

**KLASIFIKASI ASPEK ULASAN PELANGGAN RESTORAN
DI INDONESIA MENGGUNAKAN *NEURAL NETWORK***

SKRIPSI

Oleh :
RADINA MUTIA HAIRA
NIM. 210605110016



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025**

**KLASIFIKASI ASPEK ULASAN PELANGGAN RESTORAN
DI INDONESIA MENGGUNAKAN *NEURAL NETWORK***

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
RADINA MUTIA HAIRA
NIM. 210605110016

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025**

HALAMAN PERSETUJUAN

**KLASIFIKASI ASPEK ULASAN PELANGGAN RESTORAN
DI INDONESIA MENGGUNAKAN *NEURAL NETWORK***

SKRIPSI

Oleh :
RADINA MUTIA HAIRA
NIM. 210605110016

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 05 Juni 2025

Pembimbing I,


Dr. Cahyo Crysdiyan, M.Cs
NIP. 19740424 200901 1 008

Pembimbing II,


Okta Qomaruddin Aziz, M.Kom
NIP. 19911019 201903 1 013

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Ir. Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

**KLASIFIKASI ASPEK ULASAN PELANGGAN RESTORAN
DI INDONESIA MENGGUNAKAN *NEURAL NETWORK***

SKRIPSI

Oleh :

RADINA MUTIA HAIRA

NIM. 210605110016

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 16 Juni 2025

Susunan Dewan Penguji

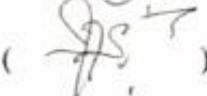
Ketua Penguji : Dr. M. Amin Hariyadi, M.T
NIP. 19670018 200501 1 001

Anggota Penguji I : Khadijah Fahmi Hayati Holle, M.Kom
NIP. 19900626 202203 2 002

Anggota Penguji II : Dr. Cahyo Crysdian, M.CS
NIP. 19740424 200901 1 008

Anggota Penguji III : Okta Qomaruddin Aziz, M.Kom
NIP. 19911019 201903 1 013

()

()

()

()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Ir. Fachrul Kurniawan, M.MT., IPU

NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Radina Mutia Haira
NIM : 210605110016
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : Klasifikasi Aspek Ulasan Pelanggan Restoran
di Indonesia Menggunakan *Neural Network*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 16 Juni 2025
Yang membuat pernyataan,



Radina Mutia Haira
NIM.210605110016

MOTTO

“Semua dalam hidup itu ada ujiannya, ada prosesnya, jadi terus lah berusaha dan jangan pernah berhenti bersyukur”

HALAMAN PERSEMBAHAN

Dengan memanjatkan puji dan syukur ke hadirat Allah SWT atas limpahan rahmat, nikmat, dan pertolongan-Nya, penulis dapat menyelesaikan penyusunan skripsi ini. Shalawat dan salam semoga selalu tercurah kepada Nabi Muhammad SAW, teladan mulia sepanjang zaman. Sebagai bentuk rasa syukur, penulis persembahkan skripsi ini kepada :

1. Almh. mamah, Hj. Tina Rofiqoh, S.H., tercinta, yang semasa hidupnya senantiasa memberikan doa, dukungan, kasih sayang serta pelajaran berharga. Nilai-nilai kehidupan yang beliau wariskan menjadi sumber kekuatan bagi penulis untuk terus tegar dan melangkah hingga saat ini.
2. Abah, H. Akhmad Haitami, SP.MP., dan ibu sambung penulis, Dewi, yang dengan penuh kasih sayang senantiasa mendoakan, menguatkan, dan memberikan semangat dalam setiap perjalanan yang penulis lalui.
3. Dosen pembimbing, Bapak Dr. Cahyo Crysdiyan, M.Cs dan Bapak Okta Qomaruddin Aziz, M.Kom., dengan sabar membimbing, mengarahkan, serta memberikan banyak masukan berharga selama proses penyusunan skripsi ini.
4. Dosen penguji, Bapak Dr. M. Amin Hariyadi, M.T dan Ibu Khadijah Fahmi Hayati Holle, M.Kom., yang telah memberikan saran dan masukan berharga demi perbaikan skripsi ini.
5. Diri penulis sendiri, yang telah bekerja keras, tetap bertahan dan tidak menyerah di tengah berbagai tantangan, rasa lelah, dan keraguan selama

proses penulisan skripsi ini. Terima kasih kepada diri sendiri yang sudah sabar, terus berusaha, dan selalu yakin bahwa setiap langkah kecil sangat berarti untuk menuju penyelesaian.

Penulis juga menyampaikan terima kasih yang tulus kepada semua pihak yang telah memberikan bantuan, dukungan, dan doa selama proses pengerjaan skripsi ini. Semoga segala kebaikan yang telah diberikan mendapatkan balasan terbaik dari Allah SWT.

KATA PENGANTAR

Assalammualaikum Warrahmatullahi Wabbarakatuh

Segala puji hanya milik Allah SWT, Tuhan seluruh alam, yang telah memberikan rahmat dan karunia-Nya sehingga penulis diberikan kesehatan dan kelancaran dalam menyelesaikan skripsi yang berjudul “Klasifikasi Aspek Ulasan Pelanggan Restoran di Indonesia Menggunakan *Neural Network*.” Shalawat dan salam semoga terus tercurahkan kepada Nabi Muhammad SAW.

Penulis mengungkapkan rasa terima kasih kepada seluruh pihak yang telah memberikan dukungan, bimbingan, serta bantuan selama proses penyusunan skripsi ini kepada :

1. Prof. Dr. H. M. Zainuddin, M.A., selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Hariani, M.Si., selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, ST., M.MT., IPU, selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Dr. Cahyo Crys dian, M.Cs., selaku Dosen Pembimbing I, yang telah membimbing dan memberikan arahan selama proses penulisan skripsi ini.
5. Okta Qomaruddin Aziz, M.Kom., selaku Dosen Pembimbing II, yang telah memberikan masukan dan saran yang sangat membantu dalam penulisan skripsi ini.

6. Dr. M. Amin Hariyadi, M.T., selaku Ketua Penguji dan Khadijah Fahmi Hayati Holle, M.Kom., selaku Dosen Penguji II, yang telah memberikan kritik dan saran untuk memperbaiki dan menyempurnakan skripsi ini.
7. Dr. Ririen Kusumawati, M.Kom., selaku Dosen Wali, yang telah mendampingi penulis dalam proses akademik, memberikan arahan, serta membantu dalam penyusunan rencana studi selama masa perkuliahan.
8. Seluruh Dosen, Admin, dan tenaga laboratorium di Program Studi Teknik Informatika atas segala bantuan, pelayanan, dan dukungan selama proses studi penulis.
9. Almh. mamah tercinta, Hj. Tina Rofiqoh, S.H., abah, H. Akhmad Haitami SP.MP., dan ibu sambung penulis, ibu Dewi yang dengan kasih sayang dan doa yang tiada henti telah menjadi sumber kekuatan, semangat, dan motivasi bagi penulis dalam menyelesaikan studi dan penulisan skripsi ini.
10. Muhammad Adlany Haira, adik laki-laki penulis, yang selalu menyemangati dengan candaan dan tawanya, membuat suasana menjadi lebih hangat hingga skripsi ini selesai.
11. Keluarga besar penulis yang terus memberikan dukungan, doa, dan semangat tiada henti, menjadi sumber kekuatan penulis selama penulisan skripsi ini.
12. Eka Mira Novita Subroto, sahabat seperjuangan dalam bimbingan yang senantiasa mendampingi, memberi kekuatan, saran, dan semangat selama proses perkuliahan sampai penyusunan skripsi ini selesai.

13. Sahabat-sahabat terdekat penulis, Siti Julfa yang selalu menjadi tempat berbagi keluh kesah, Heny Rimadana, Shafira Halmahera, Vivin Octavia Cahyani dan Putri Atina Zahra, yang senantiasa menemani dan memberikan semangat di setiap langkah penyusunan skripsi.
14. Kak Fahrendra Khoirul Ihtada, yang telah banyak membantu melalui arahan, saran, serta pengalaman yang dibagikan, sehingga sangat berperan dalam kelancaran penyusunan skripsi ini.
15. Teman-teman Kost Muslimah Rayhan, yaitu Laili, Tasya, Reghina, dan Shakilla, yang senantiasa memberikan semangat, dukungan, serta doa kepada penulis selama proses penyusunan skripsi ini.
16. Teman-Teman “ASTER” Teknik Informatika angkatan 2021, yang telah menemani penulis atas kebersamaan, semangat, serta kenangan indah yang akan selalu menjadi bagian berharga dari perjalanan studi ini.
17. Semua pihak yang telah memberikan kebaikan dalam berbagai bentuk selama proses penyusunan skripsi ini, meskipun tidak bisa disebutkan satu per satu, penulis mengucapkan terima kasih yang sebesar-besarnya.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna. Semoga tulisan ini tetap bisa bermanfaat bagi yang membaca, dan jadi langkah awal untuk terus belajar dan berkembang ke depannya.

Wassalamualaikum Warrahmatullahi Wabbarakatuh

Malang, 16 Juni 2025

Penulis

DAFTAR ISI

HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	ix
DAFTAR ISI	xii
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xvi
ABSTRAK	xvii
ABSTRACT	xviii
البحث مستخلص	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah	5
1.3 Tujuan Penelitian	5
1.4 Batasan Masalah	5
1.5 Manfaat Penelitian	5
BAB II KAJIAN PUSTAKA	7
2.1 Pemodelan Aspek Ulasan.....	7
2.2 Klasifikasi Aspek Ulasan	8
BAB III DESAIN DAN IMPLEMENTASI	13
3.1 Akuisisi Data.....	13
3.2 Desain Sistem.....	16
3.2.1 <i>Preprocessing</i>	17
3.2.1.1 <i>Data Cleaning</i>	18
3.2.1.2 <i>Case Folding</i>	19
3.2.1.3 <i>Tokenizing</i>	20
3.2.1.4 <i>Punctuation Removal</i>	21
3.2.1.5 <i>Stopword Removal</i>	22
3.2.1.6 <i>Stemming</i>	23
3.2.2 <i>Word2Vec</i>	24
3.2.3 <i>Neural Network</i>	35
3.2.3.1 <i>Proses Training</i>	39
BAB IV UJI COBA DAN PEMBAHASAN	44
4.1 Persiapan Data Uji	44
4.2 Skenario Pengujian	46
4.2 Hasil Uji Coba.....	52
4.2.1 Model P1	52
4.2.2 Model P2	54
4.2.3 Model P3	56

4.2.4 Model Q1	58
4.2.5 Model Q2	60
4.2.6 Model Q3	62
4.2.7 Model R1	64
4.2.8 Model R2	66
4.2.9 Model R3	68
4.3 Pembahasan.....	70
BAB V KESIMPULAN DAN SARAN.....	96
5.1 Kesimpulan	96
5.2 Saran.....	97
DAFTAR PUSTAKA	

DAFTAR GAMBAR

Gambar 3.1 Desain sistem	16
Gambar 3.2 Alur tahapan <i>preprocessing</i>	17
Gambar 3.3 <i>Flowchart</i> tahap <i>Data Cleaning</i>	18
Gambar 3.4 <i>Flowchart</i> tahap <i>Case Folding</i>	19
Gambar 3.5 <i>Flowchart</i> tahap <i>Tokenizing</i>	20
Gambar 3.6 <i>Flowchart</i> tahap <i>Punctuational Removal</i>	21
Gambar 3.7 <i>Flowchart</i> tahap <i>Stopword Removal</i>	22
Gambar 3.8 <i>Flowchart</i> tahap <i>Stemming</i>	23
Gambar 3.9 Konsep dasar <i>Word2Vec</i> Skip-Gram	25
Gambar 3.10 Arsitektur <i>Word2Vec</i> Skip-Gram.....	26
Gambar 3.11 Grafik perubahan nilai <i>loss</i> pada <i>training Word2vec</i>	32
Gambar 3.12 Diagram alur tahapan model <i>Word2Vec</i> Skip-Gram	34
Gambar 3.13 Desain topologi <i>Neural Network</i>	35
Gambar 3.14 Diagram alur proses <i>training</i> dalam <i>Neural Network</i>	43
Gambar 4.1 Distribusi jumlah dataset berdasarkan label.....	45
Gambar 4.2 Diagram alur proses <i>testing</i> dalam <i>Neural Network</i>	49
Gambar 4.3 Grafik <i>cost</i> proses <i>training</i> pada model P1	53
Gambar 4.4 <i>Confusion matrix</i> model P1.....	53
Gambar 4.5 Hasil performa model P1	54
Gambar 4.6 Grafik <i>cost</i> proses <i>training</i> pada model P2	55
Gambar 4.7 <i>Confusion matrix</i> model P2.....	55
Gambar 4.8 Hasil performa model P2	56
Gambar 4.9 Grafik <i>cost</i> proses <i>training</i> pada model P3	57
Gambar 4.10 <i>Confusion matrix</i> model P3.....	57
Gambar 4.11 Hasil performa model P3	58
Gambar 4.12 Grafik <i>cost</i> proses <i>training</i> pada model Q1	59
Gambar 4.13 <i>Confusion matrix</i> model Q1	59
Gambar 4.14 Hasil performa model Q1	60
Gambar 4.15 Grafik <i>cost</i> proses <i>training</i> pada model Q2	61
Gambar 4.16 <i>Confusion matrix</i> model Q2.....	61
Gambar 4.17 Hasil performa model Q2.....	62
Gambar 4.18 Grafik <i>cost</i> proses <i>training</i> pada model Q3	63
Gambar 4.19 <i>Confusion matrix</i> model Q3	63
Gambar 4.20 Hasil performa model Q3.....	64
Gambar 4.21 Grafik <i>cost</i> proses <i>training</i> pada model R1.....	65
Gambar 4.22 <i>Confusion matrix</i> model R1	65
Gambar 4.23 Hasil performa model R1	66
Gambar 4.24 Grafik <i>cost</i> proses <i>training</i> pada model R2.....	67
Gambar 4.25 <i>Confusion matrix</i> model R2	67
Gambar 4.26 Hasil performa model R2	68

Gambar 4.27 Grafik <i>cost</i> proses <i>training</i> pada model R3.....	69
Gambar 4.28 <i>Confusion matrix</i> model R3	69
Gambar 4.29 Hasil performa model R3	70
Gambar 4.30 Grafik hasil evaluasi skenario pengujian	74
Gambar 4.31 Grafik hasil evaluasi berdasarkan <i>node hidden layer</i> 1	75
Gambar 4.32 Grafik hasil evaluasi berdasarkan nilai <i>learning rate</i>	75
Gambar 4.33 Hasil akurasi setiap <i>fold</i> pada model P1	80
Gambar 4.34 Hasil akurasi setiap <i>fold</i> pada model P2	81
Gambar 4.35 Hasil akurasi setiap <i>fold</i> pada model P3	82
Gambar 4.36 Hasil akurasi setiap <i>fold</i> pada model Q1	83
Gambar 4.37 Hasil akurasi setiap <i>fold</i> pada model Q2.....	84
Gambar 4.38 Hasil akurasi setiap <i>fold</i> pada model Q3.....	85
Gambar 4.39 Hasil akurasi setiap <i>fold</i> pada model R1	86
Gambar 4.40 Hasil akurasi setiap <i>fold</i> pada model R2	87
Gambar 4.41 Hasil akurasi setiap <i>fold</i> pada model R3	88
Gambar 4.42 <i>Word cloud</i> kata dominan pada kelas kualitas pelayanan.....	90
Gambar 4.43 <i>Word cloud</i> kata dominan pada kelas makanan	91
Gambar 4.44 <i>Word cloud</i> kata dominan pada kelas suasana	92

DAFTAR TABEL

Tabel 3.1 Dataset ulasan pelanggan restoran	16
Tabel 3.2 Dataset setelah proses data <i>cleaning</i>	19
Tabel 3.3 Dataset hasil proses <i>case folding</i>	20
Tabel 3.4 Dataset hasil proses <i>tokenizing</i>	21
Tabel 3.5 Dataset hasil proses <i>punctuational removal</i>	22
Tabel 3.6 Dataset hasil proses <i>stopword removal</i>	23
Tabel 3.7 Dataset hasil proses <i>stemming</i>	24
Tabel 3.8 Hasil representasi vektor setiap kata dalam <i>Word2vec</i> Skip-Gram.....	33
Tabel 3.9 Hasil representasi vektor <i>Word2vec</i> Skip-Gram.....	34
Tabel 4.1 Distribusi ulasan berdasarkan label	44
Tabel 4.2 Skenario pengujian	47
Tabel 4.3 Rasio pembagian data dan kriteria pemberhentian pelatihan	48
Tabel 4.4 Hasil <i>confusion matrix</i>	50
Tabel 4.5 <i>Confusion matrix</i> kategori kualitas pelayanan.....	50
Tabel 4.6 <i>Confusion matrix</i> kategori makanan	50
Tabel 4.7 <i>Confusion matrix</i> kategori suasana	51
Tabel 4.8 Hasil <i>training</i> dari skenario pengujian.....	71
Tabel 4.9 Hasil evaluasi model dari skenario pengujian	73
Tabel 4.10 <i>Confusion matrix</i> model Q1 kategori kualitas pelayanan	77
Tabel 4.11 <i>Confusion matrix</i> model Q1 kategori makanan	77
Tabel 4.12 <i>Confusion matrix</i> model Q1 kategori suasana	77

ABSTRAK

Haira, Radina Mutia. 2025. **Klasifikasi Aspek Ulasan Pelanggan Restoran di Indonesia Menggunakan *Neural Network***. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Cahyo Crys dian, M.Cs (II) Okta Qomaruddin Aziz, M.Kom.

Kata kunci: *Ulasan Restoran, Klasifikasi Aspek, Neural Network.*

Industri kuliner di Indonesia terus berkembang, sehingga jumlah ulasan pelanggan terhadap restoran pun meningkat. Ulasan-ulasan tersebut dapat dimanfaatkan untuk mengetahui aspek-aspek yang dinilai oleh pelanggan, namun masih sedikit penelitian yang secara khusus mengklasifikasikan ulasan berdasarkan aspek tertentu. Penelitian ini bertujuan untuk membangun model klasifikasi aspek ulasan pelanggan restoran di Indonesia menggunakan metode *Neural Network*. Data ulasan diolah melalui proses *preprocessing* dan kemudian diubah menjadi representasi numerik menggunakan *Word2Vec* Skip-Gram. Penelitian ini mengukur efektivitas *Neural Network* dalam mengklasifikasikan ulasan ke dalam 3 aspek utama, yaitu kualitas pelayanan, makanan, dan suasana. Pengujian dilakukan terhadap 9 skenario model dengan kombinasi jumlah *node* pada *hidden layer* 1 dan nilai *learning rate* yang berbeda. Hasil pengujian menunjukkan bahwa model Q1 dengan 50 *node* dan *learning rate* 0,1 menghasilkan akurasi tertinggi sebesar 91,71% pada data *testing*. Sementara itu, evaluasi menggunakan *10-fold cross validation* terhadap seluruh model menunjukkan bahwa model Q3 dengan 50 *node* dan *learning rate* 0,001 memperoleh rata-rata akurasi tertinggi sebesar 84,77% dan menunjukkan performa yang stabil. Hasil penelitian ini membuktikan bahwa metode *Neural Network* dengan pengaturan *hyperparameter* yang tepat efektif dalam mengklasifikasikan aspek ulasan pelanggan restoran secara akurat dan konsisten.

ABSTRACT

Haira, Radina Mutia. 2025. **Classification of Aspects of Restaurant Customer Reviews in Indonesia Using Neural Networks.** Undergraduate Thesis. Informatics Engineering Study Program Faculty of Science and Technology Maulana Malik Ibrahim State Islamic University Malang. Supervisor: (I) Dr. Cahyo Crys dian, M.Cs (II) Okta Qomaruddin Aziz, M.Kom.

Key words: *Restaurant Reviews, Aspect Classification, Neural Network.*

The culinary industry in Indonesia continues to grow, resulting in an increasing number of customer reviews of restaurants. These reviews can be utilized to find out the aspects assessed by customers, but there are still few studies that specifically classify reviews based on certain specified aspects. This study aims to build a classification model of aspects of restaurant customer reviews in Indonesia using the Neural Network method. The review data is processed through preprocessing and then converted into numerical representation using Word2Vec Skip-Gram. This research quantifies the effectiveness of Neural Networks in classifying reviews into three main aspects, namely service quality, food, and atmosphere. Tests were conducted on 9 model scenarios with a combined number of nodes in hidden layer 1 and different learning rate values. The test results show that the Q1 model with 50 nodes and a learning rate of 0.1 produces the highest accuracy of 91.71% on the testing data. Meanwhile, evaluation using 10-fold cross-validation of all models showed that model Q3 using 50 nodes and learning rate 0.001 obtained the highest average accuracy of 84.77% and showed stable performance. The results of this study prove that the Neural Network method with the correct hyperparameter settings is effective in classifying aspects of restaurant customer reviews accurately and consistently.

مستخلص البحث

حيرا، رادينا موتيا 2025. تصنيف جوانب مراجعات عملاء المطاعم في إندونيسيا باستخدام الشبكات العصبية. البحث اجلامعي. قسم اهلندسة املعلوماتية، كلية العلوم والتكنولوجيا. جامعة مولان مالك إبراهيم الإسلامية احلكومية مالنج. المشرف الأول: د. كاهيو كريسديان، ماجستير. ملشرف الثاني: أوكتا قمر الدين عزيز، ماجستير.

الكلمات الرئيسية : مراجعات المطاعم، تصنيف الجوانب، الشبكة العصبية

تستمر صناعة الطهي في إندونيسيا في النمو، مما يؤدي إلى زيادة عدد مراجعات العملاء للمطاعم. يمكن الاستفادة من هذه المراجعات لمعرفة الجوانب التي يقيّمها العملاء، ولكن لا يزال هناك عدد قليل من الدراسات التي تصنف المراجعات على وجه التحديد بناءً على جوانب معينة محددة. تهدف هذه الدراسة إلى بناء نموذج تصنيف لجوانب مراجعات عملاء المطاعم في إندونيسيا باستخدام طريقة الشبكة العصبية. تتم معالجة بيانات المراجعات من خلال المعالجة المسبقة ثم يتم تحويلها إلى تمثيل رقمي باستخدام *Word2Vec Skip-Gram* يقيس هذا البحث فعالية الشبكات العصبية في تصنيف المراجعات إلى ثلاثة جوانب رئيسية، وهي جودة الخدمة والطعام والجو العام. أُجريت الاختبارات على 9 سيناريوهات للنموذج مع عدد مشترك من العقد في الطبقة المخفية 1 وقيم مختلفة لمعدل التعلم. تُظهر نتائج الاختبار أن نموذج Q1 الذي يحتوي على 50 عقدة ومعدل تعلم 0.1 ينتج أعلى دقة بنسبة 91.71% على بيانات الاختبار. وفي الوقت نفسه، أظهر التقييم باستخدام 10 أضعاف التحقق التبادلي لجميع النماذج أن النموذج Q3 باستخدام 50 عقدة ومعدل تعلم 0.001 حصل على أعلى متوسط دقة بنسبة 84.77% وأظهر أداءً مستقرًا. وتثبت نتائج هذه الدراسة أن طريقة الشبكة العصبية مع إعدادات المعلمة الفائقة الصحيحة فعالة في تصنيف جوانب مراجعات عملاء المطاعم بدقة واتساق.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam industri kuliner, restoran di Indonesia mengalami pertumbuhan yang pesat dan menunjukkan peningkatan signifikan di berbagai wilayah (Wisnu Brata et al., 2024). Pertumbuhan ini sejalan dengan masyarakat yang semakin mengutamakan pengalaman kuliner yang berkualitas. Di tengah persaingan yang ketat, restoran harus mampu memberikan pelayanan yang memuaskan untuk menarik perhatian pelanggan. Ulasan pelanggan terhadap berbagai aspek restoran menjadi faktor utama dalam memengaruhi tingkat kepuasan pelanggan yang mana dapat berdampak pada reputasi dan keberhasilan restoran itu sendiri.

Seiring dengan pertumbuhan industri kuliner, kebutuhan untuk memahami ulasan pelanggan restoran juga semakin meningkat. Menurut data yang dikeluarkan oleh Badan Pusat Statistika (BPS) jumlah usaha penyedia makanan minuman di Indonesia mengalami peningkatan 20,76% pada tahun 2022 dibanding dengan tahun 2021, tetapi tahun 2021 sempat mengalami penurunan 19,58% dibanding tahun 2020 dan kembali meningkat secara drastis 36,34% pada tahun 2019 (BPS Penyedia Makan Minum, n.d.). Riset yang dilakukan oleh Departemen Riset Statista, pendapatan industri makanan di Indonesia diprediksi akan mengalami kenaikan yang berkelanjutan antara tahun 2024 dan 2029. Pada akhir periode tersebut, diperkirakan industri akan mencapai pendapatan 85,1 miliar dolar AS

dengan peningkatan sebesar 34,02% selama sembilan tahun kedepan (Statista Research Department, 2024).

Pelanggan restoran saat ini semakin sering mengandalkan ulasan yang tersedia di berbagai *platform* digital. *Platform* seperti *Google Review*, *TripAdvisor* dan media sosial lainnya menyediakan ruang untuk pelanggan berbagi pengalaman mereka secara terbuka. Dengan terus berkembangnya tren penggunaan *platform* digital menjadikan jumlah ulasan yang ditulis oleh pelanggan semakin meningkat. Peningkatan ini menunjukkan bahwa pelanggan lebih aktif dalam memberikan *feedback*, sehingga membantu restoran dalam memahami dan meningkatkan pelayanan mereka.

Dengan meningkatnya jumlah ulasan yang tersedia di platform digital, pemanfaatan informasi yang terkandung di dalamnya membutuhkan pendekatan yang terstruktur, terutama dalam proses pengkategorian aspek ulasan. Dalam konteks layanan restoran, 3 aspek utama yang paling sering disoroti pelanggan adalah kualitas pelayanan, makanan, dan suasana. Tiga kategori ini dipilih berdasarkan temuan dari beberapa studi yang menunjukkan bahwa aspek tersebut sering kali menjadi sumber keluhan maupun pujian pelanggan, serta menjadi fokus utama dalam rekomendasi perbaikan. Selain itu, karakteristik data ulasan yang digunakan dalam penelitian ini juga menguatkan hal tersebut, di mana sebagian besar ulasan mengarah pada ketiga aspek tersebut. Pengelompokan yang tepat akan memudahkan restoran dalam mengidentifikasi kekuatan dan kelemahan mereka, serta mengarahkan strategi perbaikan yg lebih efektif (Zhafirah & Silitonga, 2023).

Namun, mengelompokkan ulasan secara manual ke dalam 3 kategori tersebut bisa memakan banyak waktu dan tenaga. Selain itu, metode manual cenderung tidak konsisten karena pemahaman tiap pelanggan terhadap isi ulasan bisa berbeda-beda. Perbedaan persepsi ini menimbulkan subjektivitas yang dapat menyebabkan hasil klasifikasi menjadi kurang akurat dan efisien (Hidayat et al, 2021). Oleh karena itu, diperlukan adanya sistem klasifikasi yang dapat berjalan secara otomatis. Sistem ini memungkinkan proses klasifikasi dilakukan dengan cepat dan akurat, sehingga restoran dapat membuat keputusan yang lebih tepat dan memperbaiki strategi operasional mereka.

Penelitian ini bertujuan mengklasifikasikan ulasan pelanggan restoran ke dalam 3 kategori utama, yaitu kualitas pelayanan, makanan dan suasana. Hal ini dilakukan karena masih terbatasnya penelitian yang secara khusus memisahkan ketiga kategori ini dalam konteks ulasan pelayanan restoran. Sebagian besar penelitian sebelumnya lebih terfokus pada klasifikasi sentimen positif maupun negatif, dari ulasan pelanggan. Permadi (2020) meneliti penilaian pengunjung dalam analisis sentimen pada ulasan restoran di Singapura menggunakan metode *Naive Bayes* dan mencapai akurasi sebesar 73%. Penelitian serupa juga dilakukan oleh Julianto et al (2022) terkait analisis sentimen ulasan restoran menggunakan SVM dengan hasil penelitian menunjukkan akurasi 93% dan *f1-score* 93%.

