KLASIFIKASI TINGKAT KEPARAHAN DEPRESI PADA TEKS REDDIT MENGGUNAKAN FASTTEXT DAN BIGRU DENGAN MEKANISME ATTENTION

SKRIPSI

Oleh:
MUHAMMAD MUZAKKY MU'THY
NIM. 220605110175

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026

KLASIFIKASI TINGKAT KEPARAHAN DEPRESI PADA TEKS REDDIT MENGGUNAKAN FASTTEXT DAN BIGRU DENGAN MEKANISME ATTENTION

SKRIPSI

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

Oleh :
MUHAMMAD MUZAKKY MU'THY
NIM. 220605110175

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026

HALAMAN PERSETUJUAN

KLASIFIKASI TINGKAT KEPARAHAN DEPRESI PADA TEKS REDDIT MENGGUNAKAN FASTTEXT DAN BIGRU DENGAN MEKANISME ATTENTION

SKRIPSI

Oleh :
MUHAMMAD MUZAKKY MU'THY
NIM. 220605110175

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 06 Maret 2026

Pembimbing I,	Pembimbing II,
Dr. Cahyo Crysdian, M.Cs	Okta Qomaruddin Aziz, M.Kom
NIP. 19740424 200901 1 008	NIP. 199110192019031013

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

Sumiyono, M.Kom
NIP. 19841010 201903 1 012

HALAMAN PENGESAHAN

KLASIFIKASI TINGKAT KEPARAHAN DEPRESI PADA TEKS REDDIT MENGGUNAKAN FASTTEXT DAN BIGRU DENGAN MEKANISME ATTENTION

SKRIPSI

Oleh :
MUHAMMAD MUZAKKY MU'THY
NIM. 22060511024

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 11 Maret 2026

Susunan Dewan Penguji

Ketua Penguji	: Supriyono, M. Kom NIP. 19841010 201903 1 012	( )
Anggota Penguji I	: Khadijah F.H. 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. 199110192019031013	( )

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

Supriyono, M.Kom
NIP. 19841010 201903 1 012

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: MUHAMMAD MUZAKKY MU'THY
NIM	: 220605110175
Fakultas / Program Studi	: Sains dan Teknologi / Teknik Informatika
Judul Skripsi	: KLASIFIKASI TINGKAT KEPARAHAN DEPRESI PADA TEKS REDDIT MENGGUNAKAN FASTTEXT DAN BIGRU DENGAN MEKANISME *ATTENTION*

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, 21 Mei 2026
Yang membuat pernyataan,

MUHAMMAD MUZAKKY MU'THY
NIM. 220605110175

MOTTO

“And yet, day by day, we keep moving”

"Tidak semua kerumitan hidup harus diselesaikan, dan tidak semua yang patah menuntut untuk kembali utuh. Sebagian cukup dipahami, diarsipkan dengan baik, dan dijadikan pelita untuk perjalanan selanjutnya."

HALAMAN PERSEMBAHAN

Alhamdulillahi rabbil 'alamin, puji syukur senantiasa penulis panjatkan ke hadirat Allah *Subḥānahu wa Ta'ālā* atas segala limpahan rahmat, hidayah, serta karunia-Nya sehingga penyusunan skripsi ini dapat diselesaikan dengan baik. Shalawat seiring salam semoga terus tercurahkan kepada junjungan besar Nabi Muhammad *Ṣallallāhu 'Alaihi Wasallam* yang telah membimbing umat manusia menuju cahaya kebenaran dan peradaban *addinul Islam* yang mulia. Melalui lembar persembahan ini, penulis bermaksud menyampaikan rasa terima kasih dan penghormatan yang setinggi-tingginya kepada berbagai pihak yang telah menjadi pilar kekuatan selama menempuh pendidikan di jenjang perguruan tinggi.

Karya tulis ini secara khusus penulis persembahkan sebagai wujud bakti dan cinta kasih kepada kedua orang tua tercinta yang tidak pernah lelah merapalkan doa terbaik bagi kelancaran studi penulis. Dukungan moral, pengorbanan yang tak terhitung jumlahnya, serta kasih sayang mereka merupakan sumber energi utama yang membuat penulis sanggup melewati berbagai rintangan hingga mencapai tahap akhir ini. Selain itu, rasa terima kasih yang mendalam juga ditujukan kepada saudara-saudara penulis atas segala bentuk motivasi, semangat, serta bantuan yang terus mengalir tanpa henti sejak awal masa perkuliahan hingga draf skripsi ini benar-benar rampung.

Perjalanan akademik ini tentunya tidak akan terasa bermakna tanpa kehadiran orang-orang terdekat yang selalu mendampingi dalam setiap proses pembelajaran dan pendewasaan diri. Penulis mendedikasikan apresiasi ini kepada

sahabat-sahabat masa Sekolah Menengah Atas yang terus menjaga tali silaturahmi, serta teman-teman seperjuangan angkatan 2022 yang saling merangkul agar bisa lulus bersama. Tidak luput, ucapan terima kasih yang amat istimewa penulis sampaikan kepada keluarga besar kontrakan penghuni surga atas kebersamaan, canda tawa, dan kehangatan yang selalu menjadi tempat pulang paling nyaman di tengah penatnya menempuh revisi. Apresiasi yang sama juga penulis alamatkan kepada beberapa pihak lainnya yang tidak dapat disebutkan satu per satu, namun telah memberikan andil besar dalam kelancaran penyelesaian karya ilmiah ini.

Pada akhirnya, sebuah apresiasi yang paling jujur dan mendalam penulis tujukan untuk diri sendiri. Terima kasih karena telah memilih untuk terus bertahan, berjuang, dan menolak untuk menyerah di setiap fase yang penuh tekanan dan keraguan. Terima kasih atas segala kerja keras, keteguhan hati, serta malam-malam panjang yang telah dilalui di depan layar demi menyelesaikan tanggung jawab ini hingga tuntas ke garis akhir.

Penulis menyadari sepenuhnya bahwa keberhasilan menyelesaikan skripsi ini adalah hasil dari dukungan kolektif dan kebaikan hati banyak orang yang tidak akan mampu terbalas dengan sepadan. Oleh karena itu, penulis hanya dapat memanjatkan doa tulus agar seluruh bantuan, waktu, dan energi yang telah dicurahkan oleh semua pihak bernilai ibadah serta mendapatkan ganjaran kebaikan yang berlipat ganda dari Allah *Subḥānahu wa Ta'ālā*. Semoga langkah kita semua ke depannya senantiasa diberkahi, dan karya sederhana ini diharapkan mampu memberikan manfaat yang nyata bagi perkembangan keilmuan, khususnya di bidang teknik informatika.

KATA PENGANTAR

Alhamdulillahi rabbil 'alamin, puji syukur senantiasa penulis panjatkan ke hadirat Allah *Subḥānahu wa Ta'ālā* atas segala limpahan rahmat, hidayah, serta kasih sayang-Nya yang telah memberikan kemudahan dan kelancaran bagi penulis untuk merampungkan karya ilmiah yang berjudul "KLASIFIKASI TINGKAT KEPARAHAN DEPRESI PADA TEKS REDDIT MENGGUNAKAN FASTTEXT DAN BIGRU DENGAN MEKANISME *ATTENTION*". Shalawat seiring salam semoga terus tercurahkan kepada junjungan besar Nabi Muhammad *Ṣallallāhu 'Alaihi Wasallam*, teladan utama umat manusia yang syafaatnya selalu kita harapkan di hari akhir nanti, *Aamiin*.

Penulis menyadari sepenuhnya bahwa perjalanan panjang menyelesaikan skripsi ini tidak akan pernah terwujud tanpa adanya bimbingan, ketulusan, serta dukungan kolektif dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis ingin memberikan rasa terima kasih dan penghormatan yang setinggi-tingginya kepada:

1. Prof. Dr. Hj. Ilfi Nurdiana, M.Si., CAHRM., CRMP selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang, atas kesempatan dan fasilitas yang diberikan selama penulis menempuh pendidikan.
2. Dr. Agus Mulyono, M.Kes. selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang, atas segala dukungan dan kebijakan selama masa studi penulis.

3. Supriyono, M.Kom. selaku Ketua Penguji, yang telah meluangkan waktu dan memberikan arahan serta evaluasi yang sangat berharga demi kesempurnaan skripsi ini.
4. Khadijah F.H. Holle, M.Kom. selaku Anggota Penguji I, atas ketelitian, masukan ilmiah, dan kritik konstruktif yang sangat bermanfaat bagi penelitian ini.
5. Dr. Cahyo Crysdian, M.Cs. selaku Anggota Penguji II sekaligus Dosen Pembimbing I, yang telah melimpahkan kesabaran, waktu, serta bimbingan teoretis yang mendalam dari awal hingga akhir.
6. Okta Qomaruddin Aziz, M.Kom. selaku Anggota Penguji III sekaligus Dosen Pembimbing II, atas segala kebaikan, bimbingan teknis, dan saran akademis yang krusial dalam menyempurnakan skripsi ini.
7. Segenap Dosen, Staf Administrasi, dan Laboran, yang telah membekali penulis dengan ilmu pengetahuan serta memberikan bantuan layanan administratif selama masa perkuliahan.
8. Kedua orang tua serta saudara-saudara tercinta, pilar kekuatan utama yang tiada henti melangitkan doa, memberikan pengorbanan, dan motivasi agar penulis dapat menuntaskan studi dengan lancar.
9. Teman-teman sejawat dan sahabat seperjuangan angkatan 2022, rekan bertumbuh yang luar biasa, yang selalu menjadi teman diskusi yang kritis dan saling menguatkan dalam suka maupun duka.

Terakhir, sebuah apresiasi yang paling jujur penulis tujukan untuk diri sendiri. Terima kasih karena telah memilih untuk terus bertahan, berjuang, dan

menolak untuk menyerah di setiap fase yang melelahkan. Terima kasih atas segala kerja keras, keteguhan hati, serta malam-malam panjang yang telah dilalui di depan layar demi menyelesaikan tanggung jawab ini hingga ke garis akhir. Semoga karya tulis ini dapat membawa keberkahan dan manfaat bagi perkembangan keilmuan ke depannya.

Malang, 22 April 2026

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Mu'thy, Muhammad Muzakky. 2026. **Klasifikasi Tingkat Keparahan Depresi pada Teks Reddit Menggunakan FASTTEXT DAN BIGRU dengan Mekanisme *Attention***. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Cahyo Crysdian (II) Okta Qomaruddin, M.Kom.

.

Kata Kunci: Depresi, Klasifikasi *Ordinal*, BiGRU, Mekanisme Atensi, Pembobotan Kelas, FastText.

Depresi merupakan gangguan kesehatan mental yang membutuhkan penanganan dini secara tepat. Pemanfaatan teknologi *Natural Language Processing* (NLP) memungkinkan identifikasi tingkat keparahan depresi melalui analisis teks pengguna di media sosial Reddit. Penelitian ini bertujuan mengukur kinerja *model Bidirectional Gated Recurrent Unit* (BiGRU) dengan mekanisme atensi (*attention mechanism*) dan pembobotan kelas (*class weighting*) dalam mengklasifikasikan empat level depresi (*minimum, mild, moderate, severe*) secara *ordinal*. Guna mengatasi tantangan ketidakseimbangan kelas (*class imbalance*) yang sangat ekstrem pada dataset, penelitian ini merancang *ablation study* menjadi empat skenario (*Baseline, Class Weight Only, Attention Only*, dan *Full Model*) menggunakan representasi kata FastText dan *Ordinal Output Layer*. Hasil uji coba melalui *Stratified 5-Fold Cross validation* menunjukkan skenario *Class Weight Only* memberikan performa paling optimal dan berkeadilan lintas kelas, dengan mencetak *Macro F1-Score* 32,51% serta *Quadratic weighted kappa* (QWK) tertinggi di angka 29,66%. Sebaliknya, skenario *Attention Only* justru memicu paradoks akurasi; di mana akurasi globalnya tampak tertinggi (72,60%) namun nilai QWK anjlok drastis hingga 4,68% akibat kegagalan sistematis mengenali kelas minoritas. Sementara itu, *Full Model* menghasilkan ketidakstabilan konvergensi dengan rata-rata QWK 21,71%. Penelitian ini menyimpulkan bahwa intervensi pembobotan kelas pada fungsi kerugian (*loss function*) jauh lebih fundamental dan kritis dalam memitigasi bias mayoritas dibandingkan penambahan kompleksitas arsitektur mekanisme atensi pada pemodelan teks kesehatan mental yang distribusinya sangat timpang.

ABSTRACT

Mu'thy, Muhammad Muzakky. 2026. **Depression Severity Classification on Reddit Text Using FASTTEXT and BIGRU with Attention Mechanism**. Undergraduate Thesis. Department of Informatics, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisors: (I) Dr. Cahyo Crysdian (II) Okta Qomaruddin, M.Kom.

Key words: Depression, Ordinal Classification, BiGRU, Attention Mechanism, Class weighting, FastText.

Depression is a mental health disorder that requires timely and appropriate intervention. The utilization of Natural Language Processing (NLP) technology enables the identification of depression severity through the analysis of user text on the Reddit social media platform. This study aims to evaluate the performance of a Bidirectional Gated Recurrent Unit (BiGRU) model equipped with an attention mechanism and class weighting in ordinally classifying four levels of depression (minimum, mild, moderate, severe). To address the extreme class imbalance within the dataset, an ablation study was designed across four configurations (Baseline, Class Weight Only, Attention Only, and Full Model) leveraging FastText word representations and an Ordinal Output Layer. Experimental results using Stratified 5-Fold Cross-Validation demonstrate that the Class Weight Only scenario delivers the most optimal and equitable performance across classes, achieving a Macro F1-Score of 32.51% and the highest Quadratic weighted kappa (QWK) at 29.66%. Conversely, the Attention Only scenario triggers an accuracy paradox, where the global accuracy appears highest (72.60%) but the QWK drops drastically to 4.68% due to a systematic failure to recognize minority classes. Meanwhile, the Full Model leads to convergence instability with an average QWK of 21.71%. This study concludes that class weighting intervention in the loss function is far more fundamental and critical in mitigating majority bias than increasing architectural complexity through attention mechanisms in mental health text modeling with highly skewed distributions.

البحث مستخلص

معطي، محمد مزكي .2026 .**تصنيف مستوى شدة الاكتئاب في نصوص ريديت باستخدام FastText و BiGRU مع آلية الانتباه** .أطروحة. قسم هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية، مالانج. المشرفون: (الأول) الدكتور كاهيو كريسديان، (الثاني) أوكتا قمر الدين، ماجستير في علوم الحاسوب.

الكلمات المفتاحية :الاكتئاب، التصنيف الترتيبي، BiGRU، آلية الانتباه، ترجيح الفئة، .FastText

يعد الاكتئاب اضطراباً في الصحة العقلية يتطلب تدخلاً مبكراً ومناسباً. يتيح استخدام تقنية معالجة اللغات الطبيعية (NLP) تحديد مستوى شدة الاكتئاب من خلال تحليل النصوص التي يكتبها المستخدمون على منصة التواصل الاجتماعي ريديت (Reddit). تهدف هذه الدراسة إلى تقييم أداء نموذج الوحدة المتكررة ذات البوابة ثنائية الاتجاه (BiGRU) المزود بآلية الانتباه (*Attention Mechanism*) وترجيح الفئة (*Class weighting*) في التصنيف الترتيبي لأربعة مستويات من الاكتئاب (الأدنى، الخفيف، المتوسط، الشديد). للتغلب على تحدي عدم توازن الفئات (*Class imbalance*) الشديد جداً في مجموعة البيانات، صممت هذه الدراسة تجربة استئصال (*Ablation study*) مقسمة إلى أربعة سيناريوهات (الأساس، ترجيح الفئة فقط، الانتباه فقط، والنموذج الكامل) باستخدام تمثيل الكلمات FastText وطبقة المخرجات الترتيبية (*Ordinal Output Layer*). أظهرت نتائج الاختبار باستخدام التحقق المتقاطع الطبقي خماسي الطيات (Stratified 5-*Fold Cross validation*) أن سيناريو "ترجيح الفئة فقط" (*Class Weight Only*) قدم الأداء الأمثل والأكثر عدالة عبر الفئات، حيث سجل درجة *Macro F1-Score* بلغت 32.51% وأعلى قيمة لمعامل كابا الموزون تربيعياً (QWK) بنسبة 29.66%. وعلى النقيض من ذلك، تسبب سيناريو "الانتباه فقط" (*Attention Only*) في مفارقة الدقة؛ حيث بدت الدقة الإجمالية هي الأعلى (72.60%) ولكن انخفضت قيمة QWK بشكل كبير إلى 4.68% بسبب الفشل المنهجي في التعرف على فئات الأقلية. وفي الوقت نفسه، أدى "النموذج الكامل" (*Full Model*) إلى عدم استقرار التقارب بمتوسط QWK بلغ 21.71%. تخلص هذه الدراسة إلى أن التدخل من خلال ترجيح الفئات في دالة الخسارة (*Loss Function*) يعد أكثر جوهرية وأهمية في التخفيف من تحيز الأغلبية مقارنة بزيادة تعقيد البنية من خلال آليات الانتباه في نمذجة نصوص الصحة العقلية ذات التوزيع غير المتكافئ للغاية.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Depresi merupakan gangguan mental yang umum, ditandai oleh suasana perasaan sedih, kehilangan minat atau kesenangan, gangguan tidur, penurunan energi, gangguan konsentrasi, serta munculnya pikiran tentang kematian atau bunuh diri. Kondisi ini berbeda dari perubahan suasana hati yang bersifat sementara, gejala tersebut berlangsung *minimum* dua minggu dan berdampak signifikan terhadap fungsi sosial, pekerjaan, serta aktivitas sehari-hari individu. Depresi kini menjadi perhatian serius, terutama di kalangan remaja dan mahasiswa. Secara global, World Health Organization (WHO) melaporkan bahwa pada tahun 2021 terdapat sekitar 727 ribu kematian akibat bunuh diri di seluruh dunia (World Health Organization, 2025b). Selain itu, pada tahun 2024 tercatat 332 juta orang di dunia menderita depresi. Depresi yang tidak tertangani dengan baik berpotensi mengarah pada tindakan tragis seperti bunuh diri (World Health Organization, 2025a).

Perkembangan masalah kesehatan mental juga semakin terkait dengan perubahan cara manusia berinteraksi dan mengekspresikan diri di ruang digital. Pada awal 2024, jumlah pengguna media sosial global diperkirakan mencapai 5,04 miliar atau sekitar 62,3% populasi dunia, sehingga ruang daring menjadi salah satu arena utama terbentuknya pengalaman sosial modern. Skala ini menunjukkan bahwa emosi, keluhan, dan pergulatan batin, termasuk yang berkaitan dengan depresi, semakin sering terdokumentasi dalam bentuk teks yang tersebar luas (Kemp, 2025).

Kelompok usia muda yang sering dibahas dalam konteks depresi juga termasuk pengguna media sosial yang paling intensif. Survei *Pew Research Center* menunjukkan bahwa sekitar 74% orang dewasa usia di bawah 30 tahun menggunakan setidaknya lima platform media sosial yang ditanyakan dalam survei, menggambarkan tingginya paparan terhadap interaksi dan konten lintas platform (Beshay, 2025). Pada saat yang sama, ruang daring kerap dipersepsikan sebagai tempat memperoleh dukungan. Survei *Pew* melaporkan 52% remaja menyatakan media sosial membuat mereka merasa memiliki orang yang dapat mendukung mereka melalui masa sulit (berdasarkan survei akhir 2024) (Atske, 2025).

Di antara berbagai platform, forum berbasis komunitas seperti Reddit menjadi relevan karena diskusinya bertema dan sering memuat narasi personal yang panjang, termasuk topik kesehatan mental. Secara skala, Reddit melaporkan 116,0 juta *Daily active uniques* (DAUq) pada kuartal III 2025, yang mengindikasikan tingginya aktivitas pengguna (Reddit, 2025). Literatur juga mencatat bahwa Reddit telah banyak dimanfaatkan sebagai sumber data penelitian depresi dan kecemasan, namun tetap menekankan perlunya kehati-hatian karena isu representativitas pengguna, bias komunitas, serta pertimbangan etika pemanfaatan data (Boettcher, 2021).

Sejalan dengan arah riset kesehatan mental berbasis teks, fokus kajian tidak hanya berhenti pada deteksi biner “terindikasi atau tidak”, tetapi bergerak menuju pemodelan tingkat keparahan yang bersifat bertahap. Pendekatan klasifikasi dengan data yang bersifat *ordinal* penting karena level depresi memiliki urutan yang bermakna, sehingga kesalahan prediksi antar tingkat seharusnya dipahami secara

proporsional (misalnya kesalahan antar level berdekatan berbeda implikasinya dengan kesalahan yang melompat jauh). Naseem dkk. (2022) menekankan bahwa pemodelan tingkat depresi berbasis teks perlu mempertimbangkan sifat *ordinal* tersebut agar lebih selaras dengan konsep keparahan yang bergradasi.

Dalam sebuah hadis yang diriwayatkan oleh Abu Juhaifah (RA), Rasulullah SAW membenarkan nasihat sahabatnya Salman al-Farisi:

إِنَّ لِرَبِّكَ عَلَيْكَ حَقًّا وَلِنَفْسِكَ عَلَيْكَ حَقًّا وَلِأَهْلِكَ عَلَيْكَ حَقًّا فَأَعْطِ كُلَّ ذِي حَقٍّ

"Sungguh, Tuhanmu memiliki hak yang harus kaupenuhi, dirimu memiliki hak yang harus kaupenuhi, keluargamu juga memiliki hak yang harus kaupenuhi, maka berikanlah hak mereka secara proporsional." (*HR. Bukhari*).

Hadis ini menegaskan pentingnya menjaga keseimbangan dalam kehidupan, termasuk memenuhi hak diri sendiri dengan menjaga kesehatan fisik maupun mental. Menjaga kestabilan emosional dan kesehatan jiwa merupakan bagian dari tanggung jawab pribadi seorang Muslim terhadap dirinya sendiri. Dalam konteks penelitian ini, hadis tersebut relevan karena menekankan pentingnya kesadaran individu untuk merawat kesehatan mental dan mencegah kondisi yang dapat mengarah pada depresi.

Al-Qur'an juga menegaskan pentingnya menjaga kesehatan jiwa seperti yang Allah SWT firmankan di dalam Surat Al-Baqarah [2]:286:

لَا يُكَلِّفُ اللّٰهُ نَفْسًا اِلَّا وُسْعَهَاۗ لَهَا مَا كَسَبَتْ وَعَلَيْهَا مَا اكْتَسَبَتْۗ رَبَّنَا لَا تُؤَاخِذْنَآ اِنْ نَّسِيْنَآ اَوْ اَخْطَأْنَاۚ رَبَّنَا وَلَا تَحْمِلْ عَلَيْنَآ اِصْرًا كَمَا حَمَلْتَهٗ عَلَى الَّذِيْنَ مِنْ قَبْلِنَاۚ رَبَّنَا وَلَا تُحَمِّلْنَا مَا لَا طَاقَةَ لَنَا بِهٖۚ وَاعْفُ عَنَّاۗ وَاغْفِرْ لَنَاۗ وَارْحَمْنَاۗ اَنْتَ مَوْلٰىنَا فَانْصُرْنَا عَلَى الْقَوْمِ الْكٰفِرِيْنَ ٢٨٦

"Allah tidak membebani seseorang, kecuali menurut kesanggupannya. Baginya ada sesuatu (pahala) dari (kebajikan) yang diusahakannya dan terhadapnya ada (pula) sesuatu (siksa) atas (kejahatan) yang diperbuatnya. (Mereka berdoa,)

"Wahai Tuhan kami, janganlah Engkau hukum kami jika kami lupa atau kami salah. Wahai Tuhan kami, janganlah Engkau bebani kami dengan beban yang berat sebagaimana Engkau bebankan kepada orang-orang sebelum kami. Wahai Tuhan kami, janganlah Engkau pikulkan kepada kami apa yang tidak sanggup kami memikulnya. Maafkanlah kami, ampunilah kami, dan rahmatilah kami. Engkaulah pelindung kami. Maka, tolonglah kami dalam menghadapi kaum kafir."

Ayat ini menegaskan bahwa Allah tidak membebani manusia melebihi kesanggupannya dan mengajarkan doa agar beban yang melampaui kemampuan diturunkan, kesalahan dimaafkan, dan rahmat-Nya meliputi hamba. Pesan ini memberi landasan teologis yang menentramkan bagi upaya menjaga kesehatan jiwa dari kesulitan batin, termasuk gejala depresi yang diakui sebagai beban yang boleh dimintakan keringanan, bukan untuk dipikul sendirian sampai mematahkan. Dalam perspektif tujuan utama syariat (*maqāṣid al-syarī'ah*), khususnya *ḥifẓ al-nafs* (perlindungan jiwa), ikhtiar deteksi dini dan penanganan depresi adalah bagian dari menjaga amanah kehidupan sekaligus wujud tawakal yang aktif seperti berdoa memohon keringanan, mencari pertolongan profesional, serta membangun dukungan sosial yang mencegah beban melampaui batas kemampuan. Dengan demikian, intervensi ilmiah pada kesehatan mental tidak bertentangan dengan nilai-nilai Islam, justru selaras dengan tuntunan syariat untuk meringankan kesulitan, merawat diri, dan memelihara keberlangsungan hidup secara bermartabat.

Pemanfaatan teknologi *Natural Language Processing* (NLP) dan *deep learning* memungkinkan komputer mengidentifikasi tulisan yang mengandung indikasi depresi, bahkan mengestimasi tingkat keparahannya. Dalam konteks ini, arsitektur *Bidirectional Gated Recurrent Unit* (BiGRU) dengan mekanisme atensi (*attention*) telah menunjukkan keunggulan signifikan untuk klasifikasi teks

kesehatan mental. *Model* BiGRU mampu memproses informasi teks secara *bidirectional* (dari kiri ke kanan dan sebaliknya), sehingga dapat menangkap konteks lengkap dari sebuah kalimat. Penelitian Zogan dkk. (2021) mengembangkan *Depressionnet* yang menggabungkan CNN, BiGRU bertingkat dua lapis, dan mekanisme atensi, mencapai akurasi 90,1% pada dataset Twitter. Studi komparatif Zhang dkk. (2024) pada 527.333 unggahan media sosial Cina menunjukkan bahwa BiGRU dengan atensi mencapai akurasi tertinggi 92,77% dan *F1-Score* 92,64%, mengungguli berbagai varian RNN lainnya.

Keunggulan mekanisme atensi terletak pada kemampuannya memberikan bobot kepentingan berbeda pada setiap kata dalam teks, sehingga *model* dapat fokus pada kata-kata kunci yang paling relevan untuk mendeteksi depresi (seperti \"*hopeless*\", \"*worthless*\", atau \"*suicidal thoughts*\"). Penelitian menunjukkan bahwa mekanisme atensi memberikan peningkatan akurasi sekitar 10-15% dibanding *model* tanpa atensi, sekaligus meningkatkan interpretasi melalui visualisasi bobot atensi. Untuk representasi teks, FastText dipilih karena kemampuannya menangani *kata di luar kosakata* (*OOV*) dan morfologi yang kaya melalui pendekatan *subword* (*character N-grams*). FastText terbukti *robust* terhadap variasi linguistik seperti salah tulis, slang, dan neologisme yang umum pada teks media sosial.

Aspek penting lain adalah klasifikasi tingkat keparahan depresi secara *ordinal*. Sebagian besar penelitian sebelumnya fokus pada klasifikasi biner (depresi atau tidak depresi), padahal dalam konteks klinis, depresi bersifat spektrum atau bertingkat: *minimum*, *mild*, *moderate*, hingga *severe*. Naseem dkk. (2022)

memperkenalkan pendekatan klasifikasi dengan data yang bersifat *ordinal* yang secara eksplisit memodelkan sifat hierarkis tingkat keparahan, terbukti lebih unggul dibanding klasifikasi multi-kelas konvensional. Pendekatan ini menggunakan dataset 3.553 unggahan Reddit yang dianotasi oleh ahli klinis menggunakan standar *Beck's Depression Inventory* (BDI-II), dengan distribusi: 2.587 unggahan *minimum*, 290 *mild*, 394 *moderate*, dan 282 *severe*.

Ketidakseimbangan kelas (*class imbalance*) menjadi tantangan tersendiri dalam dataset ini. Untuk mengatasinya, penelitian ini menerapkan strategi pembobotan kelas (*class weighting*) pada fungsi *loss* yang memberikan bobot lebih besar pada kelas minoritas. Zhang dkk. (2024) menunjukkan bahwa penerapan teknik *handling imbalance* dapat meningkatkan performa *model* dari akurasi 67,99% menjadi 91,52%. Penelitian ini merancang eksperimen multi-skenario untuk mengisolasi kontribusi individual dari pembobotan kelas (*class weighting*) dan mekanisme atensi, serta evaluasi efek interaksi keduanya.

Penelitian mengenai deteksi dan klasifikasi tingkat keparahan depresi berbasis teks diharapkan memiliki nilai akademis, kemanusiaan, dan religius. Melalui pengembangan sistem klasifikasi otomatis, institusi pendidikan dan profesional kesehatan mental dapat mengidentifikasi mahasiswa yang menunjukkan tanda-tanda depresi dari tulisan mereka di media sosial. Langkah ini memungkinkan intervensi dilakukan sebelum kondisi mental memburuk, sejalan dengan prinsip perlindungan jiwa dalam kerangka tujuan-tujuan utama syariat.

1.2 Pernyataan Masalah

Seberapa baik kinerja *model* BiGRU dengan mekanisme atensi dan pembobotan kelas (*class weighting*) dalam mengklasifikasikan empat tingkat keparahan depresi (*minimum*, *mild*, *moderate*, *severe*) dari teks media sosial Reddit berdasarkan metrik akurasi, presisi, *Recall*, *F1-Score*, *Mean absolute error* (*MAE*), dan *Quadratic weighted kappa* (QWK)?

1.3 Tujuan Penelitian

Mengukur kinerja *model* BiGRU dengan mekanisme atensi dan pembobotan kelas (*class weighting*) dalam mengklasifikasikan empat tingkat keparahan depresi (*minimum*, *mild*, *moderate*, *severe*) dari teks media sosial Reddit berdasarkan metrik akurasi, presisi, *Recall*, *F1-Score*, *Mean absolute error* (*MAE*), dan *Quadratic weighted kappa* (QWK).

1.4 Batasan Masalah

a. Data yang digunakan dalam penelitian ini berasal dari teks yang diunggah di *Reddit*, yang berkaitan dengan pengalaman pengguna tentang depresi.
b. *Model* tidak mempertimbangkan konteks temporal atau riwayat unggahan pengguna secara runtut kronologis, melainkan mengklasifikasikan setiap unggahan secara mandiri sebagai potret sesaat kondisi mental.

1.5 Manfaat Penelitian

Penelitian ini berpotensi menghadirkan alat *screening*/triase awal berbasis teks yang dapat menandai tingkat risiko keparahan depresi secara cepat dan terstruktur. Keluaran *model* dipetakan ke kategori risiko terukur untuk membantu

penentuan prioritas penanganan, sehingga kasus berisiko tinggi dapat diprioritaskan untuk ditindaklanjuti lebih dahulu, sementara kasus berisiko sedang atau rendah dapat diarahkan ke jalur dukungan yang sesuai. Dengan demikian, pemanfaatan waktu konsultasi menjadi lebih efisien dan cakupan layanan dapat meningkat, terutama ketika ketersediaan psikiater, psikolog klinis, dan tenaga kesehatan masih terbatas.

BAB II

STUDI PUSTAKA

2.1 Klasifikasi dengan data yang bersifat *ordinal* untuk Depresi

Deteksi depresi dari teks media sosial telah berkembang dari pendekatan biner sederhana menuju klasifikasi multi-level yang lebih bernuansa klinis. Pendekatan klasifikasi dengan data yang bersifat *ordinal*, yang secara eksplisit memodelkan sifat hierarkis tingkat keparahan, menawarkan keunggulan dibanding klasifikasi multi-kelas konvensional pada domain kesehatan mental. Bagian ini mengulas penelitian yang menerapkan klasifikasi dengan data yang bersifat *ordinal* untuk mendeteksi tingkat keparahan depresi, dengan fokus pada metodologi, kinerja *model*, dan relevansinya untuk penelitian klasifikasi dengan data yang bersifat *ordinal* depresi.

Naseem dkk. (2022) memperkenalkan pendekatan awal dalam makalahnya yang dipresentasikan pada The Web Conference 2022. Mereka menyusun dataset berjumlah 3.553 unggahan Reddit dengan distribusi 2.587 unggahan untuk kategori *minimum*, 290 untuk *mild*, 394 untuk *moderate*, dan 282 untuk *severe*. Anotasi dilakukan oleh ahli klinis menggunakan standar Beck's Depression Inventory (BDI-II) dan skema Depressive Disorder Annotation (DDA). Arsitektur yang diusulkan menggabungkan *Text graph convolutional network* untuk ekstraksi fitur, *Bidirectional* LSTM dengan mekanisme perhatian, serta lapisan klasifikasi dengan data yang bersifat *ordinal* yang dioptimasi menggunakan distribusi probabilitas lunak. *Model* ini melampaui *model* dasar terkini seperti SVM, Random Forest,

MLP, CNN, RNN, LSTM, BiLSTM, dan *Depressionnet*, dengan keunggulan utama pada kemampuan memodelkan pengurutan *ordinal* antar tingkat keparahan yang mencerminkan kontinum klinis depresi.

