Jurnal Informatika: Jurnal pengembangan IT

Vol. 11, No. 2, 2026
ISSN : 2477-5126, DOI:10.30591/jpit.v11i2.10376

Perbandingan Performa CNN, BiLSTM, dan LSTM untuk Analisis Sentimen Komentar Berbahasa Melayu Bengkulu

Julia Purnama Sari [1], Funny Farady Coastera [2], Niska Ramadani [3], Esti Asmareta Ayu [4]
[1,3]Program Studi Sistem Informasi,Universitas Bengkulu, Indonesia
[2,4] Program Studi Informatika, Universitas Bengkulu, Indonesia
juliapurnamasari@unib.ac.id, ffaradyc@unib.ac.id, niskaramadani@unib.ac.id, asmaretaayu@gmail.com

Info Artikel

Riwayat Artikel:

Received 2026-03-13
Revised 2026-06-25
Accepted 2026-07-01

Corresponding Author:

Julia Purnama Sari

Email: juliapurnamasari@unib.ac.id

Abstract *– Bengkulu Malay is an under-resourced regional language used in daily digital communication, including public comments on Instagram and TikTok. Its informal vocabulary, spelling variation, and limited language resources make sentiment classification more challenging than for standard Indonesian. This study compares Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Bidirectional Long Short-Term Memory (BiLSTM) for sentiment analysis of Bengkulu Malay comments. The dataset was collected from three local issues, namely Q1, Q2, and Q3, with 22,888 raw comments and 3,961 labeled comments after selection. The research pipeline includes data collection, labeling, text preprocessing adapted to Bengkulu Malay, Word2Vec Skip-gram embedding, data splitting into training, validation, and testing sets, model training, and evaluation using accuracy, precision, recall, F1-score, and confusion matrix. The final labeled dataset was split into 80% training and 20% testing data, and 20% of the training data was used for validation. The results show that BiLSTM achieved the best accuracy on all datasets, with 91% on Q1, 85% on Q2, and 70% on Q3. CNN achieved 69%, 60%, and 64%, while LSTM achieved 49%, 55%, and 64%. These results indicate that BiLSTM is more effective in capturing bidirectional context in Bengkulu Malay comments, although performance decreases when class imbalance becomes stronger.*

Keywords: *Bengkulu Malay, BiLSTM, CNN, LSTM, Sentiment Analysis*

Abstrak *– Bahasa Melayu Bengkulu merupakan bahasa daerah dengan sumber daya pemrosesan bahasa alami yang masih terbatas, tetapi banyak digunakan dalam komunikasi digital, termasuk komentar publik pada Instagram dan TikTok. Variasi kosakata informal, ejaan tidak baku, dan konteks lokal membuat analisis sentimen terhadap bahasa ini menjadi lebih menantang. Penelitian ini membandingkan performa Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), dan Bidirectional Long Short-Term Memory (BiLSTM) untuk analisis sentimen komentar berbahasa Melayu Bengkulu. Dataset dikumpulkan dari tiga isu lokal, yaitu Q1, Q2, dan Q3, dengan total 22.888 komentar mentah dan 3.961 komentar berlabel setelah proses seleksi. Tahapan penelitian meliputi pengumpulan data, pelabelan, preprocessing teks yang disesuaikan dengan karakteristik bahasa Melayu Bengkulu, pembentukan word embedding menggunakan Word2Vec arsitektur Skip-gram, pembagian data, pemodelan, dan evaluasi menggunakan akurasi, precision, recall, F1-score, serta confusion matrix. Dataset berlabel dibagi menjadi 80% data latih dan 20% data uji, lalu 20% data latih digunakan sebagai data validasi. Hasil pengujian menunjukkan bahwa BiLSTM memperoleh akurasi terbaik pada seluruh dataset, yaitu 91% pada Q1, 85% pada Q2, dan 70% pada Q3. CNN memperoleh akurasi 69%, 60%, dan 64%, sedangkan LSTM memperoleh akurasi 49%, 55%, dan 64%. Hasil ini menunjukkan bahwa BiLSTM lebih efektif menangkap konteks dua arah pada komentar berbahasa Melayu Bengkulu, meskipun performanya menurun ketika ketidakseimbangan kelas semakin tinggi.*

Kata Kunci*: Analisis Sentimen, Bahasa Melayu Bengkulu, BiLSTM, CNN, LSTM*

I. PENDAHULUAN

Bahasa Melayu Bengkulu merupakan salah satu kekayaan budaya Indonesia yang digunakan oleh masyarakat di Provinsi Bengkulu dalam berbagai bentuk komunikasi, termasuk di media sosial dan platform digital. Seiring dengan banyaknya pengguna Bengkulu dalam menggunakan internet dan media sosial, maka komentar-komentar berbahasa Melayu Bengkulu juga semakin banyak ditemui dalam diskusi online, baik di forum publik, platform berita, maupun media sosial. Komentar-komentar ini mencerminkan beragam sentimen, baik positif maupun negatif, terhadap isu-isu lokal seperti kebijakan pemerintah daerah, pendidikan, sosial budaya, hingga pelayanan publik.

Namun, karena sifat bahasa daerah yang memiliki struktur dan kosakata khas [1], analisis sentimen terhadap komentar berbahasa Melayu Bengkulu menjadi tantangan tersendiri. Teknik analisis sentimen konvensional yang umumnya dirancang untuk bahasa Indonesia atau Inggris sering kali kurang optimal dalam memahami makna dan

konteks dari bahasa lokal. Oleh karena itu, diperlukan pendekatan khusus berbasis pembelajaran mesin dan *deep learning* yang mampu menangkap pola dan konteks dalam bahasa Bengkulu secara lebih mendalam.

Tantangan dalam analisis sentimen komentar berbahasa Melayu Bengkulu adalah ketersediaan data yang terbatas dan belum terstruktur. Tidak seperti bahasa Indonesia yang telah memiliki banyak korpus dan sumber daya NLP, bahasa Melayu Bengkulu masih belum banyak dijadikan objek dalam pengembangan teknologi bahasa. Oleh karena itu, proses *preprocessing* seperti tokenisasi, *stopword removal*, dan *stemming* perlu disesuaikan secara khusus agar dapat menangani variasi kosakata dan struktur kalimat khas Bengkulu. Keberhasilan analisis sentimen dalam konteks ini sangat dipengaruhi oleh kualitas data dan efektivitas teknik *preprocessing* yang digunakan sebelum data dianalisis oleh model.

Saat ini, metode deep learning seperti *Convolutional Neural Network* (CNN), *Long Short-Term Memory* (LSTM), dan *Bidirectional LSTM* (BiLSTM) telah terbukti efektif dalam berbagai tugas pemrosesan bahasa alami *(Natural Language Processing*/NLP), termasuk analisis sentimen. CNN unggul dalam mengenali fitur-fitur penting pada gambar melalui proses ekstraksi fitur yang mendalam, sehingga memungkinkan sistem mendeteksi objek maupun teks dengan tingkat akurasi yang tinggi [2], sedangkan LSTM memiliki kemampuan untuk menyimpan informasi dalam periode waktu yang lama serta mengatasi permasalahan hilangnya gradien yang biasa terjadi pada RNN [3]. Sementara itu, BiLSTM menawarkan keunggulan dengan memproses data secara simultan dari dua arah, sehingga memungkinkan pengenalan pola temporal yang lebih rumit [4]. Berdasarkan keunggulan dari ketiga algoritma tersebut, dalam penerapan model *deep learning*, pemilihan arsitektur yang tepat sangat krusial untuk mencapai performa terbaik. Perbandingan ketiga model menjadi penting untuk menentukan arsitektur mana yang paling efektif untuk analisis sentimen pada komentar dalam bahasa seperti bahasa Bengkulu.

Penelitian sebelumnya telah menunjukkan bahwa CNN lebih baik dibandingkan dengan model klasifikasi Naïve Bayes [5]. Jika dibandingkan dari segi akurasi dan efektivitas, algoritma CNN menunjukkan kinerja yang lebih baik dibandingkan dengan algoritma IndoBERT [6]. LSTM lebih unggul dalam klasifikasi sentimen dibandingkan Naïve Bayes [7]. Perbandingan antara algoritma LSTM dan RNN menunjukkan bahwa LSTM dapat menghasilkan prediksi yang sedikit lebih akurat dibandingkan dengan RNN [4]. Algortima BiLSTM memiliki tingkat akurasi yang lebih tinggi dibandingkan dengan model algoritma CNN[8]. BILSTM menunjukan efektifitas lebih baik dari SVM walaupun perbedaan yang hanya sedikit [9].

