

PERBANDINGAN ALGORITMA *SUPPORT VECTOR MACHINE* DAN *LONG SHORT-TERM MEMORY* UNTUK KLASIFIKASI EMOSI MAHASISWA PADA PLATFORM X

Nazilatul Azza^{1*}, Nur Hadian²

Institut Teknologi Dan Sains Nahdlatul Ulama Pekalongan¹²

E-mail: nazilatulazza775@gmail.com^{1*}, nurhadian97@gmail.com²

Abstrak

Platform X banyak digunakan mahasiswa untuk mengekspresikan emosi terkait aktivitas akademik, namun belum dianalisis secara optimal. Penelitian ini bertujuan mengklasifikasikan emosi mahasiswa serta membandingkan kinerja *Support Vector Machine* (SVM) dan *Long Short-Term Memory* (LSTM). Metode yang digunakan adalah eksperimen komputasional dengan tahapan *crawling data*, *preprocessing*, pelabelan berbasis leksikon, serta pemodelan menggunakan TF-IDF pada SVM dan *word embedding* pada LSTM. Dataset yang digunakan sebanyak 2.914 data. Hasil menunjukkan SVM memperoleh akurasi 71,18%, lebih tinggi dibandingkan LSTM sebesar 68,95%. Hal ini menunjukkan SVM lebih stabil pada dataset terbatas, sedangkan LSTM cenderung mengalami *overfitting*. Penelitian ini menyimpulkan bahwa pemilihan algoritma harus disesuaikan dengan karakteristik data dan memberikan kontribusi dalam perbandingan metode machine learning dan deep learning.

Kata Kunci: *Analisis Sentimen; Emosi Mahasiswa; LSTM; Media Sosial X; SVM; TF-IDF*

Abstract

Platform X is widely used by students to express emotions related to academic activities, yet it has not been optimally analyzed. This study aims to classify student emotions and compare the performance of Support Vector Machine (SVM) and Long Short-Term Memory (LSTM). The method used is a computational experiment involving data crawling, preprocessing, lexicon-based labeling, and modeling using TF-IDF for SVM and word embedding for LSTM. The dataset consists of 2,914 data points. The results show that SVM achieved an accuracy of 71.18%, outperforming LSTM at 68.95%. This indicates that SVM is more stable on limited datasets, while LSTM tends to overfit. The study concludes that algorithm selection should

consider data characteristics and contributes to the comparison of machine learning and deep learning methods.

Keywords: *Sentiment Analysis; Student Emotion; LSTM; Social Media X; SVM; TF-IDF*

Submitted: 2026-03-08. Revision: 2026-03-20. Accepted: 2026-04-04. Publish: 2026-04-17.

PENDAHULUAN

Perkembangan teknologi informasi dan media sosial pada era sekarang terus memunculkan berbagai inovasi dalam membantu menjalankan berbagai aktivitas masyarakat (Bimantoro *et al.*, 2021). Salah satu bentuk pemanfaatannya adalah media sosial sebagai implementasi dari kemajuan teknologi dalam bidang komunikasi (Nisa *et al.*, 2021). Platform media sosial X menjadi contoh media yang banyak digunakan mahasiswa dalam mengekspresikan berbagai pengalaman akademik. Ekspresi yang disampaikan dalam bentuk komentar yang menjadi sumber informasi untuk memahami kondisi emosional mahasiswa secara tidak langsung (Swastika *et al.*, 2023). Namun, ekspresi emosi mahasiswa pada platform X umumnya masih bersifat tidak terstruktur dan belum dianalisis secara sistematis menggunakan pendekatan berbasis data, sehingga potensi informasi yang ada didalamnya belum dapat dimanfaatkan secara optimal untuk memahami kondisi emosional mahasiswa di lingkungan akademik.

Sejumlah penelitian sebelumnya telah mengkaji penggunaan algoritma machine learning maupun deep learning dalam analisis sentimen pada media sosial. Seperti Penelitian yang dilakukan oleh (Yanti *et al.*, 2024) membandingkan beberapa algoritma

machine learning dalam analisis sentimen ulasan aplikasi Digital Korlantas Polri dan menunjukkan bahwa *Support Vector Machine* (SVM) memperoleh persisis tertinggi sebesar 91,13%. penelitian lain oleh (Husen *et al.*, 2023) yang menganalisis opini publik terhadap Bank BSI pada Twitter juga menunjukkan bahwa metode machine learning mampu memberikan klasifikasi yang cukup baik. Selain itu, Penelitian oleh (Permata Aulia *et al.*, 2021) membandingkan berbagai kernel pada algoritma SVM dalam analisis sentimen vaksinasi COVID-19 dan menemukan bahwa kernel liner dan sigmoid menghasilkan akurasi tertinggi sebesar 0,87.

Penelitian yang dilakukan oleh (Sutranggono & Imah, 2023). Hasil penelitian ini menunjukkan penelitian menunjukkan bahwa SVM dengan kernel linear dan fitur TF-IDF menghasilkan akurasi tertinggi, yaitu 0.7506, dengan nilai *precision* 0.772, *recall* 0.748, dan *F1-score* 0.754. Kemudian ada Penelitian yang dilakukan oleh (Makhtum & Muhajir, 2023) menganalisis sentimen publik terhadap Undang-Undang Cipta Kerja (Omnibus Law) menggunakan data Twitter dan model *Support Vector Machine* (SVM) dengan beberapa kernel. Hasil penelitian menunjukkan bahwa SVM dengan kernel linear menghasilkan performa terbaik

dengan akurasi sebesar 97,06% dan nilai AUC 0,97, yang mengindikasikan tingkat klasifikasi yang sangat baik.

