Jurnal Riset Aplikasi Ilmu Data dan Sistem 1 (1), 33–46 DOI: 0000
Juli 2026

Analisis Komparatif Performa Machine Learning dan Deep Learning untuk Pemetaan Sentimen Publik Global Terhadap ChatGPT di Twitter (X)

Devin Aulia Asshafa, Yuliani Puji Astuti, Dinda Galuh Guminta*
Program Studi S1 Sains Data, Universitas Negeri Surabaya, Indonesia

Abstrak

Perkembangan teknologi *Generative Artificial Intelligence* mendorong meningkatnya percakapan publik mengenai ChatGPT di berbagai platform digital, khususnya Twitter (X). Dinamika opini tersebut penting untuk dipetakan guna memahami persepsi masyarakat global terhadap penggunaan dan dampak teknologi ini. Penelitian ini bertujuan untuk menganalisis sentimen publik global terhadap ChatGPT serta membandingkan performa model *Machine Learning* dan *Deep Learning* dalam klasifikasi sentimen berbasis *Natural Language Processing* (NLP). Data penelitian berupa tweet berbahasa Inggris dikumpulkan melalui metode *web scraping* dan diproses melalui tahapan *preprocessing*, kemudian dilakukan pelabelan otomatis menggunakan model RoBERTa ke dalam kelas positif, netral, dan negatif. Representasi fitur menggunakan TF-IDF pada model *Machine Learning* dan GloVe Twitter 200d pada model Deep Learning. Model yang digunakan meliputi *Logistic Regression* (LR), *Support Vector Machine* (SVM), *Convolutional Neural Network* (CNN), dan *Dilated CNN*. Hasil penelitian menunjukkan bahwa sentimen netral mendominasi data sebesar 43%, diikuti sentimen negatif sebesar 31% dan positif sebesar 26%. Pada skenario balanced, model terbaik diperoleh oleh LR dan Linear SVM dengan nilai *accuracy* dan F1-score sebesar 0.76, sedangkan CNN memperoleh F1-score sebesar 0.73 dan *Dilated CNN* sebesar 0.70. Hasil *cross-validation* menunjukkan seluruh model memiliki generalisasi yang baik dengan mean CV score pada rentang 0.73–0.74. Selain itu, model *Machine Learning* menunjukkan efisiensi komputasi yang lebih baik dibandingkan model *Deep Learning* berdasarkan waktu pelatihan yang lebih singkat. Secara keseluruhan, penelitian ini menunjukkan bahwa pendekatan *Machine Learning* berbasis TF-IDF masih efektif dan kompetitif untuk klasifikasi sentimen teks pendek pada media sosial.

Kata Kunci *Analisis Sentimen; ChatGPT; Machine Learning; Deep Learning; NLP; Twitter (X)*

Abstract

The development of Generative Artificial Intelligence technology has increased public discussions about ChatGPT across various digital platforms, particularly Twitter (X). These opinion dynamics are important to analyze in order to understand global public perceptions regarding the use and impact of this technology. This study aims to analyze global public sentiment toward ChatGPT and compare the performance of Machine Learning and Deep Learning models in sentiment classification using a Natural Language Processing (NLP) approach. The dataset consisted of English-language tweets collected through web scraping and processed through sev-

*Corresponding author. Email: yulianipuji@unesa.ac.id or dindaguminta@unesa.ac.id.

Received 25 Mei 2026; Revised 14 Juni 2026; Accepted 8 Juli 2026

eral preprocessing stages, followed by automatic sentiment labeling using the RoBERTa model into positive, neutral, and negative classes. Feature representation was performed using TF-IDF for Machine Learning models and GloVe Twitter 200d for Deep Learning models. The models used in this study include Logistic Regression (LR), Support Vector Machine (SVM), Convolutional Neural Network (CNN), and Dilated CNN. The results showed that neutral sentiment dominated the dataset at 43%, followed by negative sentiment at 31% and positive sentiment at 26%. In the balanced scenario, the best performance was achieved by LR and Linear SVM with an accuracy and F1-score of 0.76, while CNN achieved an F1-score of 0.73 and Dilated CNN achieved 0.70. Cross-validation results indicated that all models had good generalization performance with mean CV scores ranging from 0.73 to 0.74. In addition, Machine Learning models demonstrated better computational efficiency than Deep Learning models based on shorter training times. Overall, this study shows that TF-IDF-based Machine Learning approaches remain effective and competitive for short-text sentiment classification on social media.

Keywords *Sentiment Analysis; ChatGPT; Machine Learning; Deep Learning; NLP; Twitter (X)*

1 Pendahuluan

Generative AI, khususnya ChatGPT yang dikembangkan OpenAI, mencatat rekor pertumbuhan pengguna tercepat dalam sejarah—mencapai 100 juta pengguna hanya dalam dua bulan sejak rilis akhir 2022 (Stokel-Walker, 2023). Popularitas ini diikuti beragam respons publik: sebagian mengapresiasi peningkatan produktivitas, sementara lainnya mengkhawatirkan dampak terhadap keaslian karya, privasi, dan lapangan kerja (Shen et al., 2023). Penelitian Xu et al. (2023) terhadap 314.645 *tweet* menunjukkan 35,28% bersentimen positif, 45,79% netral, dan 18,92% negatif.