Secara interpretatif, empat label *minimum*, *mild*, *moderate,* dan *severe* merepresentasikan gradasi keparahan yang berurutan. Semakin tinggi kategorinya, semakin kuat dan semakin banyak indikator gejala depresi yang tercermin dalam teks, serta semakin besar potensi dampaknya pada fungsi sehari-hari. Mengacu pada BDI-II (*Beck Depression Inventory-II*, yaitu kuesioner pelaporan mandiri berisi 21 butir, masing-masing diberi skor 0–3, dengan total skor 0–63), kategori *minimum* umumnya menandakan indikasi gejala sangat ringan atau nyaris tidak ada pada rentang skor 0–13. Tingkat *mild* menandakan gejala ringan yang mulai jelas tetapi fungsi harian masih relatif terjaga di rentang skor 14–19, dan tingkat *moderate* menandakan gejala yang lebih konsisten serta mengganggu dengan dampak fungsi yang nyata di rentang skor 20–28. Pada tingkat *severe*, kondisi ini menandakan gejala berat yang terus-menerus dengan gangguan fungsi yang menonjol, serta dapat memuat sinyal risiko tinggi seperti gagasan kematian atau bunuh diri di rentang skor 29–63 (Ariani dkk., 2023).

Sejalan dengan hal itu, DDA (*Depressive Disorder Annotation*) dipakai sebagai kerangka anotasi untuk menandai ada atau tidaknya bukti depresi klinis di dalam sebuah teks. Kerangka ini juga berfungsi secara khusus untuk mengidentifikasi komponen gejala depresi dan pemicu stres psikososial yang dialami seseorang. Dengan demikian, pemetaan tingkat keparahan tersebut dapat

dilakukan secara tepat dengan tetap mempertimbangkan keterurutan atau hierarki antar level (Mowery dkk., 2015).

Berdasarkan penelitian tersebut, klasifikasi dengan data yang bersifat *ordinal* terbukti efektif dalam memetakan tingkat keparahan depresi karena kemampuannya memodelkan sifat hierarkis dan keterurutan antar level keparahan, sehingga lebih selaras dengan kontinuitas kondisi klinis depresi dibandingkan pendekatan klasifikasi multi-kelas konvensional. Dengan memperhitungkan hubungan *ordinal* antar kategori, pendekatan ini tidak hanya meningkatkan akurasi prediksi, tetapi juga mengurangi kesalahan klasifikasi yang bersifat ekstrem antar tingkat keparahan. Oleh karena itu, klasifikasi dengan data yang bersifat *ordinal* menjadi pendekatan yang relevan dan tepat untuk digunakan dalam penelitian deteksi tingkat keparahan depresi berbasis teks media sosial.

2.2 Arsitektur BiGRU dan Atensi untuk Deteksi Kesehatan Mental

Bidirectional Gated Recurrent Unit (BiGRU) dengan mekanisme perhatian telah muncul sebagai arsitektur yang sangat efektif untuk klasifikasi teks kesehatan mental karena menawarkan keseimbangan antara kinerja, efisiensi komputasi, dan interpretasi. Bagian ini mengulas penelitian yang mengimplementasikan BiGRU dan membandingkannya dengan arsitektur alternatif, dengan fokus pada keunggulan arsitektural serta relevansinya untuk aplikasi deteksi depresi.

Zogan dkk. (2021) mengembangkan *Depressionnet*, sebuah arsitektur hibrida yang dipresentasikan pada ACM SIGIR Conference 2021. *Model* ini menggabungkan CNN untuk ekstraksi fitur lokal, BiGRU bertingkat dua lapis, dan mekanisme perhatian aditif. Konfigurasi menggunakan 64 neuron tersembunyi per

lapis BiGRU dan dilatih pada data 4.208 pengguna Twitter yang berisi 1.797.303 tweet, dengan 51,3% pengguna terdiagnosis depresi. Hasil eksperimen menunjukkan akurasi 90,1%, presisi 90,9%, *Recall* 90,4%, dan *F1-Score* 91,2%, dengan mekanisme perhatian memberikan peningkatan 10–15% dibanding BiGRU tanpa perhatian. *Model* dilatih menggunakan pengoptimalan Adam dengan laju pembelajaran 0,001, ukuran batch 16, dan 40 *epoch*, memanfaatkan *embedding* Word2Vec pra-latih berdimensi 300 dari GoogleNews. Penelitian ini menunjukkan bahwa pemrosesan *bidirectional* pada BiGRU mampu menangkap ketergantungan jarak jauh dan fitur kontekstual dari kedua arah, sebuah aspek penting dalam memahami ekspresi kesehatan mental di mana makna sering bergantung pada konteks kalimat penuh.

Zhang dkk. (2024) melakukan studi komparatif sistematis yang dipublikasikan di JMIR Mental health dengan menganalisis 527.333 unggahan dari 3.200 pengguna media sosial Cina menggunakan berbagai varian RNN (LSTM, BiLSTM, GRU, BiGRU). Mereka menemukan bahwa BiGRU dengan mekanisme perhatian mencapai kinerja tertinggi dengan akurasi 92,77% dan *F1-Score* 92,64%, mengungguli BiLSTM-Atensi (91,35%), GRU (92,25%), BiLSTM (84,95%), dan LSTM (88,41%). *Model* dikonfigurasi dengan ukuran tersembunyi dan ukuran perhatian 256, dilatih menggunakan PyTorch pada GPU Tesla A100 dengan laju pembelajaran 1e-3. Menariknya, *model* dasar BiGRU tanpa sampling khusus hanya mencapai akurasi 67,99%, namun setelah penerapan pengambilan sampel berbasis retrieval (pengambilan kembali informasi) yang memfilter unggahan relevan dengan depresi melalui pencocokan kata kunci, kinerja meningkat drastis menjadi

91,52%. Temuan utama adalah mekanisme perhatian memberikan kenaikan kinerja signifikan, dan sampling berbasis retrieval (pengambilan kembali informasi) membantu mengurangi *noise* dengan memfokuskan *model* pada konten informatif.

Berdasarkan kajian di atas, arsitektur *Bidirectional* Gated Recurrent Unit (BiGRU) dengan mekanisme perhatian banyak digunakan dalam klasifikasi teks kesehatan mental karena mampu menangkap konteks berurutan dari dua arah, namun tetap relatif ringan dibanding arsitektur yang lebih kompleks. Dibandingkan LSTM, GRU memiliki struktur yang lebih sederhana sehingga umumnya lebih efisien tanpa kehilangan kemampuan memodelkan dependensi urutan yang penting pada teks psikologis. Mekanisme perhatian melengkapi BiGRU dengan menyorot bagian teks yang paling informatif, sehingga representasi kalimat tidak hanya bergantung pada rangkuman tersembunyi terakhir, tetapi pada token-token yang berkontribusi paling besar terhadap prediksi. Bagian ini mengulas penelitian yang menerapkan BiGRU dan atensi pada tugas deteksi depresi serta membandingkannya dengan arsitektur alternatif untuk menegaskan keunggulan arsitektural dan alasan pemilihannya dalam penelitian ini.

2.3 *Embedding* Kata untuk Representasi Teks Kesehatan Mental

Metode *embedding* kata telah mengalami perubahan mendasar dalam aplikasi kesehatan mental, berkembang dari representasi statis sederhana menuju *embedding* kontekstual yang lebih canggih. Bagian ini mengulas penelitian tentang berbagai pendekatan *embedding*, dengan fokus khusus pada FastText yang sangat cocok untuk teks media sosial berbahasa Inggris (seperti Reddit) yang penuh

dengan slang, variasi ejaan (*typo*), dan kata-kata informal yang sering berada di luar kosakata standar (*Out-of-Vocabulary*).

Tejaswini dkk. (2024) mengimplementasikan arsitektur FastText Convolution Neural network dengan Long Short-Term Memory (FCL), dipublikasikan di ACM Transactions on Asian and Low-Resource Language Information Processing. *Model* tersebut dilatih pada 13.000 sampel dari Reddit dan Twitter dan mencapai akurasi 87 persen. Keunggulan FastText terletak pada pendekatan n-gram karakter yang memecah kata menjadi unit subword, misalnya kata "playing" dipecah menjadi "<pl", "pla", "lay", "ayi", "yin", "ing", "ng>", sedangkan *embedding* akhir merupakan penjumlahan atau rata-rata dari *embedding* subword. Kemampuan membentuk *embedding* untuk kata di luar kosakata pelatihan sangat penting untuk teks media sosial yang penuh salah tulis, slang, dan neologisme. FastText menjadi opsi praktis untuk membangun representasi kata yang stabil tanpa ketergantungan pada *model* transformer domain-spesifik, terutama untuk teks media sosial yang penuh variasi informal.

Berdasarkan kajian tersebut, pemilihan *embedding* untuk klasifikasi teks kesehatan mental perlu disesuaikan dengan kebutuhan komputasi, ketersediaan data domain, serta karakteristik bahasa yang dianalisis. *Embedding* kontekstual seperti BERT sering memberikan performa tinggi, tetapi umumnya menuntut sumber daya komputasi besar dan dukungan pra-latih yang kuat. Di sisi lain, FastText menawarkan alternatif yang lebih ringan namun tetap andal untuk teks media sosial karena memanfaatkan representasi subword yang tahan terhadap variasi bentuk kata dan masalah out-of-vocabulary. FastText sangat cocok untuk teks media sosial

berbahasa Inggris (seperti Reddit) yang penuh dengan slang, variasi ejaan (typo), dan kata-kata informal yang sering berada di luar kosakata standar (Out-of-Vocabulary).

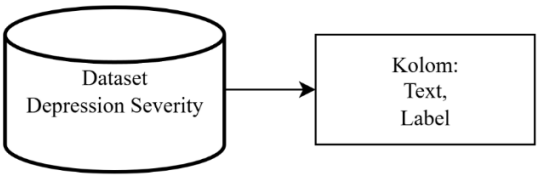
BAB III

DESAIN DAN IMPLEMENTASI SISTEM

3.1 Pengumpulan Data

Penelitian ini menggunakan Depression *Severity* Dataset yang dikembangkan oleh Naseem dkk. (2022) yang dipresentasikan pada The Web Conference 2022. Dataset ini memuat teks unggahan pengguna platform Reddit yang mengekspresikan pengalaman atau perasaan terkait depresi. Setiap entri telah dianotasi dengan label tingkat keparahan (*severity*) yang bersifat *ordinal*, yaitu *minimum*, *mild* (ringan), *moderate* (sedang), dan *severe* (parah).

Proses pelabelan dataset ini dilakukan dengan mengacu pada standar instrumen klinis, seperti *Beck's Depression Inventory* (BDI) dan *Depressive Disorder Annotation Scheme*. Hal ini memastikan bahwa struktur label yang dihasilkan mampu mencerminkan gradasi tingkat keparahan depresi secara klinis. Karakteristik tersebut menjadikan dataset ini sangat relevan untuk analisis *Natural Language Processing* (NLP) pada domain kesehatan mental, karena memungkinkan pemodelan yang jauh lebih sensitif terhadap perbedaan tingkat keparahan dibandingkan dengan pendekatan klasifikasi biner.



Gambar 3.1 Atribut Kolom pada Dataset

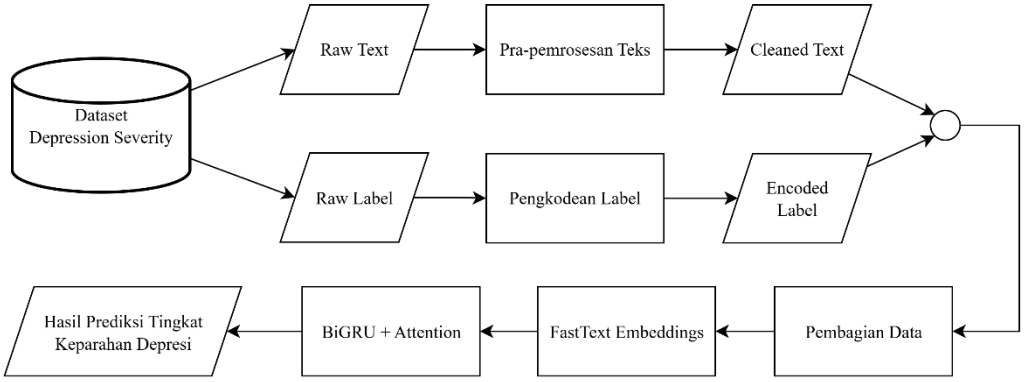
Struktur dataset (seperti yang ditunjukkan pada Gambar 3.1) terdiri dari dua kolom utama: kolom teks yang berisi isi unggahan, dan kolom label yang menunjukkan tingkat keparahan. Untuk memberikan gambaran lebih jelas mengenai karakteristik teks pada masing-masing kelas, Tabel 3.1 menyajikan beberapa contoh kalimat beserta dan *ground truth* labelnya.

Tabel 3.1 Contoh Teks dan Ground Truth

Teks Unggahan	**Label**
Just goes to show: It's not the suit. It's how you wear it. Update 2: Got the job. Thread here.	*minimum*
Nothing He kicked the door 18-36-30 He kicked it again and until he was sure his foot would bruise Calm down	*minimum*
I can’t have fun anymore. I can’t enjoy life anymore. I don’t know what to do. This is hopeless. I had to come home from work because I can’t stop crying.	*mild*
This Crippling Pain Is Getting Stronger. Cant you see I cant do this much longer This Fearless drip, The subconscious Tears. Hope someone Can see my Fear.	*mild*
I knew something was up with me. My thoughts were consuming me. Couldn’t sleep. Stressed. Worried about anything and everything.	*moderate*
Also the headaches. LOADS of headaches all the time. I’m so done. I hate this almost as bad as my brain constantly telling me I’m a POS. Anxiety is fun :)	*moderate*
I don't know... I don't know what to do. I just want out of here. It's too hard. With this house and school work.	*severe*
Idk Do I tell someone? Do I just quit? Do I talk to her about what she did? Please, any advice would be really really helpful to me!	*severe*

3.2 Desain Sistem

Desain sistem dalam penelitian ini dirancang untuk memberikan gambaran komprehensif mengenai alur kerja *model* dalam mengklasifikasikan tingkat keparahan depresi berdasarkan teks dari media sosial Reddit. Sistem beroperasi melalui serangkaian proses yang saling berkesinambungan, diawali dengan penanganan dataset mentah melalui dua alur komputasi yang terpisah. Teks mentah (*raw text*) melalui tahap pra-pemrosesan untuk membuang derau hingga menjadi teks bersih (*cleaned text*), sementara label kategori mentah (*raw label*) dikonversi menjadi representasi numerik *ordinal* melalui tahap pengkodean menjadi *encoded label*. Setelah tahapan awal tersebut selesai, data yang telah bersih dan terkodekan disatukan kembali untuk memasuki tahap pembagian data. Selanjutnya, teks bersih direpresentasikan ke dalam bentuk vektor menggunakan *FastText Embeddings*. Vektor fitur tekstual ini kemudian diproses menggunakan arsitektur pemodelan *Bidirectional Gated Recurrent Unit* dengan mekanisme atensi (BiGRU + *Attention*) untuk melakukan proses klasifikasi. Pada tahap akhir, *model* akan menghasilkan keluaran berupa prediksi tingkat keparahan depresi yang paling sesuai dengan pola dan konteks teks yang dianalisis. Secara keseluruhan, alur kerja desain sistem dalam penelitian ini diilustrasikan pada Gambar 3.2.



Gambar 3.2 Desain Sistem

3.2.1 Pra-pemrosesan Teks

Shukla & Dwivedi (2024) menunjukkan bahwa teks *user-generated* yang bersifat *noisy* dan tidak terstruktur lazim mengandung unsur tidak relevan seperti pengulangan karakter, slang, singkatan, dan akronim, sehingga pra-pemrosesan diperlukan untuk mengubah data mentah menjadi format yang lebih terstruktur sebelum tugas klasifikasi dilakukan. Mereka juga menekankan bahwa kinerja *model* klasifikasi sangat dipengaruhi oleh pilihan pra-pemrosesan, lalu menguji 13 teknik umum misalnya *lowercasing*, *stopword removal*, *stemming*, dan *lemmatization* pada berbagai *model* ML dan DL termasuk BiLSTM dan BERT. Hasilnya menegaskan bahwa sebagian teknik pra-pemrosesan dapat meningkatkan akurasi, tetapi sebagian lain tidak berdampak signifikan, dan efektivitasnya bergantung pada jenis pengklasifikasi bahkan kombinasi teknik tertentu terbukti lebih efektif pada *model* tertentu seperti BiLSTM. Temuan ini mendukung keputusan penelitian ini untuk menerapkan pra-pemrosesan yang menargetkan sumber *noise* yang umum pada teks media sosial misalnya normalisasi huruf, pembersihan token non-informatif, serta normalisasi variasi penulisan, karena pada

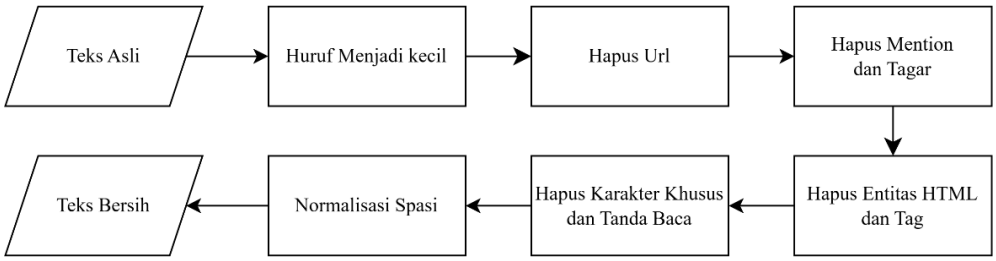
arsitektur sekuensial berbasis RNN termasuk BiGRU dengan atensi kualitas masukan yang lebih konsisten berpotensi membantu *model* menangkap pola linguistik yang relevan secara lebih stabil (Bhuiyan dkk., 2025).

Sejalan dengan landasan tersebut, analisis awal terhadap korpus Reddit yang digunakan menunjukkan keberadaan pola non-standar yang cukup dominan dan berpotensi mengganggu pembelajaran *model* apabila dibiarkan dalam bentuk mentah. Ditemukan URL yang umumnya merujuk ke sumber eksternal dan cenderung tidak menambah informasi semantik inti, misalnya tautan donasi seperti `[Team Thomas](gf.me/u/jyy4qm)` atau tautan layanan dukungan seperti `[https://thehaven.support](https://thehaven.support/)`, termasuk placeholder seperti `<url>`. Selain itu terdapat entitas HTML sebagai artefak pemformatan web seperti ` ` dan `&#x200B;` yang dapat memunculkan token tidak bermakna. Pola khas Reddit juga muncul dalam bentuk mention subreddit seperti `r/depression` atau `r/relationships` yang tidak selalu merepresentasikan kondisi psikologis penulis dan dapat menambah kosakata spesifik platform secara tidak perlu.

Di sisi lain, ekspresi emosional sering muncul sebagai karakter khusus berulang seperti `???`, `***`, `.....`, dan `!!!!`, serta pengulangan karakter untuk penekanan misalnya “wayyyyyyy”, sehingga perlu dinormalisasi agar variasi ejaan tidak memecah representasi kata dan tidak mendominasi distribusi token. Korpus juga memuat angka dalam berbagai format seperti `273`, `$2250,`, `06`, dan `$17` yang dalam konteks tertentu dapat informatif namun sering menjadi *noise* jika tidak dikendalikan. Variasi bentuk kontraksi dan kepemilikan seperti `that'd`, `Tyson's`,

atau `dm's` mencerminkan gaya informal yang menuntut konsistensi tokenisasi, sementara emotikon seperti `:-(`, `:/`, dan `D:` membawa muatan afektif yang perlu diperlakukan secara konsisten dalam pipa pemrosesan.

Terakhir, ditemui spasi berlebih yang menunjukkan ketidakteraturan format, misalnya pada frasa “asked to post here” dengan spasi ganda, sehingga perlu dirapikan untuk menjaga keseragaman masukan *model*. Temuan-temuan ini menegaskan perlunya pra-pemrosesan yang tidak sekadar “membersihkan” teks, tetapi secara khusus menargetkan sumber *noise* khas media sosial sambil mempertahankan isyarat semantik dan emosional yang relevan untuk klasifikasi tingkat keparahan depresi. Alur lengkap tahapan pra-pemrosesan yang digunakan ditunjukkan pada Gambar 3.3.



Gambar 3.3 Alur Diagram Pra-Pemrosesan

1. Huruf Menjadi Kecil

Lowercasing mengubah seluruh huruf menjadi huruf kecil untuk menghilangkan sensitivitas terhadap kapitalisasi dan meningkatkan konsistensi representasi teks, misalnya kata "DEPRESI", "Depresi", dan "depresi" harus diperlakukan identik karena maknanya sama. Konversi ini mengurangi jumlah dimensi kosakata, menstabilkan satuan kata tokenisasi, dan membantu *model* fokus pada pola linguistik yang relevan daripada variasi format tulisan. Karena pemilihan

dan urutan komponen pra-pemrosesan memengaruhi hasil akhir, *Lowercasing* umumnya diterapkan secara konsisten sebelum vektorisasi. Penelitian terdahulu menunjukkan bahwa kombinasi pra-pemrosesan yang mencakup *Lowercasing* cenderung meningkatkan akurasi pada tugas klasifikasi teks tradisional (Uysal & Gunal, 2014). Contoh implementasi *case folding* ditunjukkan pada Tabel 3.2.

Tabel 3.2 Contoh Lowercasing

Teks awal	**Hasil *Lowercasing***
"...I go rest and so ..TRIGGER AHEAD IF YOUI'RE A HYPOCONDRIAC LIKE ME..."	*"...i go rest and so ..trigger ahead if youi're a hypocondriac like me..."*

2. Hapus Url

Penghapusan URL pada penelitian ini berarti menghilangkan seluruh satuan kata (token) tautan (mis. https://...) karena URL tidak memuat sinyal sentimen yang dapat diproses langsung oleh *model* teks, cenderung menambah entri kosakata yang jarang muncul, sehingga membuat representasi teks menjadi terlalu jarang terisi dan mengganggu kemampuan *model* dalam mengenali pola secara konsisten. Langkah ini mencegah *model* menebak isi dari sumber eksternal yang tidak dianalisis sehingga fokus tetap pada konten linguistik pernyataan pengguna. Pada data media sosial, URL sering digunakan untuk membagikan konten eksternal yang tidak dapat diinterpretasikan hanya dari *string*-nya, oleh karena itu penghapusan dilakukan secara otomatis menggunakan *regular expression* yang mendeteksi pola http, https, www, atau domain standar. Studi pra-pemrosesan pada analisis sentimen *mikroblog* merekomendasikan membuang URL sepenuhnya dengan alasan tersebut, dan riset domain kesehatan mental di Twitter juga menerapkan pembersihan URL karena

biasanya tidak menyumbang informasi tekstual yang relevan (Palomino & Aider, 2022). Contoh penghapusan URL dapat dilihat pada Tabel 3.3.

Tabel 3.3 Contoh Hapus URL

Teks awal	Hasil Hapus URL
"...Please do. [Team Thomas](gf.me/u/jyy4qm)"	*"...Please do. [Team Thomas]()"*

3. Hapus Mention

Penyebutan nama pengguna atau *mention* (@username) dihapus karena merujuk pada individu spesifik yang tidak membawa nilai semantik untuk keperluan klasifikasi. Elemen ini dianggap sebagai metadata sosial yang dapat memperluas ruang kosakata secara tidak terkendali. Hal ini terjadi karena nama akun bersifat unik dan tidak baku, sehingga berisiko meningkatkan tingkat keterjarangan fitur dan membuat *model* belajar pola yang tidak konsisten. Oleh karena itu, langkah ini sejalan dengan rekomendasi pra-pemrosesan untuk membuang fitur khas media sosial agar representasi data lebih bersih dan kemampuan generalisasi *model* menjadi lebih baik (Siino dkk., 2024).

Sementara itu, penanganan tagar memiliki dua pendekatan yang umum digunakan. Beberapa metode memilih untuk hanya menghapus simbol "#" dan mempertahankan kata kuncinya karena sering mengandung informasi emosional yang relevan. Namun, penelitian ini memilih pendekatan kedua, yaitu membuang seluruh satuan kata bertanda pagar secara utuh (misalnya #sad) guna menghindari kebocoran label dan bias topik. Tagar kesehatan mental kerap dipakai sekadar sebagai penanda kampanye, sehingga penghapusan ini akan mendorong *model* untuk benar-benar belajar dari isi kalimat alih-alih menebak kelas dari sinyal

luarnya saja. Keputusan ini didukung oleh temuan bahwa tagar lebih sering digunakan untuk menarik perhatian sehingga tidak selalu mencerminkan kondisi emosional asli penulisnya (Beierle dkk., 2023). Contoh hasil dari tahap penghapusan *mention* dan tagar ini dapat dilihat selengkapnya pada Tabel 3.4.

Tabel 3.4 Contoh Hapus Mention & Tagar

Teks awal	Hasil Hapus Mention & Tagar
"...My instagram is: @thewellnesssocietyorg My twitter is: @thewellnesssoc It's currently #1 on Amazon's..."	*"...my instagram is: my twitter is: it's currently on amazon's..."*

4. Hapus entitas & tagar HTML

Penghapusan entitas dan tagar HTML pada penelitian ini bertujuan menyingkirkan *boilerplate* web seperti *markup*, navigasi, dan iklan yang tidak mencerminkan isi linguistik sehingga dapat mengganggu pembentukan fitur dan menurunkan validitas evaluasi. Langkah ini digunakan karena pemisahan konten utama dari *markup* terbukti meningkatkan kualitas representasi teks dan kinerja hilir, misalnya dengan menghapus elemen HTML sebelum pemodelan klasifikasi atau *retrieval* (pengambilan kembali informasi) (Kohlschütter dkk., 2010). Contoh hasil pembersihan HTML ditunjukkan pada Tabel 3.5.

Tabel 3.5 Contoh Hapus HTML

Teks awal	Hasil Hapus HTML
"...Here is also a link to our softball teams website: <url>..."	*"...here is also a link to our softball teams website: ..."*

5. Hapus Karakter Khusus dan Tanda Baca

Penghapusan karakter khusus dalam penelitian ini mencakup tanda baca berlebihan (mis. !!! atau ???), simbol matematika, serta emotikon, dengan tujuan

menurunkan kompleksitas teks, menstabilkan satuan kata tokenisasi, dan mengurangi keterjarangan fitur yang timbul dari sinyal non-leksikal yang tidak konsisten lintas platform, meskipun emotikon dapat membawa informasi emosional, representasi mereka dalam *encoding* teks bervariasi sehingga penelitian ini memfokuskan pada informasi linguistik murni dari kata-kata yang digunakan, sehingga hanya karakter alfanumerik dan spasi yang dipertahankan.

Jika emotikon diputuskan untuk dipertahankan, penanganan khusus dilakukan terlebih dahulu lalu pembersihan tanda baca dilakukan setelahnya. Langkah ini dilakukan dalam *pipeline* pra-pemrosesan untuk analisis sentimen dan klasifikasi teks karena pembuangan tanda baca dan simbol terbukti meningkatkan kebersihan representasi sebelum vektorisasi (Palomino & Aider, 2022). Contoh pembersihan karakter khusus ditampilkan pada Tabel 3.6.

Tabel 3.6 Contoh Hapus Karakter Khusus

Teks awal	**Hasil Hapus Karakter Khusus**
"...way before, suggeted I go rest and so ..TRIGGER AHEAD IF YOUI'RE A HYPOCONDRIAC LIKE ME: i decide..."	*"...way before suggeted i go rest and so trigger ahead if youire a hypocondriac like me i decide..."*

6. Normalisasi Spasi

Trim *Whitespace* pada penelitian ini berarti menormalkan spasi dengan menghapus spasi ganda, menghilangkan spasi *leading* dan *trailing*, serta membuang jeda tak perlu yang muncul akibat pembersihan sebelumnya, normalisasi ini menstabilkan satuan kata tokenisasi, mencegah fragmentasi kosakata akibat variasi spasi, menghindari kemunculan satuan kata (token) kosong atau berlebih, dan menjaga konsistensi representasi teks sebelum proses vektorisasi yang digunakan

dalam *pipeline* seperti FastText. Langkah sederhana ini termasuk praktik pra-pemrosesan yang direkomendasikan dalam survei metodologis modern karena kombinasi komponen pra-pemrosesan yang tepat berpengaruh pada hasil klasifikasi (Chai, 2023). Contoh implementasi trim *whitespace* ditunjukkan pada Tabel 3.7.

Tabel 3.7 Contoh Trim Whitespace

Teks awal	**Hasil Trim *Whitespace***
" It cleared up and I was okay but..."	*"it cleared up and i was okay but..."*

3.2.2 Pengkodean Label

Label target yang semula berbentuk kategori, yaitu *minimum*, *mild*, *moderate*, dan *severe*, dikonversi ke dalam format numerik berupa bilangan bulat berurutan, secara spesifik menggunakan indeks 0, 1, 2, dan 3. Pengkodean ini dipilih karena kelas keluaran memiliki urutan tingkat keparahan yang jelas, sehingga informasi peringkat hierarkis antar kelas dapat dipertahankan selama proses pembelajaran *model*. Dalam pemodelan data *ordinal,* indeks 0 hingga 3 tersebut tidak dimaknai sebagai nilai kuantitatif dengan jarak absolut yang setara, melainkan sebagai indeks urutan yang merepresentasikan tingkatan dari yang terendah hingga tertinggi. Pendekatan ini didukung oleh McCullagh (1980) yang menegaskan bahwa pemodelan *ordinal* memanfaatkan sifat keterurutan kategori tanpa harus mengasumsikan kardinalitas atau jarak yang sama antar kategori. Sejalan dengan hal itu, Bürkner & Vuorre (2019) menjelaskan bahwa data *ordinal* memiliki urutan alami, tetapi tidak bersifat metrik karena jarak psikologis atau substantif antar level sangat mungkin bervariasi.

Dalam literatur *machine learning*, klasifikasi *ordinal* dipahami sebagai tugas pemodelan dengan label yang memiliki urutan alami, sehingga penggunaan

label berurutan jauh lebih tepat dibandingkan perlakuan nominal yang mengabaikan hubungan hierarkis tersebut (Gutierrez dkk., 2016; Vargas dkk., 2020). Pemilihan representasi angka 0, 1, 2, dan 3 sangat krusial secara komputasi karena selaras dengan prinsip pengindeksan berbasis nol pada arsitektur *deep learning*. Ekosistem komputasi jaringan saraf mensyaratkan label kelas target dienkode mulai dari angka 0 hingga jumlah kelas dikurangi 1 guna mencegah pemborosan dimensi memori dan anomali probabilitas pada lapisan *output*. Pengkodean representasi ini dinilai tepat secara metodologis karena menjembatani efisiensi implementasi komputasi *deep learning* sekaligus tetap konsisten dengan karakteristik konseptual variabel keparahan (Bérchez-Moreno dkk., 2024).

Dengan menggunakan label berurutan 0 hingga 3, sistem tidak hanya dituntut untuk memprediksi kelas secara akurat, tetapi juga difasilitasi untuk meminimalkan jarak kesalahan ketika *model* melakukan prediksi yang keliru. Sebagai contoh penerapan metrik kesalahannya: memprediksi label *mild* (indeks 1) untuk data yang secara aktual berada pada kelas *moderate* (indeks 2) akan menghasilkan selisih numerik yang kecil, sehingga penaltinya jauh lebih dapat diterima dibandingkan memprediksinya sebagai kelas *minimum* (indeks 0) yang memiliki rentang selisih lebih jauh. Hal ini dimungkinkan karena representasi angka 0, 1, 2, 3 secara langsung memfasilitasi algoritma maupun metrik evaluasi untuk mempertimbangkan seberapa jauh *model* meleset dari urutan tingkat keparahan depresi yang sebenarnya. Dengan demikian, pengkodean ke dalam bilangan bulat berurutan ini memberikan representasi matematis yang paling ideal untuk skenario penelitian ini. Adapun hasil akhir dari proses transformasi label

kategori teks menjadi bentuk integer tersebut ditampilkan selengkapnya pada Tabel 3.8.

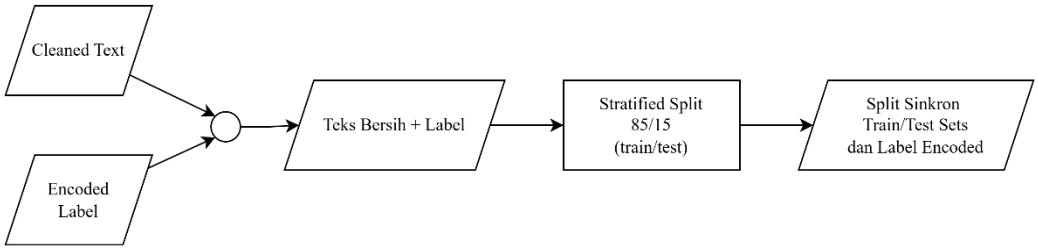
Tabel 3.8 Hasil Transformasi Tabel

Label Kategori	**Label *Integer***
minimum	0
mild	1
moderate	2
severe	3

3.2.3 Pembagian Data

Setelah melalui tahap pra-pemrosesan teks dan pengkodean label, keluaran berupa Teks Bersih (*Cleaned Text*) dan label numerik (*Encoded Label*) disatukan terlebih dahulu menjadi satu kesatuan (Teks Bersih + *Label*). Langkah penyatuan ini krusial untuk memastikan bahwa setiap baris teks tetap berpasangan secara tepat dengan label tingkat keparahannya sebelum dilakukan partisi data.

Berbeda dengan pembagian statis sederhana, penelitian ini menerapkan skema *Stratified k-fold cross validation* untuk memastikan evaluasi *model* yang lebih *robust* dan mengurangi bias yang mungkin timbul akibat pemilihan data latih yang spesifik. Alur skema pembagian data yang digunakan ditunjukkan pada Gambar 3.4.



Gambar 3.4 Skema Pembagian Data.

