Jurnal Pendidikan dan Teknologi Indonesia (JPTI)
Vol. 5, No. 7, Juli 2025, Hal. 1915-1924
DOI: https://doi.org/10.52436/1.jpti.888
p-ISSN: 2775-4227
e-ISSN: 2775-4219

Analisis Sentimen Media Sosial terhadap Isu Pagar Laut di Indonesia Menggunakan Algoritma Support Vector Machine dan Logistic Regression

Nanda Perdana[*1], Handri Santoso[2]

[1,2]Information Technology, Pradita University, Indonesia
Email: [1]nanda.perdana@student.pradita.ac.id, [2]handri.santoso@pradita.ac.id

Abstrak

Penelitian ini mencakup analisis sentimen pada media sosial terkait isu pagar laut menggunakan algoritma *Logistic Regression (LR)* dan *Support Vector Machine (SVM)*. Urgensi dari penelitian ini terletak pada meningkatnya perdebatan publik mengenai isu pagar laut yang berpotensi memicu ketegangan sosial dan politik apabila tidak dipahami dengan baik. Berkaitan dengan hal tersebut, sumber data pada penelitian ini diperoleh dari media sosial Twitter, Instagram, Facebook, dan Tiktok. Data yang telah terkumpul kemudian melalui tahap pra pemrosesan dan *labelling* menggunakan *VADER*. Hasil dari tiga rasio yang digunakan menunjukkan rasio 0,7 atau 7:3 adalah yang terbaik, dengan akurasi *SVM* 0.985382 dan akurasi LR 0.988881. Secara keseluruhan kedua algoritma memberikan hasil yang sama baiknya dan seimbang. Kesimpulan tersebut dapat dilihat dari nilai evaluasi yang meliputi *precision, recall,* dan *F1-score*. Hasil penelitian ini dapat digunakan sebagai bahan diskusi maupun analisis lebih lanjut untuk menyusun strategi komunikasi publik yang lebih efektif. Selain itu, hasil penelitian ini juga dapat digunakan oleh pemangku kebijakan, peneliti, maupun lembaga terkait untuk memantau dinamika persepsi publik.

Kata kunci: *Analisis Sentimen, Logistic Regression, Media Sosial, Support Vector Machine.*

Analyzing Sentiment in Indonesian Social Media Using Support Vector Machine and Logistic Regression Approaches on "Pagar Laut" Issue

Abstract

This research covers sentiment analysis on social media related to the sea fence issue using Logistic Regression (LR) and Support Vector Machine (SVM) algorithms. The urgency of this research lies in the increasing public debate over the sea fence issue which has the potential to trigger social and political tensions if not well understood. In this regard, the data sources in this study were obtained from social media Twitter, Instagram, Facebook, and Tiktok. The data that has been collected then goes through a pre-processing and labeling stage using VADER. The results of the three ratios used show that the ratio of 0.7 or 7:3 is the best, with SVM accuracy of 0.985382 and LR accuracy of 0.988881. Overall, both algorithms give equally good and balanced results. The conclusion can be seen from the evaluation value which includes precision, recall, and F1-score. The results of this study can be used as material for further discussion and analysis to develop a more effective public communication strategy. In addition, the results of this study can also be used by policy makers, researchers, and related institutions to monitor the dynamics of public perception.

Keywords: *Logistic Regression, Sentiment Analysis, Social Media, Support Vector Machine.*

1. PENDAHULUAN

Seiring dengan perkembangan yang pesat di era digital, akses untuk menyampaikan saran maupun kritik di publik menjadi lebih mudah melalui media sosial. Media sosial yang seringkali digunakan oleh publik meliputi Tiktok, Facebook, Instagram, dan Twitter atau yang sekarang dikenal dengan X. Media sosial tersebut dapat menjadi wadah bagi komunitas publik untuk menyampaikan pendapat terkait isu yang sedang ramai dibicarakan, salah satunya yaitu isu "pagar laut". Fenomena ini menunjukkan bagaimana platform digital telah mengubah cara masyarakat berinteraksi dan menyuarakan opininya. Opini adalah ekspresi dari pendapat yang bisa bertentangan dengan topik dan dapat disampaikan langsung secara lisan maupun tulisan [1].

Lebih jauh, penggunaan media sosial juga memiliki pengaruh yang signifikan dalam membentuk opini maupun persepsi masyarakat terhadap suatu hal, baik dalam hal sosial maupun politik. Media sosial kini berpengaruh cukup besar dalam pembentukan opini publik yang jika disalahgunakan oleh pihak tertentu, dapat digunakan sebagai sarana kontes politik. Selain itu, Penggunaan media sosial yang kurang bijak juga dapat

p-ISSN: 2775-4227
e-ISSN: 2775-4219

menimbulkan kericuhan yang berpotensi memecah belah persatuan [2]. Hal tersebut dapat terjadi, salah satunya jika berkaitan dengan interaksi sosial pada komunitas online yang berfokus pada diskusi politik [3].

Isu yang menjadi fokus penelitian ini yaitu "pagar laut" pertama kali muncul di bulan Agustus 2024 pada media sosial di Indonesia dan telah menimbulkan berbagai macam reaksi dari masyarakat. Sebagian besar masyarakat menduga adanya kasus korupsi salah satu perusahaan besar di Indonesia pada isu ini. Akibatnya, selama beberapa waktu media sosial penuh dengan opini publik terkait isu tersebut, menciptakan gelombang diskusi yang besar di berbagai platform digital. Intensitas pembahasan isu ini menunjukkan tingginya perhatian publik terhadap masalah di Indonesia.

