co/ZzwozVNmNO #Kaesang #KetumPSI #Pemilu2024 #Jokowi https://t.co/FLZbtBE7iA	Netral
3	Tempuh Cara Instan, Kiprah Politik Kaesang di PSI Tak Patut Dibanggakan https://t.co/DGXZ0j62E0	Negatif

Tabel 2 menunjukkan contoh hasil crawling data yang telah dilabeli Positif, Netral dan Negatif, dan data ini akan digunakan untuk tahap preprocessing.

3.2 Text Preprocessing

Tahapan ini adalah membersihkan teks pada dataset tweet agar siap untuk tahapan berikutnya. Tahapan ini melibatkan serangkaian proses dalam preprocessing teks seperti menghapus URL, mengurangi spasi yang berlebihan, menghilangkan hashtag, dan menghapus username. Setelah langkah pembersihan data selesai, langkah selanjutnya melibatkan kombinasi fitur preprocessing lainnya. Kombinasi fitur ini bertujuan untuk meningkatkan akurasi model machine learning. Beberapa fitur preprocessing yang dikombinasikan termasuk case folding, tokenizing, normalisasi, stopword dan stemming. Hasil pembersihan dan kombinasi fitur ini dapat dilihat dalam Tabel 3, yang menunjukkan hasil dari proses text preprocessing.

Tabel 3. Hasil proses text preprocessing

No	Proses	Sebelum Proses	Setelah proses
1	Cleaning	Tempuh Cara Instan, Kiprah Politik Kaesang di PSI Tak Patut Dibanggakan https://t.co/DGXZ0j62E0	Tempuh Cara Instan Kiprah Politik Kaesang di PSI Tak Patut Dibanggakan
2	Case Folding	Tempuh Cara Instan Kiprah Politik Kaesang di PSI Tak Patut Dibanggakan	tempuh cara instan kiprah politik kaesang di psi tak patut dibanggakan
3	Tokenizing	tempuh cara instan kiprah politik kaesang di psi tak patut dibanggakan	'tempuh', 'cara', 'instan', 'kiprah', 'politik', 'kaesang', 'di', 'psi', 'tak', 'patut', 'dibanggakan'
4	Normalisasi	'tempuh', 'cara', 'instan', 'kiprah', 'politik', 'kaesang', 'di', 'psi', 'tak', 'patut', 'dibanggakan'	'tempuh', 'cara', 'instan', 'kiprah', 'politik', 'kaesang', 'di', 'psi', 'tak', 'patut', 'dibanggakan'
5	Stopword	'tempuh', 'cara', 'instan', 'kiprah', 'politik', 'kaesang', 'di', 'psi', 'tak', 'patut', 'dibanggakan'	'tempuh', 'cara', 'instan', 'kiprah', 'politik', 'kaesang', 'psi', 'tak', 'patut', 'dibanggakan'
6	Stemming	'tempuh', 'cara', 'instan', 'kiprah', 'politik', 'kaesang', 'psi', 'tak', 'patut', 'dibanggakan'	'tempuh', 'cara', 'instan', 'kiprah', 'politik', 'kaesang', 'psi', 'tak', 'patut', 'bangga'

3.3 Eksperimen Setup

Eksperimen dilakukan dengan tujuan untuk menemukan model SVM yang optimal, yang mampu memberikan kinerja terbaiknya. Pada langkah ini, beberapa eksperimen akan dilakukan untuk mengevaluasi dampak dari penerapan beberapa langkah preprocessing teks, termasuk Stemming, Stopword dan Normalisasi. Komposisi dari langkah-langkah preprocessing ini dijelaskan dalam Tabel 4 pada Pengaturan Eksperimen berikut ini:

Tabel 4. Eksperimen setup

ID Eksperimen	Stemming	Stopword	Normalisasi
C1	✓	✗	✗
C2	✓	✓	✗
C3	✓	✓	✓



C4	✓	✗	✓
C5	✗	✓	✓
C6	✗	✗	✗
C7	✗	✓	✗
C8	✗	✗	✓

Pada Tabel 4 Experiment Setup, terlihat bahwa beberapa proses *preprocessing* diterapkan atau tidak diterapkan. Simbol centang menunjukkan bahwa tahap tersebut diterapkan, sedangkan simbol silang menunjukkan bahwa tahap tersebut tidak diterapkan dalam proses *text preprocessing*.