Menurut firman Allah SWT. dalam Surah An-Nur ayat 35:

اللَّهُ نُورُ السَّمَوَاتِ وَالْأَرْضِ ۗ مِثْلُ نُورِهِ ۖ كَمَشْكُوَةٍ فِيهَا مِصْبَاحٌ الْمِصْبَاحُ فِي زُجَاجَةٍ الزُّجَاجَةُ كَأَنَّهَا كَوْكَبٌ دُرِّيٌّ يُوقَدُ مِنْ شَجَرَةٍ مُبْرَكَةٍ زَيْتُونَةٍ لَا شَرْقِيَّةٍ وَلَا غَرْبِيَّةٍ يَكَادُ زَيْتُهَا يُضِيءُ وَلَوْ لَمْ تَمْسَسْهُ نَارٌ نُورٌ عَلَى نُورٍ يَهْدِي اللَّهُ لِنُورِهِ ۗ مَنْ يَشَاءُ وَيَضْرِبُ اللَّهُ الْأَمْثَالَ لِلنَّاسِ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ ﴿٣٥﴾

“Allah (pemberi) cahaya (kepada) langit dan bumi. Perumpamaan cahaya-Nya, seperti sebuah lubang yang tidak tembus, yang di dalamnya ada pelita besar. Pelita itu di dalam tabung kaca, (dan) tabung kaca itu bagaikan bintang yang berkilauan, yang dinyalakan dengan minyak dari pohon yang diberkahi, (yaitu) pohon zaitun yang tumbuh tidak di timur dan tidak pula di barat, yang minyaknya (saja) hampir-hampir menerangi, walaupun tidak disentuh api. Cahaya di atas cahaya (berlapis-lapis), Allah memberi petunjuk kepada cahaya-Nya bagi orang yang Dia kehendaki, dan Allah membuat perumpamaan-perumpamaan bagi manusia. Dan Allah Maha Mengetahui segala sesuatu.”(Q.S An Nur:35).

Para ulama seperti Ibn Katsir dan Al-Qurtubi menafsirkan frasa “cahaya di atas cahaya” dalam ayat ini sebagai tingkatan-tingkatan hidayah yang diberikan Allah SWT. secara bertahap dan berlapis, tergantung pada usaha dan keinginan seseorang untuk memperoleh petunjuk dan kebenaran. Dalam penelitian, konsep ini bisa dianalogikan dengan proses klasifikasi ulasan berdasarkan kategori menggunakan *Neural Network*. Metode ini bekerja melalui sejumlah lapisan yang saling terhubung, di mana setiap lapisan mempelajari informasi dari lapisan sebelumnya secara bertahap. Pendekatan ini memungkinkan sistem mengenali pola-pola tersembunyi dalam data yang mungkin tidak terlihat secara langsung. Sebagaimana hidayah diberikan secara berlapis, *Neural Network* juga memproses data secara bertahap untuk menghasilkan klasifikasi yang lebih tepat. Kemampuan mengenali struktur data yang kompleks dan beragam menjadikan *Neural Network* lebih unggul dibandingkan metode tradisional (Zheng, 2024).

Berdasarkan pertimbangan tersebut, penelitian ini diberi judul “Klasifikasi Aspek Ulasan Pelanggan Restoran di Indonesia Menggunakan *Neural Network*”, yang bertujuan untuk menghasilkan sistem klasifikasi otomatis yang mampu membantu pelaku industri makanan dalam memahami fokus dan persepsi pelanggan secara lebih mendalam melalui ulasan yang diberikan.

1.2 Pernyataan Masalah

Seberapa efektif metode *Neural Network* dalam mengklasifikasi ulasan pelanggan restoran kedalam 3 aspek utama, yaitu kualitas pelayanan, makanan dan suasana dengan pengaturan *hyperparameter* yang tepat untuk mencapai akurasi optimal?

1.3 Tujuan Penelitian

Mengukur efektivitas metode *Neural Network* dalam mengklasifikasikan ulasan pelanggan restoran secara akurat berdasarkan 3 aspek utama, dengan mempertimbangkan pengaruh pengaturan *hyperparameter* terhadap hasil akurasi model.

1.4 Batasan Masalah

- a. Data penelitian ini diperoleh dari komentar pelanggan di *Google Review* yang berasal dari restoran di 38 provinsi di Indonesia. Setiap provinsi diwakili oleh 1 hingga 3 restoran yang dipilih berdasarkan prioritas restoran yang memiliki rating 4 hingga 5 bintang, tergantung pada jumlah ulasan yang tersedia. Label ditentukan berdasarkan survei dari responden umum.
- b. Bahasa yang dipakai dalam penelitian ini terbatas pada Bahasa Indonesia.
- c. Penelitian ini hanya menganalisis ulasan yang membahas 1 aspek secara spesifik (*single label*), yaitu kualitas pelayanan, makanan dan suasana.

1.5 Manfaat Penelitian

Melalui penelitian ini, diharapkan dapat memberikan beberapa manfaat sebagai berikut. :

- a. Bagi industri restoran, dapat membantu mereka dalam mengevaluasi kualitas pelayanan, makanan dan suasana. Dengan begitu, restoran dapat mengetahui aspek mana yang perlu diperbaiki untuk meningkatkan kepuasan pelanggan secara keseluruhan.
- b. Bagi pengelola restoran, memudahkan dalam mengidentifikasi penilaian mengenai kualitas pelayanan, kesesuaian makanan serta kenyamanan suasana. Hal ini memungkinkan pengelola untuk fokus pada aspek yang memerlukan perhatian khusus dan melakukan perbaikan yang tepat.

BAB II

KAJIAN PUSTAKA

2.1 Pemodelan Aspek Ulasan

Industri kuliner saat ini semakin berkembang seiring dengan meningkatnya harapan pelanggan akan standar pelayanan restoran. Pelanggan cenderung memberikan ulasan berdasarkan berbagai aspek, mulai dari kualitas pelayanan, kesesuaian makanan, maupun suasana restoran. Ketiga aspek tersebut berperan penting dalam membentuk persepsi dan kepuasan pelanggan terhadap suatu restoran.

Setiap aspek tersebut memiliki karakteristik tersendiri dalam ulasan pelanggan. Aspek kualitas pelayanan umumnya mencakup komentar tentang keramahan staf, kecepatan penyajian, dan responsivitas layanan. Aspek makanan meliputi ulasan terkait rasa, porsi, dan penyajian hidangan. Sementara itu, aspek suasana biasanya merujuk pada kenyamanan tempat, kebersihan lingkungan, desain interior, dan suasana umum restoran.

Beberapa penelitian menunjukkan bahwa masing-masing aspek tersebut memiliki kontribusi signifikan terhadap loyalitas pelanggan. Riset yang dilakukan oleh Hurnis et al (2023) menunjukkan bahwa kecepatan dan ketepatan dalam penyediaan layanan sangat memengaruhi persepsi pelanggan terhadap kepuasan dan loyalitas mereka. Hal tersebut sejalan dengan hasil penelitian dari Grace et al (2022) yang menekankan bahwa kepuasan pelanggan memiliki hubungan langsung dengan keputusan mereka untuk kembali mengunjungi restoran tersebut.

Restoran perlu memantau serta menganalisis ulasan pelanggan secara teratur untuk mengidentifikasi masalah yang mungkin muncul dan menyesuaikan layanan sesuai preferensi pelanggan. Ulasan yang diberikan pelanggan mencerminkan pengalaman mereka secara langsung dan dapat memberikan wawasan berharga bagi manajemen untuk mengevaluasi kekuatan serta kelemahan layanan yang ada. Namun, pemrosesan ulasan dalam jumlah besar secara manual bukanlah hal yang efisien karena memakan banyak waktu dan sumber daya yang tidak sedikit (Putu et al., 2021).

Penerapan sistem otomatis menjadi solusi yang efektif dalam mengklasifikasikan dan menganalisis ulasan pelanggan. Sistem ini dirancang agar dapat mengelompokkan setiap ulasan ke dalam salah satu dari 3 aspek utama, yaitu kualitas pelayanan, makanan, atau suasana. Dengan mengenali fokus utama dalam isi ulasan, sistem dapat secara otomatis melakukan klasifikasi sesuai kategori yang paling relevan. Dalam hal ini, teknologi seperti *Neural Network* menjadi metode yang potensial karena mampu mengidentifikasi pola dalam data teks secara mendalam (Franco et al., 2024). Dengan pendekatan ini ulasan pelanggan dalam jumlah besar dapat diproses secara lebih cepat, efisien, dan memberikan hasil klasifikasi yang lebih akurat.

2.2 Klasifikasi Aspek Ulasan

Dalam upaya memahami isi ulasan pelanggan secara lebih terstruktur, proses klasifikasi menjadi langkah lanjutan yang penting setelah pemodelan aspek dilakukan. Melalui proses ini, setiap ulasan dapat diberi label berdasarkan fokus utama yang dibahas, seperti kualitas pelayanan, makanan, atau suasana. Klasifikasi

ini tidak hanya membantu pelanggan dalam menemukan informasi yang sesuai dengan preferensi mereka, tetapi juga mempermudah pihak restoran dalam mengenali aspek-aspek yang perlu diperbaiki (Atsnawiyah et al., 2022). Tujuan dari klasifikasi ini adalah untuk memberikan label yang tepat pada setiap ulasan, agar pelanggan bisa lebih mudah menemukan informasi yang sesuai. Dengan adanya klasifikasi tersebut, pihak restoran pun dapat lebih mudah mengidentifikasi aspek yang perlu diperbaiki. Selain itu, proses analisis ulasan dapat dilakukan secara lebih cepat dan akurat, terutama ketika jumlah data yang tersedia sangat besar.

Dalam penelitian yang telah dilakukan oleh Wardianto et al (2023) berfokus pada analisis sentimen pada ulasan pelanggan restoran dengan pendekatan LSTM dan algoritma *Adam Optimizer*. Tujuan dari penelitian ini adalah untuk mengklasifikasikan opini pelanggan restoran dengan berbagai kategori, yaitu makanan, tempat, pelayanan, dan harga. Data yang digunakan diambil dari ulasan *TripAdvisor* berbahasa Indonesia dengan metode *Global Max Pooling*. Berdasarkan hasil penelitian, dapat dilihat bahwa analisis sentimen berdasarkan kategori sangat penting untuk mengklasifikasikan ulasan pelanggan, dengan tingkat akurasi mencapai 78% yang mana dapat membantu *TripAdvisor* dalam memahami opini pelanggan mengenai restoran.

Dinda et al (2023) pada penelitian ini membahas analisis sentimen terhadap ulasan produk serum *Somethinc*, dengan membandingkan dua jenis serum, yaitu serum *Niacinamide* dan serum AHA BHA PHA. Kedua kategori tersebut terdapat ulasan positif dan negatif. Dalam penelitian ini, diterapkan metode *text mining* dengan beberapa tahapan, seperti pengambilan data (*crawling*), pelabelan,

pembersihan data (*cleaning*), dan pra-pemrosesan (*preprocessing*) yang meliputi transformasi, *tokenisasi*, dan *filtering*. Algoritma yang dipakai yaitu *Naive Bayes* karena dinilai memiliki probabilitas tinggi dalam mengklasifikasikan data. Hasil penelitian menunjukkan bahwa jumlah data untuk serum *niacinamide* sebanyak 720 sementara serum AHA BHA PAH sebanyak 824 data. Diperoleh akurasi pada serum AHA BHA PAH sebanyak 80%, *presisi* 84%, dan *recall* 94%. Sedangkan pada serum *niacinamide* diperoleh akurasi 64%, *presisi* 67% dan *recall* 69%. Dengan demikian, performa klasifikasi ulasan untuk serum AHA BHA PHA dinilai lebih baik.

Salsabillah et al (2024) penelitian ini membandingkan metode *Support Vector Machine (SVM)* dan *Naive Bayes Classifier (NBC)* dalam mengklasifikasikan ulasan pelanggan untuk rumah makan ayam goreng nelongso. Selain itu, analisis *Root Cause Analysis* digunakan untuk mengevaluasi hasil rekomendasi yang didasarkan pada sentimen negatif pelanggan. Dari hasil pengujian, algoritma SVM tampil lebih baik dengan akurasi 92,74%, sementara *Naive Bayes Classifier (NBC)* hanya mencapai akurasi 91,67%. Selain itu, metode *Root Cause Analysis* menghasilkan sejumlah rekomendasi perbaikan pada aspek harga, kualitas makanan, layanan, serta kenyamanan tempat. Temuan ini diharapkan dapat menjadi acuan bagi pemilik rumah makan dalam meningkatkan mutu produk dan pelayanan yang diberikan.

Wona et al (2023) penelitian ini mengangkat topik klasifikasi batik Indonesia berdasarkan ciri-ciri visual seperti corak, warna, dan bentuk melalui media berbasis *website*. Data yang digunakan diperoleh dari *platform Kaggle*,

sebanyak 3.880 citra batik. Dataset tersebut kemudian dibagi menjadi 3 bagian, yaitu 70% untuk pelatihan (2.716 gambar), 20% untuk validasi (776 gambar), dan 10% untuk pengujian (388 gambar). Dalam penelitian ini diterapkan metode *Convolutional Neural Network* (CNN) yang memungkinkan pengguna mempelajari dan mengidentifikasi motif batik Indonesia secara interaktif melalui *platform web*. Dengan demikian, penelitian ini menghadirkan sistem klasifikasi batik berbasis CNN sebagai solusi dalam pengenalan batik Indonesia.

Widayat (2021) penelitian ini memanfaatkan metode *Long Short-Term Memory* (LSTM) dan *Word2Vec* dalam melakukan analisis sentimen terhadap ulasan film. Data yang digunakan berasal dari dataset IMDB yang berisi 25.000 ulasan, dengan panjang rata-rata sekitar 233 kata per ulasan. Penerapan *Word2Vec* dilakukan melalui dua pendekatan, yaitu *Continuous Bag of Words* (CBOW) dan *Skip-Gram*, dengan variasi ukuran dimensi vektor antara 50 hingga 500. Model LSTM dirancang untuk mengolah teks yang panjang dan kompleks dengan melakukan penyesuaian parameter. Hasil penelitian menunjukkan bahwa metode *Skip-Gram* dengan dimensi vektor 100 memberikan akurasi terbaik, yaitu sebesar 88,17%, mengungguli dimensi lainnya. Penelitian ini diharapkan menjadi acuan dalam menganalisis sentimen pada data besar dan memahami opini publik secara lebih baik.

Najiyah (2023) penelitian ini berfokus pada analisis respon masyarakat Indonesia terhadap kenaikan BBM pada September 2022, menggunakan metode *Artificial Neural Network* (ANN) untuk mengklasifikasikan sentimen masyarakat menjadi 3 kategori yaitu, positif, negatif, atau netral. Data dikumpulkan dari

platform *Twitter* melalui pencarian dengan tagar #kenaikanbbm, menghasilkan sebanyak 1210 ulasan yang digunakan sebagai dataset. Sebelum dianalisis, data ini melalui beberapa tahapan *preprocessing*, seperti *tokenisasi*, normalisasi, *stemming*, dan *stopword* untuk meningkatkan kualitas data. Metode ANN dipilih karena sangat efektif untuk mengolah data teks secara efisien. Model ini dirancang dengan menyesuaikan parameter, seperti panjang input data, untuk menghasilkan analisis yang optimal. Hasil penelitian menunjukkan bahwa ANN dapat mengelompokkan sentimen masyarakat dengan tingkat akurasi yang cukup tinggi, sebesar 83,4%. Penelitian ini bermanfaat untuk memahami opini masyarakat terhadap kebijakan pemerintah dengan cara yang lebih terukur dan sistematis.

Penelitian terakhir dari Adhiatma & Qoiriah (2022) menganalisis sentimen ulasan hotel yang ditulis dalam Bahasa Indonesia menggunakan metode TF-IDF dan *Deep Neural Network*. Data yang dipakai terdiri atas 1072 ulasan hotel yang diambil dari situs tiket.com menggunakan teknik *web scraping*. Ulasan ini diklasifikasikan menjadi 3 kategori yaitu, positif, negatif, atau netral. Dalam penelitian ini, teks ulasan dikonversi menjadi vektor numerik menggunakan metode TF-IDF, kemudian diproses oleh model *Deep Neural Network* untuk melakukan klasifikasi sentimen. Eksperimen dilakukan dengan berbagai kombinasi parameter, seperti *n-gram* (1, 2, dan 3), jumlah *hidden layer*, serta nilai *learning rate*, guna memperoleh konfigurasi terbaik. Hasil optimal diperoleh pada kombinasi *n-gram* 3, 3 *hidden layer*, dan *learning rate* 0,001, dengan akurasi mencapai 85% dan *f1-Score* sebesar 80%. Temuan ini menegaskan bahwa pemilihan parameter yang sesuai berperan penting dalam meningkatkan kinerja model analisis sentimen.

BAB III

DESAIN DAN IMPLEMENTASI

3.1 Akuisisi Data

Data yang digunakan dalam penelitian ini diambil dari komentar pelanggan di *Google Review*. Pengambilan data dilakukan dengan menggunakan *Instant Data Scraper*, dengan batasan hanya pada ulasan berbahasa Indonesia. Ulasan yang dikumpulkan berasal dari restoran-restoran di 38 provinsi di Indonesia, dengan prioritas pada restoran yang memiliki tingkat kunjungan tinggi atau populer di kalangan pelanggan. Dalam hal ini, restoran yang dianggap populer dilihat dari rating yang diberikan pelanggan pada *Google Review*, yaitu berada pada rentang 4 hingga 5 bintang. Setiap provinsi diwakili oleh 1 hingga 3 restoran, tergantung pada jumlah ulasan yang tersedia. Hal ini bertujuan agar jumlah data dari setiap provinsi seimbang dan dapat mewakili pendapat pelanggan dari berbagai daerah di Indonesia.

Fokus utama dari penelitian ini adalah pada ulasan yang membahas salah satu dari 3 aspek dalam ulasan pelanggan restoran, yaitu kualitas pelayanan, makanan, atau suasana. Ketiga aspek ini dipilih karena sering menjadi sorotan utama dalam ulasan pelanggan, baik dalam bentuk pujian maupun keluhan. Selain itu, aspek-aspek ini juga sering dijadikan acuan dalam penelitian terdahulu untuk mengevaluasi kepuasan pelanggan terhadap restoran. Berdasarkan hasil pengumpulan data, sebagian besar ulasan pelanggan memang cenderung membahas salah satu dari ketiga aspek tersebut.

Untuk menjaga konsistensi terhadap fokus klasifikasi, dilakukan proses penyaringan data yang terarah dan selektif. Hanya ulasan yang membahas satu aspek secara spesifik (*single label*) yang dipilih untuk dianalisis. Ulasan yang mengandung lebih dari satu aspek sekaligus (*multi label*), seperti yang menyinggung kualitas pelayanan dan makanan dalam satu kalimat, tidak dimasukkan dalam dataset. Apabila ditemukan ulasan yang memuat lebih dari satu aspek, maka hanya bagian yang secara jelas mengarah pada salah satu aspek utama yang diambil, sedangkan bagian lainnya diabaikan. Hal ini dilakukan agar proses pelabelan menjadi lebih jelas dan tidak membingungkan, sehingga model klasifikasi yang dikembangkan dapat lebih fokus dalam belajar membedakan masing-masing aspek dengan akurat. Pendekatan ini membantu menjaga kualitas data dan memudahkan proses pelatihan model.

Data yang diperoleh masih berupa data mentah, tanpa label kategori seperti kualitas pelayanan, makanan, atau suasana. Oleh karena itu, diperlukan proses pelabelan secara manual agar data dapat digunakan untuk klasifikasi. Untuk menghindari subjektivitas dalam pelabelan data, dilakukan survei kepada sejumlah responden untuk memastikan bahwa aspek kualitas pelayanan, makanan, dan suasana yang digunakan dalam penelitian sesuai dengan persepsi umum masyarakat.

Beberapa pertanyaan dalam survei diajukan untuk memperoleh informasi yang berkaitan dengan pengalaman dan persepsi responden terhadap ulasan restoran. Pertanyaan tersebut mencakup :

1. Frekuensi responden makan di restoran dan pengalaman responden menulis ulasan restoran secara *online*, untuk memastikan bahwa responden memiliki pengalaman yang relevan dalam menilai suatu restoran.
2. Aspek yang paling sering dibahas dalam ulasan *online*, yang memberikan gambaran awal mengenai aspek-aspek yang dianggap penting oleh pelanggan dalam menilai restoran.
3. Hal-hal yang menurut responden termasuk dalam aspek kualitas pelayanan, seperti keramahan staf, kecepatan dalam pelayanan, respon restoran terhadap keluhan pelanggan, kemampuan staf dalam menjawab pertanyaan atau permintaan pelanggan dan lainnya.
4. Hal-hal yang menurut responden termasuk dalam aspek makanan, seperti cita rasa masakan, kesesuaian porsi makanan dengan harga, kematangan atau tekstur makanan, kebersihan makanan, dan lainnya.
5. Hal-hal yang menurut responden termasuk dalam aspek suasana, seperti kebersihan ruangan dan meja, kenyamanan tempat (tidak bising/sejuk/tenang), desain interior menarik, pencahayaan yang bagus, dan lainnya.

Hasil dari survei tersebut kemudian digunakan untuk menyusun panduan pelabelan. Panduan ini membantu dalam mengarahkan proses pelabelan ulasan berdasarkan pemahaman umum. Sebagai contoh, ulasan yang membahas keramahan staf dan kecepatan pelayanan diberi label kualitas pelayanan, sementara yang membahas rasa makanan dan kematangan makanan diberi label makanan.

Ulasan yang menyinggung kenyamanan tempat atau suasana ruangan diberi label suasana.

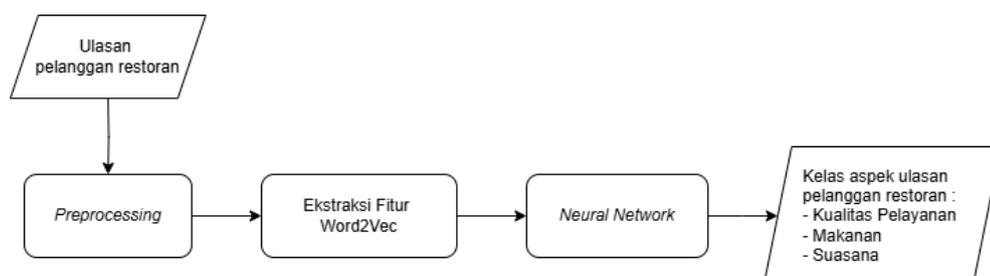
Dengan adanya panduan tersebut, proses pelabelan tidak hanya berdasarkan pendapat pribadi, tetapi mengacu pada hasil survei yang merepresentasikan persepsi umum. Dengan begitu, pelabelan jadi lebih objektif dan terarah. Hasil dari proses pelabelan ini, diperoleh sebanyak 1026 ulasan yang telah diberi label dan siap digunakan untuk pelatihan model klasifikasi. Seluruh data ulasan diperoleh dalam rentang waktu awal tahun 2024 hingga 15 September 2024. Contoh dataset yang digunakan dalam penelitian ini di tampilkan pada Tabel 3.1.

Tabel 3.1 Dataset ulasan pelanggan restoran

No	Unnamed:0	Text	Label
1	1	Tempatnya strategists dari bandara	Suasana
2	2	Cepet banget pelayanannya..	Kualitas pelayanan
3	3	Enak murah mengesyangkan pakatnya buat berempat	Makanan
4	4	Kepiting saus Dandito nya enakk poll	Makanan
5	5	Masakannya enak dan ramah di kantong	Makanan
...
1026	1026	Tempatnya luassss.. terang Dan bersih.	Suasana

3.2 Desain Sistem

Desain sistem ini mencakup alur dari pengumpulan ulasan pelanggan restoran hingga pengklasifikasiannya ke dalam kategori yang sesuai. Alur yang menggambarkan penelitian ini dapat dilihat pada Gambar 3.1.

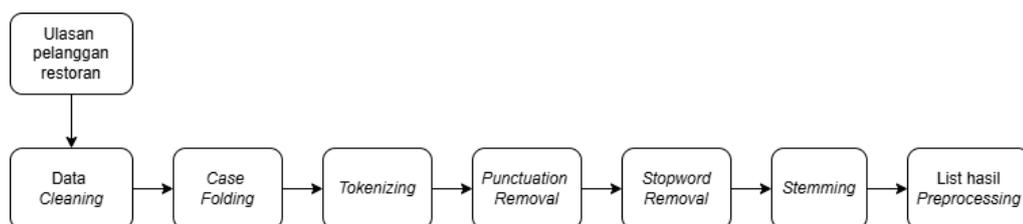


Gambar 3.1 Desain sistem

Pada penelitian ini, proses dimulai dengan mengumpulkan ulasan dari pelanggan restoran. Selanjutnya, ulasan tersebut melalui tahap *preprocessing*, dimana teks dibersihkan dan disiapkan untuk analisis. Setelah itu, fitur dari ulasan diekstraksi menggunakan metode *Word2vec*, yang mengubah teks menjadi format numerik agar dapat diproses lebih lanjut. Setelah ekstraksi fitur, data yang sudah siap akan diklasifikasikan menggunakan metode *Neural Network*. Hasil dari proses ini adalah pengelompokan ulasan ke dalam aspek yang berbeda, yaitu kualitas pelayanan, makanan atau suasana.

3.2.1 *Preprocessing*

Tahap *preprocessing* dalam pengolahan data sangat penting yang bertujuan untuk membersihkan dan menyiapkan data agar dapat dianalisis dengan lebih efektif menggunakan algoritma *machine learning*. Alur tahapan *preprocessing* dapat dilihat pada Gambar 3.2.



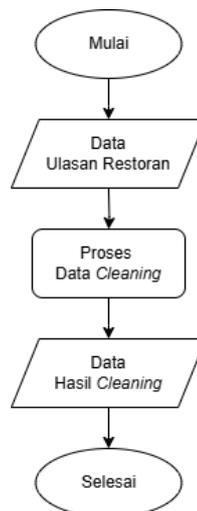
Gambar 3.2 Alur tahapan *preprocessing*

Proses *preprocessing* dimulai dengan pembersihan data (*data cleaning*), normalisasi huruf (*case folding*), memecah teks menjadi bagian-bagian kecil (*tokenizing*), penghapusan tanda baca (*punctuation removal*), penghilangan kata-kata umum (*stopword removal*), dan *stemming* untuk mengembalikan kata ke

bentuk dasar. Hasil dari tahap ini disusun menjadi data yang siap digunakan untuk proses pembobotan menggunakan *Word2vec*.

3.2.1.1 Data Cleaning

Tahapan ini merupakan tahapan penting dalam *preprocessing*. Tahapan ini berfokus pada pembersihan data mentah yang bersifat *noisy* atau tidak konsisten secara keseluruhan. Pada penelitian ini, dilakukan penghapusan kolom yang tidak diperlukan serta pengecekan data duplikat. Proses ini bertujuan untuk memastikan bahwa data yang dihasilkan adalah ulasan pelanggan restoran yang sudah bersih dan siap digunakan untuk analisis berikutnya (Ramdhani et al., 2020). Alur tahapan data *cleaning* dapat dilihat pada Gambar 3.3.



Gambar 3.3 Flowchart tahap Data Cleaning

Proses data *cleaning* terdiri dari beberapa langkah pengecekan untuk memastikan kualitas data, yang meliputi :

1. Menghapus kolom yang tidak diperlukan, kolom *Unnamed: 0* dihapus karena hanya berisi nomor indeks yang tidak berpengaruh terhadap analisis.

2. Mengecek data yang duplikat, tidak ditemukan adanya data duplikat dalam dataset yang digunakan pada Tabel 3.1.

Hasil dari proses data *cleaning* ditampilkan pada Tabel 3.2.

Tabel 3.2 Dataset setelah proses data *cleaning*

No	Text	Label
1	Tempatnya strategists dari bandara	Suasana
2	Cepet banget pelayanannya..	Kualitas pelayanan
3	Enak murah mengesyangkan paketnya buat berempat	Makanan
4	Kepiting saus Dandito nya enakk poll	Makanan
5	Masakannya enak dan ramah di kantong	Makanan
...
1026	Tempatnya luassss.. terang Dan bersih.	Suasana

3.2.1.2 Case Folding

Setelah data dibersihkan, tahap selanjutnya adalah *case folding*, yaitu mengubah seluruh teks ulasan menjadi huruf kecil (*lowercase*) (Ma'rifah et al., 2020). Proses ini bertujuan untuk menyamakan format teks, sehingga jika terdapat campuran huruf besar dan kecil, semuanya akan diubah menjadi huruf kecil. Alur tahapan proses *case folding* dapat dilihat pada Gambar 3.4, sementara hasil setelah proses *case folding* ditampilkan pada Tabel 3.3.