Berdasarkan Gambar 3.4, dataset awal terlebih dahulu dibagi menjadi dua bagian utama menggunakan metode pembagian berstrata (*stratified split*), yaitu data pengujian (*test set*) sebesar 15% dan data pengembangan (*development set*) sebesar 85%. Pemilihan proporsi ini didasarkan pada pertimbangan metodologis bahwa *test set* harus disediakan sebagai evaluasi akhir yang independen tanpa mengurangi kecukupan data untuk kebutuhan pengembangan *model*. Literatur pendukung menegaskan pentingnya penggunaan *blind test set* yang dipisahkan secara total dari proses pelatihan untuk menjamin estimasi performa *model* yang lebih andal dan konsisten (Allgaier & Pryss, 2024; Xu & Goodacre, 2018). Dengan demikian, penggunaan 85% data sebagai *development set* dipandang rasional karena mampu memaksimalkan data untuk pelatihan dan validasi silang sekaligus tetap menjaga kemandirian *hold-out test set*.

Data pengujian tersebut sejak awal disisihkan sebagai *hold-out set* dan tidak pernah dilibatkan dalam proses pelatihan maupun validasi silang agar dapat berfungsi sebagai simulasi evaluasi akhir pada data dunia nyata. Selanjutnya, pada tahap *Cross validation*, data pengembangan tersebut dibagi menjadi K bagian atau lipatan (*folds*) yang sama besar dengan nilai K=5 yang telah ditetapkan. Penerapan teknik berstrata ini menjamin bahwa setiap lipatan data memiliki proporsi distribusi

kelas tingkat keparahan depresi yang identik dengan dataset aslinya. Proses pelatihan kemudian dilakukan secara berulang sebanyak lima kali, di mana pada setiap iterasinya terdapat empat bagian data yang digunakan untuk memperbarui bobot *model* dan satu bagian sisanya digunakan sebagai data validasi guna menghindari masalah penghafalan data atau *overfitting*.

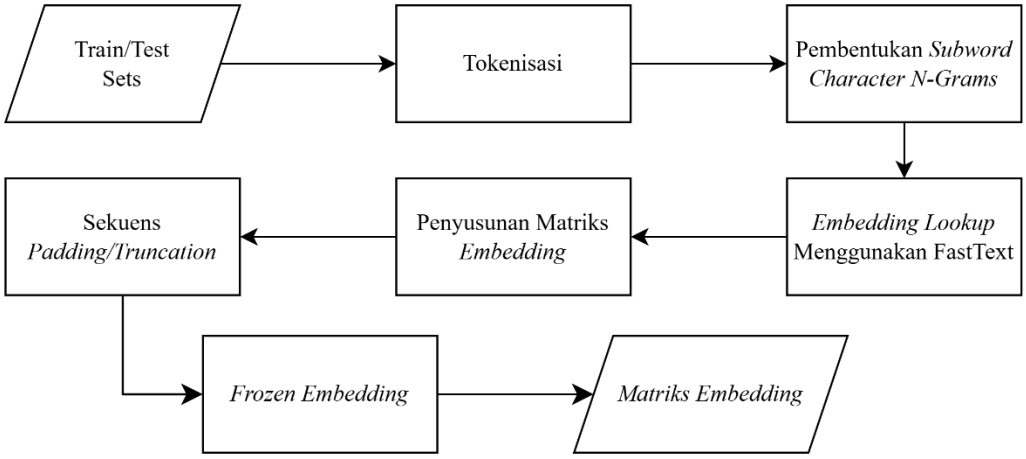
Penggunaan *K-Fold Cross validation* merupakan standar validasi yang lebih ketat dibandingkan pembagian tunggal, sebagaimana disarankan oleh (Wong & Yeh, 2020) untuk mendapatkan estimasi akurasi yang lebih andal. Secara matematis, performa akhir *model* divalidasi dengan merata-ratakan hasil evaluasi dari kelima iterasi tersebut. Pendekatan ini berfungsi untuk meminimalkan risiko variasi hasil yang disebabkan oleh faktor kebetulan saat pengambilan sampel secara acak, sehingga metrik evaluasi yang dihasilkan lebih merepresentasikan kemampuan generalisasi *model* yang sebenarnya sebelum dievaluasi secara final pada data pengujian.

3.2.4 Representasi Teks

Setelah tahap pembagian data, data train yang telah dibersihkan perlu ditransformasi menjadi representasi numerik yang dapat diproses oleh *model neural network*. Pada penelitian ini, representasi teks dilakukan dengan mengonversi setiap dokumen menjadi urutan vektor padat menggunakan FastText. Untuk mendapatkan representasi yang kaya akan memori semantik, digunakan *model pretrained* cc.en.300.bin dari *Facebook AI* yang telah dilatih pada 176 juta kata *Common Crawl*. Representasi berbasis *embedding* ini dipilih karena arsitektur BiGRU

dengan mekanisme atensi memodelkan data sebagai sekuens, sehingga informasi urutan dan konteks antar token dapat dipelajari secara *end-to-end*.

Dengan demikian, pembobotan berbasis frekuensi seperti TF-IDF tidak digunakan karena penekanan informasi penting dipelajari oleh *model* melalui dinamika sekuens pada BiGRU dan pembobotan adaptif pada atensi. FastText juga dipilih karena kemampuannya menangani token di luar kosakata (*Out-of-Vocabulary*/OOV) dan variasi morfologi (seperti kesalahan ketik atau *typo*) melalui representasi *subword* berbasis *character n-grams* (Bojanowski dkk., 2017). Alur lengkap proses representasi teks menggunakan FastText hingga menjadi masukan bagi *model* BiGRU dengan mekanisme atensi ditunjukkan pada Gambar 3.5.



Gambar 3.5 Alur Representasi Teks Menggunakan FastText

1. Satuan kata tokenisasi

Tahap awal adalah memecah teks akhir hasil pra-pemrosesan menjadi urutan token yang menyusun data latih. Pada penelitian ini digunakan *whitespace tokenization* yakni pemisahan berdasarkan spasi. Tokenisasi diperlukan karena kamus matriks FastText akan dibentuk pada level token, lalu urutan token tersebut

dipertahankan sebagai sekuens masukan bagi *model* sekuensial. Secara formal, proses ini dinyatakan dalam Rumus 3.1:

$$T = tokenize(t_{final}) = [w_1, w_2, \dots, w_n] \quad (3.1)$$

Keterangan:
t_{final} = teks akhir setelah pra-pemrosesan
$tokenize(\cdot)$ = fungsi tokenisasi berbasis spasi
T = sekuens token hasil tokenisasi
w_t = token pada posisi ke (t)
T pada notasi ($[w_1, \dots, w_T]$) menyatakan panjang sekuens token

Contoh hasil dari tahap satuan kata tokenisasi ditunjukkan pada Tabel 3.9.

Tabel 3.9 Contoh Satuan Kata Tokenisasi

Teks awal	**Hasil satuan kata tokenisasi**
"i feel drepressed"	*["i", "feel", "depressed"]*

2. Pembentukan Subword Character N-grams

Sebelum dilakukan *embedding lookup*, FastText terlebih dahulu memecah setiap token menjadi himpunan *subword* berupa *character N-grams*. Mekanisme ini merupakan keunggulan utama FastText dibandingkan metode *embedding* konvensional, karena memungkinkan representasi token yang tidak ada dalam kosakata (*out-of-vocabulary*/OOV) melalui komposisi *subword*-nya.

a) Penambahan *Boundary Markers*

Setiap token diberi penanda batas berupa karakter `<` di awal dan `>` di akhir untuk membedakan posisi karakter dalam kata. Proses ini dinyatakan sebagaimana ditampilkan pada Rumus 3.2 berikut:

$$w' = < + w + > \quad (3.2)$$

Keterangan:
w = token asli
w' = token dengan *Boundary Markers*
<,> = karakter penanda batas awal dan akhir

b) Ekstraksi Character N-grams

Dari token yang sudah diberi *Boundary Markers*, diekstraksi seluruh potongan kata dengan panjang n dalam rentang yang ditentukan ($n_{min} = 3$ hingga $n_{max} = 6$). Himpunan *subword* untuk token w dinyatakan sebagaimana ditampilkan pada Rumus 3.3 berikut:

$$G(w) = \{g : g \text{ adalah potongan kata dari } w' \mid n_{min} \leq |g| \leq n_{max}\} \quad (3.3)$$

Keterangan:
$G(w)$ = himpunan *subword* (*character N-grams*) dari token *w*
g = satu *subword* anggota $G(w)$
$|g|$ = panjang *subword* g
n_{min}, n_{max} = batas *minimum* dan maksimum panjang n-gram (umumnya 3–6)

Tabel 3.10 berikut merupakan contoh hasil ekstraksi n-gram untuk token `depresi`:

Tabel 3.10 Contoh Pembentukan *Character N-grams*

Panjang N-gram	**Hasil Ekstraksi**
3-gram	<de, dep, epr, pre, res, ess, sse, sed, ed>
4-gram	<dep, depr, epre, pres, ress, esse, ssed, sed>
5-gram	<depr, depre, epres, press, resse, essed, ssed>
6-gram	<depre, depres, epress, presse, ressed, essed>

3. Embedding Lookup Menggunakan FastText

Setelah diperoleh sekuens token, setiap token w_t diubah menjadi vektor *embedding* berdimensi d menggunakan fungsi FastText. FastText merepresentasikan token tidak hanya berdasarkan kata utuh, tetapi juga memanfaatkan *subword* berupa *character n-grams*, sehingga token yang tidak tersedia sebagai kata utuh tetap dapat memperoleh representasi melalui komponen *subword*-nya. Pemetaan token menjadi vektor *embedding* dinyatakan sebagaimana ditampilkan pada Rumus 3.4 berikut:

$$x_t = f_{FT}(w_t), \quad x_t \in R^d \tag{3.4}$$

Keterangan:
$f_{\mathrm{FT}}(\cdot)$ = fungsi *embedding* FastText
$\boldsymbol{x_t}$ = vektor *embedding* token ke t
w_{t_d} = token pada posisi ke t
$\mathbb{R}^d$ = ruang vektor berdimensi d dengan elemen bilangan riil
d = dimensi *embedding* (300d)

Secara konseptual, representasi FastText dapat dipandang sebagai kombinasi kontribusi vektor kata utuh dan vektor *subword n-grams* yang menyusun token. Formulasi konseptual tersebut ditunjukkan sebagaimana ditampilkan pada Rumus 3.5 berikut:

$$f_{FT}(w) \approx z(w) + \sum_{g \in G(w)} z(g) \tag{3.5}$$

Keterangan:
w = sebuah token
$z(w)$ = vektor kata utuh jika tersedia dalam kosakata
$G(w)$ = himpunan *subword character n grams* dari token *w*
g = satu *subword* anggota $G(w)$
$z(g)$ = vektor *embedding* untuk *subword g*
$\sum_{g \in G(w)}$ = penjumlahan seluruh vektor *subword* dalam $G(w)$
Simbol ≈ menyatakan bentuk konseptual, karena implementasi dapat bervariasi tergantung konfigurasi *model*

4. Penyusunan Matriks Embedding

Pada tahap ini, dokumen aktual dikonversi menjadi urutan indeks numerik (*Text to Sequence*) berdasarkan kamus yang telah dibentuk. Contoh hasil pemetaan satuan kata (token) ke ID numerik ditampilkan pada Tabel 3.11.

Tabel 3.11 Contoh Pemetaan Kosakata

Panjang N-gram	**Hasil Pemetaan kosakata**
["i", "feel", "depressed"]	[125, 487, 1203]

Secara matematis, sebelum dilakukan *padding*, sekuens indeks tersebut merepresentasikan matriks *embedding* yang mempertahankan urutan token. Penyusunan konseptual matriks ini dinyatakan sebagaimana ditampilkan pada Rumus 3.6 berikut, di mana baris matriks masih mengikuti panjang asli dokumen (T):

$$X = [x_1, x_2, \dots, x_T]^\top \in R^{T \times d} \tag{3.6}$$

Keterangan:
X = matriks *embedding* untuk satu dokumen
x_t = *embedding* token pada posisi ke (t)
$(\cdot)^\top$ = transpose sehingga vektor disusun menjadi baris baris matriks
$R^{T \times d}$ = matriks berukuran (T) baris dan (d) kolom

5. Sekuens Padding/Truncation

Panjang dokumen pada dataset sangat bervariasi sehingga diperlukan penyeragaman dimensi masukan agar seluruh sampel dapat diproses secara bersamaan dan memiliki bentuk tensor yang konsisten. Statistik panjang token per label menunjukkan pusat sebaran yang relatif stabil dengan nilai tengah sekitar 80 hingga 84 token dan rata-rata sekitar 84,2 hingga 91,6 token. Meskipun demikian, distribusi data tersebut tetap memiliki ekor sebaran yang panjang hingga mencapai 310 token pada sampel tertentu. Variasi inilah yang menuntut penetapan panjang sekuens maksimum dengan mempertimbangkan titik keseimbangan antara penyimpanan informasi dan efisiensi waktu komputasi *model*.

Pemilihan nilai batas panjang yang terlalu kecil akan meningkatkan pemotongan teks secara paksa dan berpotensi menghilangkan konteks penting dari sebuah kalimat. Sebaliknya, penentuan batas yang terlalu besar akan menambah porsi ruang kosong pada mayoritas teks yang lebih pendek sehingga memicu

lonjakan beban memori tanpa peningkatan kinerja yang memadai. Terkait hal ini, pengujian pada *model* berbasis GRU memperlihatkan adanya fenomena hasil komputasi yang makin menurun ketika panjang teks ditingkatkan secara berlebihan (Huang dkk., 2023). Hal tersebut membuktikan bahwa memperpanjang teks masukan melewati batas kebutuhan utamanya tidak akan memberikan peningkatan performa yang sepadan dengan beban komputasinya. Oleh karena itu, penentuan panjang sekuens harus dilakukan pada titik yang paling proporsional agar informasi intinya tetap utuh tanpa memberatkan sistem secara sia-sia.

Untuk menentukan batas yang mencakup mayoritas sampel secara konservatif, penelitian ini menetapkan t_{max} menggunakan heuristik berbasis sebaran, yaitu mengambil nilai maksimum dari $\mu + 3\sigma$ pada setiap label. Keandalan metode interval ini didukung oleh literatur yang menunjukkan bahwa formulasi batas tersebut dapat mencakup hampir seluruh variasi data yang valid tanpa terdistorsi oleh kemunculan data ekstrem (Pukelsheim, 1994). Pendekatan tersebut dipilih guna mendorong cakupan informasi yang luas tanpa harus mengikuti nilai maksimum yang bersifat anomali.

Perhitungan $\mu + 3\sigma$ berturut turut adalah *minimum* 84,2 + 3(31,1) = 177,5 *mild* 91,6 + 3(36,0) = 199,6 *moderate* 88,4 + 3(32,1) = 184,7 dan *severe* 88,2 + 3(35,5) = 194,7. Dengan mempertimbangkan hasil tersebut, ditetapkan $t_{max} = 200$ token sebagai batas global melalui pembulatan dari angka tertinggi yang ditemukan. Penetapan ini sejalan dengan studi yang membandingkan berbagai panjang sekuens dan menyimpulkan bahwa pemendekan sekuens yang terlalu agresif dapat merugikan kinerja *model*, sedangkan pemanjangan yang berlebih hanya memberi

peningkatan kecil dibanding biaya komputasi yang dibutuhkan (Dwarampudi & Reddy, 2019).

Setelah t_{max} ditetapkan, seluruh sekuens dengan panjang lebih pendek dari t_{max} ditambahkan token *padding* hingga mencapai panjang yang sama, sedangkan sekuens yang lebih panjang dipotong pada bagian akhir agar dimensi masukan seragam. Proses penambahan token *padding* ditunjukkan sebagaimana ditampilkan pada Rumus 3.7 berikut:

$$[w_1, \dots, w_T] \rightarrow [w_1, \dots, w_T, \langle PAD \rangle, \dots, \langle PAD \rangle] \tag{3.7}$$

Keterangan:
w_t = token pada posisi ke t dalam sekuens asli
T = panjang sekuens asli sebelum penyeragaman
$\langle PAD \rangle$ = token *padding* yang ditambahkan hingga panjang sekuens menjadi t_{max} atau L
$\rightarrow$ = transformasi sekuens dari bentuk asli menjadi bentuk terpad

Embedding untuk token *padding* umumnya ditetapkan sebagai vektor nol agar tidak menambahkan informasi semantik ke sekuens. Penetapan tersebut dinyatakan sebagaimana ditampilkan pada Rumus 3.8 berikut:

$$f_{FT}(\langle PAD \rangle) = 0 \tag{3.8}$$

Keterangan:
$f_{FT}(\cdot)$ = fungsi *embedding* FastText
0 = vektor nol berdimensi (d)

Selain *padding*, diterapkan *Masking* agar posisi *padding* tidak memengaruhi pemodelan sekuens maupun perhitungan atensi, mengingat kajian terdahulu menunjukkan bahwa strategi *padding* dapat memengaruhi performa *model* sekuensial. *Masking* dinyatakan sebagai vektor biner yang memberi nilai 1 untuk token valid dan 0 untuk token *padding* sebagaimana ditampilkan pada Rumus 3.9 berikut:

$$m_t = \begin{cases} 1 \\ 0 \end{cases} \quad (3.9)$$

Keterangan:
m_t = nilai *mask* pada posisi ke t
1 = menunjukkan token valid
0 = menunjukkan token *padding*
t = indeks posisi token dalam sekuens

Contoh hasil *Padding* ditampilkan pada Tabel 3.12.

Tabel 3.12 Contoh Sekuens *Padding*

Teks awal	**Hasil sekuens Padding**
[125, 487, 1203]	*[125, 487, 1203, 0, 0, 0, ..., 0]*

6. Frozen Embedding

Pada penelitian ini, *embedding* FastText digunakan sebagai *frozen embedding* sehingga bobot *embedding* tidak diperbarui selama pelatihan. Secara operasional, *embedding layer* diperlakukan sebagai parameter tetap, sehingga pembelajaran difokuskan pada parameter BiGRU dan atensi. Strategi ini dipilih untuk menjaga stabilitas representasi semantik awal dari FastText serta mengurangi risiko *overfitting*, terutama ketika ukuran data pelatihan tidak sangat besar dan ketika diinginkan pemanfaatan pengetahuan umum yang telah dipelajari FastText dari korpus luas.

7. Pencarian Pada Tabel Embedding (Embedding Lookup)

Keluaran akhir tahap representasi teks adalah matriks *embedding* berdimensi tetap yang menjadi input berurutan untuk BiGRU. Berbeda dengan konseptual pada Rumus 3.6 yang masih mengikuti panjang asli dokumen (T), setelah proses *padding*, satu dokumen utuh direpresentasikan sebagai matriks berukuran $L \times d$, sebagaimana ditampilkan pada Rumus 3.10:

$$X \in \mathbb{R}^{L \times d} \tag{3.10}$$

Keterangan:
X = matriks *embedding* dokumen setelah *padding*
L = panjang sekuens tetap
d = dimensi *embedding*

Matriks keluaran final inilah yang kemudian digunakan secara riil sebagai masukan sekuens ke lapisan BiGRU dengan mekanisme *attention* menghitung bobot kepentingan tiap posisi token untuk membentuk representasi dokumen yang lebih informatif bagi proses klasifikasi.

3.2.5 BiGRU dan Atensi

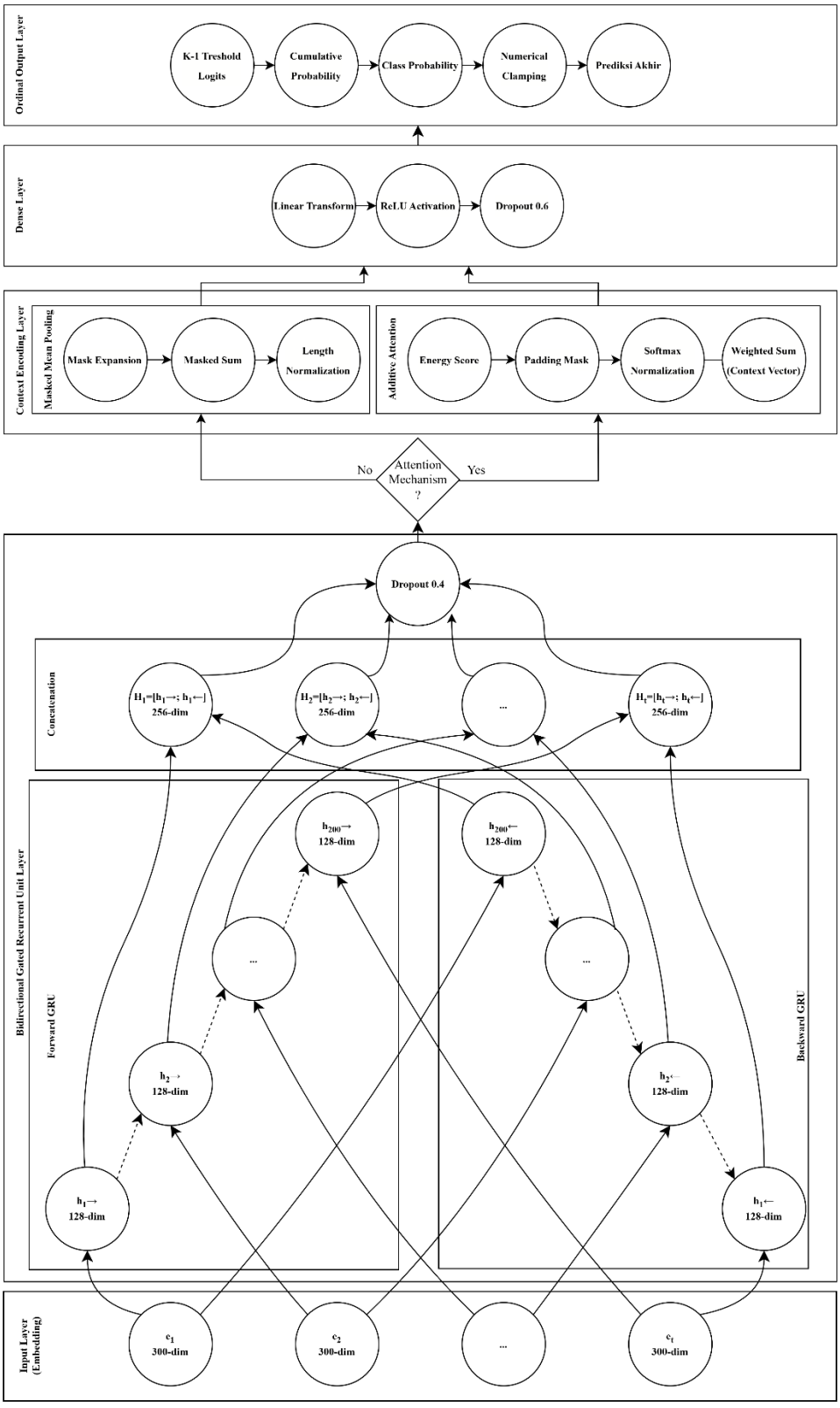
Setelah tahap ekstraksi fitur melalui FastText selesai dilakukan, representasi vektor yang dihasilkan kemudian diumpankan ke dalam *model Bidirectional Gated Recurrent Unit* (BiGRU) yang diintegrasikan dengan mekanisme atensi untuk melakukan klasifikasi tingkat keparahan depresi. Pemilihan BiGRU sebagai arsitektur utama didasarkan pada karakteristik data teks yang bersifat sekuensial, di mana hubungan antar kata memiliki dependensi temporal yang kuat. BiGRU merupakan pengembangan dari *Recurrent Neural Network* (RNN) yang memiliki kemampuan khusus untuk memproses informasi dari dua arah secara simultan, yakni arah maju (*forward*) dan arah mundur (*backwards*). Kemampuan dua arah ini sangat krusial dalam pemrosesan bahasa alami karena makna sebuah kata sering kali tidak hanya ditentukan oleh kata-kata sebelumnya, tetapi juga oleh konteks yang muncul setelahnya (Cho dkk., 2014). Dengan menangkap informasi dari kedua sisi, *model* dapat membentuk representasi dokumen yang jauh lebih kaya dan informatif dibandingkan arsitektur satu arah.

Secara teknis, keunggulan BiGRU terletak pada mekanisme gerbang (*gating mechanism*) yang terdiri dari *update gate* dan *reset gate*. Mekanisme ini memungkinkan *model* untuk secara efisien dan adaptif menentukan informasi mana yang perlu dipertahankan dan informasi mana yang harus diabaikan dari *timestep* sebelumnya. Hal ini menjadi solusi fundamental atas permasalahan gradien lenyap (*vanishing gradient*) yang sering menghambat RNN tradisional dalam mempelajari dependensi jangka panjang pada teks yang panjang (Chung dkk., 2014). Dalam konteks deteksi kondisi kesehatan mental, kemampuan untuk mempertahankan memori jangka panjang sangat penting untuk menangkap pola emosi atau sentimen yang tersebar di sepanjang kalimat (Graves & Schmidhuber, 2005).

Lebih lanjut, integrasi mekanisme atensi (*Attention Mechanism*) dalam penelitian ini bertujuan untuk mengatasi keterbatasan BiGRU dalam meringkas sekuens yang panjang menjadi satu vektor tunggal. Mekanisme atensi bekerja dengan cara memberikan bobot relevansi yang berbeda secara otomatis pada setiap *timestep*. Dalam tugas klasifikasi depresi, tidak semua kata dalam sebuah dokumen memiliki bobot informasi yang sama. Kata-kata bermuatan klinis atau emosional seperti "*hopeless*", "*worthless*", atau "*suicidal thoughts*" memerlukan atensi yang lebih besar dibandingkan kata hubung biasa. Penambahan lapisan ini memungkinkan *model* untuk "fokus" pada bagian teks yang paling menentukan hasil klasifikasi, sehingga meningkatkan performa akurasi sekaligus memberikan interpretabilitas yang lebih baik terhadap keputusan *model* (Bahdanau dkk., 2015). Penelitian terdahulu telah membuktikan bahwa kombinasi BiGRU dan atensi mampu memberikan peningkatan akurasi yang signifikan, berkisar antara 5-10%,

khususnya pada domain analisis sentimen dan deteksi kondisi mental (Baziotis dkk., 2017; Yang dkk., 2016).

Sebagai pembeda utama dalam penelitian ini dibandingkan arsitektur klasifikasi *multiclass* konvensional, bagian luaran *model* tidak menggunakan fungsi *softmax* standar. Melainkan, *model* ini dimodifikasi dengan menggunakan *Ordinal Output Layer* berbasis *Cumulative link model*. Modifikasi ini dilakukan untuk mengakomodasi sifat label depresi yang bersifat *ordinal* atau berjenjang (misalnya: *minimum* < *mild* < *moderate* < *severe*), sehingga *model* tidak hanya belajar membedakan kategori, tetapi juga memahami urutan tingkat keparahan tersebut. Secara visual, struktur hierarki dan aliran data dari *input* hingga mencapai tahap prediksi akhir dengan modifikasi lapisan *ordinal* tersebut dapat diamati pada Gambar 3.6.



Gambar 3.6 Arsitektur Modifikasi BiGRU dengan Mekanisme *Attention* dan Lapisan *Ordinal*

Selama proses propagasi maju (*forward propagation*), aliran informasi melewati setiap lapisan secara sistematis untuk mentransformasi urutan satuan kata mentah menjadi prediksi probabilitas kelas yang valid. Berikut adalah rincian fungsionalitas teknis serta formulasi matematis yang dijalankan pada setiap lapisan arsitektur sebagaimana direpresentasikan dalam Gambar 3.6:

1. *Input Layer* (*Embedding Layer*)

Pada tahap awal arsitektur pemodelan ini, lapisan masukan (*input layer*) bertugas menerima representasi data yang telah diproses pada tahap ekstraksi fitur FastText (sebagaimana telah dijelaskan pada subbab sebelumnya). Alur penerimaan data ini melanjutkan proses di mana teks mentah telah dipecah menjadi sekuens token (Rumus 3.1), diberikan penanda batas (*boundary markers*) untuk ekstraksi *subword n-grams* (Rumus 3.2 dan 3.3), lalu dipetakan menjadi vektor berdimensi tetap berdasarkan kontribusi kata utuh dan *subword*-nya menggunakan FastText *pre-trained* (Rumus 3.4 dan 3.5).

Vektor-vektor representasi tersebut kemudian disusun menjadi matriks dokumen yang mempertahankan urutan temporal sekuens (Rumus 3.6). Agar matriks ini dapat diproses oleh arsitektur *deep learning* dalam komputasi *batch*, panjang sekuens dinormalisasi menjadi seragam melalui proses *padding* atau *truncation* (Rumus 3.7). Token *padding* yang ditambahkan direpresentasikan secara absolut sebagai vektor bernilai nol (Rumus 3.8), dan untuk memastikan token buatan tersebut tidak mengganggu pemodelan konteks, diterapkan perlindungan (*masking*) biner (Rumus 3.9).

Sebagai hasil akhirnya, *Input Layer* pada *model* ini secara langsung mendistribusikan matriks *embedding* berdimensi tetap (Rumus 3.10) berukuran $L \times d$ (dengan panjang sekuens $L = 200$ dan dimensi FastText $d = 300$). Seluruh parameter *embedding* pada lapisan ini dibekukan (*frozen* / tidak ikut dilatih ulang) untuk mempertahankan integritas pengetahuan semantik linguistik bawaan. Matriks tersebut kemudian diteruskan ke lapisan *Bidirectional* GRU untuk dilakukan pemrosesan temporal.

2. *Bidirectional Gated Recurrent Unit (BiGRU) Layer*

Setelah matriks representasi kata terbentuk pada lapisan masukan, tantangan utama dalam pemrosesan bahasa alami adalah memahami struktur kalimat yang bersifat sekuensial. Sebuah kata tidaklah berdiri sendiri, maknanya sangat bergantung pada kata-kata yang mendahului maupun yang mengikutinya. Untuk menangkap ketergantungan temporal dan konteks global ini, matriks *embedding* diumpankan ke dalam lapisan *Bidirectional Gated Recurrent Unit* (BiGRU). Lapisan ini bertugas melakukan ekstraksi fitur temporal melalui pemrosesan dua arah secara paralel.

a) Inisialisasi Parameter Bobot *(Glorot Uniform)*

Sebelum jaringan syaraf memulai proses iterasi pembelajaran, parameter bobot awal harus diatur sedemikian rupa guna menghindari permasalahan gradien yang tidak stabil. Jaringan *recurrent* sangat sensitif terhadap nilai awal, bobot yang terlalu besar dapat menyebabkan gradien meledak, sementara bobot yang terlalu kecil dapat menyebabkan gradien lenyap sebelum mencapai lapisan awal. Oleh karena itu, seluruh matriks bobot jaringan (W) diinisialisasi menggunakan metode

Glorot Uniform Initialization. Metode ini menyeimbangkan variansi sinyal pada setiap lapisan berdasarkan jumlah unit masukan dan luaran, sebagaimana dirumuskan pada Rumus 3.11:

$$W \sim \mathcal{U}\left(-\sqrt{\frac{6}{d_h + d_e}}, \sqrt{\frac{6}{d_h + d_e}}\right) \tag{3.11}$$

Keterangan:
W = Matriks bobot masukan awal jaringan.
$\mathcal{U}$ = Distribusi statistik Uniform (seragam).
d_h = Dimensi vektor memori (hidden state), ditetapkan sebesar 128.
d_e = Dimensi vektor representasi kata (input *embedding*), sebesar 300.

b) Mekanisme Gerbang GRU (*Gating Mechanism*)

Inti dari komputasi GRU terletak pada kemampuannya menyaring memori secara dinamis melalui mekanisme gerbang (*gating*). Mekanisme ini memungkinkan sel jaringan untuk secara cerdas menentukan informasi mana yang relevan untuk dipertahankan dan informasi mana yang bersifat *noise* sehingga harus dilupakan. Aliran data di dalam setiap sel GRU dikendalikan oleh dua gerbang utama: *Update Gate* dan *Reset Gate*.

Langkah komputasi pertama pada setiap *timestep* (t) adalah menentukan seberapa banyak informasi dari masa lalu yang layak dipertahankan untuk memprediksi masa depan. Tugas ini dieksekusi oleh *Update Gate* (z_t). Perhitungan ini melibatkan transformasi linear dari input saat ini dan memori sebelumnya yang ditekan oleh fungsi aktivasi Sigmoid, sebagaimana ditunjukkan pada Rumus 3.12:

$$z_t = \sigma(W_z e_t + U_z h_{t-1} + b_z) \tag{3.12}$$

Keterangan:
$z_t \in R^{128}$ = Vektor luaran update gate pada *timestep* ke-t.
σ = Fungsi aktivasi Sigmoid yang memetakan nilai ke dalam rentang [0, 1].
e_t = Vektor input *embedding* kata pada *timestep* ke-t.

h_{t-1} = Keadaan tersembunyi (hidden state) dari *timestep* sebelumnya.
W_z = Matriks bobot untuk input pada update gate.
U_z = Matriks bobot untuk hidden state pada update gate.
b_z = Vektor bias untuk update gate.

Di saat yang bersamaan, jaringan harus mampu mengabaikan informasi masa lalu jika konteks kalimat berubah. Mekanisme "melupakan" ini dijalankan oleh *Reset Gate* (r_t). Gerbang ini menentukan porsi memori masa lalu yang akan diabaikan sebelum membentuk representasi baru, sesuai dengan perhitungan pada Rumus 3.13:

$$r_t = \sigma(W_r e_t + U_r h_{t-1} + b_r) \tag{3.13}$$

Keterangan:
$r_t \in R^{128}$ = Vektor luaran reset gate pada *timestep* ke-t.
σ = Fungsi aktivasi Sigmoid untuk menghasilkan nilai probabilitas filtrasi.
e_t = Vektor input *embedding* pada *timestep* ke-t.
h_{t-1} = Hidden state dari memori masa lalu.
W_r = Matriks bobot input untuk reset gate.
U_r = Matriks bobot hidden state untuk reset gate.
b_r = Vektor bias untuk reset gate.

c) Pembaruan Status Memori (*Hidden State*)

Setelah filter dari *Reset Gate* didapatkan, *model* mulai meracik informasi baru yang potensial untuk disimpan dalam memori jangka panjang. Informasi ini disebut sebagai *Candidate Hidden State* ($\widetilde{h_t}$). Proses ini mencampurkan input kata saat ini dengan sisa memori masa lalu yang telah disaring oleh *reset gate*, menggunakan fungsi aktivasi Tangen Hiperbolik ($tanh$) untuk menjaga data tetap terpusat, sebagaimana dirumuskan pada Rumus 3.14:

$$\widetilde{h_t} = \tanh(W_h e_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{3.14}$$

Keterangan:
$\widetilde{h_t} \in R^{128}$ = Vektor kandidat hidden state baru pada *timestep* ke-t.
$tanh$ = Fungsi aktivasi tangen hiperbolik dengan rentang luaran [-1, 1].
e_t = Vektor fitur kata saat ini.
r_t = Nilai filtrasi dari reset gate.

h_{t-1} = Memori hidden state sebelumnya.
W_h, U_h, b_h = Matriks bobot dan bias untuk kandidat hidden state.
$\odot$ =: Operasi perkalian elemen-per-elemen (*Hadamard product*).