Twitter (X) menjadi *platform* ideal untuk analisis opini publik karena sifatnya yang terbuka, dinamis, dan *real-time* (Stieglitz et al., 2020). Pengolahan data teks berskala besar membutuhkan pendekatan NLP, ML, dan DL. ML efektif memetakan representasi TF-IDF ke kategori sentimen, sementara DL melalui CNN mampu mempelajari representasi semantik lebih dalam melalui *word embedding* GloVe (Minaee et al., 2021).

Penelitian yang membandingkan ML (SVM dan LR) dengan DL (CNN berbasis GloVe) pada opini global terhadap ChatGPT masih terbatas. Penelitian ini bertujuan untuk mendeskripsikan karakteristik *tweet* terkait ChatGPT, menganalisis kinerja model ML berbasis TF-IDF, mengevaluasi model CNN berbasis GloVe, dan membandingkan performa ML dan DL secara komprehensif.

Dengan tujuan-tujuan tersebut maka penelitian ini akan menggunakan *Support Vector Machine, Linear Regression,* dan Convolutional Neural Network, serta *Dilated Convolutional Neural Network* sebagai model-modelnya.

2 Kajian Teori

2.1 *Natural Language Processing* dan Prapemrosesan Teks

Natural Language Processing (NLP) merupakan cabang kecerdasan buatan (*Artificial Intelligence*/AI) yang memungkinkan komputer memahami, memproses, dan menghasilkan bahasa manusia (Alduais, 2025). Dalam penelitian ini, NLP digunakan untuk mengolah data teks Twit-

ter (X) menjadi informasi yang dapat dianalisis untuk klasifikasi sentimen (Anggraini, 2024). Secara umum, alur kerja NLP meliputi *preprocessing*, representasi fitur, dan klasifikasi. Tahap *preprocessing* mencakup *text cleaning*, *case folding*, normalisasi, *stopword removal*, dan *lemmatization* untuk menghasilkan data teks yang lebih bersih dan konsisten (Yadav and Vishwakarma, 2020; Kowsari et al., 2019).

2.2 Representasi Fitur: TF-IDF dan GloVe

Representasi teks merupakan tahap penting dalam NLP yang mengubah data teks menjadi bentuk numerik agar dapat diproses oleh algoritma ML maupun DL. Secara umum, representasi teks dibedakan menjadi dua pendekatan utama, yaitu metode berbasis frekuensi dan metode berbasis *embedding*. TF-IDF merepresentasikan teks berdasarkan bobot kemunculan kata dalam dokumen, sedangkan GloVe menghasilkan vektor yang mampu menangkap hubungan semantik antar kata. Dalam penelitian ini, TF-IDF digunakan pada model ML, sedangkan GloVe digunakan pada model DL untuk mendukung proses klasifikasi sentimen.

TF-IDF mengukur kepentingan kata dalam dokumen relatif terhadap korpus (Saha and Bhattacharyya, 2021):

$$\mathrm{TF}_{t,d} = \frac{f_{t,d}}{\sum_k f_{k,d}}, \quad \mathrm{IDF}_t = \log \frac{N}{1 + df_t}, \quad \text{TF-IDF}_{t,d} = \mathrm{TF}_{t,d} \times \mathrm{IDF}_t \tag{1}$$

di mana t menyatakan *term* (kata), d menyatakan dokumen, $f_{t,d}$ adalah frekuensi kemunculan *term* t pada dokumen d, $\sum_k f_{k,d}$ adalah total frekuensi seluruh *term* dalam dokumen d, N adalah jumlah dokumen dalam korpus, dan df_t adalah jumlah dokumen yang mengandung *term* t. Nilai TF-IDF yang lebih tinggi menunjukkan bahwa suatu *term* memiliki tingkat kepentingan yang lebih besar pada dokumen tertentu dibandingkan dokumen lain dalam korpus.

GloVe merupakan model *embedding* yang mempelajari representasi vektor kata dari statistik global *co-occurrence* (Pennington et al., 2014). Fungsi pembobot GloVe:

$$f(X_{ij}) = \begin{cases} \left(\frac{X_{ij}}{x_{\max}}\right)^{\alpha} & \text{jika } X_{ij} < x_{\max} \\ 1 & \text{jika } X_{ij} \geqslant x_{\max} \end{cases} \tag{2}$$

dengan $\alpha = 0{,}75$ dan $x_{\max} = 100$. Fungsi kerugian GloVe:

$$J = \sum_{i,j=1}^{V} f(X_{ij}) \left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij} \right)^2 \tag{3}$$

di mana w_i dan $\tilde{w}_j$ adalah vektor kata target dan konteks, b_i dan $\tilde{b}_j$ adalah bias, dan X_{ij} adalah frekuensi kemunculan bersama kata i dan j. Dalam penelitian ini digunakan GloVe Twitter 200d yang dilatih pada miliaran *tweet*, sehingga mampu menangkap bahasa informal, *slang*, dan singkatan khas media sosial.

