SKANIKA: Sistem Komputer dan Teknik Informatika
Volume 8, Nomor 2, Juli 2025, Halaman 352-363 E-ISSN: 2721-4788

Implementasi Arsitektur *Recurrent Neural Network* Pada Analisis Sentimen Clash of Champions

Arif Hidayat[1*], Anindita Septiarini[2], Medi Taruk[3]
[1,2,3]Fakultas Teknik, Informatika, Universitas Mulawarman, Samarinda, Indonesia
E-mail: [1]arifhidayat@student.unmul.ac.id, [2]anindita@unmul.ac.id, [3]meditaruk@unmul.ac.id
(* : corresponding author

Abstrak

Clash of Champions adalah Program edukatif Ruangguru di YouTube yang mendapat respons beragam. Penelitian ini bertujuan untuk melakukan analisis sentiment dengan tiga arsitektur *Recurrent Neural Network* (RNN), yaitu *Vanilla Recurrent Neural* Network (Vanilla RNN), *Long Short-Term Memory* (LSTM), dan *Gated Recurrent Unit* (GRU). Data mencakup 2.100 data pelatihan, 300 validasi, 600 pengujian yang didapatkan dari Youtube dan diperkaya dengan data augmentasi menggunakan teknologi GPT-4, sedangkan data 35 komentar berasal dari survei dengan Google Form untuk uji generalisasi. Komentar diklasifikasikan dalam tiga sentimen yaitu Pro, Netral, dan Kontra. Analisis melibatkan *preprocessing*, pelatihan model, dan evaluasi dengan metrik standar. GRU menunjukkan performa terbaik dengan akurasi 99,2% dan skor F1 tertinggi. LSTM memiliki akurasi 99,0% dan *recall* 100% pada kelas Pro, sementara Vanilla RNN kurang stabil. Pada data nyata, GRU memprediksi 16 data dengan benar, lebih baik dari LSTM dengan 14 data dengan benar dan RNN dengan 13 data dengan benar. GRU unggul dalam akurasi, stabilitas, dan adaptasi terhadap data.

Kata kunci: analisis sentimen, *clash of champions*, Vanilla RNN, LSTM, GRU

Abstract

Clash of Champions is an educational program by Ruangguru on YouTube that has received mixed responses. This study aims to perform sentiment analysis using three Recurrent Neural Network (RNN) architectures: Vanilla Recurrent Neural Network (Vanilla RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). The data consists of 2,100 training samples, 300 validation samples, and 600 testing samples collected from YouTube and enriched with data augmentation using GPT-4 technology. Additionally, 35 comments from a survey conducted via Google Form are used for generalization testing. Comments are classified into three sentiments: Pro, Neutral, and Contra. The analysis involves preprocessing, model training, and evaluation using standard metrics. GRU demonstrated the best performance with an accuracy of 99.2% and the highest F1 score. LSTM achieved an accuracy of 99.0% and a recall of 100% for the Pro class, while Vanilla RNN was less stable. On real-world data, GRU correctly predicted 16 comments, outperforming LSTM with 14 correct predictions and RNN with 13 correct predictions. GRU excels in accuracy, stability, and adaptability to the data.

***Keywords**: sentiment analysis, clash of champions, Vanilla RNN, LSTM, GRU*

1. PENDAHULUAN

Dalam beberapa tahun terakhir, perkembangan teknologi digital telah memberikan pengaruh besar terhadap dunia pendidikan [1]. Salah satu bentuk inovasi yang muncul adalah penyelenggaraan Program edukatif berbasis kompetisi yang ditayangkan melalui media *daring* [2]. Program semacam ini tidak hanya berfungsi sebagai sarana hiburan, tetapi juga sebagai alat untuk memotivasi generasi muda agar lebih aktif dalam pembelajaran [3]. Salah satu contoh Program edukatif tersebut adalah *Clash of Champions*, yang diselenggarakan oleh Ruangguru dan ditayangkan melalui platform YouTube [4]. Program ini menampilkan kompetisi akademik antar mahasiswa sebagai upaya untuk meningkatkan semangat belajar dan partisipasi dalam pendidikan *digital* di Indonesia [5].

Meskipun membawa nilai-nilai edukatif yang positif, *Clash of Champions* tetap memunculkan beragam tanggapan dari masyarakat, khususnya mahasiswa. Sebagai kelompok yang terlibat langsung baik sebagai peserta maupun penonton aktif konten edukatif *digital*, mahasiswa kerap menyampaikan opini mereka melalui media sosial dan *forum* diskusi *daring* [6]. Tanggapan yang diberikan sangat bervariasi, mulai dari dukungan penuh terhadap konsep dan

pelaksanaan acara, hingga kritik terhadap aspek-aspek tertentu, seperti tekanan sosial dan ekspektasi tinggi terhadap peserta [7]. Keberagaman opini ini mencerminkan perlunya pemahaman yang lebih dalam dan sistematis terhadap persepsi mahasiswa [8]. Sayangnya, kajian akademik yang secara khusus menganalisis opini mahasiswa terhadap Program edukatif *digital* seperti ini masih terbatas [9].

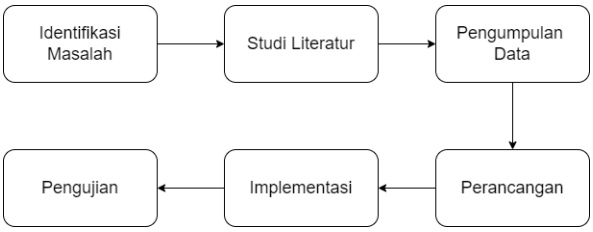
Analisis terhadap sentimen mahasiswa menjadi penting, terutama dalam konteks evaluasi efektivitas konten edukatif *digital* [10]. Dengan memahami bagaimana suatu Program diterima oleh audiensnya, pengembang dapat melakukan perbaikan yang lebih tepat sasaran [11]. Dalam hal ini, *analisis sentimen* menjadi metode yang relevan karena memungkinkan identifikasi pola opini publik secara otomatis dan objektif [12]. Namun, tantangan utama dalam *analisis sentimen* terletak pada kemampuan model untuk memahami konteks dan makna yang terkandung dalam teks, terutama dalam bahasa informal yang umum digunakan di media sosial [13].