Tahap akhir dari dinamika sel GRU adalah memperbarui status memori utama. Hasilnya adalah *Final Hidden State* ($h_t^{\rightarrow}$), yang didapatkan melalui interpolasi linear antara memori lama dan kandidat baru dengan kendali penuh dari *Update Gate*. Perhitungan transisi memori ini ditunjukkan pada Rumus 3.15:

$$h_t^{\rightarrow} = (1 - z_t) \odot h_{t-1}^{\rightarrow} + z_t \odot \widetilde{h_t} \tag{3.15}$$

Keterangan:
$h_t^{\rightarrow} \in R^{\mathbb{128}}$ = Keadaan tersembunyi (hidden state) akhir arah maju pada *timestep* ke-t.
z_t = Nilai instruksi dari update gate.
$h_{t-1}^{\rightarrow}$ = Hidden state arah maju dari siklus sebelumnya.
$\widetilde{h_t}$ = Kandidat memori baru yang siap ditulis.
$\odot$ = Operasi *Hadamard product*.

d) Ekstraksi Konteks Dua Arah (*Bidirectional Processing*)

Seluruh proses gerbang di atas merepresentasikan aliran informasi dari awal kalimat menuju akhir kalimat (*Forward* GRU). Namun, dalam memahami emosi yang mendalam pada teks depresi, konteks masa depan sering kali mengklarifikasi makna masa lalu. Oleh karena itu, arsitektur ini menjalankan fungsi *Backward* GRU yang membaca teks secara terbalik dari belakang ke depan menggunakan ruang parameter yang independen, sebagaimana ditunjukkan pada Rumus 3.16:

$$h_t^{\leftarrow} = \mathrm{GRU}_{backward}(e_t, h_{t+1}^{\leftarrow}) \tag{3.16}$$

Keterangan:
$h_t^{\leftarrow} \in R^{\mathbb{128}}$ = Hidden state arah mundur (*backward*) pada *timestep* ke-t.
$\mathrm{GRU}_{backward}$ = Fungsi pemrosesan sel GRU dengan parameter bobot mandiri.
e_t = Vektor input pada *timestep* ke-t.
$h_{t+1}^{\leftarrow}$ = Hidden state arah mundur dari posisi kata setelahnya.

e) Penggabungan Fitur dan Regularisasi

Untuk menghasilkan satu pemahaman ruang-waktu yang utuh, representasi dari masa lalu ($h_t^{\rightarrow}$) dan representasi dari masa depan ($h_t^{\leftarrow}$) digabungkan pada setiap titik kata menggunakan operasi penyambungan (*concatenation*), sebagaimana dirumuskan pada Rumus 3.17:

$$H_t = [h_t^{\rightarrow}; h_t^{\leftarrow}] \tag{3.17}$$

Keterangan:
$H_t \in R^{256}$ = Hidden state kontekstual gabungan pada *timestep* ke-t.
$h_t^{\rightarrow}$ = Vektor memori arah maju berdimensi 128.
$h_t^{\leftarrow}$ = Vektor memori arah mundur berdimensi 128.

Asal-usul angka dimensi 256 diperoleh secara langsung dari operasi ini, di mana vektor berdimensi 128 dari arah maju dipadukan secara horizontal dengan vektor berdimensi 128 dari arah mundur, menghasilkan ruang fitur yang lebih kaya. Sebagai penutup, untuk memitigasi risiko penghafalan pola yang berlebihan (*overfitting*), hasil gabungan tersebut diregularisasi melalui teknik *Dropout*, yang dirumuskan pada Rumus 3.18:

$$H_t = Dropout(H_t, p = 0.4) \tag{3.18}$$

Keterangan:
H_t = Matriks hidden state gabungan pasca-regularisasi.
p = Probabilitas penjatuhan neuron, ditetapkan sebesar 0.4.

Dengan probabilitas 40% neuron dimatikan secara acak pada setiap iterasi, jaringan dipaksa untuk membangun arsitektur pemahaman yang lebih tangguh dan tidak bergantung pada jalur fitur tertentu saja. Matriks luaran (H_t) yang telah stabil inilah yang selanjutnya dikirim menuju lapisan pengkodean konteks.

3. Context Encoding Layer

Setelah lapisan BiGRU menghasilkan matriks representasi kontekstual berdimensi 256 untuk setiap posisi kata dalam sekuens, langkah krusial berikutnya adalah merangkum seluruh informasi tersebut menjadi satu vektor representasi tunggal yang padat. Vektor ini, yang disebut sebagai vektor konteks (*context vector*), harus mampu merepresentasikan esensi dari seluruh dokumen agar dapat diproses oleh lapisan klasifikasi akhir. Dalam penelitian ini, proses pengkodean konteks dilakukan melalui dua pendekatan berbeda yang bergantung pada skenario pengujian yang digunakan.

a) Padding Masking

Tantangan pertama dalam merangkum sekuens teks adalah adanya token *padding* yang ditambahkan untuk menyeragamkan panjang data. Karena token ini tidak mengandung informasi semantik, keberadaannya tidak boleh memengaruhi perhitungan statistik maupun pembobotan konteks. Sebagai solusi matematis, diterapkan mekanisme penapisan atau *masking* biner yang diformulasikan pada Rumus 3.19:

$$m_t = \begin{cases} 1, & jika\ x_t \neq 0 \\ 0, & jika\ x_t = 0 \end{cases} \tag{3.19}$$

Keterangan:
m_t = Nilai masking biner pada posisi ke-t yang berfungsi sebagai sakelar informasi.
x_t = Indeks token asli pada posisi ke-t dalam sekuens dokumen.

Rumus ini memastikan bahwa setiap operasi perhitungan di lapisan berikutnya hanya akan memproses elemen yang memiliki nilai $m_t = 1$. Dengan menetapkan nilai 0 pada posisi *padding*, *model* secara efektif mengisolasi pengaruh

data buatan tersebut sehingga tidak mendistorsi representasi akhir dari dokumen teks pasien.

b) *Masked Mean Pooling*

Pada skenario *baseline*, perangkuman informasi dilakukan dengan menghitung nilai rata-rata dari seluruh vektor representasi tersembunyi yang dihasilkan oleh BiGRU. Namun, perhitungan rata-rata ini harus dimodifikasi agar hanya memperhitungkan token valid melalui integrasi vektor *masking*. Pendekatan ini disebut sebagai *Masked Mean Pooling* yang dirumuskan pada Rumus 3.20:

$$c = \frac{\sum_{t=1}^{T} m_t \cdot H_t}{\max(\sum_{t=1}^{T} m_t, 1)} \tag{3.20}$$

Keterangan:
$c \in R^{256}$ = Vektor konteks hasil perangkuman yang merangkum seluruh informasi sekuens.
m_t = Vektor masking biner yang mencegah kontribusi nilai padding.
$H_t \in R^{256}$ = Representasi kontekstual gabungan BiGRU pada posisi ke-t.
$\sum m_t$ = Pembagi yang merepresentasikan panjang sekuens asli tanpa menghitung padding.

Melalui Rumus 3.20, *model* menjumlahkan seluruh fitur kontekstual dan membaginya dengan jumlah kata yang sebenarnya ada dalam kalimat. Penggunaan fungsi max pada pembagi bertujuan untuk menjaga stabilitas numerik agar tidak terjadi pembagian dengan nol. Meskipun efektif sebagai standar pembanding, metode ini memiliki keterbatasan karena memberikan bobot yang sama rata pada setiap kata, tanpa membedakan kata mana yang lebih relevan untuk indikasi depresi.

c) *Additive Attention*

Untuk mengatasi keterbatasan metode rata-rata, digunakan mekanisme atensi tambahan (*Additive Attention*) yang memungkinkan *model* untuk

memberikan fokus perhatian secara dinamis pada bagian teks tertentu. Mekanisme ini bekerja melalui empat tahapan matematis yang sistematis.

Tahap pertama adalah perhitungan skor relevansi atau tingkat kepentingan setiap kata melalui transformasi non-linear. Langkah ini menghasilkan Skor Energi (u_t) yang ditunjukkan pada Rumus 3.21:

$$u_t = v^\top \tanh(W_a H_t + b_a) \tag{3.21}$$

Keterangan:
$u_t \in R$ = Skor kepentingan mentah untuk posisi ke-t.
$W_a \in R^{128\times256}$ = Matriks bobot transformasi yang dipelajari selama pelatihan.
$b_a \in R^{128}$ = Vektor bias untuk memberikan fleksibilitas pada *model*.
$v \in R^{128}$ = Vektor konteks atensi yang dipelajari untuk menentukan pola kata penting.
H_t = Masukan berupa representasi tersembunyi dari BiGRU.

Tahap kedua adalah memastikan bahwa skor energi pada posisi *padding* tidak ikut dihitung dalam tahap normalisasi. Hal ini dilakukan dengan menetapkan nilai negatif yang sangat besar pada posisi tersebut, sebagaimana dirumuskan pada Rumus 3.22:

$$u'_t = \begin{cases} u_t, & jika\ m_t = 1 \\ -10000, & jika\ m_t = 0 \end{cases} \tag{3.22}$$

Keterangan:
u'_t = Skor energi yang telah disaring melalui mekanisme perlindungan.
u_t = Skor energi asli dari hasil transformasi linear.
m_t = Vektor masking biner. Penggunaan nilai konstanta -10000 dilakukan agar pada tahap eksponensial berikutnya, nilai tersebut mendekati nol mutlak.

Tahap ketiga adalah mengubah skor energi yang telah disaring menjadi distribusi bobot probabilitas yang disebut sebagai *Attention weights* (α_t). Proses normalisasi ini menggunakan fungsi *Softmax* agar total seluruh bobot dalam satu dokumen berjumlah satu sesuai dengan Rumus 3.23:

$$\alpha_t = \frac{\exp(u'_t)}{\sum_{j=1}^{T} \exp(u'_j)} \tag{3.23}$$

Keterangan:
$\alpha_t \in [0,1]$ = Bobot perhatian yang merepresentasikan porsi kepentingan kata pada posisi ke-t.
exp = Fungsi eksponensial untuk mengubah skor menjadi nilai positif.

Tahap akhir dari mekanisme ini adalah pembentukan *Context Vector* (c) yang sesungguhnya. Vektor ini didapatkan melalui penjumlahan berbobot dari seluruh representasi tersembunyi BiGRU di mana setiap vektor H_t dikalikan dengan bobot kepercayaannya masing-masing. Hal ini dirumuskan pada Rumus 3.24:

$$c = \sum_{t=1}^{T} \alpha_t H_t \tag{3.24}$$

Keterangan:
$c \in R^{256}$ = Vektor konteks hasil agregasi cerdas yang merangkum informasi paling relevan.
α_t = Bobot perhatian yang menentukan dominasi informasi dari kata tertentu.
H_t = Representasi kontekstual dari BiGRU.

Mekanisme ini memungkinkan *model* untuk secara otomatis memperhatikan kata-kata kunci yang memiliki muatan emosional tinggi seperti *hopeless* atau *worthless* sambil mengabaikan kata hubung yang kurang relevan. Dengan cara ini vektor konteks c yang dihasilkan tidak lagi sekadar rata-rata dari kata-kata melainkan sebuah intisari yang kaya akan informasi diagnostik yang akan diteruskan menuju lapisan klasifikasi selanjutnya.

4. Dense Layer

Vektor konteks yang dihasilkan dari lapisan sebelumnya merupakan representasi tingkat tinggi yang merangkum seluruh informasi dari sekuens teks.

Namun, sebelum data tersebut dapat diklasifikasikan ke dalam kategori tingkat keparahan depresi, diperlukan sebuah transformasi untuk memetakan fitur-fitur tersebut ke dalam ruang representasi yang lebih padat dan diskriminatif. Tahap ini dilakukan pada lapisan padat atau yang dikenal sebagai *Fully connected layer*, di mana setiap elemen dalam vektor masukan dihubungkan sepenuhnya dengan unit-unit pada lapisan ini untuk mengekstraksi pola-pola yang lebih kompleks.

a) *Linear transform and Relu activation*

Langkah pertama pada lapisan ini adalah melakukan proyeksi linear dari vektor konteks berdimensi 256 menuju ruang fitur baru yang memiliki dimensi lebih kecil yaitu 128. Proyeksi ini segera diikuti oleh penerapan fungsi aktivasi untuk memperkenalkan sifat non-linearitas ke dalam *model*, yang memungkinkan jaringan untuk mempelajari batasan keputusan atau *decision boundary* yang lebih rumit. Operasi gabungan ini diformulasikan pada Rumus 3.25:

$$z' = \max(0, W_d c + b_d) \tag{3.25}$$

Keterangan:
$z' \in R^{128}$ = Vektor fitur lanjutan yang telah melalui transformasi linear dan aktivasi non-linear.
$\max(0, x)$ = Fungsi aktivasi *Rectified linear unit* (*Relu*) yang memberikan nilai nol untuk masukan negatif dan nilai identitas untuk masukan positif.
$W_d \in R^{128 \times 256}$ = Matriks pembobot yang menentukan kekuatan hubungan antar neuron.
$b_d \in R^{128}$ = Vektor bias yang memberikan fleksibilitas pergeseran pada fungsi aktivasi.
$c \in R^{256}$ = Vektor konteks yang diterima dari lapisan pengkodean konteks.

Rumus tersebut menjamin bahwa setiap tahapan perhitungan pada lapisan selanjutnya hanya akan mengolah elemen yang memiliki nilai positif. Kondisi ini dimungkinkan melalui penggunaan fungsi aktivasi *Relu* yang bertugas untuk mereset seluruh nilai masukan negatif menjadi nol secara otomatis. Dengan menghilangkan nilai-nilai negatif tersebut, jaringan saraf akan memiliki sifat non-

linear yang sangat dibutuhkan untuk mempelajari hubungan antar-data yang rumit dan tidak sederhana. Proses transformasi ini menjadi faktor kunci dalam memastikan bahwa representasi akhir dari dokumen teks yang sedang dianalisis tetap tajam serta akurat bagi kebutuhan sistem.

b) *Dropout*

Lapisan padat dengan jumlah parameter yang besar memiliki kecenderungan untuk mengalami *overfitting*, di mana *model* menghafal data latih secara berlebihan sehingga gagal melakukan generalisasi pada data baru. Sebagai mekanisme pertahanan, diterapkan teknik regularisasi berupa *Dropout* yang bekerja dengan cara menonaktifkan sejumlah unit saraf secara acak pada setiap iterasi pelatihan. Operasi ini dirumuskan pada Rumus 3.26:

$$z = Dropout(z', p = 0.6) \tag{3.26}$$

Keterangan:
$z \in R^{128}$ = Vektor representasi akhir yang siap untuk diproses ke tahap klasifikasi.
z' = Vektor fitur yang dihasilkan dari aktivasi *Relu*.
p = Probabilitas penonaktifan unit saraf yang ditetapkan sebesar 0,6.

Nilai probabilitas sebesar 0,6 mencerminkan strategi regularisasi yang cukup agresif karena 60% neuron akan dinonaktifkan secara acak dalam setiap langkah pelatihan. Hal ini memaksa jaringan untuk tidak bergantung pada keberadaan neuron tertentu saja dan mendorong terciptanya fitur-fitur yang lebih tangguh atau *robust*. Hasil dari operasi ini adalah vektor fitur z yang telah stabil dan siap untuk diteruskan menuju lapisan luaran *ordinal* untuk mendapatkan prediksi akhir tingkat keparahan depresi.

5. Ordinal Output Layer

Sebagai tahapan akhir dari arsitektur *model* ini, vektor fitur yang dihasilkan oleh lapisan padat dikonversi menjadi prediksi kelas tingkat keparahan depresi. Mengingat label depresi memiliki sifat *ordinal* yang berjenjang, penggunaan fungsi *softmax* standar dinilai kurang optimal karena fungsi tersebut memperlakukan setiap kelas sebagai kategori yang saling bebas dan setara. Sebagai solusinya, penelitian ini mengimplementasikan lapisan luaran *ordinal* berbasis *Cumulative link model* yang memanfaatkan batasan ambang untuk memisahkan tingkatan kelas secara terurut sesuai dengan teori yang dikemukakan oleh McCullagh (1980).

a) *Threshold logits*

Proses klasifikasi diawali dengan memetakan vektor fitur akhir menuju ruang logit untuk setiap ambang batas kelas. Karena terdapat empat kategori tingkatan depresi dalam penelitian ini, maka diperlukan tiga buah nilai ambang $(K-1)$ untuk mempartisi ruang probabilitas secara kumulatif. Perhitungan logit batas ini diformulasikan pada Rumus 3.27:

$$l_k = w_k^{\top} z + b_k, \quad k \in \{0,1,2\} \tag{3.27}$$

Keterangan:
l_k = Nilai logits ambang batas untuk tingkatan ke-k.
$w_k \in R^{128}$ = Vektor bobot yang dipelajari untuk membedakan ambang batas ke-k.
$z \in R^{128}$ = Vektor fitur masukan yang berasal dari lapisan padat.
$b_k \in R$ = Parameter bias yang berhubungan dengan ambang batas ke-k.

Setiap elemen l_k mewakili skor mentah yang akan menentukan posisi suatu observasi terhadap batas-batas kelas depresi. Tidak ada fungsi aktivasi tambahan seperti *Relu* yang diterapkan pada tahap ini karena nilai logit ambang batas harus

dapat bernilai positif maupun negatif secara bebas untuk menyesuaikan dengan karakteristik fitur yang diekstraksi dari setiap dokumen teks pasien.

b) *Cumulative Probability*

Setelah nilai logit untuk setiap ambang batas didapatkan, langkah berikutnya adalah mengubah nilai mentah tersebut menjadi probabilitas kumulatif. Probabilitas ini menyatakan besarnya peluang bahwa label kelas sebenarnya berada pada tingkatan yang lebih tinggi dari ambang batas k. Transformasi ini menggunakan fungsi aktivasi Sigmoid sebagaimana dirumuskan pada Rumus 3.28:

$$P(Y > k) = \frac{1}{1 + \exp(-l_k)} \tag{3.28}$$

Keterangan:
$P(Y > k)$ = Peluang kumulatif bahwa kelas sebenarnya berada pada tingkatan yang lebih besar dari k.
exp = Fungsi eksponensial.
l_k = Nilai logits ambang batas untuk tingkatan ke-k.

Fungsi Sigmoid secara efektif memetakan nilai logit yang tidak terbatas ke dalam rentang rasio antara angka nol hingga satu. Nilai probabilitas kumulatif yang mendekati angka satu menunjukkan tingkat kepercayaan *model* yang tinggi bahwa sampel tersebut melampaui batas kelas k, sementara nilai yang mendekati angka nol menunjukkan kecenderungan sampel berada pada tingkatan yang lebih rendah.

c) *Class probability*

Berdasarkan nilai probabilitas kumulatif yang telah diperoleh, probabilitas absolut untuk setiap kelas depresi dihitung melalui selisih antar probabilitas kumulatif yang berdekatan. Pendekatan ini memastikan bahwa distribusi

probabilitas yang dihasilkan mencerminkan urutan alami dari kelas depresi tersebut. Perhitungan probabilitas per kelas dirumuskan pada Rumus 3.29:

$$P(Y = k) = \begin{cases} 1 - P(Y > 0), & jika\ k = 0 \\ P(Y > k - 1) - P(Y > k), & jika\ 0 < k < K - 1 \\ P(Y > K - 2), & jika\ k = K - 1 \end{cases} \quad (3.29)$$

Keterangan:
$P(Y = k)$ = Probabilitas bahwa sampel termasuk dalam kelas depresi tingkat ke-k.
$P(Y > k)$ = Nilai probabilitas kumulatif yang dihitung pada tahap sebelumnya.
K = Jumlah keseluruhan kelas *ordinal* yang dalam penelitian ini bernilai 4.

Melalui formulasi ini, probabilitas untuk kelas terendah yaitu *minimum* dihitung dari komplemen ambang batas pertama. Probabilitas untuk kelas tertinggi yaitu *severe* diambil langsung dari nilai ambang batas terakhir. Untuk kelas-kelas di antaranya, probabilitas ditentukan oleh luas area di antara dua ambang batas yang berurutan. Konstruksi ini secara otomatis menjamin bahwa jumlah seluruh probabilitas kelas akan bernilai tepat satu.

d) *Numerical clamping* dan Prediksi Akhir

Selama fase pelatihan *model*, terdapat risiko terjadinya ketidakstabilan numerik apabila nilai probabilitas kelas sangat mendekati nol sehingga dapat menyebabkan kegagalan perhitungan pada fungsi kerugian. Untuk mencegah hal tersebut, dilakukan proses penstabilan melalui teknik *clamping* sebagaimana dirumuskan pada Rumus 3.30:

$$P_{final}(Y = k) = \text{clamp}(P(Y = k), 10^{-7}, 1 - 10^{-7}) \quad (3.30)$$

Keterangan:
$P_{final}(Y = k)$ = Nilai probabilitas yang telah melalui tahap penstabilan numerik agar tetap aman untuk perhitungan logaritma.
clamp = Fungsi pembatas nilai agar tetap berada dalam rentang yang ditetapkan.

Hasil probabilitas tersebut kemudian digunakan untuk menentukan keputusan klasifikasi akhir dengan memilih kelas yang memiliki probabilitas paling tinggi. Keputusan prediksi ini diformulasikan pada Rumus 3.31:

$$\hat{y} = \arg\max_{k} P\,(Y = k) \tag{3.31}$$

Keterangan:
$\hat{y}$ = Keputusan prediksi akhir tingkat keparahan depresi hasil olah jaringan.
$\arg\max_{k}$ = Operasi untuk mencari indeks kelas k yang memiliki nilai probabilitas tertinggi.

e) *Class weighting Strategy*

Sebagai langkah tambahan untuk menangani masalah ketidakseimbangan distribusi data pada setiap tingkatan depresi, *model* menerapkan strategi pembobotan kelas saat proses perhitungan kerugian. Teknik ini memberikan penalti yang lebih besar bagi kesalahan prediksi pada kelas-kelas yang jumlah sampelnya sedikit. Perhitungan bobot kelas ini menggunakan metode akar kuadrat yang dihaluskan sebagaimana dirumuskan pada Rumus 3.32:

$$w_c = \sqrt{\frac{N}{K \cdot N_c}} \tag{3.32}$$

Keterangan:
w_c = Nilai bobot penalti untuk kelas depresi ke-c selama proses pelatihan.
N = Total jumlah seluruh sampel data yang digunakan dalam penelitian.
K = Jumlah kategori tingkat keparahan depresi.
N_c = Jumlah sampel yang tersedia pada kategori kelas depresi ke-c.

Penggunaan akar kuadrat dalam rumus ini bertujuan untuk menghaluskan rasio pembobotan agar penalti yang diberikan tidak terlalu ekstrem namun tetap efektif untuk meningkatkan sensitivitas *model* terhadap kelas minoritas. Dengan pengintegrasian strategi pembobotan ini, *model* diharapkan mampu menghasilkan

performa yang lebih seimbang di seluruh tingkat keparahan depresi tanpa terdistorsi oleh dominasi kelas mayoritas.

3.3 Lingkungan Penerapan dan Pengujian Sistem

Pengembangan, pelatihan, dan pengujian *model* kecerdasan buatan dalam penelitian ini dilakukan di dalam suatu lingkungan komputasi yang terisolasi dan terkendali. Pencatatan spesifikasi lingkungan ini bertujuan untuk memastikan validitas empiris serta menjamin bahwa seluruh eksperimen yang dilakukan bersifat dapat direproduksi (*reproducible*) oleh peneliti lain di masa mendatang. Pemrosesan data bahasa alami (*Natural Language Processing*) berbasis *deep learning* membutuhkan daya komputasi yang masif, terutama dalam fase pembaruan bobot (*weight updating*) pada jaringan saraf tiruan. Oleh karena itu, pelatihan *model* dalam penelitian ini sangat bergantung pada akselerasi *Graphics Processing Unit* (GPU). Rincian spesifikasi perangkat keras (*hardware*) yang digunakan dalam pengujian disajikan pada Tabel 3.13.

Tabel 3.13 Spesifikasi Perangkat Keras Pengujian

Komponen Perangkat Keras	**Spesifikasi Detail**
Prosesor Utama (CPU)	AMD Ryzen 5 5500
Memori Utama (RAM)	16 GB Viper Steel 3000 MHz
Papan Induk (Motherboard)	Gigabyte B450M DS3H
Kartu Grafis (GPU)	NVIDIA GeForce GTX 1080 8 GB VRAM

Selain infrastruktur fisik, stabilitas pengujian juga dipengaruhi oleh konfigurasi tumpukan perangkat lunak (*software stack*). Penelitian ini mengandalkan bahasa pemrograman Python dengan pustaka utama PyTorch untuk

membangun arsitektur jaringan. Rincian perangkat lunak yang digunakan disajikan pada Tabel 3.14.

Tabel 3.14 Spesifikasi Perangkat Lunak Pengujian

Komponen Perangkat Lunak	**Spesifikasi / Versi**
Sistem Operasi	Windows 11 Pro 64-bit (Build 22631)
Bahasa Pemrograman	Python 3.10.19 (Anaconda Distribution)
Kerangka Kerja (Framework)	PyTorch 2.1.2
Akselerasi Komputasi Paralel	CUDA Toolkit 11.8 dan cuDNN 8700
Pustaka Pemrosesan Data	Pandas (2.0+), NumPy (1.23+), Scikit-Learn (1.2+)
Pustaka Representasi Teks	Fasttext-wheel, NLTK (3.8+)

Guna memastikan bahwa inisialisasi bobot awal pada *model* selalu konsisten di setiap putaran eksperimen, ditetapkan nilai *random seed* global sebesar 42. Selain itu, mode deterministik pada arsitektur cuDNN diaktifkan (cudnn.deterministic = True) agar algoritma perhitungan konvolusi GPU tidak berubah-ubah, yang berpotensi menghasilkan selisih metrik pada pengujian berulang.

BAB IV

UJI COBA DAN PEMBAHASAN

4.1 Desain Uji Coba

Untuk mengevaluasi kontribusi individual dari setiap komponen arsitektur *model* terhadap performa klasifikasi, penelitian ini merancang eksperimen multi-skenario yang sistematis dan komprehensif. Desain eksperimen ini bertujuan untuk mengisolasi efek dari pembobotan kelas (*class weighting*) dengan mengukur seberapa besar peningkatan performa yang diberikan oleh penanganan *class imbalance*. Selain itu, eksperimen ini juga bertujuan untuk mengisolasi efek dari mekanisme atensi dengan mengukur kontribusi atensi dalam meningkatkan kemampuan *model* untuk fokus pada fitur yang relevan. Terakhir, eksperimen ini mengevaluasi efek interaksi dengan memahami bagaimana kombinasi pembobotan kelas dan atensi bekerja secara sinergis.

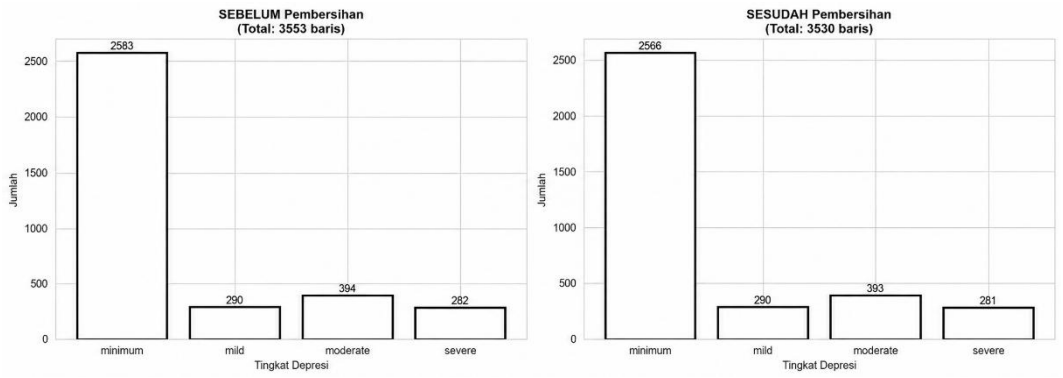
Desain eksperimen mengikuti prinsip *ablation study* di mana setiap komponen *model* dihilangkan atau ditambahkan secara sistematis untuk mengukur kontribusinya terhadap performa keseluruhan (Meyes dkk., 2019). Pendekatan ini memungkinkan analisis yang lebih mendalam dibanding hanya melatih satu *model* dengan fitur lengkap (*full-featured*), karena dapat mengidentifikasi komponen mana yang paling berpengaruh dan apakah penambahan kompleksitas *model* sepadan dengan peningkatan performa.

4.1.1 Persiapan Data

Pada tahap eksplorasi awal, berkas data yang digunakan memiliki 3.553 baris data dengan seluruh entri terisi lengkap (tidak terdapat *missing values* maupun *string* kosong). Namun, hasil analisis lebih mendalam menemukan adanya anomali berupa 19 duplikat eksak (pasangan teks dan label yang sama persis) serta 2 kasus ambiguitas di mana teks yang identik memiliki label yang saling bertentangan (satu teks berlabel *minimum* dan *severe*, serta satu teks lain berlabel *minimum* dan *moderate*).

Untuk menjaga kualitas supervisi dan menghindari ambiguitas yang dapat membingungkan *model* saat proses pelatihan, dilakukan tahapan pembersihan data dalam satu alur. Proses ini menghapus 4 baris entri yang memiliki label inkonsisten, dilanjutkan dengan menghapus duplikat eksak dan hanya menyisakan satu entri representatif (menghapus 19 baris).

Setelah proses pembersihan selesai, diperoleh 3.530 baris data bersih yang merepresentasikan sekitar 99,35% dari total data awal. Pembersihan ini hanya menyebabkan perubahan kecil pada distribusi label, dengan rincian sebagai berikut: kelas *minimum* berkurang dari 2.587 menjadi 2.566 data, kelas *mild* tetap berjumlah 290 data, kelas *moderate* berkurang dari 394 menjadi 393 data, dan kelas *severe* berkurang dari 282 menjadi 281 data. Meskipun masalah ketidakseimbangan antar kelas (*class imbalance*) masih ada dalam dataset, data yang dihasilkan kini jauh lebih konsisten dan siap digunakan untuk eksperimen klasifikasi *ordinal*. Visualisasi perbandingan sebaran label sebelum dan sesudah proses pembersihan dapat dilihat pada Gambar 4.1.



Gambar 4.1 Diagram Lingkaran Distribusi Label Dataset Sebelum dan Sesudah Pembersihan

4.1.2 Definisi Empat Skenario Pengujian

Penelitian ini mendefinisikan empat skenario pengujian yang berbeda terhadap dataset yang digunakan. Setiap skenario merepresentasikan konfigurasi *model* dengan kombinasi tertentu dari fitur pembobotan kelas dan mekanisme atensi. Sebagai penegasan penting, seluruh skenario pengujian ini secara konsisten menggunakan lapisan *ordinal* atau *ordinal layer* pada tahap akhir klasifikasi untuk memastikan struktur berjenjang dari tingkat keparahan depresi tetap terjaga. Rincian dari keempat skenario tersebut dijabarkan sebagai berikut:

a) *Model* dasar (Tanpa Pembobotan kelas dan Tanpa Atensi)

Skenario *model* dasar menggunakan arsitektur BiGRU standar tanpa penambahan fitur khusus. *Model* ini dilatih dengan bobot setara untuk semua kelas dan tidak menggunakan mekanisme atensi untuk menggabungkan *output* BiGRU.

b) *Class Weight Only* (Dengan Pembobotan kelas dan Tanpa Atensi)

Skenario ini menambahkan pembobotan kelas (*class weighting*) pada fungsi *loss* untuk mengatasi ketidakseimbangan distribusi data. *Model* pada skenario ini tetap menggunakan global *Average Pooling* untuk agregasi urutan teks tanpa melibatkan mekanisme atensi.

c) Atensi *Only* (Tanpa Pembobotan kelas dan Dengan Atensi)

Skenario ini mengganti global *Average Pooling* dengan mekanisme atensi untuk secara adaptif memfokuskan *model* pada *timesteps* yang paling relevan. Proses pelatihan pada konfigurasi ini tidak menyertakan fitru pembobotan kelas sehingga seluruh label dievaluasi menggunakan bobot yang setara.

d) *Full Model* (Dengan Pembobotan kelas dan Atensi)

Skenario *full model* menggabungkan kedua fitur utama secara bersamaan untuk menunjang performa komputasi. Konfigurasi ini menggunakan pembobotan kelas untuk mengatasi ketimpangan data sekaligus menerapkan mekanisme atensi untuk penyeleksian fitur linguistik yang adaptif. Ini adalah konfigurasi *model* yang paling lengkap dan diharapkan memberikan performa terbaik.

