

# Perbandingan Algoritma *Support Vector Machine* dan *Long Short-Term Memory* untuk Klasifikasi Sentimen Aplikasi K24KLIK

Bianca Mohacindy<sup>1,\*</sup>, Novita Rahmayuna<sup>2</sup>, Nur Wakhidah<sup>3</sup>

<sup>1</sup>Fakultas Teknologi Informasi dan Komputer, Program Studi Sistem Informasi, Universitas Semarang, Kota Semarang, Indonesia

<sup>2</sup>*School of Information Systems, Information Systems*, Universitas Bina Nusantara, Kota Semarang, Indonesia

<sup>3</sup>Fakultas Teknologi Informasi dan Komputer, Program Studi Teknik Informatika, Universitas Semarang, Kota Semarang, Indonesia

Email: <sup>1,\*</sup> biancamohacindy6@gmail.com, <sup>2</sup>novita.rahmayuna@binus.ac.id, <sup>3</sup>ida@usm.ac.id

<sup>\*)</sup> Email Penulis Utama

**Abstrak**— Perkembangan teknologi digital dalam sektor kesehatan telah mendorong hadirnya layanan farmasi berbasis aplikasi, salah satunya K24Klik. Aplikasi ini menyediakan pemesanan obat, konsultasi daring dengan apoteker, dan akses informasi kesehatan. Ulasan pengguna menjadi sumber penting untuk mengevaluasi kualitas layanan dan kepuasan pelanggan. Penelitian ini membandingkan performa algoritma klasifikasi sentimen *Support Vector Machine (SVM)* maupun *Long Short-Term Memory (LSTM)* terhadap 1.000 ulasan pengguna K24Klik dari web scraping dari Google Play Store. Data diolah melalui tahapan *cleansing, case folding, tokenizing, stopword removal, dan stemming*, lalu diberi label sentimen positif, negatif, dan netral menggunakan model *Transformer*. Eksperimen dilakukan pada tiga rasio data latih-uji (80:20, 70:30, 60:40) untuk memeriksa konsistensi model. Hasil menunjukkan *SVM* memperoleh *F1-score* tertinggi 0.88 pada kelas positif dan negatif, sedangkan *LSTM* mencapai akurasi 0.84 dengan performa lebih merata termasuk pada kelas netral. Penelitian ini menegaskan pentingnya pemilihan algoritma sesuai karakteristik data; *SVM* unggul pada data seimbang dan terstruktur, sementara *LSTM* efektif untuk data tidak seimbang dan konteks teks kompleks. Kontribusi utama studi ini adalah penerapan pelabelan otomatis berbasis *Transformer* yang terbukti akurat untuk membangun sistem analisis sentimen real-time di layanan farmasi digital.

**Kata Kunci:** Sentimen, K24Klik, *Long Short-Term Memory*, *Support Vector Machine*, Klasifikasi Teks, *Web Scraping*

**Abstract**— The development of digital technology in the health sector has encouraged the presence of application-based pharmacy services, one of which is K24Klik. This application provides drug ordering, bold consultations with pharmacists, and access to health information. User reviews are an important source for transmitting service quality and customer satisfaction. This study compares the performance of the *Support Vector Machine (SVM)* and *Long Short-Term Memory (LSTM)* sentiment classification algorithms on 1,000 K24Klik user reviews collected through web scraping from the Google Play Store. The data was processed through the stages of cleaning, case folding, tokenization, stopword removal, and stemming, then labeled with positive, negative, and neutral sentiments using the *Transformer* model. Experiments were conducted on three training-test data ratios (80:20, 70:30, 60:40) to check the consistency of the model. The results showed that *SVM* obtained the highest *F1-score* of 0.88 in the positive and negative classes, while *LSTM* achieved an accuracy of 0.84 with more even performance including in the neutral class. This study emphasizes the importance of selecting an algorithm according to data characteristics; *SVM* excels on balanced and structured data, while *LSTM* is effective for imbalanced data and complex text contexts. The main contribution of this study is the application of *Transformer*-based auto-labeling that is proven to be accurate to build a real-time sentiment analysis system in digital pharmacy services

**Keywords:** Sentiment, K24Klik, *Long Short-Term Memory*, *Support Vector Machine*, Text Classification, *Web Scraping*

## 1. PENDAHULUAN

Perkembangan teknologi digital telah memberikan dampak besar dalam mengubah pola konsumsi masyarakat, termasuk dalam mengakses layanan kesehatan. Layanan farmasi berbasis aplikasi digital menjadi salah satu solusi praktis yang kini banyak digunakan. Aplikasi K24Klik hadir sebagai layanan farmasi daring yang menyediakan fitur pembelian obat, konsultasi dengan apoteker, dan akses informasi kesehatan secara digital. Dalam konteks ini, data ulasan dan penilaian pemakai di Google Play Store menjadi sumber informasi penting untuk mengevaluasi mutu layanan, kepuasan pengguna, serta kinerja sistem aplikasi tersebut. Sayangnya, ulasan pengguna umumnya berbentuk teks bebas (*unstructured data*), sehingga memerlukan pendekatan komputasional yang cerdas agar dapat diproses dan dimanfaatkan secara otomatis [1]. Analisis klasifikasi sentimen sangat penting untuk memahami opini pengguna terhadap layanan farmasi digital secara cepat dan objektif. Hasil analisis

ini dapat menjadi dasar dalam pengambilan keputusan dan peningkatan kualitas layanan. Analisis klasifikasi sentimen terhadap ulasan pengguna menjadi metode efektif untuk menggali opini publik secara *real-time* dan sistematis [2]. Studi-studi sebelumnya telah banyak mengkaji klasifikasi sentimen dalam berbagai *domain*. Nurochman dan Ashiilah [3] membandingkan algoritma *Support Vector Machine (SVM)* dan *Long Short-Term Memory (LSTM)* untuk memprediksi gangguan kecemasan dari cuitan di platform *Twitter*. Hasilnya menunjukkan bahwa *LSTM* lebih unggul dalam mengenali konteks emosional yang kompleks, terutama pada data teks tidak terstruktur [4]. Di sisi lain, Rahmawati [5] menggunakan metode *Naïve Bayes* untuk klasifikasi sentimen pada ulasan aplikasi K24Klik, tetapi performanya masih terbatas dalam menangkap makna tidak kontekstual [6]. Pratama dan Handayani [7] menunjukkan bahwa *LSTM* menghasilkan klasifikasi yang lebih akurat pada *domain* aplikasi kesehatan, sebagaimana diterapkan pada ulasan aplikasi PeduliLindungi.