Sebagai solusi atas tantangan tersebut, pendekatan berbasis *Deep Learning*, khususnya arsitektur *Recurrent Neural Network* RNN, menawarkan kemampuan pemrosesan data teks yang bersifat sekuensial dan kontekstual [14]. RNN mampu mempertahankan informasi dari tahap sebelumnya dalam sebuah urutan teks, sehingga lebih unggul dibanding metode tradisional seperti *Naive Bayes*, SVM, atau pendekatan berbasis leksikon [15] [16]. Beberapa varian RNN yang banyak digunakan dalam *Natural Language Processing NLP* antaralain *Vanilla RNN*, *Long Short-Term Memory* LSTM, dan *Gated Recurrent Unit* GRU, yang masing-masing memiliki keunggulan dalam menangkap makna semantik dan mengatasi masalah hilangnya konteks pada teks panjang [17] [18] [19].

Penelitian ini menganalisis sentimen mahasiswa terhadap Program Clash of Champions menggunakan klasifikasi berbasis RNN. Data terdiri dari 2.100 pelatihan, 300 validasi, 600 pengujian dari komentar YouTube, serta 35 komentar survei untuk uji generalisasi. Tiga arsitektur RNN, yaitu Vanilla RNN, LSTM, dan GRU, dibandingkan menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1 Score*. Hasil diharapkan mendukung pengembangan analisis sentimen yang lebih akurat untuk konten edukatif digital.

2. METODE PENELITIAN

Metode penelitian adalah langkah sistematis untuk mencapai tujuan secara ilmiah dan terukur [20], berfungsi sebagai panduan agar Proses penelitian dapat dipertanggungjawabkan [21]. Dalam penelitian ini, metode meliputi bagian utama dari identifikasi masalah hingga pengujian seperti pada gambar 1.



Gambar 1. Bagan Tahapan Pelaksanaan Penelitian

Gambar 1 menunjukkan enam tahapan penelitian: identifikasi masalah, studi literatur, pengumpulan data, perancangan, implementasi, dan pengujian. Penelitian dimulai dengan kajian sentimen *Clash of Champions*. Data komentar media sosial dan survei dikumpulkan dan diProses untuk melatih model Vanilla RNN, LSTM, dan GRU. Evaluasi menggunakan *metrik accuracy*, *precision, recall*, dan F1 *Score* serta pengujian kemampuan generalisasi model.

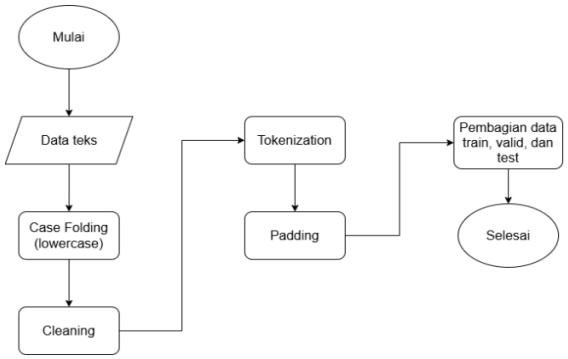
2.1. Pengumpulan Data

Pengumpulan data dilakukan dalam dua tahap, pertama, menyusun 3000 data teks yang didapatkan dari video terkait Clash of Champions pada aplikasi YouTube berisi komentar dan label sentimen yaitu Pro, Netral, dan Kontra dengan penambahan data augmentasi untuk

memperkaya variasi data teks menggunakan teknologi GPT-4. Data ini dirancang sesuai konteks *Clash of Champions*. Kedua, mengumpulkan 35 tanggapan yang didapatkan dari survey jajak pendapat melalui Google Form dengan struktur serupa tanggapan dan label sebagai data pengujian. Data uji ini mencerminkan opini masyarakat dan digunakan untuk mengevaluasi performa model dalam mengklasifikasikan sentimen.

2.2. Perancangan Data

Perancangan data untuk penelitian ini terdapat beberapa tahapan agar data dapat digunakan dengan baik. Proses perancangan data meliputi *case folding lowercase*, *cleaning*, *tokenization*, *padding*, dan pembagian data menjadi data *training*, *validation*, dan *testing* pada model.

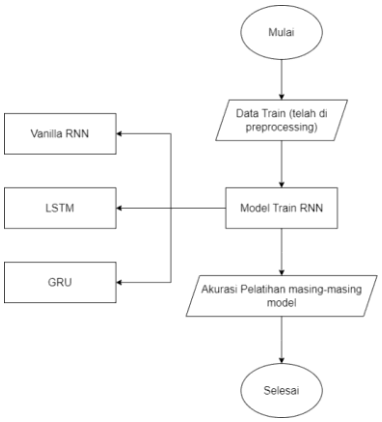


Gambar 2. Proses pada Perancangan Data

Gambar 2 memperlihatkan alur perancangan data meliputi case folding mengubah huruf menjadi kecil, cleaning menghapus karakter tidak relevan, tokenization memecah teks menjadi token, dan padding menyamakan panjang token. Data hasil Proses kemudian dibagi menjadi data training, validation, dan testing untuk melatih, menyetel, dan menguji model.

2.3. Perancangan Proses

Penelitian ini menganalisis sentimen menggunakan tiga arsitektur RNN: Vanilla RNN, LSTM, dan GRU. Data komentar yang sudah dilakukan *preproccesing* dibagi menjadi data pelatihan 70%, validasi 10%, dan pengujian 20%. Model dilatih secara terpisah dengan TensorFlow menggunakan *categorical crossentropy* dan optimizer Adam. Evaluasi performa memakai metrik *accuracy*, *precision*, *recall*, dan *F1 Score* untuk memilih model terbaik dalam mengklasifikasikan sentimen Pro, Netral, dan Kontra. Rancang Proses dapat dilihat pada gambar 3.