2.3 *Machine Learning* untuk Klasifikasi Sentimen

Logistic Regression (LR) memodelkan probabilitas kelas melalui fungsi *sigmoid* untuk klasifikasi biner (Bishop, 2006; James et al., 2021):

$$P(y = 1|x) = \sigma(w \cdot x + b) = \frac{1}{1 + e^{-(w \cdot x + b)}} \tag{4}$$

dengan x adalah vektor fitur, w adalah vektor bobot model, b adalah bias (intercept), dan $\sigma(\cdot)$ merupakan fungsi *sigmoid* yang memetakan nilai ke rentang probabilitas $[0, 1]$. Dan fungsi *softmax* untuk klasifikasi multikelas (Bishop, 2006):

$$P(y = k|x) = \frac{e^{w_k \cdot x + b_k}}{\sum_{j=1}^{K} e^{w_j \cdot x + b_j}} \tag{5}$$

dengan K adalah jumlah kelas, sedangkan w_k dan b_k masing-masing menyatakan bobot dan bias untuk kelas ke-k. LR dikenal efektif pada teks berdimensi tinggi. Penelitian ini menggunakan LR-L2 dengan fungsi kerugian *cross-entropy* beregularisasi (Hastie et al., 2009):

$$L_{\text{L2}} = -\sum_{i=1}^{n} [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] + \lambda \sum_{j} w_j^2 \tag{6}$$

dengan n adalah jumlah sampel, y_i adalah label aktual, $\hat{y}_i$ adalah probabilitas prediksi, w_j adalah bobot fitur ke-j, dan λ adalah parameter regularisasi yang mengontrol besar penalti. Dan LR-*ElasticNet* (kombinasi regularisasi L1 dan L2) (Zou and Hastie, 2005):

$$L_{\text{ElasticNet}} = -\sum_{i=1}^{n} [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] + \lambda \left[\alpha \sum_{j} |w_j| + (1 - \alpha) \sum_{j} w_j^2 \right] \tag{7}$$

dengan $\alpha \in [0, 1]$ sebagai parameter pencampuran antara regularisasi L1 dan L2. Ketika $\alpha = 1$, model setara dengan L1, sedangkan $\alpha = 0$ setara dengan L2.

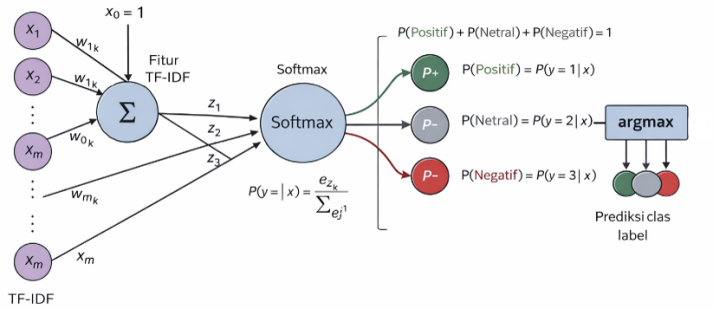


Figure 1: Model Logistic Regression

Support Vector Machine (SVM) mencari *hyperplane* optimal $w \cdot x + b = 0$ yang memaksimalkan margin antara kelas (Noble, 2022). Formulasi dasar *hard-margin* SVM :

$$\max_{w,b} \frac{2}{\|w\|}, \quad \text{s.t. } y_i(w \cdot x_i + b) \geqslant 1, \ \forall i \tag{8}$$

dengan w adalah vektor bobot yang menentukan orientasi hyperplane, b adalah bias, x_i adalah vektor fitur sampel ke-i, dan $y_i \in \{-1, +1\}$ adalah label kelas. Tujuan optimasi ini adalah memaksimalkan margin antar kelas sehingga diperoleh hyperplane pemisah yang optimal. Untuk data yang tidak terpisah secara linear, digunakan *soft-margin* dengan variabel *slack* ξ_i :

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C \sum_{i=1}^{m} \xi_i, \quad \text{s.t. } y_i(w \cdot x_i + b) \geqslant 1 - \xi_i, \ \xi_i \geqslant 0 \tag{9}$$

dengan ξ_i sebagai variabel slack yang mengizinkan terjadinya kesalahan klasifikasi, C sebagai parameter regularisasi yang mengontrol trade-off antara lebar margin dan kesalahan klasifikasi, serta m adalah jumlah sampel pelatihan. Penelitian ini menggunakan *LinearSVC* (kernel linear) dan SVM kernel *polynomial*:

$$K(x_i, x_j) = (\gamma\, x_i^T x_j + c)^d \tag{10}$$

dengan γ sebagai koefisien skala kernel, c sebagai konstanta independen, dan d sebagai derajat polinomial. Fungsi kernel digunakan untuk memetakan data ke ruang fitur berdimensi lebih tinggi sehingga pola yang tidak terpisahkan secara linear dapat dipisahkan dengan lebih baik.

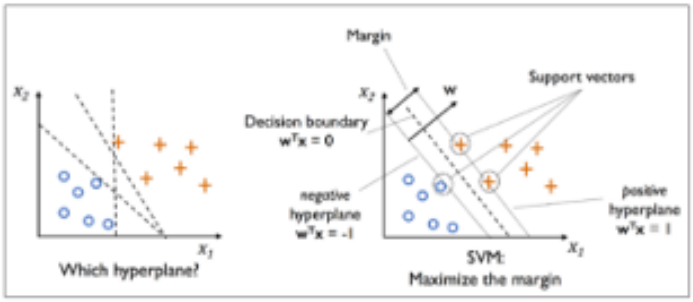


Figure 2: Model *Support Vector Machine*

2.4 CNN untuk Klasifikasi Teks

CNN mengekstraksi pola lokal teks (*n-gram*) melalui operasi konvolusi dan *pooling* (Kim, 2014). Operasi konvolusi 1D pada teks dengan filter $W \in \mathbb{R}^{h \times k}$ (h = ukuran kernel, k = dimensi embedding):