Selain pendekatan machine learning, beberapa penelitian juga menerapkan algoritma *deep learning* seperti *Long Short-Term Memory* (LSTM) dalam analisis sentimen. Penelitian oleh (Putra *et al.*, 2021) menunjukkan bahwa algoritma LSTM mampu menghasilkan akurasi sebesar 92% dalam klasifikasi ekspresi teks berbahasa jawa. Penelitian lainnya oleh (Azrul *et al.*, 2024) yang menganalisis emosi sentimen pengguna twitter mengenai perkembangan Artificial Intelligence menunjukkan bahwa model LSTM mampu mencapai akurasi sebesar 74,25%. Selain itu Penelitian oleh (Pipin & Kurniawan, 2022) juga menunjukkan bahwa penerapan LSTM menghasilkan akurasi sebesar 80,42%. Hasil penelitian tersebut menunjukkan bahwa baik metode *machine learning* maupun *deep learning* mempunyai kemampuan yang baik dalam melakukan analisis sentimen pada data teks media sosial.

Kemudian terdapat Penelitian oleh (Sanabila Khoirunnisa, 2025) mengkaji analisis sentimen pada media sosial Twitter terkait Pemilihan Presiden Indonesia 2024 menggunakan algoritma Long Short-Term Memory (LSTM). Hasil penelitian menunjukkan bahwa kombinasi LSTM, TF-IDF, Word2Vec, dan Adam mampu menghasilkan performa terbaik dengan akurasi 83,04% dan F1-score 82,62%. Selain itu terdapat penelitian lain yang dilakukan oleh (Romadhoni *et al.*, 2022) membahas analisis sentimen terhadap tanggapan

masyarakat mengenai PERMENDIKBUD No. 30 Tahun 2021 pada media sosial Twitter dengan membandingkan metode *Naïve Bayes* dan *Long Short-Term Memory* (LSTM). Hasil penelitian menunjukkan bahwa metode LSTM memiliki kinerja yang lebih baik dibandingkan *Naïve Bayes*, dengan nilai akurasi sebesar 77%, presisi 84%, recall 75%, dan F1-Score 80%, sedangkan *Naïve Bayes* memperoleh akurasi 76%.

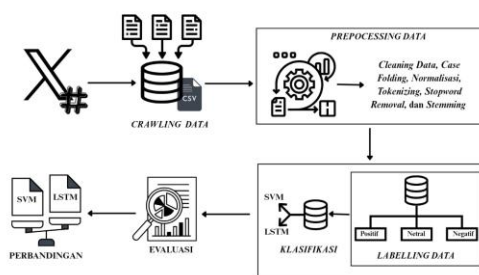
Meskipun berbagai penelitian sesudahnya telah menerapkan algoritma *machine learning* maupun *deep learning* dalam analisis sentimen pada media sosial, sebagian besar penelitian tersebut masih berfokus pada analisis sentimen umum seperti opini publik terhadap kebijakan, layanan, maupun teknologi tertentu. Selain itu, beberapa penelitian umumnya hanya menggunakan satu metode algoritma tanpa melakukan perbandingan secara mendalam antara metode *machine learning* dan *deep learning*. Penelitian yang membandingkan kinerja kedua pendekatan tersebut dalam konteks klasifikasi emosi mahasiswa pada media sosial, khususnya pada platform X, masih relatif terbatas.

Berdasarkan kondisi tersebut, penelitian ini mengusulkan perbandingan antara algoritma *Support Vector Machine* (SVM) sebagai representasi metode machine learning dan *Long Short-Term Memory* (LSTM) sebagai representasi metode deep learning dalam melakukan klasifikasi emosi mahasiswa pada platform X. Melalui

perbandingan kedua algoritma tersebut, penelitian ini dimaksudkan dapat memberikan kontribusi dalam mengetahui metode yang lebih efektif dalam mengklasifikasikan emosi mahasiswa pada data teks media sosial serta memperkaya kajian mengenai penerapan *machine learning* dan *deep learning* dalam analisis sentimen pada konteks akademik

METODE PENELITIAN

Penelitian ini memanfaatkan pendekatan kuantitatif melalui metode eksperimen komputasional untuk menilai kinerja algoritma *Support Vector Machine* (SVM) dan *Long Short-Term* (LSTM) dalam melakukan klasifikasi emosi mahasiswa dalam platform media sosial X. Pendekatan komparatif dilakukan dengan mengevaluasi performa kedua algoritma dengan memanfaatkan metrik evaluasi, yaitu *akurasi*, *precision*, *recall*, dan *F1-score*. Alur penelitian meliputi tahapan pengumpulan data, *preprocessing*, pelabelan data, *ekstraksi fitur*, pemodelan, serta evaluasi model yang ditunjukkan pada Gambar 1.



Gambar 1. Alur Penelitian

A. Pengumpulan Data

Pada tahap pengumpulan data dilakukan memanfaatkan metode *crawling* data pada platform media sosial X. pada proses pengambilan data dilakukan secara otomatis menggunakan *Google colab* dan memanfaatkan *library sncrape* (*library python* yang memungkinkan proses *scraping* data dari media sosial X tanpa memerlukan akses resmi API resmi).