Gambar 3. Proses Pelatihan RNN

Gambar 3 menunjukkan alur Proses pelatihan model RNN, dimulai dari data yang telah di *preProcessing*, kemudian dilatih menggunakan tiga arsitektur berbeda: Vanilla RNN, LSTM, dan GRU. Masing-masing model dilatih secara terpisah namun dengan alur yang seragam. Setelah

pelatihan, dilakukan evaluasi akurasi dari setiap model untuk membandingkan performa ketiganya.

2.4. Perancangan Pengujian

Pengujian model dilakukan untuk menilai kinerja *klasifikasi sentimen* menggunakan *confusion matrix* dan metrik seperti *accuracy*, *precision*, *recall*, dan *F1 Score* [6]. *Confusion matrix* yang akan digunakan dapat dilihat pada tabel 1.

Tabel 1. *Confusion Matrix* dengan Kelas Sebanyak 3

Prediksi	Aktual		
	Kelas A	**Kelas B**	**Kelas C**
Kelas A	TP_A	FP_BA	FP_CA
Kelas B	FP_AB	TP_B	FP_CB
Kelas C	FP_AC	FP_BC	TP_C

Tabel 1 menunjukkan tabel *confusion matrix* sebagai tolak ukur evaluasi model. *Accuracy* mengukur ketepatan keseluruhan, *precision* menunjukkan Proporsi prediksi benar per kelas, *recall* menilai kemampuan mengenali data kelas sebenarnya, dan *F1 Score* menyeimbangkan *precision* dan *recall*. Selanjutnya, model diuji dengan data *unseen* untuk mengukur kemampuan *generalisasi* pada data baru. Perhitungan evaluasi dapat dilihat pada rumus (1) hingga (4), dengan i sebagai patokan kelas sasaran.

$$Accuracy = \frac{TP_A + TP_B + TP_C}{total\ keseluruhan\ nilai} \tag{1}$$

$$Precision_i = \frac{TP_i}{TP_i + FP_i} \tag{2}$$

$$Recall_i = \frac{TP_i}{TP_i + FN_i} \tag{3}$$

$$\text{F1 } Score_i = 2 \times \frac{\text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \tag{4}$$

3. HASIL DAN PEMBAHASAN

3.1 Visualisasi Data

Tahap awal pengolahan data dalam penelitian ini dimulai dengan visualisasi data teks keseluruhan, Visualisasi data berguna untuk melihat dan mendeskripsikan data yang akan digunakan dimana data yang akan dipakai dan dipecah dalam pelatihan terdiri dari 3.000 baris dan dua kolom, yaitu kolom Teks yang memuat opini dalam bentuk kalimat dan kolom Label yang menunjukkan kategori sentimen Pro, Netral, dan Kontra. Komentar Pro bernada positif, Kontra berisi kritik, dan Netral bersifat deskriptif tanpa emosi kuat.

3.2 *Preprocessing* data

Setelah data divisualisasikan, tahap selanjutnya adalah *preprocessing* untuk membersihkan dan menyiapkan data teks sebelum diproses oleh model. Langkah pertama adalah *case folding*, yaitu mengubah seluruh huruf menjadi huruf kecil agar kata seperti “Pemerintah” dan “pemerintah” dianggap sama, misalnya teks “Pemerintah Harus TEGAS” diubah menjadi “pemerintah harus tegas”. Selanjutnya, dilakukan *cleaning* untuk menghapus karakter yang tidak relevan seperti tanda baca, angka, dan simbol, sehingga teks “pemerintah harus tegas!!!” menjadi “pemerintah harus tegas”. Setelah itu, dilakukan *tokenization* dengan memecah teks menjadi kata-kata terpisah, misalnya kalimat “pemerintah harus tegas” menjadi ["pemerintah", "harus", "tegas"]. Terakhir, dilakukan *padding* untuk menyamakan panjang urutan kata dengan menambahkan nilai nol 0 di akhir token yang lebih pendek, contohnya ["pemerintah", "harus", "tegas"] menjadi ["pemerintah", "harus", "tegas", 0, 0].

3.3 Pembagian Data

Setelah melalui tahapan *preprocessing*, data dibagi menjadi tiga bagian utama, yaitu data latih, data validasi, dan data uji. Tujuan dari pembagian ini adalah untuk memisahkan Proses pelatihan model, pemantauan kinerja selama pelatihan, dan pengujian akhir guna memperoleh evaluasi performa yang objektif. Dari total 3.000 data hasil *augmentasi*, sebanyak 2.100 data dengan persentase 70% digunakan sebagai data latih, 300 data dengan persentase 10% sebagai data validasi, dan 600 data dengan persentase 20% sebagai data uji. Data uji ini disusun secara seimbang, dengan jumlah label yang Proporsional untuk setiap kelas sentimen guna memastikan bahwa hasil evaluasi menggunakan *confusion matrix* multikelas berukuran 3×3 dapat menggambarkan performa model secara baik pada masing-masing kategori.

3.4 Penerapan Proses

Penelitian ini menerapkan tiga arsitektur RNN, yaitu Vanilla RNN, LSTM, dan GRU yang dibangun dengan *TensorFlow* menggunakan struktur dasar serupa, terdiri atas lapisan *embedding* berdimensi 128, satu lapisan inti RNN serta dua lapisan *dense* dengan aktivasi *ReLU* dan *softmax* untuk klasifikasi sentimen Pro, Netral, dan Kontra yang dapat dilihat pada tabel 2.