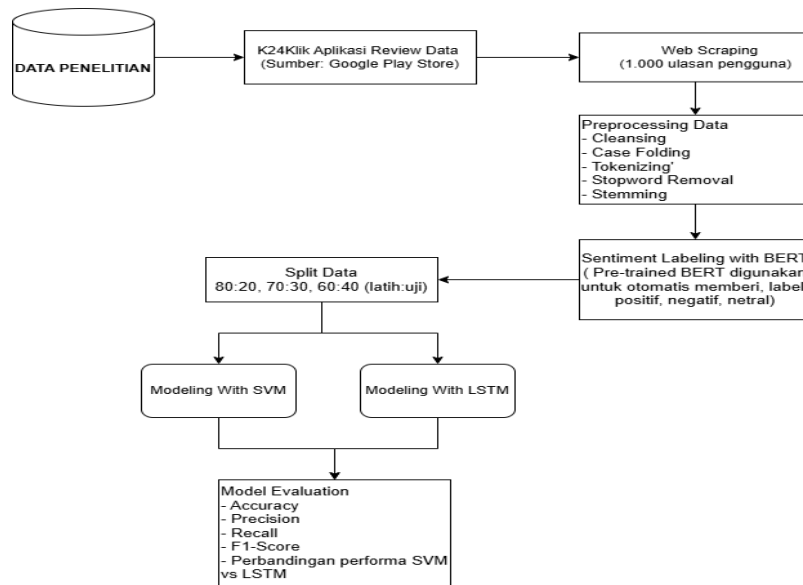
Namun demikian, kajian khusus terkait klasifikasi sentimen pengguna terhadap layanan farmasi digital seperti K24Klik masih sangat terbatas. Sebagian besar penelitian sebelumnya lebih banyak berfokus pada platform *e-commerce* seperti Tokopedia, dengan penggunaan algoritma *SVM* untuk klasifikasi ulasan. Dalam konteks klasifikasi sentimen, dua algoritma yang paling banyak digunakan adalah *SVM* dan *LSTM*. *SVM* dikenal sebagai metode *supervised learning* yang efektif untuk klasifikasi teks pendek karena kemampuannya dalam mencari *hyperplane* pemisah optimal [8]. Sebaliknya, *LSTM* merupakan bagian dari arsitektur jaringan saraf dalam (*deep neural network*) yang dirancang untuk mengenali urutan kata serta memahami konteks secara dinamis, menjadikannya cocok untuk data ulasan yang panjang dan kompleks [9]. Mengingat karakteristik dan keunggulan masing-masing metode, evaluasi langsung dalam *domain* farmasi digital menjadi hal yang relevan dan penting untuk dilakukan. Berdasarkan hal ini, penelitian ini membandingkan algoritma *SVM* dan *LSTM* untuk klasifikasi sentimen dalam evaluasi pengguna K24Klik. Akurasi, presisi, recall, dan skor F1 digunakan untuk mengevaluasi evaluasi pelanggan Google Play Store. Penelitian ini juga mengkaji bagaimana rasio data pelatihan-pengujian memengaruhi kinerja model. Penelitian ini diharapkan dapat memberikan rekomendasi algoritma berbasis analisis sentimen terbaik untuk pengembangan layanan apotek digital.

## 2. METODE PENELITIAN

### 2.1 Data Penelitian

Data dalam penelitian dari kolom ulasan pengguna aplikasi K24Klik secara publik di Google Play Store. Pengambilan data dilakukan secara otomatis menggunakan metode web *scraping* dengan pemrograman Python pada platform Google Colab [10]. Proses *scraping* dilaksanakan langsung oleh peneliti dengan memanfaatkan *auth token* aplikasi K24Klik, sehingga memungkinkan akses data yang lebih lengkap, seperti isi ulasan, nilai *rating*, dan tanggal ulasan. Data yang diperoleh mencerminkan opini dan pengalaman pengguna terhadap layanan aplikasi K24Klik. Setelah pengumpulan, data disimpan dalam format *.csv*, kemudian dilakukan pembersihan dengan menghapus duplikasi, *spam*, serta ulasan kosong yang tidak relevan. Total ulasan yang berhasil dikumpulkan dan siap dianalisis berjumlah sekitar 1.000 data, yang selanjutnya diproses menggunakan teknik *text mining* dan diklasifikasikan dengan algoritma *Support Vector Machine (SVM)* dan *Long Short-Term Memory (LSTM)*.

Untuk pengujian algoritma *SVM*, digunakan dua konfigurasi model, yaitu *SVM Default* dan *SVM Balanced*. *SVM Default* adalah model dengan parameter bawaan tanpa penyesuaian bobot kelas. Sementara itu, *SVM Balanced* menggunakan parameter *class\_weight='balanced'* untuk menyesuaikan bobot kelas berdasarkan distribusi data. Konfigurasi ini digunakan karena data berlabel netral hanya berjumlah 45 dari 1.000 data, sehingga *balancing* penting agar model tidak *bias* terhadap kelas mayoritas. Pengambilan sampel berstrata digunakan untuk mendistribusikan data secara merata di seluruh kelas selama pelatihan dan penilaian. Eksperimen ini menggunakan rasio 80:20, 70:30, dan 60:40. Tabel 1 menyajikan data pelatihan dan pengujian untuk setiap kelas sentimen dalam rasio data 80:20, 70:30, dan 60:40. Tujuan penggunaan dua konfigurasi model *SVM (Default dan Balanced)* adalah untuk membandingkan performa klasifikasi terhadap data yang tidak seimbang.



**Gambar 1.** Metode Penelitian

Sumber: hasil olahan peneliti (2025)

**Tabel 1.** Distribusi Jumlah Data Latih dan Uji per Kelas Sentimen, Sumber: hasil olahan peneliti (2025)

Rasio	Kelas	Data Latih	Data Uji
80:20	Negatif	417	105
	Positif	346	87
	Netral	36	9
70:30	Negatif	365	157
	Positif	303	130
	Netral	32	13
60:40	Negatif	313	209
	Positif	260	173
	Netral	27	18

Tabel 1 tersebut menyajikan jumlah data latih maupun uji untuk masing-masing kelas sentimen (Negatif, Positif, Netral) berdasarkan tiga skenario rasio data ialah 80:20, 70:30 maupun 60:40. Pembagian dilakukan menggunakan metode *stratified sampling* agar distribusi label tetap seimbang dalam subset data. Hal ini penting untuk mengurangi potensi bias terhadap kelas mayoritas, khususnya karena data netral memiliki jumlah yang sangat kecil dibandingkan dua kelas lainnya. Dengan pendekatan ini, model memiliki peluang yang lebih adil untuk belajar dari semua kelas secara proporsional.