Berkaitan dengan fenomena tersebut, muncul pertanyaan mengenai bagaimana sebenarnya opini publik pada isu "pagar laut". Untuk menjawab pertanyaan tersebut, penelitian ini bertujuan untuk menganalisis sentimen publik pada media sosial, khususnya terkait isu "pagar laut". Analisis sentimen sendiri merupakan sebuah teknik mengumpulkan data dari berbagai sumber media sosial dengan tujuan mendapatkan umpan balik dari para penggunanya [4]. Teknik ini memungkinkan peneliti untuk memahami secara lebih mendalam bagaimana masyarakat merespons dan memaknai isu tersebut.

Beberapa algoritma dalam pembelajaran mesin yang sering digunakan untuk analisis sentimen meliputi *Naive Bayes, Support Vector Machine, Logistic Regression, K-Nearest Neighbor,* dan *Random Forest*. Analisis sentimen yang digunakan pada penelitian ini merupakan salah satu cabang dari *Natural Language Processing (NLP)*. Untuk mendapatkan hasil yang komprehensif dan akurat, peneliti menggunakan dua metode analisis, yaitu *Support Vector Machine (SVM)* dan *Logistic Regression (LR)*. Penggunaan dua metode ini diharapkan dapat memberikan pemahaman yang lebih mendalam dan hasil analisis yang lebih akurat.

Penelitian yang dilakukan oleh P. Arsi and R. Waluyo,pada 2021 dengan judul "Analisis Sentimen Wacana Pemindahan Ibu Kota Indonesia Menggunakan Algoritma Support Vector Machine (SVM)" menunjukkan hasil yang cukup tinggi, yaitu 96,68% akurasi, 95,82% *precision*, dan 94,04% nilai untuk *recall* [5]. Penelitian tersebut menggunakan dataset dari media sosial Twitter dengan jumlah data 1.116 *tweets*. Selain itu, penelitian lainnya yang dilakukan oleh Idris dkk. pada 2023 menunjukkan hasil akurasi pada 98% dan nilai *f-1 score* berada pada 0,98 atau 98% [6]

Penelitian lainnya yang dilakukan oleh B. Ramadhani, R. R. Suryono, and K. Kunci pada tahun 2024 dengan judul "Komparasi Algoritma Naive Bayes dan Logistic Regression Untuk Analisis Sentimen Metaverse" menunjukkan hasil yang cukup tinggi juga, yaitu 95% akurasi, disertai dengan 94% *precision*, dan 93% *recall* [7]. Penelitian tersebut juga menggunakan data dari media sosial Twitter dengan jumlah 6.728 data komentar dari masyarakat terkait dengan metaverse.

Beberapa penelitian tersebut menjadi dasar pemilihan algoritma yang digunakan pada penelitian ini, yaitu *Support Vector Machine* (SVM) dan *Logistic Regression* (LR). Lebih lanjut terkait pembahasan kedua algoritma tersebut pada Sub Bab 2.1 dan 2.2 berikut.

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif dengan metode analisis sentimen untuk mengolah data dari media sosial Tiktok, Instagram, Twitter, dan Facebook. Jumlah data yang digunakan dalam penelitian ini yaitu 118.124 yang diperoleh melalui ekstraksi *post* dan *comment* dari empat media sosial tersebut. Metode SVM dan LR dapat dilihat pada bagian berikut.

2.1. *Support Vector Machine (SVM)*

SVM merupakan algoritma yang sering digunakan untuk memecahkan masalah klasifikasi, salah satunya analisis sentimen [4]. SVM sering digunakan pada analisis sentimen karena memiliki akurasi yang lebih baik dibanding dengan algoritma lainnya. Prinsip utama dari SVM adalah dengan mencari *optimal hyperplane* yang memaksimalkan pemisahan antara kelas-kelas yang berbeda dalam ruang fitur [8]. Dengan kata lain, SVM berupaya menemukan batas keputusan yang paling ideal untuk memisahkan kelas-kelas tersebut dengan tujuan meminimalkan kesalahan klasifikasi. Keunggulan SVM dalam menangani data berdimensi tinggi dan kemampuannya untuk menggunakan *kernel trick* juga berkontribusi pada performanya yang baik dalam tugas-tugas analisis sentimen yang kompleks [9]. Berikut adalah persamaan dari algoritma SVM.

$$(W.X_i) + b = 0 \quad (1)$$

$$(W.X_i + b) \geq 1 \quad (2)$$

p-ISSN: 2775-4227
e-ISSN: 2775-4219

$$(W.X_i + b) \leq 1 \quad (3)$$

Persamaan 1 merupakan *hyperplane* yang berfungsi sebagai batas keputusan antara dua kelas (positif dan negatif). Persamaan 2 merupakan margin untuk kelas positif (+1). Persamaan 3 merupakan margin untuk kelas negatif (-1).

2.2. *Logistic Regression (LR)*

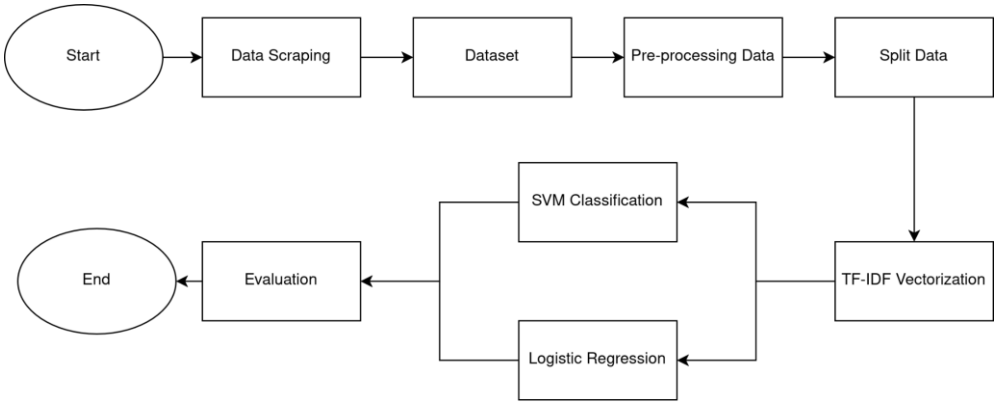
LR adalah salah satu metode pada *Machine Learning* dengan cara kerja menangkap sebuah vektor variabel dan mengevaluasi koefisien atau bobot untuk setiap variabel input [10]. Proses ini bertujuan untuk memodelkan probabilitas kelas berdasarkan kombinasi linear variabel input yang telah diberi bobot. Output dari model LR berupa probabilitas, yang kemudian digunakan untuk memprediksi kelas yang paling mungkin. Karena kemampuannya untuk memberikan interpretasi yang jelas terhadap pengaruh setiap variabel input terhadap probabilitas kelas, LR sering digunakan berbagai aplikasi, salah satunya analisis sentiment. Berikut adalah persamaan dari algoritma LR.