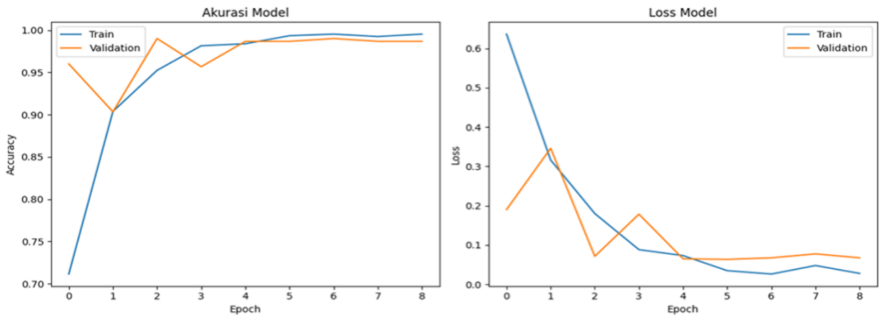
Tabel 2. Penggambaran Arsitektur RNN yang digunakan

Urutan	Lapisan	Spesifikasi	Aktivasi
1	Input Layer	Shape: (100,) (panjang sekuens 100 token)	-
2	Embedding Layer	input_dim = 10000 (kosakata 10.000 kata), output_dim = 128 (dimensi vektor kata)	-
3	RNN Layer	128 unit, return_sequences = False, menggunakan salah satu dari SimpleRNN, LSTM, atau GRU	Tanh
4	Dense Layer 1	32 neuron	ReLU
5	Dense Layer 2	3 neuron (kelas Pro, Netral, Kontra)	Softmax

Pada tabel 2, arsitektur RNN pada tabel terdiri dari lima lapisan, dimulai dari input berisi 100 *token* yang diubah menjadi vektor 128 dimensi melalui *embedding*. Representasi ini diproses oleh lapisan RNN dengan 128 *unit* dan aktivasi *tanh*, lalu diteruskan ke *dense* 32 *neuron* beraktivasi *ReLU* dan *dense* output 3 *neuron* beraktivasi *softmax* untuk klasifikasi Pro, Netral, atau Kontra. Rancangan ini bertujuan untuk membandingkan performa ketiga arsitektur dalam mengklasifikasikan data teks berbahasa Indonesia secara optimal.

3.5 Hasil Pengujian Vanilla RNN

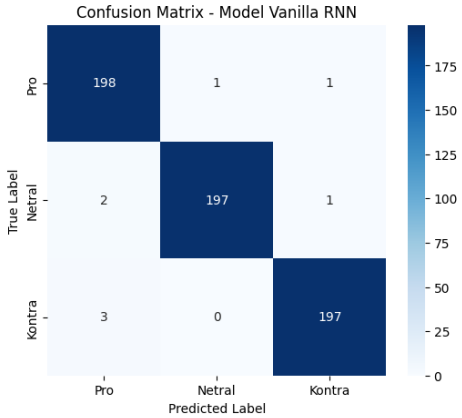
Model dari Vanilla RNN ini digunakan untuk analisis sentimen. Vanilla RNN dipilih sebagai pendekatan awal karena memiliki struktur sederhana dan mampu memProses data sekuensial dengan mempertimbangkan keterkaitan antar kata berdasarkan urutan kemunculannya [22]. Hasil pengujian disajikan melalui grafik akurasi dan loss Gambar 4, confusion matrix Gambar 5, dan tabel evaluasi Tabel 3, yang merepresentasikan kinerja model selama pelatihan pada setiap epoch.



a b

Gambar 4. Grafik a *accuracy* dan b *loss* dari pelatihan model Vanilla RNN

Gambar 4 memaparkan hasil *accuracy* dan *loss* selama pelatihan. Grafik *accuracy* a menunjukkan peningkatan konsisten hingga stabil di atas 0,95 setelah *epoch* ke-4, menandakan model mampu mempelajari pola dengan baik. Grafik *loss* b mengalami penurunan tajam di awal, lalu stabil pada nilai rendah, menunjukkan keberhasilan meminimalkan kesalahan. Pengujian dengan data uji menghasilkan *confusion matrix* yang divisualisasikan pada Gambar 6



Gambar 5. *Confusion Matrix* dari Pengujian Model Vanilla RNN

Gambar 5 menunjukkan visualisasi *confusion matrix* dari 600 data uji pada tiga kategori sentimen, yaitu Pro, Netral, dan Kontra. Sebagian besar prediksi berada pada diagonal utama dengan *true positive* masing-masing 198 untuk Pro, 197 untuk Netral, dan 197 untuk Kontra. FP adalah prediksi salah masuk kelas tertentu, yaitu Pro diprediksi sebagai Netral sebanyak 2 dan Kontra sebanyak 3, Netral diprediksi Pro sebanyak 1, serta Kontra diprediksi Pro sebanyak 1 dan Netral sebanyak 1, FN adalah data asli kelas tertentu yang diprediksi salah, seperti Pro diprediksi Netral sebanyak 1 dan Kontra sebanyak 1, Netral diprediksi Pro sebanyak 2 dan Kontra sebanyak 1, serta Kontra diprediksi Pro sebanyak 3. TN adalah data yang bukan kelas tersebut dan tidak diprediksi sebagai kelas itu. Hasil *classification report* dari model Vanilla RNN dihitung dengan persamaan (1) hingga (4) dan ditampilkan pada Tabel 3.

1. *Accuracy*

$$Accuracy = \frac{592}{600} = 0{,}987$$

2. *Precision*

$$Precision_{\text{Pro}} = \frac{198}{198+5} = \frac{198}{203} = 0{,}975$$

$$Precision_{\text{Netral}} = \frac{197}{197+1} = \frac{197}{198} = 0{,}995$$

$$Precision_{\text{Kontra}} = \frac{197}{197+2} = \frac{197}{199} = 0{,}990$$

3. *Recall*

$$Recall_{\text{Pro}} = \frac{198}{198+2} = \frac{198}{200} = 0{,}990$$

$$Recall_{\text{Netral}} = \frac{197}{197+3} = \frac{197}{200} = 0{,}985$$

$$Recall_{\text{Kontra}} = \frac{197}{197+3} = \frac{197}{200} = 0{,}985$$

4. *F1 Score*

$$F1_{\text{Pro}} = \frac{2\cdot(0{,}9754\cdot 0{,}99)}{0{,}9754+0{,}99} = 0{,}981$$

$$F1_{\text{Netral}} = \frac{2\cdot(0{,}9949\cdot 0{,}985)}{0{,}9949+0{,}985} = 0{,}989$$

$$F1_{\text{Kontra}} = \frac{2\cdot(0{,}9899\cdot 0{,}985)}{0{,}9899+0{,}985} = 0{,}987$$

