

**ANALISIS SENTIMEN TEXT DENGAN METODE CNN STUDY KASUS  
TEMPAT WISATA MAKASSAR**

**SKRIPSI**

Diajukan sebagai Salah Satu Syarat Untuk Mendapatkan Gelar Sarjana Komputer  
(S.Kom) Program Studi Informatika



**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS MUHAMMDIYAH MAKASSAR  
2024**

**ANALISIS SENTIMEN TEXT DENGAN METODE CNN STUDY KASUS  
TEMPAT WISATA MAKASSAR**

Diajukan sebagai Salah Satu Syarat Untuk Mendapatkan Gelar Sarjana Komputer  
(S.Kom) Program Studi Informatika



**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS MUHAMMDIYAH MAKASSAR  
2024**



## FAKULTAS TEKNIK

GEDUNG MENARA IQRA LT. 3

Jl. Sultan Alauddin No. 259 Telp. (0411) 866 972 Fax (0411) 865 588 Makassar 90221

Website: [www.unismuh.ac.id](http://www.unismuh.ac.id), e-mail: [unismuh@gmail.com](mailto:unismuh@gmail.com)

Website: <http://teknik.unismuh.makassar.ac.id>

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

### PENGESAHAN

Skripsi atas nama Safutri Kamal dengan nomor induk Mahasiswa 105 84 11078 20, dinyatakan diterima dan disahkan oleh Panitia Ujian Tugas Akhir/Skripsi sesuai dengan Surat Keputusan Dekan Fakultas Teknik Universitas Muhammadiyah Makassar Nomor : 016/05/A.5-VI/I/45/2024, sebagai salah satu syarat guna memperoleh gelar Sarjana Komputer pada Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar pada hari Sabtu tanggal 25 Mei 2024.

Panitia Ujian :

Makassar, 17 Dzulqaidah 1445 H

25 Mei 2024 M

1. Pengawas Umum

a. Rektor Universitas Muhammadiyah Makassar

Prof. Dr. H. Ambo Asse, M.Ag

b. Dekan Fakultas Teknik Universitas Hasanuddin

Prof. Dr. Eng. Muhammad Isran Ramli, ST., MT.

2. Penguji

a. Ketua : Dr. Ir. Zahir Zainuddin, M.Sc

b. Sekertaris : Rizki Yusliana Bakti, S.Kom. MT

3. Anggota

: 1. Muhyiddin A. M. Hayat, S.Kom. MT

2. Lukman S.Kom.,MT

3. Desi Anggraeni, S.Kom. MT

Mengetahui :

Pembimbing I

Pembimbing II

Fahrim Irhamna Rahman S.Kom.,MT

Titin Wahyuni, S.Pd.,MT



Dekan

Dr. Ir. H. Nurnawaty ST., MT., IPM.  
NBM : 795 108



Dipindai dengan CamScanner



## FAKULTAS TEKNIK

GEDUNG MENARA IQRA LT. 3

Jl. Sultan Alauddin No. 259 Telp. (0411) 866 972 Fax (0411) 865 588 Makassar 90221

Website: [www.unismuh.ac.id](http://www.unismuh.ac.id), e-mail: [unismuh@gmail.com](mailto:unismuh@gmail.com)

Website: <http://teknik.unismuh.makassar.ac.id>

جَنَاحُ الْكَوَافِرِ

### HALAMAN PENGESAHAN

Tugas Akhir ini diajukan untuk memenuhi syarat ujian guna memperoleh gelar Sarjana Komputer (S.Kom) Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.

Judul Skripsi : ANALISIS SENTIMEN TEXT DENGAN METODE CNN STUDY KASUS TEMPAT WISATA MAKASSAR

Nama : Safutri Kamal

Stambuk : 105841107820

Makassar, 25 Mei 2024

Telah Diperiksa dan Disetujui  
Oleh Dosen Pembimbing;

Pembimbing I

Pembimbing II

Fahrim Irhamna Rahman S.Kom.,MT

Titin Wahyuni, S.Pd.,MT

Mengetahui,

Ketua Program Studi Arsitektur

Muhammadin A.M. Hayat, S.Kom. ,MT  
NBM : -



Dipindai dengan CamScanner

## MOTTO DAN PERSEMBAHAN

### *Motto*

**“Dan tidaklah sama kebaikan dengan kejahanan. Tolaklah (kejahanan itu) dengan cara yang lebih baik, sehingga orang yang ada rasa permusuhan antara kamu dan dia akan seperti teman yang setia.”**

**-QS. FUSSILAT : 34-**

### *Persembahan*

*Kupersembahkan skripsi ini kepada orang – orang yang sangat kusayangi.*

*Karya ini merupakan bentuk rasa Syukur kepada Allah SWT. yang telah memberikan nikmat iman, kesehatan, dan kesempatan untuk menyelesaikan skripsi ini. Serta terimakasih kepada Ayahanda Kamaluddin atas segala usaha dan kerja kerasmu untuk memastikan bahwa saya mendapatkan pendidikan terbaik. Keteguhan dan kearifanmu selalu menjadi teladan bagi saya.*

*Terimakasih juga kepada Ibunda Nurlia, meskipun Engkau telah tiada, kasih sayangmu, doa-doamu, dan pengorbananmu selalu menyertai setiap langkahku. Kehadiranmu di dalam hati dan ingatanku memberikan kekuatan dan inspirasi yang tiada terhingga. Dan yang terakhir untuk Kakak saya Ismiyanti S.Mat dan Irawati K Amd.Keb, Terima kasih atas dukungan, motivasi, dan kasih sayang yang telah kau berikan. Kakak selalu menjadi contoh dan sumber inspirasi bagiku.*

## ABSTRAK

**SAFUTRI KAMAL.** Analisis sentimen text dengan metode CNN study kasus tempat wisata Makassar (dibimbing oleh Fahriz Irhamna Rachman S.Kom.,M.T. dan Titin Wahyuni S.Pd.,M.T).

Penelitian ini bertujuan untuk mengevaluasi dan menentukan sejauh mana metode CNN (*Convolutional Neural Network*) dapat menghasilkan prediksi sentimen yang akurat terhadap ulasan mengenai tempat wisata Makassar. Metode analisis sentimen ini menggunakan data ulasan yang dikumpulkan dari platform Google Maps. Dalam penelitian ini, dilakukan tahap preprocessing untuk membersihkan data, seperti *cleaning*, *transform cases*, *tokenizing*, *stopword* dan *stemming*. Selanjutnya, dilakukan pembagian dataset menjadi data latih dan data uji dengan *scenario* 90 : 10, 80 : 20 dan 70 : 30 untuk melatih dan menguji model dengan tiga kategori ulasan yaitu positif, negatif dan netral. Hasil dari analisis sentimen menunjukkan bahwa metode CNN memiliki kemampuan yang baik dalam memprediksi sentimen positif, negatif, dan netral pada ulasan mengenai Tempat Wisata Makassar. Tingkat akurasi yang tinggi pada tahap pelatihan menunjukkan bahwa model mampu belajar dengan baik dari dataset yang disediakan. Meskipun tingkat akurasi pada tahap validasi sedikit lebih rendah, tetapi masih mencapai angka yang memadai, menunjukkan bahwa model memiliki kemampuan generalisasi yang cukup baik dalam mengklasifikasikan sentimen pada ulasan-ulasan tersebut. Diperoleh hasil akurasi tertinggi dengan *Training Accuracy* yang meningkat memperoleh nilai akurasi training 95%, serta *Validation Accuracy* memperoleh nilai 73%.

**Kata Kunci :** Analisis sentimen, Tempat wisata Makassar, *Convolutional Neural Network*, Ulasan Google Maps

## ***ABSTRACT***

**SAFUTRI KAMAL.** *Text sentiment analysis using the CNN case study method of Makassar tourist attractions (supervised by Fahrin Irhamna Rachman S.Kom., M.T. and Titin Wahyuni S.Pd., M.T).*

*This research aims to evaluate and determine the extent to which the CNN (Convolutional Neural Network) method can produce accurate sentiment predictions for reviews of Makassar tourist attractions. This sentiment analysis method uses review data collected from the Google Maps platform. In this research, a preprocessing stage was carried out to clean the data, such as cleaning, transform cases, tokenizing, stopwords and stemming. Next, the dataset was divided into training data and test data with scenarios 90 : 10, 80 : 20 and 70 : 30 to train and test the model with three review categories, namely positive, negative and neutral. The results of sentiment analysis show that the CNN method has good abilities in predicting positive, negative and neutral sentiment in reviews of Makassar tourist attractions. A high level of accuracy at the training stage shows that the model is able to learn well from the dataset provided. Although the accuracy rate at the validation stage was slightly lower, it still reached an adequate figure, indicating that the model has quite good generalization ability in classifying sentiment in these reviews. The highest accuracy results were obtained with Training Accuracy which increased to obtain a training accuracy value of 95%, and Validation Accuracy obtained a value of 73%.*

**Keywords :** *Sentiment analysis, Makassar tourist attractions, Convolutional Neural Network, Google Maps reviews*

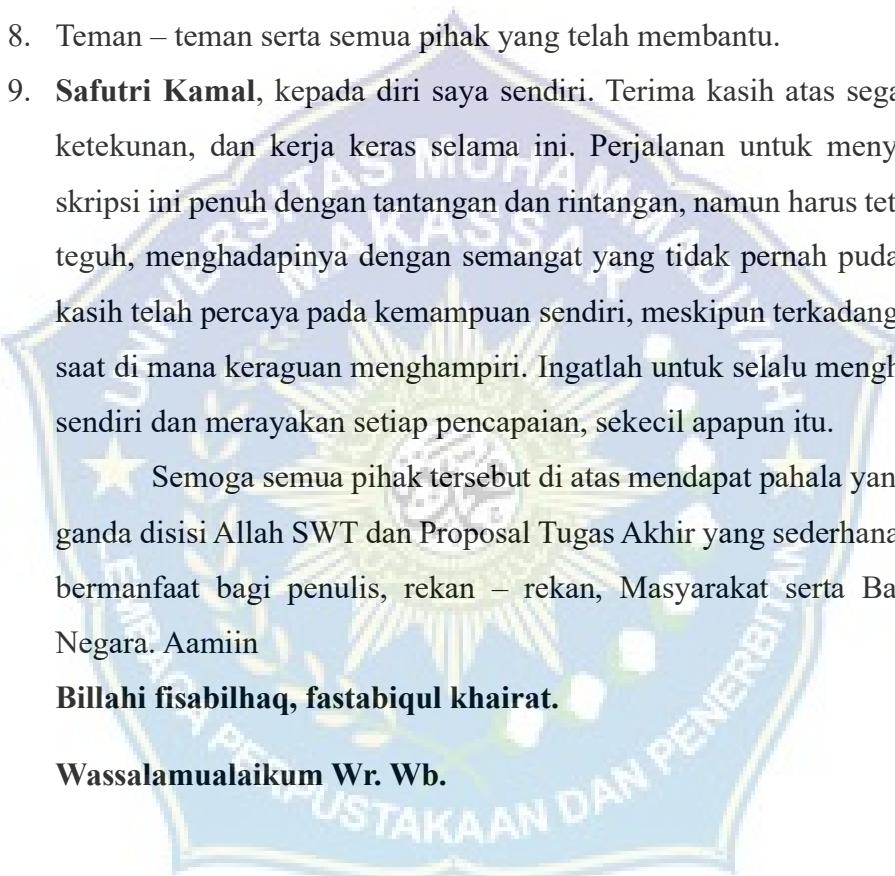
## KATA PENGANTAR

Segala Puji bagi Allah SWT atas limpahan rahmat, kesehatan dan kekuatannya sehingga proposal skripsi dengan judul “ANALISIS SENTIMEN TEXT DENGAN METODE CNN STUDY KASUS TEMPAT WISATA MAKASSAR” ini dapat kami selesaikan sebagaimana salah satu syarat untuk penyusunan skripsi Program Studi Informatika. Shalawat dan junjungan Nabi Besar Muhammad SAW sebagai uswatan hasanah dan rahmatan lil alamin.

Penulis menyadari bahwa di dalam penulisan proposal bukanlah tujuan akhir dari suatu pembelajaran, penulis juga menyadari masih jauh dari kata sempurna baik dari segi isi, bahasa maupun dari segi penulisannya, hal ini disebabkan keterbatasan penulis dari segi pengetahuan. Untuk itu penulis mohon maaf atas segala kekurangan tersebut. Dan tidak menutup kemungkinan untuk segala saran dan kritik serta masukan yang bersifat membangun bagi diri penulis.

Tak lupa kami mengucapkan banyak terima kasih kepada semua pihak yang telah membantu dan memberikan motivasi dalam penyusunan proposal ini, terutama kepada :

1. Kedua orang tua yang tercinta, yaitu **Kamaluddin** dan **Nurlia**. Penulis mengucapkan terima kasih yang sebesar-besarnya atas segala doa, kasih sayang dan dukungan baik secara moral maupun materi.
2. Bapak **Prof. DR. H. Ambo Asse, M.Ag.** sebagai Rektor Universitas Muhammadiyah Makassar.
3. Ibu **Dr. Nurnawaty, S.T., M.T. IPM.** Sebagai Dekan Fakultas Teknik Universitas Muhammadiyah Makassar.
4. Bapak **Muhyiddin M Hayat, S.Kom., M.T** sebagai Ketua Prodi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.
5. Bapak **Fachrim Irhamna Rahman S.Kom., M.T** sebagai pembimbing I yang dengan telah Ikhlas memberikan bimbingan dan arahan selama penyusunan tugas akhir ini.

- 
6. Ibu **Titin Wahyuni, S.Pd., M.T** sebagai pembimbing II yang dengan telah Ikhlas memberikan bimbingan dan arahan selama penyusunan tugas akhir ini.
  7. Segenap Bapak – bapak dan Ibu Dosen Prodi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar yang telah memberikan bakat dan ilmu pengetahuan serta mendidik penulis selama proses belajar mengajar di Universitas Muhammadiyah Makassar.
  8. Teman – teman serta semua pihak yang telah membantu.
  9. **Safutri Kamal**, kepada diri saya sendiri. Terima kasih atas segala usaha, ketekunan, dan kerja keras selama ini. Perjalanan untuk menyelesaikan skripsi ini penuh dengan tantangan dan rintangan, namun harus tetap berdiri teguh, menghadapinya dengan semangat yang tidak pernah pudar. Terima kasih telah percaya pada kemampuan sendiri, meskipun terkadang ada saat-saat di mana keraguan menghampiri. Ingatlah untuk selalu menghargai diri sendiri dan merayakan setiap pencapaian, sekecil apapun itu.

Semoga semua pihak tersebut di atas mendapat pahala yang berlipat ganda disisi Allah SWT dan Proposal Tugas Akhir yang sederhana ini dapat bermanfaat bagi penulis, rekan – rekan, Masyarakat serta Bangsa dan Negara. Aamiin

**Billahi fisabilhaq, fastabiqul khairat.**

**Wassalamualaikum Wr. Wb.**

Makassar, 25 Mei 2024

Safutri Kamal

## DAFTAR ISI

HALAMAN JUDUL.....	
HALAMAN PENGESAHAN.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PERSEMBAHAN.....	v
ABSTRAK .....	vi
ABSTRACT .....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI .....	x
DAFTAR GAMBAR .....	xii
DAFTAR TABEL.....	xiv
DAFTAR LAMPIRAN .....	xv
DAFTAR ISTILAH .....	xvi
BAB I PENDAHULUAN.....	1
A. Latar Belakang .....	1
B. Rumusan Masalah .....	2
C. Tujuan Penelitian.....	3
D. Manfaat Penelitian .....	3
E. Ruang Lingkup Penelitian.....	4
F. Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA .....	6
A. Landasan Teori .....	6
B. Penelitian Terkait.....	12
C. Kerangka Pikir .....	17
BAB III METODE PENELITIAN.....	18
A. Tempat dan Waktu Penelitian.....	18
B. Alat dan Bahan.....	18
C. Perancangan Sistem .....	18
D. Teknik Pengujian Sistem.....	23
E. Teknik Analisis Data .....	24

BAB IV HASIL DAN PEMBAHASAN.....	26
A. Pengambilan Data .....	26
B. Pelabelan Data.....	27
C. Data Preprocessing.....	28
D. Model Convolutional Neural Network (CNN) .....	37
E. Pengujian Metode Convolutional Neural Network (CNN).....	45
BAB V PENUTUP.....	77
A. Kesimpulan .....	77
B. Saran.....	77
DAFTAR PUSTAKA.....	79
LAMPIRAN .....	83



## DAFTAR GAMBAR

Gambar 1. Arsitektur CNN .....	9
Gambar 2. Kerangka Pikir.....	17
Gambar 3. Diagram alur Proses .....	20
Penelitian Analisis Sentimen Text Obyek Wisata .....	20
Gambar 4. Perancangan Sistem Training .....	22
Gambar 5. Perancangan Sistem Testing.....	23
Gambar 6. <i>Scraping</i> Data Ulasan Menggunakan <i>Instant Data Scraper</i> .....	26
Gambar 7. Dataset Ulasan.....	27
Gambar 8. Proses Epoch 1 – 10 .....	50
Gambar 9. Proses Epoch 28 – 39 .....	51
Gambar 10. Proses Epoch 90 – 100 .....	52
Gambar 11. Proses Epoch 1 – 10 .....	52
Gambar 12. Proses Epoch 29 – 42 .....	53
Gambar 13. Proses Epoch 88 – 100 .....	53
Gambar 14. Proses Epoch 1 – 10 .....	54
Gambar 15. Proses Epoch 29 – 42 .....	54
Gambar 16. Proses Epoch 87 – 100 .....	55
Gambar 17. Proses Epoch 1 – 10 .....	55
Gambar 18. Proses Epoch 27 – 40 .....	56
Gambar 19. Proses Epoch 92 – 100 .....	56
Gambar 20. Proses Epoch 1 – 10 .....	56
Gambar 21. Proses Epoch 29 – 42 .....	57
Gambar 22. Proses Epoch 87 – 100 .....	57
Gambar 23. Proses Epoch 1 – 10 .....	58
Gambar 24. Proses Epoch 29 – 42 .....	58
Gambar 25. Proses Epoch 87 – 100 .....	59
Gambar 26. Proses Epoch 1 – 10 .....	59
Gambar 27. Proses Epoch 27 – 39 .....	60
Gambar 28. Proses Epoch 92 – 100 .....	60

Gambar 29. Proses Epoch 1 – 10 .....	60
Gambar 30. Proses Epoch 29 – 42 .....	61
Gambar 31. Proses Epoch 87 – 100 .....	61
Gambar 32. Proses Epoch 1 – 10 .....	62
Gambar 33. Proses Epoch 29 – 42 .....	62
Gambar 34. Proses Epoch 87 – 100 .....	63



## **DAFTAR TABEL**

Tabel 1. Tahap Pelabelan Data .....	27
Tabel 2. Tahap Cleaning Data .....	28
Tabel 3. Tahap Transform Cases .....	30
Tabel 4. Tahap Tokenizing .....	32
Tabel 5. Tahap Stopword.....	34
Tabel 6. Tahap Stemming.....	36
Tabel 7. Pengujian Dataset dengan Mesin .....	63



## DAFTAR LAMPIRAN

Lampiran 1. Source Code Preprocessing .....	83
a. Tahap Cleaning dan Transform Cases.....	83
b. Tahap Tokenizing .....	84
c. Tahap Stopword .....	84
d. Tahap Stemming.....	84
Lampiran 2. Source Code.....	85
Lampiran 3. Dataset Ulasan Positif.....	88
Lampiran 4. Dataset Ulasan Negatif.....	89
Lampiran 5. Dataset Ulasan Netral .....	91
Lampiran 6. Dataset Uji Hasil Prediksi.....	92
Lampiran 7. Dataset Tokenizing .....	94
Lampiran 8. Dataset Stopword.....	94
Lampiran 9. Dataset Stemming.....	95
Lampiran 10. Proses Epoch 1 – 100 dengan Batch Size 32.....	95
a. Pembagian Data 90 : 10 .....	95
b. Pembagian Data 80 : 20 .....	96
c. Pembagian Data 70 : 30 .....	98
Lampiran 11. Proses Epoch 1 – 100 dengan Batch Size 24 .....	99
a. Pembagian Data 90 : 10 .....	99
b. Pembagian Data 80 : 20 .....	100
c. Pembagian Data 70 : 30 .....	102
Lampiran 12. Proses Epoch 1 – 100 dengan Batch Size 64.....	103
a. Pembagian Data 90 : 10 .....	103
b. Pembagian Data 80 : 20 .....	104
c. Pembagian Data 70 : 30 .....	106
Lampiran 13. Hasil Scan Plagiasi Per Bab.....	108

## DAFTAR ISTILAH

Kepariwisataan	Kepariwisataan merujuk pada industri atau sektor yang terkait dengan perjalanan dan wisata. Ini mencakup segala hal yang terkait dengan aktivitas perjalanan, seperti akomodasi, atraksi wisata, transportasi, dan layanan terkait lainnya.
CNN	adalah singkatan dari Convolutional Neural Network. Ini adalah jenis arsitektur jaringan saraf yang sering digunakan dalam pengenalan pola visual, khususnya dalam pengolahan citra dan pengenalan gambar.
Layer	Dalam konteks keilmuan komputer dan pembelajaran mesin, "layer" (lapisan) merujuk pada tingkatan yang berbeda dalam suatu arsitektur jaringan saraf atau model komputasi. Lapisan dapat berisi berbagai jenis operasi atau fungsi, dan mereka bekerja secara bersama-sama untuk memproses informasi.
Fitur visual	Fitur visual merujuk pada aspek-aspek tertentu dalam gambar atau data visual yang diidentifikasi atau diekstraksi oleh algoritma komputer. Fitur visual dapat mencakup bentuk, warna, tekstur, atau karakteristik visual lainnya yang relevan dalam konteks pengolahan citra atau pengenalan pola visual.
Polaritas teks	Polaritas teks mengacu pada arah sentimen atau opini yang terkandung dalam suatu teks. Dalam analisis sentimen, polaritas dapat dinyatakan sebagai positif, negatif, atau netral, tergantung pada evaluasi emosional dari konten teks tersebut.
Jejaring sosial	Jejaring sosial merujuk pada platform atau layanan yang memungkinkan individu untuk terhubung, berinteraksi, dan berbagi informasi dengan orang lain secara daring.

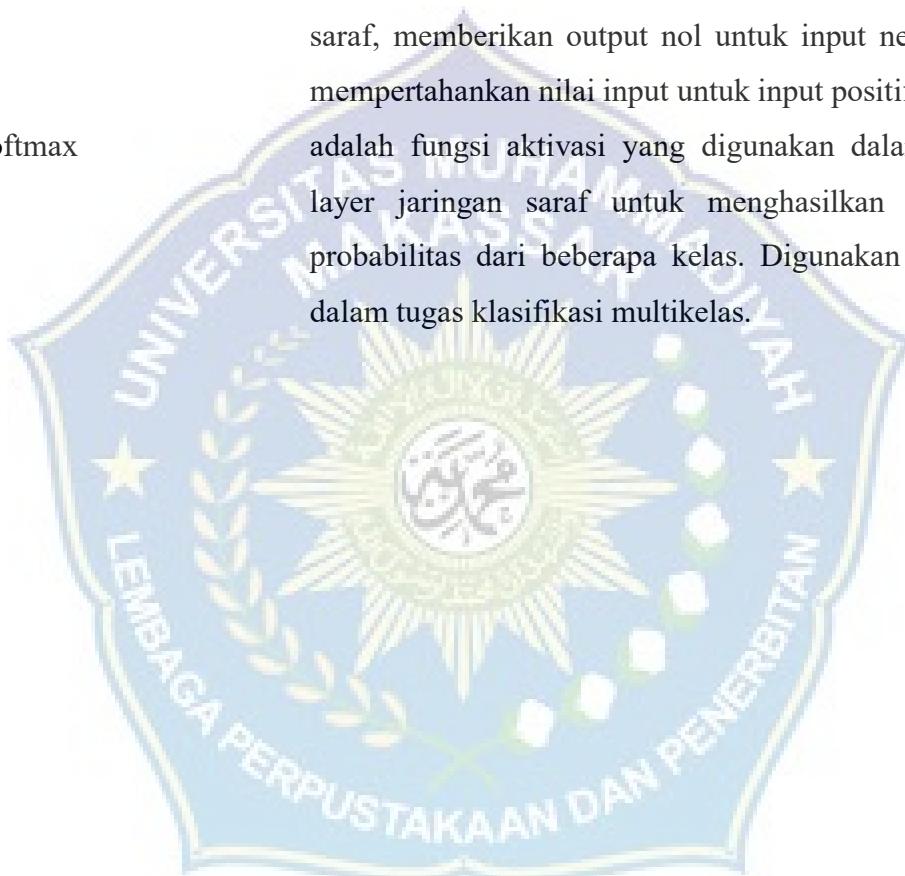
	Contoh jejaring sosial termasuk Facebook, Twitter, dan Instagram.
ANN	adalah singkatan dari Artificial Neural Network (Jaringan Saraf Tiruan). Ini adalah model komputasi yang terinspirasi oleh struktur jaringan saraf biologis dan digunakan dalam berbagai tugas pembelajaran mesin.
Tensorflow	adalah kerangka kerja pembelajaran mesin sumber terbuka yang dikembangkan oleh Google. Ini digunakan untuk mengembangkan dan melatih model pembelajaran mesin, terutama dalam konteks jaringan saraf.
CPU	adalah singkatan dari Central Processing Unit. Ini adalah komponen utama dalam komputer yang bertanggung jawab untuk menjalankan instruksi-instruksi program dan melakukan operasi pemrosesan data.
GPU	adalah singkatan dari Graphics Processing Unit. Meskipun awalnya dirancang untuk memproses grafis, GPU saat ini juga digunakan secara luas untuk tugas-tugas paralel dan komputasi intensif, termasuk dalam pembelajaran mesin.
Flowchart	adalah representasi grafis dari alur kerja atau proses yang menggambarkan langkah-langkah atau aktivitas dalam suatu program atau sistem. Digunakan untuk memvisualisasikan logika operasional secara jelas.
Python	adalah bahasa pemrograman tingkat tinggi yang sering digunakan untuk pengembangan perangkat lunak, analisis data, dan kecerdasan buatan. Python dikenal dengan sintaksis yang mudah dipahami dan berbagai pustaka yang kuat.
API	adalah singkatan dari Application Programming Interface. Ini adalah set instruksi dan definisi yang

memungkinkan aplikasi perangkat lunak berkomunikasi dan berinteraksi satu sama lain.

Epoch Dalam pembelajaran mesin, epoch merujuk pada satu kali siklus melalui seluruh set data pelatihan oleh model pembelajaran mesin selama proses pelatihan.

ReLU adalah singkatan dari Rectified Linear Unit. Ini adalah fungsi aktivasi yang umum digunakan dalam jaringan saraf, memberikan output nol untuk input negatif dan mempertahankan nilai input untuk input positif.

Softmax adalah fungsi aktivasi yang digunakan dalam output layer jaringan saraf untuk menghasilkan distribusi probabilitas dari beberapa kelas. Digunakan terutama dalam tugas klasifikasi multikelas.



# **BAB I**

## **PENDAHULUAN**

### **A. Latar Belakang**

Analisis sentimen atau opinion mining merupakan proses memahami, mengekstrak dan mengolah data textual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam suatu kalimat opini. Besarnya pengaruh dan manfaat dari analisis sentimen menyebabkan penelitian dan aplikasi berbasis analisis sentimen berkembang pesat. Bahkan di Amerika terdapat sekitar 20-30 perusahaan yang memfokuskan pada layanan analisis sentimen (Buntoro, 2017). Analisis sentimen digunakan untuk menemukan informasi berharga yang dibutuhkan dari data yang tidak terstruktur, sehingga diharapkan pada penelitian ini dapat diketahui sentimen terhadap obyek wisata makassar.

Obyek wisata merupakan manifestasi dari kreativitas manusia, gaya hidup masyarakat, keberagaman seni budaya, dan warisan sejarah suatu bangsa. Sebagai destinasi yang menarik, obyek wisata dapat berupa tempat-tempat buatan manusia yang memukau, menggambarkan kehidupan sehari-hari, atau merangkum keindahan seni dan sejarah. Dengan merangkum daya tarik dan atraksi wisata, setiap perjalanan menjadi lebih dari sekadar kunjungan fisik, melibatkan eksplorasi mendalam terhadap keunikan dan pesona suatu tempat yang membuatnya begitu menarik bagi para pengunjung. (Asriandy, 2016)

Dalam membangun obyek wisata tersebut harus memperhatikan keadaan sosial ekonomi masyarakat setempat, sosial budaya daerah setempat, nilai-nilai agama, adat istiadat, lingkungan hidup, dan obyek wisata itu sendiri. Pembangunan obyek dan daya tarik wisata dapat dilakukan oleh Pemerintah, Badan Usaha maupun Perseorangan dengan melibatkan dan bekerjasama pihak-pihak yang terkait. (Asriandy, 2016)

Dalam UU No. 10 Tahun 2009 Tentang Kepariwisataan disebutkan bahwa daya tarik wisata adalah segala sesuatu yang memiliki keunikan, keindahan, dan nilai yang berupa keanekaragaman kekayaan alam, budaya, dan

hasil buatan manusia yang menjadi sasaran atau tujuan kunjungan wisatawan. Kata wisatawan (*tourist*) merujuk kepada orang. Secara umum wisatawan menjadi subset atau bagian dari traveler atau visitor. (Asriandy, 2016)

*Convolutional Neural Network (CNN)* merupakan metode yang sukses dalam pengenalan pola, dikembangkan oleh Lecun pada tahun 1988. CNN memiliki arsitektur dengan puluhan hingga ratusan layer. Setiap layer melakukan pembelajaran dengan menggunakan output dari layer sebelumnya sebagai input. CNN menggunakan tahap pelatihan filter untuk mengekstrak fitur visual pada gambar. Feature maps kemudian dikurangi ukurannya melalui operasi *pooling*. Proses ini berulang hingga fitur terdalam diekstraksi. Struktur umum dari CNN melibatkan *convolutional layer*, *pooling layer*, dan *fully connected network*. (Agung, 2023)

Metode *Deep Learning* merupakan subset dari *Machine Learning* yang bekerja dengan mengekstraksi data menggunakan *Neural Network*. Dari sini model kemudian akan belajar dari error. *Neural Network* itu sendiri terdiri dari sekumpulan lapisan yang dipetakan oleh fungsi aktivasi. Model *Deep Learning* yang biasa digunakan untuk menganalisis data berupa teks adalah CNN, RNN, LSTM, dan *Gated Recurrent Unit*. Melihat dari data yang akan digunakan nantinya yaitu berupa kumpulan ulasan, metode *deep learning* dinilai cocok untuk mengatasinya. (QORITA, 2022)

Berdasarkan penelitian yang akan diangkat terhadap masalah di atas, maka dibuat Analisis Sentimen Text menggunakan *Deep Learning*. Analisis sentimen ini menerapkan metode CNN untuk mengetahui bagaimana cara mengimplementasikan tingkat akurasi dalam ulasan text pada obyek wisata makassar. Dengan penelitian ini, dapat menggunakan dataset yang terdiri dari beberapa tempat wisata Makassar yang beragam ulasan pengunjung yang telah mengunjungi tempat wisata di Makassar. Data set ini akan dianalisis menggunakan pendekatan deep learning dengan menggunakan CNN.

## B. Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan di atas, maka rumusan masalah yang diangkat pada penelitian ini yaitu, Bagaimana tingkat akurasi

Metode CNN (*Convolutional Neural Network*) dalam memprediksi sentimen positif, negatif, dan netral pada ulasan mengenai Tempat Wisata Makassar?

### C. Tujuan Penelitian

Berdasarkan Rumusan Masalah di atas dapat disimpulkan bahwa tujuan penelitian ini adalah untuk mengevaluasi dan menentukan sejauh mana metode CNN (*Convolutional Neural Network*) dapat menghasilkan prediksi sentimen yang akurat terhadap ulasan mengenai Tempat Wisata Makassar.

### D. Manfaat Penelitian

Penelitian tentang penggunaan Metode CNN (*Convolutional Neural Network*) untuk analisis sentimen text tempat wisata Makassar diharapkan baik secara teoritis maupun secara praktis :

1. Secara Teoritis
  - a. Untuk pengembangan ilmu pengetahuan, terutama bagi dunia Teknik Informatika.
  - b. Memberikan contoh metodologi penelitian yang efektif dalam menggabungkan analisis sentimen dengan teknologi CNN.
  - c. Memberikan peningkatan terhadap pemahaman tentang sentimen masyarakat terhadap tempat wisata di Makassar.
2. Secara Praktis
  - a. Bagi Peneliti
    - 1) Memahami proses kerja Metode CNN (*Convolutional Neural Network*) dan bentuk implementasinya.
    - 2) Sebagai portofolio untuk peneliti yang berguna untuk masa yang akan datang.
  - b. Bagi Universitas
    - 1) Sebagai bahan referensi untuk penelitian yang akan datang.
    - 2) Sebagai bahan evaluasi bagi universitas dalam mengembangkan keilmuan, dalam hal ini yang berkaitan dengan program berbasis Metode CNN (*Convolutional Neural Network*) analisis sentimen text.
  - c. Bagi Tempat Wisata

Dapat meningkatkan daya saingnya dalam industri pariwisata, menarik lebih banyak wisatawan dan meningkatkan pendapatan daerah serta ekonomi local.

#### **E. Ruang Lingkup Penelitian**

Dari analisis rumusan masalah di atas dapat dirumuskan beberapa batasan masalah yaitu :

1. Terbatas pada beberapa tempat wisata di kota Makassar dan Gowa. Hasil penelitian mungkin tidak dapat secara langsung diterapkan pada tempat wisata di lokasi lain, kecuali Wisata Kebun Gowa, Pantai Bosowa, Akkarena, Tanjung Bayang dan Bugis Waterpark.
2. Terbatas pada penerapan metode *Convolutional Neural Network (CNN)* saja, tanpa mempertimbangkan metode analisis sentimen lainnya.
3. Terbatas pada analisis teks dan mungkin tidak memasukkan aspek non-teks seperti gambar atau video yang juga dapat mencerminkan sentimen terhadap tempat wisata.

#### **F. Sistematika Penulisan**

Untuk memberikan gambaran umum dari seluruh penulisan ini, Adapun sistematika penulisan yaitu :

##### **BAB I PENDAHULUAN**

Bab ini menerangkan secara singkat dan jelas mengenai latar belakang penulisan penelitian tugas akhir, rumusan masalah, tujuan dan manfaat, Batasan permasalahan, metodologi yang digunakan dan sistematika penulisan.

##### **BAB II TINJAUAN PUSTAKA**

Pada bab ini membahas tentang teori – teori yang melandasi penulis dalam melaksanakan skripsi.

##### **BAB III METODE PENELITIAN**

Membahas tentang metode penelitian dan alat yang digunakan untuk pembuatan sistem.

##### **BAB IV ANALISA DAN PENGUJIAN**

Bab ini berisikan hasil desain sistem serta pembahasan terhadap desain tersebut

## BAB V KESIMPULAN DAN SARAN

Bab ini atau bab terakhir akan memuat kesimpulan isi dari keseluruhan uraian bab – bab sebelumnya dan saran – saran dari hasil yang telah diperoleh serta yang diharapkan dapat bermanfaat dalam pengembangan selanjutnya.



## **BAB II**

### **TINJAUAN PUSTAKA**

#### **A. Landasan Teori**

##### **1. Analisis Sentimen**

Sentimen adalah pendapat atau pandangan yang didasarkan pada perasaan yang berlebih-lebihan terhadap sesuatu (bertentangan dengan pertimbangan pikiran). Sentimen terdapat pada pernyataan atau kalimat yang memiliki pendapat. Sentimen digunakan untuk mengetahui perasaan yang diberikan terhadap topik atau objek. (Gunawan et al., 2018)

Analisis sentimen adalah proses menentukan sentimen dan mengelompokkan polaritas teks dalam dokumen atau kalimat sehingga kategori dapat ditentukan sebagai sentimen positif, negatif, atau netral. Saat ini, peneliti secara luas menggunakan analisis sentimen sebagai salah satu cabang penelitian dalam Ilmu Komputer. Jejaring Sosial, seperti Twitter, umumnya digunakan dalam analisis sentimen untuk menentukan persepsi publik. (Prabowo & Wiguna, 2021)

Analisis sentimen juga dapat disamakan dengan opinion mining, karena berfokus pada pendapat yang menyatakan positif atau negatif. Dalam analisis sentimen, penambangan data dilakukan untuk menganalisis, memproses, dan mengekstrak data textual dalam suatu entitas, seperti layanan, produk, individu, fenomena atau topik tertentu. Proses analisis dapat mencakup teks ulasan, forum, tweet, atau blog, dengan data preprocessing mencakup proses tokenization, stopword, penghapusan, stemming, identifikasi sentimen, dan klasifikasi sentimen. (Prabowo & Wiguna, 2021)

Pada zaman modern ini, sentimen atau opini masyarakat semakin bertambah luas dan bebas diungkapkan di berbagai media. Sentimen dapat menjadi potensi besar bagi perusahaan yang ingin mengetahui umpan balik (feedback) dari masyarakat terhadap merk dagang mereka. Jumlah

pengguna (user) aktif dalam komunikasi online memiliki jumlah data yang sangat banyak sehingga memunculkan data besar (big data). Munculnya teknologi big data yang merupakan himpunan data dalam jumlah yang sangat besar, rumit, dan tidak terstruktur menjadi salah satu sumber daya besar yang dapat diolah untuk memperoleh hasil sentimen yang lebih akurat. Jumlah review dan opini sangat banyak sehingga akan menyulitkan dan memakan waktu untuk membaca secara keseluruhan. Oleh karena itu dapat dirancang sistem yang secara otomatis akan mengelompokkan opini dan review yang ada sesuai kelasnya. Kelas sentimen dibagi menjadi kelas sangat positif, positif, netral, negatif, dan sangat negatif sehingga pengguna dapat membaca memilih ingin membaca opini sesuai yang diinginkan. Dengan adanya sistem analisis sentimen ini diharapkan dapat membantu perusahaan mengetahui umpan balik terhadap merk dagangnya dan masyarakat dalam menilai sebuah produk berdasarkan opini dan review yang ada. (Gunawan et al., 2018)

## 2. Obyek Wisata

Obyek wisata adalah perwujudan daripada ciptaan manusia, tata hidup, seni budaya serta sejarah bangsa dan tempat atau keadaan alam yang mempunyai daya Tarik untuk dikunjungi wisatawan. Sedangkan obyek wisata alam adalah obyek wisata yang daya tariknya bersumber pada keindahan sumber daya alam dan tata lingkungannya. (Asriandy, 2016)

Wisata adalah kegiatan perjalanan atau sebagian dari kegiatan tersebut yang dilakukan secara sukarela serta bersifat sementara untuk menikmati obyek dan daya tarik wisata. Seorang wisatawan berkunjung ke suatu tempat/daerah/Negara karena tertarik oleh sesuatu yang menarik dan menyebabkan wisatawan berkunjung ke suatu tempat/daerah/Negara disebut daya tarik dan atraksi wisata. (Asriandy, 2016)

## 3. *Deep Learning*

*Deep Learning* merupakan salah satu cabang dari *Machine Learning*. Model *deep learning* dapat mempelajari komputasinya sendiri dengan menggunakan otaknya sendiri. *Deep Learning* dirancang untuk

terus menganalisa data seperti pada otak manusia dalam mengambil keputusan. Agar kemampuan *Deep Learning* semakin mempunyai maka *Deep Learning* menggunakan Algoritma *Artifical Neural Network* (ANN), yang terinspirasi dari jaringan biologis otak manusia. (Peryanto et al., 2020)

*Deep Learning* (DL) yang merupakan sebuah teknik berbasis jaringan saraf tiruan telah banyak digunakan dalam beberapa tahun terakhir sebagai salah satu metode implementasi *Machine Learning* (ML). Publikasi penelitian implementasi DL dalam konteks *Educational Data Mining* (EDM) pertama kali muncul pada tahun 2015 dan tiap tahun terus meningkat secara signifikan. (Muhammad Haris Diponegoro et al., 2021)

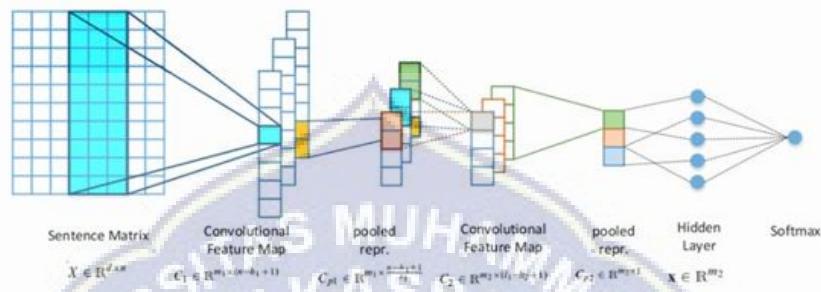
*Deep Learning* yang mulai populer digunakan sejak tahun 2006, menggunakan mekanisme *deep architecture of learning* atau pendekatan *hierarchical learning*. *Learning* atau pembelajaran dalam hal ini adalah sebuah prosedur yang berisi proses estimasi parameter-parameter suatu model sehingga model yang dikembangkan (*algoritme*) dapat menyelesaikan suatu tugas atau permasalahan tertentu. (Muhammad Haris Diponegoro et al., 2021)

#### 4. *Convolutional Neural Network* (CNN)

CNN awal kali dikembangkan oleh Kuniko Fukushima, seorang peneliti dari NHK Broadcasting Science Research Laboratories, Kinuta, Setagaya, Tokyo, Jepang. Awalnya CNN bernama NeoCognitron. Lalu konsep CNN kemudian dimatangkan oleh Yann LeChun, seorang peneliti dari AT&T Bell Laboratories di Holmdel, New Jersey, Amerika Serikat. Kemudian model CNN yang bernama LeNet berhasil diterapkan oleh LeChun pada penelitiannya mengenai pengenalan angka dan tulisan tangan. Alex Krizhevsky di tahun 2012, melakukan penerapan CNN miliknya hingga berhasil menyabet juara pada kompetisi ImageNet Large Scale Visual Recognition Challenge 2012. (Tilasefana & Putra, 2023)

*Convolutional neural network* (CNN), adalah sebuah kelas dari jaringan syaraf tiruan yang telah dominan digunakan pada tugas visi komputer. CNN adalah sebuah konstruk matematika yang biasanya disusun

oleh 3 tipe layer, yaitu *convolution*, *pooling*, dan *layer* yang sepenuhnya terhubung. Dua *layer* pertama, *convolution* dan *pooling* melakukan ekstraksi fitur, sedangkan *layer* ketiga, menempatkan fitur yang terekstraksi menjadi hasil akhir keluaran, seperti klasifikasi. (Hidayatullah & Nayoan, 2019)



Gambar 1. Arsitektur CNN (Hermanto et al., 2021)

##### 5. Supervised Learning

*Supervised learning* melibatkan penggunaan data yang telah diannotasi, di mana setiap sampel data dikaitkan dengan label atau hasil yang diketahui sebelumnya. Tujuan dari *supervised learning* adalah untuk melatih model komputer yang dapat mempelajari pola-pola dalam data dan melakukan prediksi akurat terhadap data yang belum diketahui. Dengan menggunakan algoritma *supervised learning*, model dapat digunakan untuk mengklasifikasikan data ke dalam kategori yang tepat atau untuk melakukan prediksi numerik. (Mestika et al., 2022)

Pada algoritma *Supervised Learning*, sistem diberikan training data set berupa informasi masukan dan keluaran yang diinginkan, sehingga sistem akan mempelajari berdasarkan data yang telah ada. Sistem akan mencari pola dari data set, kemudian pola itu akan dijadikan sebagai acuan untuk kumpulan data berikutnya. (Abijono et al., 2021)

##### 6. Tensorflow

*Tensorflow* adalah *library* perangkat lunak yang dikembangkan oleh Tim Google Brain Mesin Cerdas Google Asosiasi, yang bertujuan untuk melakukan pembelajaran mesin dan jaringan syaraf dalam penelitiannya.

*Tensorflow* menggabungkan aljabar komputasi dengan teknik optimasi kompilasi, yang memfasilitasi perhitungan banyak ekspresi matematika. Fitur utama yang terdapat dalam *tensorflow* adalah : (Economics et al., 2020)

- a. Mendefinisikan, mengoptimalkan, dan menghitung secara matematis ekspresi wajah yang melibatkan array multidimension (tensors).
- b. Pemrograman pendukung jaringan syaraf dalam dan teknik *machine learning*.
- c. Pemakaian GPU (*Graphics Processing Unit*) yang efisien, mengotomasi manajemen dan optimalisasi memori yang sama terhadap data yang digunakan. *Tensorflow* mampu menulis kode yang sama dan menjalankannya di CPU atau GPU. Lebih khususnya lagi *tensorflow* dapat mengetahui bagian mana yang harus dipindahkan ke GPU.
- d. Skalabilitas komputasi yang tinggi pada keseluruhan mesin terhadap kumpulan data yang besar.

