JOURNAL OF
SOFTWARE ENGINEERING AND
COMPUTATIONAL INTELLIGENCE (JSECI)
INFORMATICS ENGINEERING, FACULTY OF COMPUTER SCIENCE
UNIVERSITAS INDO GLOBAL MANDIRI

Implementasi *Deep Learning* Untuk Analisis Sentimen Pada Data X Dalam Prediksi Tren *Korean Style*

Siska Lestari[1], Hermanto[2], Dian Hafidh Zulfikar[3*]
[1,2,3]Sistem Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Raden Intan Lampung

[1]siskalestaryy@gmail.com, [2]hermanto@radenintan.ac.id, [3*]Dianhafidhzulfikar_uin@radenintan.ac.id,

ABSTRACT

This study implements a deep learning model based on Convolutional Neural Network (CNN) for sentiment analysis of Indonesian-language tweets related to the Korean Style trend. A dataset of 7,187 tweets was collected via web crawling using keywords such as "Korean Style", "K-pop", and "Korean fashion". The preprocessing pipeline included case folding, removal of special characters and emojis, stopword elimination, tokenization, and lemmatization. Word2Vec and FastText embeddings were employed for text representation. The CNN model classified tweets into three sentiment categories: positive, negative, and neutral. Evaluation metrics included accuracy, precision, recall, F1-score, and confusion matrix. Results showed 71.26% validation accuracy with the highest F1-score of 0.81 for the neutral class, while negative sentiment classification remained weak due to class imbalance. Word2Vec outperformed FastText in stability. This research contributes to sentiment analysis in Indonesian social media using deep learning and provides insights into public opinion on Korean cultural trends.

Keywords*: Sentiment Analysis, Convolutional Neural Network (CNN), Korean Style*

ABSTRAK

Penelitian ini bertujuan mengimplementasikan *deep learning* berbasis *Convolutional Neural Network* (CNN) untuk analisis sentimen pada data X berbahasa Indonesia terkait tren *Korean Style*. Data diperoleh sebanyak 7.187 tweet melalui *web crawling* dengan kata kunci "Korean Style", "K-pop", dan "Korean fashion". Tahapan pra-pemrosesan meliputi *case folding*, penghapusan karakter khusus dan emoji, *stopword removal*, tokenisasi, dan *lemmatization*. Data direpresentasikan menggunakan *word embedding* Word2Vec dan FastText. Model CNN kemudian dilatih untuk mengklasifikasikan sentimen menjadi tiga kelas: positif, negatif, dan netral. Evaluasi menggunakan akurasi, precision, *recall*, F1-score, serta *confusion matrix*. Hasil menunjukkan akurasi validasi sebesar 71,26%, dengan F1-score tertinggi 0,81 pada kelas netral. Namun, performa kelas negatif rendah akibat *class imbalance*. Word2Vec memberikan hasil lebih stabil dibanding FastText. Penelitian ini berkontribusi pada pengembangan analisis sentimen berbasis *deep learning* untuk bahasa Indonesia, serta memberi wawasan mengenai opini publik terhadap tren budaya Korea.

Kata kunci: : Analisis Sentiment, *Convolutional Neural Network* (CNN), *Korean Style*

Vol. 3 No. 2
Submitted October 27, 2025
Revised November 05, 2025.
Accepted November 29,2025

1. PENDAHULUAN

Perkembangan teknologi digital telah mendorong transformasi besar dalam berbagai aspek kehidupan, termasuk komunikasi, bisnis, dan budaya. Media sosial menjadi produk utama era digital yang berfungsi sebagai ruang interaksi Masyarakat untuk berbagi informasi, mengekspresikan opini, sekaligus mempengaruhi tren global. Platform seperti Facebook, X, Instagram dan tiktok kini tidak hanya digunakan sebagai sarana komunikasi personal, melaikan juga sebagai medium yang mampu merepresentasikan opini publik *secara real time*. Hasil survei Badan Pusat Statistic menunjukan bahwa lebih dari 70% Masyarakat Indonesia menggunakan media sosial sebagai sumber utama informasi,[1] yang menegaskan perannya dalam membentuk presepsi publik.

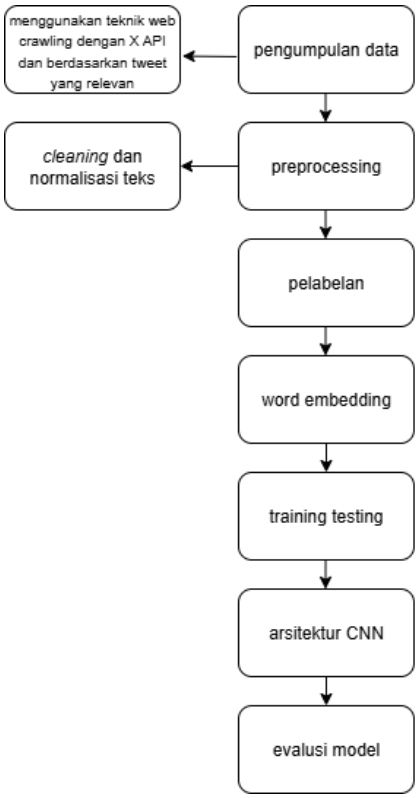
Salah satu fenomena budaya yang mendapat sorotan luas di media sosial Adalah *Korean Wave* atau *Hallyu*, yang mencakup music, drama, fashion, hingga gaya hidup korea Selatan.[2] Tren ini memunculkan istilah *Korean style* yang sangat populer dikalangan remaja dan dewasa muda di Indonesia. Adopsi gaya busana, music K-pop, hingga tren kosmestik korea menjadi indikasi kuat pengaruh budaya korea terhadap gaya hidup Masyarakat Indonesia. Dengan meningkatnya volume percakapan tentang *korean style* di media sosial, analisis terhadap opini publik menjadi penting unttuk memahami arah perkembangan tren ini, baik dari dari perspektif akademik, bisnis, maupun sos ial.