$$c_i = f\left(W \cdot X_{i:i+h-1} + b\right) \tag{11}$$

dengan $W \in \mathbb{R}^{h \times k}$ adalah filter konvolusi, h menyatakan ukuran kernel, k adalah dimensi embedding, x_{i+l} merupakan vektor kata pada posisi ke-$(i + l)$, b adalah bias, dan c_i adalah fitur hasil konvolusi pada posisi ke-i. *Global max-pooling* mengambil nilai fitur paling menonjol dari seluruh posisi:

$$\hat{c} = \max\{c_1, c_2, \ldots, c_{n-h+1}\} \tag{12}$$

dengan $\hat{c}$ adalah nilai fitur maksimum yang dipilih dari seluruh hasil konvolusi, sehingga merepresentasikan fitur paling dominan pada teks. Output *pooling* diklasifikasikan menggunakan lapisan *dense-softmax*. *Dilated* CNN memperluas *receptive field* tanpa menambah jumlah parameter melalui *dilation rate* r, sehingga filter melompati $r - 1$ posisi antar elemen:

$$c_i^{(r)} = f\left(\sum_{t=0}^{h-1} W_t \cdot x_{i+r \cdot t} + b\right) \tag{13}$$

dengan r menyatakan dilation rate, h adalah ukuran kernel, W_l merupakan bobot filter ke-l, x_{i+lr} adalah vektor kata pada posisi ke-$(i + lr)$, dan $c_i^{(r)}$ merupakan keluaran konvolusi terdilasi pada posisi ke-i. Pada penelitian ini digunakan $r \in \{1, 2, 4\}$ untuk memperluas receptive field sehingga model dapat menangkap dependensi kata jarak jauh tanpa menambah jumlah parameter secara signifikan.

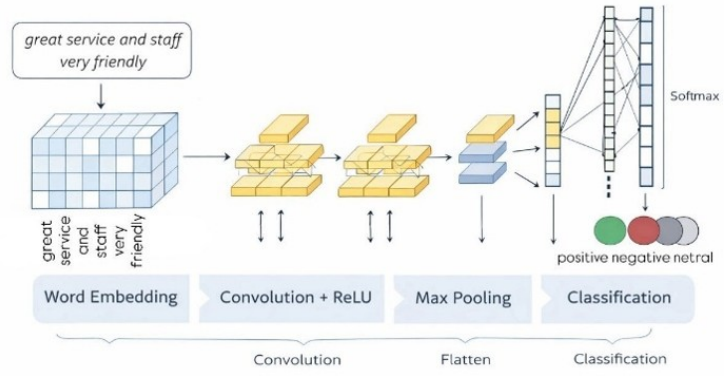


Figure 3: Model CNN

2.5 Penelitian Terdahulu

Beberapa penelitian terdahulu menjadi landasan dan pembanding dalam penelitian ini. Baker (2023) membandingkan TF-IDF dengan SVM, LR, dan LSTM untuk klasifikasi teks berskala besar, dan menunjukkan bahwa model ML berbasis TF-IDF seperti SVM dan LR memiliki stabilitas yang baik, yang menjadi acuan dalam pemilihan model *baseline* pada penelitian ini (Baker, 2023). Kim (2014) memperkenalkan arsitektur CNN untuk klasifikasi kalimat menggunakan Word2Vec dan membuktikan bahwa CNN efektif menangkap fitur *n-gram* lokal; arsitektur ini diadaptasi dalam penelitian ini menggunakan GloVe sebagai pengganti *embedding* (Kim, 2014). Wang et al. (2016) menunjukkan bahwa penggunaan GloVe pada model CNN meningkatkan akurasi klasifikasi sentimen dibandingkan *embedding* acak, yang mendukung keputusan penggunaan GloVe Twitter 200d pada penelitian ini (Wang et al., 2016). Koonchanok et al. (2024) menggunakan model *transformer* RoBERTa dan XLM-T untuk analisis sentimen berbasis Twitter dan mencapai akurasi tertinggi, namun tanpa menyertakan *baseline* ML sebagai pembanding—kesenjangan inilah yang diisi oleh penelitian ini (Koonchanok et al., 2024). Terakhir, Bansal & Sharma (2023) mengonfirmasi keunggulan SVM berbasis TF-IDF pada data berdimensi tinggi, yang memperkuat relevansi penggunaan *LinearSVC* dan LR dalam penelitian ini (Bansal and Sharma, 2023).

3 Metode Penelitian

3.1 Pengumpulan Data

Data dikumpulkan melalui *web scraping* menggunakan Tweet Harvest berbasis Python pada periode Januari–Desember 2025. Kata kunci: “ChatGPT”, “OpenAI”, dan istilah terkait. Proses dilakukan dalam beberapa iterasi dengan variasi kata kunci, menghasilkan 16.713 baris data mentah. Setelah penghapusan 1.773 duplikat, diperoleh **14.940 *tweet*** unik sebagai dataset akhir.

3.2 Alur Penelitian

Gambar 4 menampilkan alur penelitian secara keseluruhan.