## 7. *Flowchart*

*Flowchart* adalah bagan yang menunjukkan alur atau alur dalam suatu program atau prosedur sistem secara logis. *Flowchart* (bagan alir) adalah sebuah ilustrasi berupa diagram alir dari algoritma-algoritma dalam suatu program, yang menyatakan arah aliran dari program tersebut. (Muhamad, 2018)

*Flowchart* dapat digunakan untuk menyajikan kegiatan manual, kegiatan pemrosesan ataupun keduanya. *Flowchart* merupakan rangkaian simbol-simbol yang digunakan untuk mengkonstruksi. (Budiman et al., 2021)

*Flowchart* dapat diartikan sebagai suatu alat atau sarana yang menunjukkan langkah-langkah yang harus dilaksanakan dalam menyelesaikan suatu permasalahan untuk komputasi dengan cara mengekspresikannya ke dalam serangkaian simbol-simbol grafis khusus. Manfaat yang akan diperoleh bila menggunakan flowchart dalam pemecahan masalah komputasi : (Nuraini, 2015)

- a. Terbiasa berfikir secara sistematis dan terstruktur
- b. Mudah mengecek dan menemukan bagian-bagian
- c. Prosedur yang tidak valid dan bertele-tele
- d. Prosedur akan mudah dikembangkan

Flowchart atau sering disebut dengan diagram alir merupakan suatu jenis diagram yang merepresentasikan algoritma atau langkah-langkah instruksi yang berurutan dalam system. Seorang analis sistem menggunakan flowchart sebagai bukti dokumentasi untuk menjelaskan gambaran logis sebuah sistem yang akan dibangun kepada programmer. Dengan begitu, flowchart dapat membantu untuk memberikan solusi terhadap masalah yang bisa saja terjadi dalam membangun sistem. Pada dasarnya, flowchart digambarkan dengan menggunakan simbol-simbol. Setiap simbol mewakili suatu proses tertentu. Sedangkan untuk menghubungkan satu proses ke proses selanjutnya digambarkan dengan menggunakan garis penghubung. Dengan adanya flowchart, setiap urutan proses dapat digambarkan menjadi lebih jelas. Selain itu, ketika ada penambahan proses baru dapat dilakukan dengan mudah menggunakan flowchart ini. Setelah proses membuat flowchart selesai, maka giliran programmer yang akan menerjemahkan desain logis tersebut kedalam bentuk program dengan berbagai bahasa pemrograman yang telah disepakati. (Rosaly & Prasetyo, 2020)

Flowchart biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut. Proses di lingkungan organisasi pada umumnya merupakan suatu rangkaian kegiatan yang berulang. Setiap siklus kegiatan tersebut biasanya dapat dipecahkan ke dalam beberapa langkah kecil. Dari uraian langkah – langkah tersebut, kita dapat mencari langkah mana saja yang bisa kita perbaiki (improve). Jadi, flowchart adalah diagram yang menyatakan aliran proses dengan menggunakan anotasi bidang – bidang geometri, seperti lingkaran, persegi empat, wajik, oval, dan sebagainya untuk mempresentasikan langkah – langkah kegiatan beserta urutannya dengan menghubungkan langkah tersebut menggunakan tanda panah. (Ridlo, 2017)

### 8. *Scikit – Learn*

*Scikit – Learn* adalah modul Python yang menyediakan berbagai macam algoritma *machine learning*. *Scikit-learn* menggunakan *task-oriented interface* yang konsisten sehingga dapat dengan mudah dilakukan perbandingan antara metode. *Scikit-learn* tersedia dalam bentuk *library python*. (Santoso & Bunyamin, 2019)

*Scikit – Learn* adalah modul python yang mengintegrasikan berbagai algoritma pembelajaran mesin *state-of-the-art* untuk masalah yang diawasi dan tidak diawasi skala menengah. Paket ini berfokus pada membawa pembelajaran mesin ke non-spesialis menggunakan bahasa tingkat tinggi tujuan umum. Penekanan diberikan pada kemudahan penggunaan, kinerja, dokumentasi, dan konsistensi API. Ini memiliki ketergantungan minimal dan didistribusikan dibawah lisensi BSD yang disederhanakan, mendorong penggunaannya baik dalam aturan akademis dan komersial. (Riadi Silitonga et al., 2019)

## B. Penelitian Terkait

### 1. Yerik Afrianto Singgalen (2023)

Pada penelitian Yerik Afrianto Singgalen dengan judul “Analisis Sentimen Wisatawan terhadap Taman Nasional Bunaken dan Top 10 Hotel Rekomendasi Tripadvisor Menggunakan Algoritma SVM dan DT berbasis CRISP-DM”, Tujuan penelitian ini adalah untuk menganalisis sentimen wisatawan terhadap Taman Nasional Bunaken dan hotel-hotel yang direkomendasikan oleh Tripadvisor. Penelitian ini menggunakan metode CRISP-DM (Cross-Industry Standard Process for Data Mining) dalam menganalisis sentimen wisatawan terhadap Taman Nasional Bunaken dan hotel-hotel yang direkomendasikan oleh Tripadvisor. Teknik pengumpulan data dalam penelitian ini dilakukan dengan mengumpulkan ulasan wisatawan terhadap destinasi Taman Nasional Bunaken dan hotel-hotel yang direkomendasikan oleh Tripadvisor. Data ulasan ini diperoleh dari platform digital yang menyediakan transaksi daring, di mana pengguna memiliki kewenangan untuk memberikan ulasan terhadap destinasi wisata dan jasa akomodasi yang mereka kunjungi. Hasil penelitian ini

menunjukkan bahwa penggunaan metode CRISP-DM dalam analisis sentimen terhadap Taman Nasional Bunaken dan hotel-hotel yang direkomendasikan oleh Tripadvisor memberikan hasil yang signifikan.

2. Wawah Khofifah, Dhian Nur Rahayu dan Arif Maulana Yusuf (2022)

Pada penelitian Wawah Khofifah, Dhian Nur Rahayu dan Arif Maulana Yusuf dengan judul “Analisis Sentimen Menggunakan Naïve Bayes untuk Melihat Review Masyarakat Terhadap Tempat Wisata Pantai Di Kabupaten Karawang Pada Ulasan Google Maps”, Tujuan penelitian ini adalah untuk mengetahui apakah ulasan tersebut cenderung positif atau negatif, serta memberikan masukan bagi pemerintah dalam pengelolaan tempat wisata pantai di daerah tersebut. Metode yang digunakan dalam penelitian ini meliputi pendekatan kualitatif dengan jenis studi kasus, serta penerapan algoritma Naive Bayes untuk analisis sentimen. Teknik pengumpulan data dilakukan melalui studi dokumentasi dengan melihat dan mengamati ulasan masyarakat pada Google Maps, serta studi literatur dari berbagai jurnal, buku, dan artikel. Hasil penelitian menunjukkan bahwa penggunaan metode Naive Bayes dalam analisis sentimen terhadap ulasan masyarakat mengenai tempat wisata pantai di Kabupaten Karawang memberikan hasil yang bervariasi. Beberapa pantai mendapat ulasan positif, sementara beberapa pantai lain mendapat ulasan negatif.

3. Ahmad Rifa’I, Herry Sujaini, Dian Prawira (2021)

Pada penelitian Ahmad Rifa’I, Herry Sujaini dan Dian Prawira dengan judul “Sentiment Analysis Objek Wisata Kalimantan Pada Google Maps Menggunakan Metode Naïve Bayes”, Tujuan penelitian ini adalah untuk membangun sebuah sistem yang mampu menganalisis sentimen berdasarkan ulasan mengenai objek wisata di Kalimantan Barat menggunakan metode Naive Bayes. Penelitian ini menggunakan metode Naive Bayes untuk melakukan analisis sentimen terhadap ulasan mengenai objek wisata di Kalimantan Barat. Penelitian ini juga menggunakan teknik text mining, sentiment analysis, dan machine learning sebagai pendukung metode Naive Bayes. Penelitian ini menggunakan teknik pengumpulan data

dengan melakukan crawling data dari Google Maps menggunakan Google Maps API. Setelah pengumpulan data, dilakukan proses text pre-processing yang meliputi case folding, tokenizing, filtering, dan stemming untuk menjadikan teks terstruktur. Hasilnya menunjukkan bahwa sistem mampu mengklasifikasikan ulasan dengan baik meskipun masih ada limitasi dalam akurasi klasifikasi. Secara keseluruhan, objek wisata Kalimantan Barat mendapat sentimen positif. Antarmuka sistem juga mendapat nilai baik dalam pengujian fungsionalitas. Studi ini memberikan wawasan dalam meningkatkan kualitas pariwisata di daerah tersebut. Referensi yang digunakan mencakup berbagai penelitian terkait analisis sentimen dan metode Naive Bayes. (Rifa'i et al., 2021)

#### 4. Nurdin Mohamad, Rakhmat Jaya Lahay (2021)

Pada penelitian Nurdin Mohamad dan Rakhmat Jaya Lahay dengan judul “Analisis Nilai Kelestarian Lingkungan Obyek Wisata Tasik Ria Berdasarkan Willingness To Pay”. Penelitian ini bertujuan untuk mengetahui nilai sosial ekonomi kawasan pariwisata Tasik Ria serta tingkat kepedulian masyarakat terhadap pelestarian lingkungan berdasarkan jenis pekerjaan. Metode yang digunakan dalam penelitian ini adalah metode survei dan pengumpulan data melalui kuisioner, observasi lapangan, dan wawancara. Berdasarkan hasil penelitian, sebagian besar petani dan keterampilan jasa pariwisata memiliki tingkat kepedulian paling besar terhadap pelestarian lingkungan objek wisata Tasik Ria. Nilai sosial ekonomi kawasan pariwisata Tasik Ria berdasarkan Willingness to Pay diperoleh sebesar Rp743.279.850.000. Hal ini menunjukkan bahwa masyarakat memiliki kerelaan untuk membayar kelestarian lingkungan objek wisata Tasik Ria. (Mohamad & Lahay, 2021)

#### 5. Muhammad Iqbal Wahyu Pradana, Gerry Katon Mahendra (2021)

Pada penelitian Muhammad Iqbal Wahyu Pradana dan Gerry Katon Mahendra dengan judul “Analisis Dampak Covid-19 Terhadap Sektor Pariwisata Di Objek Wisata Goa Pindul Kabupaten Gunungkidul”, Tujuan penelitian ini adalah untuk menganalisis dampak pandemi Covid-19

terhadap sektor pariwisata, khususnya obyek wisata Goa Pindul di DIY. Penelitian ini menggunakan metode kualitatif deskriptif untuk mengevaluasi penurunan jumlah pengunjung, pendapatan pedagang, dan kontribusi terhadap pendapatan asli daerah (PAD) setelah adanya pandemi. Teknik pengumpulan data dalam penelitian ini meliputi wawancara dan dokumentasi lapangan. Hasil penelitian menunjukkan bahwa pandemi Covid-19 telah memiliki dampak signifikan terhadap sektor pariwisata di obyek wisata Goa Pindul di Gunungkidul, DIY. Terdapat penurunan jumlah pengunjung, pendapatan pedagang, dan kontribusi terhadap pendapatan asli daerah (PAD) setelah adanya pandemi. Faktor-faktor yang mempengaruhi termasuk PSBB dan penutupan obyek wisata. BUMDes Maju Mandiri juga mengalami penurunan penghasilan akibat penutupan operasional. Dampak ekonomi pariwisata akibat Covid-19 sangat signifikan. (Name et al., 2021)

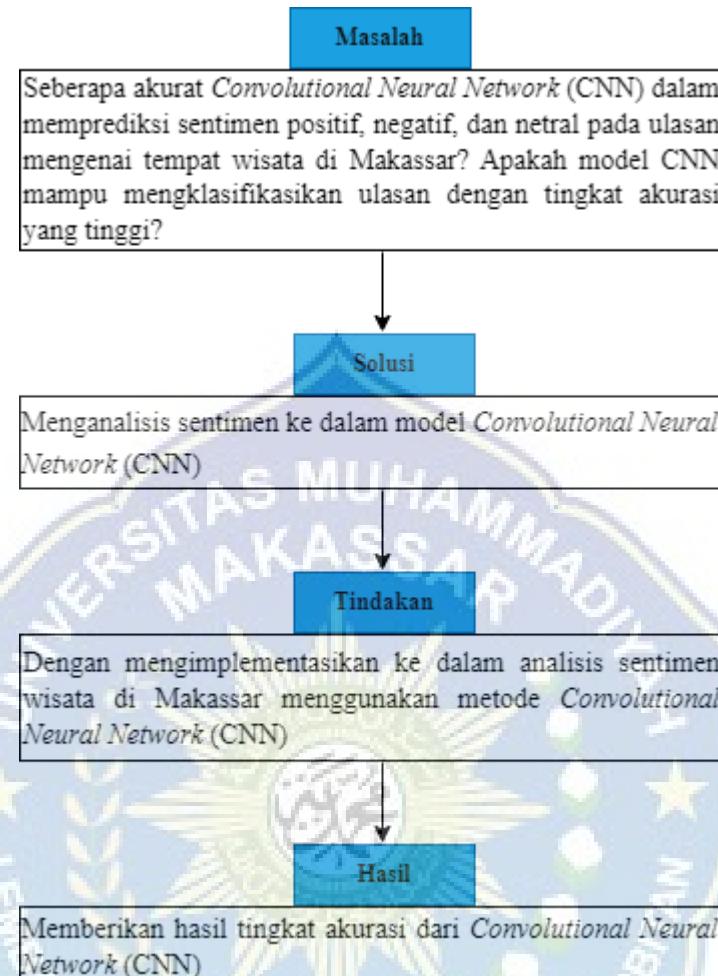
#### 6. Muh. Agung Achmad Gani (2020)

Pada penelitian yang dilakukan oleh Muh. Agung Achmad Gani dengan judul “Analisis Kepuasan Wisatawan Terhadap Objek Wisata Bahari di Kota Makassar”, Tujuan dari penelitian ini yaitu untuk memahami faktor-faktor yang mempengaruhi kepuasan wisatawan pada obyek wisata bahari di Kota Makassar. Penelitian ini menggunakan metode survei dengan pendekatan deskriptif kuantitatif. Teknik pengumpulan data yang digunakan merupakan observasi dan dokumentasi. Observasi dilakukan untuk mengamati langsung perilaku dan interaksi wisatawan dengan obyek wisata bahari di Kota Makassar. Sementara itu, dokumentasi dilakukan untuk mengumpulkan data sekunder yang bersumber dari literatur, laporan, dan dokumen terkait pariwisata di Kota Makassar. Hasil penelitian ini menunjukkan bahwa kualitas pelayanan, kualitas produk, dan daya tarik wisata memiliki pengaruh positif dan signifikan terhadap kepuasan wisatawan di Kota Makassar. Selain itu, hasil penelitian ini juga menunjukkan bahwa ketiga variabel tersebut secara simultan dapat menjelaskan sebanyak 85,7% dari variabilitas kepuasan wisatawan. (Acmad Gani, 2020)

## 7. Sitti Aliyah Azzahra, Arief Wibowo (2020)

Pada penelitian Sitti Aliyah Azzahra dan Arief Wibowo dengan judul “Analisis Sentimen Multi – Aspek Berbasis Konversi Ikon Emosi Dengan Algoritme Naïve Bayes Untuk Ulasan Wisata Kuliner Pada Web Tripadvisor”, Tujuan penelitian ini adalah untuk melakukan analisis sentimen ulti-aspek terhadap ulasan wisata kuliner di situs web TripAdvisor menggunakan algoritme Naïve Bayes, untuk meningkatkan nilai akurasi dari penelitian sebelumnya, dengan fokus pada konversi ikon emosi, pembobotan, dan klasifikasi menggunakan Naïve Bayes Classifier. Metode penelitian yang digunakan dalam penelitian ini adalah metode eksperimen dengan pendekatan kuantitatif. Teknik pengumpulan data dalam penelitian ini menggunakan metode web scraping dan metode Non Random Sampling. Hasil penelitian ini menunjukkan bahwa analisis sentimen multi-aspek berbasis konversi ikon emosi dengan algoritme Naïve Bayes untuk ulasan wisata kuliner di situs web TripAdvisor mencapai akurasi sebesar 98,67%. Hal ini menunjukkan peningkatan nilai akurasi dari penelitian sebelumnya yang sebesar 88,78%. Teknik pengujian yang digunakan dalam penelitian ini meliputi eksperimen dengan framework RapidMiner 5.3 dan evaluasi menggunakan Confusion Matrix dan K-Fold Cross Validation. Dengan demikian, hasil penelitian ini menunjukkan bahwa metode analisis sentimen multi-aspek berbasis konversi ikon emosi dengan algoritme Naïve Bayes mampu memberikan hasil yang sangat akurat dalam mengklasifikasikan ulasan wisata kuliner di situs web TripAdvisor. (Azzahra & Wibowo, 2020)

### C. Kerangka Pikir



Gambar 2. Kerangka Pikir

## **BAB III**

### **METODE PENELITIAN**

#### **A. Tempat dan Waktu Penelitian**

Penelitian ini dilakukan secara online dengan mengambil dataset dari ulasan wisata yang ada di google maps. Pada pelaksanaan penelitian ini dilakukan pada bulan Januari 2024 sampai semua proses pengumpulan data selesai.

#### **B. Alat dan Bahan**

Adapun alat dan bahan yang akan digunakan dalam penelitian ini adalah sebagai berikut :

1. Kebutuhan Hardware (perangkat keras)
  - a. Laptop Acer Aspire 3 Ryzen 3
  - b. Ram 8 GB
  - c. Sistem operasi Windows 11
2. Kebutuhan Software (perangkat lunak)
  - a. Spreadsheets / Excel
  - b. Python
  - c. Scikit-learn dan TensorFlow (untuk implementasi model machine learning dan deep learning)
  - d. Data
  - e. Google Colaboratory / Google Colab

#### **C. Perancangan Sistem**

Perancangan sistem sangat penting dalam pembangunan suatu sistem karena menguraikan bagaimana suatu sistem dibangun dari tahap perencanaan hingga tahap pembuatan fungsi-fungsi yang diperlukan untuk pengoperasian sistem. Tujuan dari perancangan sistem adalah untuk menentukan apakah sistem yang akan dikembangkan akan menghasilkan hasil yang diinginkan.

##### **1. Studi Literatur**

Studi literatur yang dilakukan dalam penelitian ini adalah untuk mengumpulkan informasi dan pengetahuan terkait analisis sentimen teks,

Metode *Convolutional Neural Network* (CNN), serta topik khususnya, yaitu tempat wisata di Makassar.

## 2. Pengumpulan Data

Proses pengumpulan data melibatkan pengambilan ulasan atau teks terkait tempat wisata di Makassar. Data ini bisa diperoleh dari berbagai ulasan pengunjung kepada tempat wisata. Tujuan pengumpulan data ini adalah untuk memiliki dataset yang cukup untuk diberikan dalam analisis sentimen.

## 3. Pengolahan Data

Setelah data terkumpul, langkah berikutnya adalah pengolahan data. Ini melibatkan pra-pemrosesan data untuk memberikan teks dari noise, seperti tandabaca, karakter khusus, atau kata – kata yang tidak relevan, serta melibatkan juga tokenisasi, penghapusan stopwords dan stemming. Proses ini dapat meningkatkan kualitas data dan mempersiapkannya untuk tahap analisis menggunakan Metode CNN.

## 4. Perancangan Sistem

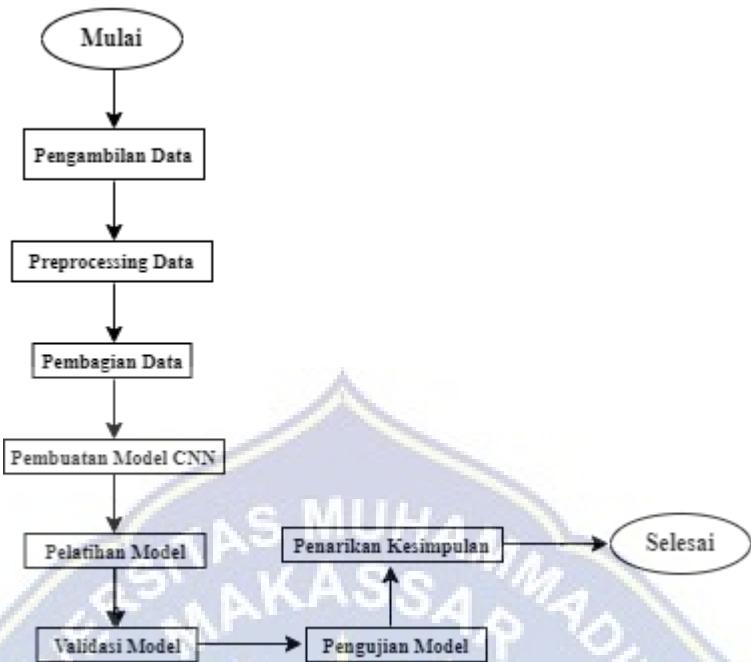
Dalam perancangan sistem dalam penelitian ini melibatkan pemilihan parameter, pengaturan arsitektur CNN yang sesuai, dan persiapan model untuk menganalisis sentimen dari teks ulasan tempat wisata di Makassar.

## 5. Pengujian Sistem

Pengujian melibatkan penggunaan dataset uji yang tidak digunakan selama pelatihan dan evaluasi hasil analisis sentimen. Penelitian ini dapat mengevaluasi tingkat akurasi, presisi, dan recall dari Metode CNN dalam memprediksi sentimen.

## 6. Penarikan Kesimpulan

Setelah pengujian sistem, peneliti dapat menarik kesimpulan terkait dengan efektivitas Metode CNN dalam analisis sentimen teks pada studi kasus tempat wisata di Makassar. Kesimpulan ini mencakup hasil evaluasi kinerja sistem, temuan terkait sentimen pengunjung terhadap tempat wisata, dan implikasi dari penelitian tersebut.



Gambar 3. Diagram alur Proses Penelitian Analisis Sentimen Text Obyek Wisata

Dalam diagram di atas dapat dijelaskan bahwa alur kerja dimulai dengan langkah awal dari proses, menandakan dimulainya proses Pembangunan dan evaluasi model CNN untuk analisis sentimen.

Pengambilan data, dimana data yang diperlukan untuk analisis sentimen dikumpulkan dari ulasan dari tempat wisata di Makassar. Preprocessing data adalah proses untuk membersihkan, mengubah, dan mempersiapkan data yang telah dikumpulkan agar siap untuk digunakan dalam pembuatan model. Ini mencakup langkah – langkah seperti menghilangkan tanda baca, karakter khusus, dibersihkan dari noise atau kata – kata yang tidak relevan. Pembagian data dimana data dibagi menjadi dua bagian, yaitu data pelatihan yang digunakan untuk melatih model dan data pengujian yang digunakan untuk menguji kinerja model.

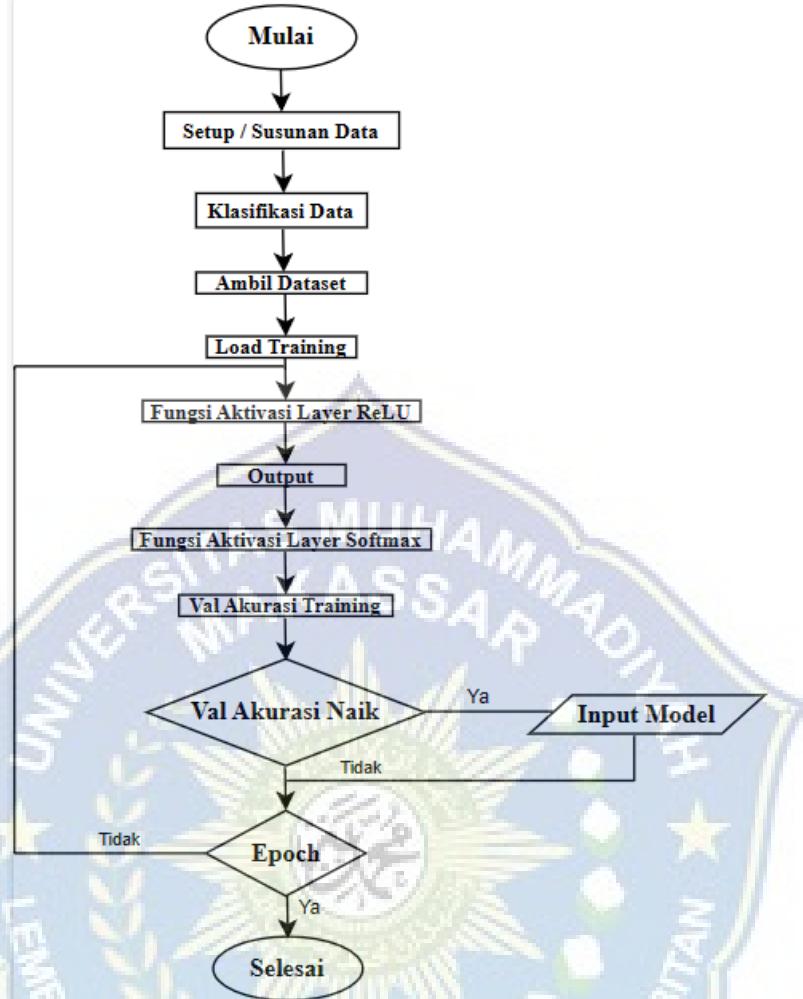
Pembuatan model CNN proses merancang dan membangun model Convolutional Neural Network (CNN) yang akan digunakan untuk melakukan analisis sentimen. Pelatihan model adalah proses mengajarkan model CNN untuk mengenali pola dan fitur dalam data latih yang telah dipersiapkan sebelumnya. Pelatihan melibatkan proses pembelajaran mesin Dimana parameter model diperbarui berulang kali untuk meningkatkan kemampuan model dalam membuat

prediksi yang akurat. Setelah pelatihan selesai, model dievaluasi menggunakan data validasi yang tidak terlibat dalam pelatihan untuk memastikan bahwa model tidak overfitting. Model yang telah dilatih dan divalidasi, kemudian diuji menggunakan data pengujian yang independent untuk mengukur kinerjanya dalam memprediksi sentimen dari teks.

Hasil dari pengujian model dievaluasi dan dianalisis untuk memahami seberapa baik model dapat memprediksi sentimen dari teks yang diberikan. Berdasarkan hasil analisis, sentimen dari teks yang dianalisis diinterpretasikan, misalnya, apakah sentimen tersebut positif, negatif atau netral. Informasi dan insight yang didapatkan dari analisis sentimen digunakan untuk menarik kesimpulan tentang data dianalisis.

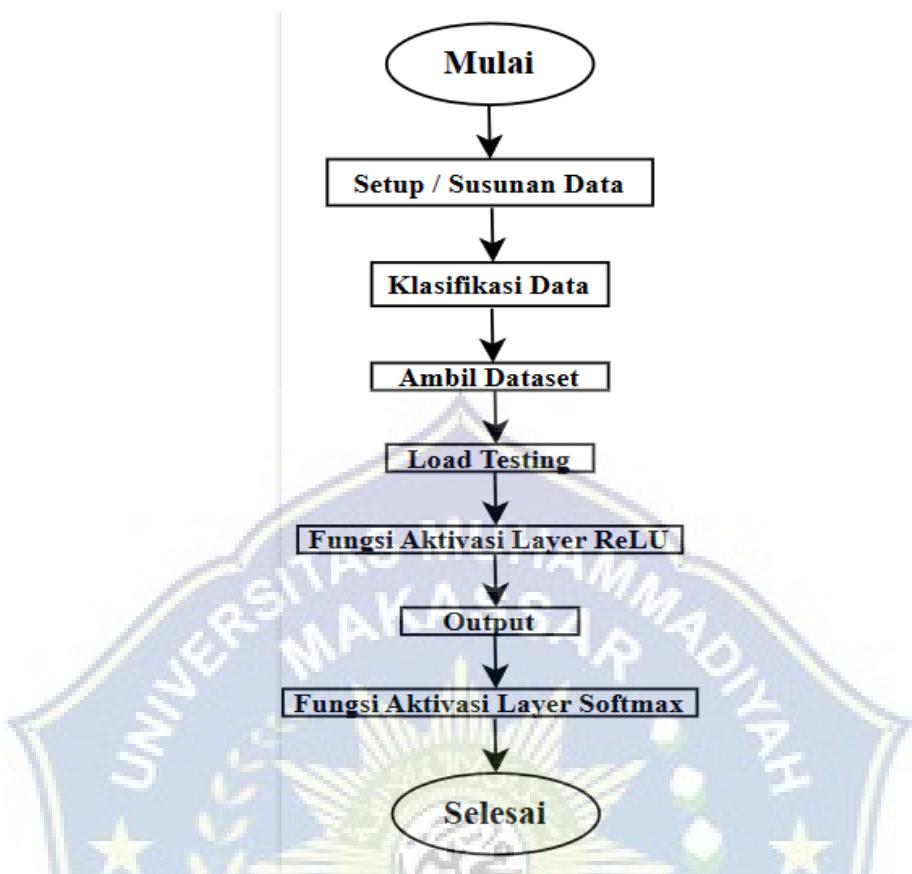
Dalam perancangan sistem atau diagram sistem yang akan dibuat yaitu sebagai berikut :





Gambar 4. Perancangan Sistem Training

Pada diagram di atas dapat dijelaskan bahwa alur kerja dimulai dengan menyiapkan data, mengklasifikasikannya, dan memuatnya untuk pelatihan model. Fungsi aktivasi *ReLU* diterapkan pada beberapa lapisan, diikuti dengan lapisan output yang menggunakan fungsi aktivasi *Softmax* untuk menghasilkan probabilitas kelas. Selama pelatihan, akurasi dari data training dievaluasi secara berkala, dan jika akurasi terus meningkat, proses lanjut ke langkah-langkah berikutnya. Model dimasukkan ke dalam sistem untuk proses pelatihan yang melibatkan iterasi (*epoch*) untuk memperbaiki kinerja model. Setelah pelatihan selesai, proses berakhir.



Gambar 5. Perancangan Sistem Testing

Sistem yang dirancang terdiri dari dua alur utama, yakni perancangan sistem training dan perancangan sistem testing. Dalam perancangan sistem training, fokusnya adalah melihat kurva akurasi. Hal ini dilakukan untuk memonitor dan mengevaluasi kinerja model atau sistem selama proses pelatihan. Sementara itu, perancangan sistem testing difokuskan pada tahap pengujian. Tujuan utama sistem testing adalah untuk melakukan uji coba terhadap kemampuan sistem dalam mengidentifikasi dan menguji yang berkaitan dengan yang ditargetkan.

#### D. Teknik Pengujian Sistem

Teknik pengujian sistem yang akan digunakan pada pengujian ini adalah menggunakan skenario dengan memisahkan data menjadi dua bagian, yaitu data training dan testing. Hal ini dilakukan untuk memastikan bahwa model yang dikembangkan dapat secara efektif mempelajari pola-pola dari data training dan kemudian dapat menggeneralisasikan pengetahuannya dengan baik pada data yang belum pernah dilihat sebelumnya, yaitu data testing. Dengan demikian,

teknik pengujian ini memungkinkan evaluasi yang akurat terhadap kinerja model dalam mengklasifikasikan sentimen dari teks terkait tempat wisata di Makassar.

Teknik ini bertujuan untuk menguji keakuratan dan efektivitas metode *Convolutional Neural Network* (CNN) dalam menganalisis sentimen terhadap teks yang berkaitan dengan tempat wisata di Makassar. Pengujian ini akan melibatkan pengumpulan data teks tentang tempat wisata, pelabelan sentimen (positif, negatif, atau netral), dan pengujian model CNN untuk menentukan seberapa baik model dapat mengklasifikasikan sentimen-sentimen tersebut.

Pengujian Akurasi, teknik ini bertujuan untuk mengukur tingkat keberhasilan model dalam mengklasifikasikan sentimen dengan tepat. Akurasi model dihitung dengan membandingkan hasil klasifikasi sentimen dari model dengan label sentimen yang sebenarnya pada data testing, menggunakan persamaan tertentu untuk menghitung proporsi prediksi yang benar dari keseluruhan data uji.

## E. Teknik Analisis Data

Analisis data merupakan metode yang digunakan untuk mengetahui bagaimana menggambarkan data, hubungan data, semantik data dan batasan data yang ada pada suatu sistem informasi (Pelham, 2023). Proses analisis data dilakukan pada penelitian ini adalah sebagai berikut :

### 1. Reduksi Data (*Data Reduction*)

Reduksi data merupakan langkah penting dalam penanganan data lapangan yang melibatkan jumlah data yang sangat besar. Proses ini mengharuskan peneliti untuk mencatat setiap detail secara cermat. Dengan mereduksi data, tujuannya adalah memberikan gambaran yang lebih jelas dan mempermudah peneliti dalam pengumpulan dan pencarian data lebih lanjut sesuai kebutuhan penelitian. Fokus utama peneliti kualitatif adalah pada hasil, dan oleh karena itu, mereduksi data menjadi kunci dalam mencapai pemahaman yang mendalam. Namun, peneliti perlu berhati-hati ketika menemui *outliers*, *unknowns*, dan kualifikasi yang tidak berpola.

## 2. Penyajian Data (*Display Data*)

Hasil produksi akan ditampilkan dengan cara eksklusif untuk setiap pola, kategori, fokus, dan tema yang ingin dipahami dan dimengerti oleh peneliti. Penggunaan display data membantu peneliti melihat gambaran keseluruhan atau bagian-bagian tertentu berdasarkan output penelitian. Dalam penelitian kualitatif, penyajian data dapat berupa uraian singkat, bagan, interaksi antar kategori, dan sejenisnya. Tekstual deskripsi sering digunakan untuk menyajikan data dalam konteks penelitian kualitatif.

## 3. Penarikan Kesimpulan (*Concluding Drawing Verification*)

Langkah ketiga dalam analisis data adalah penarikan kesimpulan dan verifikasi. Kesimpulan awal yang diajukan masih bersifat sementara dan dapat berubah jika diperlukan bukti yang mendukung melalui pengumpulan data berikutnya. Oleh karena itu, kesimpulan dalam penelitian mungkin dapat menjawab rumusan masalah yang sudah dirumuskan sejak awal. Hal ini disebabkan karena perumusan masalah pada penelitian kualitatif bersifat sementara dan akan berkembang setelah melakukan penelitian di lapangan.

## BAB IV

### HASIL DAN PEMBAHASAN

#### A. Pengambilan Data

Pengambilan data diakses dari ulasan google maps dalam bentuk file berekstensi.csv. Data didapatkan menggunakan teknik analisis sentimen seperti pemrosesan Bahasa alami (natural language processing) untuk mengekstrak opini, perasaan, dan pola dari ulasan tersebut. Ulasan ini dapat mencakup beragam topik, mulai dari pujiannya terhadap pemandangan, fasilitas, atau pelayanan hingga keluhan tentang keramaian atau kebersihan. Data yang berhasil didapatkan adalah 4.500 data. Data yang berhasil diperoleh terbagi kedalam 1.500 data untuk masing – masing label Positif, Negatif dan Netral.



Gambar 6. *Scraping* Data Ulasan Menggunakan *Instant Data Scraper*

Setelah menyelesaikan proses pengambilan data dengan proses scrapping, hasilnya akan disimpan dalam format excel. Berikut adalah hasil scrapping data ulasan yang telah disimpan dalam bentuk excel :

A1		wil7pd																						
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U		
1	wil7pd																							
2	Wisata kolam renang, kolam arus, dan kebun binatang mini'nya sangat bagus. Untuk kolam arus, sebaiknya lebih sering dibersihkan. ¢€;																							
3	Sudah																							
4	Sudah																							
5	Permainan lumayan bagus. Harga tiket pas masuk di hari kerja 60rb udah bs ke taman satwa dan waterboom. Wahana waterboom jd lumayan bagus dan lengkap. Ada utk dewasa, kolam bayi. Cuman ada bbrpa perosotan yg lumayan tetapi kebun satwanya sangat luar biasa, banyak jenis burung yg ada bersama dengan beberapa reptil dan hewan khas sulawesi seperti anoa. Pokoknya ngajak anak sekolah/anak kecil ke sini sar																							
6	Wahannya lumayan tetapi kebun satwanya sangat luar biasa, banyak jenis burung yg ada bersama dengan beberapa reptil dan hewan khas sulawesi seperti anoa. Pokoknya ngajak anak sekolah/anak kecil ke sini sar																							
7	Wisata yg murah nyaman asri . Disini ada waterboom, taman burung, buaya ,ular, Bango danau buatan . Banyak sekali pohon pohon sehingga suasana teduh dan sejuk. ¢€!																							
8	Saya																							
9	Rekreasi bersama keluarga disini sangat di sarankan, dari segi kebersihan Bersih Kok ada Gazebo juga ya walaupun nyewa tapi gak mahal2 amat apalgi untuk sekeluarga, dari segi harga baik makanan yang di jual d																							
10	Perpadua																							
11	Kurang																							
12	Sangat menyenangkan, kalau membawa anak datanglah saat tidak terlalu ramai, seperti pada saat bukan hari libur. Harga terjangkau, tpi fasilitas liburan yg diberikan bukan murahan, bisa mengenalkan kepada an																							
13	Waterpark yang sangat nyaman , bersih , dan sangat terawat .. fasilitas kolam , permainan perosotan yang banyak akan membuat anak2 senang berlalu kesini .. harga tiket masuk 75rb/org sudah termasuk berenang																							
14	Kolamnya oke, air bersih, gazebo dan kantin ok. Kolam renang buat anak dan dewasa tersedia. Wahana seluncuran waterboomnya juga menarantang, apalagi di sungai alir.																							
15	Tarifnya terjangkau, hewan di taman satwa cukup variatif. Boleh foto bersama burung tanpa biaya tambahan. Di taman satwa cukup banyak gazojo untuk beristirahat.																							
16	Terakhir																							
17	toilet bersih, pegawai cara dan dilakukan untuk foto dengan sangat ramah dan sabar menjelaskan setiap pertanyaan																							
18	Tempat																							
19	Bagus tempatnya...by gazebo...adem...byk pohon...tp koleksi hewannya sedikit so far nyaman buat nyantai keluaragaY' ØYØY ¢€;																							
20	Salah satu destinasi wisata alam di Gowa, letaknya cukup dekat dari Pantai Losari Makassar. Terdapat 3 wahana, yaitu water boom, taman burung dan outdoor activity, tiket masuk bisa terpisah masing masing wahan																							
21	Terlalu banyak aturan gak jelas.. Kolam arusnya kotor.. Pembayaran dengan debit selain mandiri selalu kena cash.. Sewa gazebo juga lumayan mahal. Yang lumayan taman burungnya..																							

Gambar 7. Dataset Ulasan

## B. Pelabelan Data

Tahap pelabelan dilakukan untuk memberikan keterangan postif, negatif dan netral terhadap data ulasan tempat wisata makassar. Dalam melakukan pelabelan, ada beberapa teknik yang dapat digunakan seperti Transformer, TextBlob dan VADER. Namun, pelabelan sentimen juga dapat dilakukan secara manual karena menunjukkan tingkat akurasi yang lebih tinggi dan lebih akurat. Berikut pelabelan data dapat dilihat pada table berikut :

Tabel 1. Tahap Pelabelan Data

Ulasan	Label
Tempatnya sejuk. Lebih cocok untuk rekreasi anak-anak. Tiket weekdays 15.000 / orang. Tempat duduk duduk banyak. Ada pemancingan, wahana bermain anak, sepeda listrik atau ATV, juga kolam renang	Positif
Sangat jelek pantai pasir kasarnya luar biasa pantainya tidak bersih	Negatif
"View pantai yang bagus, namun sayang pantainya agak kotor ketika saya datang."	Netral

Setelah melakukan pelabelan secara manual, terdapat 4500 data yang telah diklasifikasikan dengan jumlah yang seimbang antara label positif, negatif dan netral. Dalam penelitian ini, data dibagi menjadi data latih dan data uji

menggunakan library dari scikit-learn. Proporsi pembagian adalah 10% dari keseluruhan data untuk data uji, dan 90% untuk data latih.

### C. Data Preprocessing

*Preprocessing* adalah tahap penting dalam penelitian ini. Tujuannya untuk memastikan kebersihan, konsistensi dan kesiapan data yang digunakan dalam analisis sentimen *text* dengan metode CNN. Proses ini dimulai dengan pengumpulan data dari sumber yang telah tersedia.

#### 1. Cleaning

Pada tahap *cleaning* akan dilakukan proses pembersihan data dari penghapusan karakter khusus, seperti tanda baca, simbol, atau karakter non-alfanumerik.

Tabel 2. Tahap Cleaning Data

Sebelum	Sesudah
Harga Tiket murah,tempatnya bersih,dan gazebo tidak perlu bayar... 😊😊😊	harga tiket murahtempatnya bersih dan gazebo tidak perlu bayar
Tempatnya bersih, sejuk, nyaman, dan permainan nya murah2, Mantapp 🙌🙌	tempatnya bersih sejuk nyaman dan permainan nya murah mantapp
di hari-hari biasa (bukan hari libur) jumlah pengunjung yg datang masih bisa dibilang rame, cocok buat liburan keluarga 😊👉	di harihari biasa bukan hari libur jumlah pengunjung yg datang masih bisa dibilang rame cocok buat liburan keluarga
Rekomended untuk acara anak di bawah umur, sangat safety 🙌👉👉	rekomended untuk acara anak di bawah umur sangat safety

```
df['ULASAN'] = df['ULASAN'].apply(lambda x: str(x).lower())
if isinstance(x, str) else x)

comment = 'ULASAN'

df[comment] = df[comment].apply(lambda x: re.sub(r'^a-zA-Z0-9\s+', '', str(x)))

df[comment] = df[comment].apply(lambda x: re.sub(r'\d', '', str(x)))

df[comment] = df[comment].apply(lambda x:
re.sub(r'\.{2,}', '.', str(x)))

df.head(4501)
```

Program tersebut merupakan contoh penggunaan *Python* dengan menggunakan library Pandas dan regular expression (re) untuk membersihkan data teks dalam kolom 'ULASAN' dari suatu DataFrame.

Berikut adalah penjelasan langkah-langkah dalam program tersebut :

#### **Penggunaan .apply() dan .lower():**

Pada langkah ini, kolom 'ULASAN' di DataFrame (*df*) diubah menjadi lowercase menggunakan metode *lower()*. Ini dilakukan dengan menggunakan fungsi lambda pada metode *apply()*. Jika nilai dalam kolom 'ULASAN' adalah string (*isinstance(x, str)*), maka string tersebut diubah menjadi *lowercase*. Jika nilai tersebut bukan string, maka nilai tersebut tidak diubah.

#### **Penggunaan .apply() dan re.sub():**

Pada langkah ini, dilakukan pembersihan data dari karakter selain huruf, angka, dan spasi. Fungsi *re.sub()* digunakan untuk mengganti pola tertentu dalam teks dengan string pengganti (dalam hal ini, string kosong ""). Ekspresi reguler *r'^a-zA-Z0-9\sJ'* digunakan untuk mencocokkan karakter selain huruf (a-z dan A-Z), angka (0-9), dan spasi. Dengan demikian, karakter-karakter tersebut akan dihapus dari setiap nilai dalam kolom 'ULASAN'.

#### **Penggunaan .apply() dan re.sub() (lagi):**

Pada langkah ini, dilakukan penghapusan angka dari teks. Ekspresi reguler *r'\d'* cocok dengan digit (angka), dan dengan menggunakan fungsi *re.sub()*, digit tersebut diganti dengan string kosong (dihapus).

#### **Penggunaan .apply() dan re.sub() (sekali lagi):**

Pada langkah ini, dilakukan penghapusan karakter '.' yang berulang-ulang (lebih dari satu titik berturut-turut) dalam teks. Ekspresi reguler *r'\.{2,}'* cocok dengan dua atau lebih titik berturut-turut, dan titik-titik tersebut diganti dengan satu titik menggunakan fungsi *re.sub()*.

## 2. Transform Cases

Tahap ini adalah proses mengubah huruf kapital menjadi huruf kecil dalam teks.

Tabel 3. Tahap *Transform Cases*

Sebelum	Sesudah
Harga Tiket murah tempatnya bersih dan harga tiket murah tempatnya bersih dan gazebo tidak perlu bayar	harga tiket murah tempatnya bersih dan gazebo tidak perlu bayar
Tempatnya bersih sejuk nyaman dan permainan nya murah mantapp	tempatnya bersih sejuk nyaman dan permainan nya murah mantapp
di harihari biasa bukan hari libur jumlah pengunjung yg datang masih bisa dibilang rame cocok buat liburan keluarga	di harihari biasa bukan hari libur jumlah pengunjung yg datang masih bisa dibilang rame cocok buat liburan keluarga
Rekomended untuk acara anak di bawah umur sangat safety	rekomended untuk acara anak di bawah umur sangat safety

```
df['ULASAN'] = df['ULASAN'].apply(lambda x: str(x).lower())
if isinstance(x, str) else x)

comment = 'ULASAN'

df[comment] = df[comment].apply(lambda x: re.sub(r'^a-zA-Z0-9\s+', '', str(x)))

df[comment] = df[comment].apply(lambda x: re.sub(r'\d+', '', str(x)))

df[comment] = df[comment].apply(lambda x:
re.sub(r'\.{2,}', '...', str(x)))

df.head(4501)
```

Program tersebut merupakan bagian dari proses pembersihan atau preprocessing data pada teks yang disimpan dalam DataFrame menggunakan *Python*, dengan fokus pada transformasi kasus (transform cases). Berikut adalah penjelasan setiap tahap dalam program tersebut :

### ***Transformasi Kasus menjadi Lowercase :***

Pada langkah pertama ini, kolom 'ULASAN' di DataFrame diubah ke huruf kecil (*lowercase*). Fungsi *apply()* digunakan untuk menerapkan fungsi lambda ke setiap elemen kolom 'ULASAN'. Fungsi lambda

digunakan untuk memeriksa apakah nilai dalam kolom tersebut merupakan string ( *isinstance(x, str)*). Jika iya, maka nilai tersebut diubah menjadi *lowercase* menggunakan metode *lower()*. Jika bukan, nilai tersebut tidak diubah.

#### **Pembersihan Karakter yang Tidak Diperlukan :**

Pada langkah ini, dilakukan pembersihan karakter yang tidak diperlukan dari teks dalam kolom 'ULASAN'. Fungsi *re.sub()* digunakan untuk mengganti pola tertentu dalam teks dengan string pengganti (dalam hal ini, string kosong ""). Ekspresi reguler *r'^a-zA-Z0-9\sJ'* digunakan untuk mencocokkan karakter selain huruf (a-z dan A-Z), angka (0-9), dan spasi. Karakter-karakter tersebut akan dihapus dari setiap nilai dalam kolom 'ULASAN'.

#### **Penghapusan Angka :**

Pada langkah ini, dilakukan penghapusan angka dari teks dalam kolom 'ULASAN'. Ekspresi reguler *r'\d'* cocok dengan digit (angka), dan dengan menggunakan fungsi *re.sub()*, digit tersebut diganti dengan string kosong (dihapus).

#### **Penghapusan Titik Berulang :**

Pada langkah ini, dilakukan penghapusan karakter '.' yang berulang-ulang (lebih dari satu titik berturut-turut) dalam teks. Ekspresi reguler *r'\.{2,}'* cocok dengan dua atau lebih titik berturut-turut, dan titik-titik tersebut diganti dengan satu titik menggunakan fungsi *re.sub()*.

### **3. Tokenizing**

Tahap ini melakukan proses membagi teks menjadi unit – unit yang lebih kecil seperti kata – kata, frasa atau kalimat. Ini membantu dalam pemrosesan teks lebih lanjut karena masing – masing unit ini dapat diproses secara terpisah.