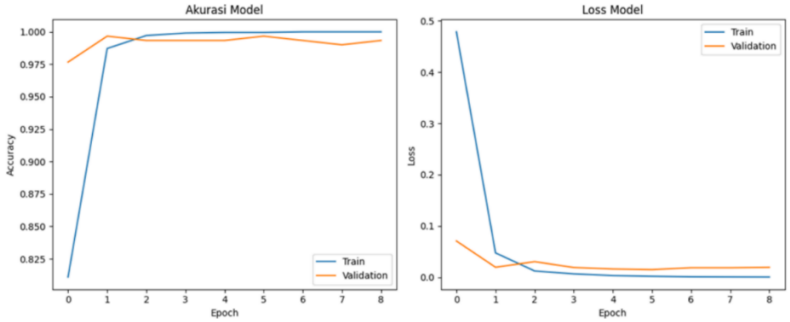
Tabel 3. *Classification Report* dari Pengujian Model Vanilla RNN

Accuracy	0,987		
	Precision	*Recall*	*F1 Score*
Pro	0,975	0,990	0,981
Netral	0,995	0,985	0,989
Kontra	0,990	0,985	0,987

Tabel 3 menampilkan hasil *classification report* pada setiap kelas dari model Vanilla RNN yang telah sesuai dengan perhitungan *confusion matrix*. Model ini memiliki *accuracy* tinggi sebesar sebesar 0,987. *Precision* tertinggi diperoleh pada kelas Netral sebesar 0,995 dan terendah pada kelas Pro sebesar 0,975. *Recall* tertinggi terdapat pada kelas Pro sebesar 0,990 dan terendah pada kelas Netral serta Kontra sebesar 0,985. *F1 Score* tertinggi dicapai pada kelas Netral sebesar 0,989 dan terendah pada kelas Pro sebesar 0,981.

3.6 Hasil Pengujian LSTM

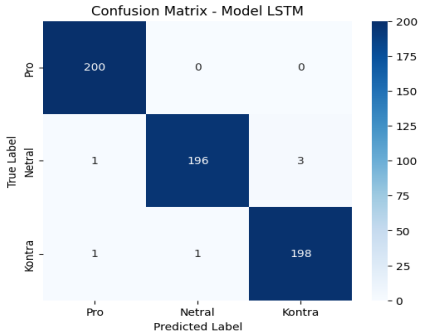
Hasil pengujian model LSTM disajikan dalam grafik *accuracy* dan *loss* Gambar 6, *confusion matrix* Gambar 7, serta tabel evaluasi Tabel 4. LSTM dipilih sebagai pendekatan kedua karena kemampuannya mempelajari dependensi jangka panjang pada data berurutan, sehingga sesuai untuk teks komentar yang memiliki konteks antar kata [22]. Grafik *accuracy* dan *loss* pada setiap *epoch* yang menggambarkan kinerja model selama pelatihan dapat dilihat pada Gambar 6.



a b

Gambar 6. Grafik a *accuracy* dan b *loss* dari pelatihan model LSTM.

Gambar 6 menampilkan performa model LSTM selama pelatihan melalui grafik *accuracy* a dan *loss* b pada garis pelatihan *train* dan validasi *validation*. Grafik *accuracy* menunjukkan peningkatan cepat dan stabil di atas 0,98 setelah *epoch* kedua, sedangkan grafik *loss* menurun tajam di awal dan berkonvergensi mendekati nol. Hasil ini mencerminkan kemampuan model dalam belajar secara efektif dan meminimalkan kesalahan. Pengujian dengan data uji menghasilkan *confusion matrix* yang divisualisasikan pada Gambar 7



Gambar 7. *Confusion Matrix* dari Pengujian Model LSTM

Gambar 7 menunjukkan visualisasi *confusion matrix* dari 600 data uji pada tiga kategori sentimen: Pro, Netral, dan Kontra. Gambar di atas menunjukkan visualisasi confusion matrix dari

600 data uji pada tiga kategori sentimen, yaitu Pro, Netral, dan Kontra. Sebagian besar prediksi berada pada diagonal utama dengan true positive masing-masing 200 untuk Pro, 196 untuk Netral, dan 198 untuk Kontra. FP adalah prediksi salah masuk kelas tertentu, yaitu Pro diprediksi sebagai Netral sebanyak 1 dan Kontra sebanyak 1, Netral diprediksi Pro sebanyak 0 dan Kontra sebanyak 1, serta Kontra diprediksi Pro sebanyak 0 dan Netral sebanyak 3. FN adalah data asli kelas tertentu yang diprediksi salah, seperti Pro diprediksi Netral sebanyak 1 dan Kontra sebanyak 1, Netral diprediksi Pro sebanyak 0 dan Kontra sebanyak 3, serta Kontra diprediksi Pro sebanyak 1 dan Netral sebanyak 1. Hasil *classification report* dari model LSTM dihitung dengan persamaan (1) hingga (4) dan ditampilkan pada Tabel 4.