X atau yang dulunya dikenal dengan twitter dipilih sebagai sumber data karena memiliki karakteristik unggul, yaitu kemampuannya menyajikan informasi singkat, padat, dan *real time*, serta delengkapi fitur hashtag (#) yang memudahkan pengelompokan topik tertentu. Dengan demikian, platform ini menjadi representasi yang relevan untuk menggali opini Masyarakat tentang fenomena budaya.[3] Namun, analisis data dari X memiliki tantangan tersendiri, antara lain form teks yang pendek, bahasa yang informal, penggunakan emoji dan slang, serta dinamika percakapan yang cepat berubah. Tantangan-tantangan ini menuntut penggunaan metode analisis berbasis kecerdasan buatan yang mampu menangkap kompleksitas data teks.

Analisis sentiment merupakan salah satu pendekatan dalam bidang *Natural Language Processing* (NLP) yang bertujuan mengidentifikasi sikap, opini atau emosi public dari data teks. Metode trdisional berbasis *Machine Learning* seperti Naïve Bayes, *Support Vector Machine* (SVM), dan *Maximum Etropy* telah lama digunakan dalam penelitian analisis sentiment.[4] Namun, metode tersebut memiliki keterbatasan dalam memahami pola kompleks dan *non linear* pada teks pendek yang cenderung ambigu. Oleh karena itu, pendekatan berbasis *Deep Learning* semakin banyak diadopsi karena kemampuannya menangkap representasi fitur yang lebih kaya dan kompleks.[5]

2. METODE PENELITIAN

Penelitian ini merupakan penelitian eksperimen yang bertujuan membangun dan mengevaluasi model *Deep Learning convolutional neural network* (CNN) untuk klasifikasi sentimen berbahasa indonesia terkait tren *korean style.*[6] Tahapan metodelogo yang digunakan dapat dijelaskan sebagai berikut:



Gambar 1. Alur Penelitian

2.1 Pengumpulan Data

Proses pengumpulan data dilakukan dengan teknik *web crawling* menggunakan *Application Programming Interface* (API) serta pemrograman *Python* untuk mengekstraksi *tweet* yang relevan. Kata kunci pencarian yang digunakan meliputi istilah "*Korean Style*", "*K-pop*", dan "*Korean fashion*" yang dipilih karena dianggap mewakili fenomena tren budaya Korea di kalangan pengguna X.[7] Setelah proses pengumpulan, dilakukan penyaringan agar hanya tweet berbahasa Indonesia yang digunakan sehingga sesuai dengan konteks penelitian. Dari hasil pengumpulan tersebut, diperoleh total 7.187 tweet, kemudian dilakukan pembersihan tweet duplikat, dan spam hingga data yang di peroleh sebesar 5.620 tweet bersih, data inilah yang kemudian dijadikan sebagai dataset utama untuk tahap pra-pemrosesan dan pelatihan model CNN.

2.2 Pra-Pemrosesan Data

Setiap tweet yang diperoleh kemudian diproses melalui serangkaian tahapan pra-pemrosesan teks. Tahap pertama adalah *case folding*, yaitu mengubah seluruh huruf dalam tweet menjadi huruf kecil agar konsistensi data terjaga. Selanjutnya dilakukan pembersihan karakter khusus, seperti URL, mention, hashtag, angka, tanda baca, maupun emoji yang dianggap tidak relevan terhadap analisis sentimen. Setelah itu, dilakukan *stopword removal* untuk menghapus kata-kata umum yang tidak memiliki kontribusi terhadap makna sentimen.[8] Proses berikutnya adalah tokenisasi, yaitu pemecahan teks menjadi unit kata, kemudian dilanjutkan dengan lemmatization untuk mengembalikan setiap kata ke bentuk dasarnya. Sebagai langkah akhir, dilakukan padding sehingga setiap tweet memiliki panjang yang seragam, yaitu 100 token, agar dapat diproses secara optimal oleh model CNN.[9] tahap ini bertujuan untuk memproses data mentah dari hasil *crawling* agar dapat digunakan ditahap selanjutnya. Adapun contoh hasil *cleaning* data ditunjukan pada Tabel 1 berikut ini.