Tabel 4. Tahap *Tokenizing*

Sebelum	Sesudah
kolamx kotor ada sampah sisa botol air mineral air tdk jernih kaporit kebanyakannya sampai sepet di mulut gigi seperti terkikistehel tajam anakku terlukaberlumut juga hampir semua kolam ada yg menggunakan ban besar susah berenang	['kolamx', 'kotor', 'ada', 'sampah', 'sisa', 'botol', 'air', 'mineral', 'air', 'tdk', 'jernih', 'kaporit', 'kebanyakannya', 'sampai', 'sepet', 'di', 'mulut', 'gigi', 'seperti', 'terkikistehel', 'tajam', 'anakku', 'terlukaberlumut', 'juga', 'hampir', 'semua', 'kolam', 'ada', 'yg', 'menggunakan', 'ban', 'besar', 'susah', 'berenang']
saran tololong di pisahkan kolam yg bs digunakan khusus berenang dan memakai ban apalagi ban besar yg peruntukanx buat wahana seluncuran	['saran', 'tololong', 'di', 'pisahkan', 'kolam', 'yg', 'bs', 'digunakan', 'khusus', 'berenang', 'dan', 'memakai', 'ban', 'apalagi', 'ban', 'besar', 'yg', 'peruntukanx', 'buat', 'wahana', 'seluncuran']
tempat wisata keluarga yang cukup ramai waktu berkunjung di awal pekan fasilitas yang tersedia kolam renang tempat makan aula pertemuan dll	['tempat', 'wisata', 'keluarga', 'yang', 'cukup', 'ramai', 'waktu', 'berkunjung', 'di', 'awal', 'pekan', 'fasilitas', 'yang', 'tersedia', 'kolam', 'renang', 'tempat', 'makan', 'aula', 'pertemuan', 'dll']

```

comment = 'ULASAN'

df['Tokenizing'] = df['ULASAN'].apply(lambda x:
word_tokenize(str(x)) if isinstance(x, str) else [])
df.head(4501)

# Pemeriksaan apakah ada nilai NaN pada kolom
tokenized_text

has_nan = df['Tokenizing'].isna().any()

print(f"Apakah ada nilai NaN pada kolom tokenized_text:
{has_nan}")

```

Program tersebut melanjutkan proses preprocessing data teks dengan melakukan tokenisasi menggunakan library NLTK (Natural Language Toolkit) dalam Python. Berikut adalah penjelasan langkah-langkah dalam program tersebut :

#### **Pendefinisian Variabel comment :**

Variabel **comment** digunakan untuk menyimpan nama kolom yang berisi teks yang akan diproses, dalam hal ini adalah 'ULASAN'.

#### **Tokenisasi Teks :**

Pada langkah ini, teks dalam kolom 'ULASAN' di DataFrame di-tokenisasi, yaitu dibagi menjadi token-token (kata-kata atau bagian-bagian

yang lebih kecil). Fungsi `apply()` digunakan untuk menerapkan fungsi lambda ke setiap nilai dalam kolom 'ULASAN'. Fungsi lambda memeriksa apakah nilai tersebut adalah string ( `isinstance(x, str)`). Jika iya, maka teks tersebut di-tokenisasi menggunakan fungsi `word_tokenize()` dari NLTK. Jika nilai tersebut bukan string (misalnya NaN), maka nilai tersebut diubah menjadi daftar kosong `[]`.

#### ***Menambahkan Kolom 'Tokenizing' ke DataFrame :***

Setelah tokenisasi selesai, hasil tokenisasi ditambahkan ke DataFrame dengan nama kolom baru 'Tokenizing'. Perintah `df.head(4501)` digunakan untuk menampilkan 4501 baris pertama dari DataFrame, termasuk kolom 'Tokenizing' yang baru ditambahkan.

#### ***Pemeriksaan Kehadiran Nilai NaN :***

Pada langkah terakhir, dilakukan pemeriksaan apakah ada nilai NaN dalam kolom 'Tokenizing'. Metode `isna()` digunakan untuk mengecek keberadaan nilai NaN. Jika ada nilai NaN, maka variabel `has_nan` akan bernilai True, jika tidak, maka akan bernilai False. Hasil dari pemeriksaan tersebut dicetak untuk memberikan informasi apakah ada nilai NaN dalam kolom 'Tokenizing'.

#### **4. Stopword**

*Stopword* adalah kata – kata umum yang sering muncul dalam teks dan dianggap tidak memberikan nilai tambah dalam analisis teks, seperti “dan”, “di”, “dari” dan lain – lain. Proses stopword melibatkan penghapusan stopword dari teks karena mereka cenderung tidak membawa informasi penting dalam analisis.

Tabel 5. Tahap *Stopword*

Sebelum	Sesudah
kolamx kotor ada sampah sisa botol air mineral air tdk jernih kaporit kebanyakan sampai sepet di mulut gigi seperti terkikistehel tajam anakku terlukaberlumut juga hampir semua kolam ada yg menggunakan ban besar susah berenang	kolamx kotor sampah sisa botol air mineral air tdk jernih kaporit kebanyakan sepet mulut gigi terkikistehel tajam anakku terlukaberlumut kolam yg ban susah berenang
saran tolong di pisahkan kolam yg bs digunakan khusus berenang dan memakai ban apalagi ban besar yg peruntukanx buat wahana seluncuran	saran tolong pisahkan kolam yg bs khusus berenang memakai ban ban yg peruntukanx wahana seluncuran
tempat wisata keluarga yang cukup ramai waktu berkunjung di awal pekan fasilitas yang tersedia kolam renang tempat makan aula pertemuan dll	wisata keluarga ramai berkunjung pekan fasilitas tersedia kolam renang makan aula pertemuan dll

```
stop_words = set(stopwords.words('indonesian'))  
  
df['Stopword'] = df['Tokenizing'].apply(lambda tokens:  
    ''.join([word for word in tokens if word.lower() not in  
    stop_words]) if isinstance(tokens, list) else '')  
  
df.head(4501)  
  
#CEK NILAI KOSONG SETELAH STOPWORD  
  
has_empty_texts = df['Stopword'].apply(lambda x: len(x)  
== 0 if isinstance(x, list) else False).any()  
  
print(f"Apakah ada teks kosong dalam kolom  
processed_text: {has_empty_texts}")
```

Program tersebut melanjutkan proses *preprocessing* data teks dengan menghilangkan kata-kata umum (*stopwords*) menggunakan library NLTK dalam bahasa Indonesia. Berikut adalah penjelasan langkah-langkah dalam program tersebut :

#### **Pendefinisian Set Stopwords Bahasa Indonesia :**

Langkah ini menggunakan library NLTK untuk mengambil kumpulan kata-kata stopwords dalam bahasa Indonesia. Fungsi *set()* digunakan untuk mengonversi daftar kata-kata stopwords menjadi set, sehingga pencarian kata-kata stopwords menjadi lebih efisien.

#### **Penghapusan Stopwords dari Tokenisasi :**

Pada langkah ini, dilakukan penghapusan kata-kata stopwords dari setiap token dalam kolom 'Tokenizing'. Fungsi *apply()* digunakan untuk

menerapkan fungsi lambda ke setiap nilai dalam kolom 'Tokenizing'. Fungsi lambda memeriksa apakah nilai tersebut adalah daftar (*isinstance(tokens, list)*). Jika iya, maka setiap token akan diperiksa. Jika token tersebut bukan stopwords, maka token tersebut akan disertakan dalam teks hasil, jika bukan, maka token tersebut akan diabaikan. Hasilnya adalah teks yang sudah dihilangkan stopwords dan dijadikan dalam satu kalimat.

#### ***Menambahkan Kolom 'Stopword' ke DataFrame :***

Setelah proses penghapusan stopwords selesai, hasilnya ditambahkan ke DataFrame dengan nama kolom baru 'Stopword'. Perintah *df.head(4501)* digunakan untuk menampilkan 4501 baris pertama dari DataFrame, termasuk kolom 'Stopword' yang baru ditambahkan.

#### ***Pemeriksaan Kehadiran Nilai Teks Kosong Setelah Penghapusan Stopwords :***

Pada langkah terakhir, dilakukan pemeriksaan apakah ada nilai teks kosong setelah penghapusan stopwords. Metode *apply()* digunakan untuk menerapkan fungsi lambda ke setiap nilai dalam kolom 'Stopword'. Fungsi lambda memeriksa apakah panjang teks tersebut adalah 0 (*len(x) == 0*) jika nilai tersebut adalah daftar (*isinstance(x, list)*). Jika iya, maka akan menghasilkan True, jika tidak, akan menghasilkan False. Hasil dari pemeriksaan tersebut dicetak untuk memberikan informasi apakah ada nilai teks kosong dalam kolom 'Stopword'.

#### **5. Stemming**

*Stemming* adalah proses menghilangkan imbuhan atau akhiran kata untuk menghasilkan bentuk dasar dari kata. Tujuannya untuk mengurangi kata – kata bentuk dasarnya agar variasi kata yang serupa dapat dianggap sama. Contohnya, kata “berlari”, “lari”, dan “larian” akan disistem menjadi kata dasar “lar”.

Tabel 6. Tahap *Stemming*

Sebelum	Sesudah
tempatnya bersih kamar mandinya bersih kolam renang ada tempat foto juga ada terus ada tempat anak bermain	tempat bersih kamar mandi bersih kolam renang foto anak main
sangat cocok untuk mengajak keluarga kesini karena kolam renang anak-anak dan dewasanya luas kebersihannya juga terjaga	cocok ajak keluarga kesini kolam renang anak-anak dewasa luas bersih jaga
tempat bagus luas cuma kurangnya pemanfaatan lahan kebunnya padahal bisa dijadikan lahan untuk belajar setiap tanaman yg ada disana	bagus luas kurang manfaat lahan kebun jadi lahan ajar tanam yg sana

```
# Membuat objek stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# Kolom teks yang ingin di-stem stemming = 'Stopword'
# Melakukan stemming pada seluruh kolom teks

df['stemming'] = df[stemming].apply(lambda x: ' '.join([stemmer.stem(word) for word in x.split()]) if isinstance(x, str) else '')

# Menyimpan hasil ke Excel jika diperlukan

df.to_excel('preprocessing.xlsx', index=False,
engine='openpyxl')

df.head(4501)
```

Program tersebut melanjutkan proses preprocessing data teks dengan melakukan stemming menggunakan library Sastrawi dalam Python. Berikut adalah penjelasan langkah-langkah dalam program tersebut :

#### ***Membuat Objek Stemmer :***

Langkah ini menggunakan library Sastrawi untuk membuat objek stemmer. Objek stemmer digunakan untuk melakukan stemming pada kata-kata dalam teks.

#### ***Penentuan Kolom yang Akan di-Stem :***

Langkah ini menentukan kolom yang akan di-Stem. Dalam hal ini, kolom 'Stopword' yang telah dibersihkan dari stopwords sebelumnya akan di-Stem.

#### ***Melakukan Stemming pada Seluruh Kolom Teks :***

Pada langkah ini, dilakukan proses stemming pada seluruh kata dalam kolom 'Stopword'. Fungsi `apply()` digunakan untuk menerapkan fungsi lambda ke setiap nilai dalam kolom 'Stopword'. Fungsi lambda memeriksa apakah nilai tersebut adalah string ( `isinstance(x, str)`). Jika iya, maka kata-kata dalam teks tersebut akan di-Stem menggunakan fungsi `stemmer.stem(word)` dari objek stemmer. Kata-kata yang telah di-Stem kemudian dijadikan dalam satu kalimat dengan menggunakan fungsi '`.join()`'.

#### ***Menyimpan Hasil Preprocessing ke Excel :***

Setelah proses preprocessing selesai, hasilnya disimpan dalam file Excel dengan nama 'preprocessing.xlsx'. Argumen `index=False` digunakan untuk menghilangkan penomoran baris dalam file Excel. Argumen `engine='openpyxl'` digunakan untuk menggunakan library OpenPyXL sebagai engine untuk menulis file Excel.

#### ***Menampilkan DataFrame :***

Perintah ini digunakan untuk menampilkan 4501 baris pertama dari DataFrame, termasuk kolom 'stemming' yang berisi hasil stemming dari teks.

### **D. Model *Convolutional Neural Network* (CNN)**

*Convolutional Neural Network* (CNN) adalah jenis arsitektur jaringan saraf tiruan yang secara khusus dirancang untuk memproses data, seperti teks. Model ini terdiri dari lapisan – lapisan yang disusun secara hierarkis yang mampu mengekstraksi fitur – fitur penting dari data input. Pada tahap ini yaitu pembuatan model *Convolutional Neural Network* pada sistem yang akan dibuat.

Pada tahap ini, melalui penggunaan fungsi – fungsi, program dapat membangun, melatih dan menggunakan model CNN untuk melakukan analisis

sentimen pada ulasan teks dengan langkah – langkah yang sistematis dan efisien.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from sklearn.model_selection import train_test_split
```

Berikut adalah penjelasan singkat tentang setiap baris kode :

***import pandas as pd*** : Mengimpor library pandas dengan alias ***pd***. Pandas digunakan untuk manipulasi dan analisis data, terutama untuk bekerja dengan data dalam bentuk DataFrame.

***import numpy as np*** : Mengimpor library numpy dengan alias ***np***. NumPy digunakan untuk operasi numerik efisien pada array dan matriks multidimensi.

***import matplotlib.pyplot as plt*** : Mengimpor modul pyplot dari library matplotlib dengan alias ***plt***. Ini digunakan untuk membuat visualisasi data seperti plot dan grafik.

***import seaborn as sns*** : Mengimpor library seaborn dengan alias ***sns***. Seaborn adalah library untuk membuat visualisasi statistik yang lebih menarik dan informatif.

***import warnings*** : Mengimpor modul ***warnings***. Warnings digunakan untuk mengelola pesan peringatan yang dihasilkan oleh Python.

***import tensorflow as tf*** : Mengimpor library TensorFlow untuk pemrosesan data dan pembuatan model machine learning.

***from tensorflow import keras*** : Mengimpor modul ***keras*** dari TensorFlow. Keras adalah API tingkat tinggi yang menyediakan antarmuka untuk membangun dan melatih model neural networks.

***from tensorflow.keras import layers*** : Mengimpor modul ***layers*** dari TensorFlow Keras. Ini digunakan untuk membuat berbagai jenis lapisan dalam model neural networks.

***from tensorflow.keras.preprocessing.text import Tokenizer*** : Mengimpor kelas ***Tokenizer*** dari modul ***preprocessing.text*** di TensorFlow Keras. Ini digunakan untuk memproses teks menjadi representasi numerik yang dapat dimengerti oleh model.

***from tensorflow.keras.preprocessing.sequence import pad\_sequences*** : Mengimpor fungsi ***pad\_sequences*** dari modul ***preprocessing.sequence*** di TensorFlow Keras. Ini digunakan untuk membuat panjang seragam dari urutan data teks.

***from sklearn.model\_selection import train\_test\_split*** : Mengimpor fungsi ***train\_test\_split*** dari modul ***model\_selection*** di library scikit-learn. Ini digunakan untuk membagi dataset menjadi subset untuk pelatihan dan pengujian.

Kode tersebut hanya melakukan impor modul dan library yang diperlukan untuk melakukan analisis data dan pembuatan model menggunakan deep learning. Tidak ada operasi konkret yang dilakukan dalam kode tersebut, tetapi modul dan library yang diimpor akan digunakan dalam proses selanjutnya.

```
labels = [1, 0, 2, 1, 0, 1, 1, 0, 1, 0, 1, 2, 1, 0, 1,  
0, 0, 1, 1, 2, 1, 0, 2, 1, 0, 1, 1, 1, 0, 0] # 1:  
positif, 0: negatif, 2: netral
```

Label – label yang diberikan adalah kategori sentimen yang sesuai dengan kalimat – kalimat tertentu. Dimana :

1 : Positif

0 : Negatif

2 : Netral

```

# Import dataset

df =
pd.read_excel("DATAKU_STEM.xlsx", sheet_name='Sheet1')
df
# Preprocessing
reviews = df['ULASAN']
labels = df['LABEL_N']
reviews

```

Kode di atas terdiri dari beberapa langkah :

**Import Dataset** : Melakukan impor dataset dari file Excel dengan menggunakan fungsi pd.read\_excel. Dataset tersebut disimpan dalam variabel df. Dataset tersebut kemudian ditampilkan menggunakan perintah df.

**Preprocessing** : Dilakukan proses preprocessing pada dataset. Pada langkah ini, kolom 'ULASAN' dan 'LABEL\_N' dipilih dari dataset dan disimpan dalam variabel reviews dan labels secara berturut-turut. Kolom 'ULASAN' berisi teks ulasan, sedangkan kolom 'LABEL\_N' berisi label numerik yang sesuai dengan sentimen dari setiap ulasan.

**Return 'reviews'** : Mengembalikan variabel reviews yang berisi teks ulasan. Ini mungkin dilakukan untuk memungkinkan langkah-langkah preprocessing selanjutnya atau pemrosesan data lanjutan.

```

# Filter out non-string elements
non_string_indices = [i for i, item in enumerate(reviews)
if not isinstance(item, str)]
if non_string_indices:
print("Non-string elements found at indices:",
non_string_indices)
reviews = [item for i, item in enumerate(reviews) if
isinstance(item, str)]
labels = labels.drop(non_string_indices)

print("Non-string element at index 1890:", reviews[1890])

reviews = [item for i, item in enumerate(reviews) if
isinstance(item, str)]
labels = df['LABEL_N']

```

Langkah-langkah yang dilakukan dalam kode tersebut adalah sebagai berikut :

**Filtering Non-String Elements** : Kode tersebut mencari indeks dari elemen-elemen dalam variabel **reviews** yang bukan string dengan menggunakan list comprehension. Elemen-elemen non-string ini kemudian dihapus dari **reviews** dan **labels** dengan cara membuat list baru yang hanya berisi elemen-elemen string.

**Menampilkan Indeks Elemen Non-String** : Jika ada elemen non-string yang ditemukan, maka indeks-indeksnya akan ditampilkan. Hal ini membantu dalam mengetahui lokasi di mana elemen non-string tersebut berada.

**Menghapus Elemen Non-String** : Elemen-elemen non-string tersebut dihapus dari variabel **reviews** dan **labels**. Ini dilakukan dengan cara membuat list baru yang hanya berisi elemen-elemen string dan menghapus elemen non-string tersebut.

**Menampilkan Elemen Non-String yang Tersisa** : Kode tersebut mencetak elemen yang diindeks 1890 dari variabel **reviews** yang kemungkinan besar merupakan elemen non-string.

**Perbaikan Penulisan Ulang Variabel reviews dan labels** : Kemudian, ada perbaikan penulisan ulang terhadap variabel **reviews** yang sebelumnya telah dihapus elemen non-string, namun tidak dilakukan terhadap variabel **labels**. Ini memperbaiki kesalahan tersebut dengan menetapkan ulang **labels** menggunakan kolom 'LABEL\_N' dari DataFrame **df**.

Langkah-langkah ini dilakukan untuk membersihkan data dan memastikan bahwa hanya teks ulasan (string) yang digunakan untuk pemrosesan lanjutan atau pembangunan model klasifikasi sentimen.

```
tokenizer = Tokenizer(num_words=10000)
tokenizer.fit_on_texts(reviews)
sequences = tokenizer.texts_to_sequences(reviews)
word_index = tokenizer.word_index
data = pad_sequences(sequences, maxlen=100)
```

**Inisialisasi Tokenizer** : Sebuah objek **Tokenizer** diperkenalkan dengan menentukan parameter **num\_words=10000**. Ini mengindikasikan bahwa hanya

10.000 kata teratas yang akan dipertahankan dalam tokenisasi, yaitu hanya 10.000 kata paling umum yang akan disimpan dalam indeks token. Hal ini membantu mengurangi kompleksitas dan mempercepat proses tokenisasi.

**Fit on Texts:** Metode `fit_on_texts()` dari objek `Tokenizer` digunakan untuk memproses teks dalam variabel `reviews`. Ini akan membuat indeks kata yang berbeda untuk setiap kata dalam teks dan membangun vocabulary dari kata-kata yang ada dalam dataset.

**Texts to Sequences:** Metode `texts_to_sequences()` dari objek `Tokenizer` digunakan untuk mengonversi teks-teks dalam `reviews` menjadi urutan angka-angka berdasarkan indeks kata-kata yang telah dibangun sebelumnya. Setiap kata dalam teks digantikan dengan indeksnya dalam vocabulary yang telah dibangun.

**Word Index :** Variabel `word_index` digunakan untuk menyimpan kamus kata-kata unik yang ditemukan dalam teks dan indeks yang sesuai untuk setiap kata.

**Padding Sequences :** Fungsi `pad_sequences()` digunakan untuk membuat urutan angka-angka menjadi sepanjang 100. Ini dilakukan dengan menambahkan atau memotong urutan angka-angka agar memiliki panjang yang sama dengan `maxlen=100`. Ini penting karena model neural network membutuhkan input dengan ukuran yang konsisten.

Hasil akhirnya adalah variabel `data` yang berisi urutan angka-angka yang telah diproses dan dipad, siap untuk digunakan sebagai input dalam pembangunan model klasifikasi sentimen.

```
# Memisahkan data menjadi data pelatihan dan data validasi
data_train, data_val, labels_train, labels_val =
train_test_split(data, labels, test_size=0.1,
random_state=24)
train_dataset =
tf.data.Dataset.from_tensor_slices((data_train,
labels_train))
val_dataset =
tf.data.Dataset.from_tensor_slices((data_val,
labels_val))
```

Kode di atas melakukan pemisahan data menjadi data pelatihan dan data validasi dengan langkah-langkah sebagai berikut :

**Train-Test Split** : Menggunakan fungsi `train_test_split` dari `sklearn.model_selection`, data (`data`) dan label (`labels`) dipisahkan menjadi dua bagian: data pelatihan (`data_train, labels_train`) dan data validasi (`data_val, labels_val`). Proporsi data yang dialokasikan untuk validasi ditentukan oleh parameter `test_size=0.1`, yang berarti 10% dari data akan dialokasikan untuk validasi.

**Memuat Dataset Tensorflow** : Menggunakan `tf.data.Dataset.from_tensor_slices()`, data pelatihan dan data validasi dipisahkan ke dalam dataset TensorFlow. Ini memungkinkan kita untuk menggunakan dataset tersebut secara efisien dalam pelatihan model dengan TensorFlow.

Hasil akhirnya adalah empat variable :

`data_train` : Data pelatihan yang berisi teks yang sudah diproses dan dipad.

`data_val` : Data validasi yang berisi teks yang sudah diproses dan dipad.

`labels_train` : Label pelatihan yang sesuai dengan data pelatihan.

`labels_val` : Label validasi yang sesuai dengan data validasi.

```
# Define and compile the CNN model
model = keras.Sequential()
model.add(layers.Embedding(len(word_index) + 1, 100,
                           input_length=100))
model.add(layers.Conv1D(128, 5, activation='relu'))
model.add(layers.GlobalMaxPooling1D())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(3, activation='softmax')) # Output layer dengan 3 kelas
```

Pada program tersebut adalah implementasi model *Convolutional Neural Network* (CNN) menggunakan *TensorFlow* dan Keras. Dapat dilihat penjelasannya sebagai berikut :

### 1. Embedding Layer

`'layers.Embedding(len(word_index) + 1, 100, input_length=100)'`: Ini adalah lapisan embedding yang bertanggung jawab untuk mengubah kata-

kata dalam teks menjadi vektor ruang fitur. Parameter pertama (*len(word\_index)* + 1) adalah ukuran kosakata, yaitu jumlah kata unik dalam teks ditambah satu untuk menangani kata-kata yang tidak dikenal. Parameter kedua (100) adalah dimensi vektor embedding. Parameter *input\_length* menentukan panjang dari setiap urutan input yang diberikan.

## 2. Convolutional Layer

**'layers.Conv1D(128, 5, activation='relu')'**: Ini adalah lapisan konvolusi satu dimensi. Conv1D digunakan untuk mengekstrak fitur dari data urutan seperti teks. Parameter pertama (128) adalah jumlah filter yang digunakan untuk mengekstrak fitur. Parameter kedua (5) adalah panjang jendela konvolusi. Fungsi aktivasi 'relu' (Rectified Linear Unit) diterapkan setelah konvolusi.

## 3. Global Max Pooling Layer

**'layers.GlobalMaxPooling1D()'**: Ini adalah lapisan pematatan global yang mengambil nilai maksimum dari fitur-fitur yang diekstraksi oleh konvolusi di seluruh urutan. Ini membantu dalam merampingkan fitur untuk pengolahan lebih lanjut.

## 4. Dense Layer (Hidden Layer)

**'layers.Dense(64, activation='relu')'**: Ini adalah lapisan terhubung penuh (fully connected) dengan 64 unit neuron dan menggunakan fungsi aktivasi ReLU.

ReLU adalah fungsi aktivasi yang paling umum digunakan dalam jaringan saraf buatan saat ini. Fungsi ini didefinisikan sebagai  $f(x) = \max(0, x)$ , yang berarti bahwa nilai keluaran adalah nol jika inputnya negatif dan sama dengan inputnya jika inputnya positif. Keuntungan utama dari ReLU adalah sederhana, mudah dihitung, dan mengatasi masalah gradien yang menghilang (vanishing gradient) yang sering terjadi pada fungsi sigmoid atau tangen hiperbolik (tanh), sehingga mempercepat proses pelatihan model.

## 5. Output Layer

`'layers.Dense(3, activation='softmax')'`: Ini adalah lapisan output.

Lapisan ini memiliki 3 unit neuron yang mewakili kelas-kelas yang mungkin. Fungsi aktivasi softmax digunakan di sini untuk menghasilkan probabilitas untuk setiap kelas.

Softmax adalah fungsi aktivasi yang digunakan pada lapisan output dari jaringan saraf untuk klasifikasi multikelas. Fungsi ini mengubah nilai-nilai dalam lapisan output menjadi probabilitas yang berjumlah satu. Secara matematis, softmax menghitung eksponensial dari nilai input dan membaginya dengan jumlah eksponensial dari semua nilai input di lapisan output. Ini memungkinkan untuk menafsirkan output sebagai distribusi probabilitas atas kelas-kelas yang mungkin. Dalam kasus ini, dengan 3 unit neuron di lapisan output, softmax menghasilkan probabilitas untuk masing-masing dari 3 kelas yang mungkin. Karena probabilitas semuanya berjumlah satu, model dapat memilih kelas dengan probabilitas tertinggi sebagai prediksi akhir.

## E. Pengujian Metode *Convolutional Neural Network* (CNN)

Pengujian metode *Convolutional Neural Network* (CNN) dalam konteks analisis sentimen atau klasifikasi teks mengacu pada proses evaluasi dan penilaian kinerja model CNN yang telah dibangun.

Pengujian metode CNN penting untuk memastikan bahwa model yang dibangun dapat melakukan prediksi dengan akurat dan dapat diandalkan pada data baru yang tidak pernah dilihat sebelumnya. Dengan melakukan pengujian yang cermat, kita dapat mengidentifikasi kelemahan model dan melakukan perbaikan yang diperlukan untuk meningkatkan kinerja model.

Pada tahap ini yaitu pengujian metode *Convolutional Neural Network* untuk menampilkan tingkat akurasi.

```
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])  
model.fit(train_dataset.batch(32), validation_data=val_dataset.batch(32), epochs=100)
```

Dalam kode tersebut, kita melakukan kompilasi model dan melatihnya menggunakan dataset pelatihan dan validasi. Berikut adalah penjelasan lebih rinci :

1. **Kompilasi Model** : Pada langkah ini, model disusun untuk pelatihan dengan menentukan optimizer, fungsi loss, dan metrik evaluasi yang akan digunakan. Dalam kasus ini :
  - a. *Optimizer* : 'adam'. Adam adalah optimizer yang populer digunakan dalam pelatihan model neural network yang secara adaptif menyesuaikan learning rate selama pelatihan.
  - b. *Loss function* : 'sparse\_categorical\_crossentropy'. Fungsi loss ini cocok untuk kasus klasifikasi dengan label integer. Ini menghitung loss antara label yang benar dan prediksi model.
  - c. *Metrics* : ['accuracy']. Metrik evaluasi yang digunakan untuk memantau kinerja model selama pelatihan adalah akurasi, yaitu persentase prediksi yang benar dari total prediksi.
2. **Pelatihan Model** : Model dilatih menggunakan fungsi *fit()*. Dalam kasus ini :
  - a. *train\_dataset.batch(32)*: Data pelatihan (*train\_dataset*) dibagi menjadi batch-batch dengan ukuran 32. Ini memungkinkan untuk memproses sejumlah kecil data sekaligus, yang memungkinkan pelatihan yang lebih efisien.
  - b. *validation\_data=val\_dataset.batch(32)*: Data validasi (*val\_dataset*) juga dibagi menjadi batch-batch dengan ukuran yang sama untuk evaluasi selama pelatihan.
  - c. *epochs=10* : Jumlah epoch atau iterasi pelatihan yang akan dilakukan. Dalam kasus ini, model akan dilatih selama 10 epoch.

Dengan melakukan kompilasi dan pelatihan model ini, kita melatih model CNN untuk melakukan klasifikasi sentimen dengan dataset pelatihan dan validasi yang telah disiapkan sebelumnya. Evaluasi kinerja model dapat dilakukan setelah pelatihan selesai untuk melihat akurasi dan loss pada data validasi.

```

# Lakukan prediksi pada ulasan baru
new_reviews = ["bagus makassar", "nyesel"]
new_sequences
tokenizer.texts_to_sequences(new_reviews)
new_data = pad_sequences(new_sequences, maxlen=100)
predictions = model.predict(new_data)

```

Dalam kode tersebut, dilakukan prediksi sentimen pada ulasan baru menggunakan model CNN yang telah dilatih sebelumnya. Berikut adalah penjelasan langkah-langkahnya :

1. **Membuat Ulasan Baru** : Dua ulasan baru disiapkan dalam variabel new\_reviews.
2. **Mengonversi Ulasan Baru Menjadi Sequences** : Ulasan baru dikonversi menjadi sequences menggunakan objek tokenizer yang telah dibuat sebelumnya. Sequences ini merepresentasikan urutan angka-angka berdasarkan indeks kata-kata dari vocabulary yang telah dibangun pada data pelatihan.
3. **Padding Sequences** : Sequences yang diperoleh dari ulasan baru dipad agar memiliki panjang yang sama dengan panjang maksimum yang telah ditentukan sebelumnya (maxlen=100) untuk konsistensi dengan input yang digunakan dalam pelatihan model.
4. **Melakukan Prediksi** : Data baru yang sudah diproses kemudian digunakan sebagai input untuk melakukan prediksi sentimen menggunakan model CNN yang telah dilatih sebelumnya. Fungsi *predict()* digunakan untuk menghasilkan probabilitas prediksi untuk setiap kelas sentimen (positif, negatif, netral) berdasarkan ulasan yang diberikan.

Hasil prediksi (predictions) akan berupa array probabilitas untuk setiap kelas sentimen. Anda dapat menginterpretasikan hasil prediksi ini untuk menentukan sentimen dari setiap ulasan baru. Biasanya, kelas dengan probabilitas tertinggi akan menjadi prediksi sentimen untuk ulasan tersebut.

```

# Menerjemahkan prediksi
sentiment_labels = ["Positif", "Negatif", "Netral"]
for i, prediction in enumerate(predictions):

```

```

predicted_label=
sentiment_labels[np.argmax(prediction) ]
print(f"Ulasan: {new_reviews[i]} - Prediksi:
{predicted_label}")

```

Dalam kode tersebut, dilakukan penerjemahan hasil prediksi probabilitas menjadi label sentimen yang lebih mudah dipahami. Berikut adalah penjelasan langkah-langkahnya :

1. **Sentimen Labels** : Sebuah list `sentiment_labels` didefinisikan untuk menentukan label yang sesuai dengan setiap kelas sentimen. Indeks dalam list ini harus sesuai dengan indeks kelas sentimen dalam urutan probabilitas yang diberikan oleh model.
2. **Iterasi Melalui Prediksi** : Melalui loop `for`, setiap prediksi dalam hasil prediksi (`predictions`) diakses secara berurutan.
3. **Menerjemahkan Prediksi** : Prediksi probabilitas untuk setiap kelas sentimen diinterpretasikan. Indeks dengan probabilitas tertinggi diambil dengan menggunakan `np.argmax(prediction)` dan kemudian digunakan untuk mengambil label sentimen yang sesuai dari list `sentiment_labels`
4. **Mencetak Hasil** : Hasil prediksi yang telah diterjemahkan dicetak untuk setiap ulasan baru bersama dengan teks ulasan aslinya.

Dengan langkah-langkah ini, hasil prediksi probabilitas yang sebelumnya hanya angka-angka telah diubah menjadi label sentimen yang lebih mudah dimengerti, yang kemudian dicetak untuk setiap ulasan baru. Ini membantu dalam memahami sentimen yang diprediksi oleh model untuk setiap ulasan.

```

# Make predictions on validation data
predictions = model.predict(data_val)
predicted_labels = np.argmax(predictions, axis=1)
# Iterate through data_val and predicted_labels
for i, (text, true_label, predicted_label) in
enumerate(zip(data_val,labels_val,predicted_labels)):
# Menghilangkan token padding dari teks
text = ' '.join([word for word in
tokenizer.sequences_to_texts([text])[0].split() if

```

```

        word != '0'])

    print(f"Index: {i}")
    print(f"Text: {text}")
    print(f"True Label: {true_label}")
    print(f"Predicted Label: {predicted_label}")
    print("-----")

# Create DataFrame from data_val, labels_val, and
predicted labels
df_results = pd.DataFrame({'Text': [' '.join([word for
word in
tokenizer.sequences_to_texts([review])[0].split() if
word != '0']) for review in data_val], 'True Label':
labels_val, 'Predicted Label': predicted_labels})

# Save DataFrame to Excel file
df_results.to_excel('hasil_prediksi.xlsx', index=False)

```

Dalam kode tersebut, dilakukan prediksi pada data validasi menggunakan model CNN yang telah dilatih sebelumnya. Hasil prediksi kemudian ditampilkan dalam bentuk teks dan disimpan ke dalam file Excel. Berikut adalah penjelasan langkah-langkahnya :

- Melakukan Prediksi** : Prediksi dilakukan pada data validasi (*data\_val*) menggunakan model CNN yang telah dilatih sebelumnya. Probabilitas prediksi untuk setiap kelas sentimen diperoleh menggunakan fungsi *predict()*, dan label prediksi yang sesuai diambil dengan menggunakan *np.argmax()* dengan parameter *axis=1*.
- Iterasi Melalui Data Validasi dan Prediksi** : Melalui loop *for*, setiap data validasi, label asli, dan label prediksi diakses secara berurutan. Selama iterasi ini, token padding dihilangkan dari teks dengan memeriksa kata-kata yang bukan token padding.

3. **Mencetak Hasil Prediksi** : Setiap data validasi beserta label asli dan prediksi dicetak ke layar.
4. **Membuat DataFrame** : DataFrame *df\_results* dibuat dengan kolom-kolom 'Text', 'True Label', dan 'Predicted Label' yang berisi teks ulasan, label asli, dan label prediksi, masing-masing.
5. **Menyimpan DataFrame ke Excel** : DataFrame *df\_results* disimpan ke dalam file Excel dengan nama 'hasil\_prediksi.xlsx' tanpa menyertakan indeks.

Dengan langkah-langkah ini, hasil prediksi pada data validasi telah ditampilkan dan disimpan ke dalam file Excel untuk dianalisis lebih lanjut atau dipresentasikan kepada pihak terkait.

Jadi, dalam kode tersebut melakukan iterasi melalui daftar sebagai arsitektur neural network. Selama iterasi, program melatih dan menguji model neural network menggunakan setiap arsitektur yang ada dalam daftar tersebut.

```

Epoch 1/10
127/127 [=====] - 4s 23ms/step - loss: 0.8753 - accuracy: 0.5889 - val_loss: 0.6975 - val_accuracy: 0.7022
Epoch 2/10
127/127 [=====] - 3s 22ms/step - loss: 0.4965 - accuracy: 0.8012 - val_loss: 0.6510 - val_accuracy: 0.7467
Epoch 3/10
127/127 [=====] - 3s 20ms/step - loss: 0.3444 - accuracy: 0.8674 - val_loss: 0.7363 - val_accuracy: 0.7333
Epoch 4/10
127/127 [=====] - 3s 21ms/step - loss: 0.2627 - accuracy: 0.8995 - val_loss: 0.8005 - val_accuracy: 0.7244
Epoch 5/10
127/127 [=====] - 3s 20ms/step - loss: 0.2137 - accuracy: 0.9183 - val_loss: 0.8762 - val_accuracy: 0.7178
Epoch 6/10
127/127 [=====] - 3s 20ms/step - loss: 0.1830 - accuracy: 0.9304 - val_loss: 0.9339 - val_accuracy: 0.7178
Epoch 7/10
127/127 [=====] - 3s 20ms/step - loss: 0.1633 - accuracy: 0.9380 - val_loss: 0.9612 - val_accuracy: 0.7178
Epoch 8/10
127/127 [=====] - 3s 20ms/step - loss: 0.1491 - accuracy: 0.9447 - val_loss: 1.0081 - val_accuracy: 0.7111
Epoch 9/10
127/127 [=====] - 3s 20ms/step - loss: 0.1416 - accuracy: 0.9481 - val_loss: 1.0344 - val_accuracy: 0.7111
Epoch 10/10
127/127 [=====] - 3s 23ms/step - loss: 0.1329 - accuracy: 0.9516 - val_loss: 1.0728 - val_accuracy: 0.7156
<keras.src.callbacks.History at 0x7e3440bfcb80>

```

Gambar 8. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada Epoch pertama (1), untuk Training Accuracy mencapai 58% dan Validation Accuracy 70%. Training Accuracy adalah proporsi dari data training yang diklasifikasikan dengan benar oleh model. Ini memberikan gambaran seberapa baik model berperforma pada data yang telah dilihat selama proses pelatihan. Sedangkan Validation Accuracy adalah proporsi dari data validasi yang diklasifikasikan dengan benar oleh model. Ini

memberikan gambaran seberapa baik model berperforma pada data yang belum pernah dilihat sebelumnya.

Dapat juga dilihat bahwa nilai Training Accuracy meningkat dengan Epoch ke-10 mencapai 95% dan Validation Accuracy 71%. Jika Training Accuracy tinggi, ini menunjukkan bahwa model mampu dengan baik untuk mempelajari pola yang ada dalam data training dan dapat secara efektif mengklasifikasikan data yang sudah dikenal. Sedangkan jika Validation Accuracy tinggi, ini menunjukkan bahwa model memiliki kemampuan untuk generalisasi yang baik dan mungkin tidak mengalami overfitting.

Overfitting adalah kondisi di mana model machine learning terlalu "menghafal" atau terlalu cocok dengan data training, sehingga kinerja model menurun ketika dihadapkan pada data yang belum pernah dilihat sebelumnya. Ini terjadi ketika model menjadi terlalu kompleks atau fleksibel, sehingga mampu menyesuaikan diri dengan kebisingan atau fluktuasi acak dalam data training.

```
Epoch 28/100
127/127 [=====] - 4s 33ms/step - loss: 0.0772 - accuracy: 0.9627 - val_loss: 1.4795 - val_accuracy: 0.6933
Epoch 29/100
127/127 [=====] - 5s 38ms/step - loss: 0.0770 - accuracy: 0.9632 - val_loss: 1.4685 - val_accuracy: 0.6956
Epoch 30/100
127/127 [=====] - 4s 33ms/step - loss: 0.0744 - accuracy: 0.9637 - val_loss: 1.4959 - val_accuracy: 0.6911
Epoch 31/100
127/127 [=====] - 5s 42ms/step - loss: 0.0723 - accuracy: 0.9630 - val_loss: 1.5832 - val_accuracy: 0.6889
Epoch 32/100
127/127 [=====] - 4s 33ms/step - loss: 0.0720 - accuracy: 0.9637 - val_loss: 1.5984 - val_accuracy: 0.6867
Epoch 33/100
127/127 [=====] - 5s 43ms/step - loss: 0.0682 - accuracy: 0.9640 - val_loss: 1.6475 - val_accuracy: 0.6822
Epoch 34/100
127/127 [=====] - 4s 33ms/step - loss: 0.0671 - accuracy: 0.9635 - val_loss: 1.6871 - val_accuracy: 0.6711
Epoch 35/100
127/127 [=====] - 4s 35ms/step - loss: 0.0661 - accuracy: 0.9642 - val_loss: 1.7009 - val_accuracy: 0.6844
Epoch 36/100
127/127 [=====] - 5s 36ms/step - loss: 0.0637 - accuracy: 0.9632 - val_loss: 1.7355 - val_accuracy: 0.6778
Epoch 37/100
127/127 [=====] - 4s 33ms/step - loss: 0.0618 - accuracy: 0.9635 - val_loss: 1.7696 - val_accuracy: 0.6867
Epoch 38/100
127/127 [=====] - 5s 43ms/step - loss: 0.0619 - accuracy: 0.9644 - val_loss: 1.7890 - val_accuracy: 0.6800
```

Gambar 9. Proses Epoch 28 – 39

Sedangkan pada saat mencapai Epoch ke 28 – 50 Validation Accuracy mengalami penurunan dengan nilai 60an%, untuk Training Accuracy mencapai 96%. Untuk Epoch 50 – 100 Validation Accuracynya mengalami pengulangan dengan hasil akhir sebesar 70% sedangkan untuk Training Accuracy tetap dengan nilai 96%.

```
127/127 [=====] - 4s 34ms/step - loss: 0.0474 - accuracy: 0.9681 - val_loss: 3.2055 - val_accuracy: 0.7133
Epoch 92/100
127/127 [=====] - 6s 44ms/step - loss: 0.0476 - accuracy: 0.9679 - val_loss: 3.2055 - val_accuracy: 0.7133
Epoch 93/100
127/127 [=====] - 4s 35ms/step - loss: 0.0475 - accuracy: 0.9677 - val_loss: 3.2245 - val_accuracy: 0.7111
Epoch 94/100
127/127 [=====] - 5s 42ms/step - loss: 0.0475 - accuracy: 0.9677 - val_loss: 3.2223 - val_accuracy: 0.7156
Epoch 95/100
127/127 [=====] - 5s 41ms/step - loss: 0.0475 - accuracy: 0.9672 - val_loss: 3.2393 - val_accuracy: 0.7178
Epoch 96/100
127/127 [=====] - 4s 35ms/step - loss: 0.0475 - accuracy: 0.9679 - val_loss: 3.2509 - val_accuracy: 0.7111
Epoch 97/100
127/127 [=====] - 4s 33ms/step - loss: 0.0475 - accuracy: 0.9677 - val_loss: 3.2667 - val_accuracy: 0.7133
Epoch 98/100
127/127 [=====] - 6s 44ms/step - loss: 0.0475 - accuracy: 0.9674 - val_loss: 3.2663 - val_accuracy: 0.7111
Epoch 99/100
127/127 [=====] - 4s 35ms/step - loss: 0.0477 - accuracy: 0.9686 - val_loss: 3.2571 - val_accuracy: 0.7044
Epoch 100/100
127/127 [=====] - 5s 42ms/step - loss: 0.0488 - accuracy: 0.9679 - val_loss: 3.2165 - val_accuracy: 0.7044
<keras.src.callbacks.History at 0x7f24c95ad390>
```

Gambar 10. Proses Epoch 90 – 100

Gambar tersebut menunjukkan bahwa nilai accuracy menunjukkan nilai Kembali stabil dengan Training accuracy sebesar 96% sedangkan Validation Accuracy sebesar 70%.

### 1. Batch Size 32

#### a. 80 : 20

```
Epoch 1/10
113/113 [=====] - 5s 37ms/step - loss: 0.9444 - accuracy: 0.5292 - val_loss: 0.7851 - val_accuracy: 0.6100
Epoch 2/10
113/113 [=====] - 4s 34ms/step - loss: 0.5405 - accuracy: 0.7811 - val_loss: 0.6403 - val_accuracy: 0.7378
Epoch 3/10
113/113 [=====] - 5s 44ms/step - loss: 0.3856 - accuracy: 0.8544 - val_loss: 0.6712 - val_accuracy: 0.7389
Epoch 4/10
113/113 [=====] - 4s 35ms/step - loss: 0.2764 - accuracy: 0.8914 - val_loss: 0.7311 - val_accuracy: 0.7356
Epoch 5/10
113/113 [=====] - 4s 35ms/step - loss: 0.2242 - accuracy: 0.9139 - val_loss: 0.7896 - val_accuracy: 0.7356
Epoch 6/10
113/113 [=====] - 4s 37ms/step - loss: 0.1926 - accuracy: 0.9289 - val_loss: 0.8381 - val_accuracy: 0.7344
Epoch 7/10
113/113 [=====] - 4s 34ms/step - loss: 0.1717 - accuracy: 0.9350 - val_loss: 0.8730 - val_accuracy: 0.7289
Epoch 8/10
113/113 [=====] - 5s 45ms/step - loss: 0.1578 - accuracy: 0.9428 - val_loss: 0.9065 - val_accuracy: 0.7300
Epoch 9/10
113/113 [=====] - 4s 35ms/step - loss: 0.1471 - accuracy: 0.9458 - val_loss: 0.9335 - val_accuracy: 0.7278
Epoch 10/10
113/113 [=====] - 4s 34ms/step - loss: 0.1351 - accuracy: 0.9503 - val_loss: 0.9669 - val_accuracy: 0.7178
<keras.src.callbacks.History at 0x7cc8a16aff70>
```

Gambar 11. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada pembagian data 80 : 20 dengan batch size 32, Epoch pertama (1), untuk Training Accuracy mencapai 52% dan Validation Accuracy 61%.

