

Analisis Sentimen Gambar pada Media Sosial dengan Pendekatan Deep Learning

Ronaldito Juan Bantaras T^{a1}, Made Agung Raharja^{a2}

^aProgram Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Udayana
Jalan Raya Kampus Udayana, Bukit Jimbaran, Kuta Selatan, Badung, Bali Indonesia
¹ronalditotarigan30@email.com
²made.agung@unud.ac.id

Abstract

Sentiment analysis of images on social media using a deep learning approach is an interesting research topic in the field of artificial intelligence. It involves data collection, training deep neural network models, testing and evaluation, and application and analysis on social media. The results of this analysis provide valuable insights to users in understanding user responses to content, detecting evolving sentiment trends, and providing important insights for business purposes and decision-making. Deep learning offers a strong and effective method for understanding emotional expressions within images shared on social media platforms.

Keywords: *Sentiment analysis, image analysis, social media, deep learning*

1. Pendahuluan

Dalam era digital yang terus berkembang, media sosial telah menjadi platform yang sangat populer untuk berbagi konten dan berinteraksi dengan pengguna lainnya. Jutaan gambar diposting setiap hari di media sosial, mencerminkan beragam pengalaman dan ekspresi emosional dari penggunanya. Untuk memahami tanggapan pengguna terhadap konten dan mendapatkan wawasan yang berharga, analisis sentimen gambar pada media sosial menjadi semakin penting.

Analisis sentimen tradisional biasanya dilakukan dengan memanfaatkan teks atau data teks, seperti postingan, komentar, atau ulasan. Namun, dengan kemajuan dalam bidang kecerdasan buatan dan khususnya dalam deep learning, kita sekarang dapat menerapkan metode yang serupa untuk menganalisis sentimen dalam gambar. Pendekatan deep learning menggunakan neural networks yang kuat untuk mengenali dan memahami ekspresi emosional yang terkandung dalam gambar.

Penelitian ini akan fokus pada analisis sentimen gambar pada media sosial dengan pendekatan deep learning. Penulis akan menggunakan teknik pengenalan gambar yang dalam untuk mengklasifikasikan gambar-gambar tersebut menjadi sentimen yang terkait, seperti positif, negatif, atau netral. Dengan memanfaatkan kekuatan deep learning dapat mengembangkan model neural network yang akan dilatih menggunakan dataset gambar yang relevan.

Proses analisis sentimen gambar dengan deep learning melibatkan langkah-langkah yang mencakup pengumpulan data gambar yang relevan dari media sosial, penyiapan dan preprocessing data, pelatihan model neural network yang dalam, serta pengujian dan evaluasi kinerja model tersebut. Setelah melalui tahap ini, model akan siap untuk diterapkan pada gambar-gambar baru yang ditemukan di media sosial untuk menganalisis sentimen mereka secara otomatis.

Dengan menerapkan analisis sentimen gambar pada media sosial, kita dapat mendapatkan wawasan yang berharga tentang bagaimana pengguna merespons konten yang dibagikan dan mendeteksi tren sentimen yang sedang berkembang. Hal ini dapat memberikan manfaat dalam

berbagai konteks, seperti penelitian akademik, pengembangan strategi pemasaran yang efektif, dan manajemen merek di era digital yang kompetitif.

Penelitian ini diharapkan berharap dapat memberikan kontribusi dalam pemahaman dan pengembangan teknik analisis sentimen gambar pada media sosial dengan pendekatan deep learning. Diharapkan hasil penelitian ini dapat memberikan wawasan yang berharga kepada para pengguna dalam memahami tanggapan pengguna terhadap konten di media sosial serta memberikan pemahaman yang lebih dalam tentang sentimen dan ekspresi emosional yang terkandung dalam gambar-gambar yang dibagikan di platform tersebut.