Tabel 1. Hasil Cleaning Crawling Data

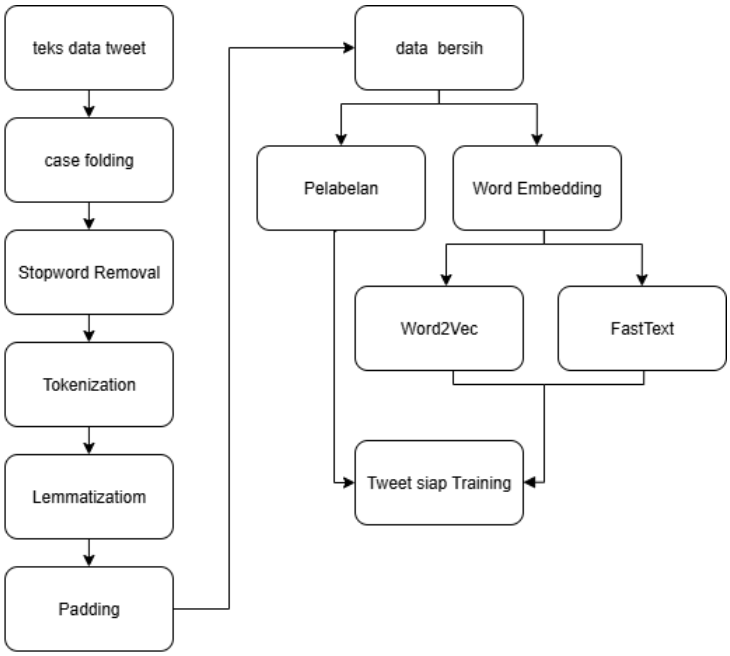
Aktivasi/ kondisi	Hasil
Tweet Awal	"@ootd_id Temukan Bayi - 3 Tahun - Dress Anak Yin Hua Korea Style | Dress Premium Kualitas Import Baju Anak Perempuan Gaya Korean Dres Lengan Pendek Dress Kensi Kiosbalitaaprillia dengan potongan 55%! Hanya Rp54.500. ✨ Dapatkan segera di Shopee! https://t.co/BcPGH365q1 #ShopeeID https://t.co/XN3EY2yxq"
CaseFolding	@ootd_id temukan bayi - 3 tahun - dress anak yin hua korea style | dress premium kualitas import baju anak perempuan gaya korean dres lengan pendek dress kensi kiosbalitaaprillia dengan potongan 55%! hanya rp54.500. ✨dapatkan segera di shopee! https://t.co/bcpgh365q1 #shopeeid https://t.co/xn3ey2yxqf
Menghapus Mention	<mention>ootd_id temukan bayi - 3 tahun - dress anak yin hua korea style | dress premium kualitas import baju anak perempuan gaya korean dres lengan pendek dress kensi kiosbalitaaprillia dengan potongan 55%! hanya rp54.500. ✨dapatkan segera di shopee! https://t.co/bcpgh365q1 #shopeeid https://t.co/xn3ey2yxqf
Menghapus Hastag (#)	ootd_id temukan bayi - 3 tahun - dress anak yin hua korea style | dress premium kualitas import baju anak perempuan gaya korean dres lengan pendek dress kensi kiosbalitaaprillia dengan potongan 55%! hanya rp54.500. ✨ dapatkan segera di shopee! https://t.co/bcpgh365q1 <hastag>shopeeid https://t.co/xn3ey2yxqf
Menghapus URL	<mention>ootd_id temukan bayi - 3 tahun - dress anak yin hua korea style | dress premium kualitas import baju anak perempuan gaya korean dres lengan pendek dress kensi kiosbalitaaprillia dengan potongan 55%! hanya rp54.500. ✨ dapatkan segera di shopee! <URL> <hastag>shopeeid <URL>
Menghapus Angka	<mention>ootd_id temukan bayi - <angka>tahun - dress anak yin hua korea style | dress premium kualitas import baju anak perempuan gaya korean dres lengan pendek dress kensi kiosbalitaaprillia dengan potongan <angka>%! hanya rp✨<angka>dapatkan segera di shopee! <URL> <hastag>shopeeid <URL>
Menghapus Emoticon	<mention>ootd_id temukan bayi - 3 tahun - dress anak yin hua korea style | dress premium kualitas import baju anak perempuan gaya korean dres lengan pendek dress kensi kiosbalitaaprillia dengan potongan <angka>%! hanya rp<emoticon><angka>dapatkan segera di shopee! <URL> <hastag>shopeeid <URL>
Menghapus Simbol	temukan bayi tahun dress anak yin hua korea style dress premium kualitas import baju anak perempuan gaya korean dres lengan pendek dress kensi kiosbalitaaprillia dengan potongan hanya rp dapatkan segera di shopee
Stopword Removal	temukan bayi tahun dress anak yin hua korea style dress premium kualitas import baju anak perempuan gaya korean dres lengan pendek dress kensi kiosbalitaaprillia potongan rp dapatkan segera shopee
Tokenisasi	'temukan', 'bayi', 'tahun', 'dress', 'anak', 'yin', 'hua', 'korea', 'style', 'dress', 'premium', 'kualitas', 'import', 'baju', 'anak', 'perempuan', 'gaya', 'korean', 'dres', 'lengan', 'pendek', 'dress', 'kensi', 'kiosbalitaaprillia', 'potongan', 'rp', 'dapatkan', 'segera', 'shopee'
Lemmatization	'temu', 'bayi', 'tahun', 'dress', 'anak', 'yin', 'hua', 'korea', 'style', 'dress', 'premium', 'kualitas', 'impor', 'baju', 'anak', 'perempuan', 'gaya', 'korean', 'dres', 'lengan', 'pendek', 'dress', 'kensi', 'kiosbalitaaprillia', 'potong', 'rp', 'dapat', 'segera', 'shopee'

Setelah tokenisasi, setiap tweet diubah menjadi urutan angka yang mewakili indeks kata dalam kamus *tokenizer*. Karena panjang tweet bervariasi, dilakukan padding dengan menambahkan nol agar seluruh input memiliki panjang seragam, sehingga dapat diproses oleh model CNN yang memerlukan dimensi tetap. Contoh *padding* di tunjukan pada Tabel 2 berikut ini.

Tabel 2. Hasil Proses Padding

Proses	Hasil	Keterangan
Sebelum Padding (contoh sequence):	[2, 5, 13, 19, 15, 3, 21, 11, 7]	Panjang: 9 kata
Setelah Padding (Contoh Padded)	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ..., 2, 5, 13, 19, 15, 3, 21, 11, 7]	Panjang: 100 token (misalnya, panjang maksimum ditentukan 100)

Untuk memudahkan pemahaman alur kerja penelitian, Gambar 2 menyajikan diagram proses pra-pemrosesan, pelabelan, dan *word embedding* yang dilakukan sebelum tahap pelatihan model sebagai berikut.



Gambar 2. Alur pra-pemrosesan

2.3 Pelabelan Data

Tahap pelabelan dilakukan untuk memberikan kategori sentimen pada setiap tweet yang telah melalui proses penyaringan dan pra-pemrosesan, dengan klasifikasi ke dalam tiga kelas utama yaitu positif, negatif, dan netral.[10] Proses pelabelan dilaksanakan secara semi-otomatis, dimulai dengan pelabelan awal berbasis leksikon menggunakan metode VADER (*Valence Aware Dictionary and Sentiment Reasoner*). Metode ini memberikan skor polaritas (*compound score*) pada setiap tweet berdasarkan daftar kata yang memiliki bobot sentimen tertentu. Tweet dengan *compound score* $\geq +0{,}05$ diberi label positif, tweet dengan *compound score* $\leq -0{,}05$ diberi label negatif, dan tweet dengan skor di antara kedua ambang batas tersebut diberi label netral.[11] Contoh hasil pelabelan di tunjukan pada Tabel 3 berikut.