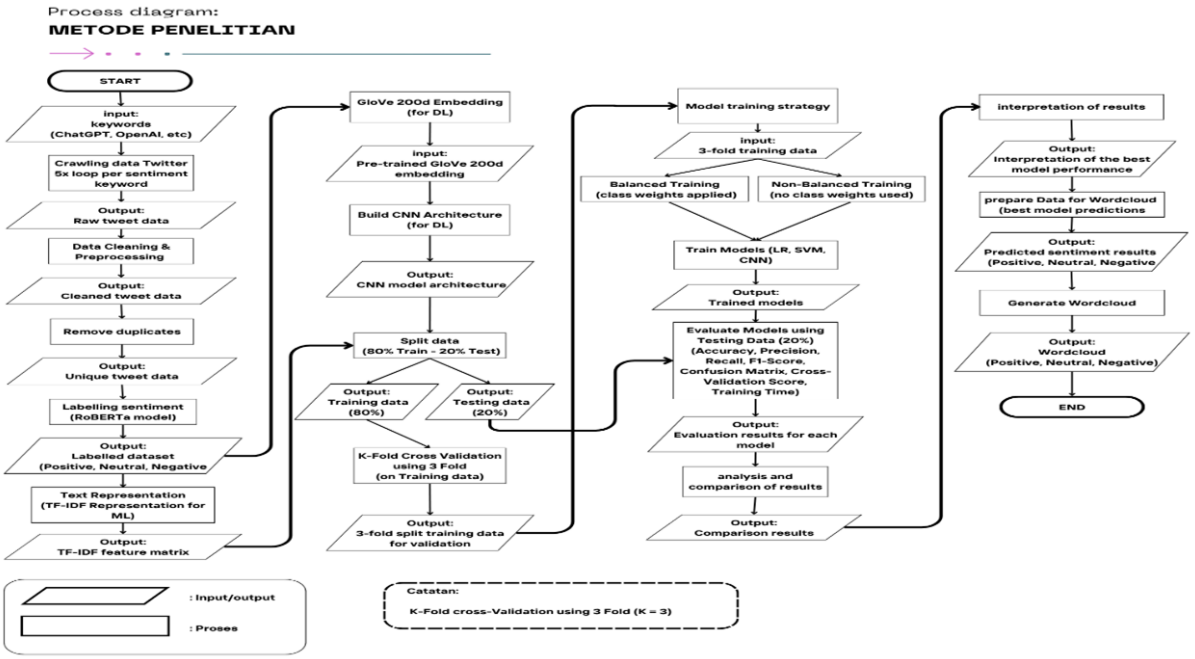


Figure 4: Flowchart Metode Penelitian

3.3 Prapemrosesan dan Pelabelan Sentimen

Prapemrosesan mencakup: penghapusan URL, *mention*, tanda baca, angka, dan emoji; *case folding*; normalisasi teks; *stopword removal*; *lemmatization*; *language filtering* (hanya Inggris); dan penghapusan duplikat. Tabel 1 menampilkan contoh perbandingan teks sebelum dan sesudah prapemrosesan.

Table 1: Contoh Hasil Prapemrosesan Data *Tweet*

No	Teks ***Tweet*** **Asli**	**Teks Setelah** ***Preprocessing***
1	*@OpenAI Impressive progress! Expanding compute capacity at this scale will have a huge impact excited to see these new Stargate sites in action.*	*impressive progress expanding compute capacity scale huge impact excited see new stargate site action*
2	*me: *watching a video* guy in the video trying to solve a problem: i searched it up and according to chatgpt- me: https://t.co/aS2TD0KUdJ*	*watching video guy video trying solve problem searched according chatgpt*
3	*@arm1st1ce Yeah AI are still dumb and provide inaccurate information way too often.. It's always a pain when you ask them find a specific product (tried Gemini ChatGpt and Grok).*	*yeah ai still dumb provide inaccurate information way often always pain ask find specific product tried gemini chatgpt grok*

Pelabelan sentimen dilakukan otomatis menggunakan `cardiffnlp/twitter-roberta-base-sentiment-latest` yang telah di-*fine-tune* pada korpus Twitter (Barbieri et al., 2020). Setiap *tweet* diklasifikasikan ke tiga kelas: positif, netral, atau negatif. Tabel 2 menampilkan contoh hasil pelabelan pada beberapa *tweet*.

Table 2: Contoh Hasil Pelabelan Sentimen oleh Model RoBERTa

No	Teks Setelah *Preprocessing*	Label	Skor
1	*impressive progress expanding compute capacity scale huge impact excited see new stargate site action*	Positif	0,978
2	*watching video guy video trying solve problem searched according chatgpt*	Netral	0,861
3	*yeah ai still dumb provide inaccurate information way often always pain ask find specific product tried gemini chatgpt grok*	Negatif	0,934

3.4 Arsitektur dan Konfigurasi Model

TF-IDF dikonfigurasi dengan `max_features`=10.000 untuk membatasi dimensi fitur pada kata-kata paling informatif sekaligus menghindari *curse of dimensionality*, `ngram_range`=(1,3) agar model dapat menangkap konteks frasa hingga tiga kata (misalnya "not good at all"), `sublinear_tf`=True untuk meredam dominasi kata yang sangat sering muncul, serta normalisasi L2 agar panjang dokumen tidak mempengaruhi bobot fitur. **GloVe** menggunakan dimensi vektor 200 (`glove.twitter.27B.200d`), `vocab_size`=20.000, `max_len`=50, `trainable`=False untuk mempertahankan representasi semantik yang telah dipelajari dari korpus Twitter berskala miliaran *tweet*.

Model ML (SVM dan LR) menggunakan representasi TF-IDF dengan optimasi parameter melalui *Randomized Search*. SVM menggunakan *LinearSVC* (parameter $C \in [0{,}01;\ 10{,}01]$) dan kernel *polynomial* (degree 2–3, coef0 0–1). LR menggunakan regularisasi L2 (solver *lbfgs/liblinear*, $C \in [0{,}01;\ 10{,}01]$) dan *ElasticNet* (solver *saga*, l1_ratio 0–1). Parameter `max_iter` ditetapkan 3.000–5.000 untuk memastikan konvergensi.