Gambar 3.4 Flowchart tahap *Case Folding*

Tabel 3.3 Dataset hasil proses *case folding*

Text (Sebelum <i>Case Folding</i>)	Text (Setelah <i>Case Folding</i>)
Tempatnya strategists dari bandara	tempatnya strategists dari bandara
Cepet banget pelayanannya..	cepat banget pelayanannya..
Enak murah mengesyangkan pakatnya buat berempat	enak murah mengesyangkan pakatnya buat berempat
Kepiting saus Dandito nya enakk poll	kepiting saus dandito nya enakk poll
Masakannya enak dan ramah di kantong	masakannya enak dan ramah di kantong
...	...
Tempatnya luassss.. terang Dan bersih.	tempatnya luassss.. terang dan bersih.

3.2.1.3 *Tokenizing*

Tokenizing adalah tahapan dalam *preprocessing* yang bertujuan memisahkan teks menjadi elemen-elemen kecil, seperti kata atau frasa, yang disebut token (Kurniawati et al., 2022). Tahap ini membantu sistem dalam memproses dan menganalisis teks lebih mudah. Proses ini membuat teks ulasan lebih terorganisir sehingga lebih mudah dipahami oleh sistem. Alur tahapan *tokenizing* dapat dilihat pada Gambar 3.5, sementara hasil setelah proses *tokenizing* ditampilkan pada Tabel 3.4.

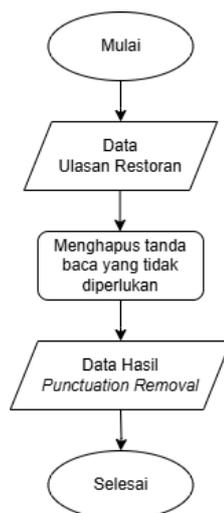
Gambar 3.5 *Flowchart* tahap *Tokenizing*

Tabel 3.4 Dataset hasil proses *tokenizing*

Text (Sebelum <i>Tokenizing</i>)	Text (Setelah <i>Tokenizing</i>)
tempatnnya strategists dari bandara	['tempatnnya', 'strategists', 'dari', 'bandara']
cepat banget pelayanannya..	['cepat', 'banget', 'pelayanannya..']
enak murah mengesyangkan paketnya buat berempat	['enak', 'murah', 'mengesyangkan', 'paketnya', 'buat', 'berempat']
kepiting saus dandito nya enakk poll	['kepiting', 'saus', 'dandito', 'nya', 'enakk', 'poll']
masakannya enak dan ramah di kantong	['masakannya', 'enak', 'dan', 'ramah', 'di', 'kantong']
...	...
tempatnnya luassss.. terang dan bersih.	['tempatnnya', 'luassss..', 'terang', 'dan', 'bersih.']

3.2.1.4 *Punctuation Removal*

Punctuation removal adalah tahapan pada *preprocessing* yang bertujuan menghilangkan semua tanda baca, simbol atau emoji dari data ulasan. Dengan menghapus elemen-elemen yang tidak diperlukan ini, sistem dapat lebih mudah mengidentifikasi kata-kata dan konteks yang ada dalam teks (Tabassum & Patil, 2020). Dengan melakukan penghapusan tanda baca, teks ulasan akan lebih mudah untuk diproses. Alur tahapan *punctuation removal* dapat dilihat pada Gambar 3.6, sementara hasil setelah proses *punctuation removal* ditampilkan pada Tabel 3.5.

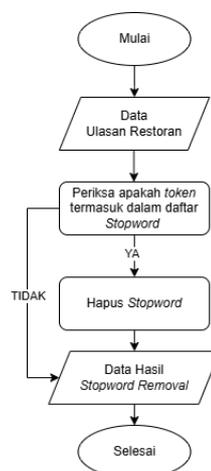
Gambar 3.6 *Flowchart* tahap *Punctuational Removal*

Tabel 3.5 Dataset hasil proses *punctuational removal*

Text (Sebelum <i>Punctuational Removal</i>)	Text (Setelah <i>Punctuational Removal</i>)
['tempatya', 'strategists', 'dari', 'bandara']	['tempatya', 'strategists', 'dari', 'bandara']
['cepat', 'banget', 'pelayanannya..']	['cepat', 'banget', 'pelayanannya']
['enak', 'murah', 'mengeyangkan', 'paketnya', 'buat', 'berempat']	['enak', 'murah', 'mengeyangkan', 'paketnya', 'buat', 'berempat']
['kepiting', 'saus', 'dandito', 'nya', 'enakk', 'poll']	['kepiting', 'saus', 'dandito', 'nya', 'enakk', 'poll']
['masakannya', 'enak', 'dan', 'ramah', 'di', 'kantong']	['masakannya', 'enak', 'dan', 'ramah', 'di', 'kantong']
...	...
['tempatya', 'luassss..', 'terang', 'dan', 'bersih.']	['tempatya', 'luassss', 'terang', 'dan', 'bersih']

3.2.1.5 Stopword Removal

Stopword removal adalah proses membuang kata-kata yang bersifat umum dan kurang relevan dalam analisis teks. Kata-kata ini biasanya sering muncul dalam percakapan sehari-hari, seperti “dan”, “di”, “atau” dan sebagainya (Astuti et al., 2022). Proses dilakukan dengan hati-hati karena konteks ulasan penting untuk memahami aspek yang dinilai, sehingga analisis lebih terfokus pada kata-kata yang mencerminkan pendapat pengguna. Alur tahapan *stopword removal* dapat dilihat pada Gambar 3.7, sementara hasil setelah proses *stopword removal* ditampilkan pada Tabel 3.6.

Gambar 3.7 Flowchart tahap *Stopword Removal*

Tabel 3.6 Dataset hasil proses *stopword removal*

Text (Sebelum <i>Stopword Removal</i>)	Text (Setelah <i>Stopword Removal</i>)
['tempatya', 'strategists', 'dari', 'bandara']	['tempatya', 'strategists', 'bandara']
['cepat', 'banget', 'pelayanannya']	['cepat', 'banget', 'pelayanannya']
['enak', 'murah', 'mengeyangkan', 'paketnya', 'buat', 'berempat']	['enak', 'murah', 'mengeyangkan', 'paketnya', 'buat', 'berempat']
['kepiting', 'saus', 'dandito', 'nya', 'enakk', 'poll']	['kepiting', 'saus', 'dandito', 'enakk', 'poll']
['masakannya', 'enak', 'dan', 'ramah', 'di', 'kantong']	['masakannya', 'enak', 'ramah', 'kantong']
...	...
['tempatya', 'luassss', 'terang', 'dan', 'bersih']	['tempatya', 'luassss', 'terang', 'bersih']

3.2.1.6 Stemming

Stemming adalah mengubah kata-kata yang memiliki bentuk yang berbeda-beda (seperti kata kerja yang mendapat awalan atau akhiran) menjadi bentuk dasar atau bentuk paling sederhana dari kata tersebut (Ulgasesa et al., 2022). Tujuan dari *stemming* adalah agar komputer hanya fokus pada kata dasarnya, bukan variasi kata yang membuatnya bingung. Dengan melakukan *stemming*, variasi kata yang tidak perlu dapat dihilangkan, sehingga sistem dapat lebih fokus pada makna inti dari kata-kata yang dianalisis. Alur tahapan *stemming* dapat dilihat pada Gambar 3.8, sementara hasil setelah proses *stemming* ditampilkan pada Tabel 3.7.

Gambar 3.8 Flowchart tahap *Stemming*

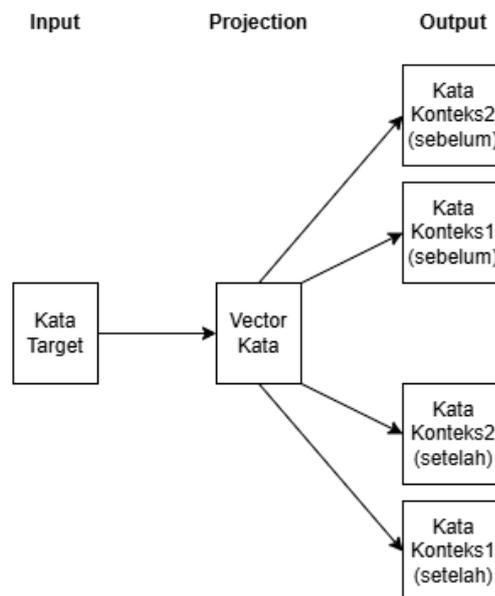
Tabel 3.7 Dataset hasil proses *stemming*

Text (Sebelum <i>Stemming</i>)	Text (Setelah <i>Stemming</i>)
['tempatya', 'strategists', 'bandara']	['tempat', 'strategists', 'bandara']
['cepat', 'banget', 'pelayanannya']	['cepat', 'banget', 'layan']
['enak', 'murah', 'mengeyangkan', 'paketnya', 'buat', 'berempat']	['enak', 'murah', 'eyang', 'paket', 'buat', 'empat']
['kepiting', 'saus', 'dandito', 'enakk', 'poll']	['kepiting', 'saus', 'dandito', 'enakk', 'poll']
['masakannya', 'enak', 'ramah', 'kantong']	['masakan', 'enak', 'ramah', 'kantong']
...	...
['tempatya', 'luassss', 'terang', 'bersih']	['tempat', 'luassss', 'terang', 'bersih']

3.2.2 *Word2Vec*

Tahapan berikutnya dalam penelitian ini adalah ekstraksi fitur menggunakan *Word2vec*, sebuah teknik yang penting dalam pemrosesan bahasa alami (*Natural Language Processing*). *Word2vec* berfungsi untuk mengubah kata-kata dalam teks menjadi angka atau vektor, yang memungkinkan komputer memahami hubungan antar kata (Mikolov et al., 2013). Vektor ini tidak hanya mewakili kata itu sendiri, tetapi juga menggambarkan arti kata tersebut berdasarkan kata-kata lain disekitarnya. Kata-kata yang artinya mirip atau sering muncul bersama dalam kalimat akan memiliki vektor yang serupa.

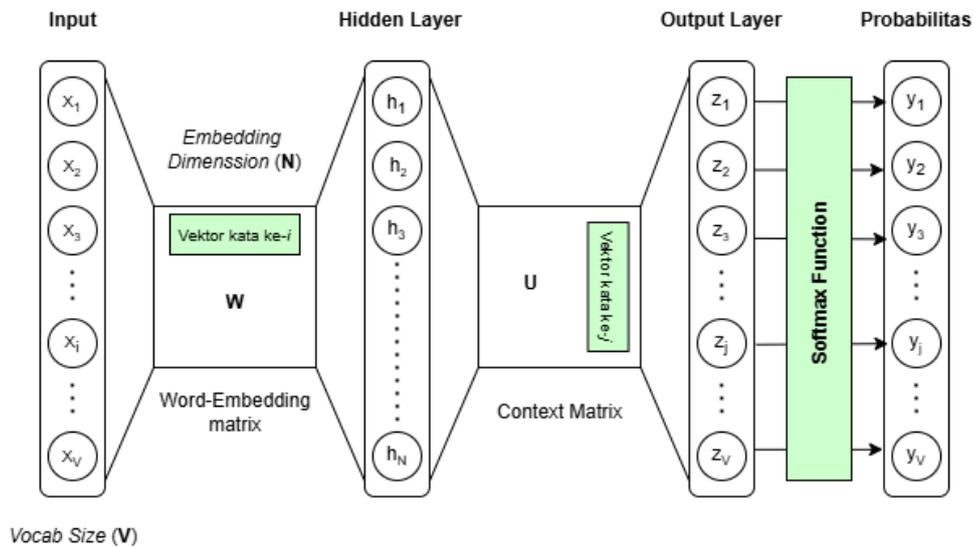
Untuk mencapai hal ini, *Word2vec* menggunakan 2 model utama yaitu, CBOW (*Continuous Bag of Words*) dan Skip-Gram. Kedua model ini memiliki pendekatan yang berbeda dalam memproses konteks kata untuk menghasilkan representasi kata yang lebih akurat. Dalam penelitian ini, model Skip-Gram dipilih sebagai pendekatan utama karena lebih efektif dalam memprediksi kata-kata yang ada disekitar kata target. Model Skip-Gram bekerja dengan memilih satu kata target dalam kalimat, kemudian mencoba memprediksi probabilitas kata-kata yang dapat muncul sebagai konteks dari kata target tersebut (Kim & Kim, 2022). Konsep dasar dari *Word2vec* Skip-Gram dapat dilihat pada Gambar 3.9.



Gambar 3.9 Konsep dasar *Word2Vec* Skip-Gram

Pada model Skip-Gram, salah satu faktor penting yang memengaruhi bagaimana konteks kata dipilih adalah *window size*. Hal ini disebabkan karena *window size* menentukan jumlah kata yang akan diambil sebagai konteks di sekitar kata target (Drzik & Kapusta, 2024). Misalkan, *window size* ditetapkan menjadi 2, maka model akan memilih 2 kata sebelum dan 2 kata setelah kata target untuk menjadi kata kontekstual. Namun, jika kata target berada diawal kalimat, maka tidak ada kata konteks untuk kata sebelum kata target, dan sebaliknya, jika kata target berada di akhir kalimat, maka tidak ada kata konteks untuk kata setelahnya. Sebagai contoh, dalam kalimat “menu makan minum cukup variasi ada snack cemilan”, dengan kata target “variasi” dan *window size* = 2, maka kata kontekstual yang dipilih adalah {minum cukup | variasi} dan {variasi | ada snack}. Adapun parameter lainnya dalam perhitungan *Word2vec*, seperti *learning rate*, *embedding dimension*, dan *epoch*, memiliki peran penting dalam mengoptimalkan pelatihan

model. Proses pelatihan *Word2vec* dapat dipahami lebih jelas melalui arsitektur *Word2Vec* Skip-Gram pada Gambar 3.10.



Gambar 3.10 Arsitektur *Word2Vec* Skip-Gram

Proses implementasi algoritma *Word2Vec* Skip-Gram untuk membentuk vektor representasi kata terdiri dari beberapa tahapan berikut :

1. Inisialisasi parameter model berupa *window size*, *learning rate*, *embedding dimension* (N), *epoch*, serta matriks bobot W (*input layer* ke *hidden layer*) dan bobot U (*hidden layer* ke *output layer*).
2. Membuat *vocabulary* dengan memberi indeks pada kata unik berdasarkan frekuensi kemunculannya.
3. Menentukan kata target dan kata konteks untuk setiap kata dalam kalimat sesuai dengan *window size*. Pada penelitian ini, digunakan *window size* = 2, mengacu pada penelitian Irianto et al (2024) yang menunjukkan bahwa penggunaan *window size* = 2 memberikan kinerja lebih baik dibandingkan ukuran lainnya. Dalam proses ini, kata target dipilih dan diberi nilai 1 dalam

vektor *one-hot encoding*, sementara kata konteks digunakan untuk melatih model agar dapat memprediksi kata target.

4. *Feedforward*

- a. Perhitungan aktivasi pada *hidden layer* menggunakan persamaan 3.1

$$h = W^T \times x \quad (3.1)$$

Keterangan :

h = *hidden layer*

W^T = *transpose* matrik bobot W yang menghubungkan *input layer* ke *hidden layer*

x = *input* vektor

Persamaan 3.1 digunakan untuk untuk mengambil vektor dari kata *input*. Kata *input* dikalikan dengan bobot W yang ditranspos untuk menghasilkan vektor h , yaitu bentuk angka yang mewakili makna kata tersebut. Vektor ini nanti digunakan untuk memprediksi kata-kata konteks di sekitarnya.

- b. Perhitungan aktivasi pada *output layer* menggunakan persamaan 3.2

$$z = U^T \times h \quad (3.2)$$

Keterangan :

z = vektor skor *output*

U^T = *transpose* dari matrik bobot U yang menghubungkan *hidden layer* ke *output layer*

h = *hidden layer*

Persamaan 3.2 digunakan untuk menghitung skor *output*, yaitu seberapa besar kemungkinan setiap kata dalam *vocabulary* menjadi kata konteks berdasarkan kata *input*. Vektor h , yang merupakan hasil dari *hidden layer*, dikalikan dengan bobot U yang ditranspos. Hasilnya adalah z ,

sebuah vektor skor yang mempresentasikan potensi setiap kata dalam kosakata menjadi kata konteks.

5. Menghitung probabilitas kata konteks berdasarkan hasil *output layer* dengan menggunakan fungsi *softmax* pada persamaan berikut.

$$s = \frac{\exp(z_j - \max(z))}{\sum_{j=1}^V \exp(z_j - \max(z))} \quad (3.3)$$

$$y = \text{softmax}(z) \quad (3.4)$$

Keterangan :

s = probabilitas indeks kata yang diprediksi menjadi kata konteks

y = probabilitas prediksi kemunculan kata konteks berdasarkan kata target

V = jumlah total kata dalam dataset (*vocab_size*)

Persamaan 3.3 digunakan untuk menghitung probabilitas kata konteks berdasarkan skor yang dihasilkan dari *output layer* menggunakan fungsi *Softmax*. Sedangkan persamaan 3.4 menunjukkan bahwa probabilitas prediksi y untuk kata konteks adalah hasil dari penerapan fungsi *Softmax* pada skor *output* z .

6. Menghitung nilai *error* untuk mengukur selisih antara hasil prediksi dan representasi sebenarnya dari kata konteks menggunakan persamaan 3.5

$$e = y - y_{true} \quad (3.5)$$

Keterangan :

e = nilai *error*

y_{true} = vektor *one-hot* untuk kata konteks sebenarnya

Persamaan 3.5 digunakan untuk menghitung selisih antara hasil prediksi model y dan representasi vektor kata konteks sebenarnya y_{true} . Selisih ini

disebut *error e*, dan akan digunakan dalam proses *backpropagation* untuk memperbaiki bobot model.

7. *Backpropagation*

- a. Perhitungan gradien untuk pembaruan bobot *output layer* ke *hidden layer* dengan rumus yang terdapat pada persamaan 3.6

$$dU = h \times e^T \quad (3.6)$$

Keterangan :

dU = gradien yang diperlukan untuk memperbarui bobot U antara *hidden layer* dan *output layer*

e^T = *transpose* dari nilai *error e*

Persamaan 3.6 digunakan untuk menghitung *gradien dU*, yaitu perubahan bobot yang dibutuhkan antara *hidden layer* dan *output layer*.

Nilai ini nantinya digunakan untuk memperbarui bobot *output* agar prediksi semakin mendekati target.

- b. Perhitungan gradien untuk pembaruan bobot *hidden layer* ke *input layer* dengan rumus yang terdapat pada persamaan 3.7

$$dW = x \times (U \times e)^T \quad (3.7)$$

Keterangan :

x = vector *one-hot* untuk kata target

dW = gradien yang diperlukan untuk memperbarui bobot W antara *input layer* dan *hidden layer*

U = bobot yang menghubungkan *hidden layer h* ke *output layer z*

Persamaan 3.7 digunakan untuk menghitung *gradien dW*, yaitu perubahan bobot yang dibutuhkan antara *input layer* dan *hidden layer*.

Nilai ini menunjukkan seberapa besar penyesuaian yang perlu dilakukan

pada bobot *input* agar representasi kata menjadi lebih akurat dalam memprediksi konteks.

8. *Update bobot*

- a. Perhitungan pembaruan bobot baru untuk *input layer* ke *hidden layer* dengan rumus yang terdapat pada persamaan 3.8

$$W = W_{lama} - (learning\ rate \times dW) \quad (3.8)$$

Keterangan :

W = nilai bobot baru setelah pembaruan

W_{lama} = nilai bobot sebelum diperbarui

learning rate = seberapa besar perubahan yang dilakukan pada bobot

Persamaan 3.8 digunakan untuk menghitung bobot W baru, yang bertujuan untuk memperbaiki bobot agar kesalahan prediksi di iterasi sebelumnya dapat dikurangi pada proses selanjutnya.

- b. Perhitungan pembaruan bobot baru untuk *hidden layer* ke *output layer* dengan rumus yang terdapat pada persamaan 3.9

$$U = U_{lama} - (learning\ rate \times dU) \quad (3.9)$$

Keterangan :

U = nilai bobot baru setelah pembaruan

U_{lama} = nilai bobot sebelum diperbarui

Persamaan 3.9 digunakan untuk memperbarui bobot U baru, dengan tujuan agar model dapat terus belajar dan menghasilkan prediksi *output* yang semakin akurat seiring berjalannya proses *training*.

9. Setelah mendapat bobot baru, proses *training* dilakukan agar model berusaha memaksimalkan kemungkinan kemunculan kata-kata konteks disekitarnya berdasarkan kata target.

10. Model melakukan *training* selama jumlah *epoch* dan nilai *learning rate* yang telah ditentukan. Dalam penelitian ini, jumlah *epoch* yang digunakan adalah sebanyak 100. Pemilihan jumlah *epoch* tersebut merujuk pada penelitian terdahulu oleh Dewananda et al (2022), yang menunjukkan bahwa *training Word2Vec* dengan 100 *epoch* mampu menghasilkan representasi kata yang optimal dan meningkatkan kinerja model berdasarkan hasil evaluasi. Sementara itu, nilai *learning rate* diperoleh melalui serangkaian percobaan dengan berbagai nilai, dan ditemukan bahwa *learning rate* sebesar 0.01 memberikan hasil *training* yang paling optimal.
11. *Loss* dihitung menggunakan fungsi *cross-entropy* pada setiap *epoch* untuk mengetahui total kesalahan prediksi berdasarkan persamaan 3.10

$$loss = - \sum_{context} \log (y_{context}) \quad (3.10)$$

Keterangan :

$loss$ = nilai kerugian yang menunjukkan seberapa baik atau buruk model

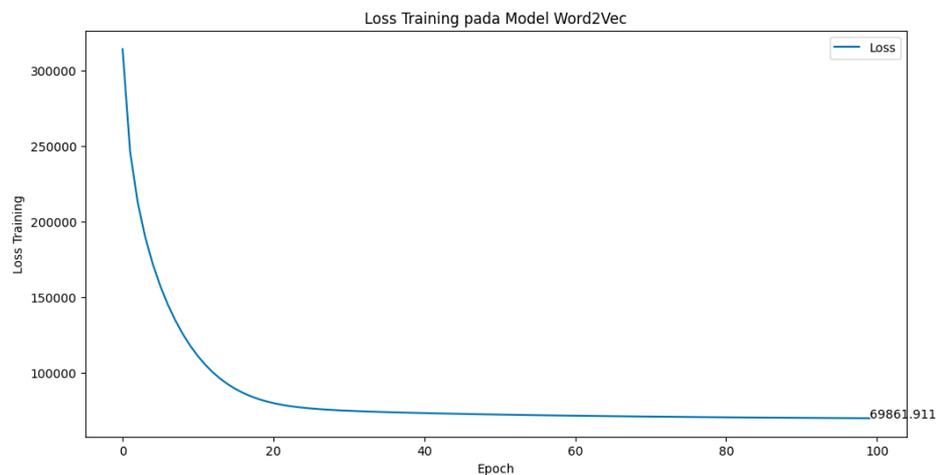
$\sum_{context}$ = menjumlahkan semua nilai untuk setiap kata konteks yang relevan

$y_{context}$ = probabilitas yang diprediksi oleh model untuk kata konteks tertentu

Persamaan 3.10 digunakan untuk menghitung total *loss* dalam satu *epoch* dengan menggunakan fungsi *cross-entropy*. Nilai *loss* ini mengukur seberapa besar ketidaksesuaian antara distribusi probabilitas yang diprediksi oleh model dengan distribusi probabilitas yang sebenarnya.

Sebagai hasil dari proses *training*, grafik perubahan nilai *loss* pada model *Word2Vec* Skip-Gram ditampilkan pada Gambar 3.11. Grafik ini menunjukkan penurunan nilai *loss* yang konsisten seiring bertambahnya *epoch*, meskipun nilai *loss* awalnya cukup besar. Penurunan tersebut

menunjukkan bahwa model semakin baik dalam mempelajari hubungan antar kata dan semakin akurat dalam memprediksi kata konteks.



Gambar 3.11 Grafik perubahan nilai *loss* pada *training Word2vec*

12. Setelah proses *training* selesai, vektor *embedding* untuk setiap kata dari matriks bobot W yang menghubungkan *input layer* dengan *hidden layer* diambil. Vektor-vektor ini mewakili dan menggambarkan makna setiap kata dalam bentuk numerik.

Berdasarkan data ulasan pelanggan restoran yang terdapat pada Tabel 3.1, model *Word2Vec* Skip-Gram dilatih menggunakan *embedding dimension* sebanyak 100. Setiap kata dalam data pelatihan kemudian direpresentasikan dalam bentuk vektor dengan 100 dimensi. Hasil dari proses *training* ini menghasilkan sebanyak 1332 vektor untuk kata-kata unik yang membentuk keseluruhan *vocabulary* dalam data. Seluruh hasil representasi vektor setiap kata dalam *Word2vec* Skip-Gram ditampilkan pada Tabel 3.8.

Tabel 3.8 Hasil representasi vektor setiap kata dalam *Word2vec* Skip-Gram

No	Kata	Vektor embedding <i>Word2Vec</i> (100 Dimensi)
1	makan	[0.45759812, 0.13022806, - 0.079872206,, -0.4753401, 0.0025066417]
2	enak	[0.116382465, 0.14127941, 0.10932935,, 0.39364654, -0.071357444]
3	tempat	[0.2934719, 0.38956144, 0.26297244,, 0.4571786, 0.25047594]
4	layan	[-0.42475516, 0.44296253, 0.24280503,, 0.04075472, 0.44966072]
5	nyaman	[-0.3859589, - 0.0138381235, 0.8667329,, 0.1361453, -0.342249]
...
1332	terang	[0.5097906, 1.4627746, 0.059870347,, 1.1380336, 0.02964964]

13. Setelah memperoleh vektor *embedding* untuk setiap kata, langkah terakhir adalah menggabungkan vektor-vektor tersebut untuk mewakili seluruh kalimat. Proses ini dilakukan dengan menghitung rata-rata semua vektor *embedding* dalam kalimat, sehingga menghasilkan satu vektor yang mencerminkan makna keseluruhan kalimat, seperti yang terdapat dalam persamaan 3.11.

$$r = \frac{1}{M} \sum_{k=1}^M W \quad (3.11)$$

Keterangan :

r = vektor *embedding* rata-rata kalimat

M = jumlah keseluruhan kata dalam kalimat

k = indeks setiap kata dalam kalimat

W = vektor *embedding* untuk setiap kata dalam kalimat

Hasilnya adalah vektor berdimensi 100 yang merepresentasikan makna keseluruhan kalimat. Vektor representasi kalimat ini selanjutnya digunakan sebagai *input* dalam proses pelatihan model *Neural Network*. Hasil representasi vektor *Word2Vec* Skip-Gram untuk seluruh kalimat dapat dilihat pada Tabel 3.9.