Tabel 3. Contoh pelabelan

Contoh tweet setelah pra-pemrosesan	Compound Score	Label
"aku suka banget gaya fashion korea"	0.71	Positif
"tidak tertarik sama sekali dengan korean style"	-0.62	Negative
"trend baju korea lagi banyak dibicarakan di twitter"	0.03	Netral

Hasil pelabelan otomatis ini kemudian divalidasi secara manual oleh peneliti untuk memastikan ketepatan klasifikasi, mengingat karakteristik bahasa media sosial yang kerap mengandung slang, campuran bahasa, atau ironi. Validasi dilakukan secara menyeluruh dengan melibatkan minimal dua annotator independen, dan konsistensi hasil anotasi diukur menggunakan *Cohen's Kappa* untuk menilai tingkat kesepakatan antar-annotator.[12] Melalui kombinasi metode otomatis dan validasi manual ini, dataset yang dihasilkan tidak hanya lebih efisien dalam proses pelabelan, tetapi juga memiliki kualitas dan reliabilitas yang tinggi sebagai dasar pelatihan model *Convolutional Neural Network* (CNN) untuk analisis sentimen.

2.4 Representasi Teks (*Word Embedding*)

Setelah proses pra-pemrosesan, data teks hasil tokenisasi harus diubah menjadi representasi numerik agar dapat diproses oleh model *deep learning*. Pada penelitian ini digunakan dua metode *word embedding* yaitu Word2Vec dan FastText sebagai pembanding.

1. Word2Vec

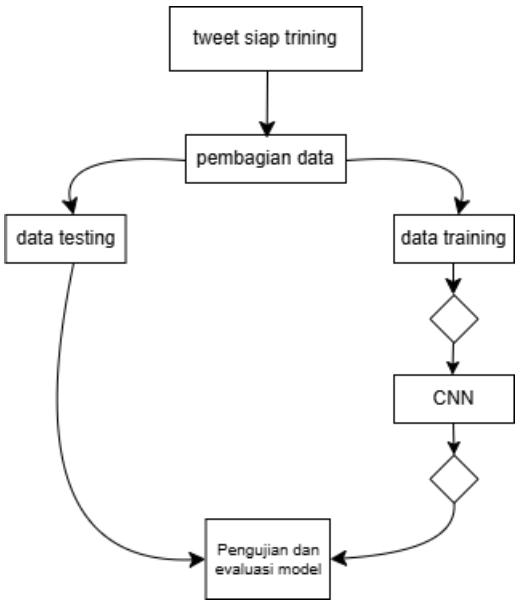
   Word2Vec merupakan salah satu teknik *shallow neural network* yang digunakan untuk mempelajari representasi vektor kata berdasarkan konteks kemunculannya dalam korpus teks. Metode ini memiliki dua arsitektur utama, yaitu *Continuous Bag of Words* (CBOW) yang memprediksi kata target berdasarkan konteks kata di sekitarnya, dan *Skip-Gram* yang memprediksi konteks kata di sekitar berdasarkan kata target. Word2Vec menghasilkan vektor berdimensi tetap sehingga kata-kata yang memiliki kemiripan makna akan direpresentasikan dengan jarak vektor yang berdekatan.[13] Pendekatan ini terbukti efektif dalam menangkap hubungan semantik antar kata, namun memiliki keterbatasan karena hanya memproses kata secara utuh sehingga kata baru (*out of vocabulary*) sulit diwakili.
2. FastText

   FastText dikembangkan untuk mengatasi keterbatasan Word2Vec terhadap kata baru atau kata tidak dikenal. FastText merepresentasikan kata sebagai kumpulan *character n-grams*, sehingga setiap kata diuraikan menjadi potongan sub-kata.[14] Dengan pendekatan ini, kata yang tidak ditemukan dalam kamus dapat tetap direpresentasikan melalui kombinasi n-gram penyusunnya. Keunggulan ini sangat relevan pada data X yang sarat singkatan, typo, dan kata tidak baku.

Berdasarkan hasil pengujian pada dataset yang berukuran relatif kecil, model CNN dengan representasi teks menggunakan Word2Vec menunjukkan tingkat akurasi validasi yang lebih stabil dibandingkan model dengan FastText. Kondisi ini disebabkan oleh fakta bahwa Word2Vec dilatih secara langsung menggunakan kumpulan data tweet bertema *Korea Style*, sehingga representasi vektor yang dihasilkan lebih sesuai dengan konteks dan karakteristik data penelitian. Sementara itu, model FastText yang digunakan merupakan model yang telah melalui proses pelatihan awal menggunakan data teks umum berbahasa Indonesia, sehingga kurang mampu menangkap variasi leksikal dan gaya bahasa khas media sosial. Selain itu, nilai F1-score pada model berbasis Word2Vec juga tercatat lebih tinggi, yang mengindikasikan kemampuan model tersebut dalam mengenali pola sentimen secara lebih konsisten, meskipun pada teks yang bersifat informal. Temuan ini menunjukkan bahwa pelatihan model *embedding* pada data dengan domain spesifik dapat menghasilkan representasi yang lebih kontekstual dan mendukung kinerja model analisis sentimen secara lebih optimal.

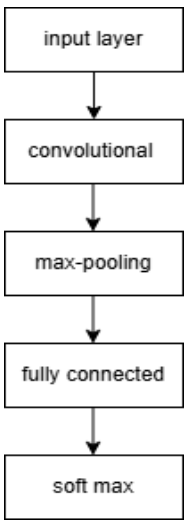
2.5 Training testing

Tahap berikutnya adalah *training data*, yaitu proses melatih model dengan input berupa vektor teks terkait topik *Korean style*. Dataset dibagi menjadi data *training* dan data *testing* dengan rasio 80:20.[15] Pelatihan CNN dilakukan menggunakan *library TensorFlow* berbasis *Python*, dengan skenario yang memanfaatkan data berlabel, metode *word embedding*, dan pengaturan parameter CNN.[16] Hasil tahap ini berupa model CNN dari setiap skenario pelatihan yang telah dijalankan. Diagram alur proses ditampilkan pada Gambar 3 berikut ini.