### 2.2 Pre-processing Data

Tahapan ini merujuk pada pendekatan yang digunakan oleh Nurochman & Ashiilah dan diadaptasi untuk data ulasan aplikasi K24Klik. *Pre-processing* Data Tahapan ini merujuk pada pendekatan yang digunakan oleh Nurochman & Ashiilah dan diadaptasi untuk data ulasan aplikasi K24Klik. Tahapan ini mencakup *cleansing*, *case folding*, *tokenizing*, *stopword removal* maupun *stemming*, sebagaimana umum diterapkan pada preprocessing teks Bahasa Indonesia [11]. Tabel 2 menyajikan hasil tahapan *preprocessing* terhadap data ulasan pengguna, mulai dari teks asli hingga proses *stemming*.

**Tabel 2.** Hasil tahapan *preprocessing* data dari teks asli hingga proses *stemming*, Sumber: hasil olahan peneliti (2025)

Tahapan <i>Preprocessing</i>	Hasil
Data Asli	Jujur ini aplikasi bagus tapi untuk layanan konsultasi dokternya kurang memuaskan, saya udah jelasin keluhan saya tetapi dokternya sendiri malah ngeresepin obat yang lain, saya minta obat minum sama salep.
<i>Cleansing</i>	Jujur ini aplikasi bagus tapi untuk layanan konsultasi dokternya kurang memuaskan saya udah jelasin keluhan saya tetapi dokternya sendiri malah ngeresepin obat yang lain saya minta obat minum sama salep
<i>Case Folding</i>	jujur ini aplikasi bagus tapi untuk layanan konsultasi dokternya kurang memuaskan saya udah jelasin keluhan saya tetapi dokternya sendiri malah ngeresepin obat yang lain saya minta obat minum sama salep
<i>Tokenizing</i>	['jujur', 'ini', 'aplikasi', 'bagus', 'tapi', 'untuk', 'layanan', 'konsultasi', 'dokternya', 'kurang', 'memuaskan', 'saya', 'udah', 'jelasin', 'keluhan', 'saya', 'tetapi', 'dokternya', 'sendiri', 'malah', 'ngeresepin', 'obat', 'yang', 'lain', 'saya', 'minta', 'obat', 'minum', 'sama', 'salep']
<i>Stopword Removal</i>	['jujur', 'aplikasi', 'bagus', 'layanan', 'konsultasi', 'dokternya', 'kurang', 'memuaskan', 'udah', 'jelasin', 'keluhan', 'dokternya', 'sendiri', 'malah', 'ngeresepin', 'obat', 'lain', 'minta', 'obat', 'minum', 'salep']
<i>Stemming</i>	['jujur', 'aplikasi', 'bagus', 'layan', 'konsultasi', 'dokter', 'kurang', 'muas', 'udah', 'jelasin', 'keluh', 'dokter', 'sendiri', 'malah', 'ngeresep', 'obat', 'lain', 'minta', 'obat', 'minum', 'salep']

### 2.3 Labeling Data

Pelabelan sentimen pada data percakapan yang telah dibersihkan atau ulasan pengguna mengikuti tahap persiapan. Konteks emosional teks menentukan label: positif, negatif, dan netral. Pelabelan ini dilakukan secara otomatis menggunakan model berbasis algoritma Transformer, salah satu metode NLP yang paling ampuh. Konsep ini menggunakan mekanisme self-attention untuk menganalisis makna dan kehalusan frasa serta merekam asosiasi kata tanpa kehilangan konteks. Penelitian ini menggunakan model Transformer BERT yang telah dilatih sebelumnya [12], yang telah diadaptasi untuk tugas klasifikasi sentimen. Proses pelabelan diawali dengan tahap tokenisasi, yaitu memecah kalimat menjadi token-token sesuai struktur bahasa alami. Tokenisasi ini kemudian dikonversi menjadi representasi numerik dan dimasukkan ke dalam model. Model memproses representasi tersebut untuk menghasilkan vektor kalimat menggunakan token khusus [CLS], yang mewakili makna keseluruhan input. Vektor ini kemudian diteruskan ke layer klasifikasi untuk menghasilkan skor (logit) bagi setiap kategori sentimen. Skor tersebut diubah menjadi probabilitas dari fungsi *softmax* maupun label probabilitas tertinggi ditetapkan sebagai hasil klasifikasi.

Sebagai contoh, ulasan seperti “Aplikasi yang sangat bermanfaat” menghasilkan skor *softmax* [0.87, 0.08, 0.05] untuk kelas [positif, netral, negatif], sehingga sistem akan melabeli data tersebut sebagai positif. Pendekatan ini menghasilkan pelabelan yang konsisten dan relevan, bahkan untuk kalimat yang kompleks atau ambigu. *Dataset* yang digunakan pada tahap ini terdiri dari 1.000 data percakapan yang sebelumnya dari proses pembersihan dari duplikasi, karakter tidak penting, dan normalisasi teks. Berdasarkan hasil pelabelan otomatis menggunakan model *Transformer*, diperoleh distribusi sebagai berikut: 522 data berlabel negatif, 433 data berlabel positif, dan 45 data berlabel netral. Ketiga kategori ini menjadi dasar dalam proses pelatihan dan evaluasi model klasifikasi sentimen pada tahap selanjutnya. Tabel 3 menyajikan contoh hasil pelabelan otomatis terhadap data ulasan pengguna berdasarkan tiga kategori sentimen ialah positif, negatif maupun netral. Distribusi lengkap hasil pelabelan pada Tabel 1.