$$logit(S) = b_0 + b_1M_1 + b_2M_2 + b_3M_3 \ldots . b_kM_k (4)$$

Persamaan 4 merupakan rumus logit yang memodelkan hubungan antara probabilitas suatu kejadian dan kombinasi linear dari variabel prediktor, di mana logit dari probabilitas tersebut merupakan fungsi linear dari nilia-nilai prediktor yang dikalikan dengan koefisiennya masing-masing.

2.3. Tahapan Penelitian

Tahapan dari penelitian ini dapat dilihat pada gambar Gambar 1.



Gambar 1. Flowchart Penelitian

Pada Gambar 1, penelitian dimulai dengan melakukan pengumpulan data atau *data scrapping* dan hasilnya berupa dataset. Setelah dataset didapatkan, selanjutnya dilakukan *pre-processing data* untuk membersihkan elemen-elemen seperti tanda baca dan karakter maupun tag dari html yang tidak diperlukan. Setelah data dibersihkan, selanjutnya dibagi dengan beberapa perbandingan untuk data latih dan data uji. Setiap data yang sudah dibagi akan digunakan pada *TF-IDF Vectorization* untuk pemberian skor berdasarkan frekuensi kemunculannya. Setelah diberi skor, data digunakan pada kedua algoritma dan hasilnya dapat dilihat pada bagian evaluasi.

2.3.1 Data Scraping

Tahap pertama yaitu mengumpulkan data atau *data scraping*. Proses ini menerapkan pengambilan konten post, komentar, dan balasan dari komentar yang ada. Setiap data diambil dan dipetakan sesuai dengan kebutuhan. Misalnya waktu, konten, nama pengguna, dan tautan. Setelah semua data diambil, data disimpan dalam file *csv* untuk memudahkan proses *load* data ketika sudah diimplementasikan pada kode program. Jumlah data yang berhasil dikumpulkan dan digunakan pada penelitian ini berjumlah 118.124 (seratus delapan belas ribu).

2.3.2 Pre-processing Data

Tahap kedua setelah dataset diperoleh yaitu pra-pemrosesan data. Tahap ini meliputi pembersihan data dari tag-tag html dan melakukan tokenisasi atau memecah kalimat menjadi kata-kata. Selain pembersihan dari tag-tag

p-ISSN: 2775-4227
e-ISSN: 2775-4219

html, dilakukan juga *stop words removal* atau pembuangan kata-kata yang tidak signifikan atau tidak memiliki deskripsi yang jelas. Jenis *stop words removal* yang digunakan adalah dengan library nltk untuk teks berbahasa Indonesia. Contoh data yang sudah melalui proses pra-pemrosesan dapat dilihat pada Tabel 1.

Tabel 1. *Pre-processing* data

Original	Processed
\nNanti urusan reklamasi	nanti urusan reklamasi
<p>pagar laut menjadi isu yang sangat ramai diperbincangkan<p>	pagar laut menjadi isu yang sangat ramai diperbincangkan

Setelah proses pra-pemrosesan data selesai, dataset selanjutnya dilakukan proses labeling dengan tiga nilai sentimen, yaitu *neutral, positive,* dan *negative*. Proses pelabelan data dilakukan dengan menggunakan *Valence Aware Dictionary and sEntiment Reasoner (VADER)*. VADER adalah metode yang menggabungkan pendekatan *human based* dengan validasi empiris yang didasarkan pada analisis kualitatif dan pengalaman manusia [11].

2.3.3 Split Data

Tahap selanjutnya yaitu *split data* atau pembagian data antara data latih dan data tes. Pembagian data pada penelitian ini memiliki tiga rasio latih:tes, yaitu 80:20, 70:30, dan 60:40. Penggunaan tidak rasio ini bertujuan untuk menemukan kombinasi paling optimal dengan tingkat akurasi dan *precision* paling tinggi dari kedua algoritma yang digunakan yaitu SVM dan LR.

2.3.4 TF-IDF Vectorizer

Tahap selanjutnya yaitu penerapan TF-IDF vectorizer. TF-IDF merupakan kependekan dari *inverse document frequency (IDF)* dan *term frequency (TF)*[9]. Cara kerja dari metode ini adalah dengan memberikan skor pada setiap kata, sesuai dengan frekuensi kemunculannya. Persamaan dari TF-IDF vectorizer dapat dilihat pada Persamaan 5 berikut.

$$tf\ idf\ (t,d,D) = tf(t,d) X\ idf\ (t,D) \qquad (5)$$

Di mana t adalah *terms*, d adalah *document*, dan D adalah *collection of document*. Semakin sedikit kemunculan teks pada dokumen atau corpus, maka nilai IDF akan semakin tinggi dan berpotensi meningkatkan skor TF-IDF jika frekuensi istilah (TF) dalam dokumen tersebut juga tinggi.