4.1.3 Metrik Evaluasi

Untuk mengevaluasi dan membandingkan performa dari empat eksperimen, digunakan serangkaian metrik evaluasi yang komprehensif:

a) Akurasi (*Accuracy*)

Akurasi mengukur proporsi prediksi yang benar dari seluruh jumlah prediksi yang dilakukan. Metrik ini paling dasar dalam klasifikasi dan memberikan gambaran umum tentang seberapa sering *model* membuat keputusan yang tepat. Secara formal, akurasi dihitung menggunakan Rumus 4.1:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{\text{Jumlah Prediksi Benar}}{\text{Total Sampel}} \qquad (4.1)$$

Keterangan:
TP (*True Positive*) = jumlah sampel yang diklasifikasikan benar sebagai kelas positif
TN (*True Negative*) = jumlah sampel yang diklasifikasikan benar sebagai kelas negatif atau bukan target
FP (*False Positive*) = jumlah sampel yang sebenarnya bukan kelas target tetapi salah

diklasifikasikan sebagai kelas tersebut
FN (*False Negative*) = jumlah sampel yang sebenarnya termasuk dalam kelas target tetapi gagal dikenali oleh *model*

Meskipun akurasi sering digunakan, metrik ini bisa menyesatkan jika digunakan pada dataset dengan distribusi kelas yang tidak seimbang, karena akurasi tinggi dapat diperoleh hanya dengan memprediksi kelas mayoritas.

b) *Precision* per *Class*

Precision menunjukkan proporsi prediksi positif untuk kelas tertentu yang benar-benar tepat. Nilai *Precision* yang tinggi berarti *model* jarang salah mengklasifikasikan kelas lain sebagai kelas tersebut. *Precision* untuk kelas ke-k dihitung menggunakan Rumus 4.2:

$$Precision_k = \frac{TP_k}{TP_k + FP_k} \tag{4.2}$$

Keterangan:
$Precision_k$ = *Precision* untuk kelas ke-k
TP_k = jumlah prediksi benar untuk kelas ke-k (*True Positive*)
FP_k = jumlah prediksi salah yang diklasifikasikan ke kelas ke-k (*False Positive*)

Precision tinggi menunjukkan bahwa ketika *model* memprediksi suatu sampel sebagai kelas k, prediksi tersebut cenderung benar.

c) *Recall*

Recall mengukur kemampuan *model* untuk menangkap semua baris data dari kelas tertentu. Nilai tinggi berarti *model* berhasil mendeteksi sebagian besar contoh dari kelas tersebut. *Recall* untuk kelas ke-k dihitung menggunakan Rumus 4.3:

$$Recall_k = \frac{TP_k}{TP_k + FN_k} \tag{4.3}$$

Keterangan:
$Recall_k$ = *Recall* untuk kelas ke-k
TP_k = jumlah prediksi benar untuk kelas ke-k
FN_k = jumlah baris data dari kelas ke-k yang gagal dikenali oleh *model* (*False Negative*)

Recall sangat penting dalam kasus di mana kesalahan tipe *False Negative* harus diminimalisasi, misalnya untuk mendeteksi tingkat keparahan depresi yang tinggi.

d) *F1-Score per Class*

F1-Score adalah rata-rata harmonik dari *Precision* dan *Recall*, yang digunakan untuk menyeimbangkan keduanya dalam satu metrik. *F1-Score* sangat berguna ketika distribusi kelas tidak merata. *F1-Score* untuk kelas ke-k dihitung dengan Rumus 4.4:

$$\mathrm{F1}_k = 2 \times \frac{Precision_k \times Recall_k}{Precision_k + Recall_k} \tag{4.4}$$

Keterangan:
$\mathrm{F1}_k$ = *F1-Score* untuk kelas ke-k
$Precision_k$ = *Precision* untuk kelas ke-k
$Recall_k$ = *Recall* untuk kelas ke-k

Nilai F1 yang tinggi menunjukkan bahwa *model* tidak hanya akurat dalam prediksi (*Precision* tinggi), tetapi juga berhasil menangkap semua baris data yang relevan (*Recall* tinggi). *F1-Score* memberikan timbal balik yang seimbang antara kedua metrik tersebut.

e) *Macro-Average F1*

Macro F1 memberikan rata-rata dari semua *F1-Score* per kelas, tanpa mempertimbangkan ukuran atau frekuensi masing-masing kelas. Ini membantu mengukur performa *model* secara seimbang di seluruh kelas, terutama memberikan bobot yang sama untuk kelas minoritas. Perhitungannya mengikuti Rumus 4.5:

$$\mathrm{F1}_{\mathrm{macro}} = \frac{1}{K}\sum_{k=1}^{K} \mathrm{F1}_k \tag{4.5}$$

Keterangan:
$\mathrm{F1}_{macro}$ = *macro*-average *F1-Score*
K = jumlah total kelas (pada penelitian ini $K = 4$)
$\mathrm{F1}_k$ = *F1-Score* untuk kelas ke-k

Metrik ini ideal untuk menilai performa terhadap kelas minoritas yang sering terabaikan oleh *model*. *Macro* F1 memberikan gambaran apakah *model* bekerja dengan baik secara konsisten di semua kelas, bukan hanya pada kelas mayoritas.

f) *Weighted Average F1*

*Weight*ed F1 menghitung rata-rata tertimbang dari *F1-Score* berdasarkan jumlah sampel di setiap kelas. Metrik ini merefleksikan performa *model* yang lebih representatif terhadap distribusi aktual data. *Weight*ed F1 dirumuskan dalam Rumus 4.6:

$$\mathrm{F1}_{\mathrm{weighted}} = \frac{\sum_{k=1}^{K} n_k \times \mathrm{F1}_k}{\sum_{k=1}^{K} n_k} \tag{4.6}$$

Keterangan:
$\mathrm{F1}_{weighted}$ = *weight*ed-average *F1-Score*
n_k = jumlah sampel untuk kelas ke-k
$\mathrm{F1}_k$ = *F1-Score* untuk kelas ke-k
K = jumlah total kelas

Metrik ini memberikan gambaran realistis tentang kinerja *model* secara keseluruhan, terutama pada dataset yang tidak seimbang, karena kelas dengan jumlah sampel lebih banyak akan memiliki pengaruh lebih besar pada skor akhir.

g) *Mean absolute error (MAE)*

Karena penelitian ini menggunakan *ordinal regression*, *MAE* menjadi metrik penting untuk mengukur rata-rata besaran kesalahan prediksi dalam skala *ordinal*. *MAE* mengukur seberapa jauh prediksi *model* menyimpang dari kelas sebenarnya, dengan memperhitungkan jarak antar kelas. *MAE* dihitung menggunakan Rumus 4.7:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \widehat{y_i}| \tag{4.7}$$

Keterangan:
MAE = *Mean absolute error*
N = jumlah total sampel
y_i = kelas sebenarnya untuk sampel ke-i
$\widehat{y_i}$ = kelas yang diprediksi untuk sampel ke-i
$|\cdot|$ = nilai absolut

Nilai *MAE* yang lebih rendah menunjukkan bahwa prediksi *model* lebih dekat dengan kelas sebenarnya. Misalnya, memprediksi kelas 2 untuk sampel kelas 3 (*MAE* = 1) lebih baik daripada memprediksi kelas 1 (*MAE* = 2), karena struktur *ordinal* data dipertimbangkan.

h) *Quadratic weighted kappa (QWK)*

Quadratic weighted kappa mengukur kesepakatan antara prediksi *model* dengan label sebenarnya, dengan memberikan penalti lebih besar untuk kesalahan yang lebih jauh dalam skala *ordinal*. QWK sangat sesuai untuk tugas klasifikasi dengan data yang bersifat *ordinal* karena mempertimbangkan tingkat keparahan kesalahan. QWK dihitung menggunakan Rumus 4.8:

$$\text{QWK} = 1 - \frac{\sum_{i,j} w_{ij} O_{ij}}{\sum_{i,j} w_{ij} E_{ij}} \tag{4.8}$$

Keterangan:
QWK = *Quadratic weighted kappa*
O_{ij} = jumlah observasi pada sel (i, j) dari *Confusion* matrix (kelas sebenarnya i, prediksi j)
E_{ij} = jumlah ekspektasi pada sel (i, j) jika prediksi acak
w_{ij} = bobot untuk sel (i, j), dihitung sebagai:

$$w_{ij} = \frac{(i-j)^2}{(K-1)^2} \tag{4.8a}$$

dengan K adalah jumlah kelas.

Quadratic weighted kappa (QWK) mengukur tingkat kesepakatan antara label prediksi dan label sebenarnya dengan mempertimbangkan sifat *ordinal* antar kelas. Nilai QWK = 1 menunjukkan kesepakatan sempurna, QWK = 0 berarti kesepakatan setara dengan prediksi acak, sedangkan QWK < 0 menandakan kesepakatan lebih buruk daripada acak. Keunggulan QWK adalah pemberian penalti kuadratik terhadap besarnya kesalahan, sehingga kesalahan yang jauh akan dihukum lebih berat, misalnya memprediksi kelas 1 untuk sampel yang sebenarnya kelas 4 akan mendapat penalti jauh lebih besar dibanding memprediksi kelas 3. Karena itu, QWK sangat relevan untuk deteksi tingkat keparahan depresi, sebab kesalahan prediksi yang besar antar level dapat menghasilkan implikasi interpretasi yang lebih serius dibanding kesalahan yang hanya bergeser satu tingkat.

4.1.4 Uji Signifikansi Statistik (*Paired t-test*)

Untuk memastikan bahwa perbedaan kinerja antar skenario *model* bukan terjadi karena kebetulan akibat variasi lipatan data (*folds*), penelitian ini menggunakan uji signifikansi statistik berupa uji-t berpasangan (*Paired sample t-test*). Uji berpasangan ini dipilih karena metrik evaluasi yang dibandingkan berasal

dari *model* berbeda yang dievaluasi pada sampel data uji yang sama pada setiap iterasi *Stratified k-fold cross validation*.

Paired t-test mengevaluasi apakah rata-rata selisih metrik (seperti Akurasi, *Macro F1*, atau QWK) dari dua skenario *model* yang berbeda memiliki perbedaan yang bermakna secara statistik. Perhitungan nilai t dirumuskan sebagai rumus 4.9 berikut:

$$t = \frac{\bar{d}}{S_d/\sqrt{n}} \tag{4.9}$$

Keterangan:
t = nilai t-hitung
$\bar{d}$ = rata-rata dari selisih nilai metrik antar dua *model* pada setiap *fold*
S_d = standar deviasi dari selisih nilai metrik tersebut
n = jumlah pasangan data pengujian (dalam penelitian ini $n = 5$, sesuai dengan jumlah *folds*)

Pengambilan keputusan dalam uji signifikansi ini didasarkan pada nilai probabilitas (*p-value*) dengan taraf signifikansi α sebesar 0,05 atau 5%. Pengujian ini menggunakan rumusan Hipotesis Nol (H_0) yang berasumsi bahwa tidak ada perbedaan performa yang signifikan antara kedua skenario model, serta Hipotesis Alternatif (H_1) yang menyatakan kebalikannya, yakni terdapat perbedaan performa yang signifikan di antara kedua skenario tersebut. Berdasarkan rumusan tersebut, jika nilai *p-value* lebih kecil dari batas α, maka H_0 akan ditolak yang berarti perbedaan rata-rata metrik evaluasi antara kedua model terbukti nyata dan signifikan secara statistik, serta bukan sekadar kebetulan acak yang muncul dari proses pembagian data latih dan data uji.

4.2 Hasil Uji Coba

Dataset yang digunakan dalam penelitian ini berjumlah 3.530 data bersih. Mengingat ukuran data teks yang relatif kecil untuk standar *deep learning*, arsitektur *model* disesuaikan guna menghindari terjadinya masalah penghafalan data latih (*overfitting*). Berdasarkan pertimbangan batasan memori GPU (VRAM 8 GB) serta uji coba empiris, kapasitas jaringan diturunkan secara konservatif. Lapisan BiGRU dan lapisan *Dense* masing-masing disederhanakan menjadi 1 lapis tunggal dengan 128 unit tersembunyi. Pengaturan lengkap parameter arsitektur dan pelatihan *model* disajikan pada Tabel 4.1.

Tabel 4.1 Konfigurasi Parameter Pelatihan *Model* (Hyperparameter)

Kategori	Parameter	Nilai / Konfigurasi
Pengaturan Optimasi	Learning Rate Awal	0.0001 (1e-4)
	Pengoptimal (Optimizer)	Adam
	Ukuran Batch (Batch Size)	32
	Batas *Epoch* Maksimal	150 *Epoch*
Fungsi Kerugian (*Loss*)	Combined *Ordinal Loss*	Focal: 1.0, BCE: 0.3, Penalty: 0.2

Terdapat beberapa keputusan teknis penting yang dicerminkan dalam Tabel 4.1. Pertama, penetapan panjang maksimal (*max_length*) sebesar 200 token didasarkan pada distribusi rentang token teks mayoritas, di mana kelebihan kata akan dipotong dari belakang (*post-truncating*) dan kekurangan kata disisipi nilai nol (*post-padding*). Kedua, algoritma dilatih menggunakan presisi campuran otomatis (*Automatic mixed Precision* / AMP) untuk mempercepat komputasi GPU tanpa mengorbankan stabilitas arah pembelajaran (*gradient*). Ketiga, penerapan *Early stopping* difungsikan untuk menghentikan proses iterasi secara otomatis

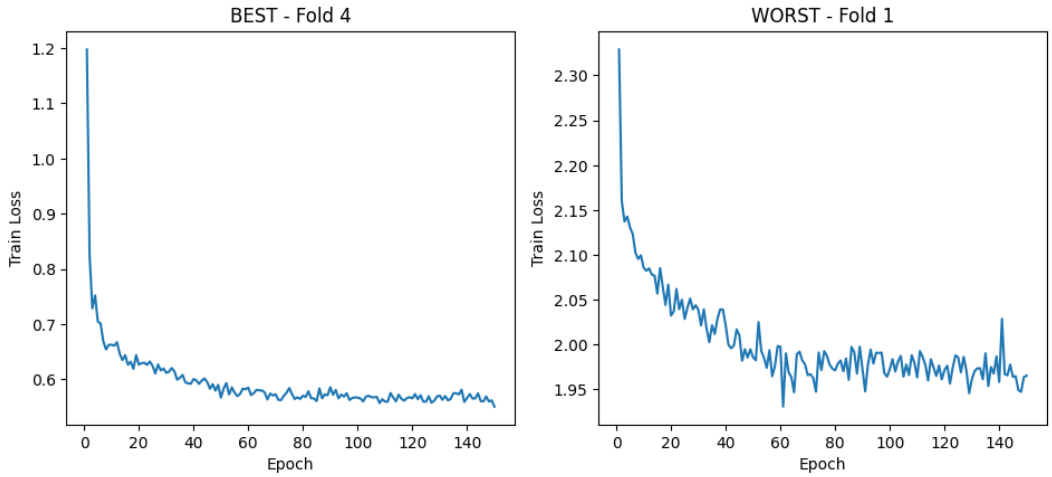
apabila *model* tidak menunjukkan peningkatan performa pada data validasi selama 10 iterasi (*epoch*) berturut-turut, sehingga memastikan *model* menyimpan bobot paling optimal.

Pengujian *model* dengan konfigurasi di atas dieksekusi melalui kerangka studi ablasi (*ablation study*) yang dirancang untuk mengisolasi dan mengukur kontribusi dari setiap komponen rekayasa arsitektur. Pendekatan ini membagi eksperimen ke dalam empat skenario utama yang dievaluasi secara independen menggunakan satu himpunan data uji (*test set*) yang sama, yakni terdiri dari 530 sampel kalimat. Data uji ini merepresentasikan tantangan riil dari distribusi data di lapangan yang sangat timpang, di mana kelas *minimum* mendominasi dengan 385 sampel, sementara representasi kondisi klinis yang lebih serius secara berurutan hanya memiliki 44 sampel untuk kelas *mild*, 59 sampel untuk *moderate*, dan 42 sampel untuk *severe*. Evaluasi yang mengandalkan kondisi ketidakseimbangan absolut ini sengaja dipertahankan untuk menguji ketahanan (*robustness*) sistem pemrosesan bahasa alami dalam kondisi dunia nyata.

4.2.1 Kinerja Skenario *Model* Dasar (*Baseline*)

Skenario pertama dirancang sebagai eksperimen kontrol untuk menetapkan tolok ukur kinerja *model* BiGRU standar pada *dataset* yang memiliki ketidakseimbangan kelas ekstrem. Pada konfigurasi ini, *model* tidak menggunakan mekanisme tambahan apapun, tidak ada penerapan pembobotan kelas (*class weighting*) pada fungsi kerugian maupun mekanisme atensi (*Attention Mechanism*) untuk seleksi fitur. *Model* dilatih murni untuk meminimalkan kesalahan prediksi berdasarkan frekuensi kemunculan data. Tujuan utama skenario ini adalah untuk

mengidentifikasi bias alami *model* ketika dihadapkan pada data mayoritas yang dominan, serta membuktikan apakah arsitektur dasar mampu menangkap pola pada kelas minoritas tanpa intervensi khusus.



Gambar 4.2 Grafik Training *Loss* Skenario *Attention Only*
(Kiri: *Fold* 0 - Best, Kanan: *Fold* 3 - Worst)

Dinamika penurunan *loss model Baseline* yang divisualisasikan pada Gambar 4.2 menunjukkan kontras performa antara *Fold* 4 (Best) dan *Fold* 1 (Worst), di mana pada *Fold* 4 terlihat kurva konvergensi yang sangat sehat dengan penurunan *loss* konsisten dari nilai awal 1.1974 hingga mencapai titik terendah 0.5504 pada *epoch* 87. Penurunan total sebesar 0.647 poin ini mengindikasikan keberhasilan *model* dalam mengekstrak fitur dominan dari data latih secara efektif tanpa hambatan berarti. Sebaliknya, pada *Fold* 1 terlihat gejala kesulitan belajar yang signifikan dengan *loss* awal sangat tinggi sebesar 2.3288 yang hanya mampu turun sekitar 0.38 poin ke angka 1.9635 setelah 86 *epoch* pelatihan. Kegagalan *model* pada *Fold* 1 untuk turun di bawah angka 1.9456 menunjukkan bahwa *model* terjebak dalam *local minima* yang buruk, sehingga gagal menemukan pola yang

dapat digeneralisasi dan cenderung hanya melakukan prediksi secara acak atau terjebak pada bias kelas mayoritas.

Tabel 4.2 Hasil Evaluasi Statistik Skenario 1 (*Baseline*)

Fold	**Accuracy**	***Macro Precision***	***Macro Recall***	***Macro* F1**	**Weighted F1**	***MAE***	**QWK**
1	71.89%	25.40%	32.69%	28.50%	64.83%	54.72%	35.96%
2	69.43%	25.23%	34.50%	28.33%	64.45%	62.45%	37.00%
3	72.64%	18.16%	25.00%	21.04%	61.13%	54.34%	0.00%
4	70.19%	25.51%	34.76%	28.82%	64.56%	60.57%	36.48%
5	73.21%	28.19%	28.07%	26.25%	63.98%	51.70%	11.83%
Mean	71.47%	24.50%	31.00%	26.59%	63.79%	56.76%	24.25%
Std Dev	1.61%	3.75%	4.30%	3.26%	1.52%	4.54%	17.26%

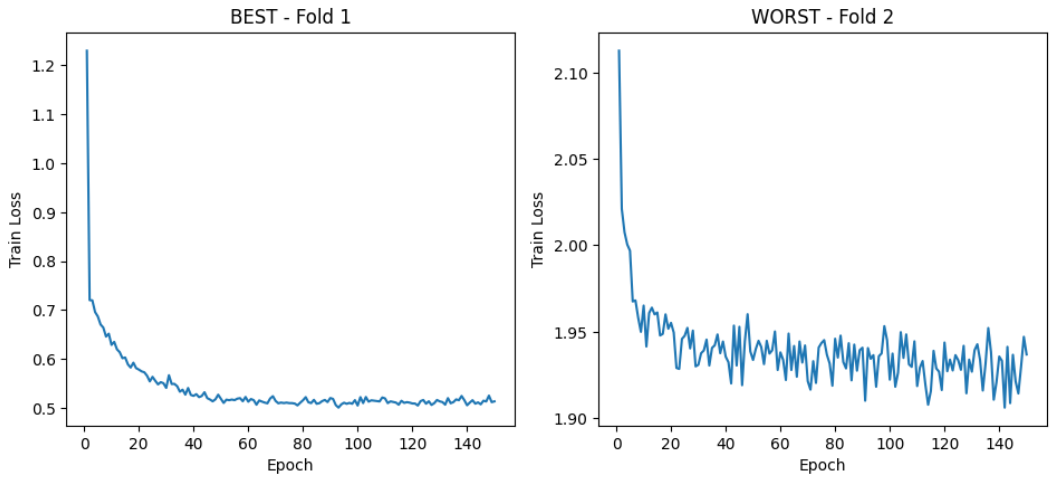
Analisis kuantitatif pada Tabel 4.2 menyoroti fenomena "paradoks akurasi" yang lazim terjadi pada kasus klasifikasi dengan sebaran data yang tidak seimbang. Meskipun *model Baseline* mencatatkan rata-rata akurasi yang terlihat cukup tinggi sebesar 71,47%, angka ini bersifat kurang representatif karena tidak merepresentasikan kemampuan *model* yang sebenarnya dalam membedakan antar-kelas. Kelemahan ini terungkap secara jelas melalui metrik pendukung lainnya, seperti rata-rata *Macro Precision* (24,50%), *Macro Recall* (31,00%), dan *Macro F1* (26,59%) yang sangat rendah. Angka-angka tersebut mengindikasikan bahwa tingginya akurasi semata-mata didorong oleh kecenderungan *model* untuk terus menebak kelas mayoritas, sementara ia gagal mengidentifikasi atau membedakan kelas minoritas dengan baik.

Ketimpangan performa ini semakin dipertegas oleh tingginya variabilitas antar-*fold*, yang menunjukkan ketidakstabilan *model* saat dihadapkan pada

potongan data yang berbeda. Hal ini paling mencolok terlihat pada nilai *Quadratic weighted kappa* (QWK) yang rata-ratanya hanya mencapai 24,25%. Pada pengujian *Fold* 3, *model* bahkan mencatatkan nilai QWK anjlok hingga 0,00%, yang berarti kesepakatan prediksi *model* sama sekali tidak lebih baik daripada tebakan acak. Standar deviasi QWK yang sangat besar (17,26%) menegaskan bahwa tanpa adanya perlakuan penyeimbang data (*data balancing*), *model Baseline* ini sangat sensitif terhadap komposisi data latih dan cenderung belum dapat diandalkan untuk skenario penerapan praktis.

4.2.2 Kinerja Skenario Penerapan Pembobotan Kelas (*Class Weight Only*)

Skenario kedua menerapkan intervensi berupa pembobotan kelas (*Class weighting*) pada fungsi *loss*. Strategi ini bertujuan untuk memitigasi dominasi kelas mayoritas dengan memberikan penalti yang lebih besar ketika *model* salah memprediksi kelas minoritas (*Mild, Moderate, Severe*). Secara teoritis, hal ini akan memaksa gradien optimasi untuk bergerak ke arah yang memperbaiki performa pada kelas-kelas kecil, meskipun risiko kesalahan pada kelas mayoritas mungkin sedikit meningkat. Eksperimen ini menguji apakah modifikasi fungsi kerugian saja sudah cukup untuk membangun *model* klasifikasi *ordinal* yang *robust* tanpa mengubah arsitektur jaringan saraf.



Gambar 4.3 Grafik Training *Loss* Skenario *Class Weight Only*
(Kiri: *Fold* 1 - Best, Kanan: *Fold* 2 - Worst)

Analisis grafik *loss* pada Gambar 4.3 memperlihatkan dampak langsung dari penambahan bobot penalti pada fungsi objektif, di mana pada *Fold* 1 (Best) kurva *loss* berhasil ditekan dari angka awal 1.2298 hingga mencapai titik terendah 0.5006 sebelum berakhir di 0.5134 pada *epoch* 88. Penurunan signifikan sebesar 0.7164 poin ini lebih dalam dibandingkan *model Baseline*, yang menunjukkan bahwa pembobotan kelas efektif memaksa *model* belajar lebih keras guna meminimalkan kesalahan pada kelas minoritas melalui proses belajar yang stabil dan gradual. Sebaliknya, pada *Fold* 2 (Worst), kurva *loss* dimulai dari angka yang sangat tinggi sebesar 2.1127 dan berakhir stagnan di angka 1.9470 dengan nilai *minimum* hanya mencapai 1.9060, mencerminkan kemajuan minimal sebesar 0.2 poin saja. Stagnasi pada nilai *loss* yang tinggi ini mengindikasikan bahwa pada lipatan data tersebut, *model* gagal menemukan kompromi antara memprediksi kelas mayoritas dan minoritas sehingga terjebak pada performa sub-optimal yang tidak mampu mengalami perkembangan berarti selama pelatihan.

Tabel 4.3 Hasil Evaluasi Statistik Skenario 2 (*Class Weight Only*)

Fold	**Accuracy**	***Macro Precision***	***Macro Recall***	***Macro* F1**	**Weighted F1**	***MAE***	**QWK**
1	68.49%	29.44%	36.24%	30.41%	65.07%	62.08%	40.19%
2	69.43%	39.72%	39.28%	38.93%	68.12%	55.28%	36.31%
3	72.64%	18.16%	25.00%	21.04%	61.13%	54.34%	0.00%
4	69.81%	39.84%	36.84%	35.92%	67.14%	57.74%	33.79%
5	71.13%	42.37%	37.31%	36.28%	68.03%	53.02%	38.02%
Mean	70.30%	33.91%	34.93%	32.52%	65.90%	56.49%	29.66%
Std Dev	1.62%	10.11%	5.67%	7.12%	2.93%	3.57%	16.75%

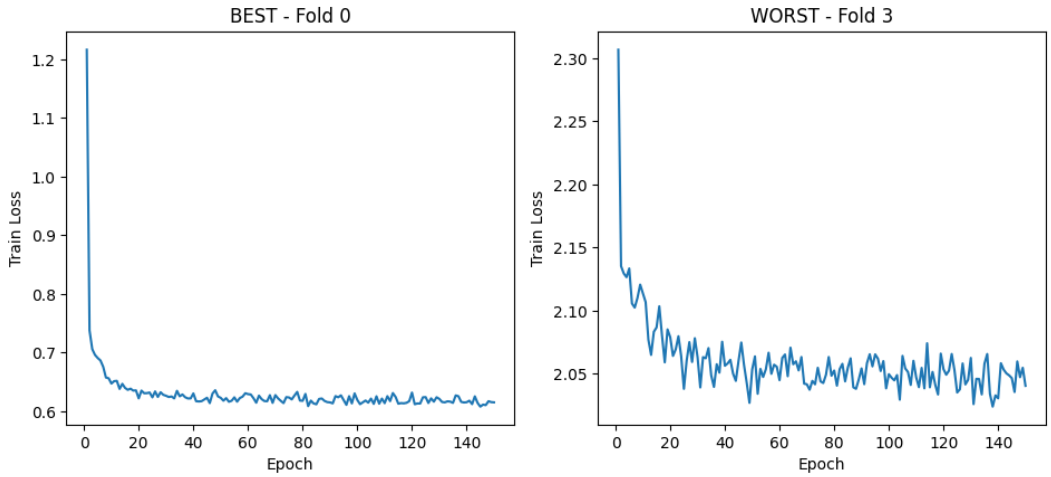
Dampak positif dari penerapan pembobotan kelas (*class weight*) terlihat cukup jelas pada Tabel 4.3. Meskipun rata-rata akurasi global mengalami sedikit penurunan menjadi 70,30% (turun dari 71,47% pada skenario *Baseline*), metrik-metrik yang lebih sensitif terhadap penanganan keseimbangan kelas justru menunjukkan tren peningkatan. Hal ini dibuktikan dengan naiknya rata-rata *Macro F1* menjadi 32,52% dan *Quadratic weighted kappa* (QWK) yang merangkak naik ke angka 29,66%. Peningkatan pada metrik-metrik krusial ini mengonfirmasi bahwa sedikit penurunan pada akurasi global merupakan kompromi yang sangat sepadan demi mendapatkan kemampuan deteksi dan pengenalan kelas minoritas yang jauh lebih baik.

Beberapa *fold* bahkan menunjukkan peningkatan performa yang cukup menjanjikan, seperti pada *Fold* 1 dan *Fold* 5 yang mencatatkan nilai QWK di kisaran 38% hingga 40%, menempatkannya dalam kategori kesepakatan sedang (*moderate agreement*). Kendati demikian, tantangan terkait stabilitas *model* masih belum terpecahkan secara menyeluruh. Hal ini terbukti dari hasil pada *Fold* 3 yang

kembali gagal mengidentifikasi pola dengan baik dan mencatatkan QWK 0,00%, sama seperti performa terburuk di skenario sebelumnya. Standar deviasi yang masih cukup tinggi pada metrik *Macro Precision* (10,11%) dan QWK (16,75%) menandakan bahwa variasi pada data latih masih memberikan pengaruh yang signifikan terhadap keandalan prediksi *model* akhir.

4.2.3 Kinerja Skenario Penerapan Mekanisme Atensi (*Attention Only*)

Skenario ketiga mengisolasi peran mekanisme atensi (*Attention Mechanism*) tanpa bantuan pembobotan kelas. Hipotesis yang diuji adalah apakah kemampuan *model* untuk memberikan bobot kepentingan yang berbeda pada setiap kata secara otomatis dapat memilah fitur relevan untuk kelas minoritas, tanpa perlu memanipulasi fungsi *loss*. Pada skenario ini, lapisan *Global Average Pooling* digantikan oleh lapisan atensi yang memungkinkan agregasi fitur secara adaptif.



Gambar 4.4 Grafik Training *Loss* Skenario *Attention Only*
(Kiri: *Fold* 0 - Best, Kanan: *Fold* 3 - Worst)

Analisis grafik *loss* pada Gambar 4.4 menyingkap fenomena menarik mengenai mekanisme atensi tanpa panduan bobot kelas, di mana pada *Fold* 0 (Best)

terlihat penurunan visual yang mulus dari 1.2161 ke 0.6158, namun angka akhir tersebut relatif lebih tinggi dibanding *best loss* pada skenario lain (~0.50) dan cenderung cepat mendatar (*plateau*). Kondisi ini mengindikasikan bahwa *model* dengan cepat menemukan strategi "aman" untuk meminimalkan *loss* rata-rata dengan memprediksi semua sampel sebagai kelas mayoritas alih-alih mempelajari fitur kompleks untuk membedakan kelas. Sebaliknya, pada *Fold* 3 (Worst) terjadi kegagalan konvergensi total yang ditunjukkan dengan *loss* awal sangat tinggi sebesar 2.3067 dan tetap bertahan tinggi di angka 2.0547 tanpa pernah turun di bawah 2.02 sepanjang pelatihan. Kegagalan ini menandakan bahwa pada lipatan data tersebut, *model* sama sekali tidak mampu menemukan pola yang konsisten, baik untuk pola mayoritas maupun minoritas, sehingga performa tetap stagnan pada nilai *error* yang besar.

Tabel 4.4 Hasil Evaluasi Statistik Skenario 3 (*Attention Only*)

Fold	**Accuracy**	***Macro Precision***	***Macro Recall***	***Macro* F1**	**Weighted F1**	***MAE***	**QWK**
1	72.64%	18.16%	25.00%	21.04%	61.13%	54.34%	0.00%
2	72.64%	18.16%	25.00%	21.04%	61.13%	54.34%	0.00%
3	72.64%	18.16%	25.00%	21.04%	61.13%	54.34%	0.00%
4	72.45%	24.27%	28.65%	26.04%	63.58%	53.77%	23.41%
5	72.64%	18.16%	25.00%	21.04%	61.13%	54.34%	0.00%
Mean	72.60%	19.38%	25.73%	22.04%	61.62%	54.23%	4.68%
Std Dev	0.08%	2.73%	1.63%	2.24%	1.10%	0.25%	10.47%

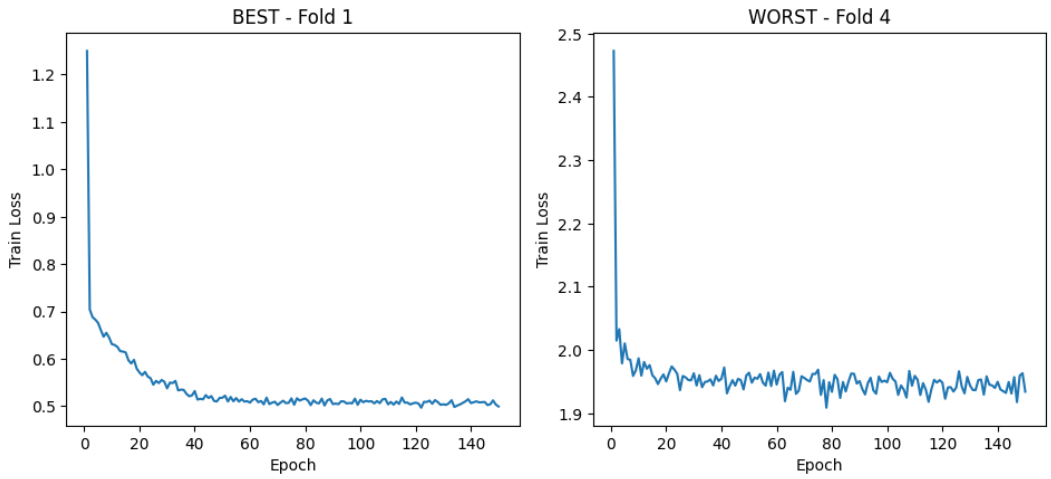
Hasil evaluasi pada Tabel 4.4 menempatkan skenario *Attention Only* sebagai *model* dengan performa terburuk dalam hal kemampuan klasifikasi *ordinal* secara keseluruhan. Meskipun rata-rata akurasinya tampak paling tinggi (72,60%) dan

sangat stabil (standar deviasi hanya 0,08%), indikator penanganan kelas minoritas justru menunjukkan kegagalan fatal. Hal ini dibuktikan oleh rata-rata metrik *Quadratic weighted kappa* (QWK) yang mengalami penurunan drastis ke angka 4,68%. Kejatuhan ini sangat signifikan karena pada empat dari lima lipatan pengujian (*Fold* 1, 2, 3, dan 5), *model* secara absolut mencatatkan nilai QWK tepat 0,00%, yang berarti prediksinya tidak memiliki nilai kesepakatan sama sekali jika dibandingkan dengan klasifikasi acak.