1. *Accuracy*

   $Accuracy = \frac{594}{600} = 0{,}990$

2. *Precision*

   $Precision_{Pro} = \frac{200}{200+2} = \frac{200}{202} = 0{,}990$

   $Precision_{Netral} = \frac{196}{196+1} = \frac{196}{197} = 0{,}995$

   $Precision_{Kontra} = \frac{198}{198+3} = \frac{198}{201} = 0{,}985$

3. *Recall*

   $Recall_{Pro} = \frac{200}{200+0} = \frac{200}{200} = 1{,}000$

   $Recall_{Netral} = \frac{196}{196+4} = \frac{196}{200} = 0{,}980$

   $Recall_{Kontra} = \frac{198}{198+2} = \frac{198}{200} = 0{,}990$

4. *F1 Score*

   $F1_{Pro} = \frac{2\cdot(0{,}9901\cdot 1{,}0)}{0{,}9901+1{,}0} = 0{,}995$

   $F1_{Netral} = \frac{2\cdot(0{,}9949\cdot 0{,}98)}{0{,}9949+0{,}98} = 0{,}985$

   $F1_{Kontra} = \frac{2\cdot(0{,}9851\cdot 0{,}99)}{0{,}9851+0{,}99} = 0{,}986$

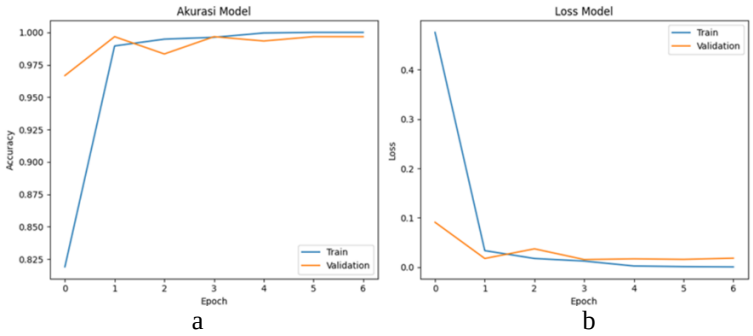
Tabel 4. *Classification Report* dari Pengujian Model LSTM

Accuracy	0,990		
	Precision	*Recall*	*F1 Score*
Pro	0,990	1,000	0,995
Netral	0,995	0,980	0,985
Kontra	0,985	0,990	0,986

Tabel 3 menampilkan hasil *classification report* pada setiap kelas dari model LSTM yang telah sesuai dengan perhitungan *confusion matrix*. Model ini memiliki *accuracy* tinggi sebesar 0,990. *Precision* tertinggi diperoleh pada kelas Netral sebesar 0,995 dan terendah pada kelas Kontra sebesar 0,985. *Recall* tertinggi terdapat pada kelas Pro sebesar 1,000 dan terendah pada kelas Netral sebesar 0,980. *F1 Score* tertinggi dicapai pada kelas Pro sebesar 0,995 dan terendah pada kelas Netral sebesar 0,985.

3.7 Hasil Pengujian GRU

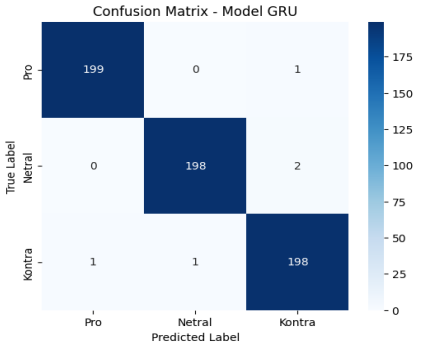
Hasil pengujian model GRU disajikan dalam grafik *accuracy* dan *loss* Gambar 8, *confusion matrix* Gambar 9, serta tabel evaluasi Tabel 4. GRU dipilih sebagai pendekatan ketiga karena strukturnya lebih sederhana dibandingkan LSTM namun tetap mampu menangani data berurutan secara efisien, sehingga cocok untuk memProses teks komentar yang memiliki hubungan kontekstual antar kata [22]. Grafik *accuracy* dan *loss* pada setiap *epoch* yang merepresentasikan kinerja model selama pelatihan ditampilkan pada Gambar 8.



a b

Gambar 8. Grafik a *Accuracy* dan b *loss* dari PelatihanM GRU

Gambar 8 menampilkan performa model GRU selama pelatihan melalui grafik *accuracy* a dan *loss* b pada garis pelatihan *train* dan validasi *validation*. Grafik *accuracy* menunjukkan peningkatan tajam pada *epoch* awal, dengan nilai pelatihan dan validasi di atas 0,98 sejak *epoch* kedua, lalu stabil mendekati 1 pada *epoch* berikutnya. Grafik *loss* menurun drastis di awal pelatihan dan berkonvergensi pada nilai sangat rendah mendekati nol mulai *epoch* ketiga. Pengujian dengan data uji menghasilkan *confusion matrix* yang divisualisasikan pada Gambar 9



Gambar 9. *Confusion Matrix* dari pengujian model GRU

Gambar 9 menunjukkan visualisasi *confusion matrix* dari 600 data uji pada tiga kategori sentimen, yaitu Pro, Netral, dan Kontra. Sebagian besar prediksi tepat berada pada diagonal utama dengan nilai true positive masing-masing 199 untuk Pro, 198 untuk Netral, dan 198 untuk Kontra. Untuk FP, Pro diprediksi salah sebagai Netral sebanyak 0 dan sebagai Kontra sebanyak 1, Netral diprediksi sebagai Pro sebanyak 0 dan sebagai Kontra sebanyak 2, sedangkan Kontra diprediksi sebagai Pro sebanyak 1 dan sebagai Netral sebanyak 1. Sementara itu, FN menunjukkan bahwa data asli kelas Pro salah diprediksi sebagai Netral sebanyak 0 dan sebagai Kontra sebanyak 1, kelas Netral salah diprediksi sebagai Pro sebanyak 0 dan sebagai Kontra sebanyak 2, serta kelas Kontra salah diprediksi sebagai Pro sebanyak 1 dan sebagai Netral sebanyak 1. Hasil *classification report* dari model GRU dihitung dengan persamaan (1) hingga (4) dan ditampilkan pada Tabel 4.

1. *Accuracy*

$$Accuracy = \frac{595}{600} = 0{,}992$$

2. *Precision*

$$Precision_{Pro} = \frac{199}{199+1} = \frac{199}{200} = 0{,}995$$

$$Precision_{Netral} = \frac{198}{198+1} = \frac{198}{199} = 0{,}995$$

$$Precision_{Kontra} = \frac{198}{198+3} = \frac{198}{201} = 0{,}985$$

3. *Recall*

$$Recall_{Pro} = \frac{199}{199+1} = \frac{199}{200} = 0{,}995$$
$$Recall_{Netral} = \frac{198}{198+2} = \frac{198}{200} = 0{,}990$$
$$Recall_{Pro} = \frac{199}{199+1} = \frac{199}{200} = 0{,}995$$