2.3.5 Implementasi Algoritma

Tahap berikutnya yaitu implementasi algoritma SVM dan LR. Proses pelatihan model masing-masing algoritma dilakukan sebanyak tiga kali sesuai dengan jumlah tiga rasio yang digunakan. Sebelum diimplementasikan, data yang sudah dibagi akan dilakukan *scaling* menggunakan *GridSearch* untuk optimasi hyperparameter sehingga menghasilkan kombinasi terbaik. Pada implementasi SVM, *GridSearch* ini digunakan untuk mendapatkan parameter SVM terbaik, begitu juga dengan implementasi pada LR.

2.3.6 Evaluasi

Tahap terakhir yaitu evaluasi dari model. Evaluasi dilakukan untuk melihat akurasi dan *precision* serta F-1 score sehingga didapatkan performa model apakah sudah cukup bagus, underfit, ataupun overfit. Underfit sendiri berarti hasil akurasi *training* berada jauh di bawah akurasi *testing*, sedangkan overfit berarti akurasi *training* berada jauh di atas akurasi *testing*. Dari hasil tersebut, maka dapat dilakukan pelatihan ulang model dengan melakukan beberapa optimasi, seperti pra-pemrosesan data yang lebih ketat dan tokenisasi, serta jumlah iterasi maksimum yang ditingkatkan pada model di tahap pelatihan. Hasil *tuning* yang menunjukkan paling tinggi dan konsisten, akan dipilih untuk digunakan sebagai model akhir.

2.3.7 Lingkungan Pengembangan

Penelitian ini dilakukan dalam lingkungan tertutup, artinya hanya menggunakan satu perangkat utama dan semua kode hanya dapat diakses dari perangkat tersebut. Perangkat yang digunakan yaitu sebuah Laptop ACER Nitro 5 AN515-57 dengan spesifikasi CPU Intel Core I5-11400H, GPU NVIDIA RTX 3050, RAM 16GB DDR4

p-ISSN: 2775-4227
e-ISSN: 2775-4219

3200mhz, dan penyimpanan sebesar 1 *terabyte*. Selain perangkat tersebut, beberapa *library* yang digunakan dalam bahasa pemrograman Python 3.11 yaitu *Pandas, NLTK, scikit-learn, re, bs4,* dan *datetime*.

3. HASIL DAN PEMBAHASAN

3.1. Data Scraping

Data yang digunakan dalam penelitian ini dikumpulkan menggunakan kode bahasa pemrograman Python dan *API* dari platform yang tersedia. Media sosial yang digunakan meliputi Twitter, Instagram, Facebook, dan Tiktok. Data yang berhasil diekstrak dari keempat media sosial tersebut berjumlah 118.124. Data yang diambil berada dalam rentang 1 Agustus 2024 - 31 Januari 2025 dan disimpan dalam file .csv untuk memudahkan proses *load data* di kode pemrograman Python. Contoh data beserta meta datanya dapat dilihat pada Gambar 2 berikut.

Pada Gambar 2, contoh dataset meliputi beberapa kolom yaitu source atau sumber media sosial, content atau teks yang akan dianalisis, link atau tautan ke *post* tersebut, timestamp atau waktu publikasi post tersebut dalam format milidetik, serta type yang mencakup tipe apakah sebuah post, retweet, atau reply.

	source	content	link	timestamp	type
0	twitter	Di hari ketiga, mereka mencium bau ikan bakar ...	https://twitter.com/wiopepods/status/182156947...	1723130981000	reply
1	twitter	Tdk jelas pemasangan pagar bambu sepanjang +/-...	https://twitter.com/Boediantar4/status/1830907...	1725357251000	tweet
2	twitter	RT @Boediantar4: Tdk jelas pemasangan pagar ba...	https://twitter.com/shoufi_aslelmjg/status/183...	1725378107000	retweet
3	twitter	@Boediantar4 Nah...\nNanti urusan reklamasi su...	https://twitter.com/DMieaceh80579/status/18310...	1725394418000	reply
4	twitter	@Boediantar4 Klo cuma dari bambu, caranya nela...	https://twitter.com/justicemyfoot/status/18310...	1725397820000	reply
5	twitter	Oknum2 bupati camat lurah kades polisi tni tid...	https://twitter.com/3Muzaeny88005/status/18311...	1725406689000	quote
6	twitter	Gileee.. \nZaman sekarang.. \nSeenaknya maen p...	https://twitter.com/AdeSPerwiraII24/status/183...	1725411607000	quote
7	twitter	Nelayan bersatu lawan, bongkar tirai bambu CIN...	https://twitter.com/B0_Maskudin/status/1831145...	1725414157000	quote
8	twitter	@Boediantar4 Hidup @jokowi raja jawa.\nIndones...	https://twitter.com/CcVb635703/status/18311478...	1725414637000	reply
9	twitter	RT @Boediantar4: Tdk jelas pemasangan pagar ba...	https://twitter.com/TulangRakyat/status/183116...	1725418645000	retweet

Gambar 2. Preview Dataset

3.2. Pre-processing Data

Tahapan preprocessing data memiliki beberapa tahapan di dalamnya. Tahapan pertama yaitu *cleaning* data dari tag-tag HTML yang tidak diperlukan, misalnya <p>, <h1>, \n, dan lain sebagainya. Contoh data yang sudah dibersihkan dapat dilihat pada Tabel 1. Pembersihan data juga mencakup mengubah semua kata pada kalimat menjadi *lowercase* untuk memudahkan proses tokenisasi dan implementasi pada algoritma SVM dan LR, supaya memastikan kata "Saya" dan "saya" memiliki nilai yang sama. Selain membersihkan dari tag-tag html, pembersihan juga dilakukan untuk *mention, direct mention, link*, dan karakter special seperti *emoticon*. Semua tahap pembersihan data ini dilakukan menggunakan kode python dengan *regular expression* untuk memudahkan pencarian berdasarkan pola yang sesuai dengan *regular expression* yang diterapkan. Berikut preview dari kode yang mengimplementasikan filter dengan *regular expression* dapat dilihat pada Gambar 3.