```

Epoch 29/100
113/113 [=====] - 4s 35ms/step - loss: 0.0652 - accuracy: 0.9658 - val_loss: 1.3777 - val_accuracy: 0.7256
Epoch 30/100
113/113 [=====] - 4s 35ms/step - loss: 0.0657 - accuracy: 0.9653 - val_loss: 1.3683 - val_accuracy: 0.7189
Epoch 31/100
113/113 [=====] - 5s 46ms/step - loss: 0.0627 - accuracy: 0.9667 - val_loss: 1.4562 - val_accuracy: 0.7133
Epoch 32/100
113/113 [=====] - 4s 35ms/step - loss: 0.0655 - accuracy: 0.9656 - val_loss: 1.4536 - val_accuracy: 0.7156
Epoch 33/100
113/113 [=====] - 5s 42ms/step - loss: 0.0614 - accuracy: 0.9656 - val_loss: 1.5205 - val_accuracy: 0.7156
Epoch 34/100
113/113 [=====] - 4s 35ms/step - loss: 0.0627 - accuracy: 0.9653 - val_loss: 1.5574 - val_accuracy: 0.7100
Epoch 35/100
113/113 [=====] - 4s 34ms/step - loss: 0.0601 - accuracy: 0.9644 - val_loss: 1.6353 - val_accuracy: 0.7078
Epoch 36/100
113/113 [=====] - 5s 46ms/step - loss: 0.0599 - accuracy: 0.9644 - val_loss: 1.6492 - val_accuracy: 0.7156
Epoch 37/100
113/113 [=====] - 4s 35ms/step - loss: 0.0594 - accuracy: 0.9650 - val_loss: 1.6782 - val_accuracy: 0.7144
Epoch 38/100
113/113 [=====] - 5s 43ms/step - loss: 0.0586 - accuracy: 0.9653 - val_loss: 1.6626 - val_accuracy: 0.7222
Epoch 39/100
113/113 [=====] - 4s 36ms/step - loss: 0.0582 - accuracy: 0.9661 - val_loss: 1.6348 - val_accuracy: 0.7200
Epoch 40/100
113/113 [=====] - 4s 34ms/step - loss: 0.0587 - accuracy: 0.9656 - val_loss: 1.7316 - val_accuracy: 0.7133
Epoch 41/100
113/113 [=====] - 5s 46ms/step - loss: 0.0620 - accuracy: 0.9653 - val_loss: 1.6817 - val_accuracy: 0.7189
Epoch 42/100
113/113 [=====] - 4s 35ms/step - loss: 0.0595 - accuracy: 0.9633 - val_loss: 1.7014 - val_accuracy: 0.7089

```

Gambar 12. Proses Epoch 29 – 42

Dari gambar di atas dapat dilihat bahwa pada Epoch 29 untuk semua accuracy mengalami peningkatan dalam setiap prosesnya. Untuk Training Accuracy mencapai dengan nilai 96% sedangkan Validation Accuracy 72%.

```

Epoch 88/100
113/113 [=====] - 5s 43ms/step - loss: 0.0613 - accuracy: 0.9678 - val_loss: 1.8978 - val_accuracy: 0.7089
Epoch 89/100
113/113 [=====] - 4s 39ms/step - loss: 0.0483 - accuracy: 0.9703 - val_loss: 2.0879 - val_accuracy: 0.7100
Epoch 90/100
113/113 [=====] - 4s 35ms/step - loss: 0.0465 - accuracy: 0.9697 - val_loss: 1.9799 - val_accuracy: 0.7167
Epoch 91/100
113/113 [=====] - 5s 46ms/step - loss: 0.0451 - accuracy: 0.9703 - val_loss: 2.0433 - val_accuracy: 0.7133
Epoch 92/100
113/113 [=====] - 4s 35ms/step - loss: 0.0447 - accuracy: 0.9708 - val_loss: 2.1076 - val_accuracy: 0.7144
Epoch 93/100
113/113 [=====] - 4s 36ms/step - loss: 0.0446 - accuracy: 0.9689 - val_loss: 2.1427 - val_accuracy: 0.7133
Epoch 94/100
113/113 [=====] - 4s 38ms/step - loss: 0.0445 - accuracy: 0.9689 - val_loss: 2.1635 - val_accuracy: 0.7122
Epoch 95/100
113/113 [=====] - 4s 36ms/step - loss: 0.0445 - accuracy: 0.9689 - val_loss: 2.1766 - val_accuracy: 0.7144
Epoch 96/100
113/113 [=====] - 5s 47ms/step - loss: 0.0443 - accuracy: 0.9689 - val_loss: 2.2264 - val_accuracy: 0.7189
Epoch 97/100
113/113 [=====] - 5s 41ms/step - loss: 0.0445 - accuracy: 0.9678 - val_loss: 2.2206 - val_accuracy: 0.7144
Epoch 98/100
113/113 [=====] - 4s 37ms/step - loss: 0.0445 - accuracy: 0.9681 - val_loss: 2.2421 - val_accuracy: 0.7178
Epoch 99/100
113/113 [=====] - 4s 36ms/step - loss: 0.0445 - accuracy: 0.9669 - val_loss: 2.2092 - val_accuracy: 0.7144
Epoch 100/100
113/113 [=====] - 5s 47ms/step - loss: 0.0445 - accuracy: 0.9675 - val_loss: 2.2136 - val_accuracy: 0.7133

```

Gambar 13. Proses Epoch 88 – 100

Untuk proses epoch pada gambar di atas menunjukkan bahwa pada nilai Training accuracy tetap stabil atau dengan nilai yang sama yaitu 96% sedangkan Validation Accuracy mengalami penurunan 1%, untuk nilainya sebesar 71%.

## b. 70 : 30

```
Epoch 1/10
99/99 [=====] - 5s 42ms/step - loss: 0.9702 - accuracy: 0.5213 - val_loss: 0.8280 - val_accuracy: 0.6000
Epoch 2/10
99/99 [=====] - 4s 39ms/step - loss: 0.5526 - accuracy: 0.7771 - val_loss: 0.6620 - val_accuracy: 0.7370
Epoch 3/10
99/99 [=====] - 5s 50ms/step - loss: 0.3629 - accuracy: 0.8594 - val_loss: 0.6980 - val_accuracy: 0.7348
Epoch 4/10
99/99 [=====] - 4s 38ms/step - loss: 0.2650 - accuracy: 0.8981 - val_loss: 0.7671 - val_accuracy: 0.7363
Epoch 5/10
99/99 [=====] - 4s 38ms/step - loss: 0.2098 - accuracy: 0.9190 - val_loss: 0.8270 - val_accuracy: 0.7356
Epoch 6/10
99/99 [=====] - 5s 50ms/step - loss: 0.1777 - accuracy: 0.9302 - val_loss: 0.8863 - val_accuracy: 0.7370
Epoch 7/10
99/99 [=====] - 4s 36ms/step - loss: 0.1590 - accuracy: 0.9419 - val_loss: 0.9268 - val_accuracy: 0.7326
Epoch 8/10
99/99 [=====] - 4s 39ms/step - loss: 0.1440 - accuracy: 0.9451 - val_loss: 0.9679 - val_accuracy: 0.7289
Epoch 9/10
99/99 [=====] - 4s 43ms/step - loss: 0.1285 - accuracy: 0.9530 - val_loss: 1.0007 - val_accuracy: 0.7326
Epoch 10/10
99/99 [=====] - 4s 38ms/step - loss: 0.1191 - accuracy: 0.9537 - val_loss: 1.0364 - val_accuracy: 0.7370
<keras.callbacks.History at 0x7cc8b9431b40>
```

Gambar 14. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada pembagian data 70 : 30 dengan batch size 32, Epoch pertama (1), untuk Training Accuracy mencapai 52% dan Validation Accuracy 60%.

```
Epoch 29/100
99/99 [=====] - 5s 49ms/step - loss: 0.0662 - accuracy: 0.9695 - val_loss: 1.3815 - val_accuracy: 0.7074
Epoch 30/100
99/99 [=====] - 4s 39ms/step - loss: 0.0675 - accuracy: 0.9683 - val_loss: 1.3938 - val_accuracy: 0.7074
Epoch 31/100
99/99 [=====] - 4s 36ms/step - loss: 0.0645 - accuracy: 0.9692 - val_loss: 1.4487 - val_accuracy: 0.6985
Epoch 32/100
99/99 [=====] - 5s 50ms/step - loss: 0.0659 - accuracy: 0.9692 - val_loss: 1.4699 - val_accuracy: 0.7037
Epoch 33/100
99/99 [=====] - 4s 38ms/step - loss: 0.0628 - accuracy: 0.9683 - val_loss: 1.5014 - val_accuracy: 0.6941
Epoch 34/100
99/99 [=====] - 4s 42ms/step - loss: 0.0653 - accuracy: 0.9689 - val_loss: 1.5349 - val_accuracy: 0.6978
Epoch 35/100
99/99 [=====] - 4s 38ms/step - loss: 0.0625 - accuracy: 0.9686 - val_loss: 1.5742 - val_accuracy: 0.6830
Epoch 36/100
99/99 [=====] - 4s 39ms/step - loss: 0.0675 - accuracy: 0.9683 - val_loss: 1.7687 - val_accuracy: 0.6881
Epoch 37/100
99/99 [=====] - 5s 49ms/step - loss: 0.0747 - accuracy: 0.9660 - val_loss: 1.5123 - val_accuracy: 0.7096
Epoch 38/100
99/99 [=====] - 4s 36ms/step - loss: 0.0726 - accuracy: 0.9651 - val_loss: 1.5229 - val_accuracy: 0.6941
Epoch 39/100
99/99 [=====] - 4s 44ms/step - loss: 0.0658 - accuracy: 0.9673 - val_loss: 1.5650 - val_accuracy: 0.7007
Epoch 40/100
99/99 [=====] - 4s 39ms/step - loss: 0.0648 - accuracy: 0.9689 - val_loss: 1.5625 - val_accuracy: 0.6941
Epoch 41/100
99/99 [=====] - 4s 37ms/step - loss: 0.0592 - accuracy: 0.9683 - val_loss: 1.6190 - val_accuracy: 0.6904
Epoch 42/100
99/99 [=====] - 5s 49ms/step - loss: 0.0611 - accuracy: 0.9698 - val_loss: 1.6600 - val_accuracy: 0.6941
```

Gambar 15. Proses Epoch 29 – 42

Dari gambar di atas dapat dilihat bahwa pada Epoch 29 untuk semua accuracy mengalami peningkatan dalam setiap prosesnya. Untuk Training Accuracy mencapai dengan nilai 96% sedangkan Validation Accuracy 70%.

```

Epoch 87/100
99/99 [=====] - 4s 37ms/step - loss: 0.0431 - accuracy: 0.9667 - val_loss: 1.9340 - val_accuracy: 0.7222
Epoch 88/100
99/99 [=====] - 5s 50ms/step - loss: 0.0427 - accuracy: 0.9686 - val_loss: 1.9442 - val_accuracy: 0.7287
Epoch 89/100
99/99 [=====] - 4s 39ms/step - loss: 0.0428 - accuracy: 0.9679 - val_loss: 1.9617 - val_accuracy: 0.7200
Epoch 90/100
99/99 [=====] - 4s 37ms/step - loss: 0.0425 - accuracy: 0.9673 - val_loss: 1.9600 - val_accuracy: 0.7259
Epoch 91/100
99/99 [=====] - 5s 53ms/step - loss: 0.0425 - accuracy: 0.9667 - val_loss: 1.9780 - val_accuracy: 0.7281
Epoch 92/100
99/99 [=====] - 4s 39ms/step - loss: 0.0424 - accuracy: 0.9676 - val_loss: 2.0001 - val_accuracy: 0.7281
Epoch 93/100
99/99 [=====] - 4s 39ms/step - loss: 0.0437 - accuracy: 0.9667 - val_loss: 1.9979 - val_accuracy: 0.7215
Epoch 94/100
99/99 [=====] - 5s 53ms/step - loss: 0.0425 - accuracy: 0.9660 - val_loss: 2.0386 - val_accuracy: 0.7252
Epoch 95/100
99/99 [=====] - 4s 37ms/step - loss: 0.0912 - accuracy: 0.9575 - val_loss: 2.1477 - val_accuracy: 0.6711
Epoch 96/100
99/99 [=====] - 4s 41ms/step - loss: 0.1123 - accuracy: 0.9508 - val_loss: 1.9105 - val_accuracy: 0.7156
Epoch 97/100
99/99 [=====] - 4s 39ms/step - loss: 0.0663 - accuracy: 0.9654 - val_loss: 2.0518 - val_accuracy: 0.6926
Epoch 98/100
99/99 [=====] - 5s 51ms/step - loss: 0.0489 - accuracy: 0.9737 - val_loss: 2.0985 - val_accuracy: 0.7148
Epoch 99/100
99/99 [=====] - 4s 39ms/step - loss: 0.0447 - accuracy: 0.9711 - val_loss: 2.1267 - val_accuracy: 0.7104
Epoch 100/100
99/99 [=====] - 4s 40ms/step - loss: 0.0442 - accuracy: 0.9708 - val_loss: 2.0854 - val_accuracy: 0.7141

```

Gambar 16. Proses Epoch 87 – 100

Untuk proses epoch pada gambar di atas menunjukkan bahwa pada nilai Training accuracy tetap stabil dengan nilai yang sama yaitu 96% sedangkan Validation Accuracy mengalami peningkatan dengan nilai 71%.

Untuk hasil Epoch ke 100, pembagian data untuk 90 : 10 dan 80 : 20, memiliki hasil yang sama pada Training Accuracy sebesar 96%, dan untuk data 70 : 30 sebesar 97%. Sedangkan untuk Validation Accuracy dengan pembagian data 80 : 20 dan 70 : 30 memiliki nilai yang sama yaitu 71% dan untuk data 90 : 10 yaitu 70%.

## 2. Batch Size 24

### a. 90 : 10

```

Epoch 1/10
169/169 [=====] - 7s 34ms/step - loss: 0.8546 - accuracy: 0.6030 - val_loss: 0.6935 - val_accuracy: 0.7000
Epoch 2/10
169/169 [=====] - 5s 28ms/step - loss: 0.4863 - accuracy: 0.8081 - val_loss: 0.6806 - val_accuracy: 0.7400
Epoch 3/10
169/169 [=====] - 6s 33ms/step - loss: 0.3396 - accuracy: 0.8706 - val_loss: 0.7634 - val_accuracy: 0.7400
Epoch 4/10
169/169 [=====] - 5s 27ms/step - loss: 0.2583 - accuracy: 0.8985 - val_loss: 0.8357 - val_accuracy: 0.7311
Epoch 5/10
169/169 [=====] - 6s 34ms/step - loss: 0.2107 - accuracy: 0.9230 - val_loss: 0.8988 - val_accuracy: 0.7267
Epoch 6/10
169/169 [=====] - 6s 33ms/step - loss: 0.1830 - accuracy: 0.9319 - val_loss: 0.9699 - val_accuracy: 0.7222
Epoch 7/10
169/169 [=====] - 4s 27ms/step - loss: 0.1631 - accuracy: 0.9402 - val_loss: 1.0199 - val_accuracy: 0.7200
Epoch 8/10
169/169 [=====] - 5s 27ms/step - loss: 0.1505 - accuracy: 0.9427 - val_loss: 1.0539 - val_accuracy: 0.7289
Epoch 9/10
169/169 [=====] - 6s 34ms/step - loss: 0.1396 - accuracy: 0.9481 - val_loss: 1.0889 - val_accuracy: 0.7244
Epoch 10/10
169/169 [=====] - 5s 32ms/step - loss: 0.1333 - accuracy: 0.9514 - val_loss: 1.1213 - val_accuracy: 0.7156
<keras.callbacks.History at 0x7cc89aeed30>

```

Gambar 17. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada pembagian data 90 : 10 dengan batch size 24, Epoch pertama (1), untuk Training Accuracy mencapai 60% dan Validation Accuracy 70%.

```

Epoch 27/100
169/169 [=====] - 6s 33ms/step - loss: 0.0683 - accuracy: 0.9617 - val_loss: 1.6014 - val_accuracy: 0.6978
Epoch 28/100
169/169 [=====] - 5s 28ms/step - loss: 0.0679 - accuracy: 0.9612 - val_loss: 1.5709 - val_accuracy: 0.7022
Epoch 29/100
169/169 [=====] - 5s 32ms/step - loss: 0.0662 - accuracy: 0.9625 - val_loss: 1.6882 - val_accuracy: 0.6956
Epoch 30/100
169/169 [=====] - 6s 34ms/step - loss: 0.0652 - accuracy: 0.9622 - val_loss: 1.6782 - val_accuracy: 0.6867
Epoch 31/100
169/169 [=====] - 5s 28ms/step - loss: 0.0636 - accuracy: 0.9632 - val_loss: 1.7195 - val_accuracy: 0.6911
Epoch 32/100
169/169 [=====] - 5s 32ms/step - loss: 0.0635 - accuracy: 0.9625 - val_loss: 1.7228 - val_accuracy: 0.6889
Epoch 33/100
169/169 [=====] - 6s 33ms/step - loss: 0.0688 - accuracy: 0.9605 - val_loss: 1.7973 - val_accuracy: 0.6844
Epoch 34/100
169/169 [=====] - 5s 27ms/step - loss: 0.0779 - accuracy: 0.9588 - val_loss: 1.8861 - val_accuracy: 0.6867
Epoch 35/100
169/169 [=====] - 5s 28ms/step - loss: 0.0658 - accuracy: 0.9617 - val_loss: 1.8836 - val_accuracy: 0.6911
Epoch 36/100
169/169 [=====] - 5s 32ms/step - loss: 0.0609 - accuracy: 0.9640 - val_loss: 1.9629 - val_accuracy: 0.6889
Epoch 37/100
169/169 [=====] - 6s 33ms/step - loss: 0.0589 - accuracy: 0.9652 - val_loss: 2.0394 - val_accuracy: 0.6933
Epoch 38/100
169/169 [=====] - 5s 27ms/step - loss: 0.0560 - accuracy: 0.9662 - val_loss: 2.0420 - val_accuracy: 0.6933
Epoch 39/100
169/169 [=====] - 5s 30ms/step - loss: 0.0557 - accuracy: 0.9652 - val_loss: 2.0784 - val_accuracy: 0.6933
Epoch 40/100
169/169 [=====] - 5s 31ms/step - loss: 0.0571 - accuracy: 0.9654 - val_loss: 2.0728 - val_accuracy: 0.6778

```

Gambar 18. Proses Epoch 27 – 40

Dari gambar di atas dapat dilihat bahwa pada Epoch 27 untuk Training Accuracy mencapai dengan nilai 96% sedangkan Validation Accuracy mengalami penurunan dengan nilai 70%.

```

Epoch 92/100
169/169 [=====] - 6s 33ms/step - loss: 0.0476 - accuracy: 0.9684 - val_loss: 3.2449 - val_accuracy: 0.6844
Epoch 93/100
169/169 [=====] - 5s 28ms/step - loss: 0.0479 - accuracy: 0.9684 - val_loss: 3.2449 - val_accuracy: 0.6844
Epoch 94/100
169/169 [=====] - 5s 27ms/step - loss: 0.0475 - accuracy: 0.9679 - val_loss: 3.3096 - val_accuracy: 0.6822
Epoch 95/100
169/169 [=====] - 6s 33ms/step - loss: 0.0474 - accuracy: 0.9681 - val_loss: 3.3325 - val_accuracy: 0.6844
Epoch 96/100
169/169 [=====] - 5s 32ms/step - loss: 0.0475 - accuracy: 0.9674 - val_loss: 3.3090 - val_accuracy: 0.6844
Epoch 97/100
169/169 [=====] - 5s 29ms/step - loss: 0.0474 - accuracy: 0.9677 - val_loss: 3.3515 - val_accuracy: 0.6800
Epoch 98/100
169/169 [=====] - 5s 27ms/step - loss: 0.0477 - accuracy: 0.9672 - val_loss: 3.3327 - val_accuracy: 0.6844
Epoch 99/100
169/169 [=====] - 6s 33ms/step - loss: 0.0475 - accuracy: 0.9672 - val_loss: 3.3307 - val_accuracy: 0.6844
Epoch 100/100
169/169 [=====] - 5s 28ms/step - loss: 0.0475 - accuracy: 0.9677 - val_loss: 3.3654 - val_accuracy: 0.6822
<keras.callbacks.History at 0x7f24c82e5de0>

```

Gambar 19. Proses Epoch 92 – 100

Untuk proses epoch ke 100 pada gambar di atas menunjukkan bahwa pada nilai Training accuracy tetap stabil dengan nilai yang sama yaitu 96% sedangkan Validation Accuracy mengalami penurunan dengan nilai 68%.

b. 80 : 20

```

150/150 [=====] - 6s 35ms/step - loss: 0.8850 - accuracy: 0.5808 - val_loss: 0.7133 - val_accuracy: 0.6844
Epoch 2/10
150/150 [=====] - 4s 29ms/step - loss: 0.5033 - accuracy: 0.7989 - val_loss: 0.6609 - val_accuracy: 0.7267
Epoch 3/10
150/150 [=====] - 5s 32ms/step - loss: 0.3429 - accuracy: 0.8664 - val_loss: 0.7071 - val_accuracy: 0.7389
Epoch 4/10
150/150 [=====] - 4s 28ms/step - loss: 0.2556 - accuracy: 0.9019 - val_loss: 0.7780 - val_accuracy: 0.7289
Epoch 5/10
150/150 [=====] - 5s 34ms/step - loss: 0.2062 - accuracy: 0.9197 - val_loss: 0.8414 - val_accuracy: 0.7200
Epoch 6/10
150/150 [=====] - 4s 27ms/step - loss: 0.1781 - accuracy: 0.9347 - val_loss: 0.8991 - val_accuracy: 0.7178
Epoch 7/10
150/150 [=====] - 5s 32ms/step - loss: 0.1595 - accuracy: 0.9419 - val_loss: 0.9303 - val_accuracy: 0.7222
Epoch 8/10
150/150 [=====] - 4s 29ms/step - loss: 0.1489 - accuracy: 0.9464 - val_loss: 0.9774 - val_accuracy: 0.7244
Epoch 9/10
150/150 [=====] - 5s 35ms/step - loss: 0.1364 - accuracy: 0.9522 - val_loss: 1.0148 - val_accuracy: 0.7167
Epoch 10/10
150/150 [=====] - 4s 29ms/step - loss: 0.1260 - accuracy: 0.9536 - val_loss: 1.0543 - val_accuracy: 0.7111
<keras.callbacks.History at 0x7cc89b03d5d0>

```

Gambar 20. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada pembagian data 80 : 20 dengan batch size 24, Epoch pertama (1), untuk Training Accuracy mencapai 58% dan Validation Accuracy 68%.

```

Epoch 29/100
150/150 [=====] - 5s 35ms/step - loss: 0.0604 - accuracy: 0.9650 - val_loss: 1.5155 - val_accuracy: 0.7022
Epoch 30/100
150/150 [=====] - 5s 33ms/step - loss: 0.0588 - accuracy: 0.9647 - val_loss: 1.5917 - val_accuracy: 0.7000
Epoch 31/100
150/150 [=====] - 5s 31ms/step - loss: 0.0583 - accuracy: 0.9650 - val_loss: 1.6242 - val_accuracy: 0.6989
Epoch 32/100
150/150 [=====] - 4s 29ms/step - loss: 0.0587 - accuracy: 0.9647 - val_loss: 1.6588 - val_accuracy: 0.6978
Epoch 33/100
150/150 [=====] - 5s 35ms/step - loss: 0.0652 - accuracy: 0.9636 - val_loss: 1.8337 - val_accuracy: 0.6967
Epoch 34/100
150/150 [=====] - 4s 28ms/step - loss: 0.0884 - accuracy: 0.9539 - val_loss: 1.7474 - val_accuracy: 0.6833
Epoch 35/100
150/150 [=====] - 4s 29ms/step - loss: 0.0905 - accuracy: 0.9536 - val_loss: 1.5728 - val_accuracy: 0.7000
Epoch 36/100
150/150 [=====] - 5s 35ms/step - loss: 0.0704 - accuracy: 0.9644 - val_loss: 1.6918 - val_accuracy: 0.7056
Epoch 37/100
150/150 [=====] - 4s 29ms/step - loss: 0.0501 - accuracy: 0.9681 - val_loss: 1.7951 - val_accuracy: 0.7056
Epoch 38/100
150/150 [=====] - 5s 36ms/step - loss: 0.0539 - accuracy: 0.9672 - val_loss: 1.8328 - val_accuracy: 0.7078
Epoch 39/100
150/150 [=====] - 4s 28ms/step - loss: 0.0532 - accuracy: 0.9672 - val_loss: 1.8454 - val_accuracy: 0.7167
Epoch 40/100
150/150 [=====] - 4s 29ms/step - loss: 0.0534 - accuracy: 0.9656 - val_loss: 1.8446 - val_accuracy: 0.7178
Epoch 41/100
150/150 [=====] - 5s 36ms/step - loss: 0.0532 - accuracy: 0.9653 - val_loss: 1.8394 - val_accuracy: 0.7200
Epoch 42/100
150/150 [=====] - 5s 34ms/step - loss: 0.0537 - accuracy: 0.9653 - val_loss: 1.8389 - val_accuracy: 0.7144

```

Gambar 21. Proses Epoch 29 – 42

Dari gambar di atas dapat dilihat bahwa pada Epoch 29 untuk Training Accuracy mencapai dengan nilai 96% sedangkan Validation Accuracy mengalami peningkatan dengan nilai sebesar 70%.

```

Epoch 87/100
150/150 [=====] - 4s 28ms/step - loss: 0.0446 - accuracy: 0.9653 - val_loss: 2.5198 - val_accuracy: 0.7000
Epoch 88/100
150/150 [=====] - 6s 38ms/step - loss: 0.0449 - accuracy: 0.9656 - val_loss: 2.5075 - val_accuracy: 0.7133
Epoch 89/100
150/150 [=====] - 4s 29ms/step - loss: 0.0455 - accuracy: 0.9658 - val_loss: 2.5699 - val_accuracy: 0.7044
Epoch 90/100
150/150 [=====] - 5s 35ms/step - loss: 0.0448 - accuracy: 0.9683 - val_loss: 2.5847 - val_accuracy: 0.7033
Epoch 91/100
150/150 [=====] - 5s 36ms/step - loss: 0.0448 - accuracy: 0.9669 - val_loss: 2.5988 - val_accuracy: 0.7033
Epoch 92/100
150/150 [=====] - 4s 29ms/step - loss: 0.0448 - accuracy: 0.9656 - val_loss: 2.6242 - val_accuracy: 0.7044
Epoch 93/100
150/150 [=====] - 4s 29ms/step - loss: 0.0757 - accuracy: 0.9642 - val_loss: 2.9981 - val_accuracy: 0.6900
Epoch 94/100
150/150 [=====] - 5s 36ms/step - loss: 0.0978 - accuracy: 0.9558 - val_loss: 2.5640 - val_accuracy: 0.6856
Epoch 95/100
150/150 [=====] - 5s 36ms/step - loss: 0.0569 - accuracy: 0.9678 - val_loss: 2.5156 - val_accuracy: 0.6911
Epoch 96/100
150/150 [=====] - 4s 29ms/step - loss: 0.0463 - accuracy: 0.9706 - val_loss: 2.4772 - val_accuracy: 0.7056
Epoch 97/100
150/150 [=====] - 5s 36ms/step - loss: 0.0447 - accuracy: 0.9706 - val_loss: 2.5334 - val_accuracy: 0.7056
Epoch 98/100
150/150 [=====] - 4s 29ms/step - loss: 0.0441 - accuracy: 0.9703 - val_loss: 2.6189 - val_accuracy: 0.7056
Epoch 99/100
150/150 [=====] - 5s 36ms/step - loss: 0.0438 - accuracy: 0.9700 - val_loss: 2.6642 - val_accuracy: 0.7067
Epoch 100/100
150/150 [=====] - 4s 28ms/step - loss: 0.0438 - accuracy: 0.9694 - val_loss: 2.6690 - val_accuracy: 0.7100

```

Gambar 22. Proses Epoch 87 – 100

Untuk proses epoch ke 100 pada gambar di atas menunjukkan bahwa pada nilai Training accuracy tetap stabil dengan nilai yang sama yaitu 96% sedangkan Validation Accuracy mengalami peningkatan dengan nilai 71%.

### c. 70 : 30

```
Epoch 1/10
132/132 [=====] - 6s 36ms/step - loss: 0.9248 - accuracy: 0.5584 - val_loss: 0.7292 - val_accuracy: 0.6822
Epoch 2/10
132/132 [=====] - 4s 29ms/step - loss: 0.5156 - accuracy: 0.7940 - val_loss: 0.6921 - val_accuracy: 0.7185
Epoch 3/10
132/132 [=====] - 5s 38ms/step - loss: 0.3484 - accuracy: 0.8695 - val_loss: 0.7302 - val_accuracy: 0.7237
Epoch 4/10
132/132 [=====] - 4s 30ms/step - loss: 0.2473 - accuracy: 0.9022 - val_loss: 0.8043 - val_accuracy: 0.7259
Epoch 5/10
132/132 [=====] - 4s 28ms/step - loss: 0.1946 - accuracy: 0.9222 - val_loss: 0.8568 - val_accuracy: 0.7326
Epoch 6/10
132/132 [=====] - 4s 29ms/step - loss: 0.1651 - accuracy: 0.9422 - val_loss: 0.9047 - val_accuracy: 0.7267
Epoch 7/10
132/132 [=====] - 4s 29ms/step - loss: 0.1460 - accuracy: 0.9451 - val_loss: 0.9608 - val_accuracy: 0.7244
Epoch 8/10
132/132 [=====] - 5s 39ms/step - loss: 0.1348 - accuracy: 0.9495 - val_loss: 1.0143 - val_accuracy: 0.7215
Epoch 9/10
132/132 [=====] - 4s 30ms/step - loss: 0.1233 - accuracy: 0.9543 - val_loss: 1.0501 - val_accuracy: 0.7193
Epoch 10/10
132/132 [=====] - 4s 31ms/step - loss: 0.1170 - accuracy: 0.9565 - val_loss: 1.0893 - val_accuracy: 0.7178
<keras.src.callbacks.History at 0x7cc8a1a19e40>
```

Gambar 23. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada pembagian data 70 : 30 dengan batch size 24, Epoch pertama (1), untuk Training Accuracy mencapai 55% dan Validation Accuracy 68%.

```
Epoch 29/100
132/132 [=====] - 4s 29ms/step - loss: 0.0655 - accuracy: 0.9679 - val_loss: 1.3870 - val_accuracy: 0.7163
Epoch 30/100
132/132 [=====] - 4s 29ms/step - loss: 0.0636 - accuracy: 0.9667 - val_loss: 1.4289 - val_accuracy: 0.7133
Epoch 31/100
132/132 [=====] - 5s 36ms/step - loss: 0.0613 - accuracy: 0.9667 - val_loss: 1.4881 - val_accuracy: 0.7052
Epoch 32/100
132/132 [=====] - 4s 31ms/step - loss: 0.0611 - accuracy: 0.9660 - val_loss: 1.5356 - val_accuracy: 0.6970
Epoch 33/100
132/132 [=====] - 4s 31ms/step - loss: 0.0598 - accuracy: 0.9667 - val_loss: 1.5788 - val_accuracy: 0.6963
Epoch 34/100
132/132 [=====] - 4s 31ms/step - loss: 0.0648 - accuracy: 0.9644 - val_loss: 1.6352 - val_accuracy: 0.6933
Epoch 35/100
132/132 [=====] - 4s 31ms/step - loss: 0.0798 - accuracy: 0.9635 - val_loss: 1.5477 - val_accuracy: 0.7007
Epoch 36/100
132/132 [=====] - 5s 38ms/step - loss: 0.0747 - accuracy: 0.9619 - val_loss: 1.7546 - val_accuracy: 0.6837
Epoch 37/100
132/132 [=====] - 4s 31ms/step - loss: 0.0626 - accuracy: 0.9676 - val_loss: 1.7573 - val_accuracy: 0.6933
Epoch 38/100
132/132 [=====] - 4s 31ms/step - loss: 0.0576 - accuracy: 0.9683 - val_loss: 1.7727 - val_accuracy: 0.6978
Epoch 39/100
132/132 [=====] - 4s 33ms/step - loss: 0.0526 - accuracy: 0.9698 - val_loss: 1.7844 - val_accuracy: 0.7030
Epoch 40/100
132/132 [=====] - 4s 31ms/step - loss: 0.0524 - accuracy: 0.9698 - val_loss: 1.8487 - val_accuracy: 0.6963
Epoch 41/100
132/132 [=====] - 5s 39ms/step - loss: 0.0517 - accuracy: 0.9686 - val_loss: 1.8225 - val_accuracy: 0.6948
Epoch 42/100
132/132 [=====] - 4s 31ms/step - loss: 0.0515 - accuracy: 0.9698 - val_loss: 1.8023 - val_accuracy: 0.6978
```

Gambar 24. Proses Epoch 29 – 42

Dari gambar di atas dapat dilihat bahwa pada Epoch 29 untuk Training Accuracy mencapai dengan nilai 96% sedangkan Validation Accuracy mengalami peningkatan dengan nilai sebesar 71%.

```

Epoch 87/100
132/132 [=====] - 5s 37ms/step - loss: 0.0413 - accuracy: 0.9679 - val_loss: 2.3202 - val_accuracy: 0.7030
Epoch 88/100
132/132 [=====] - 4s 31ms/step - loss: 0.0427 - accuracy: 0.9683 - val_loss: 2.3028 - val_accuracy: 0.6978
Epoch 89/100
132/132 [=====] - 5s 38ms/step - loss: 0.0412 - accuracy: 0.9689 - val_loss: 2.3113 - val_accuracy: 0.7015
Epoch 90/100
132/132 [=====] - 4s 30ms/step - loss: 0.0413 - accuracy: 0.9689 - val_loss: 2.3297 - val_accuracy: 0.6978
Epoch 91/100
132/132 [=====] - 4s 32ms/step - loss: 0.0413 - accuracy: 0.9683 - val_loss: 2.3437 - val_accuracy: 0.6985
Epoch 92/100
132/132 [=====] - 5s 36ms/step - loss: 0.0412 - accuracy: 0.9679 - val_loss: 2.3342 - val_accuracy: 0.7022
Epoch 93/100
132/132 [=====] - 4s 31ms/step - loss: 0.0412 - accuracy: 0.9676 - val_loss: 2.3444 - val_accuracy: 0.7030
Epoch 94/100
132/132 [=====] - 5s 36ms/step - loss: 0.0412 - accuracy: 0.9676 - val_loss: 2.3394 - val_accuracy: 0.7015
Epoch 95/100
132/132 [=====] - 4s 31ms/step - loss: 0.0412 - accuracy: 0.9679 - val_loss: 2.3674 - val_accuracy: 0.7037
Epoch 96/100
132/132 [=====] - 4s 32ms/step - loss: 0.0411 - accuracy: 0.9689 - val_loss: 2.3859 - val_accuracy: 0.7059
Epoch 97/100
132/132 [=====] - 5s 40ms/step - loss: 0.0412 - accuracy: 0.9670 - val_loss: 2.3917 - val_accuracy: 0.7044
Epoch 98/100
132/132 [=====] - 4s 30ms/step - loss: 0.0410 - accuracy: 0.9679 - val_loss: 2.4092 - val_accuracy: 0.7022
Epoch 99/100
132/132 [=====] - 4s 31ms/step - loss: 0.0418 - accuracy: 0.9683 - val_loss: 2.3575 - val_accuracy: 0.7030
Epoch 100/100
132/132 [=====] - 4s 32ms/step - loss: 0.0415 - accuracy: 0.9679 - val_loss: 2.3947 - val_accuracy: 0.7000

```

Gambar 25. Proses Epoch 87 – 100

Untuk proses epoch ke 100 pada gambar di atas menunjukkan bahwa pada nilai Training accuracy tetap stabil dengan nilai yang sama yaitu 96% sedangkan Validation Accuracy mengalami penurunan dengan nilai 70%.

Untuk hasil Epoch ke 100, pembagian data untuk 90 : 10, 80 : 20 dan 70 : 30 memiliki hasil yang sama pada Training Accuracy sebesar 96%. Sedangkan untuk Validation Accuracy dengan pembagian data 90 : 10 yaitu 68%, data 80 : 20 yaitu 71% dan 70 : 30 yaitu 70%.

### 3. Batch Size 64

#### a. 90 : 10

```

Epoch 1/10
64/64 [=====] - 7s 76ms/step - loss: 0.9993 - accuracy: 0.5158 - val_loss: 0.8262 - val_accuracy: 0.6244
Epoch 2/10
64/64 [=====] - 4s 61ms/step - loss: 0.5769 - accuracy: 0.7699 - val_loss: 0.6241 - val_accuracy: 0.7578
Epoch 3/10
64/64 [=====] - 5s 79ms/step - loss: 0.3855 - accuracy: 0.8519 - val_loss: 0.6615 - val_accuracy: 0.7511
Epoch 4/10
64/64 [=====] - 4s 60ms/step - loss: 0.2924 - accuracy: 0.8881 - val_loss: 0.7342 - val_accuracy: 0.7489
Epoch 5/10
64/64 [=====] - 4s 59ms/step - loss: 0.2356 - accuracy: 0.9072 - val_loss: 0.8034 - val_accuracy: 0.7444
Epoch 6/10
64/64 [=====] - 5s 78ms/step - loss: 0.1995 - accuracy: 0.9249 - val_loss: 0.8717 - val_accuracy: 0.7444
Epoch 7/10
64/64 [=====] - 4s 61ms/step - loss: 0.1742 - accuracy: 0.9341 - val_loss: 0.9245 - val_accuracy: 0.7333
Epoch 8/10
64/64 [=====] - 4s 64ms/step - loss: 0.1560 - accuracy: 0.9430 - val_loss: 0.9724 - val_accuracy: 0.7333
Epoch 9/10
64/64 [=====] - 4s 60ms/step - loss: 0.1443 - accuracy: 0.9479 - val_loss: 1.0049 - val_accuracy: 0.7244
Epoch 10/10
64/64 [=====] - 4s 59ms/step - loss: 0.1344 - accuracy: 0.9501 - val_loss: 1.0371 - val_accuracy: 0.7311
<keras.callbacks.History at 0x7cc894bdd720>

```

Gambar 26. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada pembagian data 90 : 10 dengan batch size 64, Epoch pertama (1), untuk Training Accuracy mencapai 51% dan Validation Accuracy 62%.

```

Epoch 27/100
64/64 [=====] - 5s 74ms/step - loss: 0.0839 - accuracy: 0.9630 - val_loss: 1.3096 - val_accuracy: 0.6911
Epoch 28/100
64/64 [=====] - 4s 61ms/step - loss: 0.0824 - accuracy: 0.9635 - val_loss: 1.3248 - val_accuracy: 0.6822
Epoch 29/100
64/64 [=====] - 5s 78ms/step - loss: 0.0813 - accuracy: 0.9632 - val_loss: 1.3157 - val_accuracy: 0.6933
Epoch 30/100
64/64 [=====] - 4s 61ms/step - loss: 0.0799 - accuracy: 0.9640 - val_loss: 1.3328 - val_accuracy: 0.6867
Epoch 31/100
64/64 [=====] - 4s 61ms/step - loss: 0.0795 - accuracy: 0.9640 - val_loss: 1.3264 - val_accuracy: 0.6933
Epoch 32/100
64/64 [=====] - 5s 78ms/step - loss: 0.0777 - accuracy: 0.9652 - val_loss: 1.3448 - val_accuracy: 0.6889
Epoch 33/100
64/64 [=====] - 4s 60ms/step - loss: 0.0776 - accuracy: 0.9654 - val_loss: 1.3396 - val_accuracy: 0.6933
Epoch 34/100
64/64 [=====] - 4s 67ms/step - loss: 0.0764 - accuracy: 0.9664 - val_loss: 1.3688 - val_accuracy: 0.6911
Epoch 35/100
64/64 [=====] - 4s 61ms/step - loss: 0.0762 - accuracy: 0.9662 - val_loss: 1.3608 - val_accuracy: 0.6911
Epoch 36/100
64/64 [=====] - 4s 60ms/step - loss: 0.0755 - accuracy: 0.9664 - val_loss: 1.3769 - val_accuracy: 0.6889
Epoch 37/100
64/64 [=====] - 5s 76ms/step - loss: 0.0753 - accuracy: 0.9669 - val_loss: 1.3748 - val_accuracy: 0.6867
Epoch 38/100
64/64 [=====] - 4s 60ms/step - loss: 0.0739 - accuracy: 0.9669 - val_loss: 1.3995 - val_accuracy: 0.6867
Epoch 39/100
64/64 [=====] - 4s 64ms/step - loss: 0.0745 - accuracy: 0.9667 - val_loss: 1.3874 - val_accuracy: 0.6844

```

Gambar 27. Proses Epoch 27 – 39

Dari gambar di atas dapat dilihat bahwa pada Epoch 27 untuk Training Accuracy mencapai dengan nilai 96% sedangkan Validation Accuracy mengalami peningkatan dengan nilai sebesar 69%.

```

Epoch 92/100
64/64 [=====] - 4s 62ms/step - loss: 0.0522 - accuracy: 0.9627 - val_loss: 1.9247 - val_accuracy: 0.6978
Epoch 93/100
64/64 [=====] - 5s 75ms/step - loss: 0.0522 - accuracy: 0.9635 - val_loss: 1.9245 - val_accuracy: 0.6956
Epoch 94/100
64/64 [=====] - 4s 61ms/step - loss: 0.0518 - accuracy: 0.9635 - val_loss: 1.9389 - val_accuracy: 0.6933
Epoch 95/100
64/64 [=====] - 4s 61ms/step - loss: 0.0518 - accuracy: 0.9632 - val_loss: 1.9453 - val_accuracy: 0.6978
Epoch 96/100
64/64 [=====] - 5s 79ms/step - loss: 0.0517 - accuracy: 0.9632 - val_loss: 1.9460 - val_accuracy: 0.6956
Epoch 97/100
64/64 [=====] - 4s 61ms/step - loss: 0.0515 - accuracy: 0.9632 - val_loss: 1.9730 - val_accuracy: 0.7022
Epoch 98/100
64/64 [=====] - 4s 65ms/step - loss: 0.0516 - accuracy: 0.9630 - val_loss: 1.9651 - val_accuracy: 0.6978
Epoch 99/100
64/64 [=====] - 4s 63ms/step - loss: 0.0515 - accuracy: 0.9632 - val_loss: 1.9870 - val_accuracy: 0.6956
Epoch 100/100
64/64 [=====] - 4s 62ms/step - loss: 0.0513 - accuracy: 0.9642 - val_loss: 1.9751 - val_accuracy: 0.6911

```

Gambar 28. Proses Epoch 92 – 100

Untuk proses epoch ke 100 pada gambar di atas menunjukkan bahwa pada nilai Training accuracy dan Validation Accuracy tetap stabil dengan nilai yang sama yaitu 96% dan 69%.

b. 80 : 20

```

Epoch 1/10
57/57 [=====] - 5s 67ms/step - loss: 0.9972 - accuracy: 0.5331 - val_loss: 0.8150 - val_accuracy: 0.6289
Epoch 2/10
57/57 [=====] - 4s 73ms/step - loss: 0.5945 - accuracy: 0.7633 - val_loss: 0.6022 - val_accuracy: 0.7444
Epoch 3/10
57/57 [=====] - 4s 62ms/step - loss: 0.3985 - accuracy: 0.8469 - val_loss: 0.6279 - val_accuracy: 0.7422
Epoch 4/10
57/57 [=====] - 4s 63ms/step - loss: 0.2972 - accuracy: 0.8811 - val_loss: 0.6985 - val_accuracy: 0.7344
Epoch 5/10
57/57 [=====] - 5s 81ms/step - loss: 0.2368 - accuracy: 0.9108 - val_loss: 0.7488 - val_accuracy: 0.7411
Epoch 6/10
57/57 [=====] - 4s 62ms/step - loss: 0.1988 - accuracy: 0.9275 - val_loss: 0.7924 - val_accuracy: 0.7333
Epoch 7/10
57/57 [=====] - 4s 63ms/step - loss: 0.1732 - accuracy: 0.9381 - val_loss: 0.8328 - val_accuracy: 0.7322
Epoch 8/10
57/57 [=====] - 4s 71ms/step - loss: 0.1536 - accuracy: 0.9439 - val_loss: 0.8832 - val_accuracy: 0.7244
Epoch 9/10
57/57 [=====] - 4s 63ms/step - loss: 0.1405 - accuracy: 0.9500 - val_loss: 0.9276 - val_accuracy: 0.7244
Epoch 10/10
57/57 [=====] - 3s 62ms/step - loss: 0.1310 - accuracy: 0.9536 - val_loss: 0.9648 - val_accuracy: 0.7233
<keras.src.callbacks.History at 0x7cc89cb80ac>

```

Gambar 29. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada pembagian data 80 : 20 dengan batch size 64, Epoch pertama (1), untuk Training Accuracy mencapai 53% dan Validation Accuracy 62%.

```

Epoch 29/100
57/57 [=====] - 5s 85ms/step - loss: 0.0779 - accuracy: 0.9667 - val_loss: 1.2123 - val_accuracy: 0.7133
Epoch 30/100
57/57 [=====] - 4s 63ms/step - loss: 0.0779 - accuracy: 0.9683 - val_loss: 1.2447 - val_accuracy: 0.7056
Epoch 31/100
57/57 [=====] - 4s 64ms/step - loss: 0.0769 - accuracy: 0.9678 - val_loss: 1.2336 - val_accuracy: 0.7108
Epoch 32/100
57/57 [=====] - 4s 76ms/step - loss: 0.0767 - accuracy: 0.9681 - val_loss: 1.2645 - val_accuracy: 0.6989
Epoch 33/100
57/57 [=====] - 4s 70ms/step - loss: 0.0753 - accuracy: 0.9681 - val_loss: 1.2644 - val_accuracy: 0.7044
Epoch 34/100
57/57 [=====] - 4s 64ms/step - loss: 0.0753 - accuracy: 0.9686 - val_loss: 1.2957 - val_accuracy: 0.6978
Epoch 35/100
57/57 [=====] - 5s 83ms/step - loss: 0.0749 - accuracy: 0.9689 - val_loss: 1.2877 - val_accuracy: 0.7033
Epoch 36/100
57/57 [=====] - 4s 64ms/step - loss: 0.0743 - accuracy: 0.9703 - val_loss: 1.3272 - val_accuracy: 0.6956
Epoch 37/100
57/57 [=====] - 4s 65ms/step - loss: 0.0747 - accuracy: 0.9692 - val_loss: 1.3149 - val_accuracy: 0.6989
Epoch 38/100
57/57 [=====] - 5s 81ms/step - loss: 0.0735 - accuracy: 0.9706 - val_loss: 1.3617 - val_accuracy: 0.6944
Epoch 39/100
57/57 [=====] - 4s 62ms/step - loss: 0.0754 - accuracy: 0.9686 - val_loss: 1.3454 - val_accuracy: 0.7011
Epoch 40/100
57/57 [=====] - 4s 65ms/step - loss: 0.0738 - accuracy: 0.9706 - val_loss: 1.3639 - val_accuracy: 0.6956
Epoch 41/100
57/57 [=====] - 5s 85ms/step - loss: 0.0744 - accuracy: 0.9689 - val_loss: 1.3752 - val_accuracy: 0.7011
Epoch 42/100
57/57 [=====] - 4s 64ms/step - loss: 0.0728 - accuracy: 0.9700 - val_loss: 1.3978 - val_accuracy: 0.7000

```

Gambar 30. Proses Epoch 29 – 42

Dari gambar di atas dapat dilihat bahwa pada Epoch 29 untuk Training Accuracy mencapai dengan nilai 96% sedangkan Validation Accuracy mengalami peningkatan dengan nilai sebesar 71%.