**Tabel 3.** Hasil *Labeling* Data, Sumber: hasil olahan peneliti (2025)

No	Hasil Preprocessing Data	Sentiment
1	jujur ini aplikasi bagus tapi untuk layanan konsultasi dokternya kurang memuaskan, saya udah jelasin keluhan saya tetapi dokternya sendiri malah ngeresepin obat yang lain, saya minta obat minum sama salep,	Negatif
2	Pelayanan obatnya cepat. Bahkan untuk obat yang harus pakai resep dokter pun, sopnya bagus. Mantap pokoknya.	Positif
3	Aplikasi yang sangat bermanfaat, ketika keluarga mendadak sakit, kita sedang jauh dari rumah, semoga aplikasi ini terus lancer	Positif
4	Kecewa banget, kalau memang gabisa pakai pengiriman dengan benar mendingan gausah di adain kak, 2 kali kejadian pengiriman selalu salah titik, lalu pihak grab mengembalikan ke apotek, dan saya harus	Negatif

## 2.4 Modeling

### 2.4.1 Modeling Menggunakan Support Vector Machine (SVM)

Algoritma *Support Vector Machine (SVM)* digunakan suatu metode utama dalam klasifikasi sentimen ulasan pengguna aplikasi K24Klik. *SVM* dengan performa tinggi dalam menangani data berdimensi besar dan sangat efektif untuk tugas klasifikasi teks karena kemampuannya terkait *hyperplane* optimal sebagai pemisah antar kelas. Dalam penelitian ini, data teks diubah ke dalam representasi vektor numerik menggunakan metode *TF-IDF* guna menangkap pentingnya suatu kata relatif terhadap seluruh korpus. Untuk menangani ketidakseimbangan distribusi kelas, dua konfigurasi *SVM* digunakan: *SVM Default* dan *SVM Balanced*. Model *SVM Default* dikonfigurasi dengan parameter *kernel='linear'* dan  $C=1.0$ , yang merupakan nilai regulasi standar untuk mengontrol margin pemisah. Parameter lain yang tetap mengikuti nilai bawaan dari pustaka *scikit-learn* antara lain *degree=3*, *gamma='scale'*, dan *class\_weight=None*. Namun, karena *kernel* yang digunakan adalah *linear*, parameter seperti *gamma*, *degree*, dan *coef0* tidak berpengaruh terhadap proses pelatihan. Dengan konfigurasi ini, model menghasilkan nilai *intercept* sebesar  $[0.883, 0.300, -0.742]$  serta sejumlah *support vectors* yang mayoritas berasal dari data berlabel negatif, mengindikasikan dominasi kelas mayoritas dalam proses klasifikasi. Konfigurasi ini digunakan sebagai *baseline* untuk mengevaluasi performa tanpa penyeimbangan bobot kelas. Sebaliknya, *SVM Balanced* menggunakan pengaturan *class\_weight='balanced'*, yang secara otomatis menyesuaikan bobot masing-masing kelas berdasarkan distribusi label dalam data latih. Pendekatan ini bertujuan meningkatkan sensitivitas terhadap kelas minoritas tanpa mengubah struktur model secara signifikan. Evaluasi terhadap kedua konfigurasi dilakukan pada tiga skenario rasio data latih maupun uji ialah 80:20, 70:30, maupun 60:40, untuk mengamati konsistensi performa model terhadap variasi proporsi data pelatihan.

### 2.4.2 Modeling Long Short-Term Memory (LSTM)

Model *Long Short-Term Memory (LSTM)* digunakan berbasis deep learning yang mampu mengenali urutan kata dan memahami konteks semantik dalam teks ulasan pengguna. Arsitektur model terdiri atas beberapa lapisan, termasuk *embedding* untuk representasi kata, *dropout* untuk regularisasi, *LSTM layer* untuk menangkap pola sekuensial, serta *output layer* untuk didapatkan klasifikasi ke dalam tiga kategori sentimen. Model ini dikembangkan memakai *framework TensorFlow*, dengan struktur yang ringkas dan efisien untuk mengimbangi kompleksitas bahasa dalam ulasan pengguna. Rancangan ini disusun untuk menjaga keseimbangan antara performa klasifikasi dan efisiensi pelatihan. Struktur lapisan dan fungsinya dirangkum dalam Tabel 4.

**Tabel 4.** Konfigurasi Layer dan Parameter Model LSTM, Sumber: hasil olahan peneliti (2025)

No	Layer (Tipe)	Output Shape	Jumlah Parameter
1	<i>Input (InputLayer)</i>	<i>(None, 50)</i>	0
2	<i>Embedding</i>	<i>(None, 50, 128)</i>	640,000
3	<i>SpatialDropout1D</i>	<i>(None, 50, 128)</i>	0
4	<i>LSTM</i>	<i>(None, 64)</i>	49,408
5	<i>Dense</i>	<i>(None, 3)</i>	195
	<b>Total</b>		<b>689,603</b>

Total *params*: 689,603 (2.63 MB)

Trainable *params*: 689,603 (2.63 MB)

Non-trainable *params*: 0 (0.00 B)

## 2.5 Pengujian Data

Evaluasi kinerja dilakukan terhadap data yang telah diproses dan dilabeli secara otomatis menggunakan model *Transformer*. Pengujian dilaksanakan pada ketiga konfigurasi rasio data latih dan uji (80:20, 70:30, 60:40) untuk kedua model (*SVM* dan *LSTM*), dengan metrik penilaian yang mencakup *accuracy*, *presisi*, *recall* maupun *F1-score*. Hasil evaluasi memperlihatkan model *SVM* memberikan hasil yang sangat baik dan stabil, terutama untuk kelas dominan seperti positif dan negatif. Skenario rasio 60:40 menghasilkan *F1-score* tertinggi, mencapai 0.88 untuk kedua kelas tersebut. Sebaliknya, *LSTM* menunjukkan kinerja yang lebih merata terhadap semua kelas, termasuk kelas netral. Rasio 60:40 juga menjadi konfigurasi terbaik untuk *LSTM*, dengan akurasi mencapai 0.84 dan nilai *precision* serta *recall* yang seimbang di kisaran 0.57 hingga 0.58. Hasil ini menunjukkan bahwa *LSTM* unggul dalam memahami struktur kalimat yang kompleks, sedangkan *SVM* lebih optimal pada data yang lebih seimbang dan terstruktur.

## 3. HASIL DAN PEMBAHASAN

Bagian ini menunjukkan temuan kategorisasi sentimen ulasan pengguna aplikasi K24Klik. Pengikisan situs web dari Google Play Store menghasilkan 1.000 set data ulasan yang telah dibersihkan, dilipat kasusnya, ditokenisasi, dihilangkan kata kuncinya, dan di-stem. Model *Transformer* berbasis BERT yang telah dilatih sebelumnya secara otomatis memberi label ulasan sebagai baik, negatif, atau netral. Set data berlabel ini digunakan untuk melatih dan mengevaluasi algoritma klasifikasi *SVM* dan *LSTM*. Pengujian meliputi pembagian data pelatihan dan pengujian 80:20, 70:30, dan 60:40. Performa model diukur berdasarkan akurasi, presisi, recall, dan skor F1. Hasil pengujian dan performa kedua model dalam setiap situasi ditampilkan dan dianalisis di bawah ini.