4. *F1 Score*

$$F1_{Pro} = \frac{2\cdot(0{,}995\cdot 0{,}995)}{0{,}995+0{,}995} = 0{,}995$$
$$F1_{Netral} = \frac{2\cdot(0{,}9949\cdot 0{,}99)}{0{,}9949+0{,}99} = 0{,}991$$
$$F1_{Kontra} = \frac{2\cdot(0{,}9851\cdot 0{,}99)}{0{,}9851+0{,}99} = 0{,}988$$

Tabel 4. Evaluasi Pelatihan Model GRU

Accuracy	0,992		
	Precision	*Recall*	*F1 Score*
Pro	0.995	0.995	0.995
Netral	0.995	0.990	0.991
Kontra	0.985	0.995	0.988

Tabel 4 menampilkan hasil *classification report* pada setiap kelas dari model GRU yang telah sesuai dengan perhitungan *confusion matrix*. Model ini memiliki *accuracy* tinggi sebesar 0,992. *Precision* tertinggi diperoleh pada kelas Netral sebesar 0,995 dan terendah pada kelas Kontra sebesar 0,985. *Recall* tertinggi terdapat pada kelas Kontra sebesar 0,995 dan terendah pada kelas Netral sebesar 0,990. *F1 Score* tertinggi dicapai pada kelas Pro sebesar 0,995 dan terendah pada kelas Kontra sebesar 0,988.

3.8 Hasil Uji Generalisasi Model Pada Setiap Arsitektur RNN

Model RNN yang telah dievaluasi selanjutnya dilakukan pengujian model menggunakan data *unseen* dengan pelabelan manual untuk label asli, yaitu data yang belum pernah digunakan dalam Proses pelatihan maupun evaluasi model sebelumnya, guna menguji kemampuan generalisasi model terhadap data baru dengan melakukan pelabelan atau klasifikasi teks dengan menggunakan model yang telah dilatih dan dievaluasi sebelumnya. Pengujian terhadap data *unseen* ini bertujuan untuk melihat seberapa baik model mampu mengklasifikasikan data dunia nyata yang tidak dikenal sebelumnya yang dapat dilihat bentuk pelabelannya pada tabel 5.

Tabel 5. Hasil Pengujian dengan Data *Unseen*

Tanggapan	Label asli	Vanilla RNN	LSTM	GRU
Saya kurang suka karena ada orang yang sombong	Kontra	Pro	Pro	Pro
Acara nya kurang menarik menurut saya	Netral	Kontra	Kontra	Kontra
Saya suka acara ini karena membuat saya lebih termotivasi	Pro	Kontra	Pro	Pro
……	….	….	….	….
Potensinya ada, tapi masih banyak hal yang bisa diperbaiki supaya lebih seru …..	Netral	Kontra	Kontra	Kontra

Hasil uji generalisasi data unseen dengan model pada setiap arsitektur RNN, yaitu Vanilla RNN, LSTM, dan GRU dengan bentuk dari sebagian pelabelan atau klasifikasi teks yang dapat dilihat pada tabel 5. Didapatkan hasil bahwa model Vanilla RNN menghasilkan prediksi yang sesuai dengan label asli sebanyak 13 data. Model LSTM menunjukkan hasil yang sedikit lebih baik dengan total 14 prediksi yang sesuai. Sementara itu, model GRU menghasilkan prediksi yang paling banyak sesuai dengan label asli, yaitu sebanyak 16 data. Hasil ini menunjukkan adanya perbedaan kemampuan dari masing-masing arsitektur dalam mengklasifikasikan data yang belum pernah dilihat sebelumnya.

4. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa penerapan arsitektur RNN efektif dalam melakukan analisis sentimen terhadap komentar penonton acara *Clash of Champions*. Di antara ketiga arsitektur yang diuji, yaitu Vanilla RNN, LSTM, dan GRU. model GRU menunjukkan performa paling unggul dengan akurasi sebesar 99,2 % serta *F1 Score* yang konsisten tinggi pada semua kelas sentimen. Keunggulan ini tidak hanya terlihat pada hasil pelatihan dan pengujian, tetapi juga pada uji generalisasi terhadap data nyata, di mana GRU berhasil memprediksi 16 dari 35 komentar secara tepat. Hal ini menunjukkan bahwa GRU memiliki kemampuan klasifikasi dan adaptasi kontekstual yang lebih stabil dibandingkan dua arsitektur lainnya.

Sebagai tindak lanjut, disarankan agar penelitian selanjutnya mengeksplorasi konteks bahasa yang lebih kompleks, seperti ekspresi ironi atau sarkasme, yang sering muncul dalam opini digital. Model juga dapat diperluas penerapannya pada platform lain untuk menguji konsistensi performa dalam beragam lingkungan data. Selain itu, penambahan fitur analisis emosi serta perluasan jumlah dan variasi data pelatihan diharapkan dapat meningkatkan kemampuan model dalam memahami sentimen yang lebih halus dan beragam.

DAFTAR PUSTAKA

[1] E. Suncaka, "Meninjau Permasalahan Rendahnya Kualitas Pendidikan Di Indonesia," *UNISAN JURNAL: Jurnal Manajemen dan Pendidikan*, vol. 2, no. 3, pp. 36–49, 2023, [Online]. Available: https://journal.an-nur.ac.id/index.php/unisanjournal/article/view/1234

[2] S. Ledia, B. Mauli, and R. Bustam, "Implementasi Kurikulum Merdeka Dalam Meningkatkan Mutu Pendidikan," *Reslaj: Religion Education Social Laa Roiba Journal*, vol. 6, no. 1, pp. 790–806, 2024, doi: 10.47476/reslaj.v6i1.2708.

[3] M. Cholilah, A. G. P. Tatuwo, S. P. Rosdiana, and A. N. Fatirul, "Pengembangan Kurikulum Merdeka Dalam Satuan Pendidikan Serta Implementasi Kurikulum Merdeka Pada Pembelajaran Abad 21," *Sanskara Pendidikan dan Pengajaran*, vol. 01, no. 02, pp. 57–66, 2023, doi: 10.58812/spp.v1.i02.