Proses *crawling* dilakukan dengan menggunakan kata kunci #mahasiswa, #skripsi, dan #kampus yang telah ditentukan. Kemudian, *library Sncrape* digunakan untuk menelusuri dan mengumpulkan cuitan dengan batasan rentang waktu 04 Desember 2024 hingga 04 Desember 2025. Data yang diambil meliputi teks cuitan (*tweet*), tanggal unggahan, serta informasi tambahan yang relevan lainnya.

Seluruh data yang telah diperoleh selanjutnya disimpan dalam format CSV sebagai tahap awal sebelum dilakukan pengolahan selanjutnya. Dari proses *crawling* yang dilakukan, penelitian ini berhasil mengumpulkan dari proses *crawling* sebanyak 3020 data cuitan, dengan masing-masing kata kunci menghasilkan kurang lebih 1000 data. Selanjutnya, dataset ini digunakan sebagai bahan pada tahap pengolahan berikutnya.

B. Preprocessing Data

Tahap *preprocessing* dilakukan untuk membersihkan dan menyiapkan data teks agar lebih terstruktur. Proses ini meliputi *cleaning data*, *case folding*, *normalization*,

tokenizing, stopword removal, dan stemming.

Proses **Cleaning Data**, pada proses ini dilakukan untuk membersihkan teks dari elemen-elemen yang tidak relevan, seperti tanda baca, angka, hashtag, URL, situs web, serta karakter khusus lainnya yang tidak mendukung proses analisis. Setelah itu, dilakukan tahap **Case Folding** yang digunakan untuk mengubah dataset menjadi huruf kecil. Pada proses selanjutnya dilakukan **Normalization** digunakan untuk penyeragaman kata-kata tidak baku akan diubah menjadi kata baku. Proses berikutnya adalah **Tokenizing** digunakan untuk mengubah kalimat menjadi unit-unit yang lebih kecil yang disebut token. unit tersebut dapat berbentuk kata, frasa, maupun karakter, tergantung pada tingkat kerincian analisis yang diinginkan. Kemudian dilakukan proses **Stopword Removal** untuk menghapus atau menghilangkan kata-kata umum yang tidak memberikan kontribusi signifikan terhadap analisis. Proses terakhir yaitu **Stemming** di gunakan untuk mengkonversi kata-kata menjadi bentuk dasarnya dengan menghapus imbuhan, awalan, dan akhiran. Pada proses ini dilakukan dengan bantuan library Sastrawi (Emarapenta *et al.*, 2024).

C. Pelabelan Data

Selanjutnya, pelabelan dilakukan menggunakan pendekatan berbasis leksikon (*lexicon-based approach*). Dengan memberikan skor pada setiap kata untuk menentukan kategori sentimen (Nazilatul Azza, 2025; Ningsih *et al.*, 2023).

Selanjutnya dilakukan Pelabelan dengan cara menghitung jumlah kata yang termasuk dalam kategori positif dan negatif pada setiap teks. Setiap kata yang ditemukan dalam kamus positif diberikan skor +1, sedangkan kata yang ditemukan dalam kamus negatif diberikan skor -1.

D. Feature Extraction

Pada tahap ekstraksi fitur, pada model *Support Vector Machine* (SVM), menggunakan metode *Term Frequency–Inverse Document Frequency* (TF-IDF) dalam mempresentasikan teks kedalam numerik, metode ini digunakan untuk memberikan bobot berdasarkan tingkat kemunculan dan tingkat keunikannya dalam suatu dokumen (Westley *et al.*, 2022). Proses ini diimplementasikan menggunakan fungsi *TfidfVectorizer* pada library *scikit-learn* pada Python.

Sedangkan pada model *Long Short-Term Memory* (LSTM) digunakan representasi teks berbasis *sekuens* kata (*word sequence*) yang kemudian dikonversi menjadi *word embedding*. Pada tahap ini setiap kata dalam teks diubah menjadi indeks numerik menggunakan proses tokenisasi, kemudian dilakukan proses padding untuk menyeragamkan panjang sekuens pada setiap kata. Kemudian kata-kata direpresentasikan dalam bentuk vektor numerik menggunakan embedding layer (Maulana *et al.*, 2023).

E. Klasifikasi Model SVM LSTM

Pada proses klasifikasi dilakukan dengan membangun model *Support Vector*

Machine (SVM) menggunakan library *scikit-learn* pada bahasa pemrograman python dan kernel yang digunakan pada model ini yaitu kernel linear, karena kernel ini sering digunakan dalam klasifikasi teks dan memiliki performa yang baik pada data dengan dimensi fitur yang tinggi seperti hasil representasi TF-IDF. Kemudian model dilatih memakai data latih yang telah dibagi sebelumnya dengan proporsi 80% data latih dan 20% data uji.

F. Klasifikasi Model LSTM

Pada Proses klasifikasi ini digunakan untuk membangun model *Long Short-Term Memory* (LSTM) dengan menggunakan library *TensorFlow* dan *Keras* pada bahasa pemrograman Python. Pada tahap awal data teks yang sudah melalui proses *preprocessing* diganti menjadi *sekuens* numerik memakai proses tokenisasi, selanjutnya dilakukan padding untuk menyeragamkan panjang sekuens pada setiap kata, kemudian data dimasukkan kedalam embedding layer dengan ukuran 128 untuk menghasilkan representasi vektor kata.