Pada gambar 3, *filtering* yang dilakuan mencakup menghapus pola teks retweet, tag html, serta url. Masih dalam tahap preprocessing data, selanjutnya yaitu tokenizing atau memecah kalimat menjadi daftar kata. Proses ini diperlukan untuk pemberian bobot dan labelling data pada algoritma VADER. Contoh kalimat pada konten media sosial yang sudah dilakukan tokenisasi dapat dilihat pada Gambar 4.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

```python
def clean_text(text):
    if pd.isnull(text):
        return ""
    text = BeautifulSoup(text, "html.parser").get_text()
    text = re.sub(r'RT @\w+:', '', text)
    text = re.sub(r'@\w+', '', text)
    text = re.sub(r'http\S+|www\S+', '', text)
    text = re.sub(r'[^a-zA-Z\s]', '', text)
    text = text.lower()

    return text
```

Gambar 3.Filtering Regular Expression

```
Contoh 1:
Teks asli: Di hari ketiga, mereka mencium bau ikan bakar yang sangat sedap. Bapak bertanya pada Nenek, dan Nenek menjelaskan bahwa
bau itu berasal dari Kemamang, makhluk menyeramkan dengan tubuh merah menyala yang suka memanggang ikan di atas kepalanya. 🐈‍🔥
Tokens sebelum filtering: ['di', 'hari', 'ketiga', 'mereka', 'mencium', 'bau', 'ikan', 'bakar', 'yang', 'sangat', 'sedap', 'bapak',
 'bertanya', 'pada', 'nenek', 'dan', 'nenek', 'menjelaskan', 'bahwa', 'bau', 'itu', 'berasal', 'dari', 'kemamang', 'makhluk', 'meny
eramkan', 'dengan', 'tubuh', 'merah', 'menyala', 'yang', 'suka', 'memanggang', 'ikan', 'di', 'atas', 'kepalanya']
Tokens setelah filtering: ['ketiga', 'mencium', 'bau', 'ikan', 'bakar', 'sedap', 'nenek', 'nenek', 'bau', 'berasal', 'kemamang', 'm
akhluk', 'menyeramkan', 'tubuh', 'merah', 'menyala', 'suka', 'memanggang', 'ikan', 'kepalanya']

Contoh 2:
Teks asli: Tdk jelas pemasangan pagar bambu sepanjang +/-4-5km di sepanjang pantai dari Tangerang ke Jakarta itu siapa yg pasang da
n utk apa, kasihan nelayan di sepanjang pantai tsb, tdk bebas lagi mencari nafkah utk keluarganya.😭😭😭 https://t.co/rzYrgArLE9
Tokens sebelum filtering: ['tdk', 'jelas', 'pemasangan', 'pagar', 'bambu', 'sepanjang', 'km', 'di', 'sepanjang', 'pantai', 'dari',
'tangerang', 'ke', 'jakarta', 'itu', 'siapa', 'yg', 'pasang', 'dan', 'utk', 'apa', 'kasihan', 'nelayan', 'di', 'sepanjang', 'pantai
', 'tsb', 'tdk', 'bebas', 'lagi', 'mencari', 'nafkah', 'utk', 'keluarganya']
Tokens setelah filtering: ['tdk', 'pemasangan', 'pagar', 'bambu', 'km', 'pantai', 'tangerang', 'jakarta', 'pasang', 'utk', 'kasihan
', 'nelayan', 'pantai', 'tsb', 'tdk', 'bebas', 'mencari', 'nafkah', 'utk', 'keluarganya']
```

Gambar 4. Tokenizer

Setelah proses tokenisasi, selanjutnya dalam tahap preprocessing yaitu pelabelan data. Pelabelan data ini diperlukan agar data latih dan data tes memiliki target sentimen yang seharusnya. Proses ini menggunakan algoritma VADER dengan tiga kategori sentimen, yaitu *neutral, positive,* dan *negative*. Pemberian kriteria untuk *neutral, positive,* dan *negative* dapat dilihat pada Tabel 2. Total dari masing-masing sentiment hasil labeling data dapat dilihat pada Tabel 3.

Tabel 2. Kriteria Label Data

Sentimen	Kriteria Compound Score
positive	>= 0.05
negative	<= 0.05
neutral	0.05 < *compound score* < 0.05

Tabel 3. Labeling Data

Sentimen	Total
neutral	109688
positive	5788
negative	2648

Tabel 3 menunjukkan total data hasil labelling dengan sentimen netral berjumlah 109688 data, sentimen positif 5788 data, dan sentimen negatif berjumlah 2648 data. Hasil labeling tersebut akan digunakan pada tahap selanjutnya yaitu split data.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

3.3. Split Data

Dalam tahapan penelitian ini, setelah data dibersihkan dan diberi label, selanjutnya dilakukan split data dengan tiga rasio data latih dan data tes. Rasio pembagian data tersebut meliputi 80:20, 70:30, 60:40. Tiga rasio data ini diperlukan untuk mencari hasil model dengan performa paling bagus, dengan melihat pada akurasi dan presisi serta *F-1 score*. Kode yang menerapkan pembagian data dapat dilihat pada Gambar 5.