Penelitian terdahulu telah banyak membahas analisis sentimen dengan berbagai pendekatan *machine learning* dan *deep learning*. Afidah et al. [10] menguji pengaruh parameter Word2Vec pada klasifikasi sentimen. Penelitian tersebut menunjukkan pentingnya representasi kata dalam meningkatkan performa model. Namun, penelitian tersebut belum membahas bahasa Melayu Bengkulu dan belum melakukan komparasi model CNN, LSTM, dan BiLSTM pada komentar berbahasa daerah. Romadhoni dan Holle [11] membandingkan Naïve Bayes dan LSTM pada analisis sentimen Twitter terkait kebijakan publik. Penelitian tersebut relevan karena menggunakan data media sosial, tetapi objek kajiannya masih berfokus pada bahasa Indonesia dan belum menelaah karakteristik bahasa daerah.

Selanjutnya, Apriansyah et al. [12] membandingkan IndoBERT dan IndoRoBERTa untuk analisis sentimen film dokumenter. Penelitian tersebut menekankan penggunaan model berbasis Transformer, tetapi belum membahas model *deep learning* sekuensial seperti LSTM dan BiLSTM pada bahasa lokal. Pramana et al. [13] juga membandingkan IndoBERT dan BiLSTM untuk klasifikasi teks hukum. Meskipun penelitian tersebut menggunakan pendekatan *deep learning*, domain yang dikaji berbeda dan tidak menggunakan dataset komentar berbahasa Melayu Bengkulu. Dengan demikian, sebagian besar penelitian terdahulu masih berfokus pada bahasa Indonesia, domain umum, atau domain khusus seperti kebijakan publik, film dokumenter, dan teks hukum.

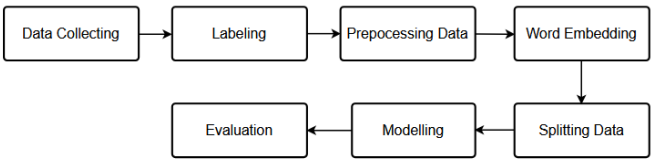
Berdasarkan pemetaan tersebut, terdapat celah penelitian yang perlu dikaji lebih lanjut. Belum banyak penelitian yang secara khusus membahas analisis sentimen pada komentar berbahasa Melayu Bengkulu. Selain itu, belum ditemukan penelitian yang membandingkan performa CNN, LSTM, dan BiLSTM dengan representasi Word2Vec pada dataset komentar berbahasa Melayu Bengkulu. Keterbatasan ini menunjukkan bahwa penelitian pada NLP bahasa daerah masih perlu dikembangkan, terutama untuk menyediakan baseline awal bagi klasifikasi sentimen berbasis *deep learning* pada bahasa Melayu Bengkulu.

Oleh karena itu, penelitian ini berfokus pada perbandingan kinerja model CNN, LSTM, dan BiLSTM dengan Word2Vec dalam klasifikasi sentimen komentar berbahasa Melayu Bengkulu. Penelitian ini diharapkan dapat mengisi gap pada pengolahan bahasa daerah, khususnya dalam analisis sentimen berbasis teks. Hasil penelitian juga diharapkan dapat menjadi acuan awal bagi peneliti dan pengembang sistem dalam memilih model *deep learning* yang tepat untuk pengolahan komentar berbahasa lokal.

II. METODE

Penelitian ini menggunakan pendekatan kuantitatif dengan desain eksperimental-komparatif. Fokus penelitian adalah membandingkan performa tiga arsitektur *deep learning*, yaitu CNN, LSTM, dan BiLSTM. Objek penelitian berupa komentar berbahasa Melayu Bengkulu yang dikumpulkan dari Instagram dan TikTok. Penelitian ini dilakukan dalam tujuh tahapan utama, yaitu data *collecting*, *labeling*, *preprocessing data*, *word embedding*, *splitting data*,

modeling, dan *evaluation*. Gambar 1 menunjukkan alur penelitian dari pengumpulan komentar sampai evaluasi model. Setiap tahap dirancang agar proses pengolahan data berbahasa Melayu Bengkulu dapat berjalan sistematis dan dapat direplikasi.

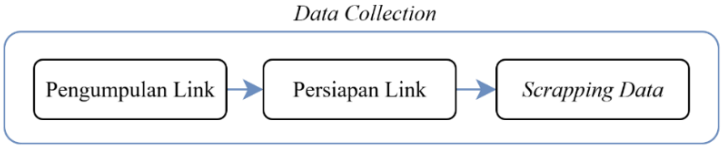


Gambar 1. Tahapan Penelitian

Adapun tahan penelitian ini sebagai berikut :

A. Data Collecting

Tahapan pertama dalam melakukan penelitian ini ialah mempersiapkan data. Persiapan ini meliputi pengumpulan link-link postingan dari ketiga topik yang diangkat. *Link* tersebut kemudian disimpan berdasarkan topik dari postingan yang kemudian akan diolah sedemikian rupa hingga dapat dipergunakan pada proses berikutnya, yaitu *scraping* data. Teknik *scraping* dilakukan untuk mengambil beberapa informasi pada kolom komentar sebuah postingan seperti komentar pengguna, nama pengguna, dan waktu pengguna mengunggah komentar tersebut [14]. Kumpulan informasi yang didapatkan dari setiap postingan ini yang kelak akan disebut sebagai dataset. Langkah-langkah pada tahap *data preparation* dapat dilihat pada gambar 2.



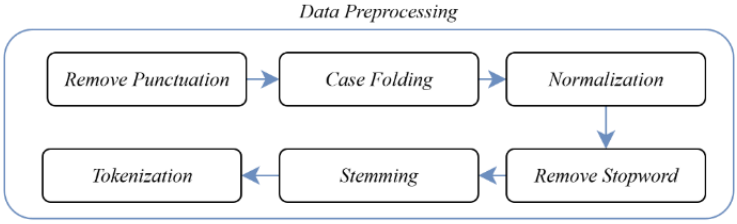
Gambar 2.Diagram Alur Data Collection

B. Labeling

Tahapan berikutnya ialah *labeling*, yaitu pemberian label sentimen pada data komentar [15]. Label dibagi menjadi tiga kelas, yaitu positif, netral, dan negatif. Kelas positif diberikan pada komentar yang menunjukkan dukungan, apresiasi, atau penilaian baik. Kelas negatif diberikan pada komentar yang menunjukkan kritik, penolakan, kekecewaan, atau hinaan. Kelas netral diberikan pada komentar informatif, tidak memuat opini jelas, atau bersifat pertanyaan. Pemberian label pada setiap kelas direpresentasikan dalam bentuk numerik. Angka satu (1) sebagai sentimen positif, angka nol (0) sebagai sentimen netral, dan minus satu (-1) sebagai sentiment negatif.

C. Preprocessing Data

Proses *data preprocessing* merupakan proses untuk membersihkan serta mempersiapkan data sebelum dilakukan pemrosesan lebih lanjut [16]. Teknik ini bertujuan untuk meningkatkan kualitas data sehingga lebih efektif dan layak untuk dianalisis. Terdapat beberapa tahapan dalam proses ini yang dapat dilihat pada gambar 3 [17]:



Gambar 3. Diagram Alur Data Preprocessing

D. Word Embedding

Setelah proses tokenisasi selesai, teknik *word embedding* seperti Word2Vec digunakan untuk menentukan vektor-vektor yang memiliki kemiripan atau kedekatan makna antar kata. Word2Vec bekerja dengan mempertimbangkan konteks kata, sehingga dibandingkan dengan metode embedding tradisional, Word2Vec menghasilkan kinerja yang lebih unggul, kecepatan pemrosesan yang lebih tinggi, serta kemampuan generalisasi yang baik.

E. Splitting Data

Langkah berikutnya adalah melakukan pembagian *(splitting)* terhadap dataset. Data tersebut akan dipisahkan ke dalam tiga kelompok utama, yakni data pelatihan *(training)*, data validasi *(validation)*, dan data pengujian *(testing).*

F. Modeling

Pada tahap modeling, tiga arsitektur diterapkan secara terpisah, yaitu CNN, LSTM, dan BiLSTM. Ketiga model menggunakan input hasil *tokenization* dan *embedding* Word2Vec agar perbandingan fokus pada perbedaan arsitektur model. Implementasi dilakukan menggunakan Python dengan pustaka TensorFlow/Keras, Gensim, Pandas, NumPy, Scikit-learn, dan Matplotlib.

G. Evaluation

Setelah proses pelatihan model selesai, evaluasi performa dilakukan menggunakan sejumlah metrik utama. Pertama, perubahan nilai loss dan akurasi selama proses training dan validasi divisualisasikan menggunakan Matplotlib untuk melihat stabilitas pembelajaran model pada setiap epoch. Kedua, *confusion matrix* digunakan untuk menunjukkan distribusi prediksi benar dan salah pada setiap kelas [18]. Ketiga, metrik *precision, recall, F1-score*, dan *accuracy* dihitung untuk menilai performa model secara lebih menyeluruh. Selain itu, penelitian ini menambahkan analisis ketidakseimbangan kelas dan uji Friedman berbasis akurasi antar dataset sebagai pengujian statistik indikatif. Uji Friedman dipilih karena membandingkan lebih dari dua model pada beberapa dataset yang sama. Namun, hasil uji statistik perlu ditafsirkan secara hati-hati karena jumlah dataset hanya tiga.