3.4 Fasttext

FastText merupakan teknik yang mengubah kalimat atau kata menjadi representasi vektor. Pada penelitian ini, FastText dikembangkan menggunakan data training, data validasi, dan data testing yang sudah digabungkan untuk membentuk kamus vektor. Kamus ini kemudian digunakan untuk mengubah teks pada data training, data validasi, dan data testing menjadi vector[25]. Model ini akan mengonversi kata-kata menjadi vektor dengan ukuran 156 (atau dapat disesuaikan untuk mendapatkan model bahasa yang optimal) menggunakan library *Gensim*. Proses vektorisasi ini dibuat dengan menghitung vektor rata-rata berdasarkan kamus vektor untuk setiap kalimat dalam tweet. Berikut adalah ilustrasi proses vektorisasi:

Kamus Word embedding Fasttext:

Kamus Vektor = ([[0.438479, -0.46834898, 1.4218011, ..., 0.6965722, -0.5345813, -0.5395061],
 [0.85484344, -0.59769094, 0.93916786, ..., -0.04030814, -0.29467, -0.30573305],
 [0.79240763, -0.5449833, 0.64742, ..., 0.70243037, -0.7618173, 0.3696998]
 [-0.1555549, 0.10468524, 0.09937996, ..., 0.05855399, -0.5391045, 0.22547351],
 [0.8579736, -0.31976995, 1.2195628, ..., 0.52612656, -0.34690166, -0.31667775],
 dtype=float32)

Kaesang = [0.86888105, -0.23392287, 1.0854172,, -0.42973515, -0.09607766]
 Pangarep = [-0.01836814, 1.1790547, 1.1848485,, -0.06350424, 0.52726656]
 Ketua = [-0.7761011, 1.1449217, -2.3063064,, -0.5181054, -0.2949075]
 PSI = [2.2331824e+00, -4.3644783e-01, 2.5847821e+00,, 3.4683180e-01]
 Kaesang pangarep ketua PSI = [2.58471698e-01, 2.86699355e-01, 2.29706049e-01,, 2.19789088e-01]

3.5 Parameter Tuning

Pada eksperimen sesuai dengan Tabel 4, dilakukan pencarian text preprocessing optimal untuk SVM. Banyak penelitian telah mengadopsi pendekatan penyetelan parameter untuk parameter *C* dan *gamma*, serta jenis kernel SVM yang digunakan untuk klasifikasi. Dalam penelitian ini, kernel yang digunakan adalah RBF, sebagaimana disarankan oleh penelitian terdahulu mengenai klasifikasi sentimen[14], yang menunjukkan bahwa kernel RBF memberikan hasil terbaik dalam klasifikasi. Oleh karena itu, parameter yang akan ditelusuri dalam penelitian ini adalah $C = [1, 10, 100, 1000]$ dan $gamma = [0.1, 0.01, 0.001, 0.0001]$, yang akan disesuaikan menggunakan metode grid search[26]. Hasil dari pencarian model optimal melalui proses seleksi fitur dan penyetelan parameter, yang dilakukan dengan menggunakan 10-fold Cross Validation, dapat ditemukan secara lengkap dalam Tabel 5. Parameter *C* dan *Gamma* yang terbaik diambil dari setiap skema eksperimen.