Gambar 3. Alur Pelatihan Dan Pengujian Model CNN

2.6 Arsitektur CNN



Gambar 4 arsitektur CNN (Moez Krichen : 2023)

Model yang digunakan dalam penelitian ini dibangun dengan arsitektur *Convolutional Neural Network* (CNN) yang dirancang khusus untuk memproses data teks. Arsitektur ini diawali dengan *embedding layer*

yang berfungsi mengubah token hasil pra-pemrosesan menjadi representasi vektor berdimensi 128 menggunakan dua jenis *word embedding*, yaitu Word2Vec dan FastText. Vektor embedding tersebut kemudian dimasukkan ke dalam *convolutional layer* satu dimensi (Conv1D) dengan 128 filter dan ukuran kernel 5 untuk mengekstraksi fitur lokal serta mendeteksi pola n-gram yang relevan dengan sentimen. Hasil ekstraksi selanjutnya diproses melalui *MaxPooling1D layer* guna mereduksi dimensi dan mempertahankan fitur paling dominan. Tahap berikutnya adalah *flatten layer* yang mengubah keluaran *pooling* menjadi vektor satu dimensi sehingga dapat dihubungkan dengan *fully connected layer* atau *dense layer* yang bertugas mengintegrasikan fitur-fitur penting.[17] Pada bagian akhir, digunakan *output layer* dengan fungsi aktivasi *Softmax* untuk menghasilkan probabilitas klasifikasi ke dalam tiga kategori sentimen, yaitu positif, netral, dan negatif. Arsitektur lengkap model CNN yang diterapkan dalam penelitian ini ditunjukkan pada Gambar 4.

2.7 Evaluasi Model

Model dievaluasi untuk menilai tingkat akurasi hasil prediksi terhadap data sebenarnya. Evaluasi dilakukan menggunakan confusion matrix, yang membandingkan hasil prediksi dengan label asli. Kinerja model kemudian diukur dan dibandingkan melalui metrik akurasi*, precision*, *recall*, dan *F1-score*.[18]

(1) Akurasi menunjukkan proporsi prediksi yang benar terhadap seluruh prediksi:

$$Akurasi = \frac{TP + TN}{TP + FP + FN + TN} \times 100\%$$

Keterangan: TP (*True Positive*), TN (*True Negative*), FP (*False Positive*), FN (*False Negative*).

(2) *Precision* mengukur ketepatan model dalam memprediksi kelas positif:

$$Precision = \frac{TP}{TP + FP}$$

(3) *Recall* menilai kemampuan model menemukan seluruh sampel positif yang sebenarnya:

$$Recall = \frac{TP}{TP + FN}$$

(4) F1-Score mempresentasikan keseimbangan antara precision dan recall:

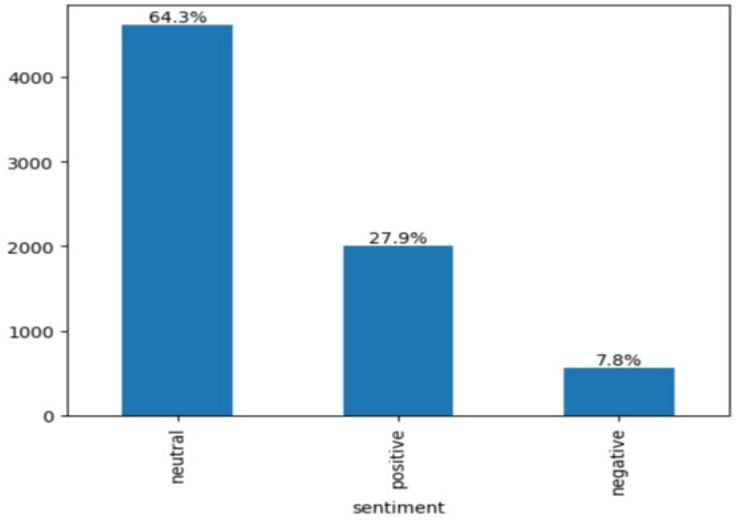
$$F1 = \frac{2 \times (Precision \times Recall)}{Precision + Recall}$$

Keempat metrik ini memberikan Gambaran menyeluruh mengenai performa model dalam klasifikasi sentiment.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Pemberian Label

Hasil dari analisis sentimen, di mana sebanyak 5.620 tweet diklasifikasikan ke dalam tiga kategori netral, positif, dan negative menggunakan pendekatan berbasis *Lexicon* dengan data yang telah melalui proses pra-pemrosesan teks sebelumnya.



Gambar 5. Grafik Analisis Sentimen

Berdasarkan grafik, mayoritas tweet (64,3%) termasuk kategori netral, menandakan sebagian besar pengguna X menyampaikan informasi tanpa ekspresi emosi yang kuat terhadap tren *Korean style*. Sebanyak 27,9% tweet tergolong positif, menunjukkan adanya ketertarikan atau apresiasi, misalnya berupa pujian, antusiasme, atau rekomendasi terkait produk dan gaya Korea. Sementara itu, hanya 7,8% tweet yang bernada negatif, yang merefleksikan kritik atau ketidakpuasan dalam jumlah kecil. Distribusi ini mengindikasikan bahwa persepsi publik terhadap *Korean style* cenderung netral hingga positif, dengan sentimen negatif relatif minim.

3.2 Implementasi *Convolutional Neural Network* (CNN)

A. Proses Preprocessing

Pada tahapan ini dilakukan penerapan *Convolutional Neural Network* (CNN) terhadap data teks hasil pra-pemrosesan yang digunakan sebagai input proses klasifikasi sentimen. Penerapan CNN ini bertujuan untuk mengetahui apakah setiap tweet yang dianalisis memiliki sentimen positif, netral, atau negatif sesuai konteks pembahasan tren *Korean Style*.