III. HASIL DAN PEMBAHASAN

Hasil penelitian ini menjelaskan data *collecting, labeling, data preprocessing, word embedding*, dan *splitting*. Selain itu, pada tahap ini dipaparkan pemodelan untuk analisis sentimen menggunakan BiLSTM, CNN, dan LSTM, beserta evaluasi dari ketiga model tersebut.

A. Data Collecting

Pengumpulan data dilakukan melalui beberapa tahapan sebagai berikut:

1. Pengumpulan link

Langkah awal tahapan ini ialah mengumpulkan *link* postingan dari beberapa pengguna pada media sosial TikTok dan Instagram. *Link* tersebut merujuk pada postingan yang mengandung gambar, video, atau reels. Pada media sosial Instagram pencarian postingan dilakukan pada akun-akun portal berita seperti akun @bengkuluinfo, @infobengkulu_ dan @infobengkuluraya, sebagai portal berita yang terkenal di kalangan masyarakat Bengkulu, serta akun tokoh-tokoh masyarakat yang memiliki kesesuaian konteks dengan topik seperti akun @helmi_hasan_dai, @ir._mian dan @rohidin.merysah.

Pencarian postingan pada media sosial TikTok dilakukan dengan menggunakan kata kunci (*keyword*) tertentu. Penggunaan kata kunci ini bertujuan mengefisiensi penemuan data yang sesuai dengan topik dengan menampilkan rekomendasi postingan-postingan terkait. Seperti pada topik pertama, kata kunci yang digunakan meliputi "Pasangan cagub 2025 paslon Helmi-Mian" dan "Pasangan cagub 2025 paslon 01 Bengkulu". Pada topik kedua menggunakan kata kunci "Pasangan cagub 2025 paslon rohidin-meriani" dan "Pasangan cagub 2025 paslon 02 Bengkulu". Sementara itu topik ketiga menggunakan kata kunci seperti "Bengkulu Bumi Merah Putih".

2. Persiapan Link

Langkah selanjutnya yang dilakukan adalah mempersiapkan *link* agar dapat digunakan dalam proses scraping. Tahapan ini diperlukan pada hasil pengumpulan *link* Instagram. Persiapan yang dilakukan antaranya dapat dilihat pada tabel 1.

TABEL 1
PERSIAPAN LINK

Jenis Perubahan	Perubahan URL
Link URL	https://www.instagram.com/bengkuluinfo/reel/ DC3HYfMzGkl/
Penghilangan nama akun pengguna pada link	https://www.instagram.com/ reel/DC3HYfMzGkl/
Mengubah konten tujuan dalam bentuk postingan	https://www.instagram.com/p/DC3HYfMzGkl/

3. Scrapping Data

Tahapan ini merupakan proses pengambilan komentar dari *link* postingan yang telah dikumpulkan sebelumnya dengan menggunakan teknik *scraping* data. Pada tahap ini peneliti menggunakan dua teknik *scraping* yang berbeda untuk tiap-tiap media sosial.

a. *Scraping* data Instagram

Pengumpulan data dari media sosial Instagram menggunakan teknik *Web Scraping* dengan bantuan *Selenium WebDriver*. Penggunaan *Selenium WebDriver* memungkinkan komputer menjalankan mesin pencarian, seperti *Chrome* dan *Firefox*, secara otomatis dengan *input* berupa *link* postingan. Hasil dari proses ini berupa data Excel yang memuat nama pengguna, komentar dan waktu komentar.

b. *Scraping* data TikTok

Pada platform TikTok komentar pada sebuah postingan didapatkan dengan menggunakan API. *Requests API comment* dilakukan berdasarkan *link* unggahan TikTok yang akan diambil komentarnya, yang selanjutnya akan tersimpan kedalam format JSON. Untuk mempermudah tahapan berikutnya, data tersebut kemudian diubah kedalam format Excel yang berisikan kolom id komentar, nama pengguna, komentar dan waktu komentar tersebut diposting. Penggunaan kedua teknik ini memiliki keterbatasan dalam mengambil komentar yang merupakan komentar balasan dari komentar utama (*reply*). Dengan itu data yang berhasil diambil hanyalah data komentar utama saja. Hasil dari tahapan *scraping* ini berupa 3 buah *dataset* yang mewakili setiap topik, yang lebih lanjut akan disebut sebagai *dataset* Q1, Q2, dan Q3. *Dataset* Q1 merupakan *dataset* dari topik sentiment publik terhadap calon gubernur pasangan 01 Helmi-Mian, *dataset* Q2 merupakan *dataset* dari topik sentiment publik terhadap calon gubernur pasangan 02 Rohidin-Meriani dan *dataset* Q3 merupakan *dataset* dari topik Sentimen publik terhadap perubahan julukan Provinsi Bengkulu. Tabel 2 merupakan keseluruhan data yang berhasil dikumpulkan.

TABEL 2
DATASET MENTAH

Dataset	Dataset Instagram	Dataset Tiktok	Total Dataset
Q1	10. 232	2.857	13.089
Q2	1.190	3.727	4.917
Q3	2.412	2.470	4.882

B. Labeling

Hasil pelabelan data pada masing-masing *dataset* dapat dilihat pada tabel 3. Terlihat bahwa donimasi kelas positif terjadi pada *dataset* Q1 diikuti kelas negatif dan netral dengan nilai seimbang. Sementara itu, dominasi kelas negatif terjadi pada *dataset* Q2 dan Q3 diikuti kelas netral dan positif.

TABEL 3
DISTRIBUSI KELAS

Dataset	Kelas			Total
	Positif	Negatif	Netral	
Q1	693	378	339	1.410
Q2	151	688	334	1.173
Q3	123	887	368	1.378

1

2

3

4

4.1

4.1.1

C. Data Preprocessing

Proses *data preprocessing* merupakan proses untuk membersihkan serta mempersiapkan data sebelum dilakukan pemrosesan lebih lanjut. Teknik ini bertujuan untuk meningkatkan kualitas data sehingga lebih efektif dan layak untuk dianalisis. Terdapat beberapa tahapan dalam proses ini meliputi:

1

2

3

4

4.1

4.1.1

4.1.2

4.1.3

i.

1. Remove Punctuation

Remove punctuation merupakan proses pembersihan data komentar dari hal-hal yang tidak diperlukan seperti *mention @username*, tag, simbol, tanda baca, *tag HTML* dan bilangan *nonAmerican Standard Code of Information Interchange* (ASCII). Pembersihan juga dilakukan pada kata yang mendapat penambahan huruf yang sama (contohnya: *samoo, masoo, yakkk*) agar kata tersebut kembali kebentuk normal. Hal ini termasuk ke dalam kesalahan penulisan dan merupakan bagian dari proses normalisasi, namun guna mengefisiensi penghapusan dilakukan pada tahap ini dengan menghapus perulangan huruf lebih dari 3 kali. Pertimbangan ini mengingat terdapat kata dengan perulangan huruf 2 kali seperti pada kata "keadaan" dan "saat". Setelah dilakukan pembersihan-pembersihan tersebut perlu dilakukan penghilangkan *whitespace leading and triling* dan *multiple whitspace into single whitespace* guna membersihkan kelebihan spasi yang dapat disebabkan oleh pembersihan yang dilakukan sebelumnya. Kemudian dilakukan pengecekan serta penghapusan baris data apabila terdapat data kosong pada kolom komentar yang mungkin terjadi akibat proses ini.

2. Case Folding

Proses berikutnya merupakan teknik mengubah semua huruf kapital (*uppercase*) pada sebuah kata menjadi huruf kecil (*lowercase*) guna menyetarakan penggunaan huruf dan menghindari perbedaan makna yang disebabkan oleh penggunaan huruf besar atau kecil [19]. Komentar bahasa Melayu Bengkulu *alasan ganti Namo ny apo emg?* "alasan mengganti Namanya memang apa?" setelah *case folding* menjadi *alasan ganti namo ny apo emg?* "alasan mengganti namanya memang apa?".