Arsitektur model yang digunakan terdiri dari dua lapisan Bidirectional LSTM, dengan jumlah 64 neuron pada lapisan pertama dan 32 neuron pada lapisan kedua. Untuk mengurangi resiko *overfitting*, ditambahkan lapisan dropout sebesar 0,5. Model dilanjutkan ke lapisan *fully connected* dengan fungsi aktivasi *softmax* untuk melakukan klasifikasi kedalam ketiga kelas sentimen.

Model dilatih menggunakan algoritma *Optimizer Adam* dengan fungsi kerugian *loss sparse categorical crossentropy*. Proses pelatihan berlangsung selama 10 *epoch* dengan *batch size* sebesar 32, berdasarkan pembagian data sebesar 80% untuk pelatihan dan 20% untuk pengujian.

G. Evaluasi Model SVM dan LSTM

Pada proses ini merupakan proses terakhir dalam penelitian yang digunakan untuk menghitung serta memberikan gambaran kinerja dan efektivitas dari model klasifikasi emosi yang sudah dilakukan. Proses evaluasi model dibuat dengan memanfaatkan beberapa metrik pengukuran, seperti *akurasi*, *presisi*, *recall*, dan *F1-score* (Safitri et al., 2025).

Confusion matrix terdiri dari empat komponen utama, yaitu True Positive, True Negative, False Positive, dan False Negative. True Positive menunjukkan jumlah data yang diprediksi positif dan sesuai dengan kondisi sebenarnya, sedangkan True Negative menunjukkan jumlah data yang diprediksi negatif dan memang negatif. False Positive merupakan data yang diprediksi positif, namun kenyataannya bernilai negatif, sedangkan False Negative adalah data yang diprediksi negatif, tetapi sebenarnya bernilai positif (Nugroho et al., 2021).

Berdasarkan nilai-nilai tersebut, *accuracy* dimanfaatkan untuk menguji peringkat ketepatan model dalam mengklasifikasikan data secara keseluruhan, *precision* diterapkan untuk mengetahui

tingkatan ketepatan prediksi model terhadap setiap kelas emosi, *recall* diterapkan untuk mengukur seberapa mampu model dalam mengenali data yang benar pada masing-masing kelas, dan sementara *F1-score* diterapkan sebagai pengukur keseimbangan antara nilai presisi dengan *recall*.

H. Perbandingan Model SVM dan LSTM

Proses perbandingan model dilakukan untuk menganalisis perbedaan kinerja antara algoritma *Support Vector Machine* (SVM) maupun *Long Short-Term Memory* (LSTM) dalam mengklasifikasikan emosi mahasiswa pada platform media sosial X. Pada tahap ini, hasil evaluasi yang sudah didapatkan dari masing-masing algoritma akan dibandingkan dengan menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. Nilai pada setiap metrik akan dianalisis untuk melihat dan mengambil keunggulan dan kelemahan dari masing-masing algoritma dalam mengenali pola emosi pada dataset. Selain itu, perbandingan dilakukan berdasarkan konsistensi hasil klasifikasi dan kemampuan model dalam menangani variasi data.

HASIL DAN PEMBAHASAN

A. Hasil

Pada proses pengumpulan data, dilakukan dengan metode *crawling* pada platform media sosial X menggunakan bahasa *python* melalui *google colab*. Data yang diambil berdasarkan *hashtag* atau kata kunci yang terdiri dari #skripsi, #mahasiswa, dan #kampus dalam rentang 4 desember

2024 hingga 4 desember 2025. Dari setiap kata kunci berhasil mengumpulkan masing-masing kurang lebih 1000 cuitan atau komentar berbahasa Indonesia, hingga total data *hashtag* yang didapatkan berjumlah 3020 cuitan. Data hasil *crawling* disimpan dalam format *csv* dan digunakan sebagai dataset awal untuk tahap pengolahan dan analisis selanjutnya.

1. *Preprocessing Data*

Hasil pada *preprocessing* menunjukkan bahwa jumlah data berkurang dari 3020 data awal, setelah dilakukan proses *cleaning*, jumlah data menjadi 2914 data. Proses *preprocessing* meliputi *cleaning*, *case folding*, *normalization*, *tokenizing*, *stopword removal*, dan *stemming*. Hasil akhir *preprocessing* disajikan pada Tabel 1.

Pada tahap 1 yaitu *cleaning* data, tahap digunakan untuk membuang komponen-komponen yang tidak relevan dengan penelitian seperti simbol, *URL*, username, data duplikat, *hashtag*, angka, tanda baca, dan emoji.

Pada Tahap 2 yaitu *case folding* mengganti semua huruf didalam dataset menjadi huruf kecil, seperti kata "Kenapa" menjadi "kenapa".

Pada Tahap 3 yaitu normalisasi untuk menyeragamkan penulisan kata tidak baku, singkatan serta kata tidak formal dengan menggunakan file kamus_baku khusus seperti kata "kuu" menjadi "ku".

Pada tahap 4, *tokenizing* dilakukan untuk memecah atau membagi teks menjadi

unit-unit kata. Serperti kata “kenapa kalau di kampus” menjadi “[‘kenapa’, ‘kalau’, ‘di’, ‘kampus’]”.