2. Metode Penelitian

2.1. Pengumpulan Data

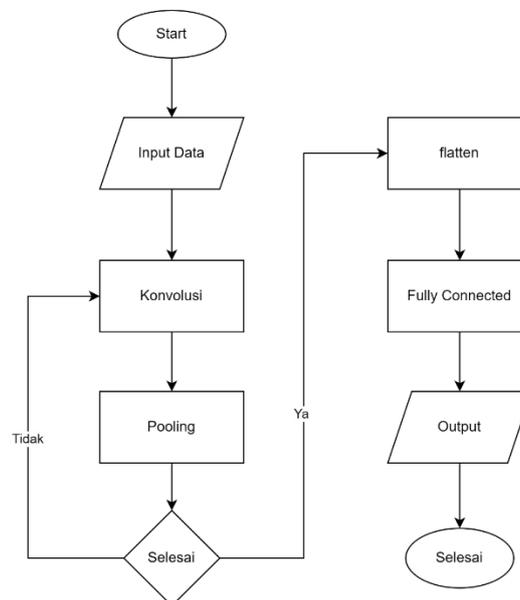
Dataset yang digunakan dalam penelitian ini terdiri dari 50 gambar yang dikumpulkan dari media sosial, seperti Twitter dan Instagram. Setiap gambar memiliki label sentimen yang telah ditentukan sebelumnya, yaitu positif, negatif, atau netral. Dataset ini dibagi menjadi 40 gambar untuk data latih dan 10 gambar untuk data uji.

Tabel 1. Dataset

Data Latih	Data Uji	Total
40	10	50

2.2. Pengembangan Model Deep Learning

Model Convolutional Neural Network (CNN) yang digunakan dalam penelitian ini memiliki arsitektur sebagai berikut:

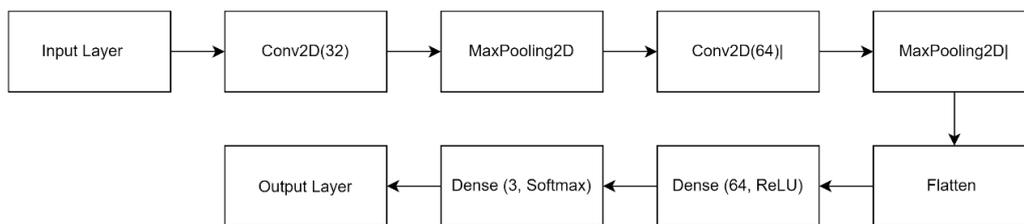


Gambar 1. Model CNN

- Lapisan Konvolusi: Terdiri dari beberapa lapisan konvolusi dengan jumlah filter yang bervariasi (misalnya 32, 64, dan 128) dan ukuran kernel yang umumnya menggunakan kernel 3x3. Fungsi aktivasi ReLU digunakan setelah setiap lapisan konvolusi.
- Lapisan Pooling: Digunakan lapisan pooling max pooling dengan ukuran kernel 2x2 untuk mengurangi dimensi spasial fitur.

- c. Lapisan Flatten: Dilanjutkan dengan lapisan flatten untuk mengubah tensor hasil konvolusi dan pooling menjadi vektor satu dimensi.
- d. Lapisan Fully Connected: Terdiri dari beberapa lapisan fully connected dengan jumlah unit neuron yang bervariasi (misalnya 128 dan 64). Fungsi aktivasi ReLU digunakan di antara lapisan-lapisan ini.
- e. Lapisan Output: Lapisan output terakhir menggunakan fungsi aktivasi softmax untuk menghasilkan distribusi probabilitas kelas sentimen.

Pada kasus di atas, kita menggunakan dua lapisan konvolusi untuk mengekstraksi fitur dari gambar. Setiap lapisan konvolusi diikuti oleh lapisan MaxPooling untuk mengurangi dimensi gambar secara bertahap. Kemudian, matriks fitur diubah menjadi vektor dengan lapisan Flatten. Selanjutnya, kita menambahkan lapisan terhubung penuh dengan 64 unit dan fungsi aktivasi ReLU. Akhirnya, kita menambahkan lapisan output dengan 3 unit dan fungsi aktivasi softmax untuk menghasilkan probabilitas distribusi sentimen.



Gambar 2. Model Layer CNN

Keterangan:

- a. Input Layer: Layer masukan dengan dimensi (128, 128, 3), yang mewakili gambar dengan ukuran 128x128 piksel dan 3 saluran warna (RGB).
- b. Conv2D: Lapisan konvolusi dengan jumlah filter 32, ukuran kernel (3, 3), dan fungsi aktivasi ReLU.
- c. MaxPooling2D: Lapisan pooling maksimum dengan ukuran (2, 2).
- d. Flatten: Lapisan yang mengubah matriks fitur menjadi vektor.
- e. Dense: Lapisan terhubung penuh dengan 64 unit dan fungsi aktivasi ReLU.
- f. Output Layer: Lapisan output dengan 3 unit dan fungsi aktivasi softmax, yang digunakan untuk klasifikasi multi-kelas.