3. Normalization

Pada proses ini dilakukan pengecekan ejaan pada kata guna mengetahui serta memperbaiki kesalahan pengejaan dalam komentar. Kesalahan tersebut meliputi:

a. *Disemoveling*, yaitu penghapusan satu atau semua fenom vokal (a, i, u, e, o) dalam sebuah kata. Contoh: *ngapo* menjadi *ngp*.

b. *Shortening*, yaitu penyingkatan kata dari bentuk asalnya yang mempengaruhi bentuk fonem. Contoh: *idak* menjadi *dak.*

c. *Space/dash removing*, yaitu penghilangan simbol spasi (-) atau penambahan huruf baru pada kata reduplikasi. Contoh: *samo-samo* menjadi *samo2*.

d. *Phonetic change*, yaitu perubahan bunyi fonem dan penulisan namun jumlah kata tetap sama. Contoh: *cubo* menjadi *cobo.*

e. *Afixasi informal*, yaitu perubahan, penambahan atau penghilangan imbuhan dalam kata. Contoh: *gegalo* menjadi *segalo*.

f. *Compounding/acronym*, yaitu penyingkatan dua atau lebih kata menjadi sebuah kata. Contoh: *sudem itu* menjadi *demtu* atau *idak usah* menjadi *daksah.*

g. *Reverse*, yaitu pembalikan urutan kata. Kesalahan ini tidak ditemukan di dalam *dataset*.

h. *Loan words*, yaitu peminjaman kata diluar bahasa Melayu Bengkulu. Contoh: amin dari bahasa Indonesia

i. Jargon, tagline kata yang menjadi popular di masyarakat. Contoh: camkoha dari macam iko ha (artinya: seperti ini lah"

4. Remove Stopword

Proses ini digunakan pada tahap *preprocessing* guna mengeliminasi kata-kata yang bersifat kurang informatif dari dalam teks [20]. Penghapusan kata ini berguna untuk mengurangi biaya komputasi dari dimensi komputasi

yang besar dan mengurangi penggunaan memori serta mempercepat waktu komputasi. Sebagaimana yang telah dijelaskan pada bab sebelumnya, saat penelitian ini dilakukan belum tersedia *stopword* bahasa Melayu Bengkulu. Dalam penelitian ini peneliti membangun list *stopword* bahasa Melayu Bengkulu berdasarkan *dataset* yang ada. Untuk mengefisiensi, proses pembuatan *stopword* dilakukan secara otomatis menggunakan teknik *Term-Based Random Sampling* (TRS). Teknik ini menerapkan algoritma perhitungan *kullback-laibler* untuk menilai kepentingan dari suatu term atau kata.

5. Stemming

Pada proses ini setiap kata akan dikembalikan ke bentuk dasarnya. Pengembalian tersebut meliputi penghilangan imbuhan atau *affix* dari dalam kata. Untuk mempermudah proses *stemming*, digunakan hasil pemrosesan sebelumnya dimana komentar yang telah melalui tahap *stopword* akan ditokenisasi untuk mendapatkan daftar kata unik dari *dataset*. Dari daftar tersebut akan dilakukan pengecekan untuk mengembalikan kata kebentuk dasarnya. Hal yang pertama kali dilakukan dalam proses *stemming* ialah menghapus partikel kata. Partikel kata merupakan kelas kata yang tidak dapat berdiri sendiri seperti partikel *-la, -ka* dan *-pun*. Kemudian dilakukan penghapusan kata ganti kepunyaan (*possesive pronoun*) seperti *-mu, -nyo*, dan *-ku*. Selanjutnya kata tersebut akan diidentifikasi untuk menemukan pemakaian imbuhan baik di awal kata (*prefix*) maupun di akhir kata (*suffix*). Berikut ini beberapa bentuk data yang telah melalui proses stemming yang dapat dilihat pada tabel 4.

TABEL 4
STEMMING DATA

Sebelum Stemming	Sesudah Stemming
inti**nyo** sapo bae jadi gubernur tolong anak asli daerah ini siap**kan** wadah kerjo supayo kami usa merantau jau untuk dapek	inti sapo bae jadi gubernur tolong anak asli daerah ini siap wadah kerjo supayo kami usa rantau jau untuk dapek
iyo nian apo din omong**an men**inggal**kan** daerah kalo rindu kek bini pakso pai ke jakarta	iyo nian apo din omong tinggal daerah kalo rindu kek bini pakso pai ke jakarta
mekak kini jelas**kan** ajo apo kerjo**kan** sampai harus mili kamu lagi	mekak kini jelas ajo apo kerjo sampai harus pili kamu lagi

6. Tokenization

Tokenization merupakan teknik membagi teks menjadi unit-unit kecil. Pada penelitian ini proses *tokenization* dilakukan menggunakan pustaka *Tensorflow.keras, preprocessing.text*. yang memungkinkan mengurai setiap kata dalam *dataset* pada kolom yang berisikan komentar. Langkah awal pada tahap ini ialah membangun indeks kata, yaitu sebuah list yang berisikan setiap kata yang ada pada kolom '*comment_stemming'*. Dengan menggunakan Pustaka *Tokenizer*, list kata tersebut akan diberikan token berupa angka terurut dan dimulai dari angka satu.

Pada saat penggunaan model seringkali terdapat kata diluar dari indeks kata, sehingga kata tersebut tidak dapat diubah dalam bentuk token angka. Untuk menangani hal ini dibuatkan sebuah token khusus yang akan menjadi token bagi setiap kata diluar dari indeks kata. Sebuah kata diluar indeks kata disebut sebagai <OOV> (*out of vocabulary*) dan ditempatkan pada indeks pertama dalam indeks kata dengan token angka bernomor satu.

Setelah mendapatkan indeks kata, langkah berikutnya ialah menyusun kembali rangkaian kata menjadi sebuah kalimat utuh. Token angka dari indeks kata akan menggantikan kata dalam sebuah kalimat sehingga data teks akan diubah kedalam bentuk vektor dalam sebuah *array*. Langkah ini disebut sebagai *Sequences*.

Dalam pemrosesan model *deep learning* dibutuhkan *input* dengan panjang tetap, sementara itu data *array* memiliki panjang yang berbeda, sesuai dengan panjang kalimat awal. Untuk menyamakan panjang setiap vektor di dalam *array* digunakan teknik *padding* dan *truncating*. Teknik *padding* digunakan untuk menambah angka 0 (nol) diakhir vektor hingga panjang vektor sama dengan panjang maksimum *input*-an. Di sisi lain Teknik *truncating* digunakan untuk menangani vektor dengan panjang melebihi *input*-an dengan memangkas kelebihan vektor.

D. Word Embedding

Word embedding merupakan teknik representasi kata dalam vektor numerik yang mampu menangkap kemiripan dan hubungan semantik antar kata dalam *dataset*. Teknik *word embedding* yang digunakan pada penelitian ini ialah Word2Vec dengan model arsitektur *Skipgram*. Teknik ini bekerja dengan memanfaatkan informasi dari kata target dan memprediksi kata-kata di sekitarnya dalam sebuah jendela (*window*) konteks tertentu. Kata yang sering digunakan dalam konteks yang sama akan menciptakan kolerasi antar kata dengan bobot yang mirip.

Ada beberapa parameter yang digunakan dalam membangun model Word2Vec. Minimum count ditetapkan sebesar 3 agar kata yang sangat jarang muncul tidak mendominasi ruang vektor. Window size ditetapkan sebesar 3 karena komentar media sosial cenderung pendek, sehingga konteks lokal di sekitar kata target lebih relevan untuk menangkap hubungan semantik. Ukuran vektor ditetapkan sebesar 300 dimensi agar representasi kata cukup kaya untuk menangkap variasi kosakata Melayu Bengkulu. Arsitektur Skip-gram digunakan karena lebih sesuai untuk mempelajari kata yang relatif jarang muncul pada korpus kecil dan bahasa dengan sumber daya terbatas. Negative

sampling tidak digunakan, sehingga pelatihan memakai Hierarchical Softmax. Pemilihan parameter ini mengacu pada penelitian Word2Vec sebelumnya dan disesuaikan dengan ukuran serta karakteristik dataset penelitian [21], [22].

a. TABEL 5
Uji Cosine Similarity

Dataset	**Kata A**	**Kata B**	**Nilai Kesamaan**
Q1	Jalan	hibrida	0.7223
Q2	ruba	saro	0.7955
Q3	bungo	langka	0.74015

Hasil uji cosine similarity pada Tabel 5 menunjukkan dalam dataset Q1, kata 'jalan' memiliki nilai kemiripan paling tinggi dengan kata 'hibrida sebesar 0.7223. Sementara pada dataset Q2 kata 'ruba' memiliki nilai kemiripan paling tinggi dengan kata 'saro'. Kemudian kata 'bungo' dan kata 'langka' pada dataset Q3 memiliki nilai kemiripan 0.74015. Nilai-nilai yang mendekati satu ini mengartikan Tingkat korelasi yang semakin tinggi antar kedua kata.

E. Splitting

Dalam penelitian ini dataset dibagi ke dalam 80% data pelatihan dan 20% data pengujian. Selanjutnya, 20% data pelatihan dialokasikan sebagai data validasi. Pembagian ini digunakan agar model memiliki data yang cukup untuk proses pelatihan, validasi selama training, dan evaluasi akhir pada data yang tidak dilihat model [23] dan [24]. Pembagian data dilakukan secara stratified split agar proporsi kelas positif, negatif, dan netral tetap terjaga pada train, validation, dan test set. Distribusi pembagian data disajikan pada tabel 6. Penanganan ketidakseimbangan kelas belum menggunakan teknik resampling atau class weighting. Oleh karena itu, pengaruh imbalance dianalisis pada bagian hasil dan pembahasan sebagai salah satu faktor yang memengaruhi performa model.

b. TABEL 6
Splitting Data

Dataset	*Train*	*Test*	*Validation*
Q1	912	285	228
Q2	789	247	198
Q3	884	277	222

F. Modeling dan Evalution

Pemodelan dalam penelitian ini tiga model yaitu BiLSTM, CNN, dan LSTM dapat dilihat berdasarkan confusion matrix. Selain itu, dilakukan evaluasi untuk mendapatkan nilai performansi model.