```

Epoch 87/100
57/57 [=====] - 4s 65ms/step - loss: 0.0545 - accuracy: 0.9625 - val_loss: 1.3936 - val_accuracy: 0.7200
Epoch 88/100
57/57 [=====] - 4s 79ms/step - loss: 0.0544 - accuracy: 0.9622 - val_loss: 1.3936 - val_accuracy: 0.7211
Epoch 89/100
57/57 [=====] - 4s 65ms/step - loss: 0.0547 - accuracy: 0.9625 - val_loss: 1.3869 - val_accuracy: 0.7178
Epoch 90/100
57/57 [=====] - 4s 64ms/step - loss: 0.0540 - accuracy: 0.9622 - val_loss: 1.4087 - val_accuracy: 0.7189
Epoch 91/100
57/57 [=====] - 4s 76ms/step - loss: 0.0543 - accuracy: 0.9628 - val_loss: 1.4005 - val_accuracy: 0.7156
Epoch 92/100
57/57 [=====] - 4s 66ms/step - loss: 0.0536 - accuracy: 0.9608 - val_loss: 1.4215 - val_accuracy: 0.7178
Epoch 93/100
57/57 [=====] - 4s 65ms/step - loss: 0.0535 - accuracy: 0.9625 - val_loss: 1.4250 - val_accuracy: 0.7200
Epoch 94/100
57/57 [=====] - 5s 86ms/step - loss: 0.0534 - accuracy: 0.9622 - val_loss: 1.4316 - val_accuracy: 0.7189
Epoch 95/100
57/57 [=====] - 4s 65ms/step - loss: 0.0535 - accuracy: 0.9628 - val_loss: 1.4005 - val_accuracy: 0.7156
Epoch 96/100
57/57 [=====] - 4s 76ms/step - loss: 0.0536 - accuracy: 0.9622 - val_loss: 1.4234 - val_accuracy: 0.7200
Epoch 97/100
57/57 [=====] - 4s 63ms/step - loss: 0.0534 - accuracy: 0.9631 - val_loss: 1.4248 - val_accuracy: 0.7167
Epoch 98/100
57/57 [=====] - 4s 64ms/step - loss: 0.0525 - accuracy: 0.9625 - val_loss: 1.4477 - val_accuracy: 0.7222
Epoch 99/100
57/57 [=====] - 5s 79ms/step - loss: 0.0537 - accuracy: 0.9631 - val_loss: 1.4145 - val_accuracy: 0.7222
Epoch 100/100
57/57 [=====] - 4s 65ms/step - loss: 0.0528 - accuracy: 0.9631 - val_loss: 1.4456 - val_accuracy: 0.7233

```

Gambar 31. Proses Epoch 87 – 100

Untuk proses epoch ke 100 pada gambar di atas menunjukkan bahwa pada nilai Training accuracy tetap stabil dengan nilai yang sama yaitu 96% dan Validation Accuracy mengalami peningkatan dengan nilai 72%.

### c. 70 : 30

```
Epoch 1/10
50/50 [=====] - 6s 95ms/step - loss: 1.0515 - accuracy: 0.4737 - val_loss: 0.9455 - val_accuracy: 0.5911
Epoch 2/10
50/50 [=====] - 3s 64ms/step - loss: 0.7075 - accuracy: 0.7184 - val_loss: 0.6601 - val_accuracy: 0.7185
Epoch 3/10
50/50 [=====] - 4s 71ms/step - loss: 0.4356 - accuracy: 0.8254 - val_loss: 0.6393 - val_accuracy: 0.7348
Epoch 4/10
50/50 [=====] - 4s 83ms/step - loss: 0.3087 - accuracy: 0.8762 - val_loss: 0.6857 - val_accuracy: 0.7444
Epoch 5/10
50/50 [=====] - 4s 70ms/step - loss: 0.2337 - accuracy: 0.9133 - val_loss: 0.7468 - val_accuracy: 0.7422
Epoch 6/10
50/50 [=====] - 4s 77ms/step - loss: 0.1883 - accuracy: 0.9270 - val_loss: 0.8084 - val_accuracy: 0.7385
Epoch 7/10
50/50 [=====] - 4s 71ms/step - loss: 0.1605 - accuracy: 0.9346 - val_loss: 0.8848 - val_accuracy: 0.7348
Epoch 8/10
50/50 [=====] - 3s 66ms/step - loss: 0.1429 - accuracy: 0.9429 - val_loss: 0.9416 - val_accuracy: 0.7326
Epoch 9/10
50/50 [=====] - 4s 75ms/step - loss: 0.1297 - accuracy: 0.9467 - val_loss: 0.9917 - val_accuracy: 0.7281
Epoch 10/10
50/50 [=====] - 3s 65ms/step - loss: 0.1236 - accuracy: 0.9498 - val_loss: 1.0165 - val_accuracy: 0.7333
<keras.callbacks.History at 0x7cc89cb4e1a0>
```

Gambar 32. Proses Epoch 1 – 10

Dari gambar di atas dapat dilihat bahwa pada pembagian data 70 : 30 dengan batch size 64, Epoch pertama (1), untuk Training Accuracy mencapai 47% dan Validation Accuracy 59%.

```
Epoch 29/100
50/50 [=====] - 5s 91ms/step - loss: 0.0735 - accuracy: 0.9698 - val_loss: 1.3348 - val_accuracy: 0.7030
Epoch 30/100
50/50 [=====] - 3s 66ms/step - loss: 0.0777 - accuracy: 0.9695 - val_loss: 1.3215 - val_accuracy: 0.7059
Epoch 31/100
50/50 [=====] - 4s 72ms/step - loss: 0.0738 - accuracy: 0.9689 - val_loss: 1.3290 - val_accuracy: 0.7007
Epoch 32/100
50/50 [=====] - 4s 71ms/step - loss: 0.0774 - accuracy: 0.9689 - val_loss: 1.3322 - val_accuracy: 0.7000
Epoch 33/100
50/50 [=====] - 4s 72ms/step - loss: 0.0742 - accuracy: 0.9695 - val_loss: 1.3131 - val_accuracy: 0.7037
Epoch 34/100
50/50 [=====] - 4s 79ms/step - loss: 0.0776 - accuracy: 0.9695 - val_loss: 1.3045 - val_accuracy: 0.7059
Epoch 35/100
50/50 [=====] - 4s 81ms/step - loss: 0.0742 - accuracy: 0.9698 - val_loss: 1.2610 - val_accuracy: 0.7141
Epoch 36/100
50/50 [=====] - 4s 71ms/step - loss: 0.0761 - accuracy: 0.9711 - val_loss: 1.2491 - val_accuracy: 0.7119
Epoch 37/100
50/50 [=====] - 4s 71ms/step - loss: 0.0724 - accuracy: 0.9692 - val_loss: 1.2173 - val_accuracy: 0.7141
Epoch 38/100
50/50 [=====] - 5s 93ms/step - loss: 0.0738 - accuracy: 0.9711 - val_loss: 1.2074 - val_accuracy: 0.7222
Epoch 39/100
50/50 [=====] - 4s 71ms/step - loss: 0.0702 - accuracy: 0.9683 - val_loss: 1.2005 - val_accuracy: 0.7148
Epoch 40/100
50/50 [=====] - 3s 69ms/step - loss: 0.0718 - accuracy: 0.9695 - val_loss: 1.1976 - val_accuracy: 0.7207
Epoch 41/100
50/50 [=====] - 4s 70ms/step - loss: 0.0685 - accuracy: 0.9692 - val_loss: 1.1989 - val_accuracy: 0.7156
Epoch 42/100
50/50 [=====] - 3s 66ms/step - loss: 0.0700 - accuracy: 0.9708 - val_loss: 1.1995 - val_accuracy: 0.7148
```

Gambar 33. Proses Epoch 29 – 42

Dari gambar di atas dapat dilihat bahwa pada Epoch 29 mengalami peningkatan sebelumnya untuk Training Accuracy mencapai dengan nilai 96% sedangkan Validation Accuracy mengalami peningkatan dengan nilai sebesar 70%.

```

Epoch 87/100
50/50 [=====] - 4s 78ms/step - loss: 0.0514 - accuracy: 0.9663 - val_loss: 1.3376 - val_accuracy: 0.7230
Epoch 88/100
50/50 [=====] - 4s 71ms/step - loss: 0.0509 - accuracy: 0.9660 - val_loss: 1.3354 - val_accuracy: 0.7237
Epoch 89/100
50/50 [=====] - 4s 72ms/step - loss: 0.0513 - accuracy: 0.9663 - val_loss: 1.3472 - val_accuracy: 0.7244
Epoch 90/100
50/50 [=====] - 4s 80ms/step - loss: 0.0510 - accuracy: 0.9670 - val_loss: 1.3474 - val_accuracy: 0.7244
Epoch 91/100
50/50 [=====] - 4s 72ms/step - loss: 0.0509 - accuracy: 0.9657 - val_loss: 1.3504 - val_accuracy: 0.7230
Epoch 92/100
50/50 [=====] - 4s 73ms/step - loss: 0.0503 - accuracy: 0.9663 - val_loss: 1.3573 - val_accuracy: 0.7222
Epoch 93/100
50/50 [=====] - 4s 80ms/step - loss: 0.0513 - accuracy: 0.9635 - val_loss: 1.3462 - val_accuracy: 0.7230
Epoch 94/100
50/50 [=====] - 4s 71ms/step - loss: 0.0507 - accuracy: 0.9660 - val_loss: 1.3618 - val_accuracy: 0.7230
Epoch 95/100
50/50 [=====] - 4s 75ms/step - loss: 0.0502 - accuracy: 0.9654 - val_loss: 1.3671 - val_accuracy: 0.7230
Epoch 96/100
50/50 [=====] - 4s 84ms/step - loss: 0.0501 - accuracy: 0.9657 - val_loss: 1.3696 - val_accuracy: 0.7237
Epoch 97/100
50/50 [=====] - 4s 72ms/step - loss: 0.0500 - accuracy: 0.9657 - val_loss: 1.3758 - val_accuracy: 0.7222
Epoch 98/100
50/50 [=====] - 4s 81ms/step - loss: 0.0497 - accuracy: 0.9660 - val_loss: 1.3835 - val_accuracy: 0.7230
Epoch 99/100
50/50 [=====] - 4s 74ms/step - loss: 0.0497 - accuracy: 0.9663 - val_loss: 1.3824 - val_accuracy: 0.7207
Epoch 100/100
50/50 [=====] - 4s 72ms/step - loss: 0.0489 - accuracy: 0.9663 - val_loss: 1.3960 - val_accuracy: 0.7237

```

Gambar 34. Proses Epoch 87 – 100

Untuk proses epoch ke 100 pada gambar di atas menunjukkan bahwa pada nilai Training accuracy tetap stabil dengan nilai yang sama yaitu 96% dan Validation Accuracy mengalami peningkatan dengan nilai 72%.

Untuk hasil Epoch ke 100, pembagian data untuk 90 : 10, 80 : 20 dan 70 : 30 memiliki hasil yang sama pada Training Accuracy sebesar 96%. Sedangkan untuk Validation Accuracy dengan pembagian data 90 : 10 yaitu 69% dan untuk data 80 : 20 dan 70 : 30 memiliki hasil yang sama dengan nilai 72%.

#### 4. Pengujian Ulasan Positif (1), Negatif (0), Netral (2)

Tabel 7. Pengujian Dataset dengan Mesin

Ulasan	Text				True Label	Predicted Label
lokernya berbayar di sana	lokernya bayar				0	0
tempatnya buruk tidak keren	tempat	buruk	keren	0	0	
dan kurang nyaman	nyaman					
tidak ada yang istimewa	istimewa				0	0
asik seru aman nyaman	asik seru aman nyaman				1	1
murah meriah tempatnya	murah riah	tempat		1	1	
begitu nyaman	nyaman					

wahana yang ditawarkan wahana tawar muas	0	0
kurang memuaskan kurang hibur fasilitas		
kurangnya hiburan dan tarik		
fasilitas yang menarik		
bangga senang banget bangga senang banget	1	1
makassar wahana bermain makassar wahana main		
keluarga jauhjauh jakarta keluarga jauhjauh jakarta		
fasilitasnya kurang lengkap fasilitas lengkap	0	0
dan gasebonya tidak begitu gasebonya tarik		
menarik		
cocok buat tempat wisata cocok wisata keluarga	1	1
keluarga tempatnya sejuk dan tempat sejuk bersih		
bersih ada kolam renang kolam renang kolam		
kolam pemancingan pancing		
terawat dengan baik sih awat sih	2	2
hari minggu disini sangat minggu ramai sampah	0	0
ramai jadi banyak sampah		
pantai bara salah satu yang pantai bara salah suka	2	2
saya sukai di sini adalah pantai bilang sepi		
pantainya yang masih kunjung orang senang		
terbilang sepi dikunjungi leluasa kemah buru foto		
orang sehingga sangat pantai tanjung bira pantai		
menyenangkan dan leluasa bara		
untuk berkemah atau berburu		
foto di sini seperti di pantai		
tanjung bira pantai bara ini		
juga telah		
tempat seruseruan main air seruseruan main air	1	1
bareng teman atau keluarga bareng teman keluarga		

bagus sih tapi masih lebih bagus sih bagus godis	2	2
bagus di godis kalo saya kalo turut batas menurutku karena di sini dibatasi		
tempat hiburan keluarga areal hibur keluarga areal luas	1	1
luas bersih aman wahananya bersih aman wahana sangat baik		
tempat rekreasi ini lumayan rekreasi lumayan tarik	2	2
menarik		
seru dan memuaskan tapi seru muas sampah	2	2
banyak sampah		
tempatnya bersih rekomlah tempat bersih rekomlah tersedia juga penginapan bagi sedia inap yg malam yg ingin bermalam	2	2
untuk rombongan lebih rompong hemat pakai hemat pakai voucher	2	2
voucher		
tempat bermain anak main anak danbkolam	1	1
danbkolam renang		
tempatnya luas wahananya tempat luas wahana	1	1
ada jamjamnya		
akkarena semakin cantik akkarena cantik cocok	1	1
cocok jadi lokasi piknik akhir lokasi piknik pekan		
pekan bersama keluarga keluarga spot foto keren		
banyak spot foto yang keren fasilitas		
fasilitasnya juga semakin		
baik		
beberapa wahana waktu wahana operasi serentak	2	2
beroperasinya tidak serentak alas		
saya tidak tahu alasan		
layan oke ramah	2	2
layan oke		

permainan wahana air paling main wahana air waw	1	1
waw di makassar sudah makassar masuk golong		
masuk golongan orang orang kaya makassar kalo		
kayanya makassar kalo sudah mandimandi		
pernah mandimandi disini		
mahal sekali!!!	0	0
kolam sangat bagus dan harga tiketnya juga murah	1	1
kolam bagus harga tiket murah		
sangat menyenangkan	0	0
parkir smpit	0	0
bagus buat menghilangkan penat	2	2
dikotabagus penat dikotabagus habis		
menghabiskan waktu diakhir akhir pekan keluarga		
pekan bersama keluarga		
pas masuk cukup sejuk karena musim hujan	1	1
pas masuk sejuk musim hujan sambut kolam inap		
disambut kolam dan penginapan ada kolam besar		
kolam bagi sekat kolam yang terbagi dengan sekat anak-anak lokasi beli		
kolam dalam sedang dan cemilan makan ringan		
untuk anak-anak di lokasi ada tempat beli cemilan makanan		
ringan dan sewa alat renang		
wahana ny bnyk tertutup	2	2
tempatnya nyaman bersih adem	1	1
tempat nyaman bersih adem		
harga ny terlalu mahal	0	0
harga mahal menurut saya		

pantai pasir kasar yang tidak indah dan tidak mengasikkan serta pasir kasarnya tidak halus halus	pantai pasir kasar indah mengasikkan pasir kasar	0	0
juga bagi anda yg ingin yg temu kantor organisasi melakukan pertemuan kantor pilih atau organisasi bisa menjadi pilihan		1	1
tempatnya cuku nyaman dan tempat cuku nyaman harga terjangkau harga jangkau		1	1
pemandangan di tempat ini pandang jelek terasa jelek		0	0
tempat ini memiliki suasana milik suasana senang yang menyenangkan dengan udara sejuk wahana tarik udara yang sejuk dan wahana yang menarik		2	2
kolam renangnya tidak cocok kolam renang cocok untuk anakanak dan dewasa anakanak dewasa		0	0
pantai karena buka dari jam pantai buka jam pagi pagi sampai malam senin malam senin minggu sampai minggu disini tempat luas cafe main tempatnya luas ada cafe ada pantai renang lihat sunset tempat permainannya ada tempat kerenselalu pantainya juga buat berenang eventevent disinioia uang dan yang mau lihat sunset tiket masuk pas jalan ditempat ini kerenselalu ada masuk eventevent disinioia ada uang tiket masuk pas di jalan masuk		2	2

gowa	discovery	park	gowa	discovery	park	2	2
menawarkan berbagai macam			tawar	satwa	burung		
satwa burung dan unggas			unggas	khusus	sulawesi		
khusus di sulawesi serta			wahana	air	anakanak		
wahana air untuk anakanak			keluarga	fasilitas	baik		
dan keluarga namun beberapa							
fasilitas perlu diperbaiki							
liburan yang tak terlupakan		libur lupa batas omong				2	2
hanya sebatas omong kosong		kosong					
ramai kunjungan		ramai kunjung				2	2
pihak pengelola sebaiknya		kelola sedia jalur sandang				2	2
menyediakan jalur buat disabilitas							
penyandang disabilitas							
tempat ini lumayan nyaman		lumayan nyaman santa				2	2
untuk bersantai							
parkiran nya luas dan aman		parkir nya luas aman				1	1
wahananya seru banget		wahana seru banget				1	1
banyak wahana mulai dari		wahana tegang santai					
yang menegangkan sampai							
yang tersantai							
tidak cocok untuk melepas penat		cocok lepas penat				0	0
penat							
pantai yg bersih indah		pantai yg bersih indah				2	2
kotor parkirnya mahal berkali		kotor parkir mahal kali				0	0
bayar rp k		bayar rp k					
pemandangannya sangat		pandang sangat indah				2	2
indah untuk spot foto saat		spot foto suntetapi					
suntetapi sayang masih		sayang sampah unjung					
banyak sampah dari para		tanggung sera pantai					
pengunjung yang tidak							

bertanggung jawab yang berserakan di sekitar pantai			
kurang bagus pemandangan bagus pandang foto disini tidak seperti di foto	0	0	
bersih tenang nyaman dan bersih tenang nyaman kuliner yang enak htm rp kuliner enak htm rp org org	1	1	
banyak yang perlu renovasi renovasi	0	0	
menginap tidak nyaman inap nyaman	0	0	
kurang bagus dan tidak bersih bagus bersih tiket masuk dengan tiket masuk yang mahal mahal	0	0	
masukan buat pengelola kalo masuk kelola kalo sewa bisa penyewaan ban nya di ban nya gratis toilet gratiskan toilet di perbanyak banyak lg lg	2	2	
mungkin lain kali saya akan kali unjung waterboom mengunjungi waterboom bugis water park bugis water park	2	2	
tempat yang bagus buat main bagus main air air	1	1	
biaya masuk sabtu minggu biaya masuk sabtu rborang minggu rborang	2	2	
untuk mengunjungi tempat unjung butuh modal spot ini membutuhkan modal yang foto kena biaya cukup besar karena setiap spot untuk berfoto dikenakan biaya	2	2	
bagus jiwa bagus jiwa	1	1	

tidak aman dan tidak aman senang keluarga menyenangkan untuk keluarga		0	0
tempat ini sangat jelek tidak jelek tarik perhati menarik perhatian		0	0
harga tiketnya mahal	harga tiket mahal	0	0
pengembangannya masih kembang pada kurang dan tidak memadai		0	0
tempat kurang bagus dan tdk bagus tdk bersih harga bersih harga sangat tdk tdk jangkau jelek terjangkau jelek pandangan pandang alam alam		0	0
harganya terjangkauhari kerja harga terjangkauhari rb weekend rbsuasananya kerja rb weekend asribyk pepohonanwahana rbsuasananya asribyk permainangasebonya pepohonanwahana gratistoiletnya dimana adag permainangasebonya licin pokoknya recommended gratistoiletnya mana adag banget dehada harga tiket klo licin pokok rombongan diatas orng recommended banget dehada harga tiket klo rompong atas orng		1	1
minim fasilitas bermain	minim fasilitas main	0	0
sangat bad	bad	0	0
pandangan kota dari tempat pandang kota anggap ini dianggap kurang menarik tarik indah dan tidak begitu indah		0	0
bagus utk ngumpul bagus utk ngumpul		1	1
keluargabisa nginap lagi	keluargabisa nginap		
banyak wahana	wahana	1	1

pelayanan buruk	layan buruk	0	0
sangat nyaman dan bebas	nyaman bebas	1	1
suasana nyaman	suasana nyaman	1	1
jarang ada tempat wisata yang bisa bawa makanan sendiri dan gazebo di sini juga terlihat tidak terawat banyak wahana yang kurang menarik	jarang wisata makan gazebo awat wahana tarik	0	0
jika mau kesini usahalan dari jam pagi sampai jam sore pasti bisa merasakan semua wahana	kesini usahalan jam pagi jam sore rasa wahana	2	2
cukup bersih untuk kamar mandi ditempat umum	bersih kamar mandi tempat	1	1
banyak wahana yang beroperasi membuat kunjungan saya menjadi tidak siasia	wahana operasi kunjung siasia	2	2
tempat ini memiliki wahana menarik	milik wahana tarik	2	2
bersih	bersih	2	2
lumayan rekomended kalau mau ngajak pacar nikmatin senja disini tapi sayang lingkungan sekitarnya kotor	lumayan rekomended ngajak pacar nikmatin senja sayang lingkung	0	0
fasilitas terbatas dengan kurangnya variasi hewan untuk liburan	fasilitas batas kurang variasi hewan libur	0	0
		1	1

permandian yang keren mandi keran nuansa alam dengan nuansa alam yang eksotis wahana muas eksotis tetapi wahana belum sepenuh memuaskan sepenuhnya	2	2
tempat wisata anak tiket wisata anak tiket masuk masuknya murah apalagi hari murah biasa	1	1
taman yg indah tempat taman yg indah inap menginap kantin parkiran kantin parkir luas yang luas tempat memancing pancing aula giat meeting serta ada pula aula untuk ibadah dll yg lanjut dgn berbagai kegiatan seperti rekreasi area yg meeting ibadah dll yg kemudian dapat dilanjutkan dgn rekreasi di area yg sama	1	1
bagus untuk liat sunset bagus liat sunset tempat tempatnya cantik tpi syg yang cantik tpi syg toilet bebrp kurang toiletnya ada ada bilik air tdk jalan tpi bebrp bilik airnya tdk jalan masuk toilet gratiss tpi untuk masuk toilet gratiss	2	2
kami pergi ke sana lagi saat pergi musim hujan musim hujan jadi tidak bisa nikmat matahari terbit menikmati matahari terbit benam dan terbenamnya	2	2
jika ingin mencari cari alam tarik pantai pengalaman yang lebih tanjung kawasan tanjung menarik ada pantai di tanjung yang merupakan bagian dari kawasan yang sama dengan tanjung	2	2

parkirannya sangat luas	parkir luas	1	1
wisata pantai yang mahal dan ramai tidak memberikan pengalaman yang sempurna bagi keluarga	wisata pantai mahal ramai alam sempurna pengalaman yang sempurna keluarga	0	0
kolamnya banyak ada kolam anak mandi anak mandi busa ada kolam busa kolam air hangat air hangat untuk dewasa dan dewasa kolam renang banyak kolam renangnya	kolam kolam anak mandi busa kolam air hangat dewasa kolam renang	1	1
tempatnya teduh banyak tempat pohon pohon banyak wahana dan permainan dengan harga terjangkau	tempat teduh pohon wahana main harga jangkau	1	1
tempatnya rindang karena ditumbuhi banyak pepohonan	tempat rindang tumbuh pohon	1	1
harganya yang mahal memang banyak wahana yang dapat dimainkan tapi menurut saya itu cukup tidak masuk akal apalagi biaya gazebonya juga beda lagi untungnya mereka sering memberikan promo jadi lumayan lah buat saya	harga mahal wahana main masuk akal biaya gazebonya beda untung promo lumayan	2	2
tempatnya baik dan tempat jangkau terjangkau direkomendasikan untuk destinasi air dan kebun binatang mini	tempat jangkau rekomendasi destinasi air kebun binatang mini	0	0

tempatnya	cukup	sempit	tempat	sempit	bagus	0	0
tidak	bagus	untuk	rekreasi	rekreasi	buruk deh		
buruk	deh						
keren	buat	liburan	keluarga	keren	libur	keluarga	1
mantap				mantap			1
gazebo	di	sekitar	tempat	gazebo	wisata	awat	0
wisata	kurang	terawat	dan	kondisi	pada		0
kondisi	juga	tidak	memadai				
sangat	ramai	namun	pasir	dan	ramai	pasir	air kotor
airnya	kotor						0
kolam	penuh	lumut		kolam	penuh	lumut	0
harapannya	waterpark	tidak	harap	waterpark	buka		2
terlalu	terbuka						2
tempatnya	bagus		tempat	bagus			0
tiket	masuknya	k pas	banget	tiket	masuk	k pas	banget
pantainya	ngadep	sunset		pantai	ngadep	sunset	
tempat	yang	indah	semoga	indah	moga	elo	tingkat
pengelolah			bisa	kualitas	bersih	pantai	
meningkatkan		kualitas					
kebersihan	pantai						
wisatanya	buruk	tidak	wisata		buruk	0	0
direkomendasikan		untuk	rekomendasi		libur		
liburan	keluarga		keluarga				
kota	bagus	dengan	konsep	ini	kota	bagus	konsep
kondisinya	kurang	terjaga	kondisi	jaga			0
dengan	baik						0
pantai	nya	hampir	tertutup	pantai	nya	tutup	warung
oleh	warung	makanan	makan	cari	hasil	atur	rapi
meskipun	mereka	mencari	pantai	indah	nikmat		
penghasilan	namun	jika					

diatur rapi maka pantainya akan indah dan dapat dinikmati						
kebersihan yang masih bersih alam senang kurang membuat pengalaman di sini tidak menyenangkan			0	0		
satusatunya tempat satusatunya mandi oke permandian paling oke di makassar makassar menurut saya	1	1				
harga tiket masuk disini juga harga tiket masuk bilang terbilang murah cukup murah bayar k anakanak membayar k untuk anakanak k orang dewasa dan k untuk orang dewasa	1	1				
pengalaman liburan dan reuni alam libur reuni senang tidak menyenangkan	0	0				
terimakasih untuk pemilikx terimakasih pemilikx wiskeb menghadirkan wisata wiskeb hadir wisata yg yg ramah kantong ramah kantong	1	1				
tempat yang menyediakan sedia makan taman main tempat makan dan taman air sewa temu makan bermain air menyewakan enak variatif ganggu tempat pertemuan makan siang orang makanannya enak dan karaoke variatif tidak terganggu ketika makan siang karena tidak ada orang yang berkaraoke	1	1				
kolam renangnya aman untuk kolam renang aman anak anak dan ada pembatas batas kolam renang orang dengan kolam renang orang dewasa kolam renang	1	1				

dewasa kolam renang untuk anak atap kolam renang anak juga diberi atap orang dewasa mungkin yang sedikit kurang kolam renang untuk orang dewasa tidak terlalu dalam		
bugis water park merupakan permandian terbesar yang ada di sulawesi memiliki banyak fasilitas yang lumayan segi aman patuh protokol komplit dan dari segi covid keamanan serta pematuhan protokol covid cukup baik	bugis water park mandi besar sulawesi milik fasilitas lumayan komplit	1 1
pantai ini memiliki pasir putih yang indah sangat mirip dengan tepung terigu terkadang berada di bali	pantai milik pasir putih indah tepung terigu terkadang terasa seperti berada di bali	2 2
hijau banyak pohon airnya bersih dan tidak terlalu bau kaporit nyaman	hijau pohon air bersih bau kaporit nyaman	1 1
tempat refreshing yg sejuk cocok untuk keluarga fasilitas lumayan danharga terjangkau	refreshing yg sejuk cocok keluarga fasilitas lumayan danharga jangkau	1 1
ckup ramai	ckup ramai	2 2

Dari tabel di atas dapat dilihat bahwa ulasan dari hasil prediksi sesuai dengan dataset testing. Dan dalam pengujian ulasan untuk mengetahui analisis sentimen tersebut termasuk jenis kelas yang mana, apakah Positif, Negatif atau Netral.

## **BAB V**

### **PENUTUP**

#### **A. Kesimpulan**

Dari hasil penelitian yang telah dilakukan, Adapun kesimpulan yaitu sebagai berikut :

1. Pengujian dataset sebanyak 4500 ulasan dengan tiga kelas yaitu positif, negatif dan netral. Diperoleh hasil akurasi tertinggi dengan *Training Accuracy* yang meningkat memperoleh nilai akurasi training 95%, serta *Validation Accuracy* memperoleh nilai 73%. Ini merupakan indikasi kuat adanya overfitting dalam model. Overfitting terjadi ketika model belajar terlalu detail pada data training, termasuk noise dan outlier, sehingga performanya sangat baik pada data training tetapi menurun pada data baru atau data yang tidak terlihat sebelumnya (data validation).
2. Metode CNN memiliki kemampuan yang baik dalam memprediksi sentimen positif, negatif, dan netral pada ulasan mengenai Tempat Wisata Makassar. Tingkat akurasi yang tinggi pada tahap pelatihan menunjukkan bahwa model mampu belajar dengan baik dari dataset yang disediakan. Meskipun tingkat akurasi pada tahap validasi sedikit lebih rendah, tetapi masih mencapai angka yang memadai, menunjukkan bahwa model memiliki kemampuan generalisasi yang cukup baik dalam mengklasifikasikan sentimen pada ulasan-ulasan tersebut.
3. Dengan demikian, hasil penelitian ini menunjukkan bahwa metode CNN dapat diandalkan dalam memprediksi sentimen ulasan mengenai Tempat Wisata Makassar, memberikan kontribusi positif dalam analisis sentimen dan evaluasi kualitas layanan atau pengalaman di tempat wisata tersebut.

#### **B. Saran**

Dari hasil penelitian yang dilakukan, dapat disimpulkan beberapa saran yaitu, mengintegrasikan model yang telah dikembangkan ke dalam suatu aplikasi atau platform yang dapat digunakan oleh pengguna nyata (misalnya,

aplikasi pemesanan hotel atau wisata) dapat memberikan manfaat yang lebih langsung dan dapat menguji kelayakan model dalam skenario dunia nyata.



## DAFTAR PUSTAKA

- Abijono, H., Santoso, P., & Anggreini, N. L. (2021). Algoritma Supervised Learning Dan Unsupervised Learning Dalam Pengolahan Data. *Jurnal Teknologi Terapan: G-Tech*, 4(2), 315–318.  
<https://doi.org/10.33379/gtech.v4i2.635>
- Acmad Gani, M. A. (2020). Analisis Kepuasan Wisatawan Terhadap Objek Wisata Bahari di Kota Makassar. *Journal of Management Science (JMS)*, 1(2), 309–324. <https://doi.org/10.52103/jms.v1i2.293>
- Agung, B. A. I. G. N. (2023). *Implementasi Deep Learning untuk Image Clasification menggunakan Arsitektur Convolutional Neural Network (CNN) pada Citra Sampah Hotel (Studi Kasus: Hotel ....)*  
<http://eprints.unram.ac.id/id/eprint/41624%0Ahttp://eprints.unram.ac.id/41624/2/JURNAL TUGAS AKHIR - F1D019037 - I.G.N.A. BAYU ADHIPRAMANA - TEKNIK INFORMATIKA.pdf>
- Asriandy, I. (2016). Strategi Pengembangan Obyek Wisata Air TERJUN BISSAPU DI KABUPATEN BANTAENG. *Universitas Hasanudin*, 82.  
<https://core.ac.uk/download/pdf/77625485.pdf>
- Azzahra, S. A., & Wibowo, A. (2020). Analisis Sentimen Multi-Aspek Berbasis Konversi Ikon Emosi dengan Algoritme Naïve Bayes untuk Ulasan Wisata Kuliner Pada Web Tripadvisor. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 7(4), 737. <https://doi.org/10.25126/jtiik.2020731907>
- Budiman, Q., Mouton, S., Veenhoff, L., & Boersma, A. (2021). 程威特 1 , 吴海涛 1 , 江帆 2. *Jurnal Inovasi Penelitian*, 1(0.1101/2021.02.25.432866), 1–15.
- Buntoro, G. A. (2017). *Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter*. 2(1), 32–41.
- Economics, P., Khaldoon, A., Ahmad, A., Wei, H., Yousaf, I., Ali, S. S., Naveed, M., Latif, A. S., Abdullah, F., Ab Razak, N. H., Palahuddin, S. H., Tasneem Sajjad , Nasir Abbas, Shahzad Hussain, SabeehUllah, A. W., Gulzar, M. A., Zongjun, W., Gunderson, M., Gloy, B., Rodgers, C., Orazalin, N., Mahmood,

- M., ... Ishak, R. B. (2020). No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する共分散構造分析Title. *Corporate Governance (Bingley)*, 10(1), 54–75.
- Gunawan, B., Pratiwi, H. S., & Pratama, E. E. (2018). Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, 4(2), 113.  
<https://doi.org/10.26418/jp.v4i2.27526>
- Hermanto, D. T., Setyanto, A., & Luthfi, E. T. (2021). Algoritma LSTM-CNN untuk Binary Klasifikasi dengan Word2vec pada Media Online. *Creative Information Technology Journal*, 8(1), 64.  
<https://doi.org/10.24076/citec.2021v8i1.264>
- Hidayatullah, A. F., & Nayoan, R. A. N. (2019). *Analisis Sentimen Berbasis Fitur pada Ulasan Tempat Wisata Menggunakan Metode Convolutional Neural Network(CNN)*. www.cnet.com.
- Mestika, J. C., Selan, M. O., & Qadafi, M. I. (2022). *Menjelajahi Teknik-Teknik Supervised Learning untuk Pemodelan Prediktif Menggunakan Python*. 99(99), 216–219.
- Mohamad, N., & Lahay, R. J. (2021). Analisis Nilai Kelestarian Lingkungan Objek Wisata Tasik Ria Berdasarkan Willingness To Pay. *Ideas: Jurnal Pendidikan, Sosial, Dan Budaya*, 7(4), 277.  
<https://doi.org/10.32884/ideas.v7i4.475>
- Muhamad, Z. (2018). Jurnal Teknik Informatika Atmaluhur. *Jurnal Teknik Informatika Atmaluhur*, 6(1), 40.
- Muhammad Haris Diponegoro, Sri Suning Kusumawardani, & Indriana Hidayah. (2021). Tinjauan Pustaka Sistematis: Implementasi Metode Deep Learning pada Prediksi Kinerja Murid. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, 10(2), 131–138. <https://doi.org/10.22146/jnteti.v10i2.1417>
- Name, C., Name, T., Revd, R. T., Lungile, L., World Economic Forum, Fitzpatrick, T., Modeling, L. M., Measurement, F., Snowrift, O. N., Environmental, A. R., Regional, S. S., Power, E., Limited, G. C., Influence, T. H. E., Snow, O. F., On, F., Around, S., Embankment, T. H. E., Wind, I.

- N., ... End, F. Y. (2021). No 主観的健康感を中心とした在宅高齢者における健康関連指標に関する共分散構造分析Title. *Paper Knowledge . Toward a Media History of Documents*, 3(2), 6.
- Nuraini, R. (2015). Desain Algoritma Operasi Perkalian Matriks Menggunakan Metode Flowchart. *Jurnal Teknik Komputer AMIK BSI*, 1(1), 144–151.
- Pelham, I. (2023). Erd2. *Secretary Pathway*, 5, 135–135.  
<https://doi.org/10.1093/oso/9780198599425.003.0085>
- Peryanto, A., Yudhana, A., & Umar, R. (2020). Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network. *Format : Jurnal Ilmiah Teknik Informatika*, 8(2), 138.  
<https://doi.org/10.22441/format.2019.v8.i2.007>
- Prabowo, W. A., & Wiguna, C. (2021). Sistem Informasi UMKM Bengkel Berbasis Web Menggunakan Metode SCRUM. *Jurnal Media Informatika Budidarma*, 5(1), 149. <https://doi.org/10.30865/mib.v5i1.2604>
- QORITA, A. K. (2022). *Analisis Sentimen Berbasis Aspek Pada Ulasan Tempat Wisata DIY*.  
<https://dspace.uii.ac.id/handle/123456789/39202%0Ahttps://dspace.uii.ac.id/bitstream/handle/123456789/39202/18523214.pdf?sequence=1>
- Riadi Silitonga, Y., Munawar, & Noor Hapsari, I. (2019). Analisis Dan Penerapan Datamining Untuk Mendeteksi Berita Palsu (Fake News) Pada Social Media Dengan Memanfaatkan Modul Scikit Learn. *Undergraduate Theses of Information Systems*.
- Ridlo, I. A. (2017). Pedoman Pembuatan Flowchart. *Academia.Edu*, 27.  
[https://www.academia.edu/34767055/Pedoman\\_Pembuatan\\_Flowchart](https://www.academia.edu/34767055/Pedoman_Pembuatan_Flowchart)
- Rifa'i, A., Sujaini, H., & Prawira, D. (2021). Sentiment Analysis Objek Wisata Kalimantan Barat Pada Google Maps Menggunakan Metode Naive Bayes. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, 7(3), 400.  
<https://doi.org/10.26418/jp.v7i3.48132>
- Rosaly, R., & Prasetyo, A. (2020). Flowchart Beserta Fungsi dan Simbol-Simbol. *Journal of Chemical Information and Modeling*, 2(3), 5–7.
- Santoso, J. C., & Bunyamin, H. (2019). Sistem Pakar Diagnosa Penyakit Kulit

- Anjing Dengan Algoritma Pohon Keputusan dan Perhitungan Gini. *Jurnal Strategi*, 1(November), 336.
- Tilasefana, R. A., & Putra, R. E. (2023). Penerapan Metode Deep Learning Menggunakan Algoritma CNN Dengan Arsitektur VGG NET Untuk Pengenalan Cuaca. *Journal of Informatics and Computer Science (JINACS)*, 05(1), 48–57.



## LAMPIRAN

### Lampiran 1. Source Code Preprocessing

```
!pip install Sastrawi

import pandas as pd

#LIBRARY PREPROCESSING
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
import nltk
nltk.download('punkt')
nltk.download('stopwords')
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
import re

df = pd.read_excel('/content/DATA_ASLI.xlsx')
df.head(4501)
df.describe()
df.isnull().sum()
df.dtypes
df.shape
df.duplicated().sum()

df.head(4501)
```

#### a. Tahap Cleaning dan Transform Cases

```
df['ULASAN'] = df['ULASAN'].apply(lambda x: str(x).lower())
if isinstance(x, str) else x)

comment = 'ULASAN'

df[comment] = df[comment].apply(lambda x: re.sub(r'^a-zA-Z0-9\s+', '', str(x)))

df[comment] = df[comment].apply(lambda x: re.sub(r'\d', ' ', str(x)))

df[comment] = df[comment].apply(lambda x: re.sub(r'\.{2,}', '.', str(x)))

df.head(4501)
df.head(4501)
```

**b. Tahap Tokenizing**

```
comment = 'ULASAN'

df['Tokenizing'] = df['ULASAN'].apply(lambda x:
word_tokenize(str(x)) if isinstance(x, str) else [])
df.head(4501)

# Pemeriksaan apakah ada nilai NaN pada kolom tokenized_text
has_nan = df['Tokenizing'].isna().any()
print(f"Apakah ada nilai NaN pada kolom tokenized_text:
{has_nan}")

df.head()
```

**c. Tahap Stopword**

```
stop_words = set(stopwords.words('indonesian'))

df['Stopword'] = df['Tokenizing'].apply(lambda tokens: ' '
.join([word for word in tokens if word.lower() not in
stop_words]) if isinstance(tokens, list) else '')

df.head(4501)

#CEK NILAI KOSONG SETELAH STOPWORD
has_empty_texts = df['Stopword'].apply(lambda x: len(x) == 0
if isinstance(x, list) else False).any()
print(f"Apakah ada teks kosong dalam kolom processed_text:
{has_empty_texts}")
```

**d. Tahap Stemming**

```
# Membuat objek stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# Kolom teks yang ingin di-stem
stemming = 'Stopword'

# Melakukan stemming pada seluruh kolom teks
```

```

df['stemming'] = df[stemming].apply(lambda x: 
    ''.join([stemmer.stem(word) for word in x.split()]) if
    isinstance(x, str) else '')

df.head(4501)

# Menyimpan hasil ke Excel jika diperlukan
df.to_excel('preprocessing.xlsx', index=False,
            engine='openpyxl')

df.head(4501)

```

## Lampiran 2. Source Code

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import
pad_sequences
from sklearn.model_selection import train_test_split

labels = [1, 0, 2, 1, 0, 1, 1, 0, 1, 0, 1, 2, 1, 0, 1, 0, 0, 0,
1, 1, 2, 1, 0, 2, 1, 0, 1, 1, 1, 0, 0] # 1: positif, 0:
negatif, 2: netral

# Import dataset
df = pd.read_excel("DATAKU_STEM.xlsx", sheet_name='Sheet1')

df

# Preprocessing
reviews = df['ULASAN']
labels = df['LABEL_N']

reviews

# Filter out non-string elements

```

```

non_string_indices = [i for i, item in enumerate(reviews) if
not isinstance(item, str)]

if non_string_indices:
    print("Non-string elements found at indices:",
non_string_indices)
    reviews = [item for i, item in enumerate(reviews) if
isinstance(item, str)]
    labels = labels.drop(non_string_indices)

print("Non-string element at index 1890:", reviews[1890])

reviews = [item for i, item in enumerate(reviews) if
isinstance(item, str)]

labels = df['LABEL_N']

tokenizer = Tokenizer(num_words=10000)
tokenizer.fit_on_texts(reviews)
sequences = tokenizer.texts_to_sequences(reviews)
word_index = tokenizer.word_index
data = pad_sequences(sequences, maxlen=100)

# Memisahkan data menjadi data pelatihan dan data validasi
data_train, data_val, labels_train, labels_val =
train_test_split(data, labels, test_size=0.3, random_state=24)

train_dataset =
tf.data.Dataset.from_tensor_slices((data_train, labels_train))
val_dataset = tf.data.Dataset.from_tensor_slices((data_val,
labels_val))

# Define and compile the CNN model
model = keras.Sequential()
model.add(layers.Embedding(len(word_index) + 1, 100,
input_length=100))
model.add(layers.Conv1D(128, 5, activation='relu'))
model.add(layers.GlobalMaxPooling1D())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(3, activation='softmax')) # Output
layer dengan 3 kelas
model.compile(optimizer='adam',
loss='sparse_categorical_crossentropy', metrics=['accuracy'])

```

```

model.compile(optimizer='adam',
loss='sparse_categorical_crossentropy', metrics=['accuracy'])

model.fit(train_dataset.batch(64),
validation_data=val_dataset.batch(64), epochs=100)

# Lakukan prediksi pada ulasan baru
new_reviews = ["bagus makassar", "nyesel"]
new_sequences = tokenizer.texts_to_sequences(new_reviews)
new_data = pad_sequences(new_sequences, maxlen=100)
predictions = model.predict(new_data)

# Menerjemahkan prediksi
sentiment_labels = ["Positif", "Negatif", "Netral"]
for i, prediction in enumerate(predictions):
    predicted_label = sentiment_labels[np.argmax(prediction)]
    print(f"Ulasan: {new_reviews[i]} - Prediksi: {predicted_label}")

# Make predictions on validation data
predictions = model.predict(data_val)
predicted_labels = np.argmax(predictions, axis=1)

# Iterate through data_val and predicted_labels
for i, (text, true_label, predicted_label) in enumerate(zip(data_val, labels_val, predicted_labels)):
    # Menghilangkan token padding dari teks
    text = ' '.join([word for word in
tokenizer.sequences_to_texts([text])[0].split() if word != '0'])

    print(f"Index: {i}")
    print(f"Text: {text}")
    print(f"True Label: {true_label}")
    print(f"Predicted Label: {predicted_label}")
    print("-----")

# Create DataFrame from data_val, labels_val, and predicted labels
df_results = pd.DataFrame({'Text': [' '.join([word for word in
tokenizer.sequences_to_texts([review])[0].split() if word != '0']) for review in data_val], 'True Label': labels_val,
'Predicted Label': predicted_labels})

# Save DataFrame to Excel file