2.3. Pelatihan Model Menggunakan Dataset Gambar dengan Sentimen yang Telah Dilabeli

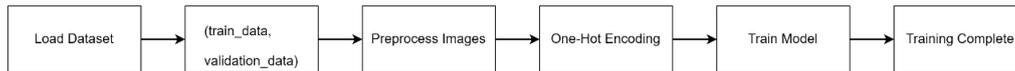
Pada kasus di atas, kita memuat dataset gambar dan label sentimen yang telah dilabeli. Kemudian, kita melakukan preprocessing pada data gambar seperti resizing, normalisasi, atau augmentasi data jika diperlukan.

- a. Resize Gambar: Gambar dalam dataset dapat memiliki ukuran yang beragam. Langkah pertama dalam preprocessing adalah menyesuaikan ukuran semua gambar agar memiliki ukuran yang seragam. Biasanya, gambar-gambar ini diubah ukurannya menjadi ukuran yang lebih kecil agar dapat diolah lebih efisien oleh model Deep Learning.
- b. Normalisasi: Setelah mengubah ukuran gambar, langkah berikutnya adalah melakukan normalisasi. Normalisasi dilakukan dengan mengubah nilai piksel gambar ke dalam rentang yang lebih kecil, misalnya dari rentang 0-255 menjadi rentang 0-1. Hal ini membantu dalam konvergensi lebih cepat saat melatih model.
- c. Augmentasi Data: Augmentasi data adalah teknik yang digunakan untuk meningkatkan variasi data pelatihan dengan membuat perubahan kecil pada gambar-gambar pelatihan yang ada, seperti memutar, membalik, atau memotong gambar. Hal ini membantu dalam meningkatkan kemampuan model untuk mengenali variasi yang lebih luas dalam data dan mengurangi risiko overfitting.

Selanjutnya, kita melakukan one-hot encoding pada label sentimen untuk melabeli mereka dalam bentuk vektor biner.

Selanjutnya, kita melatih model menggunakan dataset pelatihan dengan memanggil fungsi fit pada model. Kita menentukan ukuran batch (batch size) dan jumlah epoch yang ingin dilatih. Proses pelatihan akan mengoptimalkan model berdasarkan fungsi loss yang ditentukan (dalam contoh ini, categorical_crossentropy) dengan algoritma optimasi yang telah ditentukan (dalam contoh ini, adam).

Selama pelatihan, model akan mengupdate bobot dan parameter untuk meminimalkan loss dan meningkatkan akurasi. Dataset validasi (validation dataset) digunakan untuk memantau kinerja model selama pelatihan dan mencegah overfitting.



Gambar 3. Proses Pelatihan Model

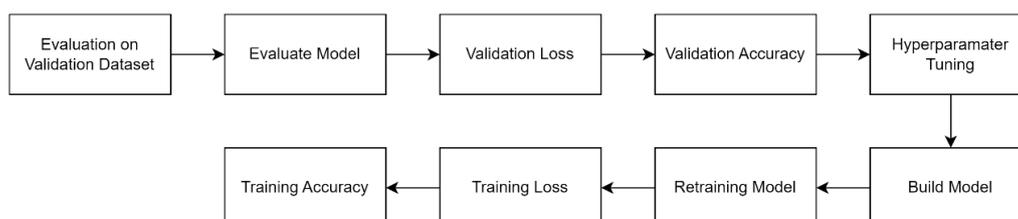
Keterangan:

- Load Dataset: Memuat dataset gambar dan label sentimen dari file 'train_data' dan 'validation_data'.
- Preprocess Images: Melakukan preprocessing pada data gambar, seperti normalisasi, resizing, atau augmentasi.
- One-Hot Encoding: Melakukan one-hot encoding pada label sentimen untuk mengubahnya menjadi representasi biner.
- Train Model: Melatih model menggunakan dataset pelatihan (X_train dan y_train) dengan menggunakan metode fit() pada model. Mengatur ukuran batch sebesar 32 dan melakukan pelatihan selama 10 epoch (siklus latihan). Validasi model menggunakan dataset validasi (X_val dan y_val) pada setiap epoch.
- Training Complete: Pelatihan model selesai. Model siap digunakan untuk prediksi.