Tabel 5. Parameter tuning Grid Search 10-fold Cross Validation

ID Eksperimen	C	Gamma	Data Training		Data Validasi	
			Accuracy (%)	F1-Score (%)	Accuracy (%)	F1-Score (%)
C1	100	0.001	95.42	95.43	51.67	51.06
C2	1	0.01	77.50	77.74	56.67	56.75
C3	1	0.01	77.08	77.19	56.67	56.56
C4	10	0.001	76.25	76.37	63.33	63.95
C5	10	0.1	99.17	99.17	51.67	50.74
C6	1	0.01	82.50	82.51	56.67	57.14
C7	1	0.01	78.75	78.83	51.67	52.24
C8	1	0.01	82.08	82.19	60.00	60.66

Pada Tabel 5 dapat dilihat bahwa memilih nilai parameter yang tepat untuk *C* dan *gamma* sangat penting untuk mendapatkan model yang baik. Nilai yang terlalu tinggi untuk *C* dan *gamma* cenderung menyebabkan overfitting, sementara nilai yang terlalu rendah dapat menyebabkan underfitting. Eksperimen tertentu menunjukkan keseimbangan yang lebih baik antara performa data pelatihan dan validasi yang diinginkan pada model yang baik.

3.6 Eksperimen Penambahan Dataset Eksternal

Pada penelitian ini, dilakukan penambahan dataset eksternal untuk memperluas dataset pelatihan yang awalnya terbatas. Penambahan dataset eksternal tersebut bertujuan untuk meningkatkan representasi data pada dataset

pelatihan. Dengan demikian, diharapkan dapat meningkatkan kinerja model dalam mengklasifikasikan sentimen dengan dataset pelatihan yang lebih beragam. Data train Kaesang yang awalnya berjumlah 240 setelah dilakukan pembagian 80% dengan data validasi akan ditambahkan dengan dataset Covid yang diambil dengan beberapa percobaan dengan menambahkan untuk setiap label (Negatif, Positif, Netral) sebanyak 100, 200, dan 300 tweet masing-masing. Selain itu, hanya tweet yang memiliki minimal 8 kata yang dipilih untuk memastikan kualitas data yang digunakan. Pada tahap ini, parameter C dan gamma yang digunakan mengacu pada hasil optimal yang diperoleh dari eksperimen pada tabel 5 sebelumnya. Berikut adalah tabel hasil eksperimen penambahan dataset eksternal dengan jumlah data yang ditambahkan untuk setiap kelas dan hasil F1 Score data Validasi yang dihasilkan:

Tabel 6. Penambahan jumlah dataset eksternal

ID Ekperimen	Penambahan data Tiap Kelas			Jumlah Penambahan	F1-Score (%)
	Negatif	Positif	Netral		
C1	100	100	100	300	57.24
	200	200	200	600	56.67
	300	300	300	900	60.02
C2	100	100	100	300	64.35
	200	200	200	600	65.87
	300	300	300	900	69.52
C3	100	100	100	300	62.60
	200	200	200	600	62.60
	300	300	300	900	67.99
C4	100	100	100	300	57.96
	200	200	200	600	52.36
	300	300	300	900	59.83
C5	100	100	100	300	57.56
	200	200	200	600	59.95
	300	300	300	900	64.99
C6	100	100	100	300	61.47
	200	200	200	600	64.56
	300	300	300	900	60.74
C7	100	100	100	300	61.83
	200	200	200	600	62.70
	300	300	300	900	66.15
C8	100	100	100	300	55.31
	200	200	200	600	57.43
	300	300	300	900	49.55
C2 (Train Covid)	100	100	100	300	34.05
	200	200	200	600	45.81
	300	300	300	900	50.22
	873	463	6.664	8.000	16.03

Dengan penambahan dataset eksternal, terjadi peningkatan pada nilai F1-Score, Pada Tabel 6 terlihat bahwa terdapat tiga nilai F1-Score tertinggi. Pada ID eksperimen C2 dengan penambahan 900 data Covid, mendapatkan nilai F1-Score data validasi 69.52 %. Pada ID eksperimen C3 dengan dataset yang sama, mendapatkan nilai F1-Score data validasi adalah 67.99 %. Terakhir pada ID eksperimen C7 dengan dataset yang sama, mendapatkan nilai F1-Score data validasi adalah 66.15 %. Oleh karena itu, model yang paling optimal adalah model ID eksperimen C2, C3 dan C7 dengan masing-masing penambahan dataset covid 900.