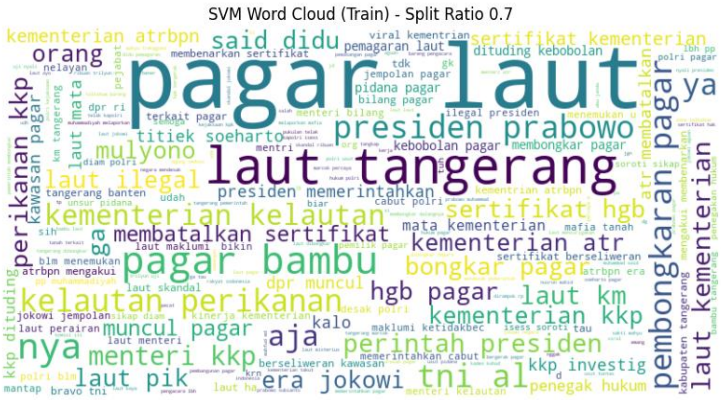
```
for ratio in split_ratios:
    print(f"\n==================== Split Ratio: {ratio} ====================")
    print("Membagi data menjadi training dan testing set...")
    X_train, X_test, y_train, y_test = train_test_split(
        X, y, test_size=(1 - ratio), random_state=42, stratify=y
    )
    print(f"Ukuran Training Set: {len(X_train)}")
    print(f"Ukuran Testing Set: {len(X_test)}")
```

Gambar 5. Split Data

Pembagian data menggunakan parameter *random_state* secara eksplisit. Hal ini penting dilakukan agar setiap kali kode dijalankan, data yang menjadi data latih dan data tes selalu konsisten. Jika parameter *random_state* tidak secara eksplisit, akan menghasilkan data yang selalu berbeda setiap split data dilakukan, sehingga akan mempersulit proses analisa dari setiap split ratio terhadap performa model yang dihasilkan. Selain *random_state*, implementasi parameter *stratify* juga diperlukan untuk memastikan proporsi yang sama. Misalnya dalam kode tersebut stratify bernilai y, dengan misal nilai y adalah 50% *neutral,* 25% *positive*, dan 25% *negative*. Maka dengan penggunaan parameter *stratify*, setiap split data dipastikan memiliki data dengan proporsi nilai y tersebut, sehingga data yang dibagi untuk *training* dan *testing* memiliki rasio sentimen yang sama sesuai dengan nilai y aslinya.

3.4. TF-IDF Vectorization

Tahap selanjutnya dalam penelitian ini yaitu penerapan TF-IDF untuk menghitung bobot dan frekuensi kemunculan kata pada setiap konten atau data pada dataset. Untuk memperjelas visualisasi frekuensi kemunculan kata pada dokumen, dapat dilihat contohnya pada Gambar 6. Gambar 6 tersebut menggambarkan frekuensi kemunculan kata atau frasa pada dataset. Semakin besar hurufnya berarti frekuensi kemunculannya lebih tinggi. Sebagai contoh, pada Gambar 6 frasa yang memiliki frekuensi kemunculan paling tinggi yaitu “pagar laut”.



Gambar 6. Word Cloud

p-ISSN: 2775-4227
e-ISSN: 2775-4219

3.5. Pelatihan Model dengan Algoritma SVM dan LR

Tahap berikut pada penelitian ini yaitu melakukan pelatihan atau implementasi dataset menggunakan kedua algoritma, SVM dan LR. Algoritma SVM sendiri pada awalnya hanya dikembangkan untuk klasifikasi dua kelas, namun semakin berkembang hingga kini dapat digunakan untuk klasifikasi multi kelas [12]. Di samping itu, data yang akan digunakan untuk algoritma SVM dan LR harus dilakukan *scaling*. *Data Scaling* sendiri merupakan teknik untuk normalisasi dan standarisasi untuk mengubah nilai numerik menjadi skala umum dengan tujuan agar proses pembelajaran oleh algoritma yang digunakan dapat dipercepat [13]. *Data Scaling* atau yang sering disebut dengan normalisasi data ini dapat membantu mempercepat fase pembelajaran dan menghindari masalah numerik seperti hilangnya akurasi karena terlalu banyak perhitungan aritmatika [14]. Beberapa yang sering digunakan untuk normalisasi yaitu *Max Normalization* dan *Min-Max Normalization*. Selain normalisasi data, pemilihan kernel pada algoritma SVM dan LR juga penting untuk memaksimalkan hasil sesuai dengan kompleksitas dataset. Pada penelitian ini, kernel SVM yang diapakai adalah RBF. RBF sendiri dikenal sebagai kernel yang memiliki performa yang bagus dalam kecepatan dan akurasi [10]. Sedangkan pada LR menggunakan *Limited-memory Broyden-Fletcher-Goldfish-Shanno Algorithm (LBFGS)*. Cara kerja dari LBFGS yaitu dengan menyimpan beberapa vektor yang mewakili aproksimasi secara implisit dengan menggunakan invers matriks Hessian [15].

Selain itu, penerapan *hyperparameter tuning* juga digunakan dalam penelitian ini untuk memastikan parameter yang digunakan adalah yang terbaik. *Hyperparameter tuning* adalah sebuah melakukan optimasi dan memilih parameter terbaik yang digunakan untuk model [16]. Secara spesifik yaitu *hyperparameter c*, yang merupakan nilai positif yang merepresentasikan invers dari kekuatan regularisasi (mengurangi *overfitting*), misalnya semakin kecil nilai c menunjukkan regularisasi yang semakin kuat [17].

Setelah melalui beberapa tahap tersebut, berikut adalah hasil akurasi *training* dan *testing* dengan tiga rasio dari implementasi SVM dan LR pada Tabel 4.

Tabel 4. Akurasi SVM dan LR pada Train dan Test

Rasio	Akurasi SVM Train	Akurasi SVM Test	Akurasi LR Train	Akurasi LR Test
0,8	0.913544	0.896846	0.999132	0.98797
0,7	0.999274	0.985382	0.999431	0.988881
0,6	0.946708	0.930793	0.998984	0.988529

Tabel 4 Akurasi SVM dan LR pada Train dan Test menunjukkan bahwa rasio *split data* memiliki pengaruh cukup signifikan terhadap akurasi dari setiap model. Dari ketiga rasio tersebut, berdasarkan data pada Tabel 4 dapat disimpulkan bahwa rasio dengan akurasi paling tinggi dan tidak *overfitting* adalah rasio 0,7 atau 70% data latih dan 30% data uji.