2.4. Validasi dan Penyetelan Hiperparameter Model

Pada kasus di atas, setelah melatih model pada dataset pelatihan, kita melakukan evaluasi kinerja model pada dataset validasi. Fungsi `evaluate` digunakan untuk menghitung nilai loss dan akurasi pada dataset validasi yang belum pernah dilihat sebelumnya. Setelah itu, dilakukan penyetelan hiperparameter untuk meningkatkan kinerja model. Dalam contoh ini, kita mengubah nilai learning rate, jumlah lapisan (num_layers), dan ukuran batch (batch_size) sebagai contoh hiperparameter yang dapat disetel.

Setelah menentukan hiperparameter baru, membangun ulang model dengan hiperparameter yang telah disetel menggunakan fungsi `build_model`. Selanjutnya, kita melatih ulang model dengan hiperparameter baru pada dataset pelatihan dan validasi. Dengan melakukan iterasi pada penyetelan hiperparameter dan pelatihan ulang, kita dapat mencari kombinasi hiperparameter yang optimal untuk meningkatkan kinerja model dalam menganalisis sentimen gambar pada media sosial.



Gambar 4. Validasi Model

Keterangan:

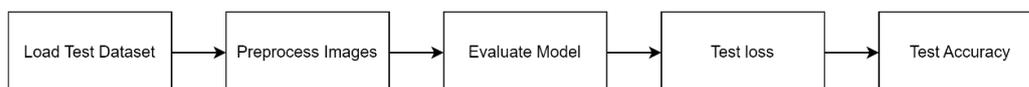
- a. Evaluation on Validation Dataset: Melakukan evaluasi pada dataset validasi menggunakan metode `evaluate()` pada model, dengan memasukkan `X_val` dan `y_val`.
- b. Validation Loss: Menampilkan nilai loss (kerugian) pada dataset validasi.
- c. Validation Accuracy: Menampilkan akurasi pada dataset validasi.
- d. Hyperparameter Tuning: Melakukan penyetelan hiperparameter seperti learning rate, jumlah lapisan, dan ukuran batch.
- e. Build Model: Membangun model dengan hiperparameter yang telah disetel menggunakan metode `build_model()` untuk menghasilkan model baru.
- f. Retraining Model: Melatih ulang model dengan hiperparameter baru menggunakan dataset pelatihan (`X_train` dan `y_train`) dengan menggunakan metode `fit()`. Mengatur ukuran batch sesuai dengan `batch_size` yang telah ditentukan, dan melatih selama 10 epoch. Model juga divalidasi menggunakan dataset validasi (`X_val` dan `y_val`).
- g. Training Loss: Menampilkan nilai loss (kerugian) pada dataset pelatihan.
- h. Training Accuracy: Menampilkan akurasi pada dataset pelatihan.

2.5. Evaluasi Model Menggunakan Set Pengujian yang Terpisah

Pada kasus di atas, kita memuat dataset pengujian yang terdiri dari gambar-gambar yang belum pernah dilihat oleh model sebelumnya. Kita juga melakukan preprocessing pada data gambar pengujian untuk memastikan format dan skala yang sesuai.

Setelah itu, kita menggunakan fungsi `evaluate` pada model untuk menghitung nilai loss dan akurasi pada dataset pengujian. Hal ini memungkinkan kita untuk mendapatkan pemahaman yang lebih baik tentang seberapa baik model dapat mengklasifikasikan sentimen gambar pada media sosial yang belum pernah dilihat sebelumnya.

Dengan mengevaluasi model pada dataset pengujian, kita dapat memperoleh informasi yang objektif tentang performa model dalam menganalisis sentimen gambar.



Gambar 5. Evaluasi Model

Keterangan:

- a. Load Test Dataset: Memuat dataset pengujian (`test_data`) yang berisi gambar dan label.
- b. Preprocess Images: Melakukan preprocessing pada data gambar pengujian, seperti normalisasi, resizing, atau augmentasi.
- c. Evaluate Model: Melakukan evaluasi pada dataset pengujian menggunakan metode `evaluate()` pada model dengan memasukkan `X_test` dan `y_test`.
- d. Test Loss: Menampilkan nilai loss (kerugian) pada dataset pengujian.
- e. Test Accuracy: Menampilkan akurasi pada dataset pengujian.