Fakta tersebut secara kuat menggugurkan hipotesis bahwa mekanisme atensi (*attention*) sudah cukup untuk menangani masalah ketidakseimbangan data secara mandiri. Alih-alih membantu, ketiadaan teknik penyeimbang data (*data balancing* atau *class weight*) justru membuat mekanisme atensi memperkuat bias yang ada pada data latih. *Model* belajar untuk mengalokasikan fokus perhatiannya hanya pada fitur-fitur yang paling dominan dan sering muncul (milik kelas mayoritas), sementara sinyal-sinyal unik yang krusial dari kelas minoritas sepenuhnya diabaikan karena dianggap oleh *model* sebagai *noise* belaka.

4.2.4 Kinerja Skenario *Model* Lengkap (*Full Model*)

Skenario terakhir menggabungkan kekuatan dari *Class weighting* dan *Attention Mechanism*. Pendekatan hibrida ini dirancang untuk menciptakan sinergi: pembobotan kelas bertugas menjaga gradien agar tidak bias ke mayoritas, sementara mekanisme atensi bertugas mengekstraksi fitur kontekstual yang paling relevan untuk membedakan tingkat keparahan. Skenario ini merepresentasikan konfigurasi *model* paling kompleks yang diharapkan dapat memberikan performa terbaik.



Gambar 4.5 Grafik Training *Loss* Skenario *Full Model*
(Kiri: *Fold* 1 - Best, Kanan: *Fold* 4 - Worst)

Visualisasi pada Gambar 4.5 menunjukkan potensi terbaik dan risiko terbesar dari *model* kompleks ini, di mana pada *Fold* 1 (Best) terlihat proses pembelajaran ideal dengan kurva yang menurun tajam secara konsisten dari *loss* 1.2495 hingga menyentuh angka 0.5035, dengan nilai *minimum* tercatat sebesar 0.4969. Penurunan drastis sebesar 0.7526 poin ini merupakan yang terbesar dibandingkan semua skenario lain, menandakan bahwa kombinasi atensi dan bobot kelas memungkinkan *model* untuk meminimalkan kesalahan secara agresif dan efisien. Sebaliknya, pada *Fold* 4 (Worst) terjadi kegagalan konvergensi parah yang dimulai dari *loss* sangat tinggi sebesar 2.4724 dan berakhir tertahan di angka 1.9633. Meskipun *model* berjuang menurunkan *error*, nilai *minimum* yang dicapai pun masih tetap tinggi di angka 1.9089, yang menunjukkan bahwa pada inisialisasi tertentu, kompleksitas tambahan dari lapisan atensi justru mempersulit *model* untuk menemukan landaian gradien yang tepat sehingga terjebak di area *loss* yang tinggi.

Tabel 4.5 Hasil Evaluasi Statistik Skenario 4 (*Full Model*)

Fold	**Accuracy**	***Macro Precision***	***Macro Recall***	***Macro* F1**	**Weighted F1**	***MAE***	**QWK**
1	70.75%	24.93%	32.84%	28.03%	64.60%	58.68%	35.26%
2	71.70%	39.17%	37.26%	37.47%	69.00%	50.19%	39.69%
3	72.64%	18.16%	25.00%	21.04%	61.13%	54.34%	0.00%
4	69.62%	37.51%	38.74%	37.54%	67.76%	58.68%	33.62%
5	72.64%	18.16%	25.00%	21.04%	61.13%	54.34%	0.00%
Mean	71.47%	27.59%	31.77%	29.02%	64.72%	55.25%	21.71%
Std Dev	1.30%	10.22%	6.55%	8.25%	3.65%	3.56%	19.95%

Evaluasi statistik pada Tabel 4.5 memperlihatkan bahwa *Full Model* memiliki potensi performa puncak tertinggi jika dibandingkan dengan seluruh skenario pengujian lainnya. Pada kondisi optimalnya, yakni saat pengujian di *Fold* 2, *model* ini berhasil mencetak rekor *Quadratic weighted kappa* (QWK) tertinggi di angka 39,69% serta *Weighted F1* sebesar 69,00%. Capaian impresif ini sukses melampaui batas performa terbaik yang dihasilkan oleh skenario *Baseline* maupun *Class Weight Only*. Fakta ini membuktikan bahwa penggabungan komprehensif antara teknik pembobotan kelas dan mekanisme atensi mampu memicu daya prediktif yang sangat tajam ketika *model* mendapatkan kondisi pelatihan yang ideal.

Meskipun memiliki potensi puncak yang luar biasa, rata-rata performa keseluruhan *model* ini justru terseret turun oleh masalah stabilitas yang cukup fatal. Kelemahan ini terlihat jelas dari terjadinya *collapse* pada *Fold* 3 dan *Fold* 5, di mana nilai QWK anjlok drastis menyentuh angka 0,00%. Besarnya standar deviasi pada metrik QWK yang mencapai 19,95% menjadi bukti nyata bahwa *Full Model* memiliki varian rentang kinerja yang paling lebar dan tidak konsisten. Karakteristik

ini menunjukkan bahwa *model* memiliki sensitivitas yang sangat tinggi. Ia bisa bekerja dengan sangat cerdas mengenali kelas minoritas, atau justru gagal total bergantung pada proses inisialisasi awal serta distribusi spesifik dari data latih yang diberikan pada tiap lipatan.

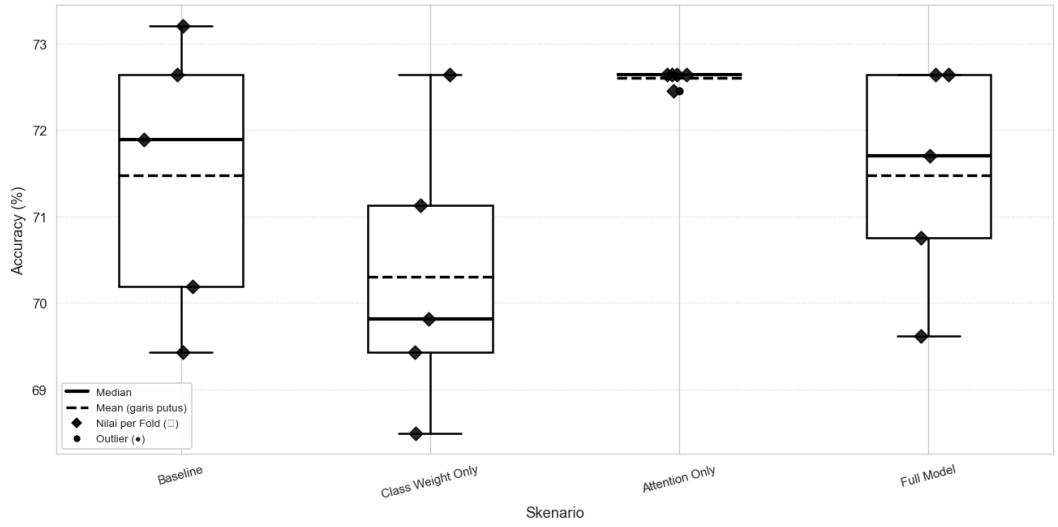
4.3 Pembahasan

Bagian ini menyajikan analisis komprehensif terhadap temuan eksperimen dari empat skenario arsitektur yang telah diuji, yaitu *Baseline*, *Class Weight Only*, *Attention Only*, dan *Full Model*. Pembahasan ini dirancang untuk memberikan gambaran yang utuh mengenai performa *model* dengan tidak hanya meninjau nilai rata-rata semata, melainkan juga menilai konsistensi kinerja antar-lipatan (*fold*) melalui distribusi statistik pada *boxplot*. Selain itu, guna memastikan bahwa perbedaan performa yang muncul bukan merupakan faktor kebetulan, dilakukan uji signifikansi statistik menggunakan *Paired sample t-test*. Melalui alur analisis ini, kesimpulan yang ditarik tidak hanya didasarkan pada perolehan skor tertinggi, tetapi juga mempertimbangkan stabilitas *model* serta kekuatan bukti statistik pada setiap metrik evaluasi yang digunakan.

4.3.1 Analisis Akurasi (*Accuracy*)

Akurasi merupakan metrik fundamental yang mengukur proporsi prediksi benar terhadap keseluruhan jumlah data uji. Dalam konteks penelitian ini, metrik tersebut dijadikan indikator awal karena sifatnya yang langsung dan mudah ditafsirkan. Namun, perlu ditekankan bahwa pada kasus klasifikasi dengan ketidakseimbangan kelas yang ekstrem seperti pada dataset Reddit ini, nilai akurasi

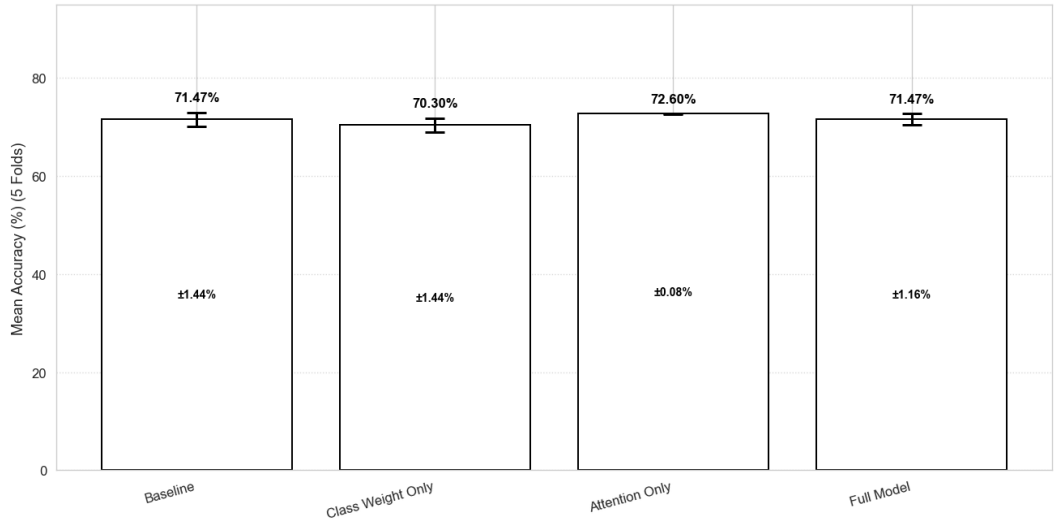
dapat bersifat manipulatif karena *model* cenderung terlihat unggul hanya dengan memprediksi kelas mayoritas secara seragam tanpa benar-benar mampu mengenali karakteristik kelas minoritas. Oleh karena itu, analisis akurasi di bawah ini ditinjau dari berbagai sudut pandang sebaran dan signifikansi.



Gambar 4.6 *Boxplot* Distribusi Akurasi pada Empat Skenario

Berdasarkan visualisasi *boxplot* pada Gambar 4.6, skenario *Baseline* menunjukkan performa akurasi yang relatif stabil meskipun masih memiliki variasi antar-lipatan dengan nilai rata-rata sebesar 71,47% ± 1,44%. Skenario *Class Weight Only* memperlihatkan penurunan akurasi menjadi 70,30% ± 1,44%, yang mengindikasikan bahwa penerapan pembobotan kelas memaksa *model* untuk mengalihkan fokus komputasinya sehingga tidak terlalu menguntungkan kelas mayoritas, yang berdampak pada berkurangnya akurasi global demi mengejar keadilan prediksi. Di sisi lain, skenario *Attention Only* mencatatkan rata-rata akurasi tertinggi sebesar 72,60% ± 0,08% dengan sebaran yang sangat sempit, menandakan konsistensi ekstrem antar-lipatan. Namun, pola yang sangat sempit pada konteks data yang tidak seimbang ini mencerminkan adanya strategi prediksi

yang sangat seragam atau bias terhadap kelas dominan. Terakhir, *Full Model* menghasilkan nilai rata-rata yang identik dengan *Baseline* (71,47% ± 1,16%), tetapi dengan variasi yang lebih kecil, mengisyaratkan bahwa kombinasi metode pada skenario ini lebih berperan dalam memperbaiki stabilitas performa daripada meningkatkan skor rata-rata secara nominal.



Gambar 4.7 Grafik Batang Rata-rata (*Mean*) ± Std Akurasi pada Empat Skenario

Peringkat performa akurasi secara visual dapat diamati pada Gambar 4.7, di mana *Attention Only* menempati posisi pertama dengan kestabilan yang sangat tinggi di hampir seluruh lipatan pengujian. Meskipun skenario tersebut memimpin secara kuantitatif, tingginya akurasi ini merupakan bentuk "paradoks akurasi" karena *model* sebenarnya gagal menangkap sinyal-sinyal unik dari kelas minoritas. *Baseline* dan *Full Model* berbagi angka rata-rata yang sama, namun *Full Model* unggul dalam aspek konsistensi karena memiliki simpangan baku yang lebih rendah (± 1,16%) dibandingkan *Baseline* (± 1,44%). Skenario *Class Weight Only* menjadi yang terendah secara rata-rata, yang mana hal ini selaras dengan prinsip bahwa

pembobotan kelas sering kali mengorbankan akurasi global sebagai kompromi untuk mendapatkan kemampuan deteksi kelas minoritas yang lebih baik pada tahap evaluasi selanjutnya.

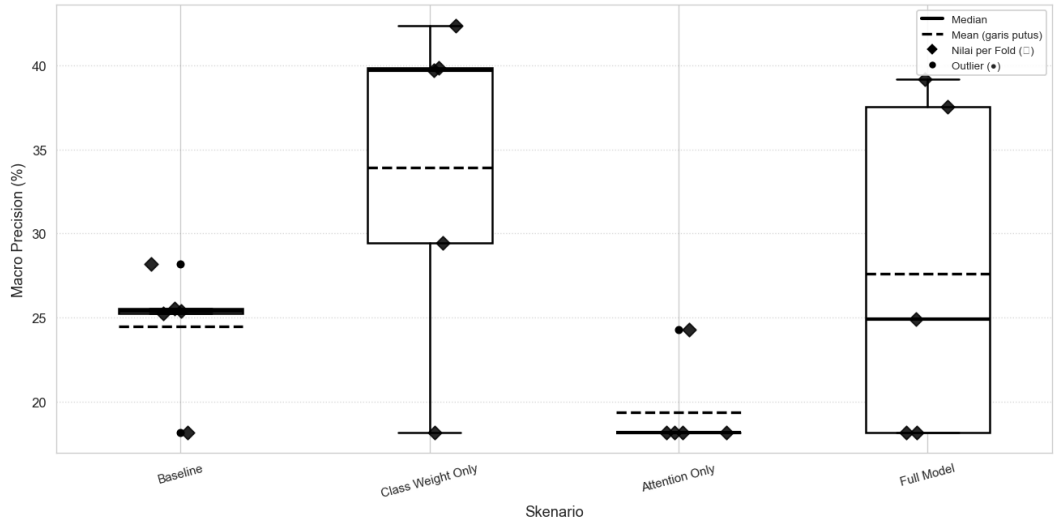
Tabel 4.6 Hasil Uji Signifikansi *Paired t-test* pada Metrik Akurasi

Perbandingan	**t**	**p-value**	**Keterangan**
Baseline vs *Class Weight Only*	+1.7303	0.158636	ns (tidak signifikan)
Baseline vs *Attention Only*	-1.6093	0.182830	ns (tidak signifikan)
Baseline vs *Full Model*	+0.0000	1.000000	ns (tidak signifikan)
Class Weight Only vs *Attention Only*	-3.2105	0.032573	* (signifikan)
Class Weight Only vs *Full Model*	-2.1866	0.094051	ns (tidak signifikan)
Attention Only vs *Full Model*	+2.0580	0.108701	ns (tidak signifikan)

Setelah melakukan pengujian statistik yang dirangkum pada Tabel 4.6, ditemukan bahwa sebagian besar pasangan perbandingan tidak menunjukkan perbedaan yang signifikan secara statistik ($p > 0,05$). Hal ini mengonfirmasi bahwa variasi nilai yang terlihat pada sebagian besar skenario hanyalah fluktuasi antar-lipatan data yang wajar. Satu-satunya perbedaan yang terbukti nyata dan signifikan adalah pada perbandingan antara *Class Weight Only* dan *Attention Only* ($p = 0,032573$), yang menegaskan secara empiris bahwa *Attention Only* memang menghasilkan strategi prediksi yang secara statistik berbeda dan lebih tinggi pada metrik akurasi dibandingkan *Class Weight Only*. Sementara itu, tidak adanya perbedaan signifikan antara *Baseline* dan *Full Model* ($p = 1,000000$) semakin memperkuat argumen bahwa penambahan komponen kompleksitas pada arsitektur hibrida tersebut belum mampu memberikan lonjakan performa akurasi yang stabil melampaui *model* dasarnya.

4.3.2 Analisis *Macro Precision*

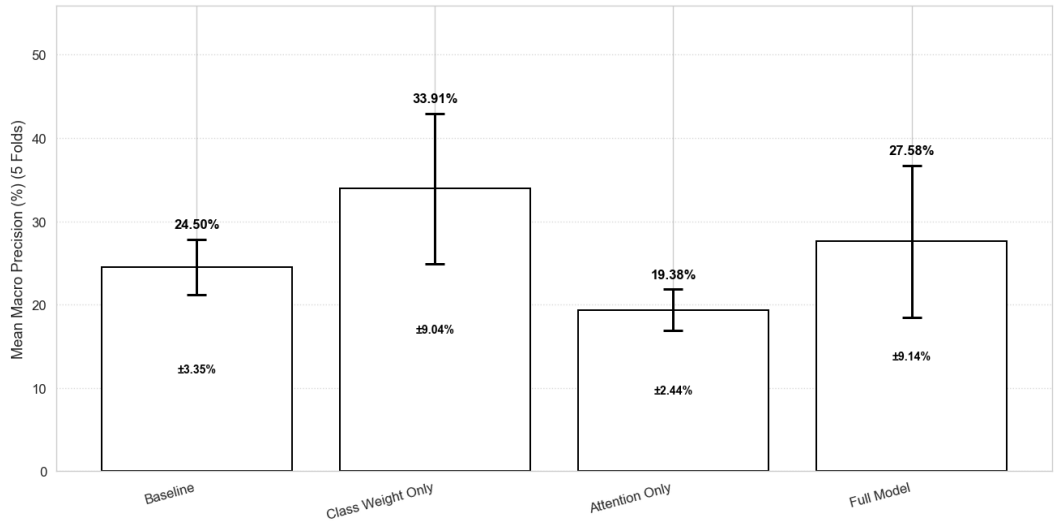
Macro Precision merupakan metrik yang mengevaluasi ketepatan prediksi *model* dengan memberikan bobot yang setara pada setiap kategori, sehingga performa pada kelas minoritas memiliki kontribusi yang sama besarnya dengan kelas mayoritas terhadap skor akhir. Metrik ini menjadi instrumen krusial untuk mengukur tingkat "kejujuran" *model* dalam melakukan klasifikasi, apakah *model* benar-benar mampu mengenali karakteristik spesifik dari setiap tingkat keparahan depresi, atau sekadar mengandalkan dominasi frekuensi pada kelas mayoritas. Dalam skenario data yang tidak seimbang (*imbalanced data*), *macro Precision* bertindak sebagai indikator deteksi *false positive*, terutama ketika *model* mulai didorong untuk lebih agresif dalam memprediksi kelas-kelas minoritas.



Gambar 4.8 *Boxplot* Distribusi *Macro Precision* pada Empat Skenario

Eksplorasi terhadap distribusi *macro Precision* melalui representasi *boxplot* pada Gambar 4.8 menyingkap variabilitas performa yang signifikan antar-skenario. Skenario *Baseline* menunjukkan rata-rata sebesar 24,50% ± 3,35% dengan rentang antarkuartil (IQR) yang sangat sempit (0,0028), yang menandakan bahwa mayoritas

lipatan (*fold*) terkonsentrasi pada nilai yang rapat namun tetap rentan terhadap penurunan tajam pada komposisi data uji tertentu. Kontras dengan hal tersebut, *Class Weight Only* berhasil mencatatkan kenaikan rata-rata tertinggi menjadi 33,91% ± 9,04%, namun diikuti dengan sebaran yang jauh lebih lebar (IQR 0,1040), mengindikasikan bahwa meskipun strategi pembobotan kelas mampu meningkatkan presisi secara agregat, efeknya masih sangat sensitif terhadap variasi distribusi data latih. Sementara itu, *Attention Only* terpuruk pada rata-rata terendah (19,38% ± 2,44%) dengan konsistensi semu (IQR 0,0000), yang menegaskan kegagalan mekanisme atensi tunggal dalam membedakan fitur antar-kelas secara akurat. Adapun *Full Model* menunjukkan ketidakstabilan paling tinggi dengan IQR mencapai 0,1935. Meskipun mampu menyentuh nilai puncak yang impresif pada *fold* tertentu, *model* ini tetap dihantui oleh risiko penurunan performa yang drastis.



Gambar 4.9 Grafik Batang Rata-rata (*Mean*) ± Std *Macro Precision* pada Empat Skenario

Visualisasi perbandingan rata-rata pada Gambar 4.9 mempertegas dominasi skenario *Class Weight Only* sebagai *model* dengan ketepatan prediksi lintas kelas

terbaik. Pencapaian rata-rata 33,91% menunjukkan bahwa intervensi pada fungsi *loss* secara efektif mengarahkan *model* untuk lebih berhati-hati dalam memberikan label kelas minoritas, sehingga meminimalkan kesalahan *false positive*. Namun, tingginya simpangan baku pada *Class Weight Only* dan *Full Model* (27,58% ± 9,14%) menjadi catatan penting bahwa kompleksitas arsitektur dan pembobotan kelas menuntut proses inisialisasi yang lebih stabil. Sebaliknya, meskipun *Baseline* dan *Attention Only* memiliki simpangan baku yang lebih kecil, keduanya secara konsisten gagal mencapai level presisi yang memadai, yang membuktikan bahwa stabilitas tanpa diikuti oleh kualitas daya pembeda fitur tidak memberikan keuntungan praktis dalam deteksi tingkat keparahan depresi.

Tabel 4.7 Hasil Uji Signifikansi *Paired t-test* pada Metrik *Macro Precision*

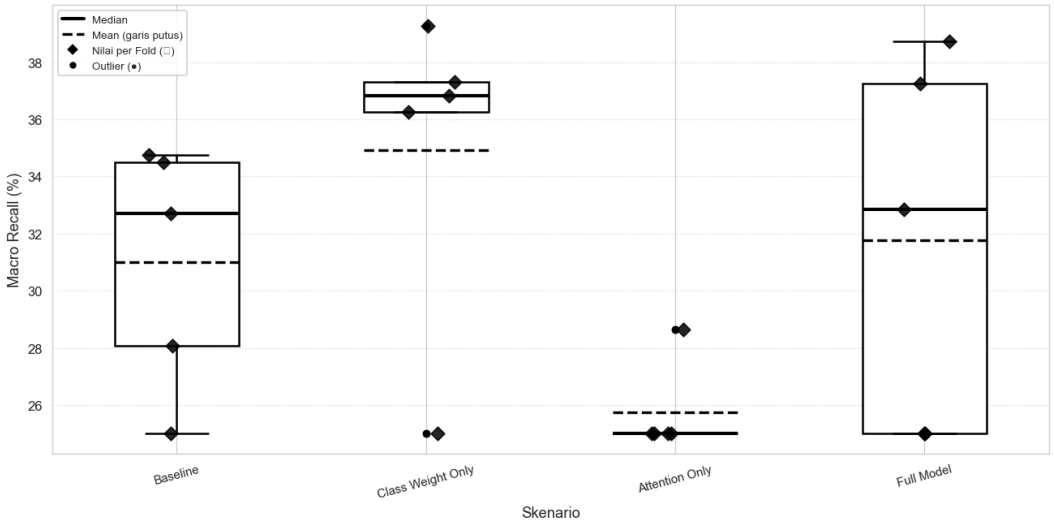
Perbandingan	**t**	**p-value**	**Keterangan**
Baseline vs *Class Weight Only*	-3.0510	0.037992	* (signifikan)
Baseline vs *Attention Only*	+2.6674	0.055957	ns (tidak signifikan)
Baseline vs *Full Model*	-0.6976	0.523824	ns (tidak signifikan)
Class Weight Only vs *Attention Only*	+3.3960	0.027379	* (signifikan)
Class Weight Only vs *Full Model*	+1.3922	0.236259	ns (tidak signifikan)
Attention Only vs *Full Model*	-2.0319	0.111973	ns (tidak signifikan)

Analisis signifikansi statistik yang dirangkum dalam Tabel 4.7 memberikan landasan bukti yang kuat bagi temuan di atas. Terdapat dua pasangan perbandingan yang menunjukkan perbedaan performa yang bermakna secara statistik pada taraf nyata 5%. Pertama, perbandingan antara *Baseline* dan *Class Weight Only* (p = 0,037992) membuktikan secara empiris bahwa penerapan *class weight* memberikan kontribusi nyata dalam meningkatkan *macro Precision*. Kedua, keunggulan *Class Weight Only* terhadap *Attention Only* (p = 0,027379) menegaskan bahwa mekanisme penyeimbang kelas jauh lebih krusial dibandingkan mekanisme atensi

dalam membangun *model* yang adil terhadap seluruh kategori. Di sisi lain, fakta bahwa *Full Model* tidak menunjukkan perbedaan signifikan terhadap *Baseline* maupun *Class Weight Only* mengindikasikan bahwa penambahan lapisan atensi pada *model* yang sudah memiliki pembobotan kelas belum mampu memberikan peningkatan presisi yang konsisten dan stabil secara statistik. Hal ini memperkuat argumen utama bahwa pembobotan kelas merupakan komponen yang paling kritikal dalam menangani ketidakseimbangan kelas pada dataset Reddit ini.

4.3.3 Analisis *Macro Recall*

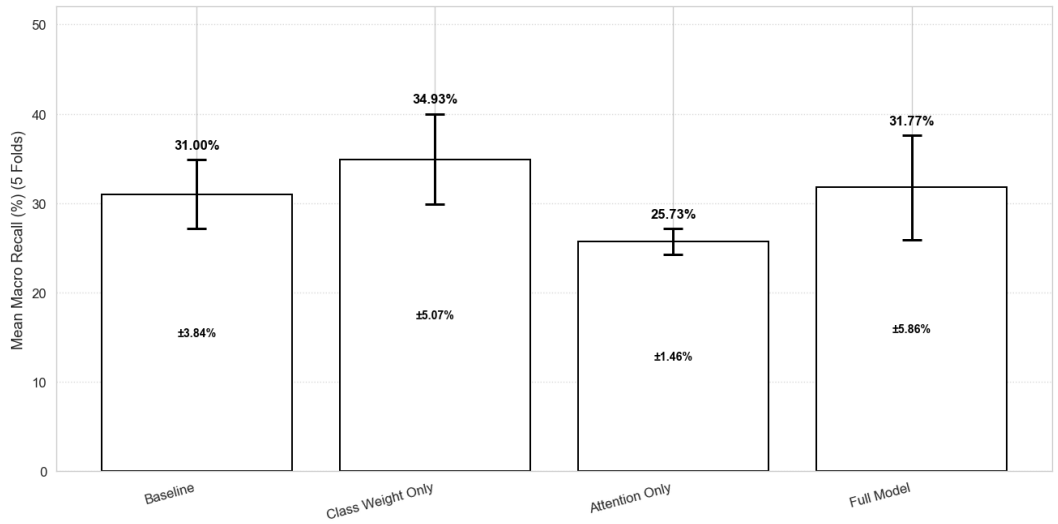
Macro Recall merupakan metrik yang mengevaluasi kemampuan *model* dalam mengidentifikasi kembali sampel dari setiap kategori secara seimbang. Perhitungan metrik ini dilakukan dengan menghitung nilai *Recall* untuk masing-masing kelas secara individu, lalu merata-ratakannya tanpa mempertimbangkan dominasi jumlah sampel pada tiap kelas. Dalam konteks klasifikasi tingkat keparahan depresi yang memiliki distribusi data tidak seimbang, *macro Recall* menjadi parameter evaluasi yang sangat krusial karena berkaitan langsung dengan risiko *false negative*. Kegagalan *model* dalam mendeteksi indikasi depresi, terutama pada tingkat keparahan yang lebih tinggi (kelas minoritas), dapat memberikan implikasi serius dalam skenario diagnosa klinis awal.



Gambar 4.10 *Boxplot* Distribusi *Macro Recall* pada Empat Skenario

Berdasarkan visualisasi *boxplot* pada Gambar 4.10, skenario *Baseline* menunjukkan performa *macro Recall* rata-rata sebesar 31,00% ± 3,84%, dengan variansi antar-lipatan (*fold*) yang berkisar antara nilai *minimum* 25,00% hingga maksimum 34,76%. Skenario *Class Weight Only* memberikan peningkatan nilai rata-rata yang signifikan menjadi 34,93% ± 5,07%, yang membuktikan bahwa penerapan pembobotan kelas berhasil mendorong *model* untuk menjadi lebih sensitif terhadap seluruh kategori, meskipun stabilitas nilainya tetap dipengaruhi oleh variasi komposisi data pada setiap lipatan. Kontras dengan hal tersebut, skenario *Attention Only* kembali berada pada performa terendah dengan rata-rata 25,73% ± 1,46% dan sebaran yang sangat sempit, di mana nilai median berada tepat pada angka 25,00%. Hal ini mengindikasikan bahwa pada sebagian besar lipatan pengujian, *model* atensi murni cenderung mengalami stagnasi pada tingkat sensitivitas yang sangat rendah, mendekati perilaku tebakan dasar. Adapun *Full Model* mencatatkan rata-rata 31,77% ± 5,86% dengan rentang sebaran yang lebih

lebar (25,00% hingga 38,74%), memperlihatkan potensi sensitivitas yang tinggi namun dengan tingkat konsistensi yang lebih rendah antar-lipatan.



Gambar 4.11 Grafik Batang Rata-rata (*Mean*) ± Std *Macro Recall* pada Empat Skenario

Perbandingan performa sensitivitas lintas kelas secara lebih jelas dapat diamati melalui Gambar 4.11. Skenario *Class Weight Only* mengukuhkan posisinya sebagai konfigurasi terbaik dengan nilai rata-rata tertinggi sebesar 34,93%, selaras dengan tujuan utama pembobotan kelas untuk memitigasi dominasi kelas mayoritas. Sebaliknya, temuan pada skenario *Attention Only* yang menghasilkan skor terendah (25,73%) menggugurkan hipotesis bahwa mekanisme atensi secara mandiri dapat mengatasi ketidakseimbangan data. Tanpa panduan dari fungsi *loss* yang diseimbangkan, mekanisme atensi justru memperkuat bias data latih dengan mengalokasikan seluruh fokusnya pada pola dominan dari kelas mayoritas, sementara sinyal-sinyal kritis dari kelas minoritas diabaikan karena dianggap sebagai derau atau *noise* belaka. Temuan ini mempertegas bahwa pada metrik yang berorientasi pada sensitivitas ini, penerapan *class weight* merupakan faktor

intervensi yang paling konsisten dalam menghasilkan peningkatan performa yang nyata.

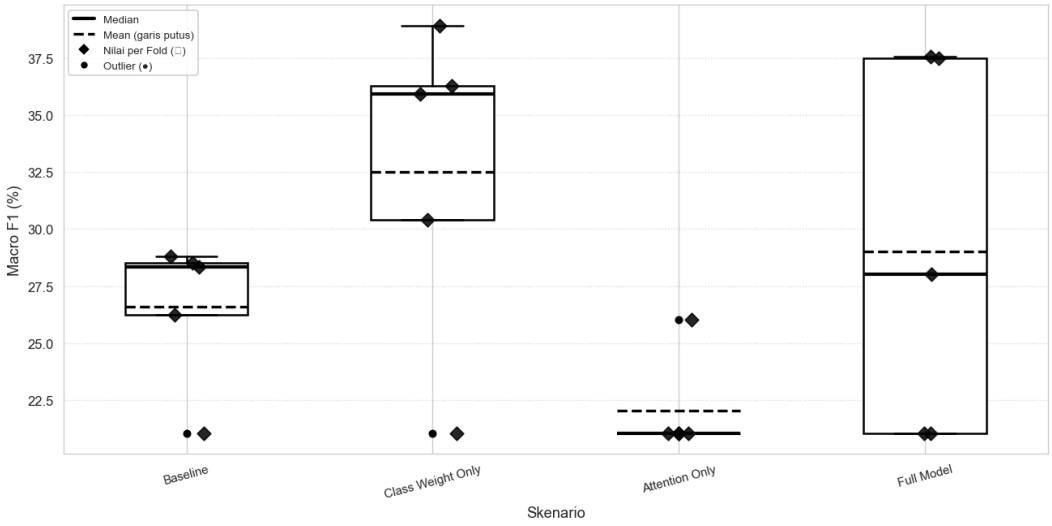
Tabel 4.8 Hasil Uji Signifikansi *Paired t-test* pada Metrik *Macro Recall*

Perbandingan	**t**	**p-value**	**Keterangan**
Baseline vs *Class Weight Only*	-2.5371	0.064179	ns (tidak signifikan)
Baseline vs *Attention Only*	+3.1211	0.035490	* (signifikan)
Baseline vs *Full Model*	-0.6228	0.567166	ns (tidak signifikan)
Class Weight Only vs *Attention Only*	+3.6771	0.021257	* (signifikan)
Class Weight Only vs *Full Model*	+1.2897	0.266660	ns (tidak signifikan)
Attention Only vs *Full Model*	-2.3564	0.077970	ns (tidak signifikan)

Analisis signifikansi statistik yang dirangkum dalam Tabel 4.8 memberikan validasi kuat terhadap pola performa yang ditemukan. Terdapat dua pasangan perbandingan yang menunjukkan perbedaan *macro Recall* yang signifikan secara statistik pada taraf nyata 5%, yaitu perbandingan *Baseline* terhadap *Attention Only* (p = 0,035490) dan *Class Weight Only* terhadap *Attention Only* (p = 0,021257). Hasil ini membuktikan secara empiris bahwa pelemahan sensitivitas *model* pada skenario *Attention Only* merupakan kegagalan sistematis dan bukan sekadar fluktuasi acak antar-lipatan data. Di sisi lain, perbandingan antara *Baseline* dan *Class Weight Only* menghasilkan nilai p sebesar 0,064179 yang berada sangat dekat dengan ambang batas 0,05, mengindikasikan adanya tren peningkatan yang kuat namun belum cukup konsisten untuk dinyatakan signifikan pada taraf uji yang ditetapkan. Sementara itu, tidak adanya perbedaan signifikan pada perbandingan yang melibatkan *Full Model* semakin mengonfirmasi bahwa variabilitas kinerja pada *model* hibrida tersebut membuat keunggulan atau kelemahannya belum stabil secara statistik.