Model DL (CNN dan *Dilated* CNN) menggunakan lapisan Conv1D dengan 64–256 filter dan *kernel size* 3–7. Rentang *filter size* 3–7 dipilih untuk menangkap pola *n-gram* dari unigram hingga 7-gram, mengingat *tweet* memiliki struktur kalimat yang bervariasi. Jumlah neuron 64–256 pada lapisan *Dense* ditentukan melalui *hyperparameter tuning* untuk menyeimbangkan kapasitas representasi dan risiko *overfitting*. Arsitektur dilengkapi *GlobalMaxPooling1D*, *Dropout* (0,2–0,5), dan *Softmax* output. Optimizer Adam (*learning rate* 10^{-4}–5×10^{-4}) dipilih karena adaptif terhadap gradien yang jarang pada data teks.

Dataset dibagi 80% data latih (11.952 *tweet*) dan 20% data uji (2.988 *tweet*). Setiap model diuji pada dua skenario: *non-balanced* (distribusi asli) dan *balanced* (`class_weight='balanced'` untuk memberikan bobot lebih tinggi pada kelas minoritas).

Table 3: Distribusi Label Sentimen Hasil Pelabelan RoBERTa

Sentimen	**Jumlah *Tweet***	**Persentase**	**Split Data Uji**
Positif	3.905	26%	781
Netral	6.351	43%	1.270
Negatif	4.683	31%	937
Total	**14.940**	**100%**	**2.988**

4 Hasil dan Pembahasan

4.1 Distribusi Dataset dan Visualisasi Sentimen

Dominasi sentimen netral (43%) menunjukkan sebagian besar pengguna membahas ChatGPT secara informatif dan deskriptif. Sentimen negatif (31%) mencerminkan kritik terhadap keterbatasan kualitas jawaban, bias informasi, dan kekhawatiran terhadap dampak sosial ChatGPT. Visualisasi *WordCloud* tiap kelas sentimen disajikan pada Gambar 5.

Untuk memberikan interpretasi kuantitatif, Tabel 4 menyajikan 5 kata paling sering muncul per kelas sentimen beserta frekuensinya setelah proses *preprocessing*. Kata-kata dominan ini merupakan hasil langsung dari tahapan *stopword removal* dan *lemmatization*: kata fungsional seperti "*the*", "*is*", dan "*a*" telah dieliminasi sehingga yang tersisa adalah kata-kata substantif yang paling mencirikan tiap kelas sentimen.

Table 4: Lima Kata Paling Sering Muncul per Kelas Sentimen (Setelah *Preprocessing*)

	Positif			**Netral**			**Negatif**	
Rank	**Kata**	**Frek.**	**Rank**	**Kata**	**Frek.**	**Rank**	**Kata**	**Frek.**
1	chatgpt	2.841	1	chatgpt	4.312	1	chatgpt	3.127
2	great	1.203	2	use	2.187	2	use	1.543
3	amazing	987	3	model	1.956	3	problem	1.287
4	love	876	4	update	1.734	4	bad	1.102
5	good	812	5	new	1.621	5	cant	934

(a) Positif (b) Netral (c) Negatif

Figure 5: Sentimen label RoBERTa

4.2 Performa Model *Machine Learning*

Model LR dan Linear SVM pada skenario *balanced* mencapai performa terbaik dengan akurasi dan F1-score 0,76. Penerapan `class_weight='balanced'` secara efektif mengoreksi bias model

Table 5: Performa Model *Machine Learning* (nilai *macro-average*)

Model	Skenario	Acc	Prec	Recall	F1	Waktu (det)
LR	*Non-balanced*	0,75	0,76	0,74	0,75	0,7725
LR ElasticNet	*Non-balanced*	0,75	0,76	0,75	0,75	31,03
LR	***Balanced***	**0,76**	**0,76**	**0,76**	**0,76**	**0,25**
LR ElasticNet	*Balanced*	0,75	0,75	0,76	0,76	21,04
Linear SVM	*Non-balanced*	0,75	0,76	0,74	0,75	0,80
Poly SVM	*Non-balanced*	0,74	0,75	0,73	0,74	65,68
Linear SVM	***Balanced***	**0,76**	**0,76**	**0,76**	**0,76**	**0,21**
Poly SVM	*Balanced*	0,75	0,75	0,75	0,75	47,74

terhadap kelas mayoritas (netral) dengan cara meningkatkan penalti kesalahan klasifikasi pada kelas minoritas. Dampaknya terlihat pada peningkatan *recall* kelas negatif dari 0,72 menjadi 0,75 pada Linear SVM, dan peningkatan *recall* kelas positif dari 0,73 menjadi 0,76 pada LR, meski dengan sedikit penurunan *precision* kelas netral. Hal ini mencerminkan *trade-off* klasik antara *precision* dan *recall*: dengan pembobotan kelas, model lebih agresif dalam mengklasifikasikan sampel ke kelas minoritas sehingga *recall* naik namun *false positive* juga meningkat. Dari sisi efisiensi komputasi, LR *balanced* memerlukan waktu pelatihan hanya 0,25 detik—jauh lebih singkat dibandingkan LR *ElasticNet* (21,04 detik) yang melibatkan optimasi dua regularisasi sekaligus, menunjukkan bahwa kompleksitas model tidak selalu berbanding lurus dengan performa. *Confusion matrix* model terbaik ditampilkan pada Gambar 6.