### 3.1 Support Vector Machine (SVM)

Hasil pengujian metode ini untuk klasifikasi sentimen aplikasi K24Klik dilakukan dengan tiga skenario pembagian data latih dan data uji, yaitu rasio 80:20, 70:30 maupun 60:40. Evaluasi dilakukan terhadap tiga kelas sentimen yaitu negatif, netral, dan positif, dengan menggunakan dua konfigurasi model, yaitu *SVM Default* dan *SVM Balanced*. Model *SVM Default* menggunakan parameter standar tanpa penyesuaian bobot kelas (*class\_weight=None*), sedangkan *SVM Balanced* menggunakan pengaturan *class\_weight='balanced'* untuk menyesuaikan bobot antar kelas berdasarkan distribusi label pada data pelatihan. Penggunaan dua konfigurasi ini bertujuan untuk mengatasi ketidakseimbangan jumlah data, terutama pada kelas netral yang hanya berjumlah 45 dari total 1.000 data. Kombinasi antara variasi rasio dan konfigurasi model ini digunakan dalam seluruh skenario pengujian, sebagaimana disajikan pada tabel berikut.

Secara umum, hasil pengujian menunjukkan bahwa model *SVM* memberikan performa yang paling stabil dan tinggi pada kelas negatif dan positif di seluruh skenario rasio. Percobaan dengan rasio 60:40 menghasilkan performa terbaik dengan *F1-score* tertinggi sebesar 0.88 pada kedua konfigurasi model, baik *SVM Default* maupun *SVM Balanced*. Nilai *precision* dan *recall* pada skenario ini juga tergolong tinggi, masing-masing mencapai  $\geq 0.83$  maupun  $\geq 0.93$ . Untuk kelas positif, performa model juga menunjukkan hasil yang konsisten, dengan *F1-score* sebesar 0.88, *precision* mencapai 0.90, dan *recall* diantara 0.85 hingga 0.86. Temuan memperlihatkan model *SVM* mampu mengenali pola sentimen dengan baik, khususnya pada kelas yang memiliki jumlah data dominan. Tabel 5 menyajikan hasil evaluasi *matrix* klasifikasi model *SVM Default* dan *SVM Balanced* terhadap tiga kelas sentimen pada masing-masing rasio pembagian data. Dapat diamati bahwa kelas netral mengalami performa terendah secara konsisten, terutama pada model *SVM Default* yang menghasilkan nilai *precision*, *recall* maupun *F1-score* sejumlah 0.00 di semua rasio. Penerapan *SVM Balanced* memberikan sedikit perbaikan, seperti terlihat pada rasio 70:30 (*F1-score* = 0.27) dan 60:40 (*F1-score* = 0.19), namun peningkatan ini belum signifikan. Untuk memperkuat analisis, Gambar 2 menampilkan *Confusion Matrix* dari model *SVM Balanced* pada ketiga rasio pembagian data. Terlihat bahwa prediksi terhadap kelas negatif (196 dari 209) dan kelas positif (145 dari 173) sangat akurat, ditandai dengan dominasi nilai diagonal dalam *matrix*. Sebaliknya, kelas netral (1 dari 18) masih banyak diklasifikasikan ke kelas negatif dan positif, memperkuat temuan bahwa jumlah data yang kecil menyulitkan model dalam mengenali pola netral secara akurat.

Tabel 5. Perbandingan Hasil *Confusion Matrix* Metode SVM

a. Kelas Netral

Rasio	Model	Precision	Recall	F1-Score	Support
80:20	SVM Default	0.00	0.00	0.00	9
80:20	SVM Balanced	0.00	0.00	0.00	9
70:30	SVM Default	0.00	0.00	0.00	13
70:30	SVM Balanced	1.00	0.15	0.27	13
60:40	SVM Default	0.00	0.00	0.00	18
60:40	SVM Balanced	0.67	0.11	0.19	18

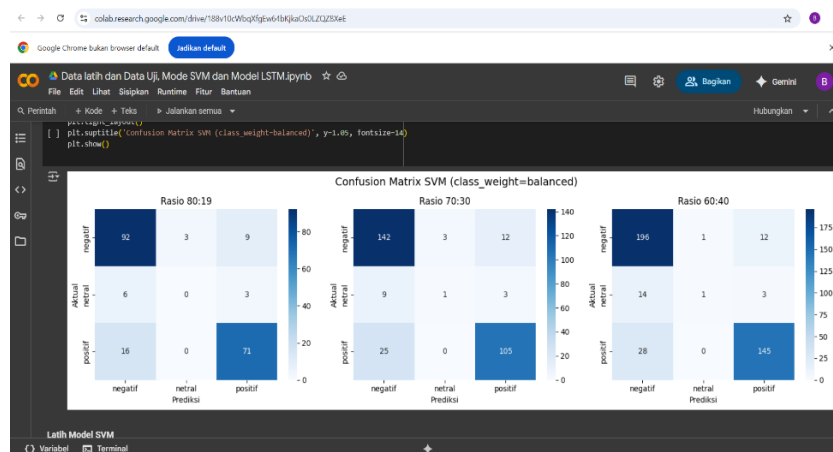
b. Kelas Negatif

Rasio	Model	Precision	Recall	F1-Score	Support
80:20	SVM Default	0.79	0.92	0.85	104
80:20	SVM Balanced	0.79	0.90	0.84	104
70:30	SVM Default	0.81	0.94	0.87	157
70:30	SVM Balanced	0.82	0.93	0.87	157
60:40	SVM Default	0.83	0.94	0.88	209
60:40	SVM Balanced	0.84	0.93	0.88	209

c. Kelas Positif

Rasio	Model	Precision	Recall	F1-Score	Support
80:20	SVM Default	0.87	0.79	0.83	87
80:20	SVM Balanced	0.85	0.79	0.82	87
70:30	SVM Default	0.89	0.81	0.85	130
70:30	SVM Balanced	0.88	0.82	0.85	130
60:40	SVM Default	0.90	0.85	0.88	173
60:40	SVM Balanced	0.90	0.86	0.88	173

Sumber: hasil olahan peneliti (2025)



Gambar 2. Confusion Matrix SVM Balanced pada tiga rasio.