4.3.4 Analisis *Macro F1-Score* (*Macro* F1)

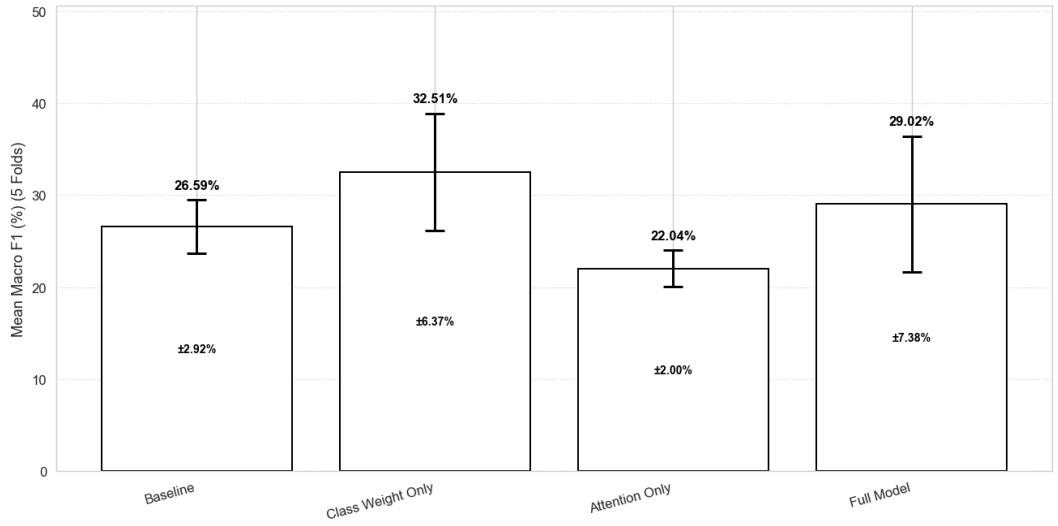
Macro F1-Score merupakan parameter evaluasi yang dihitung melalui rata-rata harmonik antara *macro Precision* dan *macro Recall*. Keunggulan utama metrik ini terletak pada perlakuannya yang setara terhadap setiap kategori tanpa dipengaruhi oleh dominasi jumlah sampel pada kelas tertentu, sehingga sering dianggap sebagai indikator paling objektif atau "jujur" dalam menilai performa *model* pada tugas klasifikasi multi-kelas yang tidak seimbang. Secara esensial, *macro F1* memberikan penilaian menyeluruh mengenai keseimbangan daya prediksi *model* dalam memetakan label secara tepat (presisi) sekaligus menangkap keberadaan sampel pada seluruh tingkat keparahan depresi (*Recall*).



Gambar 4.12 *Boxplot* Distribusi *Macro F1* pada Empat Skenario

Eksplorasi terhadap distribusi performa melalui *boxplot* pada Gambar 4.12 menyingkap dinamika yang bervariasi antar-skenario. Skenario *Baseline* menghasilkan rata-rata *macro F1* sebesar 26,59% ± 2,92% dengan sebaran yang tergolong moderat (IQR 2,25%), menunjukkan tingkat stabilitas yang cukup baik namun tetap sensitif terhadap variasi komposisi data antar-lipatan. Peningkatan

performa yang paling signifikan terlihat pada skenario *Class Weight Only* yang mencatatkan rata-rata mencapai 32,51% ± 6,37% dengan rentang nilai yang lebih lebar (21,04% hingga 38,93%). Hal ini membuktikan bahwa intervensi pembobotan kelas secara efektif mendorong perbaikan keseimbangan antara presisi dan *Recall* di seluruh lintas kelas. Di sisi lain, *Attention Only* mencatatkan performa terendah dengan rata-rata 22,04% ± 2,00% dan sebaran yang sangat sempit (IQR 0,00%), mengindikasikan kegagalan *model* dalam menghasilkan kualitas prediksi yang memadai di mana sebagian besar lipatan berkumpul pada titik performa yang sama rendahnya. Adapun *Full Model* berada pada nilai rata-rata 29,02% ± 7,38%, namun memperlihatkan sebaran paling lebar (IQR 16,43%) dengan rentang nilai yang ekstrem, menunjukkan bahwa hibridasi metode pada *model* ini berpotensi memberikan hasil yang sangat baik namun sangat rentan terhadap ketidakstabilan tinggi antar-lipatan.



Gambar 4.13 Grafik Batang Rata-rata (*Mean*) ± Std *Macro F1* pada Empat Skenario

Berdasarkan perbandingan rata-rata pada Gambar 4.13, dapat disimpulkan bahwa skenario *Class Weight Only* menempati peringkat performa terbaik, yang

mengukuhkan argumen bahwa strategi pembobotan kelas memberikan kontribusi paling nyata dalam memperbaiki keadilan prediksi *model* di seluruh kategori depresi. *Full Model* menyusul di peringkat kedua, namun catatan simpangan bakunya yang besar menegaskan bahwa performanya paling tidak stabil, di mana keberhasilan pada lipatan tertentu diimbangi oleh penurunan tajam pada lipatan lainnya. *Baseline* menempati urutan berikutnya dengan karakter yang lebih stabil meskipun secara nominal lebih rendah dibandingkan skenario pembobotan kelas. Terakhir, skenario *Attention Only* tertahan secara konsisten di posisi terbawah, memperkuat temuan bahwa tanpa mekanisme penyeimbang kelas, atensi justru memperkuat bias *model* terhadap kelas mayoritas dan memperlakukan sinyal unik dari kelas minoritas sebagai derau atau *noise*.

Tabel 4.9 Hasil Uji Signifikansi *Paired t-test* pada Metrik *Macro* F1

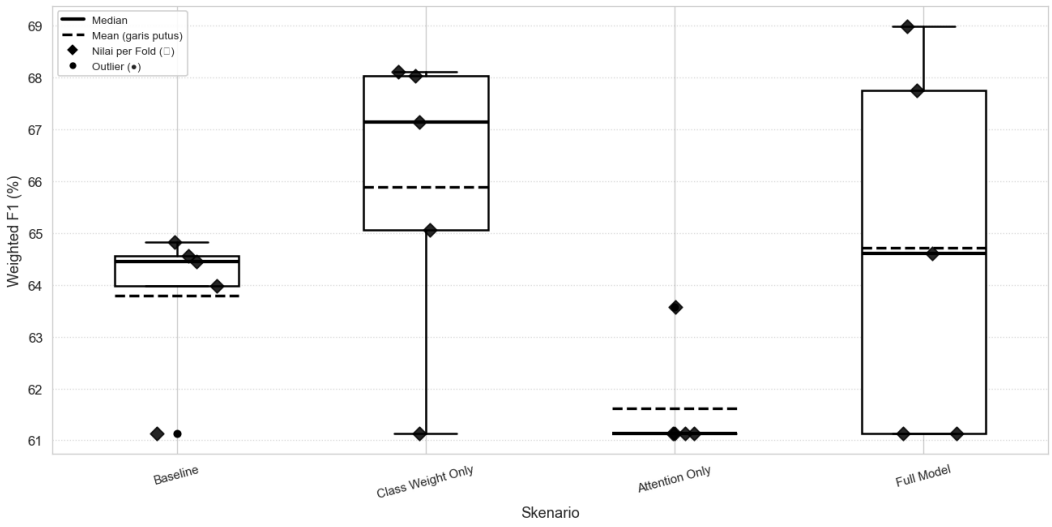
Perbandingan	**t**	**p-value**	**Keterangan**
Baseline vs *Class Weight Only*	-2.7748	0.050082	ns (tidak signifikan)
Baseline vs *Attention Only*	+3.2066	0.032694	* (signifikan)
Baseline vs *Full Model*	-0.8682	0.434295	ns (tidak signifikan)
Class Weight Only vs *Attention Only*	+3.4085	0.027066	* (signifikan)
Class Weight Only vs *Full Model*	+1.1584	0.311147	ns (tidak signifikan)
Attention Only vs *Full Model*	-2.1699	0.095820	ns (tidak signifikan)

Validitas dari pola performa tersebut dikonfirmasi melalui hasil uji statistik yang dirangkum pada Tabel 4.9. Terdapat dua pasangan perbandingan yang menunjukkan perbedaan signifikan pada taraf nyata 5%, yaitu perbandingan *Baseline* terhadap *Attention Only* (p = 0,032694) serta *Class Weight Only* terhadap *Attention Only* (p = 0,027066). Temuan ini membuktikan secara empiris bahwa skenario *Attention Only* secara statistik memang menghasilkan performa *macro F1* yang lebih rendah, sehingga kegagalannya dalam menyeimbangkan presisi dan

Recall tidak dapat dianggap sebagai sekadar variasi antar-lipatan. Sementara itu, perbandingan antara *Baseline* dan *Class Weight Only* menghasilkan nilai p sebesar 0,050082 yang berada sangat tipis di atas ambang batas 0,05. Hal ini menandakan adanya tren peningkatan yang kuat secara praktis namun belum mencapai konsistensi statistik yang cukup kaku pada jumlah sampel pengujian ini. Terakhir, tidak adanya perbedaan signifikan pada perbandingan yang melibatkan *Full Model* semakin menegaskan bahwa variansi performa yang tinggi pada *model* kompleks tersebut membuat keunggulannya belum stabil secara statistik dibanding skenario lainnya.

4.3.5 Analisis *Weighted F1-Score* (Weighted F1)

Weighted F1-Score merupakan varian dari metrik evaluasi F1 yang mengkalkulasi nilai rata-rata harmonik dengan memberikan bobot proporsional berdasarkan jumlah sampel aktual (frekuensi) pada masing-masing kelas. Berbeda dengan *macro F1* yang murni egalitarian, metrik ini memberikan kontribusi yang lebih besar pada kelas mayoritas terhadap skor akhir. Pada dataset dengan tingkat ketidakseimbangan kelas yang ekstrem seperti teks Reddit ini (di mana kelas *minimum* mendominasi 72,6% data), *weighted F1* menjadi instrumen penting untuk melihat "kinerja operasional keseluruhan" *model*. Metrik ini secara alamiah akan terlihat lebih "ramah" dan memihak pada *model* yang memiliki performa kuat di kelas dominan. Oleh karena itu, jika sebuah skenario gagal pada metrik ini, hal tersebut merupakan indikasi kegagalan ekstraksi fitur yang sangat fatal.

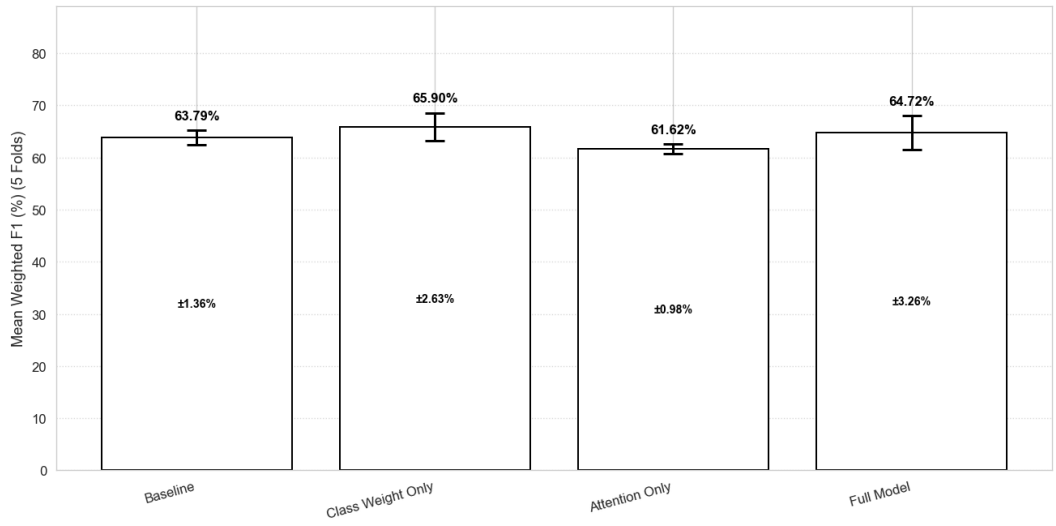


Gambar 4.14 *Boxplot* Distribusi *Weighted F1* pada Empat Skenario

Berdasarkan visualisasi *boxplot* pada Gambar 4.14, skenario *Baseline* menghasilkan performa yang cukup tangguh dengan rata-rata *weighted F1* sebesar 0,6379 ± 0,0136. Sebaran data pada skenario ini relatif rapat (IQR 0,0058; rentang 0,6113–0,6483), yang menunjukkan stabilitas prediksi pada sebagian besar lipatan (*fold*), meskipun terdapat satu lipatan anomali yang turun hingga angka 0,6113. Skenario *Class Weight Only* berhasil mencatatkan nilai rata-rata tertinggi sebesar 0,6590 ± 0,0263. Namun, peningkatan ini diiringi dengan sebaran yang lebih lebar (IQR 0,0297; rentang 0,6113–0,6812), mengindikasikan bahwa intervensi pembobotan kelas mampu mendorong performa agregat secara signifikan pada sebagian lipatan, meski belum sepenuhnya konsisten di seluruh distribusi data uji.

Di sisi lain, skenario *Attention Only* terpuruk pada nilai terendah dengan rata-rata 0,6162 ± 0,0098 dan sebaran yang sangat sempit (IQR 0,0000). Pada hampir seluruh lipatan, nilainya terpaku pada angka dominan 0,6113 dan hanya sesekali mengalami kenaikan. Pola yang sangat seragam ini merupakan bukti

bahwa mekanisme atensi tanpa penyeimbang kelas justru membatasi kapasitas *model*, menghasilkan kualitas prediksi agregat yang rendah bahkan ketika metrik evaluasinya memberikan keuntungan (*bias*) pada kelas mayoritas. Sementara itu, *Full Model* menempati posisi menengah dengan rata-rata 0,6472 ± 0,0326, namun menunjukkan variansi yang paling ekstrem (IQR 0,0663; rentang 0,6113–0,6900). Hal ini menegaskan kembali bahwa kombinasi arsitektur kompleks pada *Full Model* memicu ketidakstabilan, *model* dapat beradaptasi dengan sangat baik pada kondisi data tertentu, namun rentan mengalami kolaps pada kondisi lainnya.



Gambar 4.15 Grafik Batang Rata-rata (*Mean*) ± Std *Weighted F1* pada Empat Skenario

Konfirmasi pemeringkatan performa secara visual dapat diamati pada Gambar 4.15. Skenario *Class Weight Only* menempati peringkat puncak, disusul oleh *Full Model* di posisi kedua yang sayangnya direduksi oleh tingginya rentang simpangan baku. *Baseline* berada pada urutan ketiga dengan karakter yang jauh lebih stabil dibandingkan dua skenario di atasnya. *Attention Only* kembali menjadi skenario terlemah, sebuah temuan yang sangat ironis mengingat metrik *weighted*

F1 seharusnya menguntungkan *model* yang bias terhadap kelas mayoritas. Temuan ini secara krusial memperlihatkan bahwa strategi pembobotan kelas (*class weighting*) adalah pendekatan paling rasional dan efektif untuk meningkatkan performa agregat, sedangkan penyematan lapisan atensi secara tunggal justru merusak representasi fitur secara keseluruhan.

Tabel 4.10 Hasil Uji Signifikansi *Paired t-test* pada Metrik Weighted F1

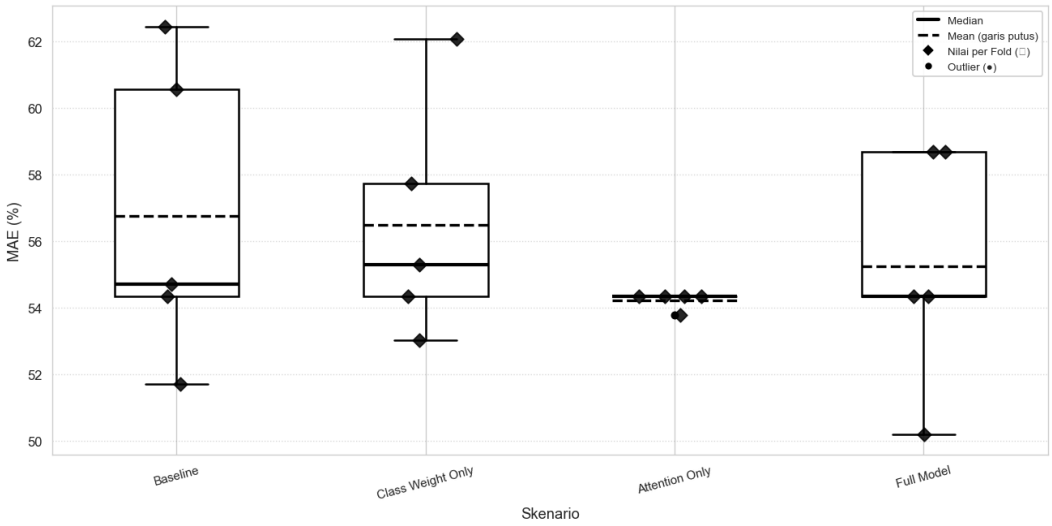
Perbandingan	**t**	**p-value**	**Keterangan**
Baseline vs *Class Weight Only*	-2.4838	0.067936	ns (tidak signifikan)
Baseline vs *Attention Only*	+3.0326	0.038683	* (signifikan)
Baseline vs *Full Model*	-0.7071	0.518527	ns (tidak signifikan)
Class Weight Only vs *Attention Only*	+3.3218	0.029330	* (signifikan)
Class Weight Only vs *Full Model*	+0.8101	0.463298	ns (tidak signifikan)
Attention Only vs *Full Model*	-2.1102	0.102466	ns (tidak signifikan)

Validitas temuan di atas diperkuat oleh pengujian statistik yang dirangkum pada Tabel 4.10. Terdapat dua pasangan perbandingan yang menunjukkan perbedaan signifikan secara statistik pada taraf nyata 5%, yaitu perbandingan *Baseline* terhadap *Attention Only* (p = 0,038683) dan *Class Weight Only* terhadap *Attention Only* (p = 0,029330). Fakta statistik ini merupakan bukti empiris yang tidak terbantahkan bahwa skenario *Attention Only* memiliki kinerja yang jauh lebih buruk dibandingkan *model* dasar maupun *model* dengan pembobotan kelas, bahkan pada metrik yang sudah mengakomodasi ketimpangan jumlah sampel. Sementara itu, perbandingan antara *Baseline* dan *Class Weight Only* tidak mencapai ambang signifikansi (p = 0,067936) meskipun secara nominal rata-rata *Class Weight Only* lebih tinggi. Hal ini dapat diinterpretasikan bahwa lonjakan *weighted F1* akibat pembobotan kelas masih terpengaruh oleh fluktuasi antar-lipatan sehingga belum cukup stabil untuk dinyatakan berbeda secara statistik. Begitu pula halnya dengan

Full Model yang tidak memiliki perbedaan signifikan terhadap *Baseline* maupun *Class Weight Only*, yang mengonfirmasi bahwa potensi keunggulan nilai pada beberapa lipatan gagal dikonversi menjadi keandalan *model* yang stabil secara keilmuan statistik.

4.3.6 Analisis *Mean absolute error* (*MAE*)

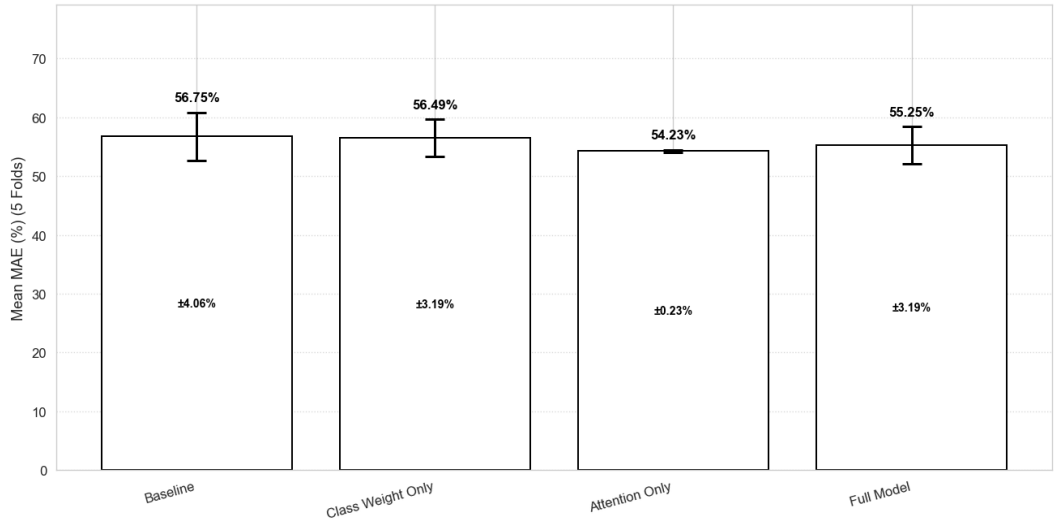
Mean absolute error (*MAE*) merupakan metrik evaluasi yang digunakan untuk mengukur rata-rata besaran kesalahan prediksi melalui perhitungan jarak absolut antara label prediksi dan label aktual. Dalam klasifikasi tingkat keparahan depresi yang bersifat *ordinal*, metrik ini memiliki relevansi yang sangat tinggi karena mampu memberikan penalti yang proporsional terhadap "jarak" kesalahan. Kesalahan prediksi yang melompat jauh, misalnya dari kategori *minimum* ke *severe*, akan dihitung sebagai beban kesalahan yang lebih besar dibandingkan kesalahan yang hanya bergeser satu tingkat keparahan. Secara teoretis, semakin kecil nilai *MAE*, maka semakin dekat prediksi *model* terhadap kondisi riil pasien, sehingga performa *model* dianggap semakin optimal. Namun, pada dataset dengan ketidakseimbangan kelas yang ekstrem, interpretasi *MAE* memerlukan ketelitian tinggi agar tidak terjebak pada angka rendah yang bersifat manipulatif.



Gambar 4.16 *Boxplot* Distribusi *MAE* pada Empat Skenario

Visualisasi distribusi *MAE* melalui *boxplot* pada Gambar 4.16 memperlihatkan bahwa skenario *Attention Only* mencatatkan rata-rata kesalahan terendah yaitu 0,5423 ± 0,0023 dengan sebaran antar-lipatan (*fold*) yang sangat sempit. Di sisi lain, skenario *Baseline* memiliki rata-rata *MAE* sebesar 0,5675 ± 0,0406 yang jauh lebih tinggi dan bervariasi, mengindikasikan bahwa jarak kesalahan prediksi dapat berubah secara signifikan bergantung pada komposisi data di setiap lipatan. Skenario *Class Weight Only* berada pada angka 0,5649 ± 0,0319, sedangkan *Full Model* mencatatkan nilai 0,5525 ± 0,0319. Meskipun *Attention Only* tampak unggul secara nominal, keunggulan ini tidak serta merta mencerminkan kemampuan *model* dalam mengenali seluruh spektrum keparahan depresi. Dalam kondisi data yang sangat timpang, nilai *MAE* yang rendah sering kali muncul sebagai efek samping dari *model* yang memusatkan prediksinya pada kelas mayoritas yang dominan. Karena sebagian besar sampel berasal dari kelas *minimum*, *model* yang bias akan memiliki jarak kesalahan nol atau kecil pada

mayoritas data, meskipun ia gagal total dalam menangkap kasus pada kelas minoritas yang justru lebih kritis.



Gambar 4.17 Grafik Batang Rata-rata (*Mean*) ± Std *MAE* pada Empat Skenario

Berdasarkan perbandingan rata-rata pada Gambar 4.17, dapat diamati bahwa skenario *Attention Only* menempati posisi dengan jarak kesalahan terkecil dan kestabilan yang paling tinggi di seluruh lipatan pengujian. *Full Model* berada di posisi kedua, menunjukkan bahwa integrasi komponen tambahan mampu menurunkan rata-rata kesalahan jika dibandingkan dengan skenario *Baseline*. Namun, rendahnya nilai *MAE* pada skenario yang melibatkan atensi tanpa penyeimbang kelas merupakan konfirmasi atas bias *model* terhadap kelas mayoritas. Hal ini mengindikasikan bahwa metrik *MAE* tidak dapat digunakan sebagai parameter tunggal untuk menetapkan skenario terbaik. Interpretasi terhadap metrik ini harus disejajarkan dengan metrik makro yang secara eksplisit menilai kinerja lintas kelas guna memastikan bahwa rendahnya kesalahan rata-rata bukan merupakan hasil dari pengabaian kelas minoritas demi mengejar statistik global yang terlihat baik.

Tabel 4.11 Hasil Uji Signifikansi *Paired t-test* pada Metrik *MAE*

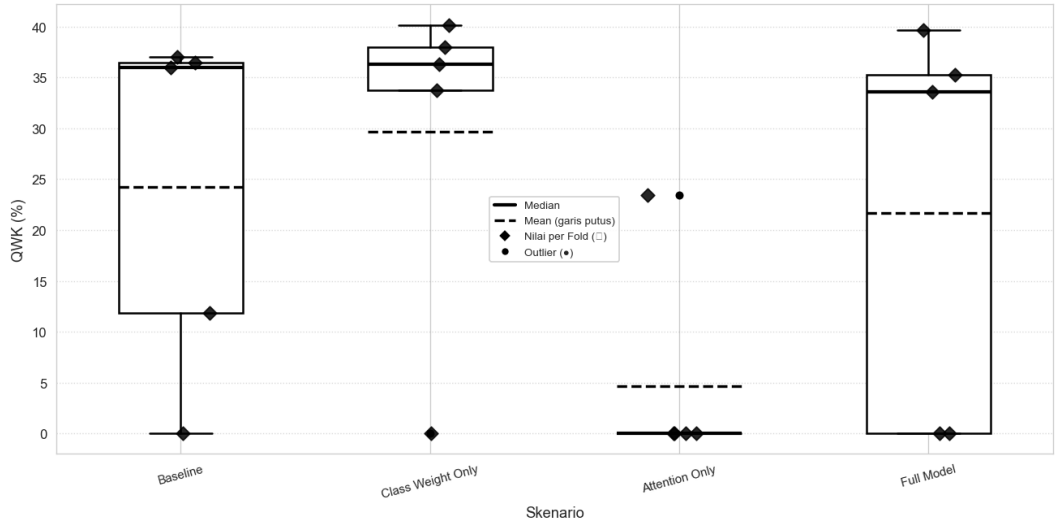
Perbandingan	t	p-value	Keterangan
Baseline vs *Class Weight Only*	+0.1102	0.917575	ns (tidak signifikan)
Baseline vs *Attention Only*	+1.2114	0.292415	ns (tidak signifikan)
Baseline vs *Full Model*	+0.5251	0.627265	ns (tidak signifikan)
Class Weight Only vs *Attention Only*	+1.3969	0.234971	ns (tidak signifikan)
Class Weight Only vs *Full Model*	+0.9789	0.383065	ns (tidak signifikan)
Attention Only vs *Full Model*	-0.6147	0.571969	ns (tidak signifikan)

Setelah dilakukan uji signifikansi statistik yang dirangkum pada Tabel 4.11, ditemukan bahwa seluruh pasangan perbandingan memiliki nilai p yang jauh melampaui ambang batas 0,05. Fakta statistik ini membuktikan bahwa tidak ada perbedaan *MAE* yang signifikan secara nyata di antara keempat skenario eksperimen yang diuji. Meskipun secara nominal skenario *Attention Only* terlihat memiliki rata-rata kesalahan yang lebih kecil, perbedaan tersebut belum cukup konsisten di seluruh lipatan data untuk dapat disimpulkan sebagai sebuah keunggulan arsitektural yang valid secara keilmuan statistik. Dengan demikian, variasi nilai *MAE* yang muncul pada setiap skenario masih berada dalam batas fluktuasi antar-lipatan yang wajar, sehingga daya pembeda utama antar-*model* lebih tepat diamati melalui metrik sensitivitas kelas yang telah dibahas pada bagian sebelumnya.

4.3.7 Analisis *Quadratic weighted kappa* (QWK)

Quadratic weighted kappa (QWK) merupakan metrik evaluasi yang dirancang secara khusus untuk menangani kasus klasifikasi *ordinal* karena tidak hanya menghitung tingkat kesesuaian prediksi dengan label aktual melainkan juga memberikan penalti yang bersifat kuadratik ketika terjadi kesalahan prediksi yang

berjarak jauh dari label sebenarnya. Berbeda dengan metrik akurasi yang dapat memberikan hasil yang tampak tinggi secara manipulatif pada kondisi data yang tidak seimbang, QWK cenderung jauh lebih sensitif terhadap kualitas keputusan *model* di seluruh tingkatan kelas. Oleh karena itu, metrik ini menjadi parameter yang sangat vital untuk menilai apakah arsitektur *model* benar benar memiliki keselarasan dengan struktur *ordinal* dari label tingkat keparahan depresi yang bersifat hierarkis.

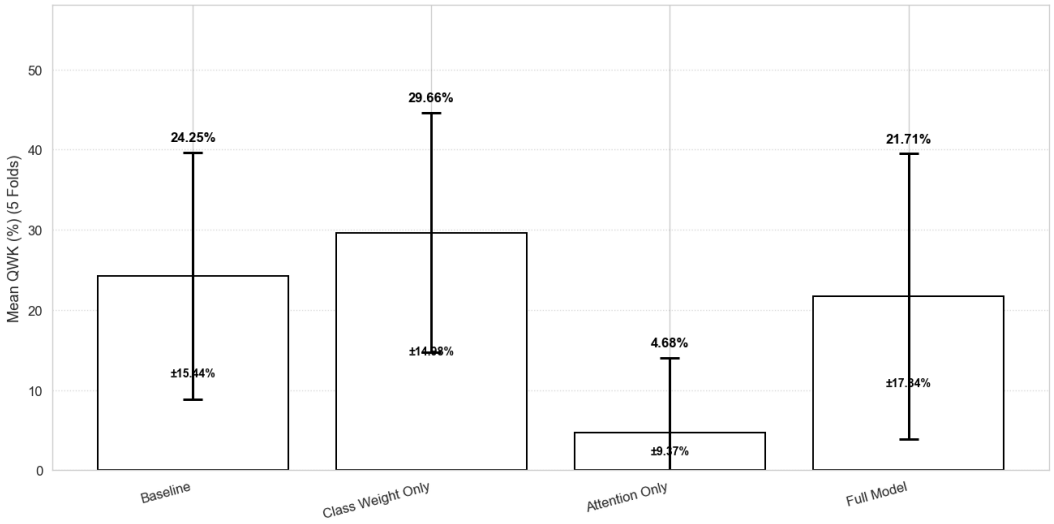


Gambar 4.18 *Boxplot* Distribusi QWK pada Empat Skenario

Berdasarkan visualisasi sebaran data melalui *boxplot* pada Gambar 4.18, terlihat dinamika performa yang sangat kontras di antara keempat skenario eksperimen. Skenario *Baseline* menghasilkan nilai QWK rata rata sebesar 0,2425 ± 0,1544 dengan rentang sebaran yang cukup lebar serta ditemukannya nilai *minimum* nol pada salah satu lipatan pengujian. Kondisi ini mengindikasikan bahwa pada komposisi data tertentu *model* dasar dapat mengalami kegagalan kesepakatan yang sangat ekstrem. Skenario *Class Weight Only* menunjukkan peningkatan performa

yang paling menjanjikan dengan rata rata sebesar 29,66% ± 14,98% disertai dengan rentang antarkuartil (IQR) yang jauh lebih sempit yaitu sebesar 0,0423. Hal ini membuktikan bahwa selain mencapai tingkat kesepakatan yang lebih tinggi, strategi pembobotan kelas juga menghasilkan konsistensi yang lebih baik di hampir seluruh lipatan pengujian.

Sebaliknya, skenario *Attention Only* berada pada level performa terendah dengan rata rata hanya sebesar 4,68% ± 9,37% dengan nilai median dan IQR yang menyentuh angka nol mutlak. Temuan ini menunjukkan bahwa pada mayoritas lipatan pengujian *model* hampir tidak mampu mencapai kesepakatan yang lebih baik daripada peluang acak dan cenderung mengalami kondisi kolaps secara sistematis. Adapun *Full Model* mencatatkan rata rata sebesar 21,71% ± 17,84% dengan sebaran data yang paling lebar di mana setidaknya seperempat dari total lipatan pengujian jatuh pada kondisi kesepakatan nol. Meskipun pada lipatan tertentu *model* ini mampu mencapai nilai maksimum hingga 0,3969, ketidakstabilannya yang tinggi menjadi catatan kritis dalam implementasi arsitektur hibrida ini.