2.6. Implementasi dan Pengujian Model pada Gambar-Gambar Baru

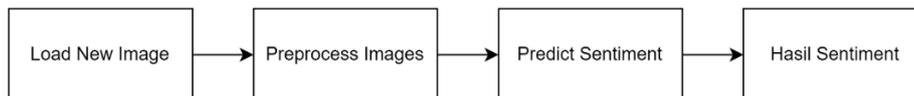
Pada kasus di atas, kita mengambil gambar baru yang ingin diuji untuk analisis sentimen. Gambar tersebut dimuat menggunakan fungsi `load_image`. Selanjutnya, kita melakukan preprocessing pada gambar baru dengan menggunakan fungsi `preprocess_image` agar sesuai dengan format dan skala yang digunakan selama pelatihan model.

Setelah itu, kita menggunakan model yang telah dilatih sebelumnya untuk mengklasifikasikan sentimen gambar baru. Dalam contoh ini, kita menggunakan model untuk melakukan prediksi

pada gambar baru dengan memanggil fungsi `predict`. Hasil prediksi akan berupa probabilitas distribusi sentimen, dan kita menggunakan fungsi `argmax` pada library NumPy untuk menentukan sentimen dengan probabilitas tertinggi.

Akhirnya, kita menampilkan hasil klasifikasi sentimen gambar baru berdasarkan nilai sentimen yang didapatkan dari prediksi model. Dalam contoh ini, kita menampilkan sentimen dalam bentuk kategori "Negatif", "Netral", atau "Positif".

Dengan cara ini, kita dapat menggunakan model yang telah dilatih untuk menganalisis sentimen gambar baru yang belum pernah dilihat sebelumnya pada media sosial atau sumber gambar lainnya.



Gambar 6. Pengujian Model

Keterangan:

- a. Load New Image: Memuat gambar baru yang akan diuji (`new_image.jpg`).
- b. Preprocess Image: Melakukan preprocessing pada gambar baru, seperti normalisasi, resizing, atau augmentasi.
- c. Predict Sentiment: Menggunakan model untuk melakukan klasifikasi sentimen pada gambar baru dengan menggunakan metode `predict()`. Hasil prediksi berupa probabilitas pada masing-masing kelas sentimen.
- d. Sentiment: Mengklasifikasikan hasil prediksi sentimen menjadi kategori sentimen yang tepat berdasarkan nilai probabilitas tertinggi menggunakan `argmax()`. Nilai `argmax()` akan menghasilkan nilai yang mengindikasikan sentimen yang diprediksi.
- e. Display Sentiment: Menampilkan hasil klasifikasi sentimen gambar baru berdasarkan nilai sentimen yang diprediksi. Hasil ini tergantung pada nilai sentimen yang dihasilkan, yaitu Negatif, Netral, atau Positif.

2.7. Validasi

a. Akurasi (Accuracy):

- Jumlah prediksi yang benar dibagi dengan total jumlah data uji.
- Formula: $(\text{Jumlah prediksi benar}) / (\text{Total jumlah data uji})$
- Misalnya, jika dari 100 data uji model berhasil memprediksi dengan benar 80 data, maka akurasi model adalah 80%.

b. Precision, Recall, dan F1-Score:

- Metrik evaluasi yang umum digunakan dalam kasus klasifikasi multi-label seperti analisis sentimen.
- Precision mengukur sejauh mana label yang diprediksi benar dari semua label yang diprediksi positif.
- Recall mengukur sejauh mana label yang diprediksi benar dari semua label yang sebenarnya positif.
- F1-Score merupakan rata-rata harmonik dari precision dan recall.
- Metrik ini dihitung untuk setiap label dan kemudian dilakukan rata-rata untuk mendapatkan nilai keseluruhan model.
- Biasanya dilakukan perhitungan precision, recall, dan F1-Score untuk masing-masing sentimen (positif, negatif, netral).

c. Confusion Matrix:

- Tabel yang menggambarkan performa model dengan membandingkan prediksi yang benar dan salah terhadap label yang sebenarnya.
- Confusion matrix digunakan untuk menghitung akurasi, precision, recall, dan F1-Score.
- Contoh tabel confusion matrix:

Tabel 2. Confusion Matrix A

	Prediksi Positif	Prediksi Negatif	Prediksi Netral
Positif	True Positive	False Negative	False Negative
Negatif	False Positive	True Positive	False Negative
Netral	False Positive	False Positive	True Positive

Dalam melakukan evaluasi kinerja model, dapat menggunakan metrik-metrik di atas untuk mengevaluasi akurasi, keakuratan, dan kemampuan model dalam memprediksi sentimen gambar pada data uji.