Tabel 3.9 Hasil representasi vektor *Word2vec* Skip-Gram

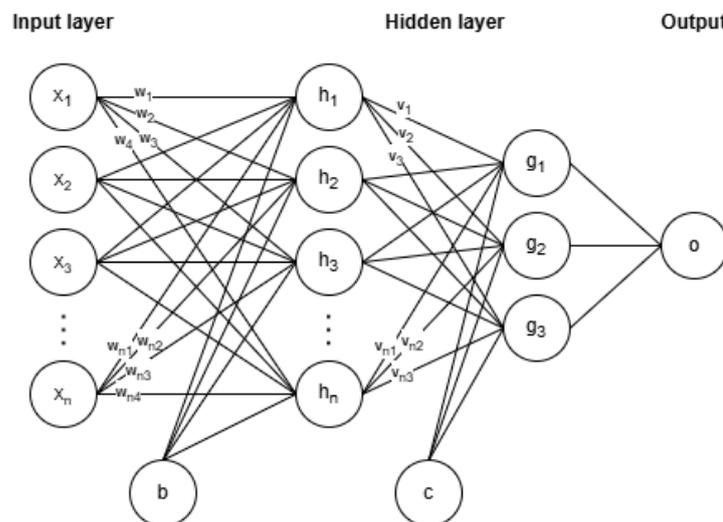
No	Text	Vektor embedding Word2Vec (100 Dimensi)
1	tempat strategists bandara	[0.25138393, -0.64706856, 0.32008174,, 0.44051966, 0.3130363]
2	cepat banget layanan	[-0.045998305, 0.11778003, 1.3858676,, 0.26071474, 0.26331803]
3	enak murah eyang paket buat empat	[-0.0028264225, -0.32789364, -0.342306,, 0.1631649, 0.14964706]
4	kepiting saus dandito enakk poll	[-0.09961661, -0.18886633, -1.0116621,, -0.009264946, 0.93141824]
5	masakan enak ramah kantong	[-0.10113463, 0.1782321, 0.76978004,, 0.16267757, -0.006509956]
...
1026	tempat luassss terang bersih	[0.2317542, -0.021801002, 0.32295787,, 0.0030753762, -0.14432926]

Alur proses tahapan pada model *Word2Vec* Skip-Gram dijelaskan pada Gambar 3.12. Diagram tersebut menggambarkan keseluruhan langkah mulai dari inialisasi parameter hingga pembaruan bobot secara iteratif, yang bertujuan menghasilkan vektor *embedding* untuk analisis lebih lanjut.

Gambar 3.12 Diagram alur tahapan model *Word2Vec* Skip-Gram

3.2.3 Neural Network

Neural Network adalah sebuah sistem yang bekerja mirip dengan cara otak manusia dalam memproses informasi (Mijwel et al., 2019). Jaringan ini terdiri dari kumpulan *neuron* buatan yang tersusun dalam beberapa lapisan, yaitu *input layer*, *hidden layer*, dan *output*. *Neuron-neuron* ini saling terhubung dan bekerja sama dalam mengolah data *input*, mengenali pola, dan membuat keputusan berdasarkan informasi yang telah dipelajari. Dengan menggunakan *Neural Network* ini komputer dapat belajar dari data yang diberikan dan menghasilkan *output* yang lebih akurat. Desain topologi *Neural Network* dapat dilihat pada Gambar 3.13.



Gambar 3.13 Desain topologi *Neural Network*

Beberapa lapisan *Neural Network* memiliki peran penting dalam pengolahan informasi. *Input layer* berfungsi untuk menerima data masukan yang akan diproses lebih lanjut. Data ini kemudian diteruskan ke *hidden layer*, yang terletak diantara *input layer* dan *output layer*. *Hidden layer* berfungsi untuk mengolah informasi lebih dalam, sehingga bisa mengenali hubungan yang lebih

kompleks dalam data. *output layer* adalah lapisan terakhir yang menghasilkan hasil akhir dari proses pengolahan data.

Proses dalam *Neural Network* dimulai dengan mengalikan *input* dengan bobot yang sesuai (Aryanti, 2023). Setelah itu, fungsi aktivasi diterapkan untuk menghasilkan *output* dari *layer* sebelumnya, yang kemudian diteruskan ke *layer* berikutnya. Setiap lapisan terdiri dari *node* yang memiliki bobot dan bias. Bobot digunakan untuk memproses informasi antar *layer*, sedangkan bias ditambahkan untuk membantu model mempelajari pola yang lebih kompleks. Dengan adanya bobot dan bias, model *Neural Network* dapat menghasilkan *output* yang lebih akurat dan tepat.

Dalam menentukan jumlah *node* pada *hidden layer*, penelitian ini mengacu pada pertimbangan dari penelitian sebelumnya Syaharuddin et al (2022) menyatakan bahwa jumlah *node* diperkirakan melalui perhitungan dan eksperimen, dengan kisaran $1/2$ hingga $1/3$ dari jumlah *input*, Chandra & Suryadibrata (2019) juga menentukan jumlah *node* berdasarkan metode *trial and error* tanpa aturan baku, dengan mencoba beberapa variasi untuk mendapatkan hasil terbaik. Berdasarkan hal tersebut, penelitian ini memilih tiga variasi jumlah *node* pada *hidden layer* 1, yaitu 25, 50, dan 100 *node*, di mana 50 *node* dipilih berdasarkan pendekatan setengah dari jumlah *input*, 100 *node* untuk menguji performa dengan jumlah *node* setara *input*, dan 25 *node* untuk mengevaluasi kinerja model dengan jumlah *node* yang lebih sedikit. Adapun penentuan jumlah *layer* dan *node* dalam penelitian ini dilakukan berdasarkan beberapa pertimbangan utama :

- a) *Input layer* : Panjang menyesuaikan *output* vektor *Word2Vec* yaitu 100 dimensi
- b) *Hidden layer 1* : Menggunakan variasi jumlah *node*, yaitu 25, 50, dan 100
- c) *Hidden layer 2* : Terdiri dari 3 *node*, Sesuai dengan jumlah kelas yang diprediksi
- d) *Output layer* : Terdiri dari 1 *node*, untuk menghasilkan prediksi dari salah satu kelas

Proses pengolahan data dalam penelitian ini dikenal dengan istilah *feedforward*, yaitu tahap di mana data mengalir secara searah dari *input layer* hingga mencapai *output layer* tanpa adanya proses umpan balik. Berikut adalah alur proses selama *Feedforward* :

1. Perhitungan di *hidden layer 1* menggunakan bobot dan bias yang terhubung dari *input layer* menuju *hidden layer 1*, kemudian diterapkan fungsi aktivasi ReLU (*Rectified Linear Unit*).

$$h_i = \max(0, \sum_{j=1}^n W_{ji} \times x_j + b) \quad (3.12)$$

Keterangan :

- h_i = nilai aktivasi dari *node* ke-*i* di *hidden layer 1*
- x_j = nilai *input layer* dari *node* ke-*j*
- W_{ji} = bobot yang menghubungkan *input layer* ke *hidden layer 1*
- b = bias pada *hidden layer 1*

Fungsi ReLU hanya meneruskan nilai positif dan mengganti nilai negatif menjadi nol. Hasil dari fungsi ini kemudian dikirim ke *hidden layer 2* sebagai *input* baru.

2. Perhitungan di *hidden layer 2* dengan menjadikan *output* dari *hidden layer 1* sebagai *input* baru.

$$g_i = \sum_{i=1}^n V_{ik} \times h_i + c \quad (3.13)$$

Keterangan :

g_i = *output* dari *node* ke- i pada *hidden layer 2* sebelum fungsi aktivasi ditetapkan

V_{ik} = bobot yang menghubungkan *hidden layer 1* ke *hidden layer 2*

c = bias pada *hidden layer 2*

Pada tahap ini, setiap *node* di *hidden layer 2* menerima masukan dari seluruh *node* di *hidden layer 1*. Masukan tersebut dikalikan dengan bobot V_{ik} , lalu dijumlahkan dan ditambahkan dengan bias c . Setelah diperoleh nilai dari masing-masing *node* pada *hidden layer 2*, perhitungan dilanjutkan menggunakan fungsi aktivasi *Softmax* untuk menghasilkan *output* akhir model.

3. Perhitungan *output* menggunakan fungsi *Softmax*. *Output* model diambil dari nilai probabilitas tertinggi sebagai hasil prediksi akhir.

$$o = \text{softmax} (\sum_{i=1}^n V_{ik} \times h_i + c) \quad (3.14)$$

Fungsi *Softmax* mengubah nilai numerik dari setiap *node* di *hidden layer 2* menjadi probabilitas dalam rentang 0 hingga 1, yang menunjukkan kemungkinan suatu ulasan termasuk ke dalam masing-masing kelas. Pada penelitian ini, *hidden layer 2* memiliki 3 *node* yang mewakili 3 kelas, yaitu kualitas pelayanan, makanan, dan suasana. Misalnya, jika *output* berupa [0.70, 0.55, 0.15], maka model akan memilih kelas dengan nilai probabilitas tertinggi sebagai hasil prediksi akhir. Oleh karena itu, meskipun *hidden layer 2* menghasilkan 3 nilai probabilitas, *output* akhir hanya 1, yaitu kelas dengan probabilitas tertinggi.

3.2.3.1 Proses *Training*

Setelah melakukan *feedforward* tahap selanjutnya adalah proses *training*. Pada tahap ini, model belajar memperbaiki bobot dan bias agar hasil prediksi mendekati *ground truth*. Proses ini menggunakan *backpropagation* yang bekerja dengan cara menghitung *error* dari *output feedforward*, menyebarkannya ke layer-layer sebelumnya, dan memperbarui parameter model. Berikut adalah alur proses selama proses *backpropagation* :

1. Perhitungan *error* menggunakan *cross-entropy loss*.

$$L = -\sum_{m=1}^3 y \times \log(o) \quad (3.15)$$

Keterangan :

L = *error* atau total nilai *loss*
 $\sum_{m=1}^3 y$ = jumlah kontribusi dari kelas yang benar
 y = nilai sebenarnya (*ground truth*)
 o = *output* hasil prediksi model

Loss dihitung dari nilai *log* probabilitas pada kelas yang benar. Nilai *loss* ini kemudian digunakan untuk mengukur seberapa jauh hasil prediksi model dibandingkan dengan label sebenarnya (*ground truth*).

2. Perhitungan *gradien error* pada *output layer* untuk melihat seberapa besar selisih antara hasil prediksi model dan label sebenarnya (*ground truth*).

$$\delta_2 = o - y \quad (3.16)$$

Gradien error ini digunakan untuk mengetahui seberapa besar selisih antara hasil prediksi model dan label sebenarnya. Nilai ini akan membantu model dalam proses *training*, khususnya saat memperbarui bobot dan bias di

hidden layer sebelumnya agar hasil prediksi berikutnya menjadi lebih akurat.

3. Perhitungan turunan bobot dan bias pada *hidden layer 2*

a. Turunan bobot V dari *hidden layer 1* ke *hidden layer 2*

$$\frac{\partial L}{\partial v} = h^T \times \delta_2 \quad (3.17)$$

Turunan bobot V yang menghubungkan *hidden layer 1* ke *hidden layer 2*. Turunan ini dihitung dengan mengalikan *output* dari *hidden layer 1* dengan nilai *error* δ_2 . Hasil dari turunan ini digunakan untuk mengetahui seberapa besar perubahan bobot yang dibutuhkan agar model dapat belajar memperbaiki kesalahan prediksi.

b. Turunan terhadap bias c pada *hidden layer 2*

$$\frac{\partial L}{\partial c} = \delta_2 \quad (3.18)$$

Turunan bias c diambil langsung dari δ_2 , karena bias c tidak dikalikan dengan *input*. Oleh karena itu, penyesuaiannya hanya bergantung pada besar kecilnya *error* yang terjadi pada layer tersebut.

4. Perhitungan *gradien error* di *hidden layer 1* serta turunan bobot dan bias

a. *Gradien error* di *hidden layer 1*

$$\delta_1 = (V^T \times \delta_2) \times h_i \quad (3.19)$$

Keterangan :

δ_1 = *gradien error* pada *hidden layer 1*

V^T = nilai *transpose* dari bobot pada *hidden layer 1* ke *hidden layer 2*

Gradien error δ_1 ini digunakan untuk mengetahui seberapa besar *error* yang diteruskan dari *hidden layer 2* ke *hidden layer 1*. Nilai ini diperoleh dengan menyebarkan kembali *error* dari *output layer* ke *hidden layer 1* melalui bobot V , lalu dikalikan dengan turunan dari fungsi aktivasi di *hidden layer 1*. Hasil dari δ_1 digunakan dalam proses pembaruan bobot dan bias pada *hidden layer 1* agar model dapat belajar dan menghasilkan prediksi yang lebih akurat pada proses selanjutnya.

- b. Turunan bobot W yang menghubungkan *input layer* ke *hidden layer 1*

$$\frac{\partial L}{\partial w} = x^T \times \delta_1 \quad (3.20)$$

Keterangan :

$\frac{\partial L}{\partial w}$ = turunan bobot w pada *hidden layer 1*

x^T = nilai *transpose* dari *input layer*

- c. Turunan terhadap bias b pada *hidden layer 1*

$$\frac{\partial L}{\partial b} = \delta_1 \quad (3.21)$$

Turunan bobot W dan bias b pada *hidden layer 1* dihitung untuk menentukan besarnya perubahan parameter yang dibutuhkan dalam proses pembaruan model. Turunan bobot W diperoleh dari hasil perkalian antara *input* yang telah ditransposisikan x^T dengan nilai *gradien error* pada *hidden layer 1* δ_1 . Sementara itu, turunan bias b diambil langsung dari nilai *gradien error* tersebut. Dikarenakan *hidden layer 1* adalah lapisan pertama yang mengolah *input*, perubahan bobot

W dan bias b di *layer* ini sangat berpengaruh terhadap hasil prediksi model.

5. Proses terakhir, *update* parameter bobot dan bias

$$V_{baru} = V_{lama} - (learning\ rate \times \frac{\partial L}{\partial V}) \quad (3.22)$$

$$c_{baru} = c_{lama} - (learning\ rate \times \frac{\partial L}{\partial c}) \quad (3.23)$$

$$W_{baru} = W_{lama} - (learning\ rate \times \frac{\partial L}{\partial W}) \quad (3.24)$$

$$b_{baru} = b_{lama} - (learning\ rate \times \frac{\partial L}{\partial b}) \quad (3.25)$$

Keterangan :

V_{baru} = nilai bobot baru pada *hidden layer* 1 dan *hidden layer* 2 setelah pembaruan

W_{baru} = nilai bobot baru pada *input layer* dan *hidden layer* 1 setelah pembaruan

V_{lama} = nilai bobot pada *hidden layer* 1 dan *hidden layer* 2 sebelum diperbaruan

W_{lama} = nilai bobot pada *input layer* dan *hidden layer* 1 sebelum diperbaruan

c_{baru} = nilai bias baru pada *hidden layer* 1 dan *hidden layer* 2 setelah pembaruan

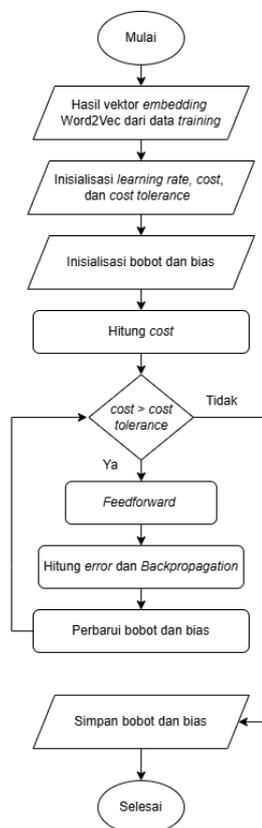
b_{baru} = nilai bias baru pada *input layer* dan *hidden layer* 1 setelah pembaruan

c_{lama} = nilai bias pada *hidden layer* 1 dan *hidden layer* 2 sebelum pembaruan

b_{lama} = nilai bias pada *input layer* dan *hidden layer* 1 sebelum pembaruan

Langkah terakhir dalam proses *training* adalah memperbarui bobot dan bias di setiap *layer*. Pembaruan ini dilakukan dengan mengurangi nilai bobot dan bias sebelumnya menggunakan hasil perkalian antara *learning rate* dan turunan dari fungsi *loss* terhadap bobot atau bias. *Learning rate* sendiri mengatur seberapa besar perubahan yang dilakukan pada parameter. Tujuan dari proses ini adalah agar model dapat memperbaiki kesalahan prediksi sebelumnya, sehingga hasil prediksi menjadi semakin akurat. Proses pembaruan ini dilakukan secara berulang selama pelatihan hingga model mendapatkan parameter yang paling optimal.

Alur dari proses *training* dimulai dari perhitungan *error* berdasarkan selisih antara nilai prediksi dan *ground truth*. Nilai *error* ini kemudian digunakan untuk menghitung *gradien*, yaitu nilai yang menunjukkan seberapa besar perubahan yang perlu dilakukan pada parameter model. *Gradien* ini menjadi dasar dalam proses pembaruan bobot dan bias yang dilakukan secara berulang dengan tujuan untuk meminimalkan *error* dan meningkatkan akurasi model secara bertahap. Alur proses *training* dalam *Neural Network* dapat dilihat pada Gambar 3.14.



Gambar 3.14 Diagram alur proses *training* dalam *Neural Network*

Gambar 3.14 memperlihatkan tahapan utama dalam proses *training Neural Network*, mulai dari perhitungan *error* hingga pembaruan bobot dan bias secara berulang untuk meningkatkan akurasi prediksi.

BAB IV

UJI COBA DAN PEMBAHASAN

4.1 Persiapan Data Uji

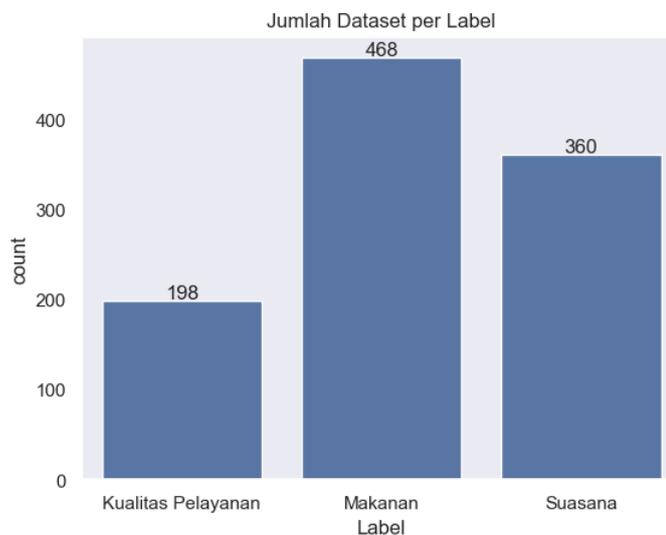
Pada penelitian ini, data yang digunakan berasal dari komentar pelanggan di *Google Review* mengenai restoran-restoran di Indonesia. Namun, data yang diperoleh masih berupa data mentah yang belum memiliki label atau *ground truth*. Untuk memastikan objektivitas pelabelan, survei dilakukan dengan melibatkan responden umum guna menetapkan label yang mencerminkan persepsi mayoritas. Rincian jumlah ulasan pelanggan restoran yang telah dilabeli berdasarkan kategori yang ditetapkan melalui hasil survei ditampilkan pada Tabel 4.1.

Tabel 4.1 Distribusi ulasan berdasarkan label

Label	Jumlah
Kualitas Pelayanan	198
Makanan	468
Suasana	360
Total	1026

Total responden : 135

Untuk melihat bagaimana sebaran jumlah data pada setiap kategori label, dilakukan pengecekan awal terhadap banyaknya ulasan di masing-masing aspek. Hasil visualisasi dari distribusi data tersebut ditampilkan pada Gambar 4.1.



Gambar 4.1 Distribusi jumlah dataset berdasarkan label

Berdasarkan Gambar 4.1, terlihat bahwa jumlah data pada masing-masing label tidak seimbang (*imbalanced*), di mana kategori makanan memiliki jumlah ulasan paling banyak dibandingkan 2 kategori lainnya, yaitu suasana dan kualitas pelayanan. Ketidakseimbangan jumlah data ini menjadi hal yang penting untuk diperhatikan, karena dapat memengaruhi kinerja model klasifikasi, terutama dalam mengenali label dengan jumlah data yang lebih sedikit. Oleh karena itu, pada tahap pelatihan dan evaluasi, distribusi label ini menjadi pertimbangan untuk memastikan bahwa hasil prediksi model tetap adil dan representatif terhadap seluruh kategori.

Untuk memastikan label yang diberikan konsisten dan sesuai, peneliti membuat panduan pelabelan berdasarkan hasil survei yang telah dilakukan. Survei ini melibatkan 135 responden yang berasal dari berbagai latar belakang, seperti pelajar atau mahasiswa, pegawai negeri, pengusaha, dan masyarakat umum lainnya. Rentang usia responden juga beragam, mulai dari umur dibawah 17 tahun hingga

di atas 45 tahun. Keberagaman ini diharapkan dapat memberikan sudut pandang yang lebih akurat terhadap penilaian isi ulasan.

Hasil survei menunjukkan bahwa mayoritas responden sepakat terhadap 3 aspek utama yang paling sering menjadi fokus perhatian dalam ulasan restoran, yaitu kualitas pelayanan, makanan, dan suasana. Secara rinci, sebanyak 71,9% responden menyebutkan aspek kualitas pelayanan, 85,9% menyebutkan aspek makanan, dan 50,4% menyebutkan aspek suasana sebagai hal yang paling mereka perhatikan saat menilai restoran. Selain ketiga aspek tersebut, terdapat pula responden yang menyebutkan aspek lain seperti harga dan kebersihan. Namun, persentase penyebutan aspek-aspek tersebut tergolong kecil dan tidak terlalu dominan dalam persepsi responden. Dukungan data survei memperkuat bahwa aspek-aspek tersebut sesuai dengan persepsi masyarakat dan valid dijadikan acuan pelabelan data ulasan.

4.2 Skenario Pengujian

Tahapan selanjutnya adalah skenario pengujian untuk mengevaluasi performa model. Pada penelitian ini, pengujian dilakukan dengan menggunakan beberapa skenario yang berfokus pada variasi jumlah *node* pada *hidden layer* 1 dan nilai *learning rate*. Seluruh skenario pengujian yang ditetapkan dalam penelitian ini ditampilkan pada Tabel 4.2

Tabel 4.2 Skenario pengujian

Model	Jumlah node <i>Hidden layer 1</i>	<i>Learning rate</i>
P1	25	0.1
P2	25	0.01
P3	25	0.001
Q1	50	0.1
Q2	50	0.01
Q3	50	0.001
R1	100	0.1
R2	100	0.01
R3	100	0.001

Pada setiap skenario, jumlah *node hidden layer 1* diatur menjadi 25, 50 dan 100, dengan variasi nilai *learning rate* sebesar 0.1, 0.01, dan 0.001. Berdasarkan penelitian Mirayanti et al (2023), nilai *learning rate* sebesar 0,01 menghasilkan performa terbaik dengan akurasi dan stabilitas yang tinggi. Meskipun demikian, penggunaan *learning rate* yang terlalu besar berisiko menyebabkan ketidakstabilan dalam proses pelatihan, sedangkan *learning rate* yang terlalu kecil dapat memperlambat laju konvergensi model. Oleh karena itu, penelitian ini memilih nilai 0,1 untuk menguji kecepatan konvergensi, sedangkan nilai 0,01 dan 0,001 digunakan untuk mengamati kestabilan proses *training* dengan laju perubahan bobot yang lebih terkontrol. Pengaturan ini bertujuan untuk mengevaluasi pengaruh *learning rate* terhadap kinerja model.

Tahap selanjutnya adalah pembagian dataset untuk pelatihan dan pengujian model. Dalam penelitian ini, data dibagi menjadi 2 bagian, yaitu data latih (*training*) dan data uji (*testing*), dengan rasio 80:20. Rasio ini diterapkan secara konsisten pada semua skenario pengujian, di mana 821 data digunakan untuk pelatihan dan 205 data digunakan untuk pengujian.

Dalam penelitian ini, teknik *early stopping* diterapkan untuk menghentikan proses pelatihan model ketika perubahan nilai *cost* tidak signifikan lagi. Proses pelatihan akan terus berjalan sampai grafik *cost* menunjukkan konvergensi, yang berarti perubahan *cost* tidak lagi terjadi dalam waktu yang cukup lama. Pelatihan akan dihentikan jika selisih perubahan nilai *cost* antar *epoch* mencapai 0,000001, yang menandakan bahwa model tidak lagi mengalami penurunan *cost* yang signifikan. Informasi mengenai rasio pembagian data dan pemberhentian pelatihan yang dipakai pada penelitian ini ditampilkan pada Tabel 4.3.

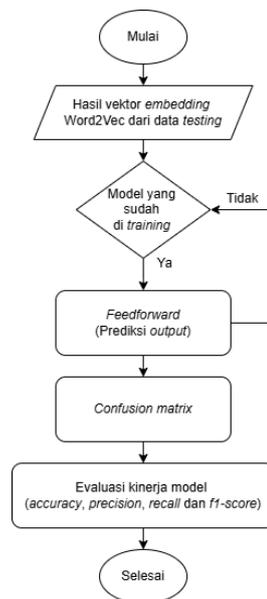
Tabel 4.3 Rasio pembagian data dan kriteria pemberhentian pelatihan

Parameter	Nilai
Rasio pembagian data <i>training</i> dan data <i>testing</i>	8:2 (821 data <i>training</i> dan 205 data <i>testing</i>)
<i>Early stopping</i> (selisih <i>cost</i> antar <i>epoch</i>)	0,000001

Sebagai perbandingan, menurut penelitian Ji et al (2021) meskipun tidak ada ketentuan pasti mengenai nilai selisih *cost*, mereka memilih nilai 0,00001 untuk menghentikan pelatihan lebih awal tanpa mengurangi kualitas model. Teknik ini bertujuan mempercepat pelatihan dan mencegah *overfitting*, sehingga menghasilkan model klasifikasi yang lebih efisien dan optimal.

Setelah proses *training* selesai dan bobot serta bias pada model diperbarui sesuai parameter dari tiap skenario, langkah selanjutnya adalah pengujian model menggunakan data uji. Pengujian dilakukan menggunakan data uji yang belum pernah dipelajari sebelumnya oleh model. Data uji yang telah melalui proses *preprocessing* dan dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram kemudian diberikan kepada model untuk dilakukan pengujian. Pada tahap ini, model tidak lagi melakukan proses *backpropagation*, melainkan hanya menerapkan

proses *feedforward* untuk menghasilkan prediksi berdasarkan parameter yang telah dipelajari pada tahap *training*. Alur proses *testing* menggunakan *Neural Network* dapat dilihat pada Gambar 4.2



Gambar 4.2 Diagram alur proses *testing* dalam *Neural Network*

Untuk menilai kinerja model, digunakan *confusion matrix* yang merangkum jumlah prediksi benar dan salah. *Confusion matrix* terdiri dari empat kategori, yaitu *True Positive* (TP), ketika model benar memprediksi data ulasan ke kategori yang tepat, *True Negative* (TN) ketika model benar memprediksi data ulasan tetapi bukan ke dalam kategori yang tepat, *False Positive* (FP) ketika model salah memprediksi data ulasan ke kategori yang tidak sesuai, dan *False Negative* (FN) ketika model salah memprediksi data ulasan dan mengabaikan kategori yang tepat. Hasil dari *confusion matrix* ini kemudian digunakan untuk mengukur kinerja model dalam mengklasifikasikan aspek-aspek ulasan pelanggan restoran, di mana A mewakili kategori kualitas pelayanan, B untuk makanan, dan C untuk suasana.

Pada masalah klasifikasi *multiclass* atau dengan lebih dari dua kategori, satu kelas akan dianggap sebagai kelas positif, sedangkan kelas lainnya dianggap sebagai kelas negatif. Hasil *confusion matrix* dalam penelitian ini ditampilkan pada Tabel 4.4.

Tabel 4.4 Hasil *confusion matrix*

Kelas Aktual	Kelas Prediksi		
	Kualitas Pelayanan (A)	Makanan (B)	Suasana (C)
Kualitas Pelayanan (A)	AA	AB	AC
Makanan (B)	BA	BB	BC
Suasana (C)	CA	CB	CC

Dalam penelitian ini, model klasifikasi menghasilkan 3 kategori yang telah dijelaskan sebelumnya. Setiap kategori ini dianalisis dengan menggunakan nilai *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN) untuk menilai kinerja setiap kelas secara terpisah. Rincian hasil *confusion matrix* untuk masing-masing kategori ditampilkan pada Tabel 4.5, Tabel 4.6 dan Tabel 4.7.