Selain itu, dilakukan percobaan dengan pada ID eksperimen C2 dengan hanya menggunakan dataset covid tanpa menggunakan dataset train Kaesang, Dalam eksperimen ini, data yang ditambahkan terdiri dari 873 data negatif, 463 data positif, dan 6.664 data netral, dengan total 8.000 data covid. Hasilnya menunjukkan bahwa meskipun terdapat penambahan data dalam jumlah besar, nilai F1-Score yang diperoleh adalah 16.03%, jauh lebih rendah dibandingkan eksperimen sebelumnya, dengan penambahan 900 data yang mencapai 50.22%. Hal ini menunjukkan bahwa penambahan data dalam jumlah besar tidak selalu memberikan peningkatan performa yang diharapkan dan kualitas data atau distribusi kelas yang mempengaruhi hasil akhir.

3.7 Pengujian Menggunakan Data Uji

Model optimal diterapkan pada data uji yang belum pernah digunakan selama proses pelatihan klasifikasi menggunakan Support Vector Machine (SVM). Prediksi kelas positif, negatif, dan netral pada data uji yang berisi 924 tweet tersebut ditampilkan dalam bentuk tabel 7 berikut.

Tabel 7. Evaluasi terhadap data uji

ID Eksperimen	Run	F1-Score (%)	Accuracy (%)	Precision (%)	Recall (%)
C7	Run1	47.19	53.09	53.25	60.33
C2	Run2	53.59	62.73	53.01	59.62
C3	Run3	50.46	60.46	51.37	55.99

Pada tabel 7 menunjukkan hasil eksperimen bahwa Run2 (C2) memberikan performa terbaik dengan F1-Score sebesar 53.59%, akurasi 62.73%, precision 53.01%, dan recall 59.62%, mengindikasikan bahwa kombinasi dataset Kaesang dan Covid (900) pada Run2 memiliki keseimbangan yang baik antara precision dan recall. Sebaliknya, Run1 (C7) memiliki F1-Score terendah sebesar 47.19% dan akurasi 53.09%, namun memiliki recall tertinggi sebesar 60.33%, menunjukkan bahwa model pada Run1 lebih baik dalam menangkap semua instance yang relevan tetapi kurang presisi. Run3 (C3) berada di tengah-tengah dengan F1-Score 50.46% dan akurasi 60.46%, serta precision dan recall yang seimbang. Dengan demikian, Run2 (C2) adalah yang paling optimal dalam hal keseimbangan antara precision dan recall, menghasilkan F1-Score dan akurasi tertinggi di antara ketiga eksperimen.