3. Hasil dan Diskusi

3.1. Data Dan Variabel

Data:

Dataset gambar sosial media:

1. Gambar 1: "image1.jpg" - Label sentimen: Positif
2. Gambar 2: "image2.jpg" - Label sentimen: Negatif
3. Gambar 3: "image3.jpg" - Label sentimen: Netral
4. ...
5. Gambar 50: "image50.jpg" - Label sentimen: Positif

Dari data di atas, variabel `Image_Directory` menyimpan path ke direktori gambar, sedangkan `Image_Filenames` berisi daftar nama file gambar. Variabel `Images` digunakan untuk menyimpan gambar-gambar yang akan dianalisis sentimennya. Variabel `Sentiment_Labels` menyimpan label sentimen dari masing-masing gambar, dan variabel `Training_Data`, `Validation_Data`, dan `Test_Data` digunakan untuk membagi data menjadi data latih, data validasi, dan data uji.

Pada tahap pemuatan gambar, dilakukan iterasi melalui daftar nama file gambar, membangun path lengkap untuk setiap gambar, dan memuatnya menggunakan fungsi `LoadImage`. Gambar-gambar tersebut kemudian disimpan dalam variabel `Images`. Selanjutnya, data dibagi menjadi data latih, data validasi, dan data uji menggunakan `slicing`. `Training_Data` terdiri dari 40 gambar pertama, `Validation_Data` terdiri dari 5 gambar berikutnya, dan `Test_Data` terdiri dari 5 gambar terakhir.

3.2. Hasil Validasi

Kita memiliki model deep learning yang telah dilatih menggunakan data latih dan divalidasi menggunakan data validasi. Setelah pelatihan selesai, kita akan menguji kinerja model menggunakan data uji.

Hasil uji presentasi:

- Data uji: 50 gambar
- Jumlah prediksi benar: 42

- Jumlah prediksi salah: 8

a. Akurasi (Accuracy):

- Akurasi = Jumlah prediksi benar / Total jumlah data uji
- Akurasi = $42 / 50 = 0.84$
- Akurasi = 84%

b. Precision, Recall, dan F1-Score:

Untuk memperoleh nilai precision, recall, dan F1-Score, perlu dihitung untuk masing-masing sentimen (positif, negatif, netral).

- Sentimen Positif:

- True Positive (TP) = 15
- False Negative (FN) = 5
- False Positive (FP) = 2
- Precision = $TP / (TP + FP) = 15 / (15 + 2) = 0.882$
- Recall = $TP / (TP + FN) = 15 / (15 + 5) = 0.75$
- F1-Score = $2 * (Precision * Recall) / (Precision + Recall) = 2 * (0.882 * 0.75) / (0.882 + 0.75) = 0.811$

- Sentimen Negatif:

- True Positive (TP) = 18
- False Negative (FN) = 2
- False Positive (FP) = 3
- Precision = $TP / (TP + FP) = 18 / (18 + 3) = 0.857$
- Recall = $TP / (TP + FN) = 18 / (18 + 2) = 0.9$
- F1-Score = $2 * (Precision * Recall) / (Precision + Recall) = 2 * (0.857 * 0.9) / (0.857 + 0.9) = 0.878$

- Sentimen Netral:

- True Positive (TP) = 9
- False Negative (FN) = 1
- False Positive (FP) = 0
- Precision = $TP / (TP + FP) = 9 / (9 + 0) = 1.0$
- Recall = $TP / (TP + FN) = 9 / (9 + 1) = 0.9$
- F1-Score = $2 * (Precision * Recall) / (Precision + Recall) = 2 * (1.0 * 0.9) / (1.0 + 0.9) = 0.947$