Sumber: hasil olahan peneliti (2025)

Namun, kelas netral, hasil evaluasi memperlihatkan model mengalami kesulitan dalam melakukan klasifikasi baik, terutama pada model SVM Default yang *consistently* menghasilkan *precision*, *recall* maupun *F1-Score* sebesar 0.00 pada semua rasio. Model SVM Balanced memberikan sedikit peningkatan performa, terutama pada rasio 70:30 dan 60:40, di mana *F1-Score* meningkat menjadi 0.27 dan 0.19. Rendahnya performa pada kelas

netral ini dipengaruhi oleh jumlah data (*support*) lebih sedikit dibandingkan dengan kelas lainnya. Secara keseluruhan, model *SVM* menunjukkan performa terbaik pada skenario rasio 60:40, khususnya untuk kelas negatif dan positif. Sementara itu, model *SVM Balanced* cenderung memberikan peningkatan pada performa kelas minoritas seperti netral, meskipun peningkatannya belum signifikan.

### 3.2 Long Short-Term Memory (LSTM)

Kinerja model *Long Short-Term Memory (LSTM)* sangat dipengaruhi oleh kemampuan generalisasinya terhadap variasi data. Wafda 2025 Menekankan pentingnya pemahaman prinsip generalisasi dalam *deep learning*, terutama ketika model dilatih dengan data yang terbatas atau tidak seimbang [13]. Oleh karena itu, arsitektur dan parameter *LSTM* dalam penelitian ini dirancang untuk menjaga keseimbangan antara kapasitas model dan kompleksitas data, dengan mempertimbangkan risiko *overfitting* dan *underfitting* secara cermat.

Pada tahap ini, pengujian terhadap data dari proses *preprocessing* maupun *labeling*, menggunakan algoritma *Long Short-Term Memory (LSTM)* menjadi metode utama dalam klasifikasi sentimen. Model *LSTM* dikonfigurasi dengan lima lapisan utama, dimulai dari *inputlayer*, *embedding*, *spatialdropout1D*, *LSTM*, hingga *dense layer* dengan tiga unit output yang merepresentasikan tiga kelas sentimen: positif, negatif maupun netral. Total parameter dalam arsitektur ini berjumlah 689.603, seluruhnya bersifat *trainable*, sebagaimana ditunjukkan pada Tabel 7. Pengujian dilakukan dengan tiga skenario pembagian data latih maupun uji ialah 80:20, 70:30, dan 60:40, dengan jumlah *epoch* sebanyak 30 kali pelatihan untuk masing-masing skenario. Variasi rasio ini bertujuan untuk mengamati performa model pada kondisi jumlah data uji yang berbeda, serta mengevaluasi stabilitas dan konsistensi model. Setiap proses pelatihan menghasilkan nilai *accuracy*, *precision*, *recall*, dan *F1-score* yang bervariasi, tergantung pada proporsi data dan kompleksitas input. Berdasarkan hasil pengujian, skenario dengan rasio 60:40 dengan performa terbaik dengan akurasi mencapai 0.84, serta keseimbangan *precision* dan *recall* di kisaran 0.57–0.58. Hal ini menunjukkan bahwa *LSTM* mampu melakukan klasifikasi secara lebih optimal ketika model diberi paparan terhadap variasi data uji yang lebih luas. Dengan hasil tersebut, model *LSTM* dinilai efektif dalam memahami pola urutan kata serta konteks semantik pada teks ulasan pengguna aplikasi K24Klik, terutama ketika jumlah *epoch* dan distribusi data uji dikonfigurasi secara tepat. Tabel 6 menyajikan hasil evaluasi model *Long Short-Term Memory (LSTM)* melaksanakan klasifikasi sentimen pada tiga skenario rasio data latih dan data uji ialah 80:20, 70:30 maupun 60:40. Evaluasi memakai *matrix accuracy*, *precision*, *recall*, dan *F1-score*, berdasarkan hasil pelatihan selama 30 *epoch*.

**Tabel 6.** Perbandingan Hasil *Confusion Matrix* Metode *LSTM*

No	Data Latih	Data Uji	Epoch	Accuracy	Precision	Recall	f1-Score
1	80%	20%	30	0.81	0.54	0.57	0.55
2	70%	30%	30	0.82	0.55	0.57	0.56
3	60%	40%	30	0.84	0.57	0.58	0.58

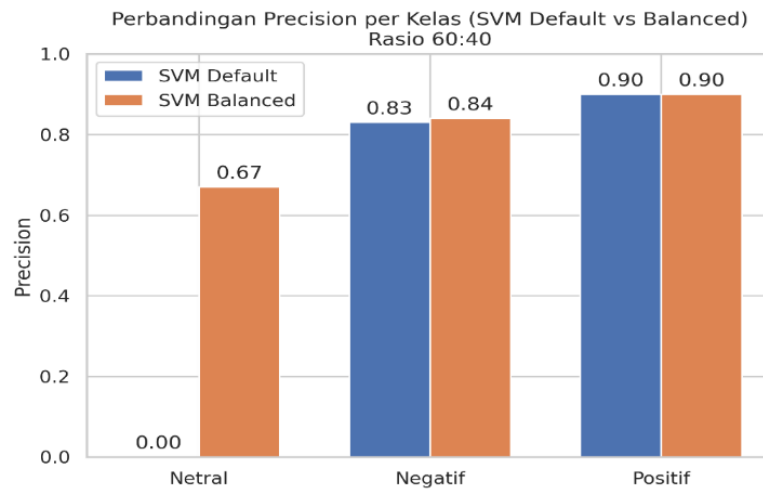
Sumber: hasil olahan peneliti (2025)

### 3.3 Perbandingan SVM dan LSTM

Dalam membandingkan algoritma *Support Vector Machine (SVM)* maupun *Long Short-Term Memory (LSTM)*, penting mempertimbangkan karakteristik dan keunggulan masing-masing. Anisa et al. [14] menyatakan bahwa kombinasi keduanya dapat menghasilkan model *hybrid* yang efektif dalam tugas klasifikasi sentimen. Meskipun dalam penelitian ini kedua algoritma diterapkan secara terpisah, pemilihan metode tetap harus disesuaikan dengan tujuan analisis maupun karakteristik data. Secara umum, *SVM* bekerja optimal pada data yang bersih dan seimbang, sedangkan *LSTM* lebih unggul dalam menangani data teks sekuensial yang kompleks. Gambar 3 menyajikan perbandingan performa awal (*baseline*) antara model *SVM* dan *LSTM* berdasarkan empat metrik evaluasi, yaitu *accuracy*, *precision*, *recall*, maupun *F1-score*. Model *SVM* memperlihatkan lebih stabil dan akurat pada semua metrik dibandingkan *LSTM*. Namun, performa *LSTM* berpotensi meningkat seiring jumlah data dan kompleksitas pelatihan. Hasil ini diperoleh dari konfigurasi awal tanpa optimasi parameter atau penyesuaian rasio data, dan sejalan dengan temuan Prastyo [15] yang menyatakan bahwa performa model sangat dipengaruhi oleh panjang teks, *noise*, dan distribusi kelas.