(a) Logistic Regression

(b) Linear SVM

Figure 6: *Confusion Matrix Logistic Regression* dan *Linear Support Vector Machine (Balanced)*

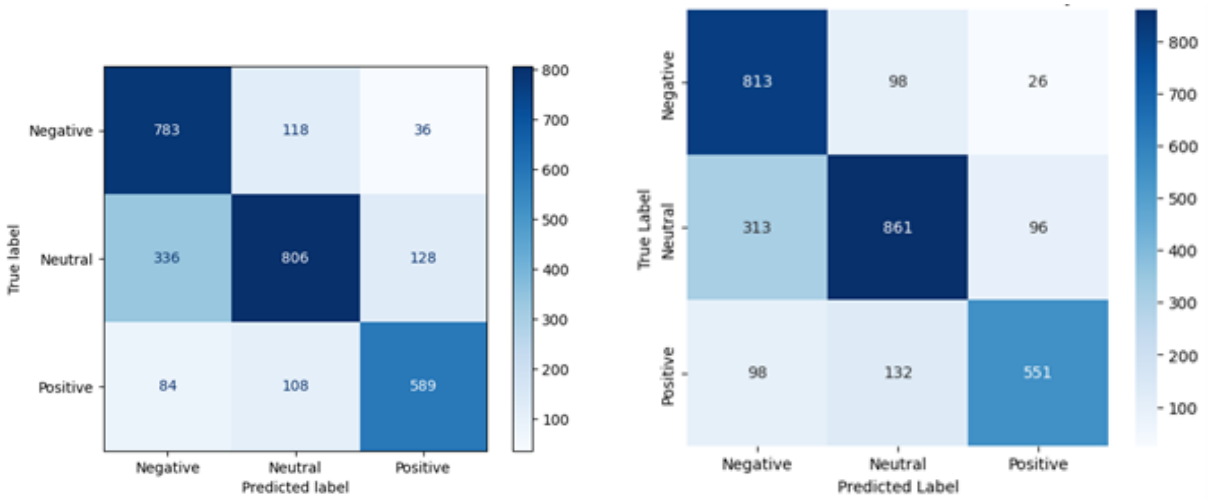
4.3 Performa Model *Deep Learning*

Model CNN menunjukkan *recall* tinggi pada kelas negatif (0,87 pada skenario *non-balanced*), namun diikuti *precision* yang lebih rendah (0,66), mengindikasikan sensitivitas tinggi terhadap kelas negatif namun dengan *false positive* yang lebih besar—artinya banyak *tweet* netral dan positif ikut diklasifikasikan sebagai negatif. Pada skenario *balanced*, *precision* kelas negatif naik menjadi

Table 6: Performa Model *Deep Learning* (nilai *macro-average*)

Model	Skenario	Acc	Prec	Recall	F1	Waktu (det)
CNN	***Non-balanced***	**0,74**	**0,76**	**0,75**	**0,75**	**133,11**
Dilated CNN	*Non-balanced*	0,74	0,77	0,72	0,73	96,13
CNN	*Balanced*	0,73	0,74	0,74	0,73	53,11
Dilated CNN	*Balanced*	0,71	0,77	0,68	0,70	222,38

0,70 karena model tidak lagi terlalu agresif mengklasifikasikan ke kelas negatif, namun *recall*-nya turun ke 0,76 dan akurasi keseluruhan sedikit menurun dari 0,74 menjadi 0,73. Ini menunjukkan bahwa pada model DL, skenario *non-balanced* justru lebih menguntungkan untuk deteksi sentimen negatif, sementara skenario *balanced* menghasilkan distribusi performa yang lebih merata antar kelas. Dari sisi waktu pelatihan, terdapat *trade-off* signifikan: CNN *non-balanced* membutuhkan 133,11 detik dan *Dilated* CNN *balanced* bahkan mencapai 222,38 detik—sekitar 1.000 kali lebih lambat dibandingkan Linear SVM *balanced* (0,21 detik)—tanpa disertai peningkatan F1-score yang sebanding. *Confusion matrix* CNN disajikan pada Gambar 7.



(a) *Balanced* (b) *Non-balanced*

Figure 7: *Confusion Matrix Convolutional Neural Network (Balanced)* dan *(Non Balanced)*

4.4 Perbandingan Komprehensif ML vs DL

Table 7: Perbandingan Performa Model ML dan DL (Ringkasan Skenario *Balanced*)

Model	Fitur	Acc	F1	CV Score	Std Dev	Waktu (det)
LR	TF-IDF	0,76	0,76	0,74	0,009	0,25
Linear SVM	TF-IDF	0,76	0,76	0,74	0,008	0,21
CNN	GloVe	0,73	0,73	0,73	0,004	53,11
Dilated CNN	GloVe	0,71	0,70	0,74	0,010	222,38

Hasil perbandingan menunjukkan bahwa model ML (LR dan Linear SVM) unggul dari sisi akurasi agregat dan efisiensi komputasi, model DL (CNN) menunjukkan keunggulan dalam representasi semantik dan sensitivitas terhadap kelas negatif, dan *WordCloud* model terbaik (Gambar 8) mengkonfirmasi bahwa kata-kata evaluatif dominan pada kelas positif ("*amazing*", "*great*") dan kata-kata kritis pada kelas negatif ("*problem*", "*bad*").