Tabel 4.5 *Confusion matrix* kategori kualitas pelayanan

Kelas Aktual	Kelas Prediksi	
	Kualitas Pelayanan (A)	Bukan Kualitas Pelayanan
Kualitas Pelayanan (A)	<i>True positive</i> (AA)	<i>False Negative</i> (AB + AC)
Bukan Kualitas Pelayanan	<i>False Positive</i> (BA + CA)	<i>True Negative</i> (BB + BC + CB + CC)

Tabel 4.6 *Confusion matrix* kategori makanan

Kelas Aktual	Kelas Prediksi	
	Makanan (B)	Bukan Makanan
Makanan (B)	<i>True positive</i> (BB)	<i>False Negative</i> (BA + BC)
Bukan Makanan	<i>False Positive</i> (AB + CB)	<i>True Negative</i> (AA + CA + AC + CC)

Tabel 4.7 *Confusion matrix* kategori suasana

Kelas Aktual	Kelas Prediksi	
	Suasana (C)	Bukan Suasana
Suasana (C)	<i>True positive</i> (CC)	<i>False Negative</i> (CA + CB)
Bukan Suasana	<i>False Positive</i> (AC + BC)	<i>True Negative</i> (AA + AB + BA + BB)

Hasil dari *confusion matrix* dapat digunakan untuk mengukur nilai *accuracy*, *precision*, *recall*, dan *f1-score* model dengan menggunakan rumus pada persamaan 4.1, 4.2, 4.3 dan 4.4.

$$Accuracy = \frac{TP}{TP+TN+FP+FN} \quad (4.1)$$

$$Precision = \frac{TP}{TP+FP} \quad (4.2)$$

$$Recall = \frac{TP}{TP+FN} \quad (4.3)$$

$$F1 - Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (4.4)$$

Keempat metrik tersebut digunakan untuk mengevaluasi kinerja model klasifikasi. *Accuracy* menunjukkan seberapa besar persentase ulasan yang berhasil diklasifikasikan dengan benar oleh model dari seluruh data ulasan yang tersedia. *Precision* mengukur sejauh mana prediksi model terhadap ulasan positif benar-benar tepat atau sesuai dengan kenyataannya. *Recall* digunakan untuk mengetahui kemampuan model dalam mengenali seluruh ulasan positif yang sebenarnya terdapat dalam data. Sementara itu, *F1-score* digunakan untuk menilai keseimbangan antara *precision* dan *recall*, terutama ketika data memiliki distribusi kelas yang tidak seimbang, sehingga dapat memberikan gambaran performa model secara lebih menyeluruh.

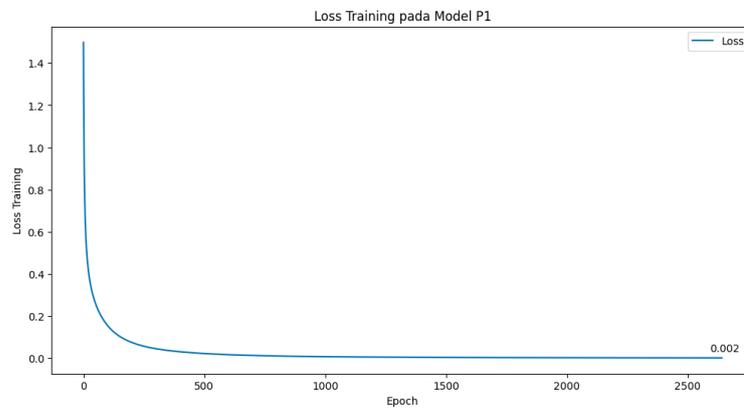
4.2 Hasil Uji Coba

Berdasarkan skenario pengujian, penelitian ini melakukan 9 kali uji coba dengan variasi jumlah *node hidden layer* 1 dan nilai *learning rate* yang berbeda. Tujuan dari uji coba ini adalah untuk menemukan kombinasi parameter terbaik yang menghasilkan model klasifikasi paling optimal dalam mengelompokkan aspek-aspek ulasan pelanggan restoran. Hasil prediksi klasifikasi kemudian dibandingkan dengan label aktual dan disusun dalam bentuk *confusion matrix*, yang merangkum jumlah prediksi benar dan salah untuk masing-masing kelas. Setiap model yang dihasilkan dievaluasi kinerjanya menggunakan *accuracy*, *precision*, *recall*, dan *f1-score*, yang dihitung berdasarkan perbandingan antara kelas aktual dengan kelas prediksi. Evaluasi ini bertujuan untuk mengukur tingkat ketepatan model dalam mengklasifikasikan data ulasan dengan benar maupun salah, serta menilai sejauh mana model mampu mengenali dan membedakan masing-masing kategori ulasan secara konsisten.

4.2.1 Model P1

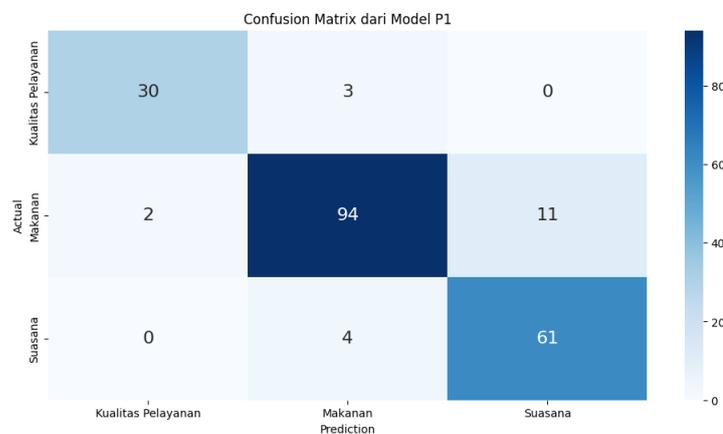
Uji coba model P1 dilakukan dengan menggunakan 25 *node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.1. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model P1 berlangsung selama 6,82 detik dengan total *epoch*. Setelah proses pelatihan

selesai, diperoleh nilai *cost* sebesar 0,002. Grafik perubahan nilai *cost* selama pelatihan model P1 ditampilkan pada Gambar 4.3.



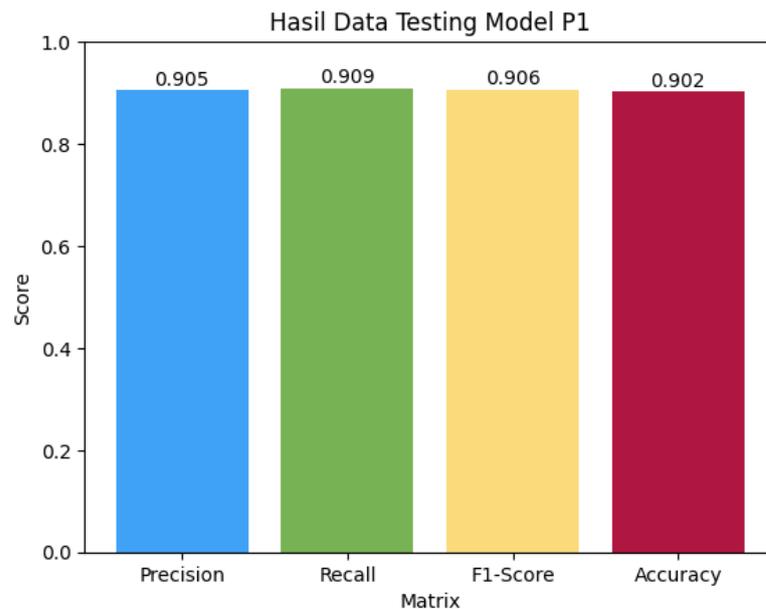
Gambar 4.3 Grafik *cost* proses *training* pada model P1

Setelah proses pelatihan selesai, model P1 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model P1 untuk data *testing* ditampilkan pada Gambar 4.4. Berdasarkan hasil tersebut, model P1 mampu memprediksi kelas dengan benar sebanyak 185 data, sedangkan 20 data lainnya salah prediksi.



Gambar 4.4 *Confusion matrix* model P1

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.5 untuk menunjukkan analisis kinerja model.

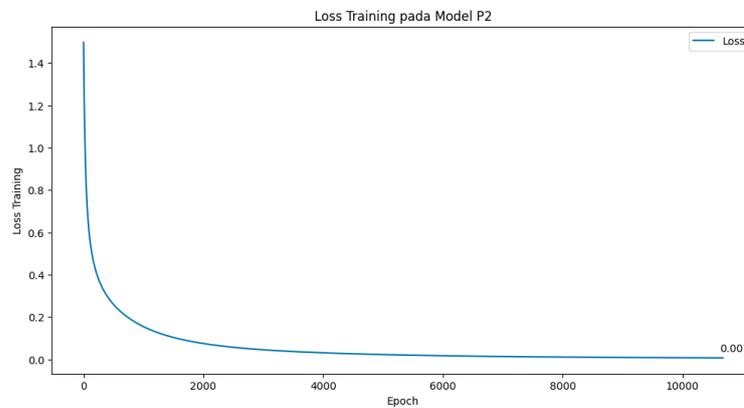


Gambar 4.5 Hasil performa model P1

4.2.2 Model P2

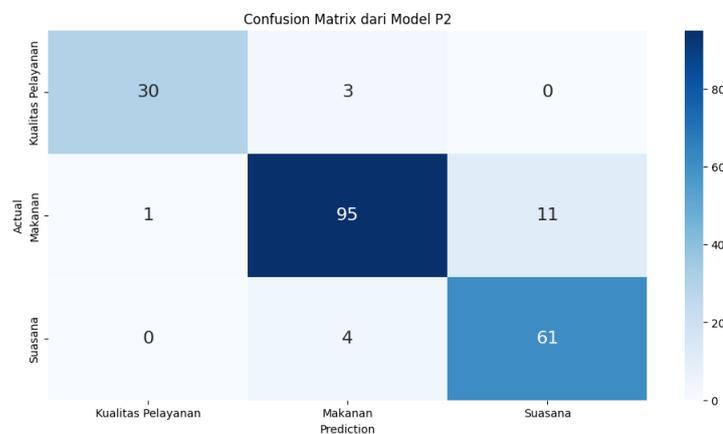
Uji coba model P2 dilakukan dengan menggunakan *25 node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.01. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model P2 berlangsung selama 17,44 detik dengan total 10675 *epoch*.

Setelah proses pelatihan selesai, diperoleh nilai *cost* sebesar 0,007. Grafik perubahan nilai *cost* selama pelatihan model P2 ditampilkan pada Gambar 4.6.



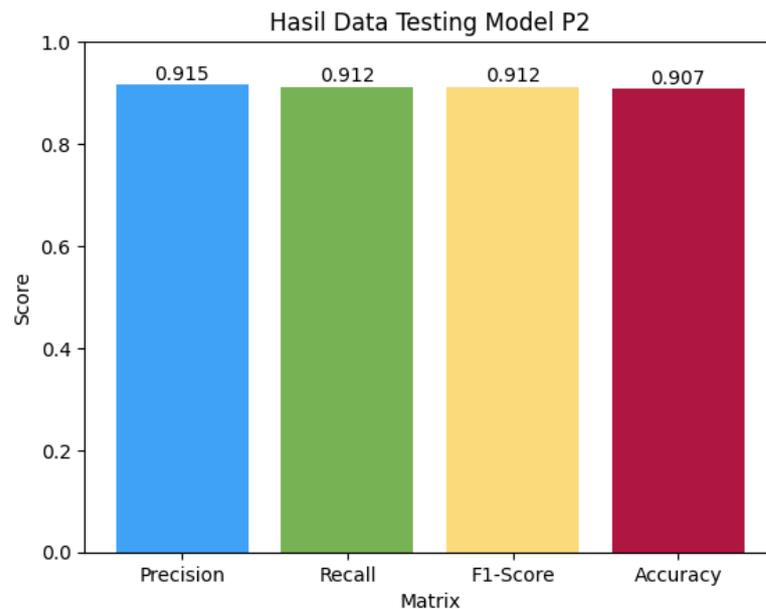
Gambar 4.6 Grafik *cost* proses *training* pada model P2

Setelah proses pelatihan selesai, model P2 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model P2 untuk data *testing* ditampilkan pada Gambar 4.7. Berdasarkan hasil tersebut, model P2 mampu memprediksi kelas dengan benar sebanyak 186 data, sedangkan 19 data lainnya salah prediksi.



Gambar 4.7 *Confusion matrix* model P2

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.8 untuk menunjukkan analisis kinerja model.

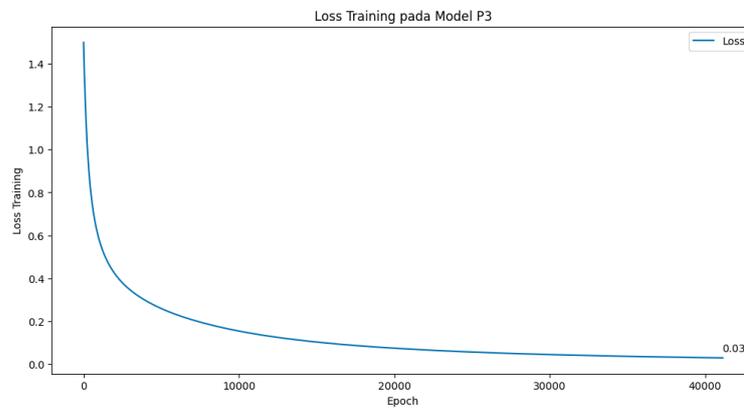


Gambar 4.8 Hasil performa model P2

4.2.3 Model P3

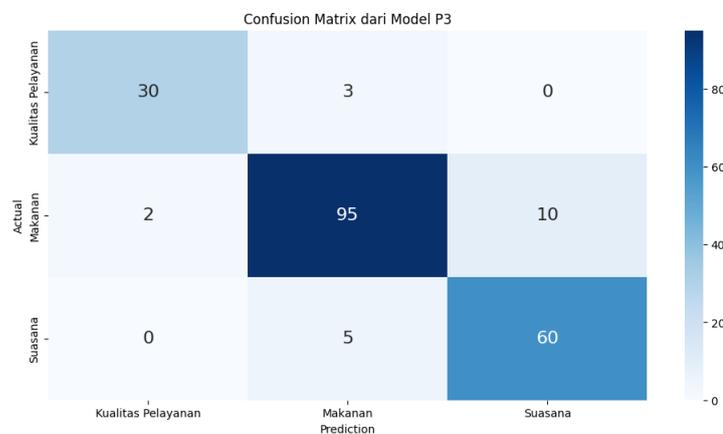
Uji coba model P3 dilakukan dengan menggunakan 25 *node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.001. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model P3 berlangsung selama 76,60 detik dengan total 41141 *epoch*.

Setelah proses pelatihan selesai, diperoleh nilai *cost* sebesar 0,030. Grafik perubahan nilai *cost* selama pelatihan model P3 ditampilkan pada Gambar 4.9.



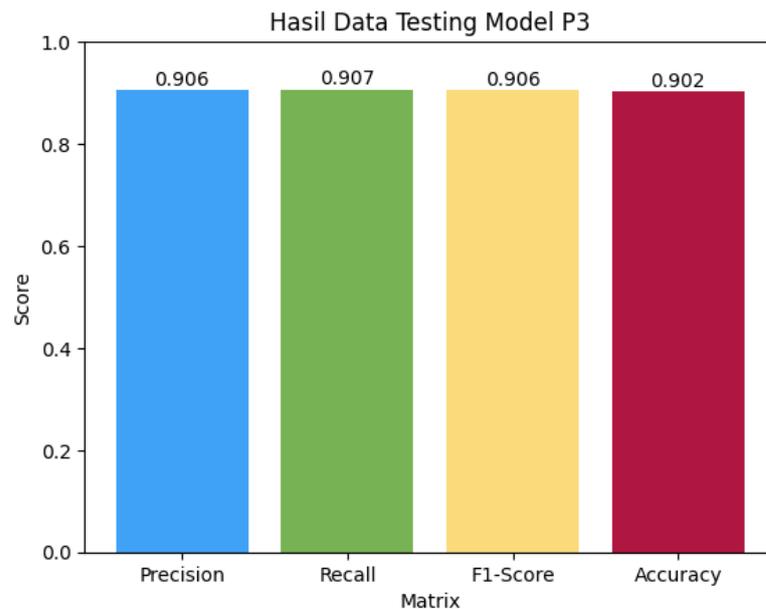
Gambar 4.9 Grafik *cost* proses *training* pada model P3

Setelah proses pelatihan selesai, model P3 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model P3 untuk data *testing* ditampilkan pada Gambar 4.10. Berdasarkan hasil tersebut, model P3 mampu memprediksi kelas dengan benar sebanyak 185 data, sedangkan 20 data lainnya salah prediksi.



Gambar 4.10 *Confusion matrix* model P3

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.11 untuk menunjukkan analisis kinerja model.

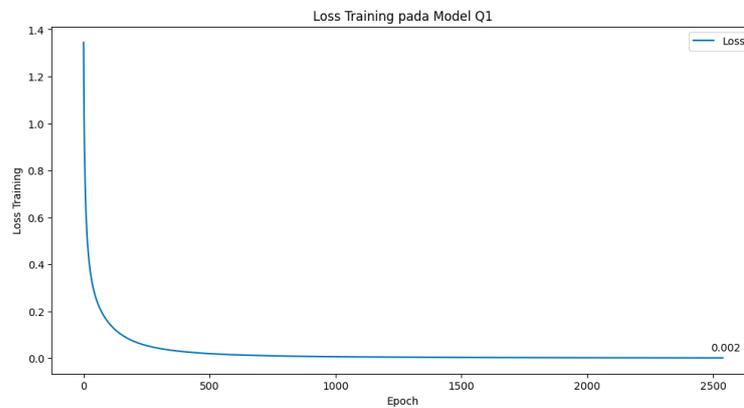


Gambar 4.11 Hasil performa model P3

4.2.4 Model Q1

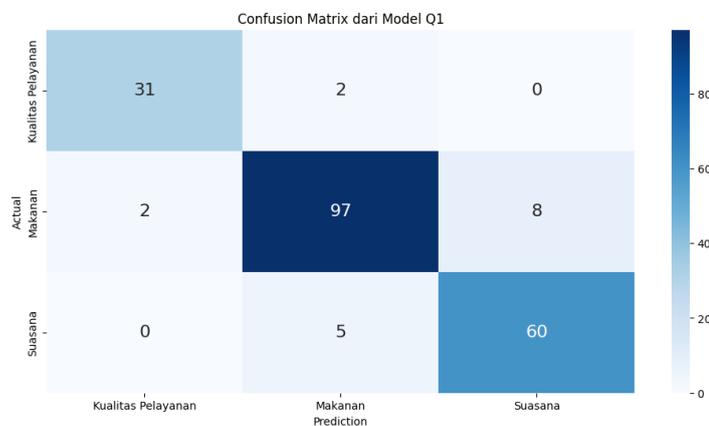
Uji coba model Q1 dilakukan dengan menggunakan 50 *node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.1. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model Q1 berlangsung selama 8,22 detik dengan total 2540 *epoch*. Setelah proses

pelatihan selesai, diperoleh nilai *cost* sebesar 0,002. Grafik perubahan nilai *cost* selama pelatihan model Q1 ditampilkan pada Gambar 4.12.



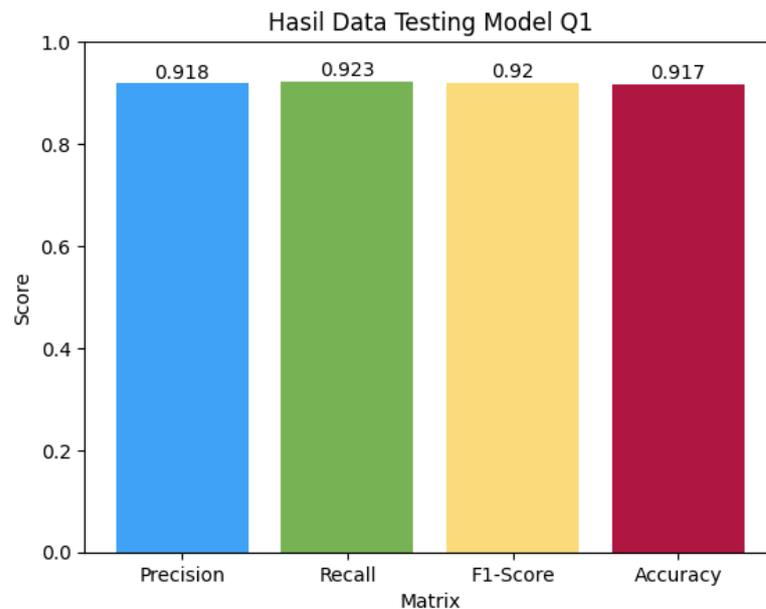
Gambar 4.12 Grafik *cost* proses *training* pada model Q1

Setelah proses pelatihan selesai, model Q1 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model Q1 untuk data *testing* ditampilkan pada Gambar 4.13. Berdasarkan hasil tersebut, model Q1 mampu memprediksi kelas dengan benar sebanyak 188 data, sedangkan 17 data lainnya salah prediksi.



Gambar 4.13 *Confusion matrix* model Q1

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.14 untuk menunjukkan analisis kinerja model.

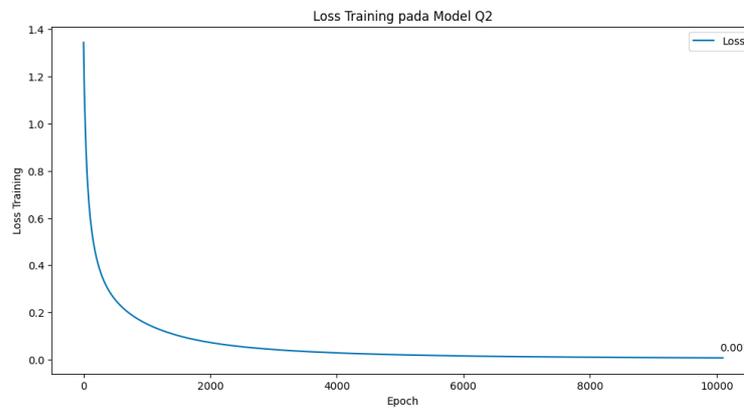


Gambar 4.14 Hasil performa model Q1

4.2.5 Model Q2

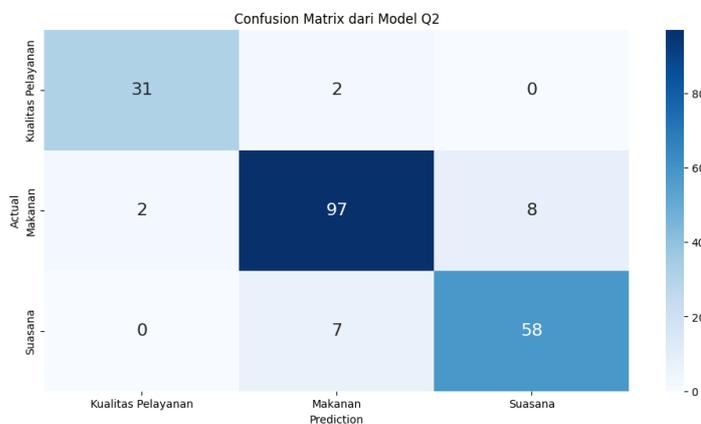
Uji coba model Q2 dilakukan dengan menggunakan 50 *node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.01. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model Q2 berlangsung selama 20,93 detik dengan total 10098 *epoch*.

Setelah proses pelatihan selesai, diperoleh nilai *cost* sebesar 0,007. Grafik perubahan nilai *cost* selama pelatihan model Q2 ditampilkan pada Gambar 4.15.



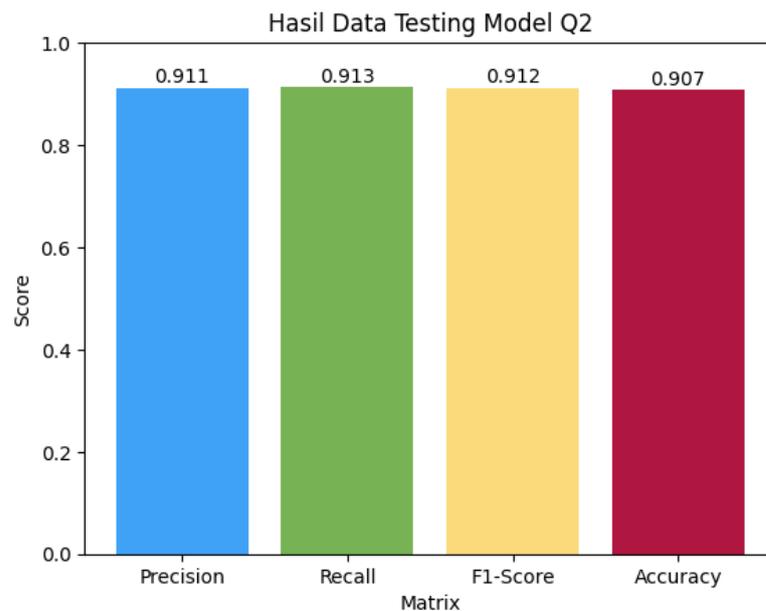
Gambar 4.15 Grafik *cost* proses *training* pada model Q2

Setelah proses pelatihan selesai, model Q2 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model Q2 untuk data *testing* ditampilkan pada Gambar 4.16. Berdasarkan hasil tersebut, model Q2 mampu memprediksi kelas dengan benar sebanyak 186 data, sedangkan 19 data lainnya salah prediksi.



Gambar 4.16 *Confusion matrix* model Q2

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.17 untuk menunjukkan analisis kinerja model.

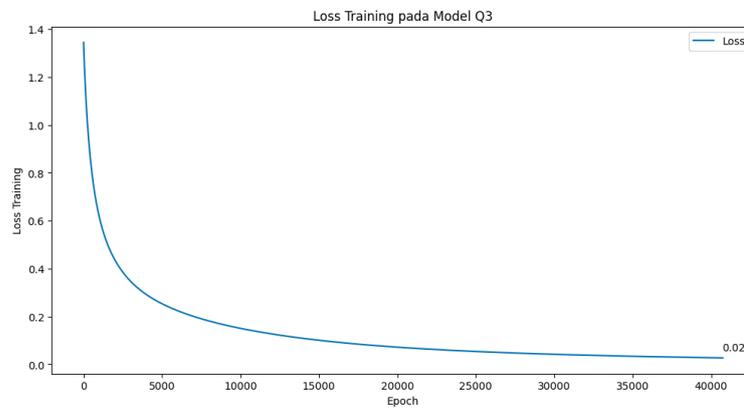


Gambar 4.17 Hasil performa model Q2

4.2.6 Model Q3

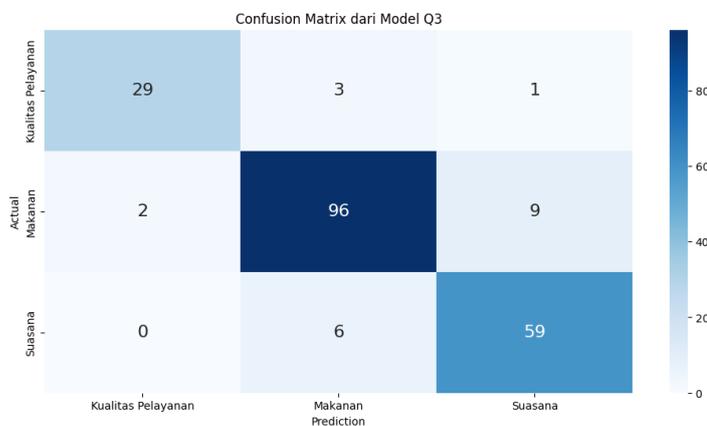
Uji coba model Q3 dilakukan dengan menggunakan 50 *node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.001. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model Q3 berlangsung selama 86,63 detik dengan total 40746 *epoch*.

Setelah proses pelatihan selesai, diperoleh nilai *cost* sebesar 0,027. Grafik perubahan nilai *cost* selama pelatihan model Q3 ditampilkan pada Gambar 4.18.



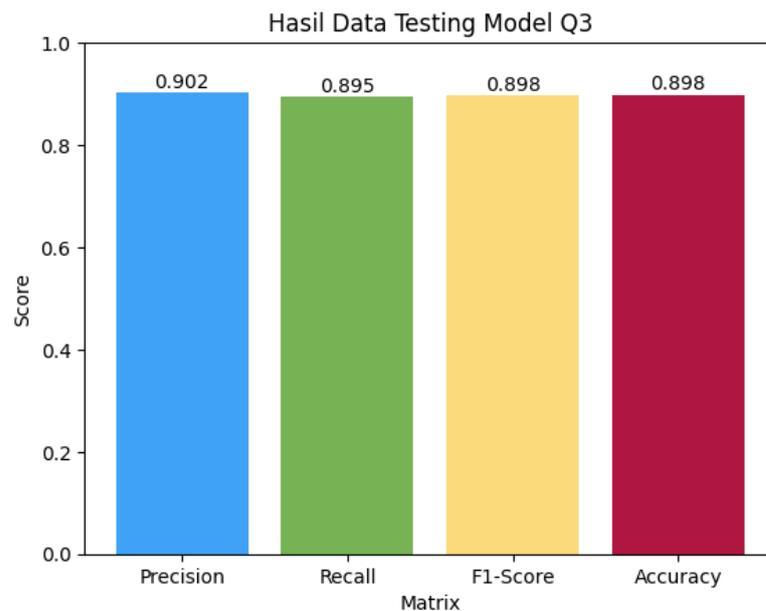
Gambar 4.18 Grafik *cost* proses *training* pada model Q3

Setelah proses pelatihan selesai, model Q3 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model Q3 untuk data *testing* ditampilkan pada Gambar 4.19. Berdasarkan hasil tersebut, model Q3 mampu memprediksi kelas dengan benar sebanyak 184 data, sedangkan 20 data lainnya salah prediksi.