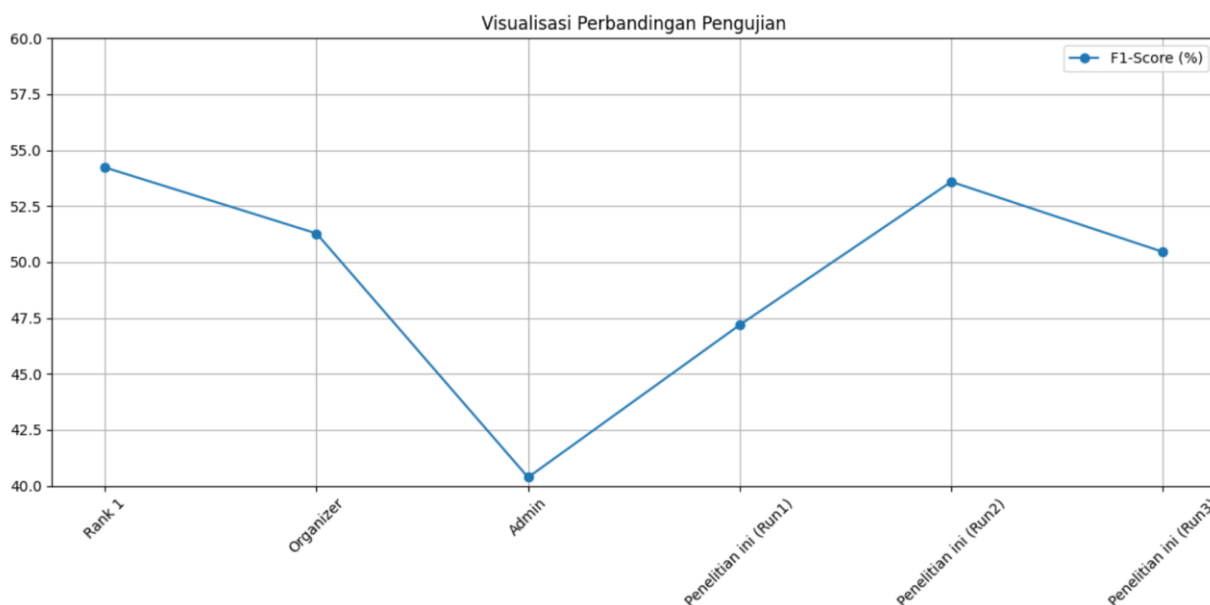
3.8 Perbandingan Pengujian

Ditinjau dari F1-score, akurasi, presisi, dan recall, penerapan SVM dengan fitur FastText mendapatkan nilai yang lebih baik dibandingkan dengan metode-metode sebelumnya. Tabel 8 memperlihatkan performa dari F1-score, akurasi, presisi, dan recall untuk penelitian sebelumnya serta hasil yang dicapai dalam penelitian ini.

Tabel 8. Perbandingan hasil pengujian

Tim	Run	Metode	F1-Score (%)	Accuracy (%)	Precision (%)	Recall (%)
Rank 1	Run2	SVM + Word2Vec	54.23	63.81	53.72	59.70
Organizer	Run2	SVM + TF - IDF	51.28	61.21	52.89	57.22
Admin	Run1	Baseline	40.38	45.45	49.53	48.80
Penelitian ini	Run2	SVM + Fasttext	53.59	62.73	53.01	59.62

3.9 Visualisasi Perbandingan Pengujian



Gambar 3. Visualisasi Perbandingan Pengujian

Pada Gambar 3 menampilkan visualisasi grafik seluruh pengujian pada penelitian ini yang akan dilakukan perbandingan antara beberapa tim pada tugas share task ini. Hasilnya menunjukkan bahwa metode SVM dengan Word2Vec menghasilkan kinerja terbaik dengan F1-Score sebesar 54.23% pada Rank1, selanjutnya diikuti oleh penelitian ini pada Run2 dengan metode SVM + FastText yang mencapai F1-Score sebesar 53.59%. Metode SVM dengan TF-IDF memiliki F1-Score sebesar 51.28% yang digunakan oleh Organizer, sementara baseline yang digunakan oleh Admin memiliki F1-Score sebesar 40.38%.

4. KESIMPULAN

Penelitian ini melakukan klasifikasi sentimen masyarakat terhadap Kaesang Pangarep yang menjadi ketua umum PSI di Twitter menggunakan metode SVM. Dataset yang digunakan diperoleh dari Twitter melalui teknik crawling. Data



pertama adalah tweet tentang sentimen terhadap Kaesang yang menjadi ketua PSI (Data Kaesang), terdiri dari sekitar 1.224 tweet yang diberi label positif, negatif, dan netral. Data Kaesang dibagi menjadi data Train sebanyak 300 tweet dan data Test sebanyak 924 tweet. Pembagian ini dilakukan untuk keperluan pelatihan dan pengujian. Data kedua adalah tweet tentang sentimen terhadap program vaksin Covid-19 (Data Covid), yang terdiri dari 8000 tweet dengan label positif, negatif, dan netral, dikumpulkan oleh penelitian lain. Karena jumlah data Train Kaesang sangat sedikit, beberapa tweet dari Data Covid digunakan untuk memperluas data Train Kaesang. Metode SVM dengan fitur word embedding FastText pada kernel RBF digunakan, dan grid search diimplementasikan untuk mencari parameter C dan gamma terbaik dengan 10-fold cross validation. Penambahan dataset eksternal menunjukkan peningkatan pada nilai F1-Score. Tiga nilai F1-Score tertinggi diperoleh pada eksperimen dengan ID C2 (69.52%), C3 (67.99%), dan C7 (66.15%) pada data validasi. Ketiga model tersebut diuji pada data uji, menghasilkan F1-Score masing-masing sebesar 53.59%, 50.46%, dan 47.19% untuk ID eksperimen C2, C3, dan C7 secara berurutan. Dari hasil tersebut dapat disimpulkan bahwa penambahan dataset eksternal cukup efektif dalam meningkatkan nilai F1-Score, menunjukkan bahwa SVM dengan kernel RBF sangat efektif dalam mengklasifikasikan sentimen tweet terkait Kaesang Pangarep sebagai Ketua PSI di Twitter.