1. Model BiLSTM

Hasil evaluasi model LSTM menggunakan tiga *confusion matrix* untuk dataset Q1, Q2, dan Q3 terlihat pada gambar 5. Pada *confusion matrix* untuk dataset Q1, model menunjukkan performa klasifikasi yang relatif optimal pada seluruh kelas. Pada kelas negatif, sebagian besar data aktual berhasil diprediksi dengan benar, dengan jumlah kesalahan klasifikasi yang relatif kecil dan tersebar ke kelas netral dan kelas positif. Kondisi ini menunjukkan bahwa model telah mampu menangkap karakteristik pola data kelas negatif secara representatif. Pola serupa juga terlihat pada kelas netral, dimana dominasi prediksi benar menunjukkan bahwa model memiliki kemampuan pemisahan fitur yang baik terhadap kelas lainnya. Sementara itu, kelas positif menunjukkan performa paling tinggi dibandingkan kelas lainnya, yang ditunjukkan oleh dominasi prediksi benar dan kesalahan klasifikasi yang sangat minimal. Secara keseluruhan, kondisi ini mengindikasikan bahwa model memiliki kemampuan generalisasi yang baik pada distribusi data pengujian pertama.

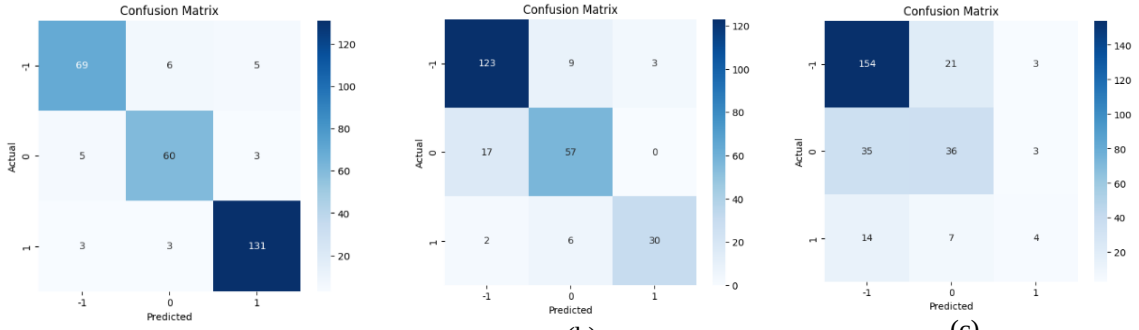
Pada *confusion matrix* untuk dataset Q2, terjadi perubahan pola performa model. Kelas negatif menunjukkan peningkatan kemampuan klasifikasi yang signifikan dengan jumlah prediksi benar yang sangat dominan. Hal ini mengindikasikan bahwa model semakin sensitif terhadap karakteristik fitur kelas negatif. Namun demikian, pada kelas netral mulai terlihat adanya penurunan performa yang ditandai dengan meningkatnya kesalahan klasifikasi ke kelas negatif. Fenomena ini menunjukkan adanya kemungkinan kemiripan distribusi fitur antara kelas netral dan kelas negatif, sehingga model mengalami kesulitan dalam membangun batas keputusan yang optimal. Pada kelas positif, model masih menunjukkan performa yang cukup baik meskipun terdapat sedikit peningkatan kesalahan klasifikasi dibandingkan *confusion matrix* dataset Q1. Kondisi ini menunjukkan adanya indikasi awal ketidakseimbangan kemampuan klasifikasi model antar kelas.

Pada *confusion matrix* untuk dataset Q3, kecenderungan bias model terhadap kelas negatif semakin terlihat jelas. Kelas negatif tetap menunjukkan dominasi prediksi benar yang sangat tinggi, meskipun terjadi peningkatan kesalahan klasifikasi ke kelas netral. Pada sisi lain, kelas netral menunjukkan penurunan performa yang cukup signifikan, dimana jumlah prediksi benar menjadi hampir sebanding dengan jumlah kesalahan klasifikasi ke kelas negatif. Kondisi ini menunjukkan bahwa model mengalami kesulitan dalam membedakan karakteristik fitur antara

kedua kelas tersebut. Penurunan performa paling signifikan terjadi pada kelas positif, dimana jumlah prediksi benar menjadi sangat rendah dibandingkan jumlah kesalahan klasifikasi.

Hal ini mengindikasikan bahwa model kehilangan kemampuan dalam mengenali karakteristik spesifik kelas positif, yang kemungkinan dipengaruhi oleh ketidakseimbangan jumlah data, tingginya tingkat noise data, atau tingginya overlap fitur dengan kelas lain.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa model BiLSTM memiliki performa yang sangat baik pada kondisi data yang relatif seimbang dan memiliki separabilitas fitur yang jelas. Namun, pada kondisi distribusi data yang lebih kompleks, model menunjukkan kecenderungan bias terhadap kelas mayoritas, yang berdampak pada penurunan performa klasifikasi pada kelas minoritas. Fenomena ini mengindikasikan bahwa stabilitas performa model sangat dipengaruhi oleh distribusi data pelatihan dan tingkat kemiripan fitur antar kelas.



(a) (b) (c)

Gambar 4. *Confusion matrix* dataset: (a) Q1; (b) Q2; (c) Q3

Perolehan akurasi untuk setiap dataset mencapai 91% pada dataset Q1, 85% pada dataset Q2, dan 70% pada dataset Q3. Berdasarkan hasil *classification report*, terlihat bahwa *dataset* Q1 sudah menunjukkan performa unggul diikuti oleh Q2. Sementara itu, pada dataset Q3, model belum dapat menemukan serta memprediksi data di setiap kelas dengan benar. Hal ini mempengaruhi rendahnya nilai *precision* dan *recall* untuk dataset ini. Hasil evaluasi model BiLSTM dapat dilihat pada tabel 7 *Classification report.*

TABEL 7
CLASSIFICATION REPORT

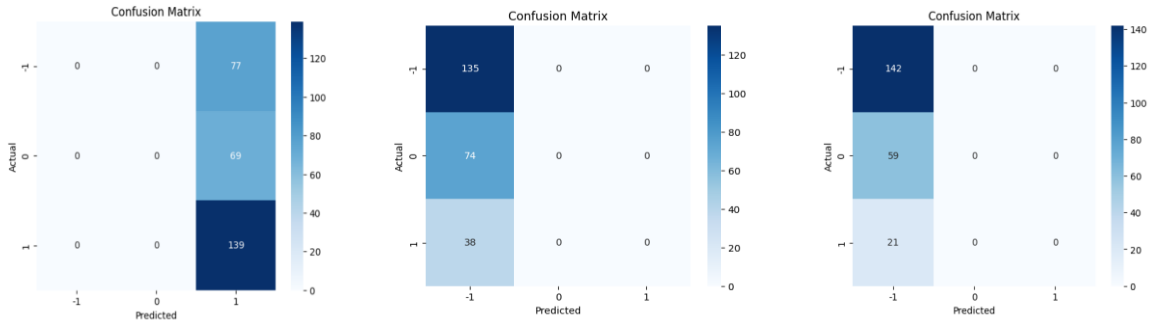
Dataset	Precision	Recall	F1-Score	Accuracy
Q1	90	90	90	91
Q2	86	82	84	85
Q3	57	50	52	70

2. LSTM

Hasil evaluasi model LSTM menggunakan tiga *confusion matrix* untuk dataset Q1, Q2, dan Q3 terlihat pada gambar 5. Pada *confusion matrix* untuk dataset Q1, model menunjukkan kecenderungan yang sangat kuat dalam memprediksi seluruh data ke dalam kelas positif. Hal ini terlihat dari distribusi prediksi dimana sebanyak 77 data dari kelas aktual negatif, 69 data dari kelas aktual netral, dan 139 data dari kelas aktual positif diprediksi sebagai kelas positif. Kondisi ini menunjukkan bahwa model memiliki kemampuan yang baik dalam mengenali pola kelas positif, namun tidak mampu membedakan karakteristik kelas negatif dan netral. Fenomena ini mengindikasikan bahwa model mengalami bias terhadap kelas tertentu yang kemungkinan disebabkan oleh distribusi data pelatihan yang tidak seimbang.

Pada *confusion matrix* untuk dataset Q2, terjadi perubahan pola prediksi dimana model cenderung mengklasifikasikan seluruh data ke dalam kelas negatif. Hal ini ditunjukkan dengan jumlah prediksi benar pada kelas negatif sebanyak 135 data. Namun, seluruh data dari kelas netral dan kelas positif juga salah diklasifikasikan ke dalam kelas negatif. Hal ini menunjukkan bahwa model belum mampu mempelajari representasi fitur yang membedakan antar kelas secara efektif. Selain itu, kondisi ini juga mengindikasikan adanya kemungkinan ketidakstabilan model selama proses pelatihan.