Gambar 4.19 *Confusion matrix* model Q3

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.20 untuk menunjukkan analisis kinerja model.

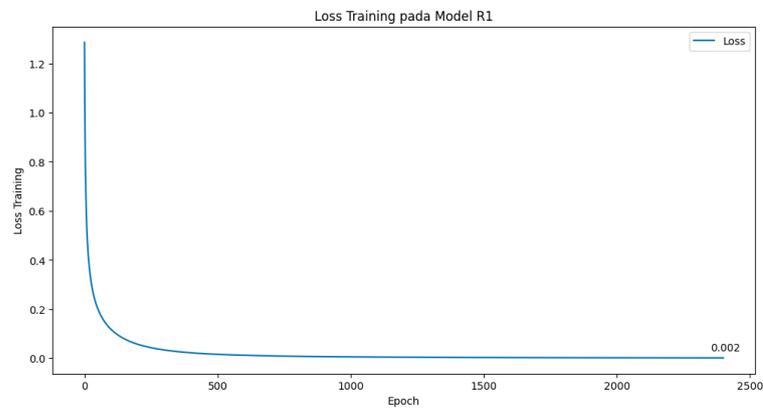


Gambar 4.20 Hasil performa model Q3

4.2.7 Model R1

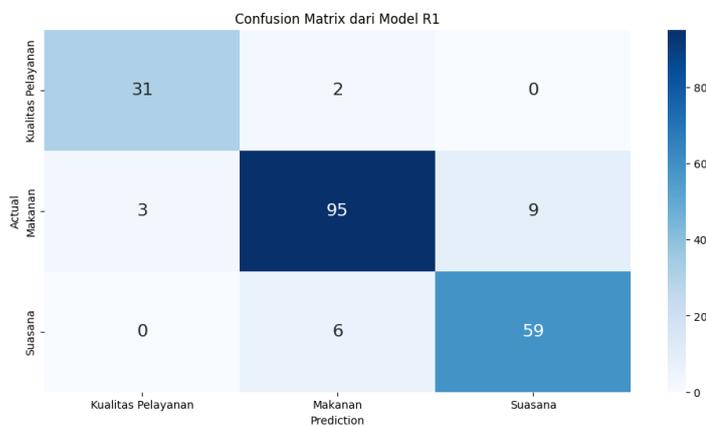
Uji coba model R1 dilakukan dengan menggunakan 100 *node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.1. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model R1 berlangsung selama 11,81 detik dengan total 2401 *epoch*. Setelah proses

pelatihan selesai, diperoleh nilai *cost* sebesar 0,002. Grafik perubahan nilai *cost* selama pelatihan model R1 ditampilkan pada Gambar 4.21.



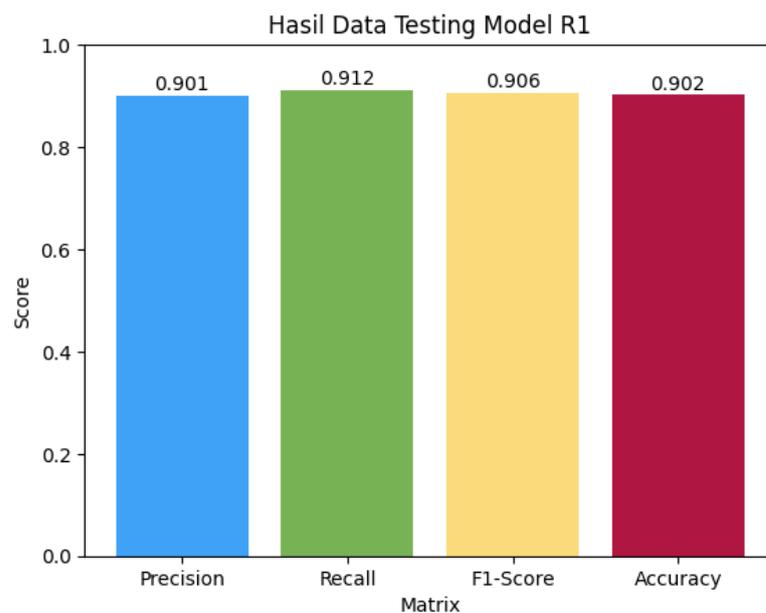
Gambar 4.21 Grafik *cost* proses *training* pada model R1

Setelah proses pelatihan selesai, model R1 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model R1 untuk data *testing* ditampilkan pada Gambar 4.22. Berdasarkan hasil tersebut, model R1 mampu memprediksi kelas dengan benar sebanyak 185 data, sedangkan 20 data lainnya salah prediksi.



Gambar 4.22 *Confusion matrix* model R1

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.23 untuk menunjukkan analisis kinerja model.

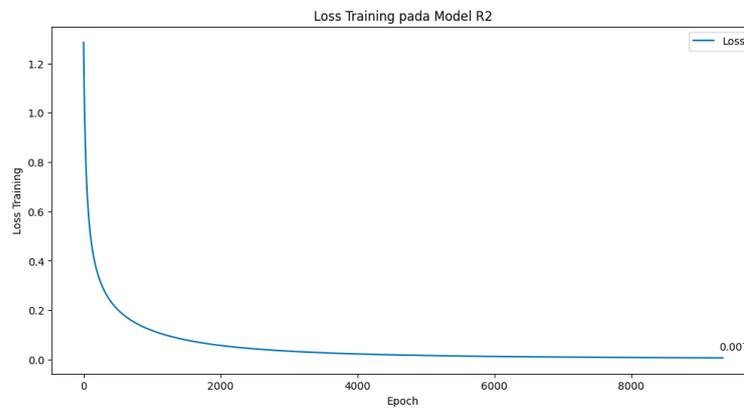


Gambar 4.23 Hasil performa model R1

4.2.8 Model R2

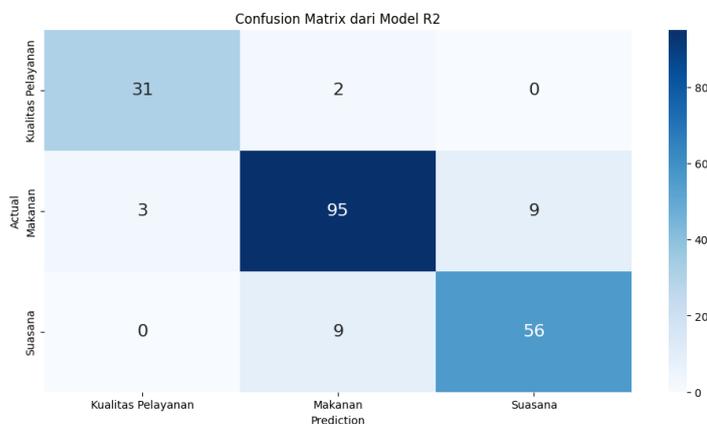
Uji coba model R2 dilakukan dengan menggunakan 100 *node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.01. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model R2 berlangsung selama 48,89 detik dengan total 9341 *epoch*.

Setelah proses pelatihan selesai, diperoleh nilai *cost* sebesar 0,007. Grafik perubahan nilai *cost* selama pelatihan model R2 ditampilkan pada Gambar 4.24.



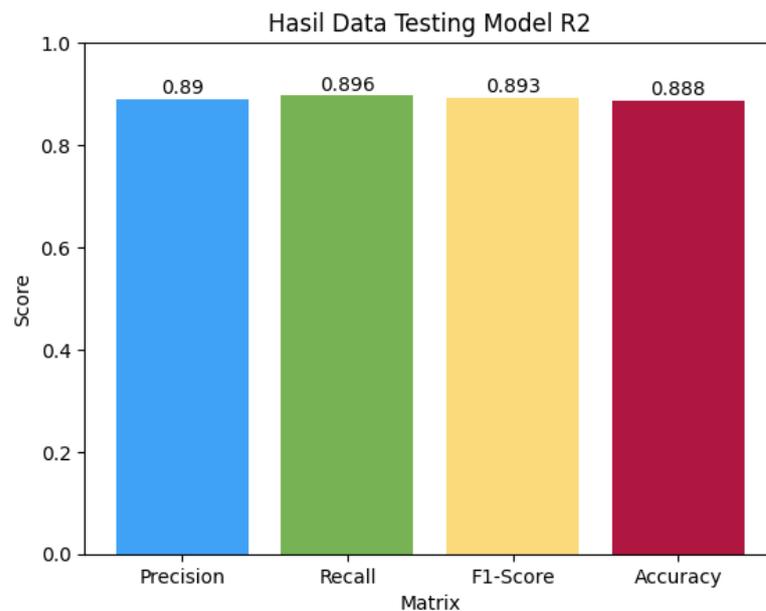
Gambar 4.24 Grafik *cost* proses *training* pada model R2

Setelah proses pelatihan selesai, model R2 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model R2 untuk data *testing* ditampilkan pada Gambar 4.25. Berdasarkan hasil tersebut, model R2 mampu memprediksi kelas dengan benar sebanyak 182 data, sedangkan 23 data lainnya salah prediksi.



Gambar 4.25 *Confusion matrix* model R2

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.26 untuk menunjukkan analisis kinerja model.

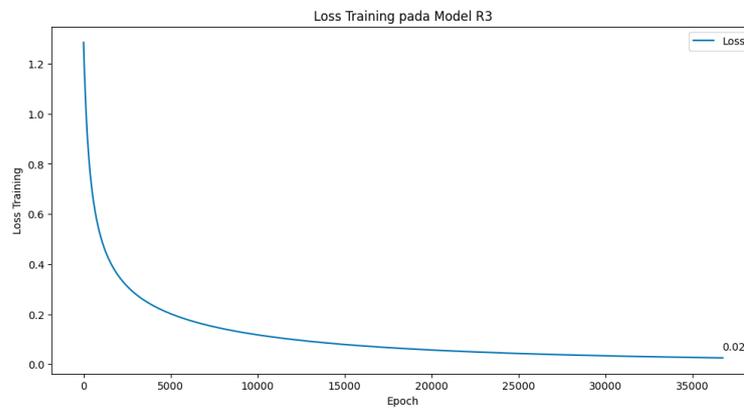


Gambar 4.26 Hasil performa model R2

4.2.9 Model R3

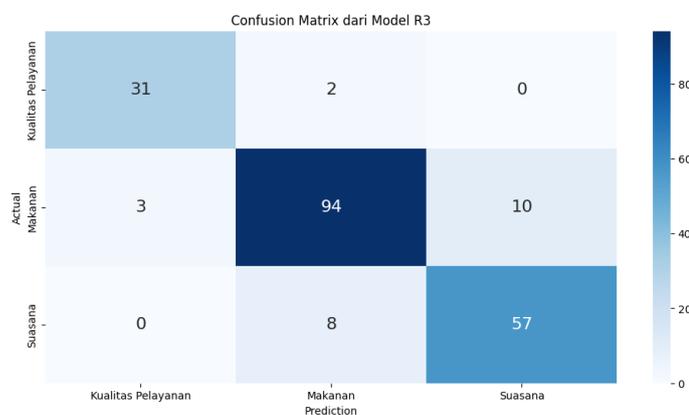
Uji coba model R3 dilakukan dengan menggunakan 100 *node* pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.001. Proses pelatihan model ini menggunakan data *training* yang telah dibagi sebelumnya dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% data sisanya digunakan untuk pengujian. Data *training* yang digunakan dalam pelatihan ini terdiri dari 821 data ulasan, yang telah dikonversi menjadi vektor menggunakan *Word2Vec* Skip-Gram. Pelatihan model R3 berlangsung selama 213,85 detik dengan total 36759 *epoch*.

Setelah proses pelatihan selesai, diperoleh nilai *cost* sebesar 0,026. Grafik perubahan nilai *cost* selama pelatihan model R3 ditampilkan pada Gambar 4.27.



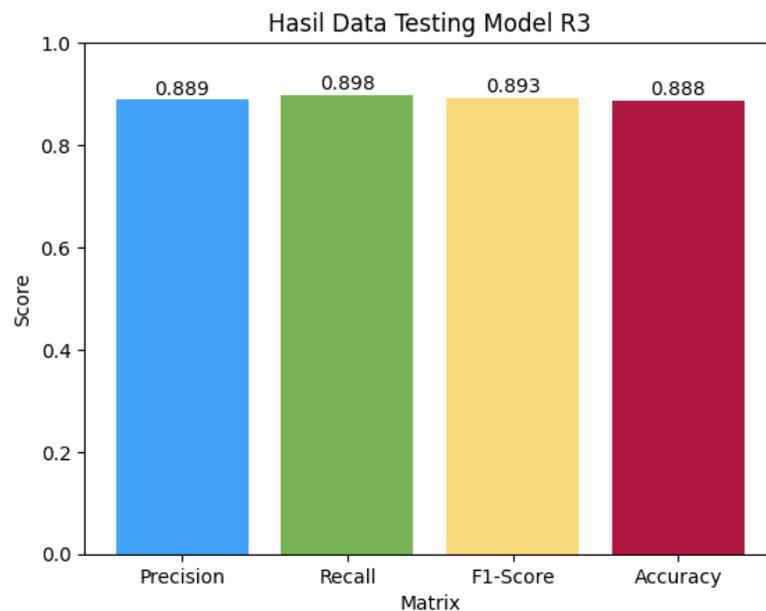
Gambar 4.27 Grafik *cost* proses *training* pada model R3

Setelah proses pelatihan selesai, model R3 diuji menggunakan 205 data *testing*, yaitu data yang sebelumnya tidak dilibatkan dalam proses pelatihan. Pengujian ini bertujuan untuk mengevaluasi sejauh mana model mampu melakukan prediksi terhadap data baru. Hasil *confusion matrix* pada model R3 untuk data *testing* ditampilkan pada Gambar 4.28. Berdasarkan hasil tersebut, model R3 mampu memprediksi kelas dengan benar sebanyak 182 data, sedangkan 23 data lainnya salah prediksi.



Gambar 4.28 *Confusion matrix* model R3

Berdasarkan perhitungan yang dilakukan pada persamaan 4.1, 4.2, 4.3 dan 4.4, diperoleh nilai-nilai yang menggambarkan performa model, seperti *accuracy*, *precision*, *recall* dan *f1-score*. Hasil dari nilai-nilai tersebut kemudian ditampilkan secara visual pada Gambar 4.29 untuk menunjukkan analisis kinerja model.



Gambar 4.29 Hasil performa model R3

4.3 Pembahasan

Berdasarkan hasil skenario pengujian yang telah dilakukan oleh masing-masing model, diperoleh informasi mengenai hasil proses pelatihan (*training*) dan pengujian (*testing*). Hasil dari proses pelatihan untuk setiap model menunjukkan beberapa nilai penting yang menggambarkan performa, yaitu nilai *cost*, jumlah *epoch*, dan durasi waktu pelatihan. Nilai *cost* memberikan gambaran bahwa seberapa baik model dapat beradaptasi dengan data *training*. Selain itu, jumlah *epoch* yang menunjukkan seberapa sering model dilatih menggunakan seluruh data

dan durasi waktu pelatihan sangat penting untuk menilai efisiensi dari proses pelatihan yang dilakukan.

Dalam penelitian ini, teknik *early stopping* diterapkan untuk menghentikan pelatihan saat perubahan nilai *cost* tidak lagi signifikan. Pelatihan dihentikan jika selisih nilai *cost* antara *epoch* saat ini dan sebelumnya lebih kecil dari 0,000001, yang menunjukkan model telah mencapai titik konvergen. Teknik ini membantu mencegah *overfitting*, sehingga model dapat menghasilkan klasifikasi ulasan restoran yang lebih optimal. Hasil proses pelatihan pada setiap model ditampilkan dalam Tabel 4.8.

Tabel 4.8 Hasil *training* dari skenario pengujian

Model	Training		
	Cost	Epoch	Durasi waktu (detik)
P1	0.001902	2643	6,82
P2	0.006938	10675	17,44
P3	0.029797	41141	76,60
Q1	0.001899	2540	8,22
Q2	0.006858	10098	20,93
Q3	0.027350	40746	86,63
R1	0.001834	2401	11,81
R2	0.006593	9341	48,89
R3	0.025501	36759	213,85

Berbagai model dalam penelitian ini dibedakan berdasarkan jumlah *node hidden layer* 1 dan nilai *learning rate*. Kode huruf pada model menunjukkan jumlah *node hidden layer* 1, yaitu P untuk 25 *node*, Q untuk 50 *node* dan R untuk 100 *node*. Sementara itu, kode angka pada model menunjukkan nilai *learning rate*, yaitu 1 untuk 0.1, 2 untuk 0.01 dan 3 untuk 0.001.

Setiap model menunjukkan variasi dalam jumlah *epoch*, nilai *cost*, dan durasi waktu pelatihan. Tabel 4.8 memperlihatkan adanya perbedaan durasi pelatihan yang signifikan antar model. Meskipun jumlah *epoch* cenderung menurun

seiring bertambahnya jumlah *node* pada *hidden layer* 1, durasi waktu pelatihan justru mengalami peningkatan. Hal ini disebabkan oleh semakin kompleksnya struktur *Neural Network* seiring bertambahnya jumlah *node*, yang menghasilkan lebih banyak parameter untuk dihitung dalam setiap *epoch*. Akibatnya, meskipun jumlah *epoch* lebih sedikit, durasi waktu pelatihan secara keseluruhan menjadi lebih lama.

Selain itu, nilai *learnig rate* juga berperan penting dalam menentukan kecepatan konvergensi model. Berdasarkan hasil yang diperoleh, model dengan *learning rate* yang lebih besar (0.1) mampu mencapai nilai *cost* yang lebih rendah dengan jumlah *epoch* yang relatif sedikit dibandingkan model dengan *learning rate* yang lebih kecil. Hal ini menunjukkan bahwa *learning rate* yang besar dapat mempercepat proses penyesuaian bobot dan mempercepat konvergensi. Namun, *learning rate* yang terlalu besar juga memiliki risiko membuat model tidak stabil. Sebaliknya, model dengan *learning rate* yang lebih kecil (0.001) cenderung lebih stabil karena melakukan pembaruan bobot secara lebih hati-hati, tetapi membutuhkan jumlah *epoch* yang lebih banyak dan durasi waktu pelatihan yang lebih lama, serta belum tentu menghasilkan performa akhir yang lebih baik.

Berdasarkan hasil pelatihan, model R1 merupakan model dengan performa yang paling optimal dibandingkan dengan model-model lainnya. Model ini menggunakan 100 *node* pada *hidden layer* 1 dan *learning rate* sebesar 0,1. Meskipun nilai *learning rate* tersebut tergolong besar dan berpotensi menyebabkan ketidakstabilan, kombinasi *hyperparameter* yang digunakan pada model R1 mampu menghasilkan pelatihan yang efisien dan stabil. Model ini berhasil

mencapai nilai *cost* terkecil, yaitu 0,001834, yang menandakan bahwa model ini memiliki tingkat kesalahan yang paling rendah selama proses pelatihan. Selain itu, model R1 berhasil mencapai konvergensi dalam 2401 *epoch* dengan durasi waktu pelatihan 11,81 detik. Meskipun terdapat model lain seperti model P1 dan model Q1 yang memiliki durasi waktu pelatihan lebih singkat, model R1 tetap dianggap paling optimal. Hal ini karena model R1 mampu mencapai keseimbangan antara efisiensi waktu pelatihan dan tingkat kesalahan yang sangat rendah. Dengan demikian, model R1 berhasil memanfaatkan *learning rate* besar secara efektif tanpa mengorbankan stabilitas pelatihan.

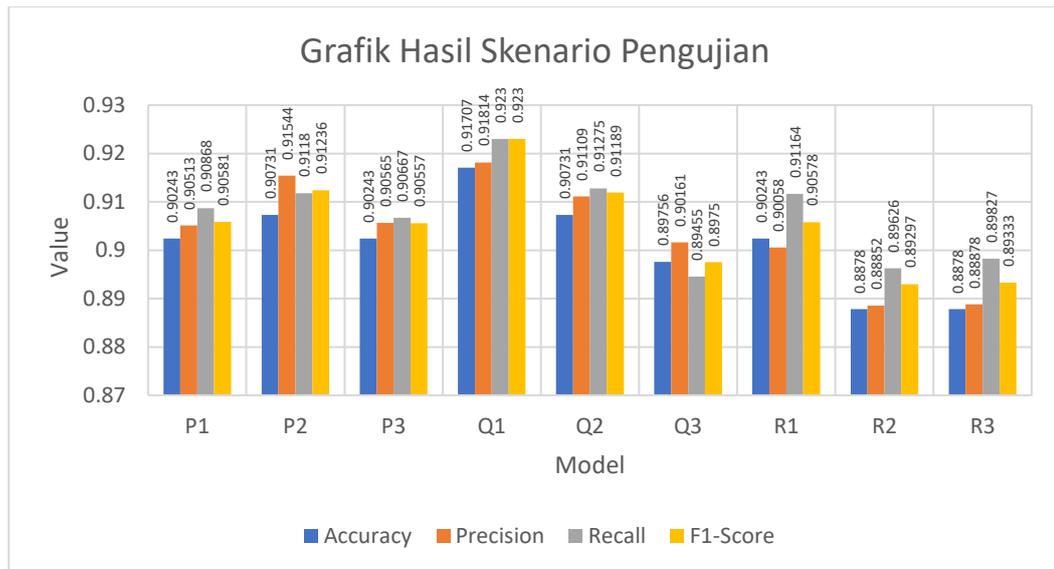
Setelah proses pelatihan selesai, model-model yang telah dilatih diuji menggunakan data *testing* yang belum pernah digunakan sebelumnya. Pengujian ini bertujuan untuk mengevaluasi seberapa baik model dalam mengenali pola dan memprediksi data baru. Evaluasi ini juga memastikan bahwa model tidak hanya bekerja dengan baik pada data *training*, tetapi juga mampu mengklasifikasikan data baru dengan akurasi yang optimal. Hasil evaluasi dari setiap model ditampilkan dalam Tabel 4.9.

Tabel 4.9 Hasil evaluasi model dari skenario pengujian

Model	Precision	Recall	F1-Score	Accuracy
P1	0,90513	0,90868	0,90581	0,90243
P2	0,91544	0,91180	0,91236	0,90731
P3	0,90565	0,90667	0,90557	0,90243
Q1	0,91814	0,92300	0,92300	0,91707
Q2	0,91109	0,91275	0,91189	0,90731
Q3	0,90161	0,89455	0,89750	0,89756
R1	0,90058	0,91164	0,90578	0,90243
R2	0,88852	0,89626	0,89297	0,88780
R3	0,88878	0,89827	0,89333	0,88780

Untuk memberikan gambaran yang lebih jelas terhadap hasil evaluasi performa model pada Tabel 4.9, dibuat grafik perbandingan performa model

berdasarkan nilai *precision*, *recall*, *f1-score*, dan *accuracy*. Visualisasi ini bertujuan untuk mempermudah analisis terhadap keunggulan dan kelemahan masing-masing model, serta membantu dalam mengidentifikasi model dengan performa paling konsisten di berbagai metrik evaluasi.

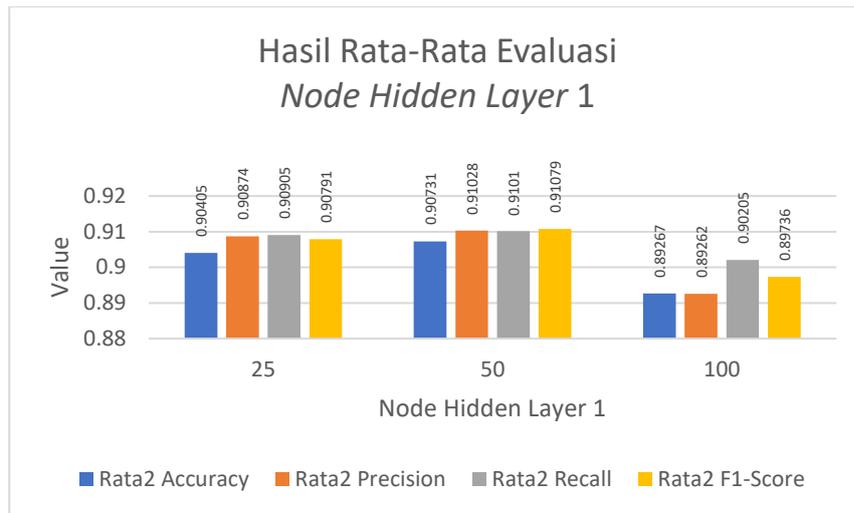


Gambar 4.30 Grafik hasil evaluasi skenario pengujian

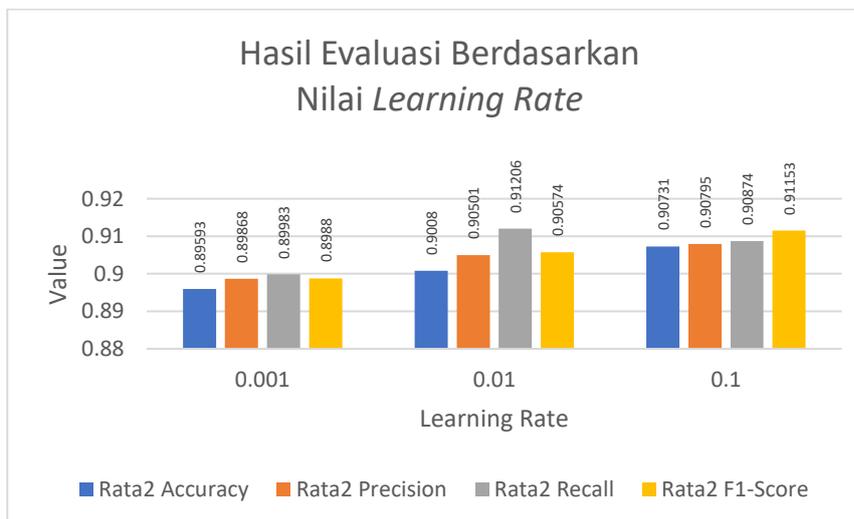
Berdasarkan hasil pengujian, keempat metrik evaluasi tersebut digunakan untuk menilai sejauh mana model mampu melakukan klasifikasi dengan baik. Secara keseluruhan, seluruh model menunjukkan hasil yang cukup baik dengan perbedaan nilai yang tidak terlalu jauh. Semakin mendekati nilai 1,0 pada masing-masing metrik, maka semakin baik pula kemampuan model dalam mengklasifikasikan ulasan secara akurat, yang menunjukkan bahwa performa model juga semakin meningkat.

Untuk memperoleh pemahaman yang lebih dalam terhadap hasil pengujian, dilakukan perbandingan antar model berdasarkan metrik evaluasi yang telah diperoleh. Tujuannya adalah untuk mengetahui model dengan performa terbaik,

serta menganalisis pengaruh jumlah *node* pada *hidden layer* 1 dan nilai *learning rate* terhadap hasil klasifikasi. Visualisasi grafik berdasarkan variasi jumlah *node* pada *hidden layer* 1 dan nilai *learning rate* dapat dilihat pada Gambar 4.31 dan Gambar 4.32.