Pada Tahap 5, *filtering* atau *stopword removed*, unit-unit pada setiap kata akan dihilangkan salah satu unit kata yang tidak mempunyai makna emosi atau kata penghubung, seperti kata “[‘kenapa’, ‘kalau’, ‘di’, ‘kampus’]” menjadi “kampus”

Pada Tahap 6, *stemming* menggunakan algoritma *stemming Nazief* dan *Adriani* yang digunakan untuk membuang imbuhan awal (*prefix*), imbuhan akhir (*suffix*) serta imbuhan sisipan (*infix*) agar semua unit kata sesuai dengan kaidah morfologi bahasa indonesia. Sampel hasil *Stemming* dapat di lihat pada Tabel 1.

Tabel 1. Sampel Hasil *Stemming*

<i>Stopword Removed</i>	<i>Stemming</i>
[‘kampus’, ‘terserap’, ‘energi’, ‘ku’, ‘cepat’, ‘pulang’]	kampus serap energi ku cepat pulang

2. Labeling Data

Setelah melalui prosespreprocessing data, data uji_df selanjutnya diproses pada tahap labelling data. Pada penelitian ini, proses labelling menerapkan metode berbasis *leksikon based approach* dengan menggunakan kamus kata positif dan negatif yang diambil dari github. Pada proses ini setiap komentar akan diberikan skor +1 jika positif dan skor -1 jika negatif. Hasil dari proses pelabelan data ini menghasilkan

klasifikasi data sentimen dibagi menjadi tiga kelas utama, yaitu positif, negatif, dan netral dengan dristribusi yang disajikan pada Tabel 2. Sampel hasil dari pelabelan pada data uji_df dapat dilihat pada Tabel 3.

Tabel 2. Sampel Hasil *Labelling Data*

<i>Stopword Removed</i>	<i>Score</i>	<i>Label</i>
[‘kampus’, ‘terserap’, ‘energi’, ‘ku’, ‘cepat’, ‘pulang’]	1	Positif

Tabel 3. Distribusi kelas sentimen

<i>Kategori</i>	<i>Jumlah</i>	<i>Persentase</i>
Positif	1.194	40,97%
Negatif	958	32,88%
Netral	762	26,15%

Pada Tabel 3 diatas menunjukkan bahwa kelas sentimen positif memiliki jumlah data terbanyak dibandingkan dengan kelas lainnya.

3. Feature Extraction

Proses ekstraksi fitur menggunakan metode TF-IDF menghasilkan matriks fitur berukuran (2914 × 9172), yang menunjukkan terdapat 2914 data dan 9172 fitur kata unik. Kata-kata seperti “kampus”, “skripsi”, dan “mahasiswa” memiliki bobot tertinggi dan memiliki kontribusi dalam representasi fitur seperti yang ditunjukkan pada Gambar 2.

```

Feature Extraction Selesai!!!
===== Bentuk Transformasi Feature Extraction =====
(2914, 9172)

===== Top 20 Terms by Total TF-IDF Score =====
term      tfidf_sum
4173      kampus    135.810789
7692      skripsi   121.565737
5907      mahasiswa 100.392479
9054      ya        82.000434
2656      gue       65.266203
669       banget    53.219694
192       akun      48.451453
4013      joki      45.999817
4256      kayak     39.231101
4140      kak       37.433411
6094      orang     35.630630
276       amp       34.910086
8511      tugas     33.650099
927       bio       33.577415
7611      sih       33.091553
1068      branding  31.915370
3884      jam       31.696616
9163      zonaung   29.345654
2862      hidup     28.238913
4639      kuliah    28.101985

```

Gambar 2. Visualisasi *Feature Extraction*

4. Pemodelan dan Evaluasi *Support Vector Machine (SVM)*

Hasil implementasi pemodelan menerapkan model *Support Vector Machine (SVM)* ditunjukkan pada Gambar 3, melalui hasil pengujian terhadap 583 data uji. Model SVM memperoleh *akurasi* sebesar 71,18%, yang menunjukkan kemampuan klasifikasi sentimen yang cukup baik dengan memanfaatkan fitur TF-IDF. Berdasarkan *classification report*, kelas positif menunjukkan performa terbaik dengan nilai *precision* 0,74, *recall* 0,86, dan *F1-score* 0,79 dari 235 data, yang mengindikasikan model mampu mengenali sebagian besar sentimen positif dengan baik. Pada kelas negatif, model menghasilkan nilai *precision* 0,80, *recall* 0,68, dan *F1-score* 0,73 dari 207 data, yang menunjukkan prediksi sentimen negatif memiliki tingkat ketepatan yang tinggi. Sementara itu, kelas netral menyajikan performa terendah dengan nilai *precision* 0,54, *recall* 0,51, dan *F1-score* 0,53 dari 141 data, yang mengindikasikan bahwa model SVM masih mengalami kesulitan dalam membedakan sentimen

netral dengan sentimen positif maupun negatif.

```

===== HASIL SVM =====
Akurasi: 0.7118

Classification Report:
              precision    recall  f1-score   support

   Negatif      0.80      0.68      0.73      207
   Netral       0.54      0.51      0.53      141
   Positif      0.74      0.86      0.79      235

 accuracy              0.71      583
 macro avg            0.69      0.68      0.68      583
 weighted avg         0.71      0.71      0.71      583

Confusion Matrix:
[[140  38  29]
 [ 25  72  44]
 [  9  23 203]]