Gambar 4.19 Grafik Batang Rata-rata (*Mean*) ± Std QWK pada Empat Skenario

Melalui representasi grafik batang pada Gambar 4.19, dapat diamati pemeringkatan performa yang mempertegas keunggulan strategi intervensi pada fungsi *loss*. Skenario *Class Weight Only* mengukuhkan posisinya sebagai konfigurasi yang paling selaras dengan struktur *ordinal* depresi. Pola ini memberikan *insight* penting bahwa pada tugas klasifikasi dengan tingkat ketidakseimbangan kelas yang sangat tinggi di mana kelas *minimum* mendominasi sebesar 72,6% dari keseluruhan dataset, mekanisme pembobotan kelas jauh lebih krusial dibandingkan mekanisme atensi tunggal.

Kegagalan fatal pada skenario *Attention Only* yang menghasilkan skor QWK mendekati nol membuktikan bahwa tanpa adanya penyeimbang kelas, mekanisme atensi justru melakukan kesalahan fokus dengan hanya mempelajari pola pola dominan dari kelas mayoritas dan menganggap sinyal emosional dari kelas minoritas sebagai derau atau *noise*. Sementara itu, rendahnya stabilitas pada *Full Model* menunjukkan bahwa penambahan parameter yang berlebihan melalui lapisan atensi pada volume dataset yang terbatas justru menyulitkan *model* dalam

mencapai konvergensi yang konsisten untuk mengenali urutan tingkat keparahan depresi secara akurat.

Tabel 4.12 Hasil Uji Signifikansi *Paired t-test* pada Metrik QWK

Perbandingan	**t**	**p-value**	**Keterangan**
Baseline vs *Class Weight Only*	-1.0175	0.366434	ns (tidak signifikan)
Baseline vs *Attention Only*	+2.6913	0.054584	ns (tidak signifikan)
Baseline vs *Full Model*	+1.0213	0.364870	ns (tidak signifikan)
Class Weight Only vs *Attention Only*	+3.0216	0.039102	* (signifikan)
Class Weight Only vs *Full Model*	+1.0413	0.356551	ns (tidak signifikan)
Attention Only vs *Full Model*	-1.9851	0.118115	ns (tidak signifikan)

Analisis signifikansi statistik yang dirangkum dalam Tabel 4.12 memberikan dasar bukti yang kuat bagi temuan di atas. Satu satunya pasangan perbandingan yang menunjukkan perbedaan yang signifikan secara nyata pada taraf uji 5% adalah antara *Class Weight Only* dan *Attention Only* dengan nilai p sebesar 0,039102. Hasil ini menegaskan bahwa keunggulan skenario pembobotan kelas terhadap skenario atensi murni merupakan perbedaan arsitektural yang valid secara ilmiah dan bukan merupakan fluktuasi acak antar lipatan data. Sementara itu, perbandingan antara *Baseline* dan *Attention Only* menghasilkan nilai p sebesar 0,054584 yang berada sangat dekat dengan ambang batas signifikansi, menunjukkan adanya kecenderungan kuat bahwa *model* dasar masih lebih baik daripada *model* yang hanya mengandalkan atensi tanpa penyeimbang kelas. Adapun perbandingan yang melibatkan *Full Model* seluruhnya tidak menunjukkan perbedaan yang signifikan, yang mana hal ini selaras dengan temuan pada *boxplot* mengenai besarnya variansi performa *model* tersebut yang membuatnya tidak konsisten secara statistik di seluruh lipatan pengujian.

4.3.8 Kesimpulan Analisis Perbandingan Skenario *Model*

Subbab ini merangkum keseluruhan temuan dari metrik evaluasi yang telah dibahas pada Subbab 4.3.1 hingga 4.3.7. Tujuan dari sintesis ini adalah untuk menegaskan kecenderungan performa dari masing-masing skenario arsitektur secara komprehensif. Ringkasan disajikan dalam dua kerangka analisis: pertama, rekapitulasi nilai rata-rata (*mean*) per metrik untuk membaca kecenderungan performa secara deskriptif. Kedua, rekapitulasi *Win–Lose–Tie* berbasis hasil uji signifikansi berpasangan untuk mengonfirmasi dominasi relatif antar-skenario berdasarkan kekuatan bukti statistik.

Tabel 4.13 Kesimpulan *Mean* Per Skenario x Metrik (dalam persen, %)

Skenario	**Accuracy**	***Macro Precision***	***Macro Recall***	***Macro* F1**	**Weighted F1**	***MAE***	**QWK**
Baseline	71.47	24.5	31	26.59	63.79	56.75	24.25
Class Weight Only	70.3	33.91	34.93	32.51	65.9	56.49	29.66
Attention Only	72.6	19.38	25.73	22.04	61.62	54.23	4.68
Full Model	71.47	27.58	31.77	29.02	64.72	55.25	21.71

Berdasarkan Tabel 4.13, skenario *Class Weight Only* mengukuhkan posisinya sebagai arsitektur yang paling konsisten dan tangguh dalam menangani tujuan klasifikasi multi-kelas pada data yang sangat tidak seimbang. Skenario ini memimpin secara absolut pada hampir seluruh metrik yang sensitif terhadap keadilan kelas, yakni *Macro Precision* (33,91%), *Macro Recall* (34,93%), *Macro F1* (32,51%), *Weighted F1* (65,90%), serta *Quadratic weighted kappa* atau QWK (29,66%). Temuan ini memberikan bukti empiris bahwa intervensi melalui

pembobotan kelas (*class weighting*) pada fungsi *loss* secara efektif memperbaiki kualitas prediksi lintas kelas dan menjaga keselarasan struktur *ordinal* tingkat keparahan depresi. Keberhasilan dalam mendeteksi kelas minoritas ini dicapai dengan mengorbankan akurasi global yang sedikit menurun menjadi 70,30%, sebuah kompromi (*trade-off*) yang sangat rasional dan diperlukan dalam skenario medis atau psikologis di mana deteksi *false negative* pada kelas kronis jauh lebih berbahaya daripada penurunan akurasi pada kelas mayoritas.

Sebaliknya, performa pada skenario *Attention Only* memperlihatkan sebuah anomali statistik yang patut diwaspadai. Secara agregat, skenario ini tampak sangat unggul dengan *Accuracy* tertinggi (72,60%) dan jarak kesalahan *MAE* terendah (54,23%). Namun, keunggulan semu tersebut runtuh ketika dievaluasi menggunakan metrik makro dan *ordinal*. Nilai *Macro Precision*, *Macro Recall*, dan *Macro F1* pada skenario ini justru terpuruk di posisi terbawah. Lebih fatal lagi, nilai QWK hancur hingga menyentuh angka 4,68%, yang mengindikasikan bahwa *model* gagal total dalam memahami struktur *ordinal* label depresi. Anomali ini membuktikan bahwa tanpa adanya panduan bobot penyeimbang, mekanisme atensi justru memperburuk bias *model* dengan hanya memusatkan "perhatiannya" pada pola kelas dominan (*minimum*) dan secara sistematis mengabaikan pola dari kelas minoritas yang krusial.

Adapun *Full Model*, yang merupakan hibridasi dari mekanisme atensi dan pembobotan kelas, menempati posisi menengah pada hampir seluruh metrik evaluasi. Meskipun secara teori penggabungan dua metode ini diharapkan mampu memberikan hasil maksimal, pada kenyataannya performa *model* ini tertahan oleh

kecenderungan variansi yang sangat tinggi antar-lipatan pengujian, seperti yang telah dibahas pada subbab-subbab sebelumnya. Kompleksitas penambahan lapisan atensi pada dataset yang memiliki ketimpangan ekstrem membuat *model* ini menjadi tidak stabil. Ia dapat memprediksi dengan sangat baik pada sebagian *fold*, namun mengalami penurunan drastis pada *fold* lainnya, sehingga gagal mengungguli stabilitas arsitektur *Class Weight Only* secara konsisten.

Untuk memvalidasi ringkasan deskriptif tersebut, dominasi antar-skenario kemudian dipadatkan melalui rekapitulasi *Win–Lose–Tie* berbasis hasil uji signifikansi *Paired t-test*. Pada ringkasan ini, "Win" merepresentasikan kondisi di mana suatu skenario secara statistik terbukti lebih baik dari skenario pembandingnya, "Lose" ketika terbukti lebih buruk, dan "Tie" ketika tidak terdapat perbedaan yang bermakna secara statistik ($p > 0,05$).

Tabel 4.14 Ringkasan Per Skenario (Win–Lose–Tie) Berdasarkan Uji Berpasangan

Skenario	Win	Lose	Tie	Total Uji	Net Win	Win Rate (%)
Class Weight Only	6	1	14	21	5	85.7
Baseline	3	1	17	21	2	75
Full Model	0	0	21	21	0	0
Attention Only	1	8	12	21	-7	11.1

Berdasarkan Tabel 4.14, *Class Weight Only* tervalidasi sebagai skenario paling dominan secara keseluruhan dengan mencatat jumlah kemenangan tertinggi (6 kali) dan hanya satu kali kekalahan, menghasilkan *win rate* sebesar 85,7%. Pola ini secara statistik sangat konsisten dengan pembahasan per metrik sebelumnya, di mana skenario ini berulang kali unggul secara signifikan terutama terhadap *Attention Only* pada metrik yang sensitif terhadap ketidakseimbangan kelas dan struktur *ordinal* (metrik makro dan QWK). *Baseline* menyusul di posisi kedua

dengan 3 kemenangan dan 1 kekalahan (75,0%), menunjukkan bahwa *model* dasar tanpa atensi masih lebih kompetitif dan rasional dibandingkan *model* yang hanya mengandalkan atensi murni.

Full Model sepenuhnya berada pada kondisi *Tie* (21 kali). Hal ini mengonfirmasi argumen sebelumnya bahwa peningkatan atau penurunan performa yang terjadi pada *model* hibrida ini murni diakibatkan oleh fluktuasi variansi yang ekstrem, sehingga tidak cukup konsisten secara statistik untuk menegaskan keunggulan maupun kelemahannya. Terakhir, *Attention Only* terbukti secara ilmiah sebagai konfigurasi yang paling tidak menguntungkan dengan jumlah kekalahan terbanyak (8 kali) dan nilai *Net Win* negatif (-7). Fakta statistik ini mengunci kesimpulan bahwa meskipun akurasinya tinggi dan *MAE*-nya rendah, arsitektur atensi tanpa pembobotan kelas memiliki kinerja yang sangat buruk dan tidak selaras dengan tujuan evaluasi klinis yang menuntut keadilan lintas tingkat keparahan depresi.

Untuk menutup ringkasan kuantitatif di atas, analisis kualitatif berikut disajikan guna memberikan contoh konkret mengenai bagaimana mekanisme pola prediksi *model* bekerja secara nyata pada tingkat linguistik. Tujuannya bukan untuk menggantikan validitas metrik, melainkan untuk memberikan "bukti naratif" yang menjabarkan mengapa metrik makro dan QWK dapat hancur meskipun akurasi global terlihat memuaskan. Perlu dicatat bahwa teks yang ditampilkan pada tabel di bawah ini telah melalui tahap *preprocessing*, sehingga karakteristik bahasanya telah disederhanakan melalui proses *lowercasing*, penghapusan tanda baca, serta pembersihan token *noise*. Teks ini merepresentasikan bentuk variabel train_clean,

yakni data riil yang diumpankan langsung ke dalam jaringan syaraf tiruan pada fase pelatihan dan pengujian.

Tabel 4.15 Contoh Prediksi *Model* pada Potongan Teks per Kelas (Analisis Kualitatif)

Text	**True Label**	***Baseline***	***Class Weight***	***Attention Only***	***Full Model***
resume thank you	*minimum*	***minimum***	***minimum***	***minimum***	***minimum***
youre losing control juststopthinking youre panicking stop panicking	*minimum*	***minimum***	***minimum***	***minimum***	***minimum***
money card work has a paycard option phone plan	*minimum*	***minimum***	***minimum***	***minimum***	***minimum***
this crippling pain is getting stronger cant you see i cant do this much longer this fearless drip the subconscious tears hope someone can see my fear	*mild*	*minimum*	*minimum*	*minimum*	*minimum*
i have noone to talk too stress ive taken dbtcbt classes yes ive tried grounding techniques and get frustrated after a bunch dont work cannot afford a pyscologist	*mild*	*minimum*	*minimum*	*minimum*	*minimum*
i cant have fun anymore i cant enjoy life anymore i dont know what to do this is hopeless i had to come home from work because i cant stop crying	*mild*	*minimum*	*minimum*	*minimum*	*minimum*
i knew something was up with me my thoughts were consuming me couldnt sleep stressed worried about anything and everything	*moderate*	*minimum*	*minimum*	*minimum*	*minimum*
also the headaches loads of headaches all the time im so done i	*moderate*	*minimum*	*minimum*	*minimum*	*minimum*

Text	**True Label**	***Baseline***	***Class Weight***	***Attention Only***	***Full Model***
hate this almost as bad as my brain constantly telling me im a pos anxiety is fun					
i woke up crying wtf is going on in my head that i dream such graphic scenes my abuse was mainly by my stepmom my dad was neglectful pretending nothing happened	*moderate*	*minimum*	*minimum*	*minimum*	*minimum*
i dont know i dont know what to do i just want out of here its too hard with this house and school work	*severe*	*minimum*	*minimum*	*minimum*	*minimum*
idk do i tell someone do i just quit do i talk to her about what she did please any advice would be really really helpful to me	*severe*	*minimum*	*minimum*	*minimum*	*minimum*
my chest has a different feeling before it would feel on fire and chaotic now it feels just wrong like i am in medical danger i am a year old woman	*severe*	*minimum*	*minimum*	*minimum*	*minimum*

Analisis kualitatif pada Tabel 4.15 membedah anatomi dari "paradoks akurasi" yang telah dibahas sebelumnya. Pada baris-baris awal yang mewakili kelas *Minimum*, seluruh *model* tanpa kecuali berhasil melakukan klasifikasi dengan benar. Pencapaian ini sepintas terlihat sebagai sebuah keberhasilan arsitektural. Namun, keberhasilan komunal ini sesungguhnya lebih banyak didikte oleh probabilitas statistik bawaan data di mana kelas *minimum* menguasai porsi mayoritas absolut dalam dataset. Selama proses iterasi pelatihan, *model*

"dibombardir" oleh pola-pola dari kelas ini, sehingga jaringan syaraf menjadi sangat fasih dalam mengenali batas keputusan untuk label *minimum*.

Ironi klasifikasi mulai terlihat ketika *model* dihadapkan pada teks dari kelas *Mild*, *Moderate*, dan *Severe*. Meskipun teks tersebut secara eksplisit mengandung muatan klinis yang sangat berat, seperti frasa *"crippling pain"*, *"hopeless"*, hingga indikasi pelecehan *"my abuse was mainly by my stepmom"*, seluruh varian *model* secara seragam mengklasifikasikan sampel tersebut ke dalam keranjang *minimum*. Kebutaan *model* terhadap konteks emosional yang intens ini merupakan bentuk bias mayoritas yang sangat ekstrem. Alih-alih mengekstraksi fitur semantik tingkat tinggi dari kelas minoritas, *model* memilih rute komputasi yang paling "aman" untuk menekan nilai fungsi kerugian (*loss*), yakni dengan menebak kelas yang paling sering muncul di data latih.

Kegagalan naratif ini adalah penjelasan logis dari temuan evaluasi kuantitatif sebelumnya. Tingginya nilai *Accuracy* yang dipamerkan oleh *model*, khususnya pada skenario *Attention Only*, hanyalah ilusi statistik yang dihasilkan dari keberhasilan *model* menjawab jutaan sampel *minimum* dengan benar, sambil sepenuhnya mengorbankan sampel *severe* dan *moderate* yang jumlahnya sedikit. Degradasi skor *Macro F1* dan QWK merupakan harga yang harus dibayar dari kegagalan identifikasi ini, menegaskan bahwa *model* belum sepenuhnya mampu memetakan ruang vektor bahasa secara berjenjang sesuai dengan struktur *ordinal* psikologis manusia. Hal ini membuktikan bahwa tanpa penyeimbang kelas (*class weight*), *model* hanya mengoptimalkan *local minima* yang menguntungkan statistik global (akurasi) namun mengorbankan sensitivitas diagnostik.

Sebagai konklusi, penelitian ini menyingkap realitas empiris bahwa pemodelan klasifikasi tingkat keparahan depresi masih menghadapi dinding keterbatasan yang curam, utamanya ketika dihadapkan pada ketidakseimbangan distribusi data yang sangat ekstrem. Meskipun mekanisme hibridasi *deep learning* seperti *Fasttext dan Bigru*, dan Atensi memiliki kemampuan teoretis untuk menangkap dependensi temporal dan muatan semantik teks, efektivitasnya dapat dengan mudah dilumpuhkan oleh bias frekuensi data.

Meskipun hasil evaluasi menunjukkan bahwa *model* klasifikasi ini masih menghadapi dinding keterbatasan akibat ketidakseimbangan distribusi data, temuan ini secara fundamental membuktikan satu hal krusial. Pendekatan *deep learning* berbasis *Natural Language Processing* (NLP) terbukti mampu menangkap pola bahasa, pilihan kata, dan kecenderungan makna yang berkaitan erat dengan kondisi psikologis seseorang. Hasil ekstraksi fitur ini memiliki potensi yang sangat besar untuk dikembangkan sebagai instrumen skrining awal yang proaktif.

Dalam kerangka pandang yang lebih luas, urgensi dari pengembangan sistem deteksi dini ini melampaui sekadar pencapaian teknis ilmu komputer. Mencegah eskalasi depresi berat yang memiliki risiko tinggi mengarah pada tindakan fatal seperti bunuh diri merupakan wujud nyata dari upaya menjaga jiwa manusia (*hifdzun nafs*). Hal ini merupakan salah satu tujuan tertinggi dari syariat Islam (*maqāṣid asy syarī'ah*) sebagaimana difirmankan oleh Allah SWT dalam Surat Al Ma'idah ayat 32:

...وَمَنْ اَحْيَاهَا فَكَاَنَّمَآ اَحْيَا النَّاسَ جَمِيْعًاۗ...

"...Dan barangsiapa yang memelihara kehidupan seorang manusia, maka seolah-olah dia telah memelihara kehidupan manusia semuanya..." (QS. Al-Ma'idah: 32)

Imam Al Qurthubi dalam kitab tafsirnya yang berjudul Al Jami li Ahkam Al Quran menguraikan bahwa menyelamatkan nyawa seseorang dari kebinasaan menempati kedudukan yang sangat mulia di sisi Allah (Al-Qurthubi, 2006). Kebinasaan ini mencakup ancaman fisik seperti tenggelam atau terbakar maupun ancaman dari penyakit yang mematikan. Mengembangkan sistem kecerdasan buatan yang mampu memberikan peringatan dini terhadap tingkat keparahan depresi adalah bentuk implementasi kontemporer dari upaya memelihara kehidupan tersebut. Jika satu nyawa dapat diselamatkan karena sistem ini berhasil mengklasifikasikan kondisi parah secara tepat sehingga penderita mendapatkan pertolongan medis tepat waktu, maka peneliti telah mengambil bagian dalam misi kemanusiaan agung yang disejajarkan dengan menyelamatkan seluruh umat manusia.

Islam mengajarkan kepedulian terhadap kondisi batin seseorang dan pentingnya mendeteksi kesedihan yang mendalam. Hal ini sebagaimana tergambar dalam kisah Nabi Yaqub yang mengadukan kesedihannya kepada Allah dalam Surat Yusuf ayat 86:

قَالَ اِنَّمَآ اَشْكُوْا بَثِّيْ وَحُزْنِيْٓ اِلَى اللّٰهِ وَاَعْلَمُ مِنَ اللّٰهِ مَا لَا تَعْلَمُوْنَ ﴿٨٦﴾

"Dia (Yaqub) menjawab, 'Hanya kepada Allah aku pengadukan kesusahan dan kesedihanku, dan aku mengetahui dari Allah apa yang tidak kamu ketahui.'" (QS. Yusuf: 86)

Dalam menafsirkan ayat ini, Quraish Shihab melalui Tafsir Al-Misbah menjelaskan bahwa pengaduan rasa sakit, kegelisahan yang memuncak (*baththh*), dan kesedihan (*huzn*) adalah bagian dari fitrah manusia ketika menghadapi tekanan psikologis yang teramat berat. Platform media sosial seperti Reddit saat ini sering

kali menjadi tempat bagi manusia modern untuk mengadukan kesedihannya layaknya sebuah munajat digital akibat keputusasaan. Penelitian ini hadir sebagai bentuk respons kewajiban kolektif atau *fardhu kifayah* untuk mendengar dan memahami keluhan tersebut menggunakan teknologi. Dengan mendeteksi tingkat keparahan depresi dari teks, kita berikhtiar membantu mereka yang sedang rapuh agar segera mendapatkan pertolongan sebelum kesedihan itu berubah menjadi keputusasaan yang fatal.

Secara spesifik, penggunaan teknologi pemrosesan bahasa alami untuk menganalisis pola kata dan makna dalam teks sejalan dengan isyarat Al Quran mengenai indikasi kondisi batin seseorang melalui gaya bicaranya. Allah berfirman dalam Surat Muhammad ayat 30:

وَلَوْ نَشَآءُ لَاَرَيْنٰكَهُمْ فَلَعَرَفْتَهُمْ بِسِيْمٰهُمْۗ وَلَتَعْرِفَنَّهُمْ فِيْ لَحْنِ الْقَوْلِۗ وَاللّٰهُ يَعْلَمُ اَعْمَالَكُمْ ﴿٣٠﴾

"Seandainya Kami berkehendak, niscaya Kami menunjukkan mereka kepadamu (Nabi Muhammad) sehingga engkau benar-benar dapat mengenali mereka melalui tanda-tandanya. Engkau pun benar-benar akan mengenali mereka melalui nada bicaranya. Allah mengetahui segala amal perbuatanmu." (QS. Muhammad: 30)

Ibnu Katsir dalam tafsirnya menjelaskan bahwa istilah *lahnil qaul* yang merujuk pada nada bicara atau gaya bahasa bermakna kandungan perkataan, gaya diksi, serta makna tersirat yang diucapkan seseorang. Isi hati dan kondisi psikologis seseorang pasti akan tercermin atau bocor melalui pilihan kata katanya (Katsir, 2004). Dalam konteks penelitian ini, algoritma FastText dan BiGRU bekerja untuk menangkap struktur kalimat tersebut. Sementara itu, mekanisme atensi berfungsi untuk memfokuskan bobot perhatian pada kata kata kunci tertentu yang merupakan representasi komputasional dari *lahnil qaul* yang mengindikasikan depresi berat.

Kata kata keputusasaan atau keinginan menyakiti diri merupakan bentuk nyata dari isyarat tersebut. Penelitian ini pada hakikatnya adalah upaya ilmiah untuk menerjemahkan isyarat *lahnil qaul* menjadi data kuantitatif yang akurat demi tujuan medis.

Selain itu, penelitian ini juga berlandaskan pada semangat optimisme bahwa setiap penyakit mental pasti memiliki solusi jika didiagnosis dengan benar. Rasulullah bersabda dalam riwayat Imam Ahmad:

تَدَاوَوْا عِبَادَ اللَّهِ فَإِنَّ اللَّهَ سُبْحَانَهُ لَمْ يَضَعْ دَاءً إِلاَّ وَضَعَ مَعَهُ شِفَاءً إِلاَّ الْهَرَمَ

"Berobatlah wahai hamba-hamba Allah, karena sesungguhnya Allah tidak menurunkan penyakit kecuali Dia menurunkan pula obatnya (penawarnya), kecuali satu penyakit, yaitu kepikunan (tua)." (HR. Ahmad No. 18454).

Dalam konteks kecerdasan buatan, sistem klasifikasi ini berperan sebagai alat diagnosis atau *wasilah* untuk menemukan *syifa* atau kesembuhan. Penggunaan *model* komputasi yang kompleks untuk memetakan gejala depresi adalah bentuk pemanfaatan akal karunia Allah untuk menyingkap solusi pengobatan yang tepat sasaran atas permasalahan umat.

Terakhir, penelitian ini merupakan manifestasi dari ayat ayat *kauniyah* yang memerintahkan manusia untuk menundukkan teknologi bagi kebaikan umat manusia melalui Surat Al Jatsiyah ayat 13:

وَسَخَّرَ لَكُمْ مَّا فِى السَّمٰوٰتِ وَمَا فِى الْاَرْضِ جَمِيْعًا مِّنْهُۗ اِنَّ فِيْ ذٰلِكَ لَاٰيٰتٍ لِّقَوْمٍ يَّتَفَكَّرُوْنَ ١٣

"Dan Dia menundukkan untukmu apa yang ada di langit dan apa yang ada di bumi semuanya, (sebagai rahmat) dari-Nya. Sesungguhnya pada yang demikian itu benar-benar terdapat tanda-tanda (kebesaran Allah) bagi kaum yang berpikir." (QS. Al-Jātsiyah: 13)

Tafsir Kementerian Agama RI (2019) menggarisbawahi frasa *yatafakkarun* atau kaum yang berpikir yang bermakna bahwa Allah menjadikan segala sesuatu di alam tunduk pada hukum hukumnya atau sunnatullah. Manusia dituntut menggunakan akal pikirannya untuk mengelola ketundukan tersebut. Matematika, vektor kata, perhitungan matriks probabilitas, hingga mekanisme atensi yang menjadi fondasi jaringan saraf tiruan pada dasarnya adalah bagian dari hukum alam dan logika yang Allah tundukkan bagi manusia. Menggunakannya untuk mendeteksi depresi adalah bentuk tafakur dan syukur atas nikmat akal beserta teknologi tersebut. Dengan demikian, penelitian ini secara harmonis memadukan kecanggihan teknologi informasi dengan nilai nilai luhur kemanusiaan dalam Islam serta membuktikan bahwa sains dan agama berjalan beriringan untuk menciptakan masyarakat yang lebih sehat secara mental dan spiritual.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini membuktikan bahwa dalam mengklasifikasikan empat tingkat keparahan depresi pada teks media sosial dengan ketidakseimbangan kelas yang ekstrem, penanganan fungsi kerugian (*loss function*) jauh lebih fundamental dibandingkan penambahan kompleksitas arsitektur pemodelan. Berdasarkan hasil pengujian empat skenario, konfigurasi *Class Weight Only* secara konsisten menjadi *model* paling optimal dan adil. Meskipun akurasi globalnya sedikit turun (70,30%), *model* ini mencatatkan performa terbaik pada metrik yang peka terhadap kelas minoritas, yaitu *Macro F1-Score* (32,51%) dan *Quadratic weighted kappa* (29,66%), yang menunjukkan kesesuaian terkuat dengan hierarki *ordinal* tingkat keparahan depresi.

Sebaliknya, skenario yang hanya mengandalkan mekanisme atensi (*Attention Only*) mengalami kegagalan sistematis dalam mengenali fitur kelas minoritas. Tingginya akurasi (72,60%) pada *model* ini terbukti bersifat manipulatif (paradoks akurasi), karena *model* hanya memusatkan pembobotannya pada kelas dominan dan mengabaikan sinyal linguistik dari indikasi depresi berat. Hal ini menegaskan bahwa tanpa intervensi penyeimbang kelas (*class weight*), mekanisme atensi justru akan memperburuk bias mayoritas dari data latih.

Meskipun penerapan pembobotan kelas mampu mengatasi dominasi data mayoritas, penelitian ini juga menemukan adanya batasan stabilitas pada arsitektur hibrida (*Full Model*). Penggabungan mekanisme atensi dan pembobotan kelas

secara bersamaan pada data teks yang terbatas justru memicu variansi performa yang sangat tinggi antar-lipatan pengujian. Hal ini mengindikasikan bahwa *model* klasifikasi depresi berbasis teks masih memerlukan eksplorasi lebih lanjut terkait metode kalibrasi fitur untuk menstabilkan konvergensi pembelajaran pada kategori data yang sangat minim.

5.2 Saran

Berdasarkan hasil penelitian yang telah disajikan, terdapat beberapa area yang masih dapat dikembangkan untuk meningkatkan kualitas dan keandalan model di masa depan. Meskipun pendekatan *Class Weight Only* terbukti paling konsisten dalam menangani ketidakseimbangan kelas, pengembangan lebih lanjut pada penelitian selanjutnya sangat diperlukan. Salah satu saran utamanya adalah penerapan metode klasifikasi dua tahap atau *2-step classification* pada arsitektur alur kerja sistem prediksi. Pada tahap pertama, model dapat difokuskan secara eksklusif untuk mendeteksi ada atau tidaknya indikasi depresi secara biner dari teks pengguna. Setelah sebuah teks terkonfirmasi memiliki indikasi depresi, tahap kedua baru dijalankan untuk mengklasifikasikan tingkat keparahannya ke dalam kategori yang berjenjang.

Selain itu, eksplorasi terhadap teknik penanganan data tidak seimbang perlu diperluas dengan membandingkan pendekatan lain di luar pembobotan kelas standar. Penelitian selanjutnya dapat menguji penggunaan *focal loss*, pembelajaran peka biaya atau *cost-sensitive learning* yang lebih terkontrol, serta strategi pengambilan sampel yang mampu meminimalkan distorsi distribusi fitur agar peningkatan performa pada pengenalan kelas minoritas dapat tetap stabil antar

lipatan pengujian. Mengingat label klasifikasi pada studi ini memiliki sifat berjenjang, pemodelan di masa depan juga sebaiknya dioptimalkan menggunakan metode yang secara khusus dirancang untuk menangani data ordinal. Pendekatan seperti regresi ordinal, klasifikasi berbasis ambang batas atau *threshold-based ordinal classification*, dan rancangan fungsi kerugian yang memberi penalti lebih besar untuk kesalahan prediksi berjarak jauh sangat layak untuk dicoba guna menstabilkan performa dan mencegah model melakukan lompatan kesalahan yang ekstrem.

Pada tahapan validasi pemodelan, evaluasi juga perlu ditingkatkan secara komprehensif untuk memastikan dan menilai ketahanan algoritma saat dihadapkan pada variasi teks di dunia nyata. Penelitian berikutnya sangat dianjurkan untuk menguji generalisasi model melalui pengujian lintas domain atau *cross-domain evaluation* menggunakan sumber data sekunder yang benar-benar terpisah guna melengkapi dan memperkuat hasil evaluasi yang sejauh ini hanya bergantung pada porsi data pengujian internal. Terakhir, penambahan fitur kalibrasi probabilitas dan pelaporan tingkat ketidakpastian prediksi sangat disarankan apabila model ini pada akhirnya diarahkan untuk implementasi sistem pendukung keputusan klinis. Fasilitas tersebut akan membantu pengguna sistem dalam memahami tingkat keyakinan prediksi model secara jauh lebih transparan, sekaligus memberikan alarm peringatan yang jelas mengenai kapan hasil tebakan otomatis tersebut memerlukan tinjauan manusia lebih lanjut sebelum sebuah keputusan diambil.

DAFTAR PUSTAKA

Allgaier, J., & Pryss, R. (2024). Practical approaches in evaluating validation and biases of machine learning applied to mobile health studies. *Communications Medicine*, *4*(1), 76. https://doi.org/10.1038/s43856-024-00468-0

Al-Qurthubi, S. M. bin A. (2006). *Al-Jami' li Ahkam Al-Qur'an*. Dar Al-Kutub Al-Ilmiyah.

Ariani, T. A., Anna, A., Rahayu, H. T., Aini, N., Windarwati, H. D., Hernawaty, T., Mudiyanselage, S. P. K., & Lin, M.-F. (2023). Psychometric testing of the indonesian version of beck depression inventory-ii among Indonesian floods survivors. *Jurnal Ners*, *18*(3), 264–273. https://doi.org/10.20473/jn.v18i3.47313

Atske, S. (2025, Juli 3). *Teens, social media and mental health*. Pew Research Center. https://www.pewresearch.org/internet/2025/04/22/teens-social-media-and-mental-health

Bahdanau, D., Cho, K., & Bengio, Y. (2015). *Neural Machine Translation by Jointly Learning to Align and Translate*. http://arxiv.org/abs/1409.0473

Baziotis, C., Pelekis, N., & Doulkeridis, C. (2017). *DataStories at SemEval-2017 Task 4: Deep LSTM with Attention for Message-level and Topic-based Sentiment Analysis*.

Beierle, F., Pryss, R., & Aizawa, A. (2023). Sentiments about Mental Health on Twitter—Before and during the COVID-19 Pandemic. *Healthcare (Switzerland)*, *11*(21). https://doi.org/10.3390/healthcare11212893

Bérchez-Moreno, F., Fernández, J. C., Hervás-Martínez, C., & Gutiérrez, P. A. (2024). Fusion of standard and ordinal dropout techniques to regularise deep models. *Information Fusion*, *106*, 102299. https://doi.org/10.1016/j.inffus.2024.102299

Beshay. (2025). *Americans' social media use*. https://www.pewresearch.org/internet/2024/01/31/americans-social-media-use/

Bhuiyan, M. I., Kamarudin, N. S., & Ismail, N. H. (2025). *Detecting Suicidal Ideation in Text with Interpretable Deep Learning: A CNN-BiGRU with Attention Mechanism*.

Boettcher, N. (2021). Studies of Depression and Anxiety Using Reddit as a Data Source: Scoping Review. *JMIR Mental Health*, *8*(11), e29487. https://doi.org/10.2196/29487

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). *Enriching Word Vectors with Subword Information*. http://www.isthe.com/chongo/tech/comp/fnv

Bürkner, P.-C., & Vuorre, M. (2019). Ordinal Regression Models in Psychology: A Tutorial. *Advances in Methods and Practices in Psychological Science*, *2*(1), 77–101. https://doi.org/10.1177/2515245918823199

Chai, C. P. (2023). Comparison of text preprocessing methods. *Natural Language Engineering*, *29*(3), 509–553. https://doi.org/DOI:10.1017/S1351324922000213

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. http://arxiv.org/abs/1412.3555

Dwarampudi, M., & Reddy, N. V. S. (2019). *Effects of padding on LSTMs and CNNs*.

Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, *18*(5), 602–610. https://doi.org/10.1016/j.neunet.2005.06.042

Gutierrez, P. A., Perez-Ortiz, M., Sanchez-Monedero, J., Fernandez-Navarro, F., & Hervas-