Gambar 4.31 Grafik hasil evaluasi berdasarkan *node hidden layer* 1



Gambar 4.32 Grafik hasil evaluasi berdasarkan nilai learning rate

Berdasarkan hasil evaluasi yang ditampilkan pada Gambar 4.31, dapat dilihat bahwa jumlah *node* pada *hidden layer* 1 tidak memberikan pengaruh yang

terlalu besar terhadap performa model. Nilai rata-rata metrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *f1-score* cenderung stabil, terutama pada jumlah *node* 25 dan 50. Namun, ketika jumlah *node* ditambah menjadi 100, performa model justru sedikit menurun. Hal ini menunjukkan bahwa penggunaan jumlah *node* yang terlalu banyak bisa menyebabkan *overfitting*, yaitu kondisi ketika model terlalu fokus pada data latih dan kurang mampu memprediksi data baru secara akurat. Sebaliknya, jumlah *node* yang terlalu sedikit juga bisa menyebabkan *underfitting*, yaitu ketika model belum cukup kuat untuk mengenali pola dalam data.

Sementara itu, pada Gambar 4.32 pengaruh nilai *learning rate* terlihat lebih signifikan terhadap hasil klasifikasi. Pada grafik, terlihat bahwa model dengan *learning rate* sebesar 0,1 menghasilkan nilai evaluasi yang paling tinggi dibandingkan dengan nilai 0,01 dan 0,001. Hal ini menunjukkan bahwa *learning rate* yang lebih besar membantu proses pelatihan model menjadi lebih cepat dan efisien, tanpa mengorbankan akurasi. Sebaliknya, *learning rate* yang terlalu kecil membuat proses pelatihan berjalan lambat dan kurang efektif, sehingga hasil klasifikasinya juga tidak optimal.

Dari seluruh model yang telah diuji, model Q1 menunjukkan performa paling optimal secara keseluruhan dengan kombinasi jumlah *node hidden layer* 1 sebanyak 50 dan *learning rate* sebesar 0,1. Jumlah *node* yang tidak terlalu sedikit maupun terlalu banyak membuat model cukup kompleks untuk mengenali pola tanpa menyebabkan *overfitting*, sedangkan *learning rate* yang digunakan memungkinkan proses pembelajaran berlangsung cepat dan efisien. Hasil evaluasi

model Q1 juga menunjukkan nilai *precision*, *recall*, *f1-score*, dan *accuracy* yang tinggi dan seimbang, sehingga model ini mampu melakukan klasifikasi secara akurat dan konsisten terhadap ketiga kategori aspek ulasan.

Untuk melihat lebih dalam bagaimana kinerja model tersebut, dilakukan analisis menggunakan *confusion matrix* untuk membantu memahami jumlah prediksi yang benar dan salah dalam setiap kategori ulasan. Hasil *confusion matrix* untuk model Q1 ditampilkan pada Tabel 4.10, 4.11 dan 4.12.

Tabel 4.10 *Confusion matrix* model Q1 kategori kualitas pelayanan

Kelas Aktual	Kelas Prediksi	
	Kualitas Pelayanan (A)	Bukan Kualitas Pelayanan
Kualitas Pelayanan (A)	31	2
Bukan Kualitas Pelayanan	2	170

Tabel 4.11 *Confusion matrix* model Q1 kategori makanan

Kelas Aktual	Kelas Prediksi	
	Makanan (B)	Bukan Makanan
Makanan (B)	97	10
Bukan Makanan	7	91

Tabel 4.12 *Confusion matrix* model Q1 kategori suasana

Kelas Aktual	Kelas Prediksi	
	Suasana (C)	Bukan Suasana
Suasana (C)	60	5
Bukan Suasana	8	132

Berdasarkan metrik evaluasi, nilai *precision* pada model Q1 adalah 0,91814 atau 91,81%. Nilai ini diperoleh dari *confusion matrix*, di mana jumlah FP untuk kategori kualitas pelayanan sebanyak 2 data dengan 31 data TP, kategori makanan memiliki 7 data FP dengan 97 data TP, dan kategori suasana memiliki 8 data FP

dengan 60 data TP. Hasil ini menunjukkan bahwa model memiliki tingkat kesalahan FP yang sangat rendah, yang berarti model cukup akurat dalam menghindari kesalahan prediksi terhadap ulasan yang tidak termasuk dalam kategori tertentu.

Pada metrik *recall*, model Q1 mendapatkan nilai 0,92300 atau 92,30%. Berdasarkan *confusion matrix*, jumlah FN untuk kategori kualitas pelayanan adalah 2 data dengan 31 data TP, kategori makanan memiliki 10 data FN dengan 97 data TP, dan kategori suasana memiliki 5 data FN dengan 60 data TP. Dari hasil tersebut, terlihat bahwa model masih mengalami kesalahan dalam mengklasifikasikan beberapa ulasan yang seharusnya termasuk dalam kategori tertentu tetapi malah dianggap bukan bagian dari kategori tersebut.

Nilai *f1-score* model Q1 adalah 0,92300 atau 92,30%. Nilai ini merupakan kombinasi antara *precision* dan *recall*, yang menunjukkan keseimbangan model dalam menghindari kesalahan prediksi (FP) serta tidak melewatkan data yang seharusnya dikategorikan (FN). Dengan nilai *f1-score* yang tinggi, dapat disimpulkan bahwa model memiliki performa yang baik dalam mengklasifikasikan ulasan ke dalam kategori yang sesuai.

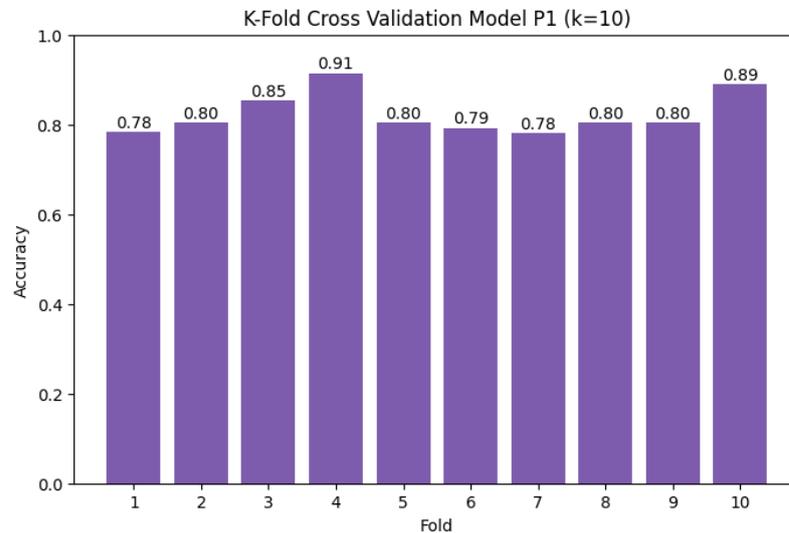
Sementara itu, nilai *accuracy* model Q1 adalah 0,91707 atau 91,71%, yang berarti model mampu mengklasifikasikan 91,71% dari seluruh data dengan benar. Meskipun nilainya cukup tinggi, masih terdapat beberapa kesalahan yang menyebabkan beberapa ulasan tidak diklasifikasikan dengan tepat.

Untuk memperoleh hasil yang lebih akurat, penelitian ini menerapkan *k-fold cross validation* untuk mengevaluasi kinerja model dengan menggunakan distribusi data yang berbeda, yang bertujuan untuk meningkatkan keandalan hasil evaluasi.

Dalam pengujian ini, digunakan nilai $k=10$, di mana data dibagi menjadi 10 *fold* dengan ukuran yang sama. Setiap *fold* akan secara bergantian digunakan sebagai data uji, sementara *fold* lainnya digunakan untuk melatih model. Setelah menguji semua *fold*, metrik performa dihitung untuk setiap iterasi, dan rata-rata dari nilai metrik tersebut dihitung untuk memberikan gambaran yang lebih stabil tentang kinerja model secara keseluruhan. Pemilihan nilai $k = 10$ didasarkan pada penelitian Oyedele (2023), yang menyatakan bahwa nilai k antara 10 hingga 20 sering dianggap optimal karena memberikan keseimbangan antara akurasi estimasi dan efisiensi komputasi. Proses *k-fold cross validation* ini diterapkan pada seluruh skenario pengujian, mencakup model P1, P2, P3, Q1, Q2, Q3, R1, R2, dan R3. Hal ini dilakukan untuk memastikan performa model pada setiap kombinasi data dan untuk menentukan skenario terbaik yang memberikan hasil evaluasi paling optimal.

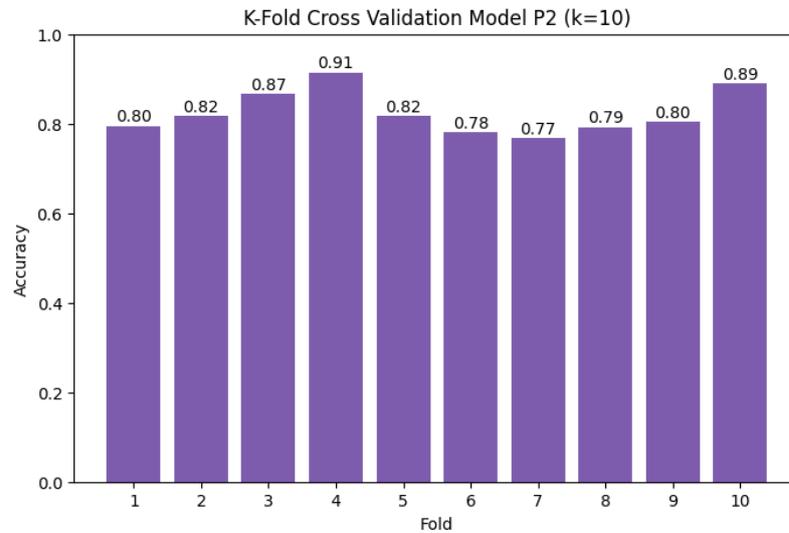
Pada model P1, digunakan jumlah *node* sebanyak 25 pada *hidden layer 1* dan nilai *learning rate* sebesar 0.1. Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,82343 atau 82,34%. Nilai akurasi tertinggi diperoleh pada *fold* ke-4 sebesar 0,91 atau 91%, sementara nilai akurasi terendah terjadi pada *fold* ke-1 dan 7 sebesar 0,78 atau 78%. Dari keseluruhan hasil tersebut, diperoleh nilai standar deviasi sebesar 0,04414. Nilai standar deviasi ini digunakan untuk mengukur seberapa besar variasi akurasi antar *fold*. Semakin kecil nilai standar deviasi, maka semakin konsisten performa model terhadap data uji yang berbeda. Oleh karena itu, dapat disimpulkan bahwa model P1 memiliki tingkat kestabilan yang cukup baik

karena perbedaan akurasi antar *fold* tidak terlalu besar. Perbandingan nilai akurasi pada setiap *fold* untuk model P1 dapat dilihat pada Gambar 4.33.



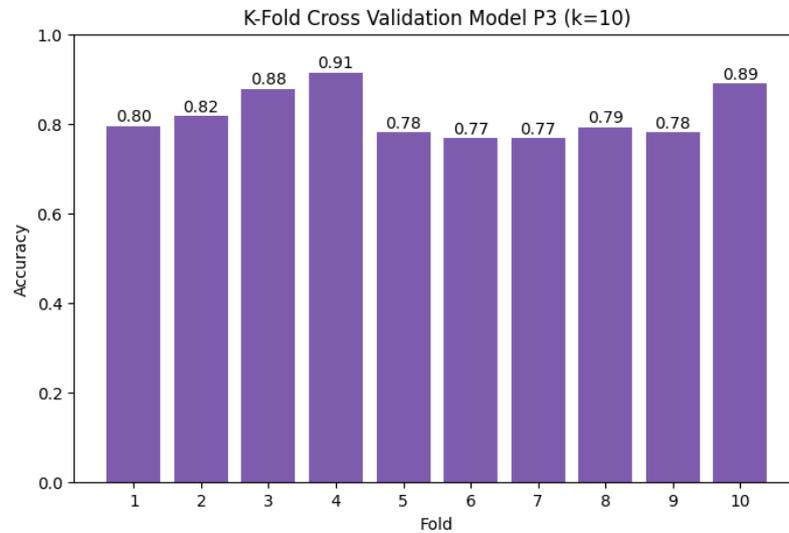
Gambar 4.33 Hasil akurasi setiap *fold* pada model P1

Pada model P2, digunakan jumlah *node* sebanyak 25 pada *hidden layer 1* dan nilai *learning rate* sebesar 0.01. Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,82464 atau 82,46%. Nilai akurasi tertinggi diperoleh pada *fold* ke-4 sebesar 0,91 atau 91%, sementara nilai akurasi terendah terjadi pada *fold* ke-6 sebesar 0,78 atau 78%. Dari keseluruhan hasil tersebut, diperoleh nilai standar deviasi sebesar 0,04649. Nilai standar deviasi tersebut menunjukkan bahwa perbedaan akurasi antar *fold* tidak terlalu besar, sehingga model P2 dapat dikatakan cukup stabil. Meskipun nilai standar deviasinya sedikit lebih tinggi dibanding model P1, konsistensi performa model tetap tergolong baik. Perbandingan nilai akurasi pada setiap *fold* untuk model P2 dapat dilihat pada Gambar 4.34.



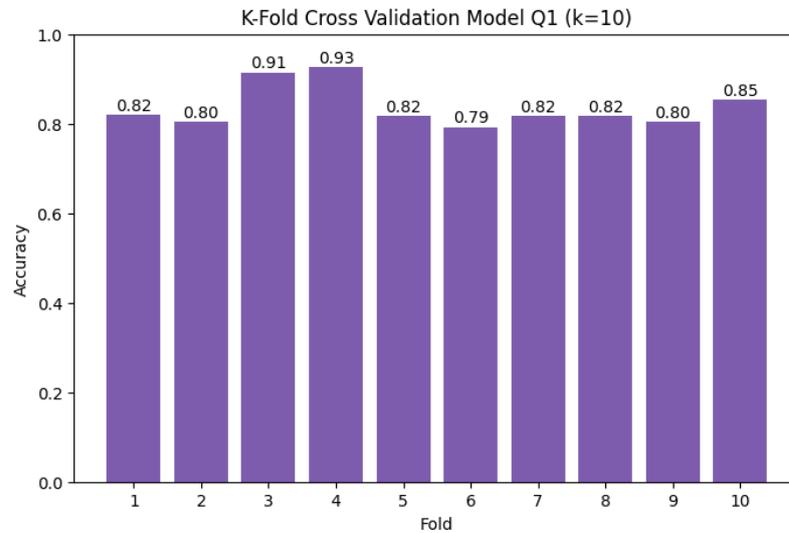
Gambar 4.34 Hasil akurasi setiap *fold* pada model P2

Pada model P3, digunakan jumlah *node* sebanyak 25 pada *hidden layer 1* dan nilai *learning rate* sebesar 0.001. Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,81854 atau 81,85%. Nilai akurasi tertinggi diperoleh pada *fold* ke-4 sebesar 0,91 atau 91%, sementara nilai akurasi terendah terjadi pada *fold* ke-6 dan 7 sebesar 0,77 atau 77%. Dari keseluruhan hasil tersebut, diperoleh nilai standar deviasi sebesar 0,05203. Meskipun nilai standar deviasi ini sedikit lebih besar dibandingkan model P1 dan P2, model P3 tetap memiliki performa yang stabil meskipun diuji dengan data yang berbeda-beda. Dengan demikian, model P3 dapat dianggap cukup stabil. Perbandingan hasil akurasi pada setiap *fold* untuk model P3 ditampilkan pada Gambar 4.35.



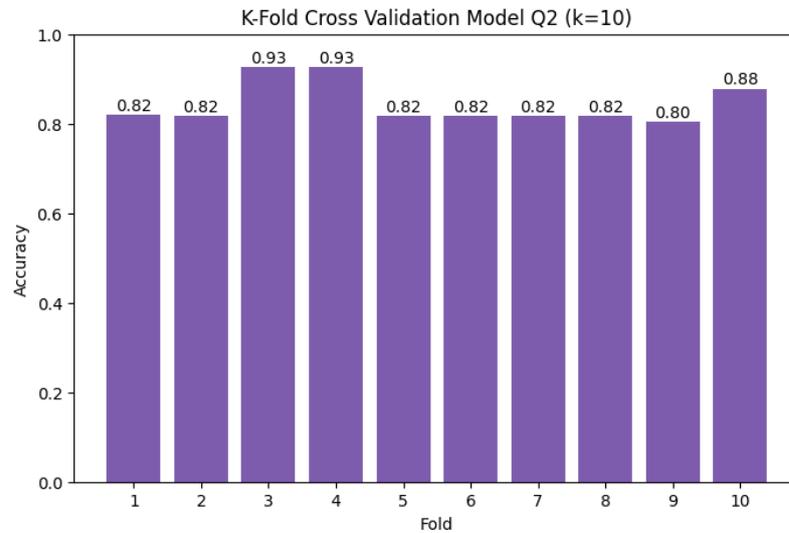
Gambar 4.35 Hasil akurasi setiap *fold* pada model P3

Selanjutnya, pada model Q1 digunakan jumlah *node* sebanyak 50 pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.1. Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,83680 atau 83,68%. Nilai akurasi tertinggi diperoleh pada *fold* ke-4 sebesar 0,93 atau 93%, sementara nilai akurasi terendah terjadi pada *fold* ke-6 sebesar 0,79 atau 79%. Dari keseluruhan hasil tersebut, diperoleh nilai standar deviasi sebesar 0,04461. Model Q1 ini memiliki tingkat kestabilan yang cukup baik karena perbedaan akurasi antar *fold* tidak terlalu besar. Perbandingan hasil akurasi pada setiap *fold* untuk model Q1 ditampilkan pada Gambar 4.36.



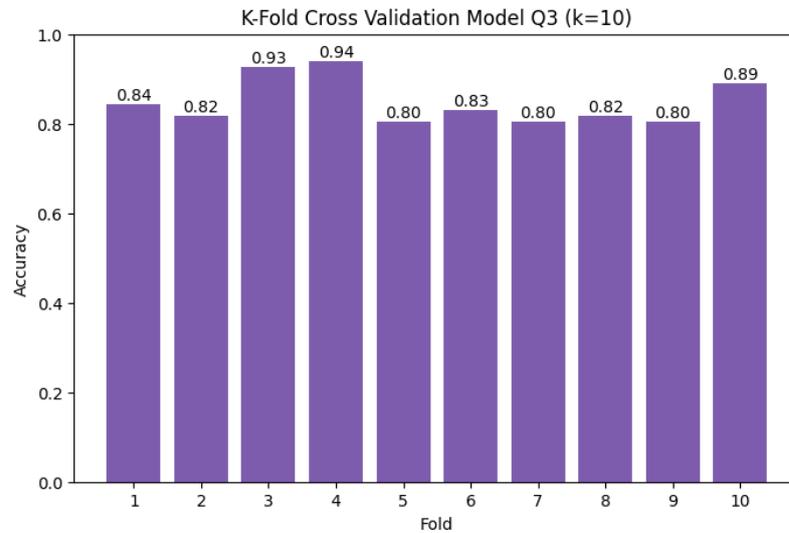
Gambar 4.36 Hasil akurasi setiap *fold* pada model Q1

Pada model Q2, digunakan jumlah *node* sebanyak 50 pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.01 Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,84412 atau 84,41%. Nilai akurasi tertinggi diperoleh pada *fold* ke-3 dan 4 sebesar 0,93 atau 93%, sementara nilai akurasi terendah terjadi pada *fold* ke-9 sebesar 0,80 atau 80%. Dari keseluruhan hasil tersebut, diperoleh nilai standar deviasi sebesar 0,04543. Meskipun nilai standar deviasi model Q2 sedikit lebih tinggi dibandingkan model Q1, tetapi variasi performa model Q2 antar *fold* tetap tergolong stabil. Perbandingan hasil akurasi pada setiap *fold* untuk model Q2 ditampilkan pada Gambar 4.37.



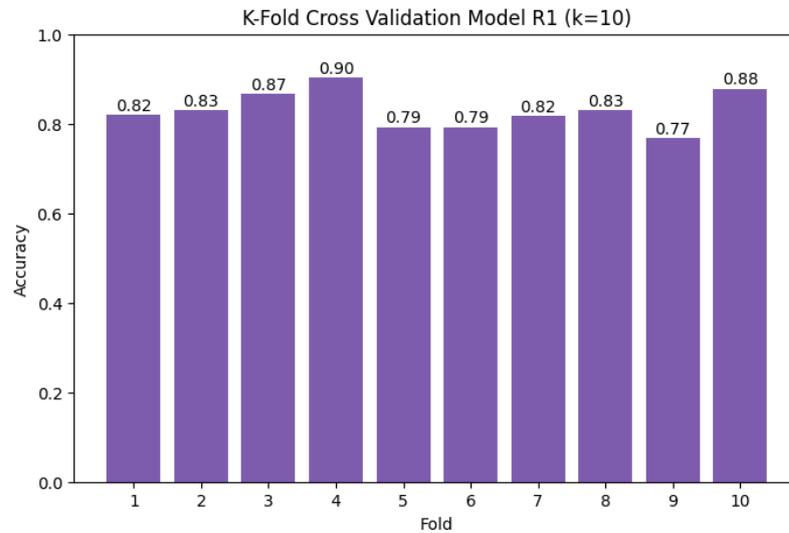
Gambar 4.37 Hasil akurasi setiap *fold* pada model Q2

Pada model Q3, digunakan jumlah *node* sebanyak 50 pada *hidden layer 1* dan nilai *learning rate* sebesar 0.001. Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,84775 atau 84,77%. Nilai akurasi tertinggi diperoleh pada *fold* ke-4 sebesar 0,94 atau 94%, sementara nilai akurasi terendah terjadi pada *fold* ke-5, 7 dan 9 sebesar 0,80 atau 80%. Dari keseluruhan hasil tersebut, diperoleh nilai standar deviasi sebesar 0,04913. Nilai ini lebih tinggi dibandingkan model Q2, tetapi konsistensi performa model tetap tergolong baik. Dengan demikian, model Q3 masih dapat dikategorikan cukup stabil terhadap variasi data uji. Perbandingan hasil akurasi pada setiap *fold* untuk model Q3 ditampilkan pada Gambar 4.38.



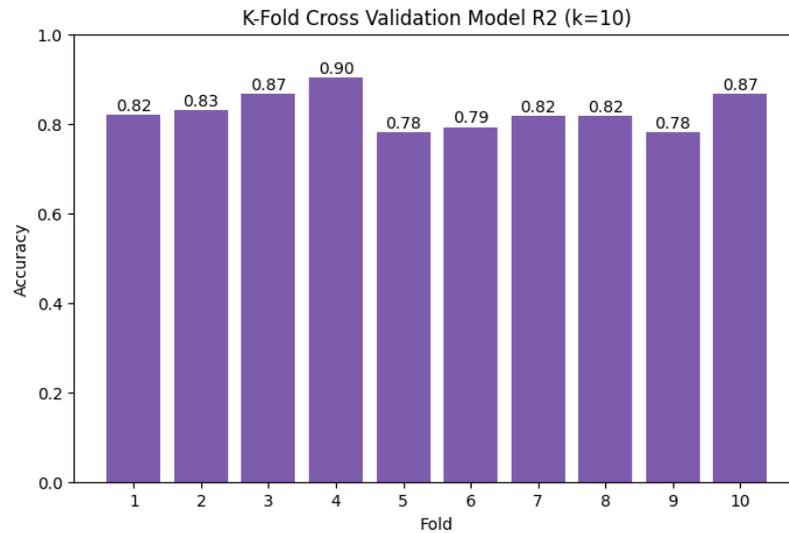
Gambar 4.38 Hasil akurasi setiap *fold* pada model Q3

Sementara itu, pada model R1 menggunakan jumlah *node* sebanyak 100 pada *hidden layer* 1 dan nilai *learning rate* sebesar 0.1. Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,82948 atau 82,95%. Nilai akurasi tertinggi diperoleh pada *fold* ke-4 sebesar 0,90 atau 90%, sementara nilai akurasi terendah terjadi pada *fold* ke-9 sebesar 0,77 atau 77%. Nilai standar deviasi yang diperoleh sebesar 0,03964. Berdasarkan nilai tersebut, model R1 dapat dianggap cukup stabil terhadap data uji yang berbeda-beda. Perbandingan hasil akurasi pada setiap *fold* untuk model R1 ditampilkan pada Gambar 4.39.



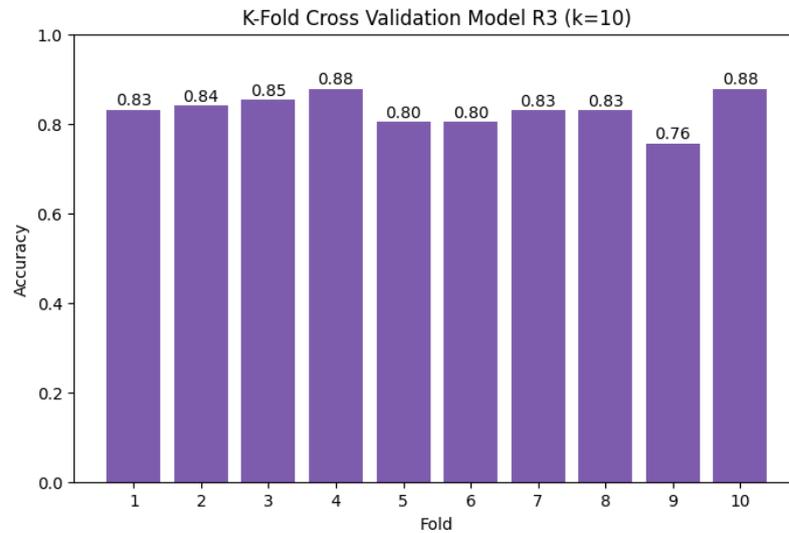
Gambar 4.39 Hasil akurasi setiap *fold* pada model R1

Pada model R2, digunakan jumlah *node* sebanyak 100 pada *hidden layer 1* dan nilai *learning rate* sebesar 0.01. Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,82704 atau 82,70%. Nilai akurasi tertinggi diperoleh pada *fold* ke-10 sebesar 0,87 atau 87%, sementara nilai akurasi terendah terjadi pada *fold* ke-7 sebesar 0,80 atau 80%. Nilai standar deviasi yang diperoleh sebesar 0,03804. Nilai ini lebih rendah dibandingkan model R1, yang menunjukkan bahwa variasi performa antar *fold* pada model R2 masih berada dalam batas stabil. Perbandingan hasil akurasi pada setiap *fold* untuk model R2 ditampilkan pada Gambar 4.40.



Gambar 4.40 Hasil akurasi setiap *fold* pada model R2

Dan yang terakhir model R3 menggunakan jumlah *node* sebanyak 100 pada *hidden layer* 1 dan nilai *learning rate* 0.001. Berdasarkan proses *k-fold cross validation* dengan nilai $k=10$ hasil pengujian menunjukkan bahwa rata-rata akurasi model sebesar 0,83069 atau 83,07%. Nilai akurasi tertinggi diperoleh pada *fold* ke-4 dan 10 sebesar 0,88 atau 88%, sementara nilai akurasi terendah terjadi pada *fold* ke-9 sebesar 0,76 atau 76%. Nilai standar deviasi yang diperoleh sebesar 0,03468. Nilai standar deviasi ini lebih rendah dibandingkan model R1 dan R2 yang menunjukkan adanya variasi akurasi antar *fold*, jadi secara umum model dapat dianggap cukup stabil. Perbandingan hasil akurasi pada setiap *fold* untuk model R3 ditampilkan pada Gambar 4.41.



Gambar 4.41 Hasil akurasi setiap *fold* pada model R3

Berdasarkan hasil *k-fold cross validation*, model Q3 dipilih sebagai model yang paling optimal dalam penelitian ini. Model ini memiliki rata-rata akurasi tertinggi, yaitu sebesar 0,847751 atau 84,77%, dibandingkan dengan model lainnya. Meskipun bukan model dengan standar deviasi terendah, nilai deviasi sebesar 0,049139 masih berada dalam rentang yang cukup stabil, yang menunjukkan bahwa performa model tidak terlalu bervariasi antar lipatan data. Selain itu, model Q3 juga memiliki nilai akurasi minimum yang tinggi, yaitu 81,70%, sehingga menandakan bahwa performanya tetap konsisten dalam setiap iterasi pengujian.