Hasil pada *confusion matrix* untuk dataset Q3 menunjukkan pola yang relatif sama dengan *confusion matrix* dataset Q2. Model tetap dominan memprediksi kelas negatif dengan jumlah prediksi benar sebesar 142 data, sementara sebanyak 59 data dari kelas netral dan 21 data dari kelas positif salah diklasifikasikan sebagai kelas negatif. Konsistensi pola ini menunjukkan bahwa model mengalami fenomena yang dikenal sebagai *mode collapse*, yaitu kondisi dimana model hanya memprediksi satu kelas dominan tanpa mempertimbangkan variasi kelas lainnya.



(a) (b) (c)

Gambar 5.*Confusion matrix* dataset: (a) Q1; (b) Q2; (c) Q3

Perolehan akurasi untuk setiap dataset mencapai 49% pada dataset Q1, 55% pada dataset Q2, dan 64% pada dataset Q3. Berdasarkan hasil *classification report*, performa model LSTM menunjukkan tren peningkatan dari dataset Q1 ke Q3. Peningkatan paling signifikan terjadi pada dataset Q3, terutama pada nilai *recall* dan *F1-score.* Namun demikian, nilai *precision* yang masih rendah menunjukkan bahwa model masih perlu dioptimalkan untuk mengurangi kesalahan prediksi dan meningkatkan ketepatan klasifikasi. Hasil evaluasi model LSTM dapat dilihat pada tabel 8 *Classification report.*

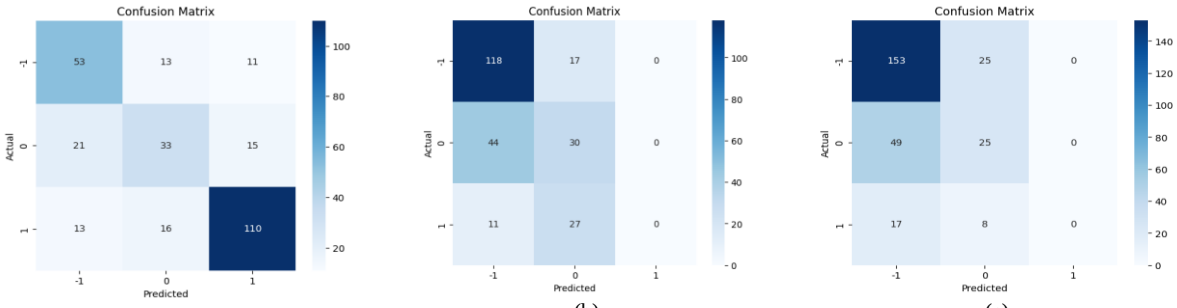
TABEL 8
CLASSIFICATION REPORT

Dataset	Precision	Recall	F1-Score	Accuracy
Q1	16	33	22	49
Q2	18	33	24	55
Q3	21	64	50	64

3. CNN

Berdasarkan hasil *confusion matrix* yang dihasilkan oleh model *Convolutional Neural Network* (CNN), dapat dilihat perkembangan performa model dalam mengklasifikasikan tiga kelas, yaitu kelas negatif, netral, dan positif. Pada *confusion matrix* untuk dataset Q1, model menunjukkan performa yang sudah sangat baik dan seimbang dalam mengenali ketiga kelas. Pada *confusion matrix* untuk dataset Q2, performa model masih menunjukkan kondisi menengah. Model sudah cukup baik dalam mengenali kelas negatif dan mulai menunjukkan peningkatan pada kelas netral. Namun, model masih belum mampu mengenali kelas positif dengan baik karena seluruh data pada kelas tersebut masih salah diklasifikasikan. Hal ini menunjukkan bahwa pembelajaran model masih belum merata untuk semua kelas dan masih terdapat bias terhadap kelas tertentu.

Pada *confusion matrix* untuk dataset Q3, model menunjukkan performa yang masih terbatas. Model mampu mengenali kelas negatif dengan cukup baik, namun masih sering salah dalam membedakan kelas netral karena banyak data kelas netral diprediksi sebagai kelas negatif. Selain itu, model sama sekali belum mampu mengenali kelas positif. Kondisi ini menunjukkan bahwa pada tahap ini model masih mengalami bias terhadap kelas mayoritas dan belum mampu mempelajari pola pada kelas minoritas.



(a) (b) (c)

Gambar 6. *Confusion matrix* dataset: (a) Q1; (b) Q2; (c) Q3

Perolehan akurasi untuk setiap dataset mencapai 69% pada dataset Q1, 60% pada dataset Q2, dan 64% pada dataset Q3. Berdasarkan hasil *classification report*, terlihat bahwa performa model terbaik terdapat pada dataset Q1, diikuti oleh Q3, dan yang terendah pada Q2. Hal ini menunjukkan bahwa model CNN kemungkinan bekerja optimal pada data dengan karakteristik seperti Q1, namun mengalami penurunan performa pada data dengan variasi atau kompleksitas seperti pada Q2. Dataset Q3 menunjukkan bahwa model mulai mampu beradaptasi kembali, terutama

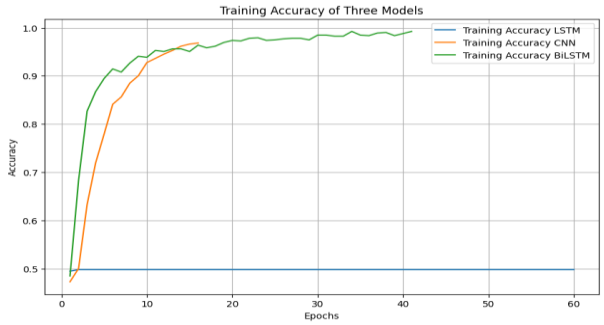
dalam meningkatkan kemampuan mendeteksi data yang benar (*recall*), meskipun masih perlu peningkatan pada ketepatan prediksi *(precision)*. Hasil evaluasi model CNN ditunjukkan pada tabel 9 *Classification report.*

TABEL 9
CLASSIFICATION REPORT

Dataset	Precision	Recall	F1-Score	Accuracy
Q1	65	65	65	69
Q2	36	43	39	60
Q3	38	64	60	64

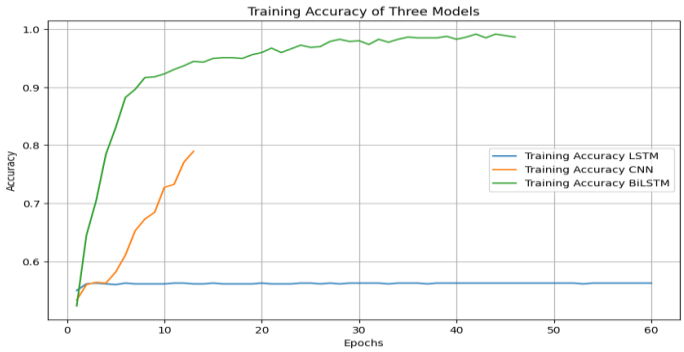
4. Perbandingan Performa BiLSTM, LSTM, dan CNN

Adapun perbandingan performa BiLSTM, LSTM, dan CNN dapat dilihat pada gambar 7,8, dan 9 grafik pelatihan model BiLSTM, LSTM dan CNN untuk setiap dataset. Selain itu, dilihat juga berdasarkan akurasi test model yang dapat dilihat pada tabel 12.



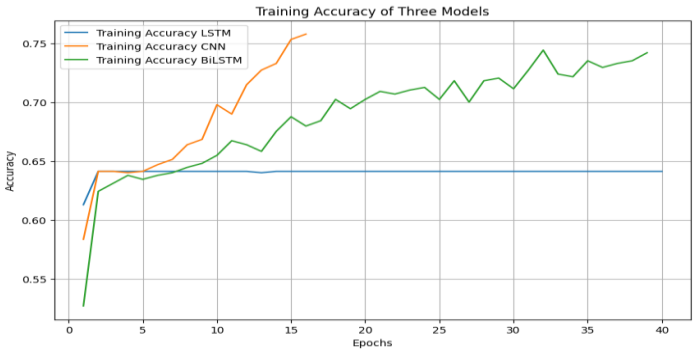
Gambar 7.Grafik Pelatihan model CNN, LSTM, dan BiLSTM Pada Dataset Q1

Grafik pada Gambar 7 memperlihatkan perbedaan grafik pelatihan data train pada dataset Q1. Grafik berwarna hijau menunjukkan perubahan kinerja model BiLSTM. Grafik berwarna oren menunjukkan perubahan kinerja model CNN. Sementara itu grafik berwarna biru menunjukkan perubahan kinerja model LSTM. Ketiganya memperlihatkan performa yang berbeda-beda. Ditinjau dari perubahan *accuracy* setiap epoch, model BiLSTM mengungguli kedua model dengan capaian akurasi hingga 99%. Model CNN juga memperlihatkan performa yang cukup baik dengan capaian akurasi 96%. Namun, di sisi lain model LSTM menunjukkan performa terburuk yang memperlihatkan tidak adanya perubahan accuracy pada model. Tidak adanya perubahan ini memperlihatkan bahwa model LSTM tidak mempelajari pola-pola pada dataset Q1.