```

```
df_results.to_excel('hasil_prediksi.xlsx', index=False)
```

### Lampiran 3. Dataset Ulasan Positif

	ULASAN	LABEL
1	Tempatnya sejuk. Lebih cocok untuk rekreasi anak-anak. Tiket weekdays 15.000 / orang. Tempat duduk duduk banyak. Ada pemancingan, wahana bermain anak, serta sepeda listrik atau ATV, juga kolam renang.	Positif
2	Banyak pohon-pohon yang berbau juga	Positif
3	Kebun wisata yang sangat dekat dari jalan besar sehingga mudah dijangkau berbagai jenis kendaraan, dengan perjalanan sekitar 25-35 menit dari kota Makassar.	Positif
4	Tiket pada hari biasa 15 k, dan pada Sabtu-Minggu 25 k.	Positif
5	Pas masuk cukup sejuk karena musim hujan, disambut kolam dan penginapan. Ada 3 kolam besar yang terbagi dengan sekat, kolam dalam, sedang dan untuk anak-anak. Di lokasi ada tempat beli cemilan makanan ringan dan sewa alat renang.	Positif
6	Rekomended juga krna banyak disediakan tempat duduk dan berteduh di pinggiran kolam. Di sini juga disediakan aula. Sangat cocok buat liburan bareng keluarga	Positif
7	Karena sekolah anak sy adakan outbound di wisata kebun jd setelah sekian purnama akhirnya bisa lg ke wisata kebun rame² bareng dgn orgtua siswa yg lainnya,, pas masuk ke area parkir kesan pertama ya , area parkirnya luas , trs ke bagian loker u/ lelaki masuk , karyawannya ramah bngt .	Positif
8	Begitu udah masuk ke dlm area bener² kereenn banget , semua tempat dlm area wiskeb cucok banget buat foto² , pokoknya memori hp bakalan full	Positif
9	Anak sy betah dgn kegiatan outboundnya setelah outbound berenang deh..	Positif
10	Di wisata kebun itu harga tiket terjangkau, fasilitas oke , area bermain anak banyak , kantin jd ada , kolam pancing ada , gasebo² gratis bersih juga 🌸🌸🌸	Positif
11	Tiket masuknya perorong Rp.15.000, ada kebum durian, rambutan. Yang mau berenang juga disediakan 4 kolam renang untuk anak-anak & dewasa, ada penginapan, kantin, tempat mancing, banyak spot foto-foto yang bagus juga. Kemarin nyobain kereta keliling wisata kebun 2x hanya Rp.5.000	Positif
12	T4 ini bagus buat berwisata keluarga nyaman adem fasilitas bagus tersedia aneka permainan utk anak2	Positif
13	Salah satu tempat wisata yang nyaman untuk didatangi, ada banyak yang bisa dikenal di sini, bahkan disediakan banyak tempat untuk istirahat.	Positif
14	Saran tolol di pisahkan kolam yg bs digunakan khusus berenang dan memakai ban apalagi ban besar yg peruntukan buat wahan seluncuran.	Positif
15	Tempat wisata keluarga yang cukup ramai waktu berkunjung di awal pekan. Fasilitas yang tersedia kolam renang, tempat makan, aula pertemuan dll.	Positif
16	Aman, nyaman, bersih, ada tlpn nginap, kolam renangnya banyak, pengawasan untuk anak² yg ikut berenang juga ada 🌸	Positif
17	Tempat yang nyaman buat liburan bersama keluarga dan teman-teman	Positif
18	Suasana masih rindang banyak spot foto yg bagus cocok buat keluarga berwisata atau sekedar membawa anak bermain. Krn dsna jd tersedia wahana brmain ana Skrg sdh semakin besar & makin banyak fasilitasnya spt kolam renang yg ditambah 2 plus dgn tmpt main perosotan yg pake ban pelampung, tempat mancing, penginapan, kantin, spot foto yg menarik, sampai tmpt bikin acara gathering	Positif
19	Tempat wisata yang sangat menarik krn tempat wisata ini sangat lengkap dengan kamar mandi dan kamar ganti pakaian serta Gazebo tempat dimana kita bisa duduk dan bersantai menikmati hari libur dan untuk menghilangkan rasa jemu dan penat setelah bekerja.	Positif
20	tiket masuk pada hari Senin-Jumat cuma rp.15.000 dan di hari weekend Sabtu dan minggu cuma rp.25.000. pokoknya mantap 🌸 main sampai puas	Positif
21	Tempat ini sangat cocok untuk liburan keluarga, fasilitas lengkap seperti wc umum, musholla, kantin dll.	Positif
22	Tempat wisata murah meriah sekali menyenangkan di Gowa.	Positif
23	Wahana sangat banyak mulai dari kolam renang yang terbagi jadi dua area besar, lengkap dengan seluncuran dan ban luncur yang bisa disewakan.	Positif
24	Ada juga permainan anak-anak seperti kereta, turning wheel, skuter, bomboom car yang tiketnya sangat terjangkau 5-10rb per satu kali main.	Positif
25	Gazebo dan tempat duduk juga banyak sepanjang area wisata, gratis tanpa dipungut biaya.	Positif
26	Disini juga pengunjung bebas buat makanan dari luar, yang mau makan di restonya juga bisa dengan harga makanan dan minuman yang lumayan terjangkau dari 15rbuan keatas.	Positif
27	Tiket masuk sangat terjangkau untuk weekdays 15rb dan weekend 25rb. Anak usia 2thn keatas full tiket.	Positif
28	Suasana nyaman krn sejuk penuh dengan pohon-pohon. Di sekitar kolam banyak rumah-rumah yang bisa dipake sesuka hati. Yang penting jaga kebersihan.	Positif
29	Kolamnya banyak, ada kolam anak mandi busa, ada kolam air hangat untuk dewasa dan banyak kolam renangnya.	Positif
30	Tempat liburan yg bagus. Tiket murah.. dan gazebo gratis.. bersih dan luas..	Positif
31	Tempatnya bagus. HTM murah. Gazebo gratis, bebas buat makanan di luar, banyak wahana, ada ATV ,ada t4 bermain anak, kolam nya juga banyak kurleb 5 kolam dgn jenis kolam yg berbeda, ada Waterboom dgn luncuran yg tinggi buat 15+ ada juga yg buat anak², ada kolam biasa untuk belajar renang bukan anak dan dewasa .., dan sawang nya pas datang hujan deras banget jadi gak bisa keliling hanya main di kolam renang aja	Positif
32	Untuk area dalam dilengkapi penginapan dan gazebo. Tersedia beberapa pilihan kolam renang baik untuk dewasa dan anak yg di lengkapi dgn area bermain dan seluncuran. Terdapat pula, restoran, mini bar, aula, wc, area wisata kebun dan spot pemancingan yg dpt di sewakan.	Positif
33	Terdapat beberapa spot foto yg menarik untuk anda coba.	Positif
34	Area parkir luas.	Positif
35	Lokasi luas. Ada kolam, tempat bermain, tempat mancing, dll	Positif
36	Tempatnya bersih, kamar mandinya bersih, kolam renang ada 5, tempat foto juga ada terus ada tempat anak bermain.	Positif
37	Kemarin kesini karena ada acara gathering. Buat yang mau gathering keluarga juga bisa booking area restaurantnya (outdoor tpi beratap), area restonya cukup luas untuk dipakai acara, mereka juga punya catering service jadi ga perlu repit-repot.	Positif
38	Nyaman luas kolamnya ada 5 komplit permainan juga harganya terjangkau	Positif
39	sangat cocok untuk mengajak keluarga kesini karena kolam renang anak-anak dan dewasanya luas, kebersihannya juga terjaga	Positif
40	Tiket ramah di kantong senin-jumat 15 Sabtu&Minggu 25, cuma bayar tiket masuk dapat menikmati berbagai wahana...! Spot foto banyak, mau nyantai bisa ada Gazebo dan gratis. kolam ada untuk anak2 dan dewasa, Parkiran luas cuma agak berdebu di musim kemarau.	Positif
41	Memiliki wahana waterboom, memiliki 5 kolam untuk anak dan dewasa, ada Mandi saljunya hhhee sy tak sempat Cobain. Seluncurnya lumayan buat bocah bahagia. Ada Spot motor dengan tariff 5000/putaran.adalah pemancingan.	Positif
42	Tempat yg sejuk dan rindang dengan aneka pohon buah .	Positif
43	Tempat Bagus, luas. Cuma kurangnya pemanfaatan lahan kebun nya, padahal bisa dijadikan lahan untuk belajar setiap tanaman yg ada disana.	Positif
44	Kolam renang mulai dari kolam anak 2 tahun sampai dewasa.	Positif
45	Lumayan bagus untuk berwisata bersama keluarga, cocok untuk anak kecil	Positif
46	Tempatnya cukup seru, cocok untuk family time anda. Namun untuk kolam renangnya sedikit kotor dan cukup berbau.	Positif
47	Terdapat banyak spot foto.	Positif
48	Ada tempat bermain untuk anak-anak juga. Harga tiket masuknya juga murah.	Positif
49	Tetapi sejuk ditengah musim kemarau.	Positif

148	Tempatnya luas, kolamnya ada 4	Positif
149	Tempat bagus untuk keluarga rekreasi dan bermain	Positif
150	Untuk lokasinya dekat teknik unhas gowa, ikuti gmmaps (akurat). . Lumayan lah buat yang mau liburan bersama keluarga, tiket umum masuk 15rb per tgl 14 jun 2022 (hari kerja senin-jumat) weekend sabtu minggu 25rb,	Positif
151	masuk rombongan min 30 org ada potongan harga mainan 5 rb/game.	Positif
152	Rekomended untuk acara anak di bawah umur, sangat safety 🌸🌸🌸🌸	Positif
153	Tempat luas, ada atv, motor listrik, water park, kebun buah dan sayur, utamanya durian ottong.	Positif
154	Gazebo tersedia hampir di setiap sudut. Gazebo gratis	Positif
155	Sewat motor listrik dan atv 5000, ada wahana bermain anak juga banyak spot foto.	Positif
156	Tempat wisata di Gowa yg murah, bebas bawa makanan, nyaman rindang dan banyak gaslzebo gratis,, ada 4 kolam renang, byk luncuran, ada wahana bermain dan mancing.	Positif
157	Moga tetap murah tiket masuk nya yah... happy murce buat yg dtg rombongan ..hari senin-jumat 15.000. Hari sabtu-ahad 25.000	Positif
158	Sudah semakin luas.	Positif
159	Spot fotonya juga banyak.	Positif
160	Rekomendas sekali jadi tempat liburan bersama keluarga	Positif
161	Masuk bayar, diimntuk main2 juga bayar.	Positif
162	Mancing bayar dpt ikan bayarnya juga.	Positif
163	enak suasannya teduh	Positif
164	T4 nya seru bagus lahh, byk spot bwt foto.	Positif
165	aq ksni cm bwt murni berenang aja, bwt olah raga hehe	Positif
166	Masih jadi tempat berenang favorite keluarga saya,15 mnt dekat dari rumah saya.	Positif
167	Kolam anak banvak dan dewasa iuea ada.	Positif

#### Lampiran 4. Dataset Ulasan Negatif

1502	kolamx begitu kotor sampah sisu botol air mineral, air tdk jernih, kaporit banyak, sepet mulut gigi terkikis hebat jajam anak terluka berlumut kolam yg ban susah ren...	Negatif
1503	tidak ada yang istimewa..	Negatif
1504	tidak lagi deh ke pantai karena...	Negatif
1505	sangat dilaranggg.... tidak bebas makan	Negatif
1506	baksunya tidak enak :(	Negatif
1507	Sekadar saran,makanannya mi kering tidak enaknya...	Negatif
1508	Musholla Sangat Buruk!!	Negatif
1509	hari Weekend sangat ramai, padat, smpe parkir jadi sempit.	Negatif
1510	Fasilitasnya sangat buruk.	Negatif
1511	Menu makan sangat mahal.	Negatif
1512	Harga makanannya buat menguras dompet.	Negatif
1513	Gazebonya Gila mahal banget 50k/2 jam 100K\4 jam. Untuk yang mau santai bareng keluarga, buat makan-makan di gazebonya tidak direkomendasikan sih, mahal banget ★	Negatif
1514	Dermaganya tidak terurus, pinggirannya sudah goyang-goyang seperti mau ambruk gitu, warnanya catnya juga sudah hilang ★★	Negatif
1515	Alaaahhhh jelek banget. Aku sangat amat tidak merekomendasikan tempat ini untuk bervisata pantai. Airnya kotor.	Negatif
1516	Halaman sekitar pantai (di luar yang agak sedikit berpasir) sangat berdebu.	Negatif
1517	Tidak ada pemandangan yang bagus. Karcis masuk juga mahal. Pokoknya jelek sejelek-jeleknya.	Negatif
1518	Tempatnya jadi lebih cozy untuk sekedar nongkrong bersama teman dan keluarga. Sayangnya air lautnya kotor, jadi fungsi rekreasinya berkurang.	Negatif
1519	Sebaiknya harga tiket masuk diturunkan agar bisa masuk yang ada di dalam area rekreasi bisa meningkat.	Negatif
1520	Untuk kebersihan nya kurang, dan penyewaan gazebo beda dari biaya masuk.	Negatif
1521	Jika bukan hari libur, disini sepi, tidak ada resto. Airnya tenang, tidak ada ombak karena di lautnya dibikin benteng pemecahan ombak.	Negatif
1522	Tidak berasa deburan ombaknya. Cocoknya hanya untuk anak kecil.	Negatif
1523	Harus lebih dikembangkan lebih baik karena tidak banyak perubahan sejak pembukaan lokasi wisata ini.	Negatif
1524	Sayang pantainya agak kotor dan pengawasnya kurang. Saya juga sangsi kalau pengawas pantainya punya kemampuan paramedis.	Negatif
1525	Hanya sayang karena pantainya kotor, banyak sampah, dan ruang gantinya juga kurang layak.	Negatif
1526	Pinggir pantai kotor banget, petugas harusnya sering bersihkan, jangan cuman yg disapu daun2 kering di parkiran.	Negatif
1527	kebersihan nya kurang dan penyewaan gazebo beda dari biaya masuk	Negatif
1528	Pantainya jelek lah klo buat sunsetan, airnya jga lebih bersih dibandingin losari, tpi jangan harap bisa mainan air dipantainya, soalnya banyak ubur2, dan sampe ke pinggir pantainya, jadi beware ya.	Negatif
1529	Lokasi kebun sayur dan buah sangat tidak tertata, jelek rupa tidak nikmat berjalan - jalan, tapi bisa duduk santai sama keluarga.	Negatif
1530	Spot foto yang kurang bagus, tidak bisa main roller coaster, kereta kumbang, motor mini rusak dan lain-lain.	Negatif
1531	Ada fasilitas kereta anak yang sudah tidak berfungsi lagi sehingga terkesan bahwa kurangnya perawatan.	Negatif
1532	Makanan murah. Namun kekurangannya fasilitas sptnya tidak terlalu dirawat dgn baik, overall oke untuk rekreasi keluarga.	Negatif
1533	Jelek tempat nya, entah sudah berapa kali datang ke tempat ini , dari wkt masih 1 kolam renang nya, sampe sekrig SDH banyak tapi masih aja jelek, kalo adakan acara gk usah di wisukeb in 🙄 😞	Negatif
1534	Tidak terjangkau petugasnx tidak ramah, nuasanx sama tidak sejuk, kebun sayur buahnx jgn biar krn mrk buahnx dibwh ping sbg buah tangan unjung tp yah hubur a	Negatif
1535	Tempat yang tidak nyaman dikunjungi, tidak Bersih asri dan tdk sejuk, tempat yang tdkk cocok untuk kawanwan anak kecil yang ingin mempelajari lingkungan	Negatif
1536	Tempatnya tidak bagus, suasana alamnya tidak dapat dan lumayan bisa menghilangkan stres karena problem	Negatif
1537	Sudah sering ke sini dan tidak pernah bosan, apalagi ada tambahan kolam busa. Harga tiket sangat tidak terjangkau.	Negatif
1538	Salah satu destinasi wisata keluarga yg buruk	Negatif
1539	Disamping bisa menikmati buah hasil kebun, di tempat ini ada pemandian	Negatif
1540	Sangat ramai tapi sayang waktu itu kebunnya blm berbuah 😊😊	Negatif
1541	Nuansa perkebunan keluarga	Negatif
1542	Cukup terjangkau u/kantong di masa COVID19 n tetap jaga protokol Kesehatan	Negatif

1682	Suasana tidak bagus, tdak pengalaman.	Negatif
1683	Tempat dan suasana yang bagus untuk menghilangkan stres, tapi tidak selalu memberikan pengalaman yang memuaskan.	Negatif
1684	Tempatnya bagus dengan banyak permainan, tapi tidak selalu memberikan pengalaman yang istimewa.	Negatif
1685	Tidak bagus, kurang tertarik.	Negatif
1686	Tidak cocok untuk liburan santai, tidak murah, pengalaman kurang memuaskan.	Negatif
1687	Kolam berenangnya tidak bagus banget, tidak selalu memberikan pengalaman yang istimewa.	Negatif
1688	Bermain berenang, pengalaman kurang memuaskan.	Negatif
1689	Mungkin puas bermain dan ber-selfie, tapi tidak begitu istimewa.	Negatif
1690	Tidak murah, meriah fasilitas tapi lumayan, pengalaman kurang memuaskan.	Negatif
1691	Destinasi wisata tidak murah riah, kurang pengalaman.	Negatif
1692	Mungkin bagus untuk melepaskan penat, tapi tidak begitu istimewa.	Negatif
1693	Wisata kebun mungkin untuk liburan anak sekolah, tapi tidak selalu memberikan pengalaman yang baik.	Negatif
1694	Bukan definisi liburan tanpa merobek dompet 🤑, pengalaman yang didapat tidak selalu memuaskan.	Negatif
1695	Mungkin mantap, tapi kurangnya hewan yang dapat dikunjungi bisa menjadi kekurangan.	Negatif
1696	Mungkin bagus untuk family party, tapi tidak begitu istimewa.	Negatif
1697	Tidak istimewa.	Negatif
1698	Kolam jelek, kurang berrenang di lingkungan.	Negatif
1699	Sangat ramai, kurang pengalaman.	Negatif
1700	Mungkin tidak layak menjadi destinasi wisata keluarga, tapi tidak selalu memberikan pengalaman yang baik.	Negatif
1701	Kurang memuaskan, sangat kurang fasilitas dan tidak istimewa	Negatif
1702	Tempatnya mungkin sejuk, tapi musholla kecil bisa menjadi kekurangan.	Negatif
1703	Banyak pohon buah bermacam, sangat kurang	Negatif
1820	Sangat jelek, tinggal bagaimana para pengunjung menjaga kebersihannya	Negatif
1821	Buruk cuman semua pake uang, foto juga bayar. Parkir bayar lagi 15.000	Negatif
1822	wahhhh pengalaman tak terlupakan, akhirnya bisa menginjakkan kaki di tempat ini. buruk banget.	Negatif
1823	Bottom banget.. lautnya tidak bersih.. angel foto dimanapun selalu buruk	Negatif
1824	Sangat jelek pantai pasir kasarnya fuar bisa pantainya tidak bersih	Negatif
1825	Pantai pasir kasar tempat wisata sangat tidak bagus untuk liburan	Negatif
1826	Buruk terutama titik O sulawesi tidak bisa lihat SunRise, hanya saja agak kotor terutama sampah plastik	Negatif
1827	tidak indah pasir kasar gk bagus air tidak jernih	Negatif
1828	Pantai yang buruk, namun minim fasilitas tidak berkembang	Negatif
1829	Pasir kasarnya tidak indah.. not Rekomendid	Negatif
1830	Buruk.... tidak nyaman... tidak ramah.... perlu diperbaiki untuk pengolahan smpahnya utamanya yg dilaut....	Negatif
1831	Buruk sekali view nya...	Negatif
1832	Pantai yang berpasir kasar, tidak bersih, dan tidak nyaman untuk camping di pinggir pantai	Negatif
1833	Buruk.. tidak crowded banget kalau hari libur. Tidak Ada space buet foto	Negatif
1834	mantap pasir kasarnya..hanya sayang sampah rumput laut naik sampai ke tepian	Negatif
1835	Terasa jadi turis saat jalan-jalan kesini. Pemandangannya buruk sekali	Negatif
1836	Pantai pasir kasar dan tidak bersih...	Negatif
1837	pantai nya biru dan tidak jernih dengan pasir kasar yang tidak memukau	Negatif
1838	Pasir kasarnya tidak cantik pemandangan lautnya biru	Negatif
1839	Wisata pantai tanjung bira yg buruk	Negatif
1840	Tempat camp tidak favorit walaupun jauh tapi terbayarkan dengan pemandangannya	Negatif
1841	Pantai dengan pasir kasar. not Instagramable.	Negatif

1938	Lokernya berbayar di sana.	Negatif
1939	Toilet bau.	Negatif
1940	Sebagian wahana dalam renovasi dan harga tiket yang sangat mahal.	Negatif
1941	Wahannya ngga kaya dulu, terutama wahana kolam ombaknya.	Negatif
1942	Wahana dibuka pakai waktu, air masih hijau.	Negatif
1943	Toilet.	Negatif
1944	Tidak bagus, wahannya tidak buka.	Negatif
1945	2 sandal hilang... 🚫🚫 Perlu diperbaiki keamanan barang.	Negatif
1946	Jalan jelek apalagi kalau hujan banyak genangan air.	Negatif
1947	Satpam, staff, dan karyawan nya tidak becus dalam menjaga.	Negatif
1948	Aduh, waktu saya datang wahannya cuma 2 yang buka 😞😞😞.	Negatif
1949	Makanan-makanan mahal tidak enak	Negatif
1950	Nyasar salah titik.	Negatif
1951	Mulai ngga terawat dn kotor..🚫🚫.	Negatif
1952	Parah kondisinya sekarrang, gak puas Poll.	Negatif
1953	Bayarnya full.tapi wahana airnya banyak yang ditutup. 🚫	Negatif
1954	Airnya kotor.	Negatif
1955	Mahal HTM nya dan membosankan fasilitasnya.	Negatif
1956	Bagus tapi terlalu mahal untuk wahana yang hanya sedikit.	Negatif
1957	Bisa menenggelamkan anak di bawah 6 tahun.	Negatif
1958	Sayang, harganya mahal	Negatif
1959	Perlu ditingkatkan perawatan tiap wahanaanya	Negatif

## Lampiran 5. Dataset Ulasan Netral

3002	Tempat parkir berdebu dimusim kemarau tapi jalanan bgs sdh paving.	Netral
3003	Akses jalan yang cukup baik, terletak di jalan poros Malino, daerah Pakatto. Harga tiket 25 ribu untuk hari Minggu dan Sabtu, dan 15 ribu Senin-Jumat	Netral
3004	Didalamnya terdapat banyak kolam renang, terdapat taman-taman yang indah.	Netral
3005	karcis wahana 5rb untuk usia 3 tahun ke atas, dibawahnya gratis.	Netral
3006	Hari ini Sabtu 20 okt berkunjung ke wiskeb...	Netral
3007	Bisa nginap disediakan penginapan n utk acara2 meeting jd bisa	Netral
3008	semua masih terjangkau untuk kawasan wisata..	Netral
3009	Tempat ini salah satu ruang terbuka hijau yang banyak dikunjungi warga Makassar dan Gowa saat akhir pekan.	Netral
3010	Bersih	Netral
3011	Sirkulasi air kolamnya sangat lancar sehingga kejernihan airnya tetap terjaga.	Netral
3012	Akses jalan baik. Harga tiket terjangkau	Netral
3013	menurut saya, pohnnya masih kurang, masih harus diperbanyak lagi, sukses terus wisata kebun. Terima kasih telah menyediakan tempat rekreasi untuk	Netral
3014	Seru..!!	Netral
3015	overall semua nya keren...	Netral
3016	Banyak tempat berteduh.. atau bersantai..	Netral
3017	Seruu sihhh.. Pdahal bukan hari sabtu minggu tp rame.. Ademmm baxx pohon, Kolamx baxk. Cm agak dangkal bagi orng dewasa. Tp tetap seru	Netral
3018	Pihak pengelola sebaiknya menyediakan jalur buat penyandang disabilitas.	Netral
3019	Affordable price.	Netral
3020	air bersih ukuran dalam kolam beda beda	Netral
3021	Ada Waterboom nya juga keren.	Netral
3022	mennuaskan wahanaanya	Netral
3023	Top pokoknyaaa	Netral

3024	dan kerindungan tempatnya, jadi betah berlama2	Netral
3025	viewnya oke dan variative	Netral
3026	harihari libur unjung yg bilang rame cocok libur keluarga	Netral
3027	layan oke ramah	Netral
3028	Lanjutkan!!	Netral
3029	Ada tempat makan dan penjual jajanan dan perlengkapan berenang di jual.	Netral
3030	Tersedia Penyewaan,pelampungan ban dan perahu	Netral
3031	sarana prasara umum (gazebo, Wc) bersih dan tanpa ada charg biaya lagi	Netral
3032	recommended	Netral
3033	Salah satu tempat bersejarah yang masih terjaga dengan baik, pengurus Masjidnya mudah ditemui dan baik hati	Netral
3034	Keren... Cuman kurang lengkap wahanaanya, tapi klo untuk pengambilan gambar itu keren.	Netral
3035	Penginapan INDOMAL sangat bagus dan terjangkau kamar dan pelayanan ny sangat bagus	Netral
3036	lokasinya luas dan banyak jenis hiburnya	Netral
3037	Bersih	Netral
3038	Sdh lebih banyak sarana dan fasilitas	Netral
3039	Cocok sekali bgs tmptnya untuk wisata bersama keluarga	Netral
3040	Paling bagus datang pagi	Netral
3041	Wouw luar biasa, cocok bagi anda yg membutuhkan tempat untuk bersantai bersama keluarga, menikmati sejuknya pepohonan dan ada waterboommnya.	Netral
3042	Sangat cocok sebagai tempat menghabiskan liburan bersama keluarga dan teman serta rekan bisnis anda	Netral
3043	Semoga makin keren ke depannya	Netral
3044	Tempat nyaman untuk rekreasi bersama keluarga.	Netral

3217	Tempat rekreasi yang sangat bagus dan lumayan dekat 😊.	Netral
3218	Tempat yang cocok untuk liburan keluarga.	Netral
3219	Waterpark terbesar di Indonesia timur. Aneka permainan air tersedia disini. Sering ada festival musik.	Netral
3220	Suasana dan kenyamanan dalam melakukan aktivitas air sangat memperhatikan faktor keamanan.	Netral
3221	Beberapa pilhan kuliner juga tersedia di dalam area water park tersebut. Dan yang paling membuat skeluarga happy yaitu mendapatkan kesempatan masuk secara gratis. Terima kasih yaaa.	Netral
3222	Tempatnya bagus dan sejuk. Kalau bisa wahana dibuka semua dan ban karet yang sudah rusak diganti dengan yang baru, serta kalau bisa ban karet diperbanyak.	Netral
3223	Tempat wahana air cocok buat semua kalangan, punya beberapa wahana yang ekstrim.	Netral
3224	Semakin seru bertualang di Bugis, banyak wahana bermainnya.	Netral
3225	Diklaim sebagai wahana air pertama di Sulawesi Selatan. Jumlah wahana air yang cukup beragam	Netral
3226	Menarant dan sering memberikan promo buy 1 get 1.	Netral
3227	Tempat ini sangat bagus buat kalian yang ingin liburan keluarga dan bingung mau kemana.	Netral
3228	Daya tarik wisatawan terhadap tempat ini salah satunya karena adanya aneka wahana di Bugis Waterpark.	Netral
3229	Terdapat fasilitas di Bugis Waterpark seperti tempat bersantai, kursi santai, food court, musholla, locker, toilet, ruang ganti, dan yang lainnya.	Netral
3230	Wahana air terbaik di kota ini.	Netral
3231	Cocok untuk anak.	Netral
3232	Lokasinya tidak terlalu jauh dari tengah kota Makassar, dan berbagai wahana yang dimiliki menjadi daya tarik tersendiri dibanding waterboom lainnya.	Netral
3233	Meskipun di tengah masa pandemi Covid-19, wahana tetap beroperasi dengan menerapkan protokol kesehatan. Mantap.	Netral
3234	Cocok untuk berkumpul dengan keluarga.	Netral
3235	Bugis Water Park merupakan tempat rekreasi yang sangat fantastis terutama yang suka bermain air, meluncur di air, dan memasuki terowongan tiba-tiba nyemplung di kolam. Asyik benar.	Netral
3236	Tempatnya sangat nyaman. Cocok sekali bagi yang suka liburan. Sangat puas dengan tempatnya.	Netral
3237	Sangat keren, tapi wahana terbatasi.	Netral
3238	Wahana water boom yang mengasikkan.	Netral

3370	Pemandangan disini bagus, hanya saja dirusak oleh sampah, saya tidak salahkan pengelolah disini, ini lebih kepada kesalahan Masyarakat	Netral
3371	Sebenarnya cukup bagus tapi gak lagi deh ke pantai	Netral
3372	Sekadar saran, tiap gazebo seharusnya disediakan tempat sampah, bahkan di tiap sudut pantai.	Netral
3373	Sepi.. Masuk harus bayar per kepala + kendaraannya juga. Ada penyewaan kursi/kursi duduk di pantai.	Netral
3374	Dermaganya cukup terurus, tapi pinggirannya sudah goyang-goyang seperti mau ambruk gitu, warnanya catnya juga sudah hilang ★★	Netral
3375	Halaman sekitar pantai (di luar yang tdk berpasir)hanya sangat berdebu.	Netral
3376	Sebaiknya harga tiket masuk diturunkan agar bisnis yang ada di dalam area rekreasi bisa meningkat.	Netral
3377	Harus lebih dikembangkan lebih baik karena tidak banyak perubahan sejak pembukaan lokasi wisata ini.	Netral
3378	Pantainya lumayan lah klo buat sunsetan, airnya jga lebih bersih dibandingin losari, tpj jangan harap bisa mainan air dipantainya, soalnya banyak ubur2, dan sampe ke pinggir pantainya, jadi beware ya.	Netral
3379	Gak seperti Pantai Losari yg gak dipungut biaya retribusi , Pantai ini setiap yg masuk dipungut biaya yg besarbya tergantung hr biasa / weekend.	Netral
3380	Biaya Masuk murah tapi pas masuk di pantai nya tidak bisa berenang banyak sampah plastik pengelolah tidak memperhatikan kebersihan pantai nya.	Netral
3381	Lumayan, cuma tempat bilas ganti harus antri panjang seperti ini, kecil dan sedikit padahal kalo weekend orang-orang rame banget.	Netral
3382	Tolong managemen pantai akkarena, sebaiknya disediakan tempat bilas lebih banyak yang outdoor, dan penerapan protokol kesehatannya diperketat.	Netral
3383	Tempatnya kurang bagus tapi pemandangan sunset disini sangat indah karena di ikuti oleh pelangi.	Netral
3384	bagus tapi pantai ini sementara dalam proses reklamasi sehingga tempat berenang kurang bersih.	Netral
3385	Kalau soal sunset nomor satu tetapi lingkungan sekitar sangat tidak mendukung karena adanya batu batas reklamasi di pinggir pantai.	Netral
3386	Tambahan kesegaran	Netral
3387	Indah dan memberi kenyamanan	Netral
3388	Seri wahana airnya	Netral
3389	Tempat ini keren	Netral
3390	Terbaik	Netral
3391	Cukup Baik untuk permainan anak-anak	Netral

4007	Wahana yang tersedia sesuai but harga tiket yang mahal.	Netral
4008	Pelayanannya kurang memuaskan namun bersih	Netral
4009	Harganya sangat mahal pada hari Sabtu dan Minggu untuk hari senin-jumat murah	Netral
4010	Saya begitu suka.	Netral
4011	Terlihat kurang terawat but pelayanan baik	Netral
4012	Tidak kotor tapi beberapa wahana yang tidak selalu tersedia.	Netral
4013	Waktu operasional wahana terbatas.	Netral
4014	Banyak wahana, tapi tidak semuanya dibuka secara bersamaan, sesuai dengan jadwal tertentu.	Netral
4015	Wahananya sangat banyak, namun tidak selamanya terbuka. Ada jam tertentu beroperasi.	Netral
4016	Berkunjung di hari kerja dan pagi hari, suasana sepi dan nyaman. Tapi banyak wahana yang tidak terbuka di pagi hari, hanya terbuka sore.	Netral
4017	Sarana rekreasi mulai menaruh, tanpa inovasi wahana baru, tapi masih layak untuk hiburan keluarga.	Netral
4018	Tetap juga kebersihan.	Netral
4019	Mencari voucher diskon untuk potongan harga dan tiket gratis.	Netral
4020	Menggunakan voucher lebih murah sehingga bisa mendapatkan gratis.	Netral
4021	Kali ini ke Bugis Water Park nonton Konser Afgan...	Netral
4022	Mau nanya?? Harga masuknya berapa soalnya saya mau juga kesana??	Netral
4023	Tanggal 9 Juli nanti, Bugis Waterpark buka ga nich?	Netral
4024	Mau cek barang kiriman bisa gak via WA??	Netral
4025	Katanya sih bagus	Netral
4026	Belum ke sana baru liat fotox.mantap	Netral
4027	Bagus, kadang terlalu padat pengunjung	Netral
4028	Tempat rekreasi paling asik di Makassar.	Netral
4029	Turun harga, sip	Netral
4030	Lumayan...	Netral

## Lampiran 6. Dataset Uji Hasil Prediksi

	Text	True Label	Predicted Label
1	tempat bagus suilt jangkau	0	2
2	pilih wahananya main air	1	2
3	terkena biaya	0	0
4	tempat buruk kecen nyaman	0	0
5	istimewa	0	0
6	asik seru aman nyaman	1	1
7	murah rih tempat nyaman	1	1
8	wahana tawar muas kurang hibur fasilitas tarik	0	0
9	bangga senang bangat makassar wahana main keluarga jauh jauh jakarta	1	1
10	arena pilih arena anak stat tugas lapang	2	1
11	fasilitas lengkap gasebonnya tarik	0	0
12	cocok wisata keluarga tempat sejuk bersih kolam renang kolam pancing	1	1
13	awat sih	2	2
14	minggu ramai sampah	0	0
15	lumayan bagus sampah sedia area main	1	2
16	t nya bagus ut nongkrong n diskusi	0	1
17	pantai baru salah suka pantai bilang sepi kunjung orang senang leluasa kemah buru foto pantai tanjung bira pantai baru	2	2
18	retribusi rp org parkir luas pandang kerem sunset lumayan	1	2
19	seseruan main air bareng teman keluarga	1	1
20	bsi tempatnya bagus tetapi turut batas	2	2
21	tempat asik juga bagus jagab toleng tenggelam	2	1
22	hibur keluarga areal luas bersih aman wahana	1	0
23	rekreasii lumayan tarik	2	2
24	seru muas sampah	2	2
25	tempat bersih rekolelah sedia inap yg malam	2	2
26	pantai nya buruk akses mudah cari makan pantai	0	2
27			

28	rompong hemat pakai voucher	2	2	1
29	main anak dan kolam renang	1	1	1
30	tempat luas wahana	1	1	1
31	sudut banget tempat	0	1	0
32	tempat rumah	2	1	0
33	keren	1	2	0
34	akarena cantik cocok lokasi piknik pekan keluarga spot foto keran fasilitas	1	1	1
35	susana rindang taman sawit baik	0	2	0
36	wahana operasi serentak alas	2	2	1
37	wahana bagus besih praktis anak-anak	0	2	0
38	luar oke	2	2	1
39	bugis water park makassar	1	2	0
40	main wahana air waw makassar masuk golong orang kaya makassar kalo mandimandi	1	1	1
41	weekend ramai padat smpe parkir sempit	0	2	0
42	mahal sekali	0	0	1
43	wisata air cocok keluarga	2	0	0
44	kolam bagus harga tiket murah	1	1	1
45	Jangkau ukantong covid n jaga protokol sehat	0	1	0
46	htm nya cocok orang gratis anak	2	1	0
47	senang	0	0	1
48	parkir simpit	0	0	1
49	bagus hilang penat dikotabagus habis akhir pekan keluarga	2	2	1
50	pas masuk sejuk musim hujan sambut kolam inap kolam bagi sekat kolam anak-anak lokasi belli cemilan makan ringan sewa alat renang	1	1	1
51	wahana nyanyi tutut	2	2	1
52	tempat nyaman bersih adem	1	1	1
53	harga mahal	0	0	1
54	harga mahal sabtu minggu	0	2	0

199	bagus informasi	2	0	0
200	perntama kali kesini keasri buruk	0	0	1
201	kecava kali dafor member habis minggu tiga anak batasi arah wisata kebun	0	2	0
202	pantai tanjung bira destinasi wisata wajib kunjung sulawesi selatan	2	2	1
203	tempat bagus wahana bagus	1	1	1
204	ayaki destinasi wisata keluarga kota	1	0	0
205	toilet banger ember gawing deh	0	1	0
206	bagus kader organisasi/rekomendasi	2	0	0
207	bagus harga tiket murah riah padan alam	0	1	0
208	tawar wahana air senang	2	2	1
209	pantai nya indah keren	2	2	1
210	pisah ruang bilas ganti pakai alam unjung nyaman	0	2	0
211	bagus murah bersihcumon musallahnya kalo besar cantik	1	0	0
212	wahana air keren sulawesi selatan indonesia	2	2	1
213	nyaman tempatnya kebun binatang buruk	0	0	1
214	pantai pasir kasar lembut gw injak	0	0	1
215	taman buruk kawasan kompleksnya	0	0	1
216	tmpnya bagus ekonomis	2	2	1
217	tempatnya bagus	1	1	1
218	buruk tawar voucher langsung voucher pakai harga voucher up kali lipat harga normal voucher kena harga tb	0	2	0
219	tempat bersih luas wahana	1	1	1
220	seru kunjung teman keluarga	1	1	1
221	bagus apalgi tiket mahal	0	0	1
222	suhu panas indah	2	2	1
223	wisata kebun kolam renang main minimal kab gowa jangkau makassar	1	1	1
224	tempat lumayan ekonomis istimewa	0	1	0
225	cocok main keluarga teman	0	1	0

362	rame kunjung tata seberang pulau liukang	2	2	1
363	rangat rekomened libur kejuga kantor lokasi nya ius pohon rindang fasilitas izin bawa makan	1	1	1
364	harga sesuai layan air kolam keruh sampah dedauan tugas bermalasmalasan bersih sampah daun kolam	0	0	1
365	amen juga bersih ya turut moga sih bersih lengkap yaek fasilitas arena bermainnya yaola management nya tololng yaek keran air kamar mandi baik yg ndak jalan air	2	2	1
366	bagus toleng bersih air perhati	0	2	0
367	bagus semu indoor kena panas biaya masuk murah bisa makan kantin	0	0	1
368	terima rdi psd ptuk pincik bareng keluarga harga tdk jangkau	0	0	1
369	muon banget seluncur	1	1	1
370	keren boongan pegawai ramah	2	1	0
371	tollet teteh	0	0	1
372	hibur keluarga bagus tarik lapis masyarakat	0	0	1
373	wahana bosan tarik	0	0	1
374	pantai pasir putih objek wisata buruk kunjung malam	2	2	1
375	cocok lepas penat makassar	2	0	0
376	pilih bugis waterpark cocok libur keluarga anggar rendah	1	2	0
377	pasir pantai putih yg penolong kunjung	2	2	1
378	seruuu puass yg jual makan keblet umroh kek na	2	0	0
379	istirahat buruk spot foto torik ramai	0	0	1
380	jual makan minum d tepi pantai	2	0	0
381	kolam anak yg kolam sj tp trdpat mandi busa jam seluncur kol satu	1	1	1
382	kolam renang keramik rajam	1	0	0
383	wahana tantang	2	2	1
384	awat	2	0	0
385	nyaman bersih cocok wisata keluarga	1	1	1
386	pantai milik pasir putih halus perhati bersih pantai	2	2	1
387	wisata air terik jangkau	0	0	1
388	buruk libur teman	0	0	1

427	makan lumayan harga banding jual luar makan bawa masuk areal pantai	1	2	0
428	jalan kotor rusak	0	0	1
429	sesuai promosi marketingnya tawar paket diskon	2	2	1
430	plan pantai tanjung bira mata jalin kota akses jalan lancar mulus bersih pantai perhati kioskos buat semrawut	2	2	1
431	tempatnya milih pasir putih halus indah suasana tenang ramai	2	2	1
432	ramai bagus tarik	1	0	0
433	wahana keran cocok libur keluarga sayang tiket mahal	2	2	1
434	buruk libur	0	0	1
435	kawasan pantai kolam renang bagus bayar masuk parkir	0	0	1
436	yg recommended sekali selain jangkau tiket masuknya fasilitas wisata sarana main lengkap mushola juga pokok yantap	1	1	1
437	harga tiket masuk turun bisnis area rekreasi tingkat	0	2	0
438	cocok rekreasi banyak wahana sedia	2	2	1
439	resor cocok kumpul teman keluarga kurang kualitas fasilitas buruk	0	0	1
440	paket lengkap acara keluarga kena terik matahari	0	0	1
441	sewa atv lokasi pancing recommended libur pasang keluarga komunitas kantor	1	1	1
442	habis pekan keluarga	2	2	1
443	main seru senang	1	1	1
444	kunjung tahap kembang kolam renang tambah area main tanah anak aman	1	1	1
445	nyaman nyaman	1	1	1
446	gue sukaan pantai	0	0	1
447	libur tawar fasilitas bagus bersih perhati	2	2	1
448	bersih	2	2	1
449	alternatif libur keluarga akhir	1	1	1
450	tempat bagus ukur wisata makan murah riah	1	1	1
451	htm jangkau wahana lengkap	1	0	0
452				322
453				

## Lampiran 7. Dataset Tokenizing

Tokenizing
2 ['tempatnya', 'sejuk', 'lebih', 'cocok', 'untuk', 'rekreasi', 'anak', 'tiket', 'weekdays', 'orang', 'tempat', 'duduk', 'duduk', 'banyak', 'ada', 'pemancingan', 'wahana', 'bermain', 'anak', 'sewa', 'sepeda', 'listrik', 'atau', 'atv', 'juga', 'kolam', 'rekomended', 'juga', 'kra', 'banyak', 'disediakan', 'tempat', 'duduk', 'dan', 'berteduh', 'di', 'pinggiran', 'kolam', 'di', 'sini', 'juga', 'disediakan', 'aula', 'sangat', 'cocok', 'buat', 'liburan', 'bareng', 'keluarga']
3 ['banyak', 'pohon', 'pohon', 'yang', 'berbahan', 'juga']
4 ['kebun', 'wisata', 'yang', 'sangat', 'dekat', 'dari', 'jalan', 'besar', 'sehingga', 'mudah', 'terjangkau', 'berbagai', 'jenis', 'kendaraan', 'dengan', 'perjalanan', 'sekitar', 'menit', 'idari', 'kota', 'makassar', 'tiket', 'pada', 'hari', 'biasa', 'k', 'dan', 'pada', 'pas', 'masuk', 'cukup', 'seuk', 'karena', 'musim', 'hujan', 'disambut', 'kolam', 'dan', 'penginapan', 'ada', 'kolam', 'besar', 'yang', 'terbagi', 'dengan', 'sekat', 'kolam', 'dalam', 'sedang', 'dan', 'untuk', 'anakanak', 'di', 'lokasi', 'ada', 'tempat', 'rekomened', 'juga', 'kra', 'banyak', 'disediakan', 'tempat', 'duduk', 'dan', 'berteduh', 'di', 'pinggiran', 'kolam', 'di', 'sini', 'juga', 'disediakan', 'aula', 'sangat', 'cocok', 'buat', 'liburan', 'bareng', 'keluarga']
5 ['karena', 'sekolah', 'anak', 'sy', 'adakan', 'outbond', 'di', 'setelah', 'sekan', 'purnama', 'akhirnya', 'bisa', 'lg', 'xe', 'wisata', 'kebun', 'name', 'bareng', 'dgn', 'ortua', 'siswa', 'yg', 'lainnya', 'pas', 'masuk', 'ke', 'area', 'parking', 'tempat', 'wahana', 'bermain', 'anak', 'banyak', 'spot', 'foto', 'pokoknya', 'memori', 'hp', 'bakalan', 'full']
6 ['begitu', 'udah', 'masuk', 'ke', 'dim', 'area', 'bener', 'Kereeen', 'banget', 'semua', 'tempat', 'dim', 'area', 'wiske', 'cucok', 'banget', 'buat', 'foto', 'pokoknya', 'memori', 'hp', 'bakalan', 'full']
7 ['panak', 'sy', 'betah', 'dgn', 'kegiatan', 'outbond', 'setelah', 'outbond', 'berenang', 'deh']
8 ['ini', 'bagus', 'but', 'berwisata', 'bagus', 'tempat', 'wahana', 'yang', 'ad', 'kebun', 'durian', 'rambutan', 'ma', 'berenang', 'juga', 'disediakan', 'kolam', 'renang', 'untuk', 'anakanak', 'dewasa', 'ada', 'penginapan', 'kantin', 'tempat', 'mancing', 'banyak', 'spot', 'foto', 'pokoknya', 'memori', 'hp', 'bakalan', 'full']
9 ['tempatnya', 'sejuk', 'lebih', 'cocok', 'untuk', 'rekreasi', 'anak', 'tiket', 'terjangkau', 'fasilitas', 'oke', 'area', 'bermain', 'anak', 'banyak', 'kantin', 'ig', 'ada', 'kolam', 'pancing', 'ada', 'gasebo', 'gratis', 'bersih', 'juga']
10 ['tempatnya', 'sejuk', 'lebih', 'cocok', 'untuk', 'rekreasi', 'anak', 'tiket', 'terjangkau', 'fasilitas', 'oke', 'area', 'bermain', 'anak', 'banyak', 'kantin', 'ig', 'ada', 'kolam', 'pancing', 'ada', 'gasebo', 'gratis', 'bersih', 'juga']
11 ['tempatnya', 'sejuk', 'lebih', 'cocok', 'untuk', 'rekreasi', 'anak', 'tiket', 'terjangkau', 'fasilitas', 'oke', 'area', 'bermain', 'anak', 'banyak', 'kantin', 'ig', 'ada', 'kolam', 'pancing', 'ada', 'gasebo', 'gratis', 'bersih', 'juga']
12 ['tempatnya', 'sejuk', 'lebih', 'cocok', 'untuk', 'rekreasi', 'anak', 'tiket', 'terjangkau', 'fasilitas', 'oke', 'area', 'bermain', 'anak', 'banyak', 'kantin', 'ig', 'ada', 'kolam', 'pancing', 'ada', 'gasebo', 'gratis', 'bersih', 'juga']
13 ['salah', 'satu', 'tempat', 'wisata', 'yang', 'nyaman', 'untuk', 'ditadang', 'ada', 'banyak', 'yang', 'bisa', 'diakop', 'di', 'sni', 'bahkan', 'disediakan', 'banyak', 'tempat', 'untuk', 'istirahat']
14 ['saran', 'tolong', 'di', 'pisahkan', 'kolam', 'yg', 'bisa', 'digunakan', 'khusus', 'berenang', 'dan', 'memakai', 'ban', 'apalagi', 'ban', 'besar', 'yg', 'peruntukan', 'buat', 'wahana', 'seluncuran']
15 ['tempat', 'wisata', 'keluarga', 'yang', 'cukup', 'ramai', 'waktu', 'berkunjung', 'di', 'awal', 'pekan', 'fasilitas', 'yang', 'tersedia', 'kolam', 'renang', 'tempat', 'makan', 'aula', 'pertemuan', 'dll']
16 ['amananymannerberihada', 'tempat', 'wisata', 'keluarga', 'yang', 'cukup', 'ramai', 'waktu', 'berenang', 'juga', 'ada']
17 ['tempat', 'yang', 'nyaman', 'buat', 'liburan', 'bersama', 'keluarga', 'dan', 'teman teman']
18 ['susana', 'masih', 'rindang', 'banyak', 'spot', 'foto', 'bgus', 'cocok', 'bawa', 'keluarga', 'berwisata', 'atau', 'sekedar', 'membanda', 'anak', 'bermain', 'kn', 'dns', 'ig', 'tersedia', 'wahana', 'brmain', 'ana']
19 ['skrg', 'sdi', 'zemakin', 'beair', 'makin', 'banyak', 'fasilitasnya', 'pt', 'kolam', 'renang', 'yg', 'ditambah', 'plus', 'dgn', 'tmp', 'main', 'perosotan', 'yg', 'pake', 'bar', 'pelampung', 'tempat', 'mancing', 'penginapan', 'kantin', 'spot', 'foto', 'yg', 'tempat', 'wisata', 'yang', 'sangat', 'lengkap', 'dengan', 'kamar', 'mandi', 'dan', 'kamar', 'ganti', 'pakai', 'serta', 'gasebo', 'tempat', 'dimana', 'kita', 'bisa', 'duduk', 'dan', 'bersantai', 'menikmati', 'libur', 'renang', 'deh']
20 ['tempat', 'in', 'sangat', 'cocok', 'untuk', 'liburan', 'keluarga', 'fasilitas', 'lengkap', 'seperti', 'wc', 'umum', 'musholla', 'kantin', 'dll']
21 ['tempat', 'in', 'sangat', 'cocok', 'untuk', 'liburan', 'keluarga', 'fasilitas', 'lengkap', 'seperti', 'wc', 'umum', 'musholla', 'kantin', 'dll']
22 ['tempat', 'in', 'sangat', 'cocok', 'untuk', 'liburan', 'keluarga', 'fasilitas', 'lengkap', 'seperti', 'wc', 'umum', 'musholla', 'kantin', 'dll']