Kombinasi antara tingginya rata-rata akurasi dan kestabilan performa ini menjadi alasan utama dipilihnya model Q3, karena model diharapkan mampu memberikan hasil klasifikasi yang akurat sekaligus andal ketika diterapkan pada data baru yang bervariasi. Dalam konteks klasifikasi ulasan pelanggan restoran yang sangat beragam, kestabilan antar lipatan menjadi faktor penting agar model tetap dapat diandalkan di berbagai kondisi. Pemilihan model Q3 juga selaras

dengan tujuan penelitian, yaitu untuk mengevaluasi pengaruh pengaturan *hyperparameter*, seperti jumlah *node* pada *hidden layer* 1 dan nilai *learning rate*, terhadap performa klasifikasi. Hasil dari seluruh proses *k-fold cross validation* ditampilkan pada Tabel 4.13.

Tabel 4.13 Hasil *k-fold cross validation* seluruh skenario pengujian

Skenario Pengujian	Akurasi <i>k-fold</i> tertinggi	Akurasi <i>k-fold</i> terendah	Rata-rata Akurasi	Standar Deviasi
P1	0,914634	0,780487	0,823435	0,044143
P2	0,914634	0,768292	0,824640	0,046497
P3	0,914634	0,768292	0,818542	0,052037
Q1	0,926829	0,792682	0,836805	0,044616
Q2	0,926829	0,804878	0,844122	0,045439
Q3	0,939024	0,817073	0,847751	0,049139
R1	0,902439	0,768292	0,829488	0,039642
R2	0,902439	0,780487	0,827049	0,038048
R3	0,878048	0,756097	0,830693	0,034684

Selain mengevaluasi performa model klasifikasi, penelitian ini juga melakukan analisis terhadap kata-kata yang dianggap paling mewakili masing-masing kategori ulasan. Analisis ini dilakukan menggunakan *Word2Vec* Skip-Gram, yang mengubah kata-kata dalam teks menjadi vektor *embedding* untuk memahami hubungan antar kata. Melalui pendekatan ini, kata-kata yang paling relevan dalam setiap kategori ulasan, yaitu kualitas pelayanan, makanan, dan suasana dapat diidentifikasi.

Pada kelas kualitas pelayanan, 10 kata dominan yang berhasil diidentifikasi melalui proses vektor *embedding* menggunakan *Word2vec* Skip-Gram adalah “layan”, “sangat”, “ramah”, “baik”, “kurang”, “cepat”, “oke”, “muas”, “masih”, dan “langgan”. Kata-kata ini diperoleh dari hasil pelatihan model dengan vektor berdimensi 100 dan *window size* yang telah ditentukan sebelumnya. Model *Word2Vec* Skip-Gram mengidentifikasi kata-kata yang sering muncul dalam

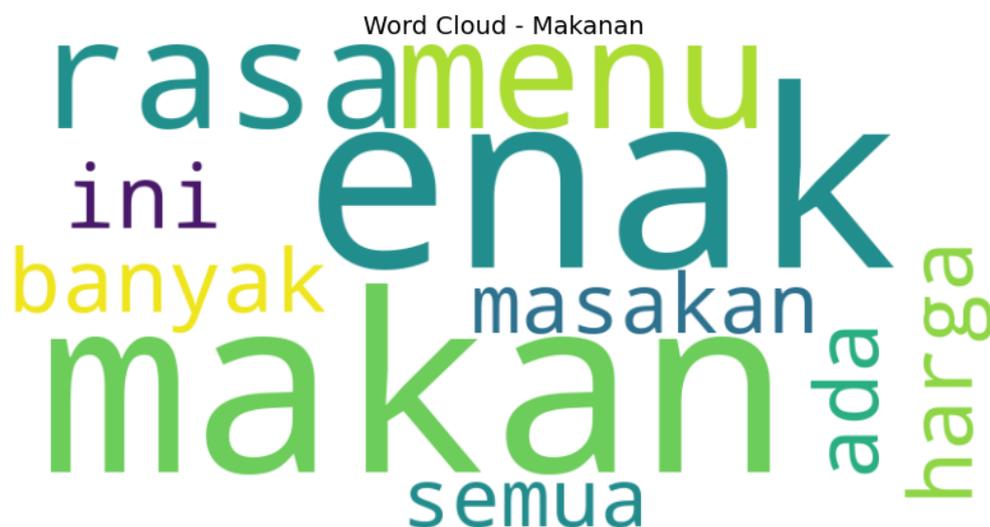
konteks yang serupa, sehingga menghasilkan representasi vektor yang saling berdekatan secara makna. Kemunculan kata-kata tersebut mencerminkan hubungan antar kata yang kuat serta menunjukkan bahwa istilah-istilah ini banyak digunakan pelanggan untuk menggambarkan interaksi antara pegawai dan pelanggan dalam aspek kualitas pelayanan. Hasil analisis kata-kata yang dominan dalam kelas kualitas pelayanan ditampilkan pada Gambar 4.42.



Gambar 4.42 *Word cloud* kata dominan pada kelas kualitas pelayanan

Pada kelas makanan, 10 kata dominan yang berhasil diidentifikasi melalui proses vektor *embedding* menggunakan *Word2vec* Skip-Gram adalah “makan”, “enak”, “rasa”, “menu”, “banyak”, “harga”, “ada”, “ini”, “masakan”, dan “semua”. Kata-kata ini diperoleh dari hasil pelatihan model dengan vektor berdimensi 100 dan *window size* yang telah ditentukan sebelumnya. Model *Word2Vec* Skip-Gram mengidentifikasi kata-kata yang sering muncul dalam konteks yang serupa, sehingga menghasilkan representasi vektor yang saling berdekatan secara makna. Kemunculan kata-kata tersebut menunjukkan adanya hubungan konteks yang kuat

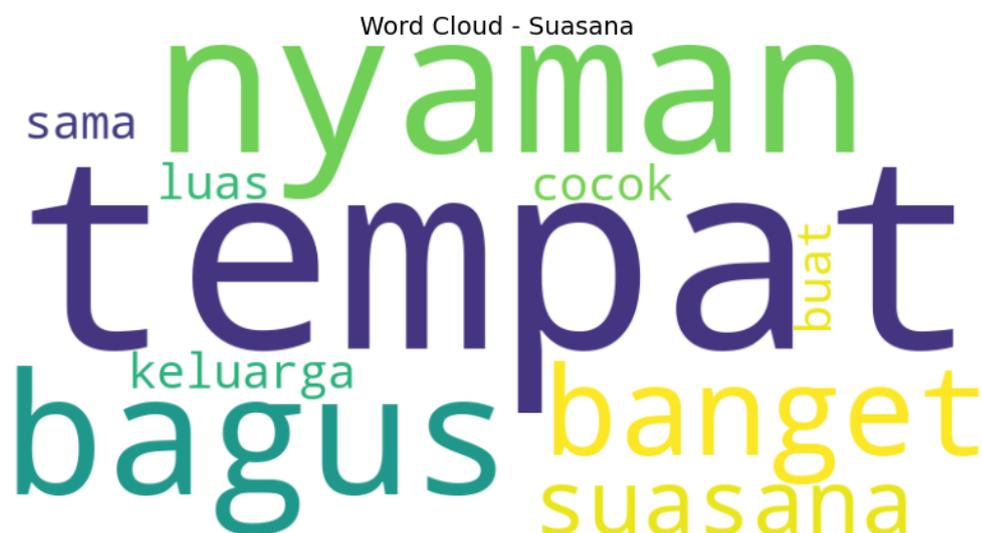
dan menggambarkan deskripsi pelanggan terhadap cita rasa, keberagaman menu, serta penilaian terhadap harga makanan dalam menggambarkan aspek makanan. Hasil analisis kata-kata yang dominan dalam kelas makanan ditampilkan pada Gambar 4.43.



Gambar 4.43 *Word cloud* kata dominan pada kelas makanan

Sementara itu, pada kelas suasana, 10 kata dominan yang berhasil diidentifikasi melalui proses vektor *embedding* menggunakan *Word2vec* Skip-Gram adalah “tempat”, “nyaman”, “bagus”, “banget”, “suasana”, “keluarga”, “cocok”, “buat”, “sama”, dan “luas”. Kata-kata ini diperoleh dari hasil pelatihan model dengan vektor berdimensi 100 dan *window size* yang telah ditentukan sebelumnya. Model *Word2Vec* Skip-Gram mengidentifikasi kata-kata yang sering muncul dalam konteks yang serupa, sehingga menghasilkan representasi vektor yang saling berdekatan secara makna. Kemunculan kata-kata tersebut mencerminkan hubungan konteks yang erat dan menunjukkan bagaimana pelanggan menilai lingkungan fisik serta kenyamanan tempat makan dalam

menggambarkan aspek suasana. Hasil analisis kata-kata yang dominan dalam kelas suasana ditampilkan pada Gambar 4.44.



Gambar 4.44 *Word cloud* kata dominan pada kelas suasana

Setiap aspek ulasan memiliki ciri khas yang tercermin dari kata-kata dominan tersebut. Hasil analisis kata menggunakan *Word2Vec* Skip-Gram tidak hanya membantu memahami konteks ulasan pelanggan, tetapi juga memberikan gambaran yang lebih jelas mengenai aspek-aspek yang paling sering diperhatikan dalam pengalaman makan di restoran. Melalui pendekatan ini, pelaku bisnis di sektor kuliner dapat mengidentifikasi bagian-bagian yang paling banyak mendapatkan sorotan atau evaluasi dari pelanggan, sehingga dapat menjadi dasar dalam melakukan perbaikan layanan secara menyeluruh.

Klasifikasi aspek ulasan pelanggan restoran memiliki peran penting dalam mempermudah pemahaman terhadap kebutuhan pelanggan secara lebih terarah dan terstruktur. Melalui pengelompokan ulasan berdasarkan kategori seperti kualitas pelayanan, makanan, dan suasana, pelaku usaha dapat melakukan evaluasi secara

lebih objektif dan sistematis. Dengan adanya klasifikasi ini, perhatian dan upaya perbaikan dapat difokuskan pada aspek-aspek yang paling sering dikomentari pelanggan, sehingga peningkatan layanan dapat dilakukan secara tepat sasaran.

Dalam perspektif Islam, proses evaluasi dan klasifikasi ini sejalan dengan nilai-nilai muhasabah atau introspeksi diri, yaitu anjuran untuk senantiasa meninjau dan memperbaiki apa yang telah dilakukan. Hal ini sebagaimana firman Allah SWT dalam QS. Al-Hasyr ayat 18 :

يَا أَيُّهَا الَّذِينَ آمَنُوا اتَّقُوا اللَّهَ وَلْتَنْظُرْ نَفْسٌ مَّا قَدَّمَتْ لِغَدٍّ وَاتَّقُوا اللَّهَ إِنَّ اللَّهَ خَبِيرٌ بِمَا تَعْمَلُونَ ﴿١٨﴾

“Wahai orang-orang yang beriman, bertakwalah kepada Allah dan hendaklah setiap orang memperhatikan apa yang telah diperbuatnya untuk hari esok (akhirat). Bertakwalah kepada Allah. Sesungguhnya Allah Maha teliti terhadap apa yang kamu kerjakan.”(Q.S Al-Hasyr:18)

Menurut Tafsir Al-Muyassar, ayat ini mengajak orang-orang beriman untuk senantiasa melakukan muhasabah terhadap amal perbuatannya agar tidak terjerumus dalam kelalaian. Evaluasi terhadap setiap tindakan menjadi bagian penting dalam kehidupan, termasuk dalam konteks dunia usaha. Dalam hal ini, sistem klasifikasi otomatis berfungsi sebagai alat bantu muhasabah yang memungkinkan pelaku usaha untuk meninjau kritik dan saran pelanggan yang sebelumnya tersebar dalam bentuk ulasan bebas dan tidak terstruktur.

Selain itu, Allah SWT juga menegaskan pentingnya memperhatikan setiap amal, sekecil apa pun, sebagaimana dalam QS. Az-Zalzalah ayat 7–8:

فَمَنْ يَعْمَلْ مِثْقَالَ ذَرَّةٍ خَيْرًا يَرَهُ ﴿٧﴾ وَمَنْ يَعْمَلْ مِثْقَالَ ذَرَّةٍ شَرًّا يَرَهُ ﴿٨﴾

“Barangsiapa yang mengerjakan kebaikan seberat dzarrahpun, niscaya dia akan melihat (balasan)nya. Dan barangsiapa yang mengerjakan kejahatan sebesar dzarrah pun, niscaya dia akan melihat (balasan)nya pula.”(Q.S Az-Zalzalah:7-8)

Berdasarkan tafsir tahlili, ayat ini menunjukkan bahwa setiap tindakan manusia, baik atau buruk, sekecil apa pun, akan mendapatkan balasannya. Hal ini menjadi pengingat bagi pelaku usaha untuk tidak menganggap remeh setiap masukan pelanggan. Kritik dan saran yang terekam dalam ulasan pelanggan harus dipandang sebagai bentuk tanggung jawab pelayanan yang harus ditindaklanjuti. Dalam hal ini, teknologi klasifikasi otomatis sangat membantu untuk memilah dan mengelompokkan ulasan pelanggan. Dengan begitu, pelaku usaha bisa lebih mudah melihat hal-hal yang perlu diperbaiki agar layanan yang diberikan bisa semakin baik.

Dalam konteks penelitian ini, proses klasifikasi aspek-aspek dalam ulasan pelanggan restoran di Indonesia menggunakan *Neural Network*, menunjukkan bahwa hasil metode ini efektif dalam mengelompokkan ulasan secara otomatis ke dalam 3 kategori utama, yaitu kualitas pelayanan, makanan, dan suasana. Efektivitas tersebut terlihat dari hasil pengujian, di mana model Q1 dengan jumlah *node hidden layer* 1 sebanyak 50 dan nilai *learning rate* 0.1 berhasil mencapai akurasi sebesar 91,71% dengan nilai *precision*, *recall*, dan *f1-score* yang tinggi dan seimbang. Selain itu, hasil evaluasi menggunakan *10-fold cross validation* menunjukkan bahwa model Q3, dengan konfigurasi 50 *node hidden layer* 1 dan nilai *learning rate* 0.01, mampu menghasilkan rata-rata akurasi tertinggi sebesar 84,77% secara konsisten di setiap lipatan data. Penelitian ini juga membuktikan bahwa pengaturan *hyperparameter*, seperti jumlah *node* pada *hidden layer* 1 dan nilai *learning rate* berperan penting dalam meningkatkan akurasi dan stabilitas model. Dengan memilih konfigurasi *hyperparameter* yang tepat, sistem klasifikasi

yang dibangun menjadi lebih andal dan stabil, sehingga dapat digunakan untuk membantu pelaku usaha memahami pengalaman pelanggan secara lebih terstruktur dan menyeluruh. Dengan begitu, pelaku usaha bisa lebih mudah melihat hal-hal yang perlu diperbaiki agar layanan yang diberikan semakin optimal.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan serangkaian pengujian terhadap 9 skenario model klasifikasi aspek ulasan pelanggan restoran di Indonesia menggunakan *Neural Network*, masing-masing model diuji dengan variasi jumlah *node* pada *hidden layer* 1 dan nilai *learning rate*. Evaluasi awal dilakukan menggunakan metrik evaluasi seperti *confusion matrix*, *accuracy*, *precision*, *recall*, dan *F1-score* pada data *testing* dengan rasio pembagian 80:20. Dari hasil pengujian tersebut, beberapa model menunjukkan performa yang cukup tinggi dan stabil, seperti model Q1 dengan jumlah *node hidden layer* 1 sebanyak 50 dan nilai *learning rate* 0.01 yang memperoleh akurasi sebesar 91,71%. Pengujian ini menjadi dasar untuk memahami pengaruh kombinasi *hyperparameter* terhadap hasil klasifikasi ulasan pelanggan restoran.

Untuk memastikan bahwa model mampu bekerja secara konsisten pada data yang bervariasi, seluruh model kemudian dievaluasi ulang menggunakan teknik *10-fold cross-validation*. Hasil evaluasi ini menunjukkan bahwa model Q3, dengan konfigurasi 50 *node* pada *hidden layer* 1 dan nilai *learning rate* 0.001, menghasilkan performa paling optimal dibandingkan model lainnya berdasarkan rata-rata akurasi sebesar 84,77% secara konsisten di setiap lipatan. Hal ini menunjukkan bahwa model tersebut memiliki tingkat generalisasi yang baik dan stabilitas performa yang kuat terhadap variasi data uji.

Secara keseluruhan, penelitian ini membuktikan bahwa metode Neural Network efektif dalam mengelompokkan ulasan pelanggan restoran ke dalam tiga aspek utama, yaitu kualitas pelayanan, makanan, dan suasana. Selain itu, kombinasi pengaturan *hyperparameter* yang tepat terbukti dapat meningkatkan kinerja model secara signifikan. Meskipun begitu, akurasi yang dihasilkan masih belum sepenuhnya optimal. Beberapa faktor yang diduga memengaruhi hasil ini antara lain adalah distribusi data yang tidak seimbang, kompleksitas pola dalam ulasan, dan keterbatasan representasi fitur dari *Word2Vec* Skip-Gram. Oleh karena itu, pengembangan lebih lanjut sangat dianjurkan untuk meningkatkan performa klasifikasi di masa mendatang.

5.2 Saran

Penelitian ini telah menghasilkan model klasifikasi aspek ulasan pelanggan restoran di Indonesia menggunakan *Neural Network* dengan performa yang cukup baik. Meskipun demikian, masih terdapat beberapa aspek yang dapat dikembangkan lebih lanjut untuk meningkatkan kualitas dan keakuratan hasil penelitian di masa mendatang. Beberapa saran yang dapat dipertimbangkan untuk penelitian selanjutnya antara lain:

1. Mengumpulkan data ulasan yang lebih banyak dari berbagai sumber, seperti *TripAdvisor* atau platform lainnya, serta memastikan distribusi data tiap kelas lebih seimbang agar hasil klasifikasi menjadi lebih akurat dan merata di setiap kategori.

2. Menggunakan pendekatan ekstraksi fitur lain yang dapat menghasilkan vektor kata dengan kualitas yang lebih baik dan relevan dibandingkan *Word2Vec* Skip-Gram.
3. Mencoba pengaturan arsitektur model *Neural Network* yang berbeda, seperti menambah jumlah *hidden layer* atau memodifikasi struktur jaringan, guna meningkatkan kemampuan model dalam memahami pola dari data ulasan.

Dengan menerapkan saran-saran tersebut, penelitian berikutnya diharapkan dapat menghasilkan model yang lebih akurat dan stabil untuk klasifikasi aspek ulasan pelanggan restoran di Indonesia.

DAFTAR PUSTAKA

- Adhiatma, F. D., & Qoiriah, A. (2022). Penerapan Metode TF-IDF dan Deep Neural Network untuk Analisa Sentimen pada Data Ulasan Hotel. *Journal of Informatics and Computer Science*, 2686–2220, 183–193.
- Aryanti, W. (2023). Penerapan Artificial Neural Network dengan Algoritma Backpropagation untuk Memprediksi Harga Saham. *Jurnal Riset Statistika*, 107–118. <https://doi.org/10.29313/jrs.v3i2.2953>
- Astuti, A. P., Alam, S., & Jaelani, I. (2022). Komparasi Algoritma Support Vector Machine Dengan Naive Bayes Untuk Analisis Sentimen Pada Aplikasi BRImo. *Bangkit Indonesia*, XI(02).
- Atsnawiyah, D., Rizan, M., & Rahmi. (2022). *Cafe Atmosphere And Food Quality As The Antecedent of Customer Satisfaction In Bulding Customer Loyalty At Masalalu Cafe Rawa Domba Jakarta*.
- Chandra, D. P., & Suryadibrata, A. (2019). Implementasi Jaringan Saraf Tiruan Backpropagation untuk Pengenalan Karakter pada Dokumen Tercetak. *ULTIMA Computing*, XI(2), 81.
- Dewananda, K. F., Rahmawati, W. M., Wardhana, S. R., & Yuliasuti, G. E. (2022). Penentuan Relevansi Artikel Ilmiah dengan Metode Word2Vec Article History ABSTRAK. *Jurnal Riset Inovasi Bidang Informatika Dan Pendidikan Informatika (KERNEL)*, 3(2).
- Dinda, M., Azzahra, M., Hafid, T., & Alam, S. (2023). Analisis Sentimen Ulasan Produk Serum Wajah Pada Beauty Brand Somethinc Menggunakan Niave Bayes Classifier. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 7, Issue 3). <https://reviews.femaledaily.com/products/treatment/s>
- Drzik, D., & Kapusta, J. (2024). *Effect of dimension size and window size on word embedding in classification tasks*. <https://doi.org/10.21203/rs.3.rs-4532901/v1>
- Franco, A. M. R., Cunha, I., & Oliveira, L. B. (2024). Evaluation of deep neural network architectures for authorship obfuscation of Portuguese texts. *Natural Language Processing Journal*, 9, 100107. <https://doi.org/10.1016/j.nlp.2024.100107>
- Grace, E., Sinaga, O. S., Silalahi, M., Ambarita, M. H., & Simatupang, S. (2022). Harga dan Kualitas Pelayanan Serta Pengaruhnya Terhadap Kepuasan Pelanggan. *Buletin Poltanesa*, 23(1), 35–44. <https://doi.org/10.51967/tanesa.v23i1.931>

- Hidayat, E. Y., Hardiansyah, R. W., & Affandy, A. (2021). Analisis Sentimen Twitter untuk Menilai Opini Terhadap Perusahaan Publik Menggunakan Algoritma Deep Neural Network. *Jurnal Nasional Teknologi Dan Sistem Informasi*, 7(2), 108–118. <https://doi.org/10.25077/teknosi.v7i2.2021.108-118>
- Hurnis, D., Ningsih, A., & Triani, M. (2023). Persepsi, Kualitas Layanan Antar Jemput Produk Dana Terhadap Kepuasan Pelanggan PT. BPR Pejuang Dharma Empat Lima. *Jurnal Ekonomi & Ekonomi Syariah*, 6(1), 804–809. <https://doi.org/10.36778/jesya.v6i1.1021>
- Irianto, M. D. R., Purbolaksono, M. D., & Bunyamin. (2024). Sentiment Analysis of Livin' by Mandiri Application Reviews Using Word2Vec Feature Extraction and KNN Method. *2024 International Conference on Data Science and Its Applications (ICoDSA)*, 236–241. <https://doi.org/10.1109/ICoDSA62899.2024.10651870>
- Ji, Z., Li, J. D., & Telgarsky, M. (2021). *Early-stopped neural networks are consistent*. <http://arxiv.org/abs/2106.05932>
- Julianto, Y., Setiabudi, D. H., & Rostianingsih, S. (2022). *Analisis Sentimen Ulasan Restoran Menggunakan Metode Support Vector Machine*. <https://publication.petra.ac.id/index.php/teknik-informatika/article/view/11888>
- Kim, D., & Kim, J.-K. (2022). *Contextual Skipgram: Training Word Representation Using Context Information*. <http://arxiv.org/abs/2102.08565>
- Kurniawati, F., Wibawa, A. P., Utama, A. B. P., & Ghosh, U. (2022). Sentiment analysis of wayang climen using naive bayes method. *Science in Information Technology Letters*, 3(2), 105–115. <https://doi.org/10.31763/sitech.v3i2.1220>
- Ma'rifah, H., Wibawa, A. P., & Akbar, M. I. (2020). Klasifikasi artikel ilmiah dengan berbagai skenario preprocessing. *Sains, Aplikasi, Komputasi Dan Teknologi Informasi*, 2(2), 70.
- Mijwel, M. M., Esen, A., & Shamil, A. (2019). Overview of Neural Networks. *Babylonian Journal of Machine Learning*, 42–45. <https://doi.org/10.58496/bjml/2023/008>
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Learning about Word Vector Representations and Deep Learning through Implementing Word2vec. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*.

- Mirayanti, N. K. R., Sariyasa, & Gunadi, G. A. (2023). Batch size and learning rate effect in covid-19 classification using CNN. In *Jurnal Mantik* (Vol. 7, Issue 3). Online.
- Najiyah, I. (2023). Analisis Sentimen Tanggapan Masyarakat Indonesia Tentang Kenaikan BBM Menggunakan Metode Artificial Neural Network. *Jurnal Responsif*, 5(1), 92–100. <https://ejurnal.ars.ac.id/index.php/jti>
- Oyedele, O. (2023). Determining the optimal number of folds to use in a K-fold cross-validation: A neural network classification experiment. *Research in Mathematics*, 10(1). <https://doi.org/10.1080/27684830.2023.2201015>
- Permadi, V. A. (2020). *Analisis Sentimen Menggunakan Algoritma Naive Bayes Terhadap Review Restoran di Singapura*. <https://www.kaggle.com/hj5992/restaurantreviews>
- Putu, N. L. P. M., Ahmad Zuli Amrullah, & Ismarmiaty. (2021). Analisis Sentimen dan Pemodelan Topik Pariwisata Lombok Menggunakan Algoritma Naive Bayes dan Latent Dirichlet Allocation. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(1), 123–131. <https://doi.org/10.29207/resti.v5i1.2587>
- Ramdhani, M. A., Maylawati, D. S., & Mantoro, T. (2020). Indonesian news classification using convolutional neural network. *Indonesian Journal of Electrical Engineering and Computer Science*, 19(2), 1000–1009. <https://doi.org/10.11591/ijeecs.v19.i2.pp1000-1009>
- Salsabillah, D. F., Ratnawati, D. E., & Setiawan, N. Y. (2024). Analisis Sentimen Ulasan Rumah Makan Menggunakan Perbandingan Algoritma Support Vector Machine dengan Naive Bayes (Studi Kasus: Ayam Goreng Nelongso Cabang Singosari, Malang). *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(1), 107–116. <https://doi.org/10.25126/jtiik.20241117584>
- Statista Research Department. (2024). *Revenue of the food market in Indonesia from 2020 to 2029*. <https://www.statista.com/statistics/1220026/indonesia-revenue-total-food-market/>
- Syahrudin, S., Fatmawati, F., & Suprajitno, H. (2022). The Formula Study in Determining the Best Number of Neurons in Neural Network Backpropagation Architecture with Three Hidden Layers. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 6(3), 397–402. <https://doi.org/10.29207/resti.v6i3.4049>
- Tabassum, A., & Patil, Dr. R. R. (2020). A Survey on Text Pre-Processing & Feature Extraction Techniques in Natural Language Processing. *International Research Journal of Engineering and Technology*, 07(06). www.irjet.net

- Ulgasesa, R., Negara, A. B. P., & Tursina, T. (2022). Pengaruh Stemming Terhadap Performa Klasifikasi Sentimen Masyarakat Tentang Kebijakan New Normal. *Jurnal Sistem Dan Teknologi Informasi (JustIN)*, 10(3), 286. <https://doi.org/10.26418/justin.v10i3.53880>
- Wardianto, Farikhin, & Nugraheni, D. M. K. (2023). *Analisis Sentimen Berbasis Aspek Ulasan Pelanggan Restoran Menggunakan LSTM Dengan Adam Optimizer* (Vol. 7, Issue 1). <http://publishing-widyagama.ac.id/ejournal-v2/index.php/jointecs>
- Widayat, W. (2021). Analisis Sentimen Movie Review menggunakan Word2Vec dan metode LSTM Deep Learning. *Jurnal Media Informatika Budidarma*, 5(3), 1018. <https://doi.org/10.30865/mib.v5i3.3111>
- Wisnu Brata, D., Purnomo, W., & Nofandi, A. (2024). Ekstraksi Ulasan Pelanggan Restoran dalam Menggali Aspek Layanan pada Google Review dan Tripadvisor sebagai Faktor Peningkatan Kualitas. *Jurnal Ilmiah Teknologi Informasi Asia*, 18(1).
- Wona, M. M. A., Asyifa, S. A., Virgianti, R., Hamid, M. N., Handoko, I. M., Sepriani, N. W. P., & Lestari, M. (2023). Klasifikasi Batik Indonesia Menggunakan Convolutional Neural Network (CNN). In *172 JURTI* (Vol. 7, Issue 2). <https://www.kaggle.com/datasets/dionisiusdh/indonesianbatik-motifs>.
- Zhafirah, N., & Silitonga, P. (2023). Meningkatkan Kepuasan Pelanggan di Soto Bu Tjondro Melalui Kepercayaan Merek. In *Panorama Nusantara* (Vol. 18). <http://ejournal.asaindo.ac.id/index.php/panorama>
- Zheng, M. (2024). Research on text classification based on neural networks. *Applied and Computational Engineering*, 41(1), 282–303. <https://doi.org/10.54254/2755-2721/41/20230768>