[4] B. Wiraning, "Ramai Clash of Champions Ruangguru, Begini Caranya Senang-senang Belajar Lewat Game," https://indiekraf.com/ramai-clash-of-champions-ruangguru-begini-caranya-senang-senang-belajar-lewat-game/.

[5] P. G. Primasari, "Orang Indonesia Nggak Cocok Dikasih Tayangan kayak *Clash of Champions* Ruangguru. Ilmunya Dikit, Dramanya Selangit," https://mojok.co/terminal/orang-indonesia-nggak-cocok-dikasih-tayangan-kayak-clash-of-champions-ruangguru/.

[6] S. Farisi and S. Hadi, "Analisis Sentimen menggunakan Recurrent Neural Network Terkait Isu Anies Baswedan Sebagai Calon Presiden 2024," 2023.

[7] I. Pakpahan and J. Pardede, "Analisis Sentimen Penanganan Covid-19 Menggunakan Metode Long Short-Term Memory Pada Media Sosial Twitter," *Jurnal Publikasi Teknik Informatika*, vol. 2, no. 1, pp. 12-25, 2023, doi: http://dx.doi.org/10.55606/jupti.v1i1.767.

[8] F. Amaliah, I. Kadek, and D. Nuryana, "Perbandingan Akurasi Metode Lexicon Based Dan Naive Bayes Classifier Pada Analisis Sentimen Pendapat Masyarakat Terhadap Aplikasi Investasi Pada Media Twitter," *Journal of Informatics and Computer Science*, vol. 3, no. 3, 2022, doi: https://doi.org/10.26740/jinacs.v3n03.p384-393.

[9] I. Wayan, D. Gafatia, and N. Hadinata, "Analisis Pro Kontra Vaksin Covid 19 Menggunakan Sentiment Analysis Sumber Media Sosial Twitter," Jurnal Pengembangan Sistem Informasi & Informatika, vol. 2, no. 1, pp. 34-42, 2021, doi: https://doi.org/10.47747/jpsii.v2i1.544.

[10] A. M. Iddrisu, *et al*., "A sentiment analysis framework to classify instances of sarcastic sentiments within the aviation sector," *International Journal of Information Management Data Insights*, vol. 3, no. 2, 2023, doi: 10.1016/j.jjimei.2023.100180.

[11] F. Wulandari, E. Haerani, M. Fikry, and E. Budianita, "Analisis sentimen larangan penggunaan obat sirup menggunakan algoritma naive bayes classifier," *Jurnal CoSciTech Computer Science and Information Technology*, vol. 4, no. 1, pp. 88–96, 2023, doi: https://doi.org/10.37859/coscitech.v4i1.4781.

[12] A. H. Hasugian, M. Fakhriza, and D. Zukhoiriyah, "Analisis Sentimen Pada Review Pengguna E-Commerce Menggunakan Algoritma Naïve Bayes," *Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD*, vol. 6, no. 1, pp. 98-107, 2023, doi: https://doi.org/10.53513/jsk.v6i1.7400.

[13] F. Alifiana, *et al*., "Analisis Sentimen Aplikasi Duolingo Menggunakan Algoritma Naïve Bayes dan Support Machine Learning," *Jurnal Device*, vol. 13, no. 2, pp. 223–230, 2023, doi: https://doi.org/10.32699/device.v13i2.5905.

[14] A. Pratama and M. Rosyda, "Analisis Sentimen dalam Aplikasi X terhadap Pengungsi Rohingya dengan LSTM," *SKANIKA: Sistem Komputer dan Teknik Informatika*, vol. 8, no. 1, pp. 95-105, 2025, https://doi.org/10.36080/skanika.v8i1.3329.

[15] F. N. Salsabilla and A. Witanti, "Analisis Sentimen Akhir Masa Jabatan Presiden Jokowi Pada Media Sosial X Menggunakan Naïve Bayes," *SKANIKA: Sistem Komputer dan Teknik Informatika*, vol. 8, no. 1, pp. 106–115, 2025, doi: https://doi.org/10.36080/skanika.v8i1.3331.

[16] N. Khoirunnisaa, *et al*., "Klasifikasi Teks Ulasan Aplikasi Netflix Pada Google Play Store Menggunakan Algoritma Naive Bayes dan SVM," vol. 7, no. 1, pp. 64–73, 2024, https://doi.org/10.36080/skanika.v7i1.3138.

[17] A. S. Berliana and M. Mustikasari, "Analisis Sentimen Pada Ulasan Aplikasi Jakartanotebook Di Google Play Menggunakan Metode Recurrent Neural Network RNN," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3, 2024, doi: 10.23960/jitet.v12i3.5067.

[18] R. Cahuantzi, X. Chen, and S. Güttel, "A comparison of LSTM and GRU networks for learning symbolic sequences," *ArXiv*, Jan. 2023, doi: https://doi.org/10.1007/978-3-031-37963-5_53.

[19] R. Saidi, F. Jarray, and M. Alsuhaibani, "Comparative Analysis of Recurrent Neural Network Architectures for Arabic Word Sense Disambiguation," in *International Conference on Web Information Systems and Technologies, WEBIST - Proceedings*, Science and Technology Publications, Lda, 2022, pp. 272–277. doi: 10.5220/0011527600003318.

[20] D. Lanasemba, "Implementasi Long-short Term Memory LSTM Untuk Generasi Feedback Berbahasa Indonesia Pada Sistem Penilaian Esai," *Jurnal Fasilkom*, vol. 14, no. 1, pp. 101–113, 2024, doi: https://doi.org/10.37859/jf.v14i1.6906.

[21] C. A. Maharani, B. Warsito, and R. Santoso, "Analisis Sentimen Vaksin Covid-19 Pada Twitter Menggunakan Recurrent Neural Network Rnn Dengan Algoritma Long Short-term Memory LSTM," *Jurnal Gaussian*, vol. 12, no. 3, pp. 403–413, 2024, doi: https://doi.org/10.14710/j.gauss.12.3.403-413.

[22] Suyanto, K. N. Ramadhani, and S. Mandala, *Deep Learning Modernisasi Machine Learning Untuk Big Data*, Bandung: Informatika Bandung, 2019.