```

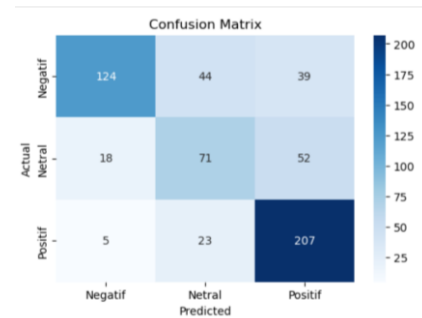
Gambar 3. Hasil Implementasi SVM

Gambar 4. WordCloud Consusion Matrix SVM

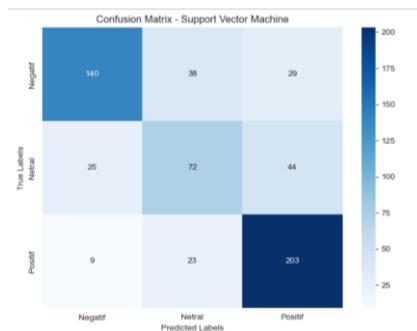
5. Pemodelan dan Evaluasi *Long Short-Term Memory (LSTM)*

Proses Pemodelan sentimen mengimplementasikan model *Long Short-Term Memory (LSTM)* dimulai dengan mengambil data uji_df yang sudah melalui tahap preprocessing sampai tekenizing sampai dalam bentuk matrik sekuens yaitu X_train_padded dan X_test_padded, yang selanjutnya digunakan sebagai masukan ke dalam model LSTM. Selanjutnya data tersebut diproses menggunakan *Embedding* dengan dimensi vektor 128, serta menggunakan model dua lapisan *Bidirectional LSTM*, dengan masing-masing 64 neuron pada lapisan pertama dan 32 neuron pada lapisan kedua. Penggunaan *Bidirectional LSTM* digunakan agar model dapat memungkinkan mempelajari konteks kata dari dua arah, yaitu maju dan mundur, sehingga meningkatkan pemahaman dalam urutan kata dalam teks.

Untuk mengurangi resiko terjadinya *overfitting*, lapisan dropout ditambahkan dengan nilai sebesar 0,5. Kemudian hasil dari lapisan LSTM diteruskan menuju lapisan *fully connected* dengan 64 unit menggunakan fungsi aktivasi *softmax*, sesuai dengan jumlah kelas sentimen yang digunakan. Model ini disusun dengan menerapkan optimizer Adam dengan fungsi *loss sparse categorical crossentropy* untuk mengevaluasi kesalahan dalam prediksi selama proses pelatihan.



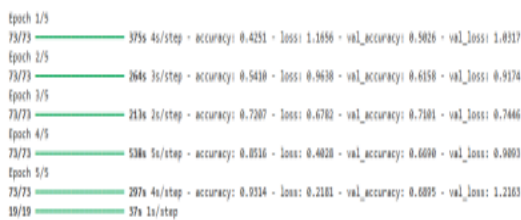
Gambar 7. WordCloud Confusion Matrix LSTM



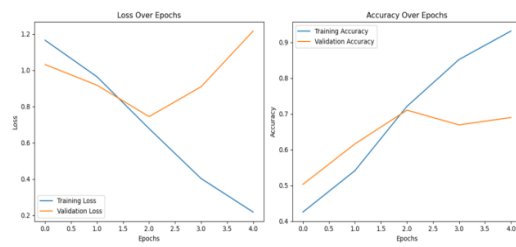
Accuracy: 0.6895368782161235
Accuracy: 68.95%

	precision	recall	f1-score	support
Negatif	0.84	0.60	0.70	207
Netral	0.51	0.50	0.51	141
Positif	0.69	0.88	0.78	235
accuracy			0.69	583
macro avg	0.68	0.66	0.66	583
weighted avg	0.70	0.69	0.68	583

Gambar 8. Hasil Implementasi LSTM



Gambar 5. Pelatihan Model LSTM



Gambar 6. Grafik Training loss

6. Perbandingan Pemodelan algoritma Support Vector Machine (SVM) dan Long Short-Term Memory (LSTM)

Perbandingan algoritma Support Vector Machine (SVM) dan Long Short-Term Memory (LSTM) dilakukan agar dapat mengetahui performa model algoritma yang memberikan klasifikasi sentimen yang terbaik. Evaluasi dilakukan memanfaatkan metrik *akurasi* serta analisis performa pada masing-masing kelas sentimen berdasarkan *classification report*.

Tabel 4. Hasil Evaluasi Algoritma SVM dan LSTM

Model	Hasil Akurasi
SVM	71,18%
LSTM	68,95%

Berdasarkan Tabel 8 hasil evaluasi, model SVM mendapatkan *akurasi* sebesar 71,18%, sedangkan model LSTM mendapatkan *akurasi* sebesar 68,95%. Hasil evaluasi ini menunjukkan model algoritma SVM memiliki tingkat *akurasi* sedikit lebih tinggi dibandingkan dengan model algoritma LSTM pada data uji. Keunggulan model SVM mendeteksi bahwa representasi fitur berbasis TF-IDF cukup efektif dalam menangkap pola sentimen secara keseluruhan, serta memberikan performa yang stabil dengan kompleksitas komputasi yang relatif rendah.

Disisi lain, dilihat dari sisi karakteristik pembelajaran, model algoritma LSTM menunjukkan keunggulan dalam memahami konteks urutan kata. Hal ini terlihat dari performa LSTM yang relatif baik pada kelas sentimen positif, dengan nilai *recall* yang cukup tinggi, yang memandakan kemampuan model dalam mengenali sebagian besar data positif.