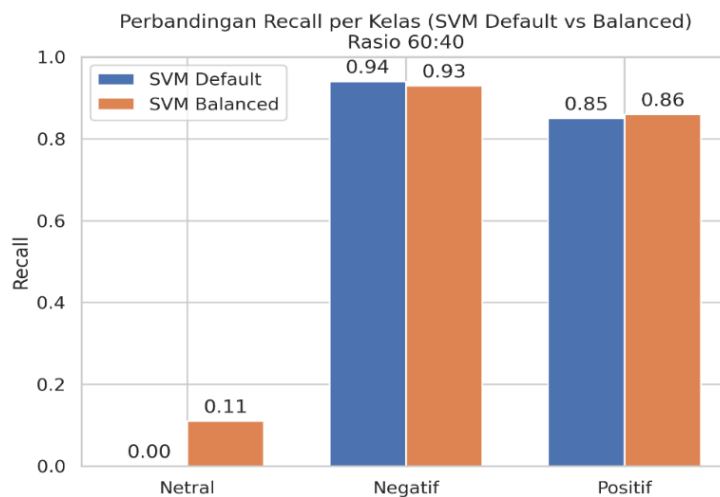
Pengujian lanjutan pada rasio data latih dan uji 60:40 menghasilkan peningkatan performa klasifikasi, khususnya pada kelas minoritas. Gambar 3 memperlihatkan perbandingan nilai *recall* pada masing-masing kelas. Terlihat bahwa pada konfigurasi *SVM Default*, *recall* untuk kelas netral sangat rendah, menandakan kesulitan model dalam mengenali kelas minoritas. Sebaliknya, pada *SVM Balanced*, *recall* kelas netral meningkat secara signifikan, menunjukkan bahwa penyesuaian bobot kelas dapat mengurangi bias terhadap kelas mayoritas. Gambar 4 memperkuat temuan tersebut dengan menampilkan perbandingan nilai *recall* antar kelas pada masing-masing konfigurasi. Balancing kelas terbukti efektif dalam meningkatkan sensitivitas model terhadap kelas netral,

yang sebelumnya cenderung diabaikan. Selanjutnya, Gambar 5 menyajikan perbandingan nilai *FI-score* antar kelas, yang menggabungkan aspek *precision* dan *recall*. Terlihat bahwa *SVM Balanced* menghasilkan performa klasifikasi yang lebih seimbang antar kelas dibandingkan *SVM Default* dan *LSTM*, menjadikannya pilihan yang lebih efektif dalam skenario data tidak seimbang



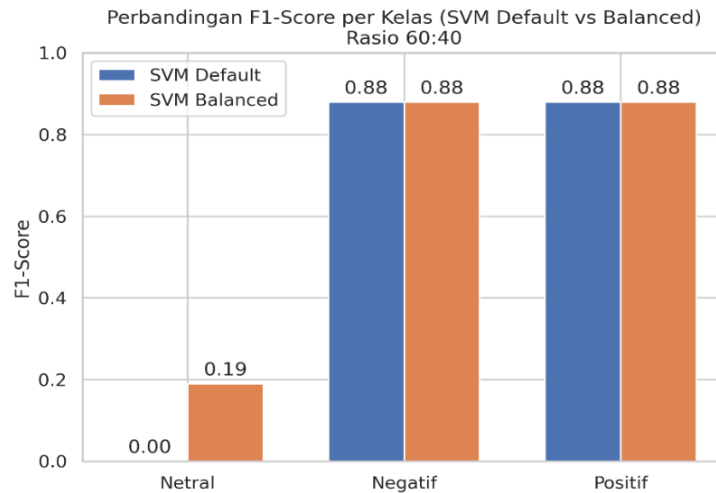
**Gambar 3.** Precision per kelas (*SVM Default vs Balanced*).

Sumber: hasil olahan peneliti (2025)



**Gambar 4.** Recall per kelas (*SVM Default vs Balanced*).

Sumber: hasil olahan peneliti (2025)



**Gambar 5.** *F1-Score per kelas (SVM Default vs Balanced).*

Sumber: hasil olahan peneliti (2025)

### 3.4 Analisis Kesalahan Klasifikasi

Meskipun algoritma *Support Vector Machine (SVM)* maupun *Long Short-Term Memory (LSTM)* memperlihatkan performa baik keseluruhan, masih terdapat kelemahan dalam klasifikasi sentimen, terutama pada kategori netral. Berdasarkan hasil evaluasi, model *SVM* cenderung gagal mengenali data berlabel netral, yang tercermin dari nilai *precision*, *recall*, dan *F1-score* sebesar 0.00 pada sebagian besar skenario pengujian. Hal ini dari total data netral lebih sedikit dibanding kelas lain (hanya 45 dari 1.000 data), sehingga model mengalami bias terhadap kelas mayoritas.

Model *LSTM*, meskipun lebih adaptif terhadap konteks kalimat, juga menunjukkan nilai *precision* dan *recall* yang relatif rendah pada kelas netral. Kemungkinan besar hal ini terjadi karena kompleksitas konteks netral yang cenderung ambigu dan kurang memiliki kata kunci emosional yang jelas, sehingga menyulitkan model untuk membedakannya dari sentimen positif ataupun negatif. Selain itu, distribusi data yang tidak seimbang menyebabkan model kurang terpapar variasi pola dalam kelas minoritas. Untuk mengatasi masalah ini, solusi dari penelitian selanjutnya antara lain adalah penerapan teknik *oversampling* (misalnya *SMOTE*), penyesuaian *threshold* klasifikasi, atau pengembangan model *hybrid* yang menggabungkan kekuatan *SVM* dan *LSTM* dalam satu *pipeline* analitik. Alternatif lainnya adalah menggunakan *fine-tuned Transformer* seperti *BERT* yang lebih sensitif terhadap konteks semantik halus.