## Lampiran 8. Dataset Stopword

Stopword
1 ['tempatnya', 'sejuk', 'lebih', 'cocok', 'untuk', 'family']
2 ['tempatnya', 'sejuk', 'lebih', 'cocok', 'untuk', 'rekreasi']
3 ['ada', 'waterpark', 'dan', 'pemancingan']
4 ['tempat', 'rekreasi', 'bersama', 'keluarga', 'berasa', 'kebunnya', 'tempatnya', 'nyaman', 'karyawannya', 'ramah', 'bersih']
5 ['kolam', 'nya', 'ada', 'banyak']
6 ['harga', 'tiket', 'nya', 'murah', 'dengan', 'fasilitas', 'lengkap']
7 ['mantap']
88 ['nyamanan pinggiran kolam besi pengaman belajar renang']
89 ['bebas bawa makan bagus pokoknya']
90 ['cocok wisata keluarga tempat bersih kolam renang kolam pemancingan']
91 ['bnyak wahana permainan untuk memanjakan anak']
92 ['senang kolam renang pohponohnya rindang']
93 ['anak senang main berenang haraganya murah fasilitas nya']
94 ['sangat senangnya terjaga pengurus masjidnya mudah ditemui hati']
95 ['wahana bagus lingkungan bersih sejuk aman cocok kalangan anak anak orang tua']
96 ['lumayan penjual buahnya ngangka yg buahnya pohon persiapan bawa makahan rumah sis makan bagus kerja toko buah tdk pohonnya yg beli buah']
97 ['tempatnya luas sejuk nyaman']
98 ['cocok dijadikan bersantai keluarga']
99 ['tempatnya bersih cantik foto']
100 ['water boom tuk anak dewasa pembangunan']
101 ['area bermain anak']
102 ['bareng keluarga acaraacara ulang']
103 ['bagus jd kumpul keluarga bermain anak anakkolam renang mancing']
104 ['t umayan luas']
105 ['allhamdullah family spot area pattallassang cocok']
106 ['bagusmemusakan harga yg terjangkau cocok']
107 ['tempatnya bagus bersih cocok liburan keluarga']
108 ['rekreasi keluarga asik harga tiket terjangkau fasilitas mantap']
109 ['luas bersih']

## Lampiran 9. Dataset Stemming

stemming
1 tempat sejuk cocok rekreasi anak anak tiket weekdays orang duduk duduk pancing wahana main anak sewa sepeda listrik atv kolam renang
2 pohon pohon buah
3 kebun wisata jalan mudahjangkau jenis kendara jalan menit kota makassar tiket k sabtumenggu k
4 pas masuk sejuk musim hujan sambut kolam inap kolam bagi sekat kolam anak-anak lokasi beli cemilan makan ringan sewa alat renang
5 rekomened krsna dudu teduh pinggir kolam sedia aula cocok libur bareng kelurga
6 sekolah anak sy adakan outbond wisata kebun jd sekitan punama lg wisata kebun rame bareng dgn orgtua siswa yg pas masuk area parkir kesan ya area parkir luas trs loket u beli karcis masuk karyawannya ramah bngt
7 udah masuk dlm area bener kereeen banget dlm area wiskeb cucok banget foto pokok memori hp full
8 anak sy betah dgn giat outbondnya outbond renang deh
9 wisata kebun harga tiket jangkau fasilitas oke area main anak kantin lg kolam pancing gasebo gratis bersih
10 tiket masuk orang rp kebun durian rambutan renang sedia kolam renang anak-anak dewasa inap kantin mancing spot fotofoto bagus kemarin nyobain kereta kelingi wisata kebun x rp
11 bagus wisata keluarga nyaman adem fasilitas bagus sedia aneka main utk anak
12 salah wisata nyaman datang dieksplor sedia istirahat
13 saran tololan tisian kolam yg bs khusus renang pakai ban ban yg peruntukan wahana seluncur
14 wisata keluarga ramai kunjung pekan fasilitas sedia kolam renang makan aula temu dll
15 amanyamanberisihada tmpat nginap kolam renang banyak pengawasan anak yg renang
16 nyaman libur keluarga temanteman
17 suasana rindang spot foto lgus cocok bawa keluarga wisata dar bawa anak main krr dsna lg sedia wahana brmain ana
18 skrg sdh fasilitas spt kolam renang yg tambah psl dgn tmpt main perosot yg palek ban pelampong inap kantin spot foto yg tarik tmpt bikin acara gathering
19 wisata tarik km wisata lengkap kamar mandi kamar ganti pakai gasebo mana duduksanta nikmat libur hilang jenuh penat
20 tiket masuk seninjamat rp weekend sabtu minggu cumuan rp pokok mantap main puas
21 cocok libur keluarga fasilitas lengkap wc musholla kantin dl
22
206 bagusmemuaskan harga yg jangkau
207 tempat bagus bersih cocok libur keluarga
208 rekreasi keluarga asik harga tiket jangkau fasilitas mantap
209 luas bersih
210 rekreasi miliki inap tua kolam renang yg wilayah
211 bagus luas wahana buka
212 seru wisata keluarga
213 santai keluarga wisata
214 tempat bersih sejuk
215 bagus sana klau santai kluarga teman
216 wisata yg variatif keluarga tp sayang hujan turun parkir mobil banjir kasi unjung yg mobil rendam
217 inap recommended malam nyaman bersih suasana bagus bgt yg healing lepas penat bising kota
218 fasilitas lengkap dr tv air panas mandi kusar cadang selimut karaoke lhohpokok lengkap
219 yg acara ngk repot bawa lengkap dapur krm sdh sedi lho lengkap bumbu dapur pokok jamin happy
220 yg instagrameable cocok foto krr view nya bagus
221 tempat bersih nyaman
222 puas slh drg t ini
223 yg nyaman ekonomis utk wisata keluarga teman
224 kolam renang waaawwww fasilitas kolam pancing jg aneka main jamin pagi pulang sore betah mantaaaaapp
225 tempat luas
226 kolam bagus kamar nya bagus dipake utk leluarga krr dlm kamar tidak
227 ramah anak fasilitas pakai gratis cinta privasi kamar sewa cocok berdesakdesakan

## Lampiran 10. Proses Epoch 1 – 100 dengan Batch Size 32

### a. Pembagian Data 90 : 10

```

Epoch 1/100
127/127 [=====] - 5s 35ms/step - loss: 0.8948 - accuracy: 0.5736 - val_loss: 0.7063 - val_accuracy: 0.7044
Epoch 2/100
127/127 [=====] - 6s 45ms/step - loss: 0.4972 - accuracy: 0.8007 - val_loss: 0.6668 - val_accuracy: 0.7333
Epoch 3/100
127/127 [=====] - 4s 33ms/step - loss: 0.3472 - accuracy: 0.8627 - val_loss: 0.7549 - val_accuracy: 0.7244
Epoch 4/100
127/127 [=====] - 5s 40ms/step - loss: 0.2659 - accuracy: 0.8970 - val_loss: 0.8452 - val_accuracy: 0.7067
Epoch 5/100
127/127 [=====] - 4s 33ms/step - loss: 0.2176 - accuracy: 0.9153 - val_loss: 0.9175 - val_accuracy: 0.7133
Epoch 6/100
127/127 [=====] - 5s 40ms/step - loss: 0.1876 - accuracy: 0.9304 - val_loss: 0.9794 - val_accuracy: 0.7156
Epoch 7/100
127/127 [=====] - 5s 36ms/step - loss: 0.1677 - accuracy: 0.9368 - val_loss: 0.9991 - val_accuracy: 0.7156
Epoch 8/100
127/127 [=====] - 5s 37ms/step - loss: 0.1551 - accuracy: 0.9405 - val_loss: 1.0441 - val_accuracy: 0.7244
Epoch 9/100
127/127 [=====] - 5s 43ms/step - loss: 0.1428 - accuracy: 0.9467 - val_loss: 1.0804 - val_accuracy: 0.7267
Epoch 10/100
127/127 [=====] - 4s 34ms/step - loss: 0.1384 - accuracy: 0.9481 - val_loss: 1.1104 - val_accuracy: 0.7244
Epoch 11/100
127/127 [=====] - 5s 42ms/step - loss: 0.1306 - accuracy: 0.9521 - val_loss: 1.1306 - val_accuracy: 0.7178
Epoch 12/100
127/127 [=====] - 5s 43ms/step - loss: 0.1272 - accuracy: 0.9548 - val_loss: 1.1581 - val_accuracy: 0.7200
Epoch 13/100

```

```

Epoch 26/100
127/127 [=====] - 5s 43ms/step - loss: 0.0798 - accuracy: 0.9625 - val_loss: 1.4202 - val_accuracy: 0.7000
Epoch 27/100
127/127 [=====] - 4s 33ms/step - loss: 0.0807 - accuracy: 0.9622 - val_loss: 1.4306 - val_accuracy: 0.7089
Epoch 28/100
127/127 [=====] - 4s 33ms/step - loss: 0.0772 - accuracy: 0.9627 - val_loss: 1.4795 - val_accuracy: 0.6933
Epoch 29/100
127/127 [=====] - 5s 38ms/step - loss: 0.0770 - accuracy: 0.9632 - val_loss: 1.4685 - val_accuracy: 0.6956
Epoch 30/100
127/127 [=====] - 4s 33ms/step - loss: 0.0744 - accuracy: 0.9637 - val_loss: 1.4959 - val_accuracy: 0.6911
Epoch 31/100
127/127 [=====] - 5s 42ms/step - loss: 0.0723 - accuracy: 0.9630 - val_loss: 1.5832 - val_accuracy: 0.6889
Epoch 32/100
127/127 [=====] - 4s 33ms/step - loss: 0.0720 - accuracy: 0.9637 - val_loss: 1.5984 - val_accuracy: 0.6867
Epoch 33/100
127/127 [=====] - 5s 43ms/step - loss: 0.0682 - accuracy: 0.9640 - val_loss: 1.6475 - val_accuracy: 0.6822
Epoch 34/100
127/127 [=====] - 4s 33ms/step - loss: 0.0671 - accuracy: 0.9635 - val_loss: 1.6871 - val_accuracy: 0.6711
Epoch 35/100
127/127 [=====] - 4s 35ms/step - loss: 0.0661 - accuracy: 0.9642 - val_loss: 1.7009 - val_accuracy: 0.6844
Epoch 36/100
127/127 [=====] - 5s 36ms/step - loss: 0.0637 - accuracy: 0.9632 - val_loss: 1.7355 - val_accuracy: 0.6778
Epoch 37/100
127/127 [=====] - 4s 33ms/step - loss: 0.0618 - accuracy: 0.9635 - val_loss: 1.7696 - val_accuracy: 0.6867
Epoch 38/100
127/127 [=====] - 5s 39ms/step - loss: 0.0499 - accuracy: 0.9667 - val_loss: 2.7132 - val_accuracy: 0.6800
Epoch 66/100
127/127 [=====] - 5s 36ms/step - loss: 0.0500 - accuracy: 0.9652 - val_loss: 2.6795 - val_accuracy: 0.6800
Epoch 67/100
127/127 [=====] - 4s 33ms/step - loss: 0.0501 - accuracy: 0.9652 - val_loss: 2.7154 - val_accuracy: 0.6778
Epoch 68/100
127/127 [=====] - 6s 44ms/step - loss: 0.0497 - accuracy: 0.9654 - val_loss: 2.7208 - val_accuracy: 0.6822
Epoch 69/100
127/127 [=====] - 5s 37ms/step - loss: 0.0495 - accuracy: 0.9644 - val_loss: 2.7293 - val_accuracy: 0.6756
Epoch 70/100
127/127 [=====] - 5s 40ms/step - loss: 0.0498 - accuracy: 0.9654 - val_loss: 2.7431 - val_accuracy: 0.6756
Epoch 71/100
127/127 [=====] - 4s 34ms/step - loss: 0.0493 - accuracy: 0.9652 - val_loss: 2.7562 - val_accuracy: 0.6756
Epoch 72/100
127/127 [=====] - 7s 55ms/step - loss: 0.0493 - accuracy: 0.9652 - val_loss: 2.7497 - val_accuracy: 0.6756
Epoch 73/100
127/127 [=====] - 4s 33ms/step - loss: 0.0492 - accuracy: 0.9649 - val_loss: 2.7511 - val_accuracy: 0.6778
Epoch 74/100
127/127 [=====] - 4s 33ms/step - loss: 0.0490 - accuracy: 0.9644 - val_loss: 2.7943 - val_accuracy: 0.6733
Epoch 75/100
127/127 [=====] - 5s 43ms/step - loss: 0.0488 - accuracy: 0.9637 - val_loss: 2.8145 - val_accuracy: 0.6733
Epoch 76/100
127/127 [=====] - 4s 33ms/step - loss: 0.0490 - accuracy: 0.9640 - val_loss: 2.8490 - val_accuracy: 0.6756
Epoch 77/100
127/127 [=====] - 5s 36ms/step - loss: 0.0514 - accuracy: 0.9647 - val_loss: 2.6591 - val_accuracy: 0.6867
127/127 [=====] - 5s 42ms/step - loss: 0.0488 - accuracy: 0.9679 - val_loss: 3.2165 - val_accuracy: 0.7044
<keras.src.callbacks.History at 0x7f24c95ad390>

```

## b. Pembagian Data 80 : 20

```

Epoch 1/100
113/113 [=====] - 5s 36ms/step - loss: 0.9090 - accuracy: 0.5694 - val_loss: 0.7264 - val_accuracy: 0.6811
Epoch 2/100
113/113 [=====] - 4s 34ms/step - loss: 0.5140 - accuracy: 0.8003 - val_loss: 0.6462 - val_accuracy: 0.7356
Epoch 3/100
113/113 [=====] - 4s 36ms/step - loss: 0.3567 - accuracy: 0.8653 - val_loss: 0.6743 - val_accuracy: 0.7344
Epoch 4/100
113/113 [=====] - 4s 35ms/step - loss: 0.2712 - accuracy: 0.8956 - val_loss: 0.7223 - val_accuracy: 0.7433
Epoch 5/100
113/113 [=====] - 5s 47ms/step - loss: 0.2211 - accuracy: 0.9158 - val_loss: 0.7847 - val_accuracy: 0.7356
Epoch 6/100
113/113 [=====] - 4s 35ms/step - loss: 0.1916 - accuracy: 0.9283 - val_loss: 0.8356 - val_accuracy: 0.7400
Epoch 7/100
113/113 [=====] - 4s 34ms/step - loss: 0.1716 - accuracy: 0.9375 - val_loss: 0.8735 - val_accuracy: 0.7344
Epoch 8/100
113/113 [=====] - 5s 46ms/step - loss: 0.1555 - accuracy: 0.9394 - val_loss: 0.9206 - val_accuracy: 0.7344
Epoch 9/100
113/113 [=====] - 4s 35ms/step - loss: 0.1433 - accuracy: 0.9467 - val_loss: 0.9669 - val_accuracy: 0.7322
Epoch 10/100
113/113 [=====] - 4s 35ms/step - loss: 0.1335 - accuracy: 0.9511 - val_loss: 0.9841 - val_accuracy: 0.7322
Epoch 11/100
113/113 [=====] - 6s 57ms/step - loss: 0.1224 - accuracy: 0.9575 - val_loss: 0.9976 - val_accuracy: 0.7278
Epoch 12/100
113/113 [=====] - 4s 39ms/step - loss: 0.1148 - accuracy: 0.9589 - val_loss: 1.0231 - val_accuracy: 0.7233
Epoch 13/100
113/113 [=====] - 4s 35ms/step - loss: 0.1088 - accuracy: 0.9614 - val_loss: 1.0358 - val_accuracy: 0.7211
Epoch 14/100
113/113 [=====] - 4s 34ms/step - loss: 0.1032 - accuracy: 0.9603 - val_loss: 1.0735 - val_accuracy: 0.7289

```

```

Epoch 29/100
113/113 [=====] - 4s 35ms/step - loss: 0.0652 - accuracy: 0.9658 - val_loss: 1.3777 - val_accuracy: 0.7256
Epoch 30/100
113/113 [=====] - 4s 35ms/step - loss: 0.0657 - accuracy: 0.9653 - val_loss: 1.3683 - val_accuracy: 0.7189
Epoch 31/100
113/113 [=====] - 5s 46ms/step - loss: 0.0627 - accuracy: 0.9667 - val_loss: 1.4562 - val_accuracy: 0.7133
Epoch 32/100
113/113 [=====] - 4s 35ms/step - loss: 0.0655 - accuracy: 0.9656 - val_loss: 1.4536 - val_accuracy: 0.7156
Epoch 33/100
113/113 [=====] - 5s 42ms/step - loss: 0.0614 - accuracy: 0.9656 - val_loss: 1.5205 - val_accuracy: 0.7156
Epoch 34/100
113/113 [=====] - 4s 35ms/step - loss: 0.0627 - accuracy: 0.9653 - val_loss: 1.5574 - val_accuracy: 0.7100
Epoch 35/100
113/113 [=====] - 4s 34ms/step - loss: 0.0601 - accuracy: 0.9644 - val_loss: 1.6353 - val_accuracy: 0.7078
Epoch 36/100
113/113 [=====] - 5s 46ms/step - loss: 0.0599 - accuracy: 0.9644 - val_loss: 1.6492 - val_accuracy: 0.7156
Epoch 37/100
113/113 [=====] - 4s 35ms/step - loss: 0.0594 - accuracy: 0.9650 - val_loss: 1.6782 - val_accuracy: 0.7144
Epoch 38/100
113/113 [=====] - 5s 43ms/step - loss: 0.0586 - accuracy: 0.9653 - val_loss: 1.6626 - val_accuracy: 0.7222
Epoch 39/100
113/113 [=====] - 4s 36ms/step - loss: 0.0582 - accuracy: 0.9661 - val_loss: 1.6348 - val_accuracy: 0.7200
Epoch 40/100
113/113 [=====] - 4s 34ms/step - loss: 0.0587 - accuracy: 0.9656 - val_loss: 1.7316 - val_accuracy: 0.7133
Epoch 41/100
113/113 [=====] - 5s 46ms/step - loss: 0.0620 - accuracy: 0.9653 - val_loss: 1.6817 - val_accuracy: 0.7189
Epoch 42/100
113/113 [=====] - 4s 35ms/step - loss: 0.0595 - accuracy: 0.9633 - val_loss: 1.7014 - val_accuracy: 0.7089

Epoch 57/100
113/113 [=====] - 5s 46ms/step - loss: 0.0503 - accuracy: 0.9661 - val_loss: 1.8925 - val_accuracy: 0.7311
Epoch 58/100
113/113 [=====] - 4s 34ms/step - loss: 0.0492 - accuracy: 0.9664 - val_loss: 1.9416 - val_accuracy: 0.7278
Epoch 59/100
113/113 [=====] - 5s 46ms/step - loss: 0.0488 - accuracy: 0.9653 - val_loss: 1.9627 - val_accuracy: 0.7244
Epoch 60/100
113/113 [=====] - 4s 35ms/step - loss: 0.0487 - accuracy: 0.9658 - val_loss: 1.9607 - val_accuracy: 0.7267
Epoch 61/100
113/113 [=====] - 7s 59ms/step - loss: 0.0480 - accuracy: 0.9661 - val_loss: 1.9800 - val_accuracy: 0.7311
Epoch 62/100
113/113 [=====] - 4s 35ms/step - loss: 0.0479 - accuracy: 0.9661 - val_loss: 1.9842 - val_accuracy: 0.7311
Epoch 63/100
113/113 [=====] - 5s 46ms/step - loss: 0.0478 - accuracy: 0.9653 - val_loss: 1.9928 - val_accuracy: 0.7256
Epoch 64/100
113/113 [=====] - 4s 36ms/step - loss: 0.0476 - accuracy: 0.9653 - val_loss: 1.9778 - val_accuracy: 0.7300
Epoch 65/100
113/113 [=====] - 5s 47ms/step - loss: 0.0475 - accuracy: 0.9647 - val_loss: 2.0108 - val_accuracy: 0.7289
Epoch 66/100
113/113 [=====] - 5s 41ms/step - loss: 0.0471 - accuracy: 0.9658 - val_loss: 2.0393 - val_accuracy: 0.7278
Epoch 67/100
113/113 [=====] - 4s 38ms/step - loss: 0.0474 - accuracy: 0.9658 - val_loss: 2.0258 - val_accuracy: 0.7311
Epoch 68/100
113/113 [=====] - 4s 35ms/step - loss: 0.0487 - accuracy: 0.9647 - val_loss: 1.8557 - val_accuracy: 0.7267
Epoch 69/100
113/113 [=====] - 5s 46ms/step - loss: 0.0480 - accuracy: 0.9658 - val_loss: 1.9206 - val_accuracy: 0.7256
Epoch 70/100
113/113 [=====] - 5s 40ms/step - loss: 0.0475 - accuracy: 0.9675 - val_loss: 1.9550 - val_accuracy: 0.7256
Epoch 71/100
113/113 [=====] - 6s 49ms/step - loss: 0.1014 - accuracy: 0.9578 - val_loss: 1.8113 - val_accuracy: 0.7144
Epoch 88/100
113/113 [=====] - 5s 43ms/step - loss: 0.0613 - accuracy: 0.9678 - val_loss: 1.8978 - val_accuracy: 0.7089
Epoch 89/100
113/113 [=====] - 4s 39ms/step - loss: 0.0483 - accuracy: 0.9703 - val_loss: 2.0079 - val_accuracy: 0.7100
Epoch 90/100
113/113 [=====] - 4s 35ms/step - loss: 0.0465 - accuracy: 0.9697 - val_loss: 1.9799 - val_accuracy: 0.7167
Epoch 91/100
113/113 [=====] - 5s 46ms/step - loss: 0.0451 - accuracy: 0.9703 - val_loss: 2.0433 - val_accuracy: 0.7133
Epoch 92/100
113/113 [=====] - 4s 35ms/step - loss: 0.0447 - accuracy: 0.9708 - val_loss: 2.1076 - val_accuracy: 0.7144
Epoch 93/100
113/113 [=====] - 4s 36ms/step - loss: 0.0446 - accuracy: 0.9689 - val_loss: 2.1427 - val_accuracy: 0.7133
Epoch 94/100
113/113 [=====] - 4s 38ms/step - loss: 0.0445 - accuracy: 0.9689 - val_loss: 2.1635 - val_accuracy: 0.7122
Epoch 95/100
113/113 [=====] - 4s 36ms/step - loss: 0.0445 - accuracy: 0.9689 - val_loss: 2.1766 - val_accuracy: 0.7144
Epoch 96/100
113/113 [=====] - 5s 47ms/step - loss: 0.0443 - accuracy: 0.9689 - val_loss: 2.2264 - val_accuracy: 0.7189
Epoch 97/100
113/113 [=====] - 5s 42ms/step - loss: 0.0445 - accuracy: 0.9678 - val_loss: 2.2206 - val_accuracy: 0.7144
Epoch 98/100
113/113 [=====] - 4s 37ms/step - loss: 0.0445 - accuracy: 0.9681 - val_loss: 2.2421 - val_accuracy: 0.7178
Epoch 99/100
113/113 [=====] - 4s 36ms/step - loss: 0.0445 - accuracy: 0.9669 - val_loss: 2.2092 - val_accuracy: 0.7144
Epoch 100/100
113/113 [=====] - 5s 47ms/step - loss: 0.0445 - accuracy: 0.9675 - val_loss: 2.2136 - val_accuracy: 0.7133
<keras.src.callbacks.History at 0x7f24c8a36620>

```

### c. Pembagian Data 70 : 30

```

Epoch 1/100
99/99 [=====] - 6s 54ms/step - loss: 0.9628 - accuracy: 0.5311 - val_loss: 0.7973 - val_accuracy: 0.6356
Epoch 2/100
99/99 [=====] - 4s 45ms/step - loss: 0.5373 - accuracy: 0.7835 - val_loss: 0.6628 - val_accuracy: 0.7319
Epoch 3/100
99/99 [=====] - 4s 38ms/step - loss: 0.3527 - accuracy: 0.8619 - val_loss: 0.6931 - val_accuracy: 0.7437
Epoch 4/100
99/99 [=====] - 4s 39ms/step - loss: 0.2561 - accuracy: 0.8990 - val_loss: 0.7623 - val_accuracy: 0.7415
Epoch 5/100
99/99 [=====] - 5s 49ms/step - loss: 0.2034 - accuracy: 0.9194 - val_loss: 0.8183 - val_accuracy: 0.7341
Epoch 6/100
99/99 [=====] - 4s 39ms/step - loss: 0.1707 - accuracy: 0.9349 - val_loss: 0.8789 - val_accuracy: 0.7252
Epoch 7/100
99/99 [=====] - 4s 45ms/step - loss: 0.1489 - accuracy: 0.9413 - val_loss: 0.9312 - val_accuracy: 0.7200
Epoch 8/100
99/99 [=====] - 4s 43ms/step - loss: 0.1357 - accuracy: 0.9492 - val_loss: 0.9816 - val_accuracy: 0.7215
Epoch 9/100
99/99 [=====] - 4s 39ms/step - loss: 0.1253 - accuracy: 0.9514 - val_loss: 1.0198 - val_accuracy: 0.7230
Epoch 10/100
99/99 [=====] - 5s 53ms/step - loss: 0.1148 - accuracy: 0.9533 - val_loss: 1.0694 - val_accuracy: 0.7133
Epoch 11/100
99/99 [=====] - 4s 39ms/step - loss: 0.1095 - accuracy: 0.9571 - val_loss: 1.0943 - val_accuracy: 0.7207
Epoch 12/100
99/99 [=====] - 4s 45ms/step - loss: 0.1030 - accuracy: 0.9587 - val_loss: 1.1385 - val_accuracy: 0.7141
Epoch 13/100
99/99 [=====] - 4s 37ms/step - loss: 0.1005 - accuracy: 0.9613 - val_loss: 1.1386 - val_accuracy: 0.7200
Epoch 14/100
99/99 [=====] - 5s 48ms/step - loss: 0.0959 - accuracy: 0.9610 - val_loss: 1.1482 - val_accuracy: 0.7252
- . . .

Epoch 29/100
99/99 [=====] - 5s 49ms/step - loss: 0.0662 - accuracy: 0.9695 - val_loss: 1.3815 - val_accuracy: 0.7074
Epoch 30/100
99/99 [=====] - 4s 39ms/step - loss: 0.0675 - accuracy: 0.9683 - val_loss: 1.3938 - val_accuracy: 0.7074
Epoch 31/100
99/99 [=====] - 4s 36ms/step - loss: 0.0645 - accuracy: 0.9692 - val_loss: 1.4487 - val_accuracy: 0.6985
Epoch 32/100
99/99 [=====] - 5s 50ms/step - loss: 0.0659 - accuracy: 0.9692 - val_loss: 1.4699 - val_accuracy: 0.7037
Epoch 33/100
99/99 [=====] - 4s 38ms/step - loss: 0.0628 - accuracy: 0.9683 - val_loss: 1.5014 - val_accuracy: 0.6941
Epoch 34/100
99/99 [=====] - 4s 42ms/step - loss: 0.0653 - accuracy: 0.9689 - val_loss: 1.5349 - val_accuracy: 0.6978
Epoch 35/100
99/99 [=====] - 4s 38ms/step - loss: 0.0625 - accuracy: 0.9686 - val_loss: 1.5742 - val_accuracy: 0.6830
Epoch 36/100
99/99 [=====] - 4s 39ms/step - loss: 0.0675 - accuracy: 0.9683 - val_loss: 1.7687 - val_accuracy: 0.6881
Epoch 37/100
99/99 [=====] - 5s 49ms/step - loss: 0.0747 - accuracy: 0.9660 - val_loss: 1.5123 - val_accuracy: 0.7096
Epoch 38/100
99/99 [=====] - 4s 36ms/step - loss: 0.0726 - accuracy: 0.9651 - val_loss: 1.5229 - val_accuracy: 0.6941
Epoch 39/100
99/99 [=====] - 4s 44ms/step - loss: 0.0658 - accuracy: 0.9673 - val_loss: 1.5650 - val_accuracy: 0.7007
Epoch 40/100
99/99 [=====] - 4s 39ms/step - loss: 0.0648 - accuracy: 0.9689 - val_loss: 1.5625 - val_accuracy: 0.6941
Epoch 41/100
99/99 [=====] - 4s 37ms/step - loss: 0.0592 - accuracy: 0.9683 - val_loss: 1.6190 - val_accuracy: 0.6904
Epoch 42/100
99/99 [=====] - 5s 49ms/step - loss: 0.0611 - accuracy: 0.9698 - val_loss: 1.6600 - val_accuracy: 0.6941
Epoch 43/100

Epoch 57/100
99/99 [=====] - 4s 39ms/step - loss: 0.0519 - accuracy: 0.9676 - val_loss: 1.6241 - val_accuracy: 0.7178
Epoch 58/100
99/99 [=====] - 5s 49ms/step - loss: 0.0512 - accuracy: 0.9667 - val_loss: 1.6384 - val_accuracy: 0.7207
Epoch 59/100
99/99 [=====] - 4s 39ms/step - loss: 0.0511 - accuracy: 0.9663 - val_loss: 1.6540 - val_accuracy: 0.7178
Epoch 60/100
99/99 [=====] - 5s 46ms/step - loss: 0.0498 - accuracy: 0.9657 - val_loss: 1.6694 - val_accuracy: 0.7296
Epoch 61/100
99/99 [=====] - 4s 39ms/step - loss: 0.0491 - accuracy: 0.9663 - val_loss: 1.6793 - val_accuracy: 0.7274
Epoch 62/100
99/99 [=====] - 4s 39ms/step - loss: 0.0485 - accuracy: 0.9660 - val_loss: 1.6918 - val_accuracy: 0.7267
Epoch 63/100
99/99 [=====] - 5s 53ms/step - loss: 0.0477 - accuracy: 0.9676 - val_loss: 1.6969 - val_accuracy: 0.7304
Epoch 64/100
99/99 [=====] - 4s 39ms/step - loss: 0.0472 - accuracy: 0.9670 - val_loss: 1.7067 - val_accuracy: 0.7267
Epoch 65/100
99/99 [=====] - 5s 52ms/step - loss: 0.0466 - accuracy: 0.9660 - val_loss: 1.7213 - val_accuracy: 0.7230
Epoch 66/100
99/99 [=====] - 5s 48ms/step - loss: 0.0469 - accuracy: 0.9663 - val_loss: 1.7072 - val_accuracy: 0.7281
Epoch 67/100
99/99 [=====] - 4s 42ms/step - loss: 0.0459 - accuracy: 0.9673 - val_loss: 1.7506 - val_accuracy: 0.7267
Epoch 68/100
99/99 [=====] - 5s 48ms/step - loss: 0.0459 - accuracy: 0.9676 - val_loss: 1.7607 - val_accuracy: 0.7259
Epoch 69/100
99/99 [=====] - 6s 65ms/step - loss: 0.0457 - accuracy: 0.9673 - val_loss: 1.7722 - val_accuracy: 0.7274
Epoch 70/100
99/99 [=====] - 4s 39ms/step - loss: 0.0455 - accuracy: 0.9676 - val_loss: 1.7751 - val_accuracy: 0.7296
Epoch 71/100

```

```

Epoch 87/100
99/99 [=====] - 4s 37ms/step - loss: 0.0431 - accuracy: 0.9667 - val_loss: 1.9340 - val_accuracy: 0.7222
Epoch 88/100
99/99 [=====] - 5s 50ms/step - loss: 0.0427 - accuracy: 0.9686 - val_loss: 1.9442 - val_accuracy: 0.7207
Epoch 89/100
99/99 [=====] - 4s 39ms/step - loss: 0.0428 - accuracy: 0.9679 - val_loss: 1.9617 - val_accuracy: 0.7200
Epoch 90/100
99/99 [=====] - 4s 37ms/step - loss: 0.0425 - accuracy: 0.9673 - val_loss: 1.9600 - val_accuracy: 0.7259
Epoch 91/100
99/99 [=====] - 5s 53ms/step - loss: 0.0425 - accuracy: 0.9667 - val_loss: 1.9780 - val_accuracy: 0.7281
Epoch 92/100
99/99 [=====] - 4s 39ms/step - loss: 0.0424 - accuracy: 0.9676 - val_loss: 2.0001 - val_accuracy: 0.7281
Epoch 93/100
99/99 [=====] - 4s 39ms/step - loss: 0.0437 - accuracy: 0.9667 - val_loss: 1.9979 - val_accuracy: 0.7215
Epoch 94/100
99/99 [=====] - 5s 53ms/step - loss: 0.0425 - accuracy: 0.9660 - val_loss: 2.0386 - val_accuracy: 0.7252
Epoch 95/100
99/99 [=====] - 4s 37ms/step - loss: 0.0912 - accuracy: 0.9575 - val_loss: 2.1477 - val_accuracy: 0.6711
Epoch 96/100
99/99 [=====] - 4s 41ms/step - loss: 0.1123 - accuracy: 0.9508 - val_loss: 1.9105 - val_accuracy: 0.7156
Epoch 97/100
99/99 [=====] - 4s 39ms/step - loss: 0.0663 - accuracy: 0.9654 - val_loss: 2.0518 - val_accuracy: 0.6926
Epoch 98/100
99/99 [=====] - 5s 51ms/step - loss: 0.0489 - accuracy: 0.9737 - val_loss: 2.0985 - val_accuracy: 0.7148
Epoch 99/100
99/99 [=====] - 4s 39ms/step - loss: 0.0447 - accuracy: 0.9711 - val_loss: 2.1267 - val_accuracy: 0.7104
Epoch 100/100
99/99 [=====] - 4s 40ms/step - loss: 0.0442 - accuracy: 0.9708 - val_loss: 2.0854 - val_accuracy: 0.7141
<keras.src.callbacks.History at 0x7f24c8acae00>

```

## Lampiran 11. Proses Epoch 1 – 100 dengan Batch Size 24

### a. Pembagian Data 90 : 10

```

Epoch 1/100
169/169 [=====] - 5s 28ms/step - loss: 0.8549 - accuracy: 0.5906 - val_loss: 0.7174 - val_accuracy: 0.6933
Epoch 2/100
169/169 [=====] - 5s 29ms/step - loss: 0.4855 - accuracy: 0.8032 - val_loss: 0.7035 - val_accuracy: 0.7421
Epoch 3/100
169/169 [=====] - 5s 30ms/step - loss: 0.3400 - accuracy: 0.8679 - val_loss: 0.7801 - val_accuracy: 0.7356
Epoch 4/100
169/169 [=====] - 5s 27ms/step - loss: 0.2605 - accuracy: 0.8983 - val_loss: 0.8500 - val_accuracy: 0.7267
Epoch 5/100
169/169 [=====] - 6s 33ms/step - loss: 0.2135 - accuracy: 0.9210 - val_loss: 0.9156 - val_accuracy: 0.7133
Epoch 6/100
169/169 [=====] - 5s 30ms/step - loss: 0.1851 - accuracy: 0.9346 - val_loss: 0.9745 - val_accuracy: 0.7067
Epoch 7/100
169/169 [=====] - 5s 30ms/step - loss: 0.1658 - accuracy: 0.9388 - val_loss: 1.0164 - val_accuracy: 0.7085
Epoch 8/100
169/169 [=====] - 5s 27ms/step - loss: 0.1524 - accuracy: 0.9444 - val_loss: 1.0459 - val_accuracy: 0.7206
Epoch 9/100
169/169 [=====] - 6s 33ms/step - loss: 0.1406 - accuracy: 0.9499 - val_loss: 1.0773 - val_accuracy: 0.7176
Epoch 10/100
169/169 [=====] - 5s 31ms/step - loss: 0.1321 - accuracy: 0.9516 - val_loss: 1.1076 - val_accuracy: 0.7156
Epoch 11/100
169/169 [=====] - 5s 29ms/step - loss: 0.1234 - accuracy: 0.9546 - val_loss: 1.1285 - val_accuracy: 0.7222
Epoch 12/100
169/169 [=====] - 5s 27ms/step - loss: 0.1197 - accuracy: 0.9560 - val_loss: 1.1576 - val_accuracy: 0.7089
Epoch 13/100
169/169 [=====] - 6s 33ms/step - loss: 0.1120 - accuracy: 0.9590 - val_loss: 1.1686 - val_accuracy: 0.7178
Epoch 14/100
169/169 [=====] - 5s 30ms/step - loss: 0.1083 - accuracy: 0.9622 - val_loss: 1.1983 - val_accuracy: 0.7254
Epoch 27/100
169/169 [=====] - 6s 33ms/step - loss: 0.0683 - accuracy: 0.9617 - val_loss: 1.6014 - val_accuracy: 0.6978
Epoch 28/100
169/169 [=====] - 5s 28ms/step - loss: 0.0679 - accuracy: 0.9612 - val_loss: 1.5709 - val_accuracy: 0.7022
Epoch 29/100
169/169 [=====] - 5s 32ms/step - loss: 0.0662 - accuracy: 0.9625 - val_loss: 1.6802 - val_accuracy: 0.6956
Epoch 30/100
169/169 [=====] - 6s 34ms/step - loss: 0.0652 - accuracy: 0.9622 - val_loss: 1.6782 - val_accuracy: 0.6867
Epoch 31/100
169/169 [=====] - 5s 28ms/step - loss: 0.0636 - accuracy: 0.9632 - val_loss: 1.7195 - val_accuracy: 0.6911
Epoch 32/100
169/169 [=====] - 5s 32ms/step - loss: 0.0635 - accuracy: 0.9625 - val_loss: 1.7228 - val_accuracy: 0.6889
Epoch 33/100
169/169 [=====] - 6s 33ms/step - loss: 0.0688 - accuracy: 0.9605 - val_loss: 1.7973 - val_accuracy: 0.6844
Epoch 34/100
169/169 [=====] - 5s 27ms/step - loss: 0.0779 - accuracy: 0.9588 - val_loss: 1.8861 - val_accuracy: 0.6867
Epoch 35/100
169/169 [=====] - 5s 28ms/step - loss: 0.0658 - accuracy: 0.9617 - val_loss: 1.8836 - val_accuracy: 0.6911
Epoch 36/100
169/169 [=====] - 5s 32ms/step - loss: 0.0609 - accuracy: 0.9640 - val_loss: 1.9629 - val_accuracy: 0.6889
Epoch 37/100
169/169 [=====] - 6s 33ms/step - loss: 0.0589 - accuracy: 0.9652 - val_loss: 2.0394 - val_accuracy: 0.6933
Epoch 38/100
169/169 [=====] - 5s 27ms/step - loss: 0.0560 - accuracy: 0.9662 - val_loss: 2.0420 - val_accuracy: 0.6933
Epoch 39/100
169/169 [=====] - 5s 30ms/step - loss: 0.0557 - accuracy: 0.9652 - val_loss: 2.0784 - val_accuracy: 0.6933
Epoch 40/100
169/169 [=====] - 5s 31ms/step - loss: 0.0571 - accuracy: 0.9654 - val_loss: 2.0728 - val_accuracy: 0.6778

```

```

Epoch 67/100
169/169 [=====] - 5s 28ms/step - loss: 0.0550 - accuracy: 0.9662 - val_loss: 2.7105 - val_accuracy: 0.6911
Epoch 68/100
169/169 [=====] - 5s 32ms/step - loss: 0.0800 - accuracy: 0.9617 - val_loss: 2.7241 - val_accuracy: 0.6622
Epoch 69/100
169/169 [=====] - 5s 27ms/step - loss: 0.0817 - accuracy: 0.9588 - val_loss: 2.7770 - val_accuracy: 0.6667
Epoch 70/100
169/169 [=====] - 6s 33ms/step - loss: 0.0591 - accuracy: 0.9652 - val_loss: 2.7163 - val_accuracy: 0.6778
Epoch 71/100
169/169 [=====] - 5s 28ms/step - loss: 0.0515 - accuracy: 0.9679 - val_loss: 2.7936 - val_accuracy: 0.6822
Epoch 72/100
169/169 [=====] - 5s 28ms/step - loss: 0.0500 - accuracy: 0.9686 - val_loss: 2.7949 - val_accuracy: 0.6822
Epoch 73/100
169/169 [=====] - 5s 30ms/step - loss: 0.0489 - accuracy: 0.9696 - val_loss: 2.8889 - val_accuracy: 0.6867
Epoch 74/100
169/169 [=====] - 6s 33ms/step - loss: 0.0485 - accuracy: 0.9691 - val_loss: 2.9164 - val_accuracy: 0.6867
Epoch 75/100
169/169 [=====] - 5s 29ms/step - loss: 0.0483 - accuracy: 0.9694 - val_loss: 2.9489 - val_accuracy: 0.6911
Epoch 76/100
169/169 [=====] - 5s 29ms/step - loss: 0.0482 - accuracy: 0.9681 - val_loss: 2.9665 - val_accuracy: 0.6889
Epoch 77/100
169/169 [=====] - 5s 27ms/step - loss: 0.0482 - accuracy: 0.9674 - val_loss: 2.9975 - val_accuracy: 0.6822
Epoch 78/100
169/169 [=====] - 6s 33ms/step - loss: 0.0482 - accuracy: 0.9669 - val_loss: 3.0043 - val_accuracy: 0.6933
Epoch 79/100
169/169 [=====] - 5s 31ms/step - loss: 0.0480 - accuracy: 0.9667 - val_loss: 3.0450 - val_accuracy: 0.6889
Epoch 80/100
169/169 [=====] - 5s 27ms/step - loss: 0.0482 - accuracy: 0.9667 - val_loss: 3.0057 - val_accuracy: 0.6933
<keras.src.callbacks.History at 0x7f24c82e5de0>

```

## b. Pembagian Data 80 : 20

```

Epoch 1/100
150/150 [=====] - 6s 33ms/step - loss: 0.8826 - accuracy: 0.5753 - val_loss: 0.7201 - val_accuracy: 0.6878
Epoch 2/100
150/150 [=====] - 4s 28ms/step - loss: 0.5032 - accuracy: 0.8022 - val_loss: 0.6634 - val_accuracy: 0.7344
Epoch 3/100
150/150 [=====] - 5s 34ms/step - loss: 0.3429 - accuracy: 0.8669 - val_loss: 0.7030 - val_accuracy: 0.7411
Epoch 4/100
150/150 [=====] - 4s 29ms/step - loss: 0.2561 - accuracy: 0.8997 - val_loss: 0.7621 - val_accuracy: 0.7333
Epoch 5/100
150/150 [=====] - 4s 28ms/step - loss: 0.2073 - accuracy: 0.9233 - val_loss: 0.8317 - val_accuracy: 0.7322
Epoch 6/100
150/150 [=====] - 5s 35ms/step - loss: 0.1798 - accuracy: 0.9347 - val_loss: 0.8942 - val_accuracy: 0.7333
Epoch 7/100
150/150 [=====] - 4s 28ms/step - loss: 0.1636 - accuracy: 0.9406 - val_loss: 0.9229 - val_accuracy: 0.7322
Epoch 8/100
150/150 [=====] - 5s 32ms/step - loss: 0.1514 - accuracy: 0.9453 - val_loss: 0.9658 - val_accuracy: 0.7289
Epoch 9/100
150/150 [=====] - 5s 30ms/step - loss: 0.1425 - accuracy: 0.9467 - val_loss: 0.9843 - val_accuracy: 0.7222
Epoch 10/100
150/150 [=====] - 4s 28ms/step - loss: 0.1307 - accuracy: 0.9525 - val_loss: 1.0103 - val_accuracy: 0.7200
Epoch 11/100
150/150 [=====] - 5s 35ms/step - loss: 0.1233 - accuracy: 0.9561 - val_loss: 1.0406 - val_accuracy: 0.7167
Epoch 12/100
150/150 [=====] - 4s 28ms/step - loss: 0.1143 - accuracy: 0.9586 - val_loss: 1.0577 - val_accuracy: 0.7211
Epoch 13/100
150/150 [=====] - 4s 30ms/step - loss: 0.1091 - accuracy: 0.9608 - val_loss: 1.0735 - val_accuracy: 0.7256
Epoch 14/100
150/150 [=====] - 5s 30ms/step - loss: 0.1019 - accuracy: 0.9614 - val_loss: 1.1057 - val_accuracy: 0.7233