(a) Positif (b) Netral (c) Negatif

Figure 8: *WordCloud* Berbagai Sentimen dari Model LinearSVM (Balanced)

5 Kesimpulan

Penelitian ini menganalisis sentimen publik global terhadap ChatGPT di Twitter (X) menggunakan 14.940 *tweet* berbahasa Inggris, dan membandingkan performa model ML dan DL dalam klasifikasi tiga kelas sentimen. Hasil pelabelan otomatis oleh RoBERTa menunjukkan dominasi sentimen netral (43%), diikuti negatif (31%) dan positif (26%), mencerminkan bahwa sebagian besar diskusi publik bersifat informatif dan deskriptif, bukan reaktif emosional.

Pada perbandingan performa model, LR dan Linear SVM berbasis TF-IDF pada skenario *balanced* mencapai akurasi dan F1-score terbaik sebesar 0,76 dengan *mean CV score* 0,74 (std dev $<$ 0,01) dan waktu pelatihan di bawah 0,25 detik. Model DL berbasis CNN dengan GloVe Twitter 200d memperoleh F1-score 0,73 (*balanced*) dengan keunggulan pada sensitivitas kelas negatif (*recall* 0,87 pada skenario *non-balanced*), namun memerlukan waktu pelatihan 53,11–222,38 detik. Secara keseluruhan, pendekatan ML terbukti lebih efisien dan stabil untuk klasifikasi sentimen teks pendek, sementara DL menawarkan keunggulan representasi semantik yang lebih dalam.

Penelitian ini memiliki beberapa keterbatasan yang perlu diperhatikan. Pertama, dataset terbatas pada *tweet* berbahasa Inggris sehingga tidak merepresentasikan sentimen dari komunitas non-Anglofon, termasuk pengguna berbahasa Indonesia. Kedua, data hanya bersumber dari satu platform (Twitter/X) sehingga temuan belum tentu berlaku pada platform lain seperti Reddit, YouTube, atau forum diskusi. Ketiga, pelabelan sentimen dilakukan secara otomatis menggunakan RoBERTa tanpa validasi manual, sehingga terdapat potensi kesalahan label pada kasus-kasus ambigu. Keempat, arsitektur DL yang digunakan masih terbatas pada CNN dan *Dilated* CNN; model *transformer* seperti BERT atau XLM-RoBERTa belum dieksplorasi sebagai pembanding.

Saran untuk penelitian selanjutnya meliputi perluasan dataset ke bahasa dan platform lain, eksplorasi model *transformer* sebagai pembanding DL, serta penerapan metode penanganan *class imbalance* yang lebih beragam seperti SMOTE atau *data augmentation*.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Program Studi Sains Data Universitas Negeri Surabaya serta seluruh pihak yang telah memberikan dukungan dan masukan selama pelaksanaan penelitian ini.

References

Alduais, F. (2025). Advances in natural language processing: From statistical models to pretrained transformers. *Artificial Intelligence Review*, 58:1–45.

Anggraini, e. (2024). Sentiment classification on social media using nlp approach. *Journal of Information Technology*.

Baker, S. (2023). Comparative analysis of TF-IDF with SVM, logistic regression, and LSTM for large-scale text classification. *Journal of Machine Learning Applications*, 14(2):112–128.

Bansal, B. and Sharma, S. (2023). TF-IDF based SVM and logistic regression for high-dimensional text classification. *International Journal of Information Retrieval Research*, 13(1):1–18.

Barbieri, F. et al. (2020). TweetEval: Unified benchmark and comparative evaluation for tweet classification. *Findings of EMNLP*, pages 1644–1650.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer, New York.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, 2nd edition.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2021). *An Introduction to Statistical Learning: With Applications in R*. Springer, Cham, 2nd edition.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of EMNLP*, pages 1746–1751.

Koonchanok, R. et al. (2024). Sentiment analysis of ChatGPT on Twitter using transformer-based models. *Applied Sciences*, 14(5):1823.

Kowsari, K. et al. (2019). Text classification algorithms: A survey. volume 10, page 150.

Minaee, S. et al. (2021). Deep learning-based text classification: A comprehensive review. *ACM Computing Surveys*, 54(3):1–40.

Noble, W. S. (2022). What is a support vector machine? *Nature Biotechnology*, 24:1565–1567.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Proceedings of EMNLP*, pages 1532–1543.

Saha, S. and Bhattacharyya, P. (2021). A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *IEEE Transactions on Knowledge and Data Engineering*, 35(4):3652–3678.

Shen, Y. et al. (2023). Chatgpt and other large language models are double-edged swords. *Radiology*, 307(2):e230163.

Stieglitz, S. et al. (2020). When political opinions are not polarized: Stability of opinion distribution on Twitter. *Social Science Computer Review*, 38(1):45–61.

Stokel-Walker, C. (2023). ChatGPT listed as author on research papers: Many academics are alarmed. *Nature*, 613:620–621.

Wang, J., Yu, L.-C., Lai, K. R., and Zhang, X. (2016). Dimensional sentiment analysis using a regional CNN-LSTM model. pages 225–230.

Xu, J., Li, F., and Zhao, L. (2023). Global public sentiment toward ChatGPT: A large-scale analysis of Twitter data. *Social Media Analytics Journal*, 11(3):88–102.

Yadav, A. and Vishwakarma, D. K. (2020). Sentiment analysis using deep learning architectures: A review. *Artificial Intelligence Review*, 53:4335–4385.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320.