## 4. KESIMPULAN

Dari hasil pengujian terhadap metode *Support Vector Machine (SVM)* maupun *Long Short-Term Memory (LSTM)* pada klasifikasi sentimen ulasan pengguna aplikasi K24Klik, keduanya menunjukkan keunggulan tersendiri. *SVM* memiliki performa stabil dan akurasi tinggi pada kelas negatif dan positif, namun kurang optimal untuk kelas netral, terutama saat distribusi data tidak seimbang. Sebaliknya, *LSTM* mampu mengklasifikasikan ketiga kelas secara lebih seimbang berkat kemampuannya memahami konteks teks, terutama pada rasio data 60:40, dengan akurasi tertinggi 84%.

Temuan ini memperlihatkan *LSTM* lebih unggul secara keseluruhan menangani data teks dengan sebaran kelas yang kompleks, sedangkan *SVM* tetap efektif untuk data yang lebih terstruktur dan seimbang. Model yang dikembangkan berpotensi diintegrasikan ke dalam sistem monitoring ulasan *real-time*, memungkinkan pengelola aplikasi seperti K24Klik dalam memantau sentimen pengguna dan meningkatkan layanan secara strategis. Penelitian selanjutnya disarankan untuk mengeksplorasi algoritma *Transformer* seperti *BERT* atau *RoBERTa*, serta menerapkan teknik penyeimbangan data seperti *SMOTE* atau *class weighting*. Penggunaan *dataset* yang lebih besar dan pengujian model *hybrid* dijadikan pendekatan untuk meningkatkan akurasi klasifikasi sentimen dalam layanan kesehatan digital.

## UCAPAN TERIMAKASIH

Penulis menyampaikan terima kasih kepada Universitas Semarang, khususnya Program Studi Sistem Informasi, atas dukungan dan fasilitas selama penelitian ini. Penghargaan juga ditujukan kepada keluarga saya, pasangan

saya, serta teman-teman atas doa, semangat maupun dukungan yang tak ternilai sepanjang proses penulisan karya ilmiah ini.

## REFERENCES

- [1] B. Priyatna and S. S. Hilabi, "Klasifikasi Sentimen Analisis Ulasan Aplikasi Alfagift Menggunakan Algoritma Long Short Term Memory.," STORAGE J. Ilm. Tek. dan Ilmu Komput., vol. 4, no. 2, pp. 48–55, 2025.
- [2] G. A. Ramadhan, "Analisis sentimen ulasan aplikasi ruangguru dengan algoritma long short term memory," (Bachelor's thesis, Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta), 2023.
- [3] Nurochman and A. Ashiilah, "Perbandingan SVM dan LSTM untuk Memprediksi Gangguan Kecemasan Berdasarkan Cuitan di Platform Aplikasi X (Twitter)," J. Teknol. Komput, vol. 10, no. 1, pp. 55–63, 2024.
- [4] W. Silalahi and A. Hartanto, "Klasifikasi sentimen support vector machine berbasis optimasi menyambut pemilu 2024.," JRST (Jurnal Ris. Sains dan Teknol., vol. 7, no. 2, pp. 245–255, 2023.
- [5] F. Damayanti, A. Rahim, and N. A. Verdikha, "Analisis Sentimen Ulasan Pengguna Terhadap Aplikasi K24klik Di Google Play Store Dengan Algoritma Naïve Bayes.," JATI (Jurnal Mhs. Tek. Inform., vol. 9, no. 2, pp. 3224–3230, 2025.
- [6] D. Darwis, N. Siskawati, and Z. Abidin, "Penerapan Algoritma Naive Bayes Untuk Analisis Sentimen Review Data Twitter Bmkg Nasional.," J. Tekno Kompak, vol. 15, no. 1, pp. 131–145, 2021.
- [7] S. Mutmainah, Khairunnas, and Khairunnisa, "Metode Deep Learning LSTM dalam Analisis Sentimen Aplikasi PeduliLindungi," Sci. J. Comput. Sci. Informatics, vol. 1, no. 1, pp. 9–19, 2024, doi: 10.34304/scientific.v1i1.231.
- [8] B. Hakim and P. Kinasih, "Sentiment Analysis of Indonesian Citizen Tweets Using Support Vector Machine on the Rebranding of Twitter to X.," J. Tekno Kompak, vol. 18, no. 2, pp. 468–479, 2024.
- [9] D. Pratama, S. Wijaya, A. S. Santosa, and P. S. Tamba, "Penerapan Neural Network Lstm Dalam Memprediksi Sentimen Pengguna Twitter Terhadap Bitcoin," J. TEKINKOM, vol. 6, no. 2, p. 2023, 2023, doi: 10.37600/tekinkom.v6i2.921.
- [10] R. Mitchell, *Web Scraping with Python: Collecting More Data from the Modern Web*. O'Reilly Media, 2018.
- [11] I. Busrayan and Andrianingsih, "Analisis Sentimen Pelanggan Terhadap Aplikasi Wondr By Bni Menggunakan Naïves Bayes, Support Vector Machine (Svm), Dan K-Nearest Neighbor (Knn)," J. Comput. Sci. Inf. Technol., vol. 2, no. 2, pp. 263–274, 2025.
- [12] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding.," Proc. 2019 Conf. North Am. chapter Assoc. Comput. Linguist. Hum. Lang. Technol. Vol. 1 (long short Pap., pp. 4171–4186, 2019, [Online]. Available: <https://aclanthology.org/N19-1423.pdf>
- [13] A. Wafda, "Aspect-Based Sentiment Analysis terhadap Cuitan Platform X tentang Kurikulum Merdeka Menggunakan IndoBERT," (Doctoral dissertation, Universitas Islam Indonesia), 2025.
- [14] D. F. N. Anisa, I. Mukhlash, and M. Iqbal, "Deteksi Berita Online Hoax Covid-19 Di Indonesia Menggunakan Metode Hybrid Long Short Term Memory dan Support Vector Machine.," J. Sains dan Seni ITS, vol. 11, no. 3, pp. A101–A108, 2023.
- [15] E. Prastyo, "Deteksi berita hoax dengan pendekatan Lexicon Based dan LSTM," (Doctoral dissertation, Universitas Islam Negeri Maulana Malik Ibrahim), 2024.