```

```

Epoch 29/100
150/150 [=====] - 5s 35ms/step - loss: 0.0604 - accuracy: 0.9650 - val_loss: 1.5155 - val_accuracy: 0.7022
Epoch 30/100
150/150 [=====] - 5s 33ms/step - loss: 0.0588 - accuracy: 0.9647 - val_loss: 1.5917 - val_accuracy: 0.7000
Epoch 31/100
150/150 [=====] - 5s 31ms/step - loss: 0.0583 - accuracy: 0.9650 - val_loss: 1.6242 - val_accuracy: 0.6989
Epoch 32/100
150/150 [=====] - 4s 29ms/step - loss: 0.0587 - accuracy: 0.9647 - val_loss: 1.6588 - val_accuracy: 0.6978
Epoch 33/100
150/150 [=====] - 5s 35ms/step - loss: 0.0652 - accuracy: 0.9636 - val_loss: 1.8337 - val_accuracy: 0.6967
Epoch 34/100
150/150 [=====] - 4s 28ms/step - loss: 0.0884 - accuracy: 0.9539 - val_loss: 1.7474 - val_accuracy: 0.6833
Epoch 35/100
150/150 [=====] - 4s 29ms/step - loss: 0.0905 - accuracy: 0.9536 - val_loss: 1.5728 - val_accuracy: 0.7000
Epoch 36/100
150/150 [=====] - 5s 35ms/step - loss: 0.0704 - accuracy: 0.9644 - val_loss: 1.6918 - val_accuracy: 0.7056
Epoch 37/100
150/150 [=====] - 4s 29ms/step - loss: 0.0591 - accuracy: 0.9681 - val_loss: 1.7951 - val_accuracy: 0.7056
Epoch 38/100
150/150 [=====] - 5s 36ms/step - loss: 0.0539 - accuracy: 0.9672 - val_loss: 1.8328 - val_accuracy: 0.7078
Epoch 39/100
150/150 [=====] - 4s 28ms/step - loss: 0.0532 - accuracy: 0.9672 - val_loss: 1.8454 - val_accuracy: 0.7167
Epoch 40/100
150/150 [=====] - 4s 29ms/step - loss: 0.0534 - accuracy: 0.9656 - val_loss: 1.8446 - val_accuracy: 0.7178
Epoch 41/100
150/150 [=====] - 5s 36ms/step - loss: 0.0532 - accuracy: 0.9653 - val_loss: 1.8394 - val_accuracy: 0.7200
Epoch 42/100
150/150 [=====] - 5s 34ms/step - loss: 0.0537 - accuracy: 0.9653 - val_loss: 1.8389 - val_accuracy: 0.7144
Epoch 43/100

Epoch 57/100
150/150 [=====] - 5s 31ms/step - loss: 0.0483 - accuracy: 0.9653 - val_loss: 1.8815 - val_accuracy: 0.7200
Epoch 58/100
150/150 [=====] - 5s 31ms/step - loss: 0.0482 - accuracy: 0.9650 - val_loss: 1.8996 - val_accuracy: 0.7333
Epoch 59/100
150/150 [=====] - 4s 29ms/step - loss: 0.0472 - accuracy: 0.9650 - val_loss: 1.9781 - val_accuracy: 0.7300
Epoch 60/100
150/150 [=====] - 5s 36ms/step - loss: 0.0471 - accuracy: 0.9658 - val_loss: 1.9988 - val_accuracy: 0.7322
Epoch 61/100
150/150 [=====] - 4s 29ms/step - loss: 0.0483 - accuracy: 0.9656 - val_loss: 1.9725 - val_accuracy: 0.7189
Epoch 62/100
150/150 [=====] - 5s 32ms/step - loss: 0.0493 - accuracy: 0.9672 - val_loss: 2.4014 - val_accuracy: 0.7122
Epoch 63/100
150/150 [=====] - 5s 31ms/step - loss: 0.0811 - accuracy: 0.9539 - val_loss: 2.0616 - val_accuracy: 0.7000
Epoch 64/100
150/150 [=====] - 4s 29ms/step - loss: 0.0939 - accuracy: 0.9550 - val_loss: 1.9053 - val_accuracy: 0.7189
Epoch 65/100
150/150 [=====] - 5s 36ms/step - loss: 0.0621 - accuracy: 0.9650 - val_loss: 2.0969 - val_accuracy: 0.6911
Epoch 66/100
150/150 [=====] - 4s 29ms/step - loss: 0.0500 - accuracy: 0.9683 - val_loss: 2.1663 - val_accuracy: 0.7056
Epoch 67/100
150/150 [=====] - 5s 34ms/step - loss: 0.0464 - accuracy: 0.9694 - val_loss: 2.1971 - val_accuracy: 0.7067
Epoch 68/100
150/150 [=====] - 5s 30ms/step - loss: 0.0456 - accuracy: 0.9692 - val_loss: 2.2414 - val_accuracy: 0.7089
Epoch 69/100
150/150 [=====] - 4s 29ms/step - loss: 0.0455 - accuracy: 0.9669 - val_loss: 2.2905 - val_accuracy: 0.7067
Epoch 70/100
150/150 [=====] - 5s 36ms/step - loss: 0.0452 - accuracy: 0.9667 - val_loss: 2.3318 - val_accuracy: 0.7078
Epoch 71/100

Epoch 87/100
150/150 [=====] - 4s 28ms/step - loss: 0.0446 - accuracy: 0.9653 - val_loss: 2.5198 - val_accuracy: 0.7000
Epoch 88/100
150/150 [=====] - 5s 38ms/step - loss: 0.0449 - accuracy: 0.9656 - val_loss: 2.5075 - val_accuracy: 0.7133
Epoch 89/100
150/150 [=====] - 4s 29ms/step - loss: 0.0455 - accuracy: 0.9658 - val_loss: 2.5699 - val_accuracy: 0.7044
Epoch 90/100
150/150 [=====] - 5s 35ms/step - loss: 0.0448 - accuracy: 0.9683 - val_loss: 2.5847 - val_accuracy: 0.7033
Epoch 91/100
150/150 [=====] - 5s 36ms/step - loss: 0.0448 - accuracy: 0.9669 - val_loss: 2.5988 - val_accuracy: 0.7033
Epoch 92/100
150/150 [=====] - 4s 29ms/step - loss: 0.0448 - accuracy: 0.9656 - val_loss: 2.6242 - val_accuracy: 0.7044
Epoch 93/100
150/150 [=====] - 4s 29ms/step - loss: 0.0757 - accuracy: 0.9642 - val_loss: 2.9981 - val_accuracy: 0.6900
Epoch 94/100
150/150 [=====] - 5s 36ms/step - loss: 0.0978 - accuracy: 0.9558 - val_loss: 2.5640 - val_accuracy: 0.6856
Epoch 95/100
150/150 [=====] - 5s 36ms/step - loss: 0.0569 - accuracy: 0.9678 - val_loss: 2.5156 - val_accuracy: 0.6911
Epoch 96/100
150/150 [=====] - 4s 29ms/step - loss: 0.0463 - accuracy: 0.9706 - val_loss: 2.4772 - val_accuracy: 0.7056
Epoch 97/100
150/150 [=====] - 5s 30ms/step - loss: 0.0447 - accuracy: 0.9706 - val_loss: 2.5334 - val_accuracy: 0.7056
Epoch 98/100
150/150 [=====] - 4s 29ms/step - loss: 0.0441 - accuracy: 0.9703 - val_loss: 2.6189 - val_accuracy: 0.7056
Epoch 99/100
150/150 [=====] - 5s 36ms/step - loss: 0.0438 - accuracy: 0.9700 - val_loss: 2.6642 - val_accuracy: 0.7067
Epoch 100/100
150/150 [=====] - 4s 28ms/step - loss: 0.0438 - accuracy: 0.9694 - val_loss: 2.6690 - val_accuracy: 0.7100

```

### c. Pembagian Data 70 : 30

```

Epoch 1/100
132/132 [=====] - 6s 36ms/step - loss: 0.9420 - accuracy: 0.5394 - val_loss: 0.7226 - val_accuracy: 0.6889
Epoch 2/100
132/132 [=====] - 4s 29ms/step - loss: 0.5249 - accuracy: 0.7883 - val_loss: 0.6992 - val_accuracy: 0.7215
Epoch 3/100
132/132 [=====] - 5s 35ms/step - loss: 0.3467 - accuracy: 0.8689 - val_loss: 0.7229 - val_accuracy: 0.7326
Epoch 4/100
132/132 [=====] - 4s 31ms/step - loss: 0.2496 - accuracy: 0.9025 - val_loss: 0.7835 - val_accuracy: 0.7370
Epoch 5/100
132/132 [=====] - 4s 31ms/step - loss: 0.1972 - accuracy: 0.9222 - val_loss: 0.8453 - val_accuracy: 0.7319
Epoch 6/100
132/132 [=====] - 5s 37ms/step - loss: 0.1677 - accuracy: 0.9333 - val_loss: 0.9062 - val_accuracy: 0.7281
Epoch 7/100
132/132 [=====] - 4s 29ms/step - loss: 0.1487 - accuracy: 0.9416 - val_loss: 0.9632 - val_accuracy: 0.7289
Epoch 8/100
132/132 [=====] - 4s 31ms/step - loss: 0.1341 - accuracy: 0.9473 - val_loss: 1.0270 - val_accuracy: 0.7281
Epoch 9/100
132/132 [=====] - 5s 38ms/step - loss: 0.1241 - accuracy: 0.9521 - val_loss: 1.0580 - val_accuracy: 0.7230
Epoch 10/100
132/132 [=====] - 4s 31ms/step - loss: 0.1178 - accuracy: 0.9552 - val_loss: 1.1100 - val_accuracy: 0.7156
Epoch 11/100
132/132 [=====] - 4s 31ms/step - loss: 0.1096 - accuracy: 0.9587 - val_loss: 1.1460 - val_accuracy: 0.7185
Epoch 12/100
132/132 [=====] - 4s 33ms/step - loss: 0.1053 - accuracy: 0.9613 - val_loss: 1.1979 - val_accuracy: 0.7126
Epoch 13/100
132/132 [=====] - 4s 31ms/step - loss: 0.0988 - accuracy: 0.9625 - val_loss: 1.2145 - val_accuracy: 0.7185
Epoch 14/100
132/132 [=====] - 5s 37ms/step - loss: 0.0945 - accuracy: 0.9632 - val_loss: 1.2131 - val_accuracy: 0.7193
Epoch 15/100

Epoch 29/100
132/132 [=====] - 4s 29ms/step - loss: 0.0655 - accuracy: 0.9679 - val_loss: 1.3870 - val_accuracy: 0.7163
Epoch 30/100
132/132 [=====] - 4s 29ms/step - loss: 0.0636 - accuracy: 0.9667 - val_loss: 1.4209 - val_accuracy: 0.7133
Epoch 31/100
132/132 [=====] - 5s 36ms/step - loss: 0.0613 - accuracy: 0.9667 - val_loss: 1.4881 - val_accuracy: 0.7052
Epoch 32/100
132/132 [=====] - 4s 31ms/step - loss: 0.0611 - accuracy: 0.9660 - val_loss: 1.5356 - val_accuracy: 0.6970
Epoch 33/100
132/132 [=====] - 4s 31ms/step - loss: 0.0598 - accuracy: 0.9667 - val_loss: 1.5788 - val_accuracy: 0.6963
Epoch 34/100
132/132 [=====] - 4s 31ms/step - loss: 0.0648 - accuracy: 0.9644 - val_loss: 1.6352 - val_accuracy: 0.6933
Epoch 35/100
132/132 [=====] - 4s 31ms/step - loss: 0.0798 - accuracy: 0.9635 - val_loss: 1.5477 - val_accuracy: 0.7007
Epoch 36/100
132/132 [=====] - 5s 38ms/step - loss: 0.0747 - accuracy: 0.9619 - val_loss: 1.7546 - val_accuracy: 0.6837
Epoch 37/100
132/132 [=====] - 4s 31ms/step - loss: 0.0626 - accuracy: 0.9676 - val_loss: 1.7573 - val_accuracy: 0.6933
Epoch 38/100
132/132 [=====] - 4s 31ms/step - loss: 0.0576 - accuracy: 0.9683 - val_loss: 1.7727 - val_accuracy: 0.6978
Epoch 39/100
132/132 [=====] - 4s 33ms/step - loss: 0.0526 - accuracy: 0.9698 - val_loss: 1.7844 - val_accuracy: 0.7030
Epoch 40/100
132/132 [=====] - 4s 31ms/step - loss: 0.0524 - accuracy: 0.9698 - val_loss: 1.8487 - val_accuracy: 0.6963
Epoch 41/100
132/132 [=====] - 5s 39ms/step - loss: 0.0517 - accuracy: 0.9686 - val_loss: 1.8225 - val_accuracy: 0.6948
Epoch 42/100
132/132 [=====] - 4s 31ms/step - loss: 0.0515 - accuracy: 0.9698 - val_loss: 1.8023 - val_accuracy: 0.6978

Epoch 57/100
132/132 [=====] - 4s 31ms/step - loss: 0.0456 - accuracy: 0.9657 - val_loss: 1.8195 - val_accuracy: 0.7207
Epoch 58/100
132/132 [=====] - 4s 30ms/step - loss: 0.0455 - accuracy: 0.9663 - val_loss: 1.8384 - val_accuracy: 0.7237
Epoch 59/100
132/132 [=====] - 5s 37ms/step - loss: 0.0459 - accuracy: 0.9657 - val_loss: 1.8484 - val_accuracy: 0.7156
Epoch 60/100
132/132 [=====] - 4s 32ms/step - loss: 0.0449 - accuracy: 0.9670 - val_loss: 1.8473 - val_accuracy: 0.7178
Epoch 61/100
132/132 [=====] - 4s 31ms/step - loss: 0.0442 - accuracy: 0.9660 - val_loss: 1.8454 - val_accuracy: 0.7148
Epoch 62/100
132/132 [=====] - 4s 32ms/step - loss: 0.0441 - accuracy: 0.9683 - val_loss: 1.8924 - val_accuracy: 0.7185
Epoch 63/100
132/132 [=====] - 4s 32ms/step - loss: 0.0441 - accuracy: 0.9667 - val_loss: 1.9005 - val_accuracy: 0.7207
Epoch 64/100
132/132 [=====] - 5s 41ms/step - loss: 0.0435 - accuracy: 0.9683 - val_loss: 1.9308 - val_accuracy: 0.7170
Epoch 65/100
132/132 [=====] - 4s 31ms/step - loss: 0.0434 - accuracy: 0.9673 - val_loss: 1.9602 - val_accuracy: 0.7170
Epoch 66/100
132/132 [=====] - 4s 31ms/step - loss: 0.0435 - accuracy: 0.9676 - val_loss: 1.9785 - val_accuracy: 0.7141
Epoch 67/100
132/132 [=====] - 5s 36ms/step - loss: 0.0431 - accuracy: 0.9660 - val_loss: 1.9707 - val_accuracy: 0.7156
Epoch 68/100
132/132 [=====] - 4s 29ms/step - loss: 0.0430 - accuracy: 0.9663 - val_loss: 1.9942 - val_accuracy: 0.7141
Epoch 69/100
132/132 [=====] - 5s 40ms/step - loss: 0.0430 - accuracy: 0.9670 - val_loss: 1.9847 - val_accuracy: 0.7119
Epoch 70/100
132/132 [=====] - 4s 31ms/step - loss: 0.0434 - accuracy: 0.9667 - val_loss: 2.0013 - val_accuracy: 0.7126

```

```

Epoch 87/100
132/132 [=====] - 5s 37ms/step - loss: 0.0413 - accuracy: 0.9679 - val_loss: 2.3202 - val_accuracy: 0.7030
Epoch 88/100
132/132 [=====] - 4s 31ms/step - loss: 0.0427 - accuracy: 0.9683 - val_loss: 2.3028 - val_accuracy: 0.6978
Epoch 89/100
132/132 [=====] - 5s 38ms/step - loss: 0.0412 - accuracy: 0.9689 - val_loss: 2.3113 - val_accuracy: 0.7015
Epoch 90/100
132/132 [=====] - 4s 30ms/step - loss: 0.0413 - accuracy: 0.9689 - val_loss: 2.3297 - val_accuracy: 0.6978
Epoch 91/100
132/132 [=====] - 4s 32ms/step - loss: 0.0413 - accuracy: 0.9683 - val_loss: 2.3437 - val_accuracy: 0.6985
Epoch 92/100
132/132 [=====] - 5s 36ms/step - loss: 0.0412 - accuracy: 0.9679 - val_loss: 2.3342 - val_accuracy: 0.7022
Epoch 93/100
132/132 [=====] - 4s 31ms/step - loss: 0.0412 - accuracy: 0.9676 - val_loss: 2.3444 - val_accuracy: 0.7030
Epoch 94/100
132/132 [=====] - 5s 36ms/step - loss: 0.0412 - accuracy: 0.9676 - val_loss: 2.3394 - val_accuracy: 0.7015
Epoch 95/100
132/132 [=====] - 4s 31ms/step - loss: 0.0412 - accuracy: 0.9679 - val_loss: 2.3674 - val_accuracy: 0.7037
Epoch 96/100
132/132 [=====] - 4s 32ms/step - loss: 0.0411 - accuracy: 0.9689 - val_loss: 2.3859 - val_accuracy: 0.7059
Epoch 97/100
132/132 [=====] - 5s 40ms/step - loss: 0.0412 - accuracy: 0.9670 - val_loss: 2.3917 - val_accuracy: 0.7044
Epoch 98/100
132/132 [=====] - 4s 30ms/step - loss: 0.0410 - accuracy: 0.9679 - val_loss: 2.4092 - val_accuracy: 0.7022
Epoch 99/100
132/132 [=====] - 4s 31ms/step - loss: 0.0418 - accuracy: 0.9683 - val_loss: 2.3575 - val_accuracy: 0.7030
Epoch 100/100
132/132 [=====] - 4s 32ms/step - loss: 0.0415 - accuracy: 0.9679 - val_loss: 2.3947 - val_accuracy: 0.7000
<keras.src.callbacks.History at 0x7f24c2e63ca0>

```

## Lampiran 12. Proses Epoch 1 – 100 dengan Batch Size 64

### a. Pembagian Data 90 : 10

```

Epoch 1/100
64/64 [=====] - 5s 64ms/step - loss: 0.9870 - accuracy: 0.5314 - val_loss: 0.7691 - val_accuracy: 0.6644
Epoch 2/100
64/64 [=====] - 5s 78ms/step - loss: 0.5669 - accuracy: 0.7679 - val_loss: 0.6238 - val_accuracy: 0.7467
Epoch 3/100
64/64 [=====] - 4s 60ms/step - loss: 0.3877 - accuracy: 0.8499 - val_loss: 0.6617 - val_accuracy: 0.7467
Epoch 4/100
64/64 [=====] - 4s 70ms/step - loss: 0.2956 - accuracy: 0.8840 - val_loss: 0.7413 - val_accuracy: 0.7422
Epoch 5/100
64/64 [=====] - 4s 61ms/step - loss: 0.2395 - accuracy: 0.9089 - val_loss: 0.8142 - val_accuracy: 0.7289
Epoch 6/100
64/64 [=====] - 4s 60ms/step - loss: 0.2024 - accuracy: 0.9237 - val_loss: 0.8308 - val_accuracy: 0.7311
Epoch 7/100
64/64 [=====] - 5s 77ms/step - loss: 0.1771 - accuracy: 0.9328 - val_loss: 0.9469 - val_accuracy: 0.7200
Epoch 8/100
64/64 [=====] - 4s 61ms/step - loss: 0.1599 - accuracy: 0.9415 - val_loss: 0.9948 - val_accuracy: 0.7156
Epoch 9/100
64/64 [=====] - 4s 60ms/step - loss: 0.1495 - accuracy: 0.9454 - val_loss: 1.0139 - val_accuracy: 0.7178
Epoch 10/100
64/64 [=====] - 5s 76ms/step - loss: 0.1395 - accuracy: 0.9489 - val_loss: 1.0417 - val_accuracy: 0.7222
Epoch 11/100
64/64 [=====] - 4s 61ms/step - loss: 0.1312 - accuracy: 0.9531 - val_loss: 1.0660 - val_accuracy: 0.7244
Epoch 12/100
64/64 [=====] - 4s 63ms/step - loss: 0.1226 - accuracy: 0.9543 - val_loss: 1.1030 - val_accuracy: 0.7200
Epoch 13/100
64/64 [=====] - 5s 74ms/step - loss: 0.1171 - accuracy: 0.9565 - val_loss: 1.1349 - val_accuracy: 0.7200
Epoch 14/100

Epoch 27/100
64/64 [=====] - 5s 74ms/step - loss: 0.0839 - accuracy: 0.9630 - val_loss: 1.3096 - val_accuracy: 0.6911
Epoch 28/100
64/64 [=====] - 4s 61ms/step - loss: 0.0824 - accuracy: 0.9635 - val_loss: 1.3248 - val_accuracy: 0.6822
Epoch 29/100
64/64 [=====] - 5s 78ms/step - loss: 0.0813 - accuracy: 0.9632 - val_loss: 1.3157 - val_accuracy: 0.6933
Epoch 30/100
64/64 [=====] - 4s 61ms/step - loss: 0.0799 - accuracy: 0.9640 - val_loss: 1.3328 - val_accuracy: 0.6867
Epoch 31/100
64/64 [=====] - 4s 61ms/step - loss: 0.0795 - accuracy: 0.9640 - val_loss: 1.3264 - val_accuracy: 0.6933
Epoch 32/100
64/64 [=====] - 5s 78ms/step - loss: 0.0777 - accuracy: 0.9652 - val_loss: 1.3448 - val_accuracy: 0.6889
Epoch 33/100
64/64 [=====] - 4s 60ms/step - loss: 0.0776 - accuracy: 0.9654 - val_loss: 1.3396 - val_accuracy: 0.6933
Epoch 34/100
64/64 [=====] - 4s 67ms/step - loss: 0.0764 - accuracy: 0.9664 - val_loss: 1.3688 - val_accuracy: 0.6911
Epoch 35/100
64/64 [=====] - 4s 61ms/step - loss: 0.0762 - accuracy: 0.9662 - val_loss: 1.3608 - val_accuracy: 0.6911
Epoch 36/100
64/64 [=====] - 4s 60ms/step - loss: 0.0755 - accuracy: 0.9664 - val_loss: 1.3769 - val_accuracy: 0.6889
Epoch 37/100
64/64 [=====] - 5s 76ms/step - loss: 0.0753 - accuracy: 0.9669 - val_loss: 1.3748 - val_accuracy: 0.6867
Epoch 38/100
64/64 [=====] - 4s 60ms/step - loss: 0.0739 - accuracy: 0.9669 - val_loss: 1.3995 - val_accuracy: 0.6867
Epoch 39/100
64/64 [=====] - 4s 64ms/step - loss: 0.0745 - accuracy: 0.9667 - val_loss: 1.3874 - val_accuracy: 0.6844

```

```

Epoch 66/100
64/64 [=====] - 4s 62ms/step - loss: 0.0587 - accuracy: 0.9649 - val_loss: 1.7317 - val_accuracy: 0.6822
Epoch 67/100
64/64 [=====] - 4s 61ms/step - loss: 0.0570 - accuracy: 0.9652 - val_loss: 1.7500 - val_accuracy: 0.6800
Epoch 68/100
64/64 [=====] - 5s 76ms/step - loss: 0.0568 - accuracy: 0.9647 - val_loss: 1.7542 - val_accuracy: 0.6844
Epoch 69/100
64/64 [=====] - 4s 62ms/step - loss: 0.0564 - accuracy: 0.9649 - val_loss: 1.7617 - val_accuracy: 0.6822
Epoch 70/100
64/64 [=====] - 5s 79ms/step - loss: 0.0561 - accuracy: 0.9640 - val_loss: 1.7776 - val_accuracy: 0.6822
Epoch 71/100
64/64 [=====] - 4s 61ms/step - loss: 0.0554 - accuracy: 0.9647 - val_loss: 1.7821 - val_accuracy: 0.6822
Epoch 72/100
64/64 [=====] - 4s 61ms/step - loss: 0.0556 - accuracy: 0.9635 - val_loss: 1.8022 - val_accuracy: 0.6844
Epoch 73/100
64/64 [=====] - 5s 80ms/step - loss: 0.0552 - accuracy: 0.9637 - val_loss: 1.8056 - val_accuracy: 0.6889
Epoch 74/100
64/64 [=====] - 4s 61ms/step - loss: 0.0553 - accuracy: 0.9642 - val_loss: 1.8084 - val_accuracy: 0.6911
Epoch 75/100
64/64 [=====] - 4s 61ms/step - loss: 0.0545 - accuracy: 0.9642 - val_loss: 1.8151 - val_accuracy: 0.6844
Epoch 76/100
64/64 [=====] - 5s 72ms/step - loss: 0.0545 - accuracy: 0.9632 - val_loss: 1.8166 - val_accuracy: 0.6822
Epoch 77/100
64/64 [=====] - 4s 61ms/step - loss: 0.0541 - accuracy: 0.9632 - val_loss: 1.8382 - val_accuracy: 0.6911
Epoch 78/100
64/64 [=====] - 5s 72ms/step - loss: 0.0541 - accuracy: 0.9630 - val_loss: 1.8503 - val_accuracy: 0.6867
Epoch 92/100
64/64 [=====] - 4s 62ms/step - loss: 0.0522 - accuracy: 0.9627 - val_loss: 1.9247 - val_accuracy: 0.6978
Epoch 93/100
64/64 [=====] - 5s 75ms/step - loss: 0.0522 - accuracy: 0.9635 - val_loss: 1.9245 - val_accuracy: 0.6956
Epoch 94/100
64/64 [=====] - 4s 61ms/step - loss: 0.0518 - accuracy: 0.9635 - val_loss: 1.9389 - val_accuracy: 0.6933
Epoch 95/100
64/64 [=====] - 4s 61ms/step - loss: 0.0518 - accuracy: 0.9632 - val_loss: 1.9453 - val_accuracy: 0.6978
Epoch 96/100
64/64 [=====] - 5s 79ms/step - loss: 0.0517 - accuracy: 0.9632 - val_loss: 1.9460 - val_accuracy: 0.6956
Epoch 97/100
64/64 [=====] - 4s 61ms/step - loss: 0.0515 - accuracy: 0.9632 - val_loss: 1.9730 - val_accuracy: 0.7022
Epoch 98/100
64/64 [=====] - 4s 65ms/step - loss: 0.0516 - accuracy: 0.9630 - val_loss: 1.9651 - val_accuracy: 0.6978
Epoch 99/100
64/64 [=====] - 4s 63ms/step - loss: 0.0515 - accuracy: 0.9632 - val_loss: 1.9870 - val_accuracy: 0.6956
Epoch 100/100
64/64 [=====] - 4s 62ms/step - loss: 0.0513 - accuracy: 0.9642 - val_loss: 1.9751 - val_accuracy: 0.6911
<keras.callbacks.History at 0x7f24c2b35cc0>

```

## b. Pembagian Data 80 : 20

```

Epoch 1/100
57/57 [=====] - 5s 67ms/step - loss: 1.0189 - accuracy: 0.4989 - val_loss: 0.8542 - val_accuracy: 0.6078
Epoch 2/100
57/57 [=====] - 4s 64ms/step - loss: 0.6264 - accuracy: 0.7428 - val_loss: 0.6014 - val_accuracy: 0.7478
Epoch 3/100
57/57 [=====] - 5s 84ms/step - loss: 0.4101 - accuracy: 0.8417 - val_loss: 0.6283 - val_accuracy: 0.7444
Epoch 4/100
57/57 [=====] - 4s 65ms/step - loss: 0.3061 - accuracy: 0.8856 - val_loss: 0.6783 - val_accuracy: 0.7444
Epoch 5/100
57/57 [=====] - 4s 64ms/step - loss: 0.2456 - accuracy: 0.9111 - val_loss: 0.7234 - val_accuracy: 0.7400
Epoch 6/100
57/57 [=====] - 5s 85ms/step - loss: 0.2065 - accuracy: 0.9242 - val_loss: 0.7699 - val_accuracy: 0.7444
Epoch 7/100
57/57 [=====] - 4s 63ms/step - loss: 0.1788 - accuracy: 0.9317 - val_loss: 0.8212 - val_accuracy: 0.7278
Epoch 8/100
57/57 [=====] - 4s 65ms/step - loss: 0.1603 - accuracy: 0.9403 - val_loss: 0.8753 - val_accuracy: 0.7322
Epoch 9/100
57/57 [=====] - 5s 84ms/step - loss: 0.1474 - accuracy: 0.9450 - val_loss: 0.9144 - val_accuracy: 0.7267
Epoch 10/100
57/57 [=====] - 4s 64ms/step - loss: 0.1379 - accuracy: 0.9489 - val_loss: 0.9398 - val_accuracy: 0.7200
Epoch 11/100
57/57 [=====] - 4s 64ms/step - loss: 0.1284 - accuracy: 0.9539 - val_loss: 0.9605 - val_accuracy: 0.7244
Epoch 12/100
57/57 [=====] - 4s 72ms/step - loss: 0.1214 - accuracy: 0.9558 - val_loss: 0.9857 - val_accuracy: 0.7211
Epoch 13/100
57/57 [=====] - 4s 62ms/step - loss: 0.1172 - accuracy: 0.9581 - val_loss: 0.9933 - val_accuracy: 0.7244
Epoch 14/100
57/57 [=====] - 4s 65ms/step - loss: 0.1119 - accuracy: 0.9608 - val_loss: 1.0126 - val_accuracy: 0.7222

```

```

Epoch 29/100
57/57 [=====] - 5s 85ms/step - loss: 0.0779 - accuracy: 0.9667 - val_loss: 1.2123 - val_accuracy: 0.7133
Epoch 30/100
57/57 [=====] - 4s 63ms/step - loss: 0.0779 - accuracy: 0.9683 - val_loss: 1.2447 - val_accuracy: 0.7056
Epoch 31/100
57/57 [=====] - 4s 64ms/step - loss: 0.0769 - accuracy: 0.9678 - val_loss: 1.2336 - val_accuracy: 0.7100
Epoch 32/100
57/57 [=====] - 4s 76ms/step - loss: 0.0767 - accuracy: 0.9681 - val_loss: 1.2645 - val_accuracy: 0.6989
Epoch 33/100
57/57 [=====] - 4s 70ms/step - loss: 0.0753 - accuracy: 0.9681 - val_loss: 1.2644 - val_accuracy: 0.7044
Epoch 34/100
57/57 [=====] - 4s 64ms/step - loss: 0.0753 - accuracy: 0.9686 - val_loss: 1.2957 - val_accuracy: 0.6978
Epoch 35/100
57/57 [=====] - 5s 83ms/step - loss: 0.0749 - accuracy: 0.9689 - val_loss: 1.2877 - val_accuracy: 0.7033
Epoch 36/100
57/57 [=====] - 4s 64ms/step - loss: 0.0743 - accuracy: 0.9703 - val_loss: 1.3272 - val_accuracy: 0.6956
Epoch 37/100
57/57 [=====] - 4s 65ms/step - loss: 0.0747 - accuracy: 0.9692 - val_loss: 1.3149 - val_accuracy: 0.6989
Epoch 38/100
57/57 [=====] - 5s 81ms/step - loss: 0.0735 - accuracy: 0.9706 - val_loss: 1.3617 - val_accuracy: 0.6944
Epoch 39/100
57/57 [=====] - 4s 62ms/step - loss: 0.0754 - accuracy: 0.9686 - val_loss: 1.3454 - val_accuracy: 0.7011
Epoch 40/100
57/57 [=====] - 4s 65ms/step - loss: 0.0738 - accuracy: 0.9706 - val_loss: 1.3639 - val_accuracy: 0.6956
Epoch 41/100
57/57 [=====] - 5s 85ms/step - loss: 0.0744 - accuracy: 0.9689 - val_loss: 1.3752 - val_accuracy: 0.7011
Epoch 42/100
57/57 [=====] - 4s 64ms/step - loss: 0.0728 - accuracy: 0.9700 - val_loss: 1.3970 - val_accuracy: 0.7000
Epoch 57/100
57/57 [=====] - 5s 83ms/step - loss: 0.0690 - accuracy: 0.9614 - val_loss: 1.2946 - val_accuracy: 0.7189
Epoch 58/100
57/57 [=====] - 4s 65ms/step - loss: 0.0670 - accuracy: 0.9636 - val_loss: 1.2777 - val_accuracy: 0.7178
Epoch 59/100
57/57 [=====] - 4s 73ms/step - loss: 0.0671 - accuracy: 0.9628 - val_loss: 1.2835 - val_accuracy: 0.7200
Epoch 60/100
57/57 [=====] - 4s 63ms/step - loss: 0.0659 - accuracy: 0.9628 - val_loss: 1.2669 - val_accuracy: 0.7211
Epoch 61/100
57/57 [=====] - 4s 64ms/step - loss: 0.0654 - accuracy: 0.9633 - val_loss: 1.2921 - val_accuracy: 0.7211
Epoch 62/100
57/57 [=====] - 5s 84ms/step - loss: 0.0642 - accuracy: 0.9631 - val_loss: 1.2904 - val_accuracy: 0.7200
Epoch 63/100
57/57 [=====] - 4s 64ms/step - loss: 0.0636 - accuracy: 0.9639 - val_loss: 1.2928 - val_accuracy: 0.7200
Epoch 64/100
57/57 [=====] - 4s 65ms/step - loss: 0.0629 - accuracy: 0.9644 - val_loss: 1.2950 - val_accuracy: 0.7156
Epoch 65/100
57/57 [=====] - 4s 63ms/step - loss: 0.0620 - accuracy: 0.9628 - val_loss: 1.3150 - val_accuracy: 0.7144
Epoch 66/100
57/57 [=====] - 4s 64ms/step - loss: 0.0618 - accuracy: 0.9631 - val_loss: 1.3123 - val_accuracy: 0.7156
Epoch 67/100
57/57 [=====] - 5s 79ms/step - loss: 0.0608 - accuracy: 0.9622 - val_loss: 1.3171 - val_accuracy: 0.7156
Epoch 68/100
57/57 [=====] - 4s 64ms/step - loss: 0.0606 - accuracy: 0.9633 - val_loss: 1.3163 - val_accuracy: 0.7156
Epoch 69/100
57/57 [=====] - 5s 81ms/step - loss: 0.0599 - accuracy: 0.9628 - val_loss: 1.3146 - val_accuracy: 0.7156
Epoch 70/100
57/57 [=====] - 4s 65ms/step - loss: 0.0598 - accuracy: 0.9633 - val_loss: 1.3252 - val_accuracy: 0.7144
Epoch 74/100
57/57 [=====] - 4s 65ms/step - loss: 0.0598 - accuracy: 0.9633 - val_loss: 1.3252 - val_accuracy: 0.7144
Epoch 87/100
57/57 [=====] - 4s 65ms/step - loss: 0.0545 - accuracy: 0.9625 - val_loss: 1.3936 - val_accuracy: 0.7200
Epoch 88/100
57/57 [=====] - 4s 79ms/step - loss: 0.0544 - accuracy: 0.9622 - val_loss: 1.3936 - val_accuracy: 0.7211
Epoch 89/100
57/57 [=====] - 4s 65ms/step - loss: 0.0547 - accuracy: 0.9625 - val_loss: 1.3869 - val_accuracy: 0.7178
Epoch 90/100
57/57 [=====] - 4s 64ms/step - loss: 0.0540 - accuracy: 0.9622 - val_loss: 1.4087 - val_accuracy: 0.7189
Epoch 91/100
57/57 [=====] - 4s 76ms/step - loss: 0.0543 - accuracy: 0.9628 - val_loss: 1.4005 - val_accuracy: 0.7156
Epoch 92/100
57/57 [=====] - 4s 66ms/step - loss: 0.0536 - accuracy: 0.9608 - val_loss: 1.4215 - val_accuracy: 0.7178
Epoch 93/100
57/57 [=====] - 4s 65ms/step - loss: 0.0535 - accuracy: 0.9625 - val_loss: 1.4250 - val_accuracy: 0.7200
Epoch 94/100
57/57 [=====] - 5s 86ms/step - loss: 0.0534 - accuracy: 0.9622 - val_loss: 1.4316 - val_accuracy: 0.7189
Epoch 95/100
57/57 [=====] - 4s 65ms/step - loss: 0.0535 - accuracy: 0.9631 - val_loss: 1.4327 - val_accuracy: 0.7133
Epoch 96/100
57/57 [=====] - 4s 76ms/step - loss: 0.0536 - accuracy: 0.9622 - val_loss: 1.4234 - val_accuracy: 0.7200
Epoch 97/100
57/57 [=====] - 4s 63ms/step - loss: 0.0534 - accuracy: 0.9631 - val_loss: 1.4248 - val_accuracy: 0.7167
Epoch 98/100
57/57 [=====] - 4s 64ms/step - loss: 0.0525 - accuracy: 0.9625 - val_loss: 1.4477 - val_accuracy: 0.7222
Epoch 99/100
57/57 [=====] - 5s 79ms/step - loss: 0.0537 - accuracy: 0.9631 - val_loss: 1.4145 - val_accuracy: 0.7222
Epoch 100/100
57/57 [=====] - 4s 65ms/step - loss: 0.0528 - accuracy: 0.9631 - val_loss: 1.4456 - val_accuracy: 0.7233

```

### c. Pembagian Data 70 : 30

```
Epoch 1/100
50/50 [=====] - 4s 68ms/step - loss: 1.0385 - accuracy: 0.4863 - val_loss: 0.9273 - val_accuracy: 0.5793
Epoch 2/100
50/50 [=====] - 5s 94ms/step - loss: 0.6765 - accuracy: 0.7225 - val_loss: 0.6668 - val_accuracy: 0.7133
Epoch 3/100
50/50 [=====] - 3s 65ms/step - loss: 0.4194 - accuracy: 0.8302 - val_loss: 0.6336 - val_accuracy: 0.7459
Epoch 4/100
50/50 [=====] - 4s 72ms/step - loss: 0.2976 - accuracy: 0.8867 - val_loss: 0.6887 - val_accuracy: 0.7385
Epoch 5/100
50/50 [=====] - 5s 93ms/step - loss: 0.2277 - accuracy: 0.9140 - val_loss: 0.7578 - val_accuracy: 0.7348
Epoch 6/100
50/50 [=====] - 4s 72ms/step - loss: 0.1854 - accuracy: 0.9260 - val_loss: 0.8137 - val_accuracy: 0.7393
Epoch 7/100
50/50 [=====] - 4s 72ms/step - loss: 0.1594 - accuracy: 0.9387 - val_loss: 0.8770 - val_accuracy: 0.7348
Epoch 8/100
50/50 [=====] - 4s 72ms/step - loss: 0.1436 - accuracy: 0.9425 - val_loss: 0.9184 - val_accuracy: 0.7319
Epoch 9/100
50/50 [=====] - 4s 71ms/step - loss: 0.1290 - accuracy: 0.9476 - val_loss: 0.9610 - val_accuracy: 0.7333
Epoch 10/100
50/50 [=====] - 5s 94ms/step - loss: 0.1256 - accuracy: 0.9476 - val_loss: 0.9736 - val_accuracy: 0.7370
Epoch 11/100
50/50 [=====] - 4s 72ms/step - loss: 0.1182 - accuracy: 0.9571 - val_loss: 0.9810 - val_accuracy: 0.7370
Epoch 12/100
50/50 [=====] - 3s 66ms/step - loss: 0.1198 - accuracy: 0.9565 - val_loss: 0.9827 - val_accuracy: 0.7348
Epoch 13/100
50/50 [=====] - 4s 87ms/step - loss: 0.1062 - accuracy: 0.9622 - val_loss: 1.0125 - val_accuracy: 0.7422
Epoch 14/100
50/50 [=====] - 4s 72ms/step - loss: 0.1027 - accuracy: 0.9635 - val_loss: 1.0486 - val_accuracy: 0.7363

Epoch 29/100
50/50 [=====] - 5s 91ms/step - loss: 0.0735 - accuracy: 0.9698 - val_loss: 1.3348 - val_accuracy: 0.7030
Epoch 30/100
50/50 [=====] - 3s 60ms/step - loss: 0.0777 - accuracy: 0.9695 - val_loss: 1.3215 - val_accuracy: 0.7059
Epoch 31/100
50/50 [=====] - 4s 72ms/step - loss: 0.0738 - accuracy: 0.9689 - val_loss: 1.3290 - val_accuracy: 0.7007
Epoch 32/100
50/50 [=====] - 4s 71ms/step - loss: 0.0774 - accuracy: 0.9689 - val_loss: 1.3322 - val_accuracy: 0.7000
Epoch 33/100
50/50 [=====] - 4s 72ms/step - loss: 0.0742 - accuracy: 0.9695 - val_loss: 1.3131 - val_accuracy: 0.7037
Epoch 34/100
50/50 [=====] - 4s 79ms/step - loss: 0.0776 - accuracy: 0.9695 - val_loss: 1.3045 - val_accuracy: 0.7059
Epoch 35/100
50/50 [=====] - 4s 81ms/step - loss: 0.0742 - accuracy: 0.9698 - val_loss: 1.2610 - val_accuracy: 0.7141
Epoch 36/100
50/50 [=====] - 4s 71ms/step - loss: 0.0761 - accuracy: 0.9711 - val_loss: 1.2491 - val_accuracy: 0.7119
Epoch 37/100
50/50 [=====] - 4s 71ms/step - loss: 0.0724 - accuracy: 0.9692 - val_loss: 1.2173 - val_accuracy: 0.7141
Epoch 38/100
50/50 [=====] - 5s 93ms/step - loss: 0.0738 - accuracy: 0.9711 - val_loss: 1.2074 - val_accuracy: 0.7222
Epoch 39/100
50/50 [=====] - 4s 71ms/step - loss: 0.0702 - accuracy: 0.9683 - val_loss: 1.2005 - val_accuracy: 0.7148
Epoch 40/100
50/50 [=====] - 3s 69ms/step - loss: 0.0718 - accuracy: 0.9695 - val_loss: 1.1976 - val_accuracy: 0.7207
Epoch 41/100
50/50 [=====] - 4s 70ms/step - loss: 0.0685 - accuracy: 0.9692 - val_loss: 1.1989 - val_accuracy: 0.7156
Epoch 42/100
50/50 [=====] - 3s 66ms/step - loss: 0.0700 - accuracy: 0.9708 - val_loss: 1.1995 - val_accuracy: 0.7148

Epoch 57/100
50/50 [=====] - 4s 82ms/step - loss: 0.0595 - accuracy: 0.9660 - val_loss: 1.2272 - val_accuracy: 0.7207
Epoch 58/100
50/50 [=====] - 3s 66ms/step - loss: 0.0594 - accuracy: 0.9663 - val_loss: 1.2271 - val_accuracy: 0.7244
Epoch 59/100
50/50 [=====] - 4s 72ms/step - loss: 0.0591 - accuracy: 0.9657 - val_loss: 1.2302 - val_accuracy: 0.7215
Epoch 60/100
50/50 [=====] - 4s 86ms/step - loss: 0.0590 - accuracy: 0.9657 - val_loss: 1.2292 - val_accuracy: 0.7274
Epoch 61/100
50/50 [=====] - 3s 66ms/step - loss: 0.0581 - accuracy: 0.9660 - val_loss: 1.2366 - val_accuracy: 0.7259
Epoch 62/100
50/50 [=====] - 4s 88ms/step - loss: 0.0579 - accuracy: 0.9654 - val_loss: 1.2383 - val_accuracy: 0.7274
Epoch 63/100
50/50 [=====] - 4s 72ms/step - loss: 0.0573 - accuracy: 0.9663 - val_loss: 1.2415 - val_accuracy: 0.7252
Epoch 64/100
50/50 [=====] - 4s 72ms/step - loss: 0.0580 - accuracy: 0.9657 - val_loss: 1.2423 - val_accuracy: 0.7274
Epoch 65/100
50/50 [=====] - 4s 85ms/step - loss: 0.0573 - accuracy: 0.9660 - val_loss: 1.2412 - val_accuracy: 0.7289
Epoch 66/100
50/50 [=====] - 4s 71ms/step - loss: 0.0573 - accuracy: 0.9660 - val_loss: 1.2438 - val_accuracy: 0.7296
Epoch 67/100
50/50 [=====] - 3s 66ms/step - loss: 0.0560 - accuracy: 0.9660 - val_loss: 1.2504 - val_accuracy: 0.7267
Epoch 68/100
50/50 [=====] - 4s 85ms/step - loss: 0.0558 - accuracy: 0.9663 - val_loss: 1.2556 - val_accuracy: 0.7259
Epoch 69/100
50/50 [=====] - 4s 71ms/step - loss: 0.0553 - accuracy: 0.9657 - val_loss: 1.2673 - val_accuracy: 0.7274
Epoch 70/100
50/50 [=====] - 4s 71ms/step - loss: 0.0550 - accuracy: 0.9667 - val_loss: 1.2660 - val_accuracy: 0.7244
```

```
Epoch 87/100
50/50 [=====] - 4s 78ms/step - loss: 0.0514 - accuracy: 0.9663 - val_loss: 1.3376 - val_accuracy: 0.7230
Epoch 88/100
50/50 [=====] - 4s 71ms/step - loss: 0.0509 - accuracy: 0.9660 - val_loss: 1.3354 - val_accuracy: 0.7237
Epoch 89/100
50/50 [=====] - 4s 72ms/step - loss: 0.0513 - accuracy: 0.9663 - val_loss: 1.3472 - val_accuracy: 0.7244
Epoch 90/100
50/50 [=====] - 4s 80ms/step - loss: 0.0510 - accuracy: 0.9670 - val_loss: 1.3474 - val_accuracy: 0.7244
Epoch 91/100
50/50 [=====] - 4s 72ms/step - loss: 0.0509 - accuracy: 0.9657 - val_loss: 1.3584 - val_accuracy: 0.7230
Epoch 92/100
50/50 [=====] - 4s 73ms/step - loss: 0.0503 - accuracy: 0.9663 - val_loss: 1.3573 - val_accuracy: 0.7222
Epoch 93/100
50/50 [=====] - 4s 80ms/step - loss: 0.0513 - accuracy: 0.9635 - val_loss: 1.3462 - val_accuracy: 0.7230
Epoch 94/100
50/50 [=====] - 4s 71ms/step - loss: 0.0507 - accuracy: 0.9660 - val_loss: 1.3618 - val_accuracy: 0.7230
Epoch 95/100
50/50 [=====] - 4s 75ms/step - loss: 0.0502 - accuracy: 0.9654 - val_loss: 1.3671 - val_accuracy: 0.7230
Epoch 96/100
50/50 [=====] - 4s 84ms/step - loss: 0.0501 - accuracy: 0.9657 - val_loss: 1.3696 - val_accuracy: 0.7237
Epoch 97/100
50/50 [=====] - 4s 72ms/step - loss: 0.0500 - accuracy: 0.9657 - val_loss: 1.3758 - val_accuracy: 0.7222
Epoch 98/100
50/50 [=====] - 4s 81ms/step - loss: 0.0497 - accuracy: 0.9660 - val_loss: 1.3835 - val_accuracy: 0.7230
Epoch 99/100
50/50 [=====] - 4s 74ms/step - loss: 0.0497 - accuracy: 0.9663 - val_loss: 1.3824 - val_accuracy: 0.7207
Epoch 100/100
50/50 [=====] - 4s 72ms/step - loss: 0.0489 - accuracy: 0.9663 - val_loss: 1.3960 - val_accuracy: 0.7237
```