B. Pembahasan

Berdasarkan hasil penelitian, model *Support Vector Machine* (SVM) menunjukkan kinerja lebih unggul disandingkan dengan *Long Short-Term Memory* (LSTM) dengan akurasi masing-masing sebesar 71,18% dan 68,95%. Selain itu, distribusi data menunjukkan bahwa kelas sentimen positif merupakan kelas dominan dengan jumlah 1.194 data (40,97%) serta dengan nilai *recall* sebesar 0,86 dan F1-score 0,79, penelitian ini sejalan dengan penelitian terdahulu oleh (Yanti *et al.*, 2024) yang menunjukkan bahwa model *Support*

Vector Machine (SVM) memiliki performa stabil pada analisis sentimen akurasi tinggi.

Keunggulan model *Support Vector Machine* (SVM) dipengaruhi oleh penggunaan TF-IDF yang efektif dalam merepresentasikan teks berbasis frekuensi dan kepentingan kata. Selain itu, penggunaan kernel linear pada model SVM juga berperan penting dalam memberikan kontribusi dalam memisahkan data secara optimal pada ruang fitur berdimensi tinggi. Secara teoritis, kernel linear sangat efektif untuk data teks karena mampu menangani jumlah fitur yang besar tanpa memerlukan transformasi kompleks, hal ini sejalan pada penelitian (Permata Aulia *et al.*, 2021) yang menghasilkan akurasi tinggi menggunakan pembobotan TF-IDF dan kernel linear. Sebaliknya LSTM.

Sebaliknya, model LSTM menunjukkan akurasi sebesar 68,95% dengan performa kelas netral yang relatif rendah (F1-score 0,511), yang mengindikasikan keterbatasan model dalam membedakan konteks pada dataset yang terbatas, selain itu pada grafik pelatihan menunjukkan adanya peningkatan akurasi data latih 0,93, sementara akurasi validasi hanya mencapai sekitar 0,71 yang menandakan adanya gejala *overfitting*. Namun, hasil ini berbeda dengan penelitian (Putra *et al.*, 2021) yang menunjukkan bahwa model *Long Short-Term Memory* (LSTM) mampu mencapai akurasi hingga 92% yang disebabkan oleh penggunaan dataset yang lebih besar. Selain itu, penelitian oleh (Azrul *et al.*, 2024) memperoleh akurasi 74,25% pada model LSTM yang menunjukkan bahwa performa deep learning sangat bergantung

pada jumlah dan kualitas data yang digunakan.

Hasil penelitian ini menunjukkan bahwa kombinasi TF-IDF dan kernel linear pada SVM lebih efektif dalam menangani dataset terbatas, sedangkan model LSTM memiliki potensi performa yang lebih tinggi jika didukung dataset yang lebih besar dan representatif.

KESIMPULAN

Temuan penelitian mengindikasikan bahwa model *Support Vector Machine* (SVM) memiliki kinerja lebih unggul dibandingkan dengan *Long Short-Term Memory* (LSTM) dalam klasifikasi emosi mahasiswa pada platform X. Dengan akurasi sebesar 71,18% dibandingkan 68,95%. SVM lebih stabil pada dataset terbatas berbasis TF-IDF, sedangkan LSTM cenderung mengalami *overfitting* meskipun mampu memahami konteks urutan kata. Temuan ini menegaskan bahwa penggunaan algoritma harus disesuaikan dengan karakteristik dan ukuran dataset. Oleh sebab itu, penelitian selanjutnya direkomendasikan menerapkan dataset yang lebih besar dan beragam serta melakukan optimasi parameter atau arsitektur model, khususnya pada LSTM, agar dapat meningkatkan performa klasifikasi secara optimal.

DAFTAR PUSTAKA

Azrul, A., Irma Purnamasari, A., & Ali, I. (2024). Analisis Sentimen Pengguna Twitter Terhadap Perkembangan Artificial Intelligence Dengan Penerapan Algoritma Long Short-Term

Memory (Lstm). *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(1), 413–421.

<https://doi.org/10.36040/jati.v8i1.8416>

Bimantoro, A., Pramesti, W. A., Bakti, S. W., Samudra, M. A., & Amrozi, Y. (2021). Paradoks Etika Pemanfaatan Teknologi Informasi di Era 5.0. *Jurnal Teknologi Informasi*, 7(1), 58–68.

<https://doi.org/10.52643/jti.v7i1.1425>

Emarapenta, J., Sinulingga, B., Cesar, H., & Sitorus, K. (2024). Analisis Sentimen Masyarakat terhadap Film Horor Indonesia Menggunakan Metode SVM dan TF-IDF Sentiment Analysis of Public towards Indonesian Horror Films Using SVM and TF-IDF Methods. 14(April), 42–53.

Husen, R. A., Astuti, R., Marlia, L., Rahmaddeni, R., & Efrizoni, L. (2023). Analisis Sentimen Opini Publik pada Twitter Terhadap Bank BSI Menggunakan Algoritma Machine Learning. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 3(2), 211–218.

<https://doi.org/10.57152/malcom.v3i2.901>

Jurnalis Pipin, S., & Kurniawan, H. (2022). Analisis Sentimen Kebijakan MBKM Berdasarkan Opini Masyarakat di Twitter Menggunakan LSTM. *Jurnal SIFO Mikroskil*, 23(2), 197–208.