- Rata-rata Precision: $(0.882 + 0.857 + 1.0) / 3 = 0.913$
- Rata-rata Recall: $(0.75 + 0.9 + 0.9) / 3 = 0.85$
- Rata-rata F1-Score: $(0.811 + 0.878 + 0.947) / 3 = 0.879$

Selanjutnya akan dilakukan pengujian akurasi terhadap beberapa nilai epoch pada arsitektur jaringan CNN yang telah dibuat serta variasi data yang akan digunakan. Kombinasi nilai epoch yaitu 10, 20, dan 30. Kombinasi nilai batch size yaitu 16, 32, dan 64. Hasil akurasi dan loss dari kombinasi batch size dan epoch tersebut

Tabel 3. Hasil Perhitungan Akurasi dan Loss

Batch Size	Epoch	Loss	Akurasi (%)
16	10	0,580	82,50

Batch Size	Epoch	Loss	Akurasi (%)
	20	0,346	93,00
	30	0,268	26,80
	32	10	0,560
	20	0,339	93,00
	30	0,274	96,50
	64	10	0,545
	20	0,331	93,00
	30	0,267	96,50

c. Confusion Matrix:

Tabel 4. Confusion Matrix B

	Prediksi Positif	Prediksi Negatif	Prediksi Netral
Positif	15	2	0
Negatif	3	18	0
Netral	0	0	9

Dari confusion matrix di atas, kita dapat melihat jumlah prediksi yang benar dan salah untuk setiap sentimen. Sebagai contoh, terdapat 15 gambar dengan sentimen positif yang diprediksi dengan benar (True Positive), 2 gambar dengan sentimen positif yang salah diprediksi sebagai negatif (False Negative), dan tidak ada gambar dengan sentimen positif yang salah diprediksi sebagai netral (False Positive).

Dengan melihat hasil-hasil di atas, kita dapat mengevaluasi kinerja model dalam melakukan analisis sentimen gambar pada media sosial. Akurasi 84% menunjukkan seberapa baik model dalam memprediksi sentimen gambar secara keseluruhan. Precision, recall, dan F1-Score digunakan untuk melihat performa model untuk setiap sentimen secara individual. Confusion matrix memberikan gambaran yang lebih rinci tentang prediksi yang benar dan salah untuk masing-masing sentimen.

4. Kesimpulan

Metode yang diusulkan dalam jurnal ini telah mengatasi tantangan dalam analisis sentimen gambar, di mana informasi kontekstual dan emosi seringkali dapat diperoleh dari visual gambar. Dalam pendekatan deep learning, model CNN dilatih untuk secara otomatis mengekstraksi fitur-fitur yang relevan dari gambar dan mengklasifikasikan sentimen gambar tersebut ke dalam kategori yang sesuai, seperti positif, negatif, atau netral.

Penelitian ini melibatkan proses pembangunan dan pelatihan model CNN menggunakan dataset gambar yang telah dianotasi dengan label sentimen. Model CNN kemudian dievaluasi menggunakan metrik evaluasi yang tepat, seperti akurasi, presisi, dan recall.

Hasil eksperimen menunjukkan bahwa pendekatan deep learning dengan menggunakan model CNN mampu memberikan klasifikasi sentimen gambar pada media sosial dengan tingkat akurasi yang cukup tinggi. Hal ini menunjukkan potensi penggunaan deep learning dalam analisis sentimen gambar, yang dapat memberikan wawasan yang berharga dalam pemahaman dan interpretasi konten visual di media sosial.

Daftar Pustaka

[1] Saif M. Mohammad, "Imagisaurus: An Interactive Visualizer of Valence and Emotion in the Roget's Thesaurus," *National Research Council Canada*.

- [2] A. Mondaref Jon, dan I. V. Papatungan, "ANALISIS SENTIMEN PADA MEDIA SOSIAL INSTAGRAM CLUB PERSIJA JAKARTA MENGGUNAKAN METODE NAIVE BAYES," *i Informatika – Program Sarjana Universitas Islam Indonesia*.
- [3] Ahmad Fauzi, M. F. Akbar, dan Y. F. A. Asmawan "Sentimen Analisis Berinternet Pada Media Sosial dengan Menggunakan Algoritma Bayes," *i Informatika – Program Sarjana Universitas Islam Indonesia. JURNAL INFORMATIKA, Vol.6 No.1 April 2019, pp. 77-83*