Implementasi model CNN pada penelitian ini dilakukan dengan memanfaatkan Pustaka *tensorflow* dan keras di lingkungan *Python.* Data hasil pra-pemrosesan sebanyak 5.620 tweet yang telah melalui tahap labeling ke-dalam tiga kelas Positif, negative, dan netral. Data dibagi dengan rasio 80% untuk data latih (*training set*) dan 20% untuk data uji (*testing set*) dengan menggunakan metode *stratified split* di *google colab* Serta batas *epoch* / literasi 10 *epoch*. Pembagian ini dilakukan secara proporsional untuk setiap kelas, ehingga perbandingan jumlah tweet pada masing-masing kelas di data latih dan uji tetap merepresentasikan distribusi aslinya. Distribusi data tiap kelas di tunjukkan pada Tabel 4 berikut ini

Tabel 4. Distribusi Data Training Dan Testing Per kelas

Kelas Sentimen	Jumlah Data	Training (80%)	Testing (20%)
Netral (1)	3.659	2.927	732
Positif (2)	1.524	1.219	305
Negatif (0)	437	349	88
Total	5620	4.495	1.125

Data input diubah menjadi urutan indeks kata dengan panjang tetap 100 token melalui proses *padding*, sedangkan label sentimen dikonversi ke dalam bentuk *one-hot encoding* sehingga setiap sampel memiliki representasi tiga dimensi sesuai jumlah kelas. Pelatihan model dilakukan dengan parameter utama yang di tunjukan pada Tabel 5.

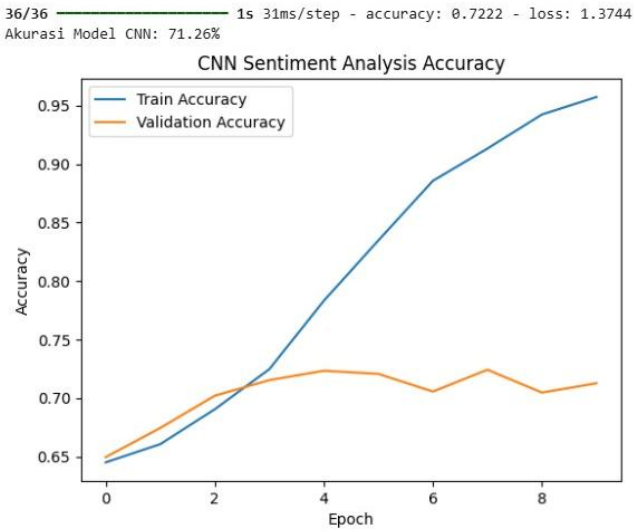
Tabel 5. Parameter dan Konfigurasi

Komponen	Konfigurasi
optimizer	adam
Loss Function	Categorical Crossentropy
Epoch	10
Bact Size	64
Filter Size	128
Kernel Size	3,4,5
Pooling	Maxpooling1D
Dropout Rate	0.5
Activation Fuction	ReLU (hidden), Softmax (output)

Pelatihan berlangsung selama 10 *epoch* dengan *optimizer Adam* yang efektif mempercepat konvergensi pada deep learning. Selama proses ini, bobot jaringan diperbarui secara iteratif untuk meminimalkan *loss function*. Setelah pelatihan selesai, performa model dievaluasi pada data uji. Hasilnya menunjukkan bahwa model mampu menangkap pola linguistik yang berkaitan dengan polaritas sentimen, seperti kata-kata positif, negatif, serta konteks negasi dalam kalimat.

3.3 Visualisasi Hasil Model

Untuk melihat performa pelatihan model CNN selama proses *training*. Dilakukan visualisasi grafik akurasi dan *loss* terhadap jumplah *epoch.* Visualisasi ini membantu dalam memahami bagaimana model belajar dari data Selma proses pelatihan berlangsung. Berikut grafik akurasi model CNN yang di tunjukan pada Gambar 6.



Gambar 6. Grafik Akurasi Model CNN

Gambar 6 emperlihatkan perkembangan akurasi model pada data pelatihan dan validasi di setiap *epoch*. Akurasi pelatihan naik tajam dari sekitar 63,68% pada *epoch* pertama menjadi 96,24% pada *epoch* ke-10, menandakan model mampu mempelajari pola data dengan baik. Sebaliknya, akurasi validasi meningkat lebih lambat, dari 64,95% hanya mencapai 71,26% di akhir pelatihan. Selisih yang cukup besar antara kedua nilai tersebut menunjukkan adanya gejala *overfitting* setelah *epoch* ke-6, ketika akurasi pelatihan terus meningkat sementara akurasi validasi cenderung stagnan. Pelatihan dengan 10 *Epoch* ditunjukkan dengan Tabel 6 berikut ini.

Tabel 6. Hasil Pelatihan model CNN

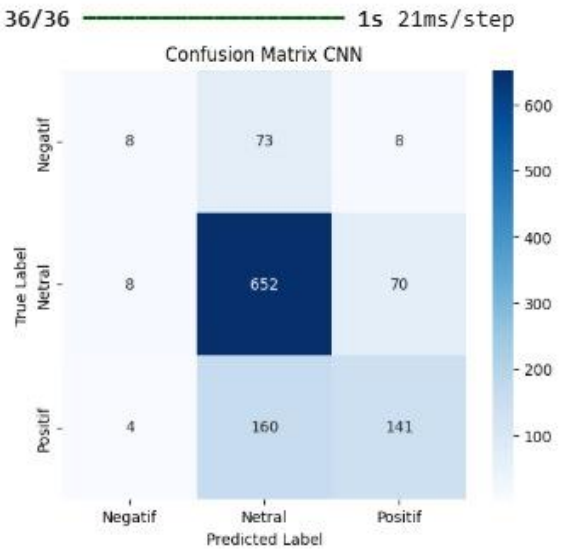
NO	*Epoch*	*Accuracy*	*Loss*	*Accuracy validation*	*Loss validation*
1.	Epoch ke-1	0.6368	0.8465	0.6495	0.8028
2.	Epoch ke-2	0.6512	0.7913	0.6744	0.7585
3.	Epoch ke-3	0.6876	0.7105	0.7020	0.7295
4.	Epoch ke-4	0.7292	0.5968	0.7153	0.7248
5.	Epoch ke-5	0.7784	0.5141	0.7233	0.7544
6.	Epoch ke-6	0.8411	0.3933	0.7206	0.7944
7.	Epoch ke-7	0.8886	0.2991	0.7055	0.9562
8.	Epoch ke-8	0.9176	0.2217	0.7242	1.0360
9.	Epoch ke-9	0.9413	0.1724	0.7046	1.1819
10.	Epoch ke-10	0.9624	0.1217	0.7126	1.4348