REFERENCES

- [1] "Tentang PSI - Partai Solidaritas Indonesia." Accessed: Apr. 09, 2024. [Online]. Available: <https://psi.id/tentang-psi/>
- [2] N. Muhamad, "Kaesang Jadi Ketua PSI, Hal Baik atau Dinasti Politik? Ini Menurut Publik," *databoks.com*. Accessed: Apr. 05, 2024. [Online]. Available: <https://databoks.katadata.co.id/datapublish/2023/10/20/kaesang-jadi-ketum-psi-hal-baik-atau-dinasti-politik-ini-menurut-publik>
- [3] A. N. Yahya, "Pro dan Kontra Kaesang Pangarep Jadi Ketua PSI," *Kompas.com*. Accessed: Apr. 05, 2024. [Online]. Available: <https://nasional.kompas.com/read/2023/09/26/16000031/pro-dan-kontra-kaesang-pangarep-jadi-ketum-psi>
- [4] S. A. Mukti M Kusairi, "SVM Method with FastText Representation Feature for Classification of Twitter Sentiments Regarding the Covid-19 Vaccination Program," *Jurnal Teknologi Informasi dan Komunikasi*, 2022, doi: <https://doi.org/10.31849/digitalzone.v13i2.11531>.
- [5] M. Sahbuddin and S. Agustian, "Support Vector Machine Method with Word2vec for Covid-19 Vaccine Sentiment Classification on Twitter," *JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING*, vol. 6, no. 1, pp. 288–297, Jul. 2022, doi: 10.31289/jite.v6i1.7534.
- [6] A. M. Rahat, A. Kahir, and A. K. M. Masum, "Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset," 2019.
- [7] M. D. H. Jasy, S. Al Hasan, M. I. K. Sagor, A. Noman, and J. M. Ji, "A Performance Evaluation of Sentiment Classification Applying SVM, KNN, and Naive Bayes," in *Proceedings - 2021 International Conference on Computing, Networking, Telecommunications and Engineering Sciences Applications, CoNTESA 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 56–60. doi: 10.1109/CoNTESA52813.2021.9657115.
- [8] A. Damayunita, R. S. Fuadi, and C. Juliane, "Comparative Analysis of Naive Bayes, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) Algorithms for Classification of Heart Disease Patients," *Jurnal Online Informatika*, vol. 7, no. 2, pp. 219–225, Dec. 2022, doi: 10.15575/join.v7i2.919.
- [9] K. Munawaroh, "Performance Comparison of SVM, Naive Bayes, and KNN Algorithms for Analysis of Public Opinion Sentiment Against COVID-19 Vaccination on Twitter," *Journal of Advances in Information Systems and Technology*, vol. 4, no. 2, 2022.
- [10] R. Yunita and M. Kamayani, "Perbandingan Algoritma SVM Dan Naive Bayes Pada Analisis Sentimen Kebijakan Penghapusan Kewajiban Skripsi," *Indonesian Journal of Computer Science*, 2023.
- [11] A. Nurdin, B. Anggo, S. Aji, A. Bustamin, and Z. Abidin, "PERBANDINGAN KINERJA WORD EMBEDDING WORD2VEC, GLOVE, DAN FASTTEXT PADA KLASIFIKASI TEKS," *Jurnal TEKNOKOMPAK*, vol. 14, no. 2, p. 74, 2020.
- [12] S. Agustian, R. Abdillah, and M. Irfansyah, "Arah baru penelitian klasifikasi teks: Memaksimalkan Kinerja Klasifikasi Sentimen dari Data Terbatas," *MALCOM (Indonesia Journal of Machine Learning and Computer)*, vol. 4, no. 3, Jul. 2024.
- [13] M. Ihsan, Benny Sukma Negara, and Surya Agustian, "LSTM (Long Short Term Memory) for Sentiment COVID-19 Vaccine Classification on Twitter," *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, vol. 13, no. 1, pp. 79–89, May 2022, doi: 10.31849/digitalzone.v13i1.9950.
- [14] M. Sahbuddin and S. Agustian, "Support Vector Machine Method with Word2vec for Covid-19 Vaccine Sentiment Classification on Twitter," *JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING*, vol. 6, no. 1, pp. 288–297, Jul. 2022, doi: 10.31289/jite.v6i1.7534.
- [15] Ash Shiddicky and Surya Agustian, "Analisis Sentimen Masyarakat Terhadap Kebijakan Vaksinasi Covid-19 pada Media Sosial Twitter menggunakan Metode Logistic Regression," *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 3, no. 2, pp. 99–106, Aug. 2022, doi: 10.37859/coscitech.v3i2.3836.
- [16] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," Jul. 2016, [Online]. Available: <http://arxiv.org/abs/1607.04606>
- [17] M. D. Rhman, A. Djunaidy, and F. Mahananto, "Penerapan Weighted Word Embedding pada Pengklasifikasian Teks Berbasis Recurrent Neural Network untuk Layanan Pengaduan Perusahaan Transportasi," 2021.
- [18] E. Bartz, T. Bartz-Beielstein, M. Zaefferer, and O. Mersmann, "Hyperparameter Tuning for Machine and Deep Learning with R A Practical Guide." 2023
- [19] M. Fajri and A. Primajaya, "Komparasi Teknik Hyperparameter Optimization pada SVM untuk Permasalahan Klasifikasi dengan Menggunakan Grid Search dan Random Search," 2023. [Online]. Available: <http://jurnal.polibatam.ac.id/index.php/JAIC>