Gambar 8. Grafik Pelatihan model CNN, LSTM, dan BiLSTM Pada Dataset Q2

Pada dataset Q2 Gambar 8, model BiLSTM masih menunjukkan performa terbaiknya dibandingkan kedua model lainnya dengan capaian akurasi 98%. Model CNN memperoleh akurasi yang jauh lebih rendah yaitu hanya 78% dan berhenti paling cepat, pada epoch 13 setelah peningkatan tajam. Sementara itu, model LSTM masih menunjukkan performa yang serupa dengan dataset Q1. Dimana model tidak mengalami perubahan accuracy hingga akhir epoch, menunjukkan model tidak mempelajari pola data pada dataset Q2.



Gambar 9. Grafik Pelatihan model CNN, LSTM, dan BiLSTM Pada Dataset Q3

Grafik pada Gambar 9 menunjukkan kinerja model pada dataset Q3. Tampak terjadi fluktuasi pada model BiLSTM dan CNN yang menandakan model belum mencapai stabilitas. Pada dataset Q3, model CNN mendapatkan akurasi yang lebih tinggi dengan epoch yang sedikit yaitu senilai 76%. Sementara model BiLSTM hanya mencapai 74% dengan jumlah epoch yang lebih banyak. Berlainan dengan itu, grafik LSTM tidak menunjukkan perubahan accuracy setelah epoch ke-3. Hal ini menunjukkan model LSTM tidak mampu mempelajari pola-pola data setelah beberapa epoch awal. Akurasi setiap model dapat dilihat pada Tabel 10.

TABEL 10
AKURASI TEST SETIAP MODEL

LSTM	LSTM	CNN	BiLSTM
Q1	49	69	91
Q2	55	60	85
Q3	64	64	70

Perbandingan performa ketiga model juga dievaluasi melalui hasil akurasi testing untuk setiap model dan dataset. Tampak bahwa akurasi tertinggi diraih oleh model BiLSTM pada dataset Q1 dengan nilai akurasi test mencapai 91%. Sementara itu, pada dataset yang sama model LSTM memperoleh akurasi paling rendah yang hanya mencapai 49% dan model CNN memperoleh akurasi 69%. Pada dataset Q2 memperlihatkan hasil yang sama, dimana perolehan akurasi tertinggi ada pada model BiLSTM senilai 85%, diikuti model CNN 60% dan terburuk ada pada model LSTM dengan akurasi 55%. Begitu pula pada dataset Q3. Akurasi test model BiLSTM mengungguli kedua model lainnya dengan capaian akurasi 70%. Sementara model CNN dan LSTM memperoleh akurasi yang sama di angka 64%. Secara keseluruhan dapat diketahui bahwa penggunaan model BiLSTM pada ketiga dataset mengungguli model CNN dan LSTM dengan perbedaan nilai akurasi yang cukup jauh. Dengan demikian, ini menunjukkan bahwa model BiLSTM lebih baik digunakan dalam melakukan analisis sentimen komentar berbahasa Melayu Bengkulu.

a. Analisis ketidakseimbangan kelas

Distribusi kelas pada Tabel 3 menunjukkan bahwa dataset Q1 relatif lebih seimbang dibandingkan Q2 dan Q3. Pada Q1, rasio kelas mayoritas terhadap minoritas adalah 693:339 atau sekitar 2,04. Pada Q2, rasio kelas negatif terhadap positif meningkat menjadi 688:151 atau sekitar 4,56. Pada Q3, rasio kelas negatif terhadap positif mencapai 887:123 atau sekitar 7,21. Peningkatan imbalance ini menjelaskan mengapa performa model menurun pada Q3, terutama untuk kelas positif. Model cenderung lebih mudah mempelajari pola kelas mayoritas, sedangkan kelas minoritas memiliki representasi fitur yang lebih sedikit selama proses pelatihan.

b. Faktor Keunggulan BiLSTM

BiLSTM menghasilkan performa terbaik karena arsitektur ini memproses urutan kata dari dua arah. Pada komentar berbahasa Melayu Bengkulu, makna sentimen sering bergantung pada kombinasi kata sebelum dan sesudah kata kunci. Misalnya, kata yang secara leksikal terlihat netral dapat berubah menjadi positif atau negatif ketika muncul bersama penanda dukungan, sindiran, atau negasi lokal. CNN lebih kuat menangkap fitur lokal, tetapi kurang optimal dalam memahami dependensi sekuensial yang lebih panjang. LSTM membaca konteks satu arah, sehingga lebih rentan kehilangan informasi setelah kalimat memiliki variasi struktur. BiLSTM lebih stabil karena menggabungkan konteks maju dan mundur.

c. Comparative Analysis dengan Penelitian Terdahulu

Hasil penelitian ini sejalan dengan studi yang menunjukkan bahwa BiLSTM mampu memberikan performa kuat pada tugas klasifikasi teks karena dapat menangkap konteks dua arah [7], [24], [13]. Temuan ini juga mendukung penelitian pada Word2Vec yang menegaskan bahwa kualitas representasi kata memengaruhi

performa *deep learning* dalam klasifikasi sentimen [10]. Jika dibandingkan dengan penelitian berbasis Transformer [12], seperti IndoBERT dan IndoRoBERTa, akurasi tertinggi penelitian ini pada Q1 sebesar 91% masih berada di bawah beberapa studi Transformer berbahasa Indonesia, tetapi konteks penelitian ini berbeda karena menggunakan bahasa daerah dengan data terbatas [12], [25]. Perbedaan tersebut menunjukkan bahwa ketersediaan korpus, imbalance, dan karakteristik bahasa lokal berpengaruh besar terhadap capaian model.

d. Pengujian Statistik

Untuk memberikan gambaran signifikansi perbedaan performa, dilakukan uji Friedman terhadap akurasi CNN, LSTM, dan BiLSTM pada tiga dataset. Hasil uji menunjukkan nilai chi-square sebesar 5,64 dengan p-value 0,060. Nilai ini menunjukkan adanya kecenderungan perbedaan performa antar model, tetapi belum signifikan pada taraf 0,05. Hasil ini perlu ditafsirkan secara hati-hati karena jumlah dataset yang diuji masih kecil. Penelitian lanjutan disarankan menggunakan *k-fold cross-validation* agar pengujian statistik memiliki dasar yang lebih kuat.

e. Kelebihan, Keterbatasan, dan Implikasi

Kelebihan penelitian ini terletak pada penggunaan dataset komentar berbahasa Melayu Bengkulu, *preprocessing* khusus bahasa lokal, dan perbandingan tiga model *deep learning* pada tiga topik berbeda. Keterbatasannya meliputi ketidakseimbangan kelas, jumlah data berlabel yang masih terbatas, belum digunakannya teknik *resampling* atau *class weighting*, serta belum adanya model berbasis Transformer sebagai pembanding. Dari sisi praktis, hasil penelitian ini dapat menjadi baseline awal untuk pengembangan sistem monitoring opini publik lokal, analisis layanan publik daerah, dan pengembangan sumber daya NLP bahasa daerah Indonesia. Dari sisi akademik, penelitian ini memperkuat urgensi pembangunan korpus dan model NLP untuk bahasa daerah yang masih *under-resourced*.

IV. SIMPULAN

Berdasarkan hasil pengujian pada tiga dataset komentar berbahasa Melayu Bengkulu, BiLSTM menjadi model dengan performa terbaik dibandingkan CNN dan LSTM. BiLSTM memperoleh akurasi 91% pada Q1, 85% pada Q2, dan 70% pada Q3. CNN memperoleh akurasi 69%, 60%, dan 64%, sedangkan LSTM memperoleh akurasi 49%, 55%, dan 64%. Keunggulan BiLSTM terjadi karena model ini mampu membaca konteks kata dari arah maju dan mundur, sehingga lebih efektif menangkap pola sentimen pada komentar pendek, informal, dan kontekstual. Meski demikian, performa seluruh model menurun pada dataset dengan ketidakseimbangan kelas yang lebih tinggi, terutama Q3. Penelitian ini berkontribusi sebagai baseline awal analisis sentimen untuk bahasa Melayu Bengkulu dan mendukung pengembangan NLP bahasa daerah Indonesia. Keterbatasan penelitian ini adalah jumlah data berlabel yang masih terbatas, distribusi kelas yang tidak seimbang, belum diterapkannya *resampling* atau *class weighting*, serta belum adanya pembanding berbasis Transformer. Penelitian selanjutnya disarankan menggunakan *k-fold cross-validation*, memperbesar korpus Melayu Bengkulu, menerapkan penanganan *imbalance* seperti SMOTE, *Random Over Sampling*, atau *class weight*, serta membandingkan model *deep learning* dengan Transformer seperti IndoBERT, IndoBERTweet, atau model multibahasa berbasis BERT.