Seiring dengan akurasi, tabel 6 menampilkan perubahan nilai *loss* pada data pelatihan dan validasi selama 10 epoch. Nilai *loss* pelatihan turun tajam dari 0,8465 menjadi 0,1217, menandakan kesalahan prediksi pada data latih semakin kecil. Sebaliknya, *loss* validasi meningkat setelah *epoch* ke-6, dari 0,7248 menjadi 1,4348 pada akhir pelatihan. Pola ini mengindikasikan terjadinya *overfitting*, yaitu kondisi ketika model terlalu menyesuaikan diri dengan data latih sehingga kemampuan generalisasi terhadap data baru menurun.

3.4 Evaluasi Model CNN

Evaluasi model CNN tidak hanya dilakukan dengan mengamati akurasi keseluruhan, tetapi juga dengan menganalisis *confusion matrix* dan *classification report* untuk mendapatkan informasi yang lebih komprehensif mengenai performa model terhadap masing-masing kelas sentimen.

a) Confusion Matrix



Gambar 7. Confusion Matrix

Gambar 7 menampilkan *confusion matrix* model CNN yang memperlihatkan jumlah prediksi benar dan salah untuk masing-masing kelas sentimen: Negatif, Netral, dan Positif. Dengan prediksi kelas sentiment yang di tunjukkan tabel 7 berikut ini.

Nilai F1-score yang rendah pada kelas negatif disebabkan oleh ketidakseimbangan distribusi data atau *class imbalance*, di mana proporsi tweet netral (64,3%) jauh lebih besar daripada positif (27,9%) dan negatif (7,8%). Ketidakseimbangan ini mendorong model untuk lebih sering memprediksi kelas mayoritas demi meminimalkan *loss*, sehingga representasi fitur kelas negatif tidak terlatih dengan baik. Selain itu, karakteristik linguistik tweet negatif yang sering mengandung sarkasme, kata gaul, dan negasi ganda semakin menyulitkan proses klasifikasi. Kombinasi kedua faktor tersebut mengakibatkan model CNN cenderung bias terhadap kelas netral.[19] Untuk mengatasi hal ini, penelitian lanjutan dapat menerapkan teknik *oversampling*, pemberian bobot kelas, atau memanfaatkan model transformer seperti IndoBERT yang lebih sensitif terhadap konteks semantik.

Tabel 7. Prediksi Kelas Sentimen

	Predicted: Negatif	**Predicted: Netral**	**Predicted: Positif**
Actual: Negatif	8	7	8
Actual: Netral	8	652	70
Actual: Positif	4	160	141

Kelas **Netral** memiliki jumlah prediksi benar terbanyak (652 dari 730), menunjukkan bahwa model paling baik dalam mengklasifikasikan sentimen netral. Kelas **Positif** cenderung sering diprediksi sebagai **Netral**, dengan 160 dari 305 tweet positif diklasifikasikan salah. Kelas **Negatif** memiliki performa terendah, di mana hanya 8 dari 89 data diklasifikasikan benar.

b) Classification report

Tabel 8. Clasifikation Report

Label	Precision	Recall	F1-Score	Suport
Negatif	0.40	0.09	0.15	89
Netral	0.74	0.89	0.81	730
Positif	0.64	0.46	0.54	305
Accuracy			**0.71**	**1124**
Marco Avg	0.59	0.48	0.50	1124
Weighted Avg	0.68	0.71	0.68	1124

- *Precision* tertinggi dicapai pada kelas Netral (0,74), menunjukkan sebagian besar prediksi netral sesuai dengan label sebenarnya.
- *Recall* tertinggi juga terdapat pada kelas Netral (0,89), menandakan sebagian besar tweet netral berhasil teridentifikasi oleh model.
- *F1-score* tertinggi sebesar 0,81 diperoleh pada kelas Netral, sedangkan kelas Negatif memiliki *F1-score* terendah (0,15), yang menandakan model masih kesulitan mengenali tweet negatif.

Secara keseluruhan, model CNN meraih akurasi 71,26% pada data uji dengan *macro average F1-score* 0,50, menunjukkan performa antar kelas belum seimbang dan keberhasilan model lebih dominan pada kelas netral. Kondisi ini menekankan perlunya penyeimbangan jumlah data dan peningkatan kemampuan model dalam mendeteksi sentimen negatif pada pengembangan berikutnya.

4. KESIMPULAN

Penelitian ini mengimplementasikan dan mengevaluasi model *Convolutional Neural Network* (CNN) untuk klasifikasi sentimen tweet berbahasa Indonesia terkait tren *Korean style*. Model CNN berhasil melakukan klasifikasi dengan akurasi rata-rata 71,26% dan F1-score tertinggi 0,81 pada kelas netral, sedangkan kelas positif mencapai 0,54 dan kelas negatif hanya 0,15. Hasil ini menunjukkan kemampuan CNN dalam mengekstraksi fitur lokal teks pendek, tetapi performa masih rendah pada kelas minoritas. Word2Vec memberikan hasil lebih stabil dalam representasi teks informal dibandingkan FastText. Tantangan utama model terletak pada ketidakseimbangan data dan kesulitan dalam mengenali sentimen negatif. Ketidakseimbangan kelas (*class imbalance*) teridentifikasi sebagai faktor utama rendahnya performa pada kelas negatif. Oleh karna itu, Penelitian selanjutnya disarankan menerapkan metode *oversampling*, *class weighting*, atau model transformer seperti IndoBERT untuk meningkatkan generalisasi model.

DAFTAR RUJUKAN

[1] A. A. R. Rahma, H. Ardianti, dan K. Firman, "Peran Media Sosial Dalam Dinamika Sosial Masyarakat Kontemporer".