3.6. Evaluasi Model SVM dan LR

Setelah melihat hasil akurasi dari model SVM dan LR pada Tabel 4, perlu juga melakukan evaluasi model untuk lebih memahami kinerja model klasifikasi secara mendalam. Nilai-nilai pada evaluasi dapat memberikan gambaran mengenai seberapa baik model yang sudah dibuat dalam membuat prediksi yang benar, termasuk mengidentifikasi kelas positif serta menghindari kesalahan klasifikasi. Hasil evaluasi model SVM dapat dilihat pada Tabel 5.

Tabel 5. Evaluasi Algoritma SVM

Rasio	Precision Train	Precision Test	Recall Train	Recall Test	F1-Score Train	F1-Score Test
0,8	0.775700	0.737188	0.968760	0.896129	0.822002	0.766443
0,7	0.994985	0.937631	0.999583	0.921138	0.997270	0.927625
0,6	0.823912	0.791920	0.980687	0.896170	0.871601	0.811580

Dari Tabel 5 Evaluasi Algoritma SVM dapat terlihat bahwa hasil evaluasi terbaik terdapat pada rasio split data 0,7 atau 70% data latih dan 30% data uji. Pada rasio tersebut, nilai *precision train* mencapai 0,99 atau hampir sempurna, di mana hal ini menunjukkan bahwa model berhasil memprediksi 99% data sebagai benar-benar positif. Sedangkan untuk *precision test* nilainya turun menjadi 0,93. Hal ini dapat diakibatkan oleh model yang terlalu kompleks atau cenderung menghafal data dan noise, bukan mempelajarinya. Hasil *recall train* menunjukkan 0,99 dan *recall test* menunjukkan 0,92. Artinya, pada data train, model berhasil mengidentifikasi 99% dari semua data positif yang seharusnya diprediksi. Sedangkan pada data test, model berhasil mengidentifikasi 92% dari semua

p-ISSN: 2775-4227
e-ISSN: 2775-4219

data positif yang seharusnya diprediksi. Hasil *F1-score* untuk train test berada di angka lebih dari 0.9 yang berarti memiliki keseimbangan yang baik antara *precision* dan *recall*.

Hasil ini berbanding lurus dengan penelitian dari Aulia dkk. dengan judul penelitian "PERBANDINGAN KERNEL SUPPORT VECTOR MACHINE (SVM) DALAM PENERAPAN ANALISIS SENTIMEN VAKSINISASI COVID-19" yang menunjukkan hasil terbaik ada pada rasio 7:3 dengan kernel RBF ada pada akurasi 0,86 [18]. Di lain sisi, hal ini berbanding terbalik dengan penelitian yang dilakukan oleh Syah dkk. dengan judul penelitian " ANALISIS SENTIMEN MASYARAKAT TERHADAP VAKSINASI COVID-19 PADA MEDIA SOSIAL TWITTER MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM)" yang menggunakan rasio 8:2 dengan hasil menunjukkan akurasi 89% [19].

Tabel 6. Evaluasi Algoritma LR

Rasio	Precision Train	Precision Test	Recall Train	Recall Test	F1-Score Train	F1-Score Test
0,8	0.991254	0.951880	0.999415	0.912948	0.995304	0.931678
0,7	0.994011	0.959127	0.999640	0.914442	0.996810	0.935844
0,6	0.98888	0.957664	0.999089	0.912704	0.993924	0.934251

Dari Tabel 6 Evaluasi Algoritma LR dapat terlihat bahwa hasil evaluasi terbaik terdapat pada rasio split data 0,7. Pada rasio tersebut, nilai *precision train* mencapai 0,99 atau hampir sempurna, di mana hal ini menunjukkan bahwa model berhasil memprediksi 99% data sebagai benar-benar positif. Sedangkan untuk *precision test* nilainya turun menjadi 0,95 yang dapat diakibatkan oleh model yang terlalu kompleks atau cenderung menghafal data dan noise, bukan mempelajarinya. Hasil *recall train* menunjukkan 0,99 pada data train, yang berarti model berhasil mengidentifikasi 99% dari semua data positif yang seharusnya diprediksi. Sedangkan pada data test, model berhasil mengidentifikasi 91% dari semua data positif yang seharusnya diprediksi. Hasil *F1-score* untuk train test berada di angka lebih dari 0.9 yang berarti memiliki keseimbangan yang baik antara *precision* dan *recall*.

Hasil yang telah disajikan dalam penelitian ini dapat digunakan ke depannya untuk kebutuhan statistik, seperti analisis opini publik agar pemerintah dapat menanggulangi dan memberikan respon yang lebih baik terhadap isu yang menimbulkan berbagai macam reaksi dari masyarakat.

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, algoritma LR dalam penelitian ini memberikan hasil yang lebih unggul dalam rasio 7:3 dibanding SVM. Selain itu, hasil evaluasi juga menunjukkan keduanya hampir seimbang di mana nilai *precision test* dan *f1-score test* lebih unggul untuk algoritma LR, sedangkan nilai *recall test* lebih unggul pada algoritma SVM. Hasil yang diperoleh pada penelitian ini dapat berbeda dengan penelitian lainnya, bergantung pada jumlah data, jenis data, kernel yang digunakan, cara pra pemrosesan data, serta seberapa kompleks model yang berhasil dilatih. Keterbatasan dalam penelitian ini mencakup dari jumlah media sosial yang digunakan sebagai sumber data, serta penggunaan library menjadi pertimbangan untuk peningkatan pada penelitian berikutnya.