<https://doi.org/10.55601/jsm.v23i2.900>

Makhtum, A. R., & Muhajir, M. (2023). SENTIMENT ANALYSIS OF OMNIBUS LAW USING SUPPORT VECTOR MACHINE (SVM) WITH LINEAR KERNEL. *BAREKENG: Journal of Mathematics and Its Applications*, 17(4), 2197–2206.

<https://doi.org/10.30598/barekengvol17iss4pp2197-2206>

Maulana, A. R., Wijoyo, S. H., & Mursityo, Y. T. (2023). *Sekolah Dasar Dan*

- Sekolah Menengah Pada Media Sosial Twitter Dengan Menggunakan Metode Word Embedding Dan Long Short-Term Memory Networks (LSTM) Sentiment Analysis Of Implementation Independent Curriculum Policy Elementary And Secondary School On Twitte.* 10(3), 523–530. <https://doi.org/10.25126/jtiik.2023106977>
- Nazilatul Azza, N. H. (2025). Tweet Sentiment Analysis with Support Vector Machine (SVM) Algorithm for PT. XYZ Digital Strategy. *G-Tech*, 9(4), 1868–1877. <https://doi.org/https://doi.org/10.70609/g-tech.v9i4.7887>
- Ningsih, A., Indonesia, S., Bahasa, F., Surabaya, U. N., Indonesia, S., Bahasa, F., & Surabaya, U. N. (2023). *PENGUNAAN LEKSIKON BAHASA INDONESIA BAKU PADA LEMBAGA DALAM PEMARTABATAN BAHASA INDONESIA DI RUANG PUBLIK (STUDI PERKUMPULAN RUANG BELAJAR AQIL)*. 10.
- Nisa, R., Amriza, S., & Supriyadi, D. (2021). Komparasi Metode Machine Learning dan Deep Learning untuk Deteksi Emosi pada Text di Sosial Media. *Penelitian Ilmu Dan Teknologi Komputer*, 13(2), 130–139.
- Nugroho, K. S., Akbar, I., Suksmawati, A. N., Komputer, I., Brawijaya, U., Mada, U. G., Teknik, F., & Malang, U. W. (2021). Deteksi Depresi dan Kecemasan Pengguna Twitter Menggunakan Bidirectional LSTM. *ArXiv, Ciastech*, 287–296. <https://doi.org/https://doi.org/10.48550/arXiv.2301.04521>
- Permata Aulia, T. M., Arifin, N., & Mayasari, R. (2021). Perbandingan Kernel Support Vector Machine (Svm) Dalam Penerapan Analisis Sentimen Vaksinisasi Covid-19. *SINTECH (Science and Information Technology) Journal*, 4(2), 139–145. <https://doi.org/10.31598/sintechjournal.v4i2.762>
- Putra, O. V., Musthafa, A., & Wibowo, K. P. (2021). Klasifikasi Ekspresi Teks Berbahasa Jawa Menggunakan Algoritma Long Short Term Memory. *Komputika : Jurnal Sistem Komputer*, 10(2), 137–143. <https://doi.org/10.34010/komputika.v11i1.4616>
- Romadhoni, Y., Fahmi, K., Holle, H., Informatika, J. T., Islam, U., Maulana, N., Ibrahim, M., & Malang, K. (2022). Analisis Sentimen Terhadap PERMENDIKBUD No . 30 pada Media Sosial Twitter Menggunakan Metode Naive Bayes dan LSTM. *JPIT*, 7(2), 118–124. <https://doi.org/10.30591/jpit.v7i2.3191>
- Safitri, A., Firmansyah, I., Yani, F., Kurniawan, M. A., & Nuryamin, Y. (2025). ANALISIS SENTIMEN RESPON PENGGUNA CHAT GPT MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE. *Jurnal Inovasi Pendidikan Dan Teknologi Informasi (JIPTI)*, 6(2), 553–566. <https://doi.org/10.52060/jipti.v6i2.3810>
- Sanabila Khoirunnisa, E. B. S. (2025). Sentiment Analysis on Social Media Using Long Short- Term Memory and Word2Vec Feature Expansion Methods with Adam Optimization. *Khazanah Informatika*, 11(1). <https://doi.org/https://doi.org/10.23917/khif.v11i1.3957>
- Sutranggono, A. N., & Imah, E. M. (2023). Tweets Emotions Analysis of Community Activities Restriction as COVID-19 Policy in Indonesia Using Support Vector Machine. *CommIT*

Journal, 17(1), 13–25.

<https://doi.org/10.21512/commit.v17i1.8189>

Swastika, W., Widodo, R. B., & Oepojo, A. A. (2023). Perbandingan Akurasi Deteksi Emosi Pada Suara Menggunakan Multilayer Perceptron, Random Forest, Decision Tree dan K-NN. *Journal of Intelligent System and Computation*, 5(1), 17–22.

<https://doi.org/10.52985/insyst.v5i1.264>

Westley, V., Thomas, D., & Rumaisa, F. (2022). Analisis Sentimen Ulasan Hotel Bahasa Indonesia Menggunakan Support Vector Machine dan TF-IDF. 6, 1767–1774.

<https://doi.org/10.30865/mib.v6i3.4218>

Yanti, Y., Hidayat, T., & Safana, N. (2024). *G-Tech : Jurnal Teknologi Terapan Deteksi Serangan Distributed Deniel of Service Pada Jaringan Sensor Nirkabel*. 8(4), 2687–2697.