- [20] A. Toha, P. Purwono, and W. Gata, “Model Prediksi Kualitas Udara dengan Support Vector Machines dengan Optimasi Hyperparameter GridSearch CV,” *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 4, no. 1, pp. 12–21, May 2022, doi: 10.12928/biste.v4i1.6079.
- [21] A. A. Soebroto, “Buku Ajar AI, Machine Learning & Deep Learning,” 2019. [Online]. Available: <https://www.researchgate.net/publication/348003841>
- [22] F. Putrawansyah, “Penerapan Metode Support Vector Machine Terhadap Klasifikasi Jenis Jambu Biji,” *JIKO (Jurnal Informatika dan Komputer)*, vol. 8, no. 1, p. 193, Feb. 2024, doi: 10.26798/jiko.v8i1.988.
- [23] H. C. Husada and A. S. Paramita, “Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM),” *Teknika*, vol. 10, no. 1, pp. 18–26, Feb. 2021, doi: 10.34148/teknika.v10i1.311.
- [24] B. Darma Darma Setiawan and Y. Arum Sari, “Klasifikasi Pola Sidik Bibir Untuk Menentukan Jenis Kelamin Manusia Dengan Metode Gray Level Co-Occurrence Matrix Dan Support Vector Machine,” 2019. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [25] M. Mustasaruddin, E. Budianita, M. Fikry, and F. Yanto, “Klasifikasi Sentiment Review Aplikasi MyPertamina Menggunakan Word Embedding FastText dan SVM (Support Vector Machine),” *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 4, no. 3, p. 526, Mar. 2023, doi: 10.30865/json.v4i3.5695.
- [26] S. Azhar, M. Fikry, S. Agustian, and I. Afrianty, “Klasifikasi Sentimen Masyarakat di Twitter Terhadap Ganjar Pranowo dengan Metode Support Vector Machine,” *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 4, no. 3, pp. 1660–1667, 2023, doi: 10.30865/klik.v4i3.1537.