DAFTAR PUSTAKA

[1] E. R. Lidinillah, T. Rohana, and A. R. Juwita, "Analisis sentimen Twitter terhadap Steam menggunakan algoritma logistic regression dan support vector machine," TEKNOSAINS: Jurnal Sains, Teknologi dan Informatika, vol. 10, no. 2, pp. 154–164, 2023, doi: 10.37373/tekno.v10i2.440

[2] M. Qadri, "Pengaruh media sosial dalam membangun opini publik," Qaumiyyah: Jurnal Hukum Tata Negara, vol. 1, no. 1, pp. 49–63, 2020, doi: 10.24239/qaumiyyah.v1i1.4

[3] L. Judijanto et al., "Pengaruh sumber informasi dan interaksi sosial di media sosial terhadap pembentukan opini politik masyarakat di Indonesia," Sanskara Ilmu Sosial dan Humaniora, vol. 1, no. 01, pp. 21–31, 2023, doi: 10.58812/sish.v1i01.303

[4] M. Isnan, G. N. Elwirehardja, and B. Pardamean, "Sentiment Analysis for TikTok Review Using VADER Sentiment and SVM Model," 8th International Conference on Computer Science and Computational Intelligence (ICCSCI 2023), 2023, doi: 10.1016/j.procs.2023.10.514

[5] P. Arsi and R. Waluyo, "Analisis sentimen wacana pemindahan ibu kota Indonesia menggunakan algoritma Support Vector Machine (SVM)," J. Teknol. Inf. dan Ilmu Komput., vol. 8, no. 1, pp. 147, 2021, doi: 10.25126/jtlik.202183944

[6] I. S. K. Idris, Y. A. Mustofa, and I. A. Salihi, "Analisis sentimen terhadap penggunaan aplikasi Shopee

p-ISSN: 2775-4227
e-ISSN: 2775-4219

menggunakan algoritma Support Vector Machine (SVM)," Jambura Journal of Electrical and Electronics Engineering, vol. 5, no. 1, pp. 32–35, 2023, doi: 10.37905/jjeee.v5i1.16830

[7] B. Ramadhani, R. R. Suryono, and K. Kunci, "Komparasi Algoritma Naïve Bayes dan Logistic Regression Untuk Analisis Sentimen Metaverse," Jurnal Media Informatika Budidarma, vol. 8, pp. 714-725, 2024, doi: 10.30865/mib.v812.7458

[8] T. A. Khan, R. Sadiq, Z. Shahid, M. M. Alam, and M. B. M. Su'ud, "Sentiment Analysis using Support Vector Machine and Random Forest," Journal of Informatics and Web Engineering, vol. 3, no. 1, February 2024, doi: 10.33093/jiwe.2024.3.1.5

[9] C. Cortes and V. Vapnik, "Support-vector networks," Mach. Learn., vol. 20, no. 3, pp. 273–297, 1995.

[10] H. P. Singh, N. Singh, A. Mishra, S. K. Sen, M. Swarnkar and D. Pandey, "Logistic Regression based Sentiment Analysis System: Rectify," *2024 IEEE International Conference on Big Data & Machine Learning (ICBDML)*, Bhopal, India, 2024, pp. 186-191, doi: 10.1109/ICBDML60909.2024.10577296.

[11] B. Kabra and C. Nagar, "Convolutional Neural Network based sentiment analysis with TF-IDF based vectorization", J Integr Sci Technol, vol. 11, no. 3, p. 503, Jan. 2023, Accessed: Feb. 25, 2025. [Online]. Available: https://pubs.thesciencein.org/journal/index.php/jist/article/view/503

[12] S. Styawati, N. Hendrastuty, and A. R. Isnain, "Analisis sentimen masyarakat terhadap program kartu prakerja pada twitter dengan metode support vector machine," Jurnal Informatika: Jurnal Pengembangan IT, vol. 6, no. 3, pp. 150-155, 2021, doi: 10.30591/jpit.v6i3.2870

[13] A. Ambarwari, Q. J. Adrian, and Y. Herdiyeni, "Analysis of the effect of data scaling on the performance of the machine learning algorithm for plant identification," Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), vol. 4, no. 1, pp. 117-122, 2020, doi: 10.29207/resti.v4i1.1517

[14] Z. Liu, "A method of SVM with normalization in intrusion detection," Procedia Environmental Sciences, vol. 11, pp. 256-262, 2011, doi: 10.1016/j.proenv.2011.12.040

[15] M. Sharma, S. K. Agarwal, and M. Bundele, "Decisive Analysis of multiple logistic regression apropos of hyper-parameters," Indian J. Comput. Sci. Eng., vol. 13, pp. 188-196, 2022, doi: 10.21817/indjcse/2022/v13i1/221301190

[16] G. I. Diaz, A. Fokoue-Nkoutche, G. Nannicini, and H. Samulowitz, "An effective algorithm for hyperparameter optimization of neural networks," *IBM Journal of Research and Development*, vol. 61, no. 4/5, pp. 9:1-9:11, 2017, doi: 10.1147/JRD.2017.2709578

[17] D. M. Cristea, I. Sima, and L. B. Iantovics, "How Good Perform Logistic Regression Algorithm for Complex Gastroenterological Image Analysis. Comparativ Analysis with Physicians Performace," 2024, doi: 10.20944/preprints202410.0683.v1

[18] Aulia, T. M. P., Arifin, N., and Mayasari, R., "Perbandingan Kernel Support Vector Machine (SVM) Dalam Penerapan Analisis Sentimen Vaksinisasi Covid-19," SINTECH Journal, vol. 4, no. 2, pp. 139-145, Oct. 2021, doi: 10.31598/sintechjournal.v4i2.762

[19] Syah, H. and Witanti, A., "Analisis Sentimen Masyarakat Terhadap Vaksinasi Covid-19 Pada Media Sosial Twitter Menggunakan Algoritma Support Vector Machine (SVM)," Jurnal Sistem Informasi Dan Informatika (SIMIKA), vol. 5, no. 1, pp. 59-67, Apr. 2022, doi: 10.47080/simika.v5i1.1411