UCAPAN TERIMAKASIH

Kami mengucapkan terima kasih kepada Fakultas Teknik Universitas Bengkulu atas dukungan dan pendanaan yang telah diberikan melalui Hibah Penelitian Skema Unggulan FT UNIB Tahun Anggaran 2025.

DAFTAR PUSTAKA

[1] T. S. & Rita, "Jurnal Pendidikan Bahasa," *Pendidik. Bhs. dan Sastra Indones.*, vol. 2, no. 1, pp. 67–73, 2017.
[2] A. I. Pradana and Wijiyanto, "G-Tech : Jurnal Teknologi Terapan," *G-Tech J. Teknol. Terap.*, vol. 8, no. 1, pp. 502–511, 2024.
[3] R. Al Kiramy, I. Permana, and A. Marsal, "Comparison of RNN and LSTM Algorithm Performance in Predicting the Number of Umrah Pilgrims at PT . Hajar Aswad Perbandingan Performa Algoritma RNN dan LSTM dalam Prediksi Jumlah Jamaah Umrah pada PT . Hajar Aswad," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. October, pp. 1224–1234, 2024.
[4] H. Awarulloh, F. Shiddieq, and D. Nurhayati, "Penggunaan Multivariat Model Bidirectional LSTM untuk Prediksi Cuaca: Optimalisasi Waktu Tanam Padi Petani Kabupaten Garut," *J. Ilm. Sinus (JIS*, vol. 23, no. 1, p. 127, 2025.
[5] P. O. A. Sunarya, R. Refianti, A. B. Mutiara, and W. Octaviani, "Comparison of accuracy between convolutional neural networks and Naïve Bayes Classifiers in sentiment analysis on Twitter," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 5, pp. 77–86, 2019, doi: 10.14569/ijacsa.2019.0100511.
[6] F. P. Pratama, Indriati, and A. Ridok, "Perbandingan metode indoBERT Dengan CNN untuk analisis sentimen berbasis aspek terhadap ulasan pelanggan (studi kasus : hotel indonesia kempinski jakarta pada website travel agent tiket.com)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 1, pp. 1–10, 2025.
[7] N. R. Ramadhan and N. Hendrastuty, "Perbandingan Algoritma Naïve Bayes dan LSTM untuk Analisis Sentimen Terhadap Opini Masyarakat Tentang Sandwich Generation," *Build. Informatics, Technol. Sci.*, vol. 6, no. 3, pp. 1677–1687, 2024, doi: 10.47065/bits.v6i3.6385.
[8] M. R. F. Kamarula and N. Rochmawati, "Perbandingan CNN dan Bi-LSTM pada Analisis Sentimen dan Emosi Masyarakat Indonesia Di Media Sosial Twitter Selama Pandemik Covid-19 yang Menggunakan Metode Word2vec," *J. Informatics Comput. Sci.*, vol. 04, pp.

219–228, 2022, doi: 10.26740/jinacs.v4n02.p219-228.
[9] R. Nursyanti, N. Alamnsyah, and T. M. Yusuf, "Volume 7 Issue 1 Aisyah Journal of Informatics and Electrical Engineering PENERAPAN APLIKASI CHATTING UNTUK KENDALI LAMPU Aisyah Journal of Informatics and Electrical Engineering Aisyah Journal of Informatics and Electrical Engineering," *Aisyah J. Informatics Electr. Eng.*, vol. 7, no. 1, pp. 9–14, 2025.
[10] D. I. Af'idah, D. Dairoh, S. F. Handayani, and R. W. Pratiwi, "Pengaruh Parameter Word2Vec terhadap Performa Deep Learning pada Klasifikasi Sentimen," *J. Inform. J. Pengemb. IT*, vol. 6, no. 3, pp. 156–161, 2021, doi: 10.30591/jpit.v6i3.3016.
[11] Y. Romadhoni and K. F. H. Holle, "Analisis Sentimen Terhadap PERMENDIKBUD No.30 pada Media Sosial Twitter Menggunakan Metode Naive Bayes dan LSTM," *J. Inform. J. Pengemb. IT*, vol. 7, no. 2, pp. 118–124, 2022, doi: 10.30591/jpit.v7i2.3191.
[12] F. M. Apriansyah, T. I. Ramadhan, C. R. Hidayat, and A. K. Wijaya, "Perbandingan IndoBERT dan IndoRoBERTa Untuk Analisis Sentimen Pada Film Dokumenter Dirty Vote," *J. Inform. J. Pengemb. IT*, vol. 10, no. 3, pp. 593–605, 2025, doi: 10.30591/jpit.v10i3.8607.
[13] M. W. A. Pramana, D. P. S. Putri, and I. K. A. Purnawan, "Comparison of IndoBERT and Bi-LSTM Models for Indonesian Law Violation Text Classification," *J. Inform. J. Pengemb. IT*, vol. 10, no. 4, pp. 1033–1043, 2025, doi: 10.30591/jpit.v10i4.8795.
[14] R. F. Chandra and D. Andini, "Implementasi Metode Naïve Bayes Pada Ulasan Pengguna Aplikasi Dana di Google Play Store," *J. Infortech*, vol. 7, no. 1, pp. 64–69, 2025, doi: 10.31294/infortech.v7i1.25977.
[15] D. Wijaya, R. A. Saputra, and F. Irwiensyah, "Analisis Sentimen Ulasan Aplikasi Samsat Digital Nasional Pada Google Playstore Menggunakan Algoritma Naïve Bayes," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 4, pp. 2369–2380, 2024, [Online]. Available: https://djournals.com/klik/article/view/1738
[16] K. F. H. Holle, R. Alfianita, and H. M. Putri, "Evaluasi Teknik Preprocessing terhadap Kinerja Multinomial Naive Bayes dalam Klasifikasi Pertanyaan Insincere," *JUSTIN (Jurnal Sist. dan Teknol. Informasi)*, vol. 12, no. 4, pp. 707–715, 2024, doi: 10.26418/justin.v12i4.82758.
[17] M. Haris, A. Suharso, E. H. Nurkifli, P. S. Informatika, U. S. Karawang, and T. Timur, "ANALISIS SENTIMEN PADA GAME EFOOTBALL DI GOOGLE PLAY STORE MENGGUNAKAN ALGORITMA INDOBERT," vol. 8, no. 6, pp. 12108–12121, 2024.
[18] J. Ipmawati, S. Saifulloh, and K. Kusnawi, "Analisis Sentimen Tempat Wisata Berdasarkan Ulasan pada Google Maps Menggunakan Algoritma Support Vector Machine: Sentiment Analysis of Tourist Attractions Based on Reviews on Google Maps Using the Support Vector Machine Algorithm," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 1, pp. 247–256, 2024.
[19] D. Rifaldi and A. Fadlil, "DECODE: Jurnal Pendidikan Teknologi Informasi TEKNIK PREPROCESSING PADA TEXT MINING MENGGUNAKAN DATA TWEET 'MENTAL HEALTH,'" *Decod. J. Pendidik. Teknol. Inf.*, vol. 3, no. 2, pp. 161–171, 2023.
[20] S. J. Angelina, A. Bijaksana, P. Negara, and H. Muhardi, "Analisis Pengaruh Penerapan Stopword Removal Pada Performa Klasifikasi Sentimen Tweet Bahasa Indonesia," *JUARA (Jurnal Apl. dan Ris. Inform.*, vol. 02, no. 1, pp. 165–173, 2023, doi: 10.26418/juara.v2i1.69680.
[21] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, "Sentiment Analysis Using Word2vec And Long Short-Term Memory (LSTM) For Indonesian Hotel Reviews," *Procedia Comput. Sci.*, vol. 179, pp. 728–735, 2021, doi: 10.1016/j.procs.2021.01.061.
[22] H. Wen and J. Zhao, "Sentiment Analysis Model of Imbalanced Comment Texts Based on BiLSTM," Jan. 2023. doi: 10.21203/rs.3.rs-2434519/v1.
[23] H. Bichri, A. Chergui, and M. Hain, "Investigating the Impact of Train / Test Split Ratio on the Performance of Pre-Trained Models with Custom Datasets," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 2, 2024, doi: 10.14569/IJACSA.2024.0150235.
[24] H. Ma'we, A. Y. Husodo, and B. Irmawati, "Performance Comparison of Naive Bayes and Bidirectional Lstm Algorithms in Bsi Mobile Review Sentiment Analysis," *J. Tek. Inform.*, vol. 6, no. 1, pp. 159–172, 2024, doi: 10.52436/1.jutif.2024.5.6.4178.
[25] F. Al, F. Ruwanto, and L. B. Handoko, "Analisis Sentimen Komentar YouTube pada Channel Edukasi Otomotif menggunakan IndoBERT," *J. Inform. J. Pengemb. IT*, vol. 11, no. 2, pp. 284–295, 2026, doi: 10.30591/jpit.v11i2.10216.