[2] A. Rahmadani dan Y. Anggarini, "Pengaruh Korean Wave dan Brand Ambassador pada Pengambilan Keputusan Konsumen," *Telaah Bisnis*, vol. 22, no. 1, hlm. 59, Jul 2021, doi: 10.35917/tb.v22i1.225.

[3] F. Es-sabery, I. Es-sabery, J. Qadir, B. Sainz-de-Abajo, dan B. Garcia-Zapirain, "A hybrid Hadoop-based sentiment analysis classifier for tweets associated with COVID-19 utilizing two machine learning algorithms: CNN, and fuzzy C4.5," *J. Big Data*, vol. 11, no. 1, hlm. 176, Des 2024, doi: 10.1186/s40537-024-01014-4.

[4] A. Kartika Sari, Akhmad Irsyad, Dinda Nur Aini, Islamiyah, dan Stephanie Elfriede Ginting, "Analisis Sentimen Twitter Menggunakan Machine Learning untuk Identifikasi Konten Negatif," *Adopsi Teknol. Dan Sist. Inf. ATASI*, vol. 3, no. 1, hlm. 64–73, Jun 2024, doi: 10.30872/atasi.v3i1.1373.

[5] J. Chen, Y. Hu, J. Liu, Y. Xiao, dan H. Jiang, "Deep Short Text Classification with Knowledge Powered Attention," 21 Februari 2019, *arXiv*: arXiv:1902.08050. doi: 10.48550/arXiv.1902.08050.

[6] A. N. Ma'aly, D. Pramesti, A. D. Fathurahman, dan H. Fakhrurroja, "Exploring Sentiment Analysis for the Indonesian Presidential Election Through Online Reviews Using Multi-Label Classification with a Deep Learning Algorithm," *Information*, vol. 15, no. 11, hlm. 705, Nov 2024, doi: 10.3390/info15110705.

[7] A. Khazaie, N. B. Seghouani, dan F. Bugiotti, "Smart Crawling: A New Approach toward Focus Crawling from Twitter," 8 Oktober 2021, *arXiv*: arXiv:2110.06022. doi: 10.48550/arXiv.2110.06022.

[8] E. Haddi, X. Liu, dan Y. Shi, "The Role of Text Pre-processing in Sentiment Analysis," *Procedia Comput. Sci.*, vol. 17, hlm. 26–32, 2013, doi: 10.1016/j.procs.2013.05.005.

[9] S. Susandri, S. Defit, dan M. Tajuddin, "Enhancing Text Sentiment Classification with Hybrid CNN-BiLSTM Model on WhatsApp Group," *J. Adv. Inf. Technol.*, vol. 15, no. 3, hlm. 355–363, 2024, doi: 10.12720/jait.15.3.355-363.

[10] Y. Qi dan Z. Shabrina, "Sentiment analysis using Twitter data: a comparative application of lexicon- and machine-learning-based approach," *Soc. Netw. Anal. Min.*, vol. 13, no. 1, hlm. 31, Feb 2023, doi: 10.1007/s13278-023-01030-x.

[11] "Sentiment Analysis using VADER - Using Python," GeeksforGeeks. Diakses: 28 September 2025. [Daring]. Tersedia pada: https://www.geeksforgeeks.org/python/python-sentiment-analysis-using-vader/

[12] M. S. Md Suhaimin, M. H. Ahmad Hijazi, dan E. G. Moung, "Annotated dataset for sentiment analysis and sarcasm detection: Bilingual code-mixed English-Malay social media data in the public security domain," *Data Brief*, vol. 55, hlm. 110663, Agu 2024, doi: 10.1016/j.dib.2024.110663.

[13] T. Mikolov, K. Chen, G. Corrado, dan J. Dean, "Efficient Estimation of Word Representations in Vector Space," 7 September 2013, *arXiv*: arXiv:1301.3781. doi: 10.48550/arXiv.1301.3781.

[14] P. Bojanowski, E. Grave, A. Joulin, dan T. Mikolov, "Enriching Word Vectors with Subword Information," 19 Juni 2017, *arXiv*: arXiv:1607.04606. doi: 10.48550/arXiv.1607.04606.

[15] A. Nazarkar, H. Kuchulakanti, C. S. Paidimarry, dan S. Kulkarni, "Impact of Various Data Splitting Ratios on the Performance of Machine Learning Models in the Classification of Lung Cancer," dalam *Proceedings of the Second International Conference on Emerging Trends in Engineering (ICETE 2023)*, vol. 223, B. Raj, S. Gill, C. A. G. Calderon, O. Cihan, P. Tukkaraja, S. Venkatesh, V. M. S., M. Mudigonda, M. Gaddam, dan R. K. Dasari, Ed., dalam Advances in Engineering Research, vol. 223. , Dordrecht: Atlantis Press International BV, 2023, hlm. 96–104. doi: 10.2991/978-94-6463-252-1_12.

[16] D. S. Asudani, N. K. Nagwani, dan P. Singh, "Impact of word embedding models on text analytics in deep learning environment: a review," *Artif. Intell. Rev.*, vol. 56, no. 9, hlm. 10345–10425, Sep 2023, doi: 10.1007/s10462-023-10419-1.

[17] A. N. Ma'aly, D. Pramesti, A. D. Fathurahman, dan H. Fakhrurroja, "Exploring Sentiment Analysis for the Indonesian Presidential Election Through Online Reviews Using Multi-Label Classification with a Deep Learning Algorithm," *Information*, vol. 15, no. 11, hlm. 705, Nov 2024, doi: 10.3390/info15110705.

[18] S. Sathyanarayanan, "Confusion Matrix-Based Performance Evaluation Metrics," *Afr. J. Biomed. Res.*, hlm. 4023–4031, Nov 2024, doi: 10.53555/AJBR.v27i4S.4345.

[19] C. Suhaeni dan H.-S. Yong, "Mitigating Class Imbalance in Sentiment Analysis through GPT-3-Generated Synthetic Sentences," *Appl. Sci.*, vol. 13, no. 17, hlm. 9766, Agu 2023, doi: 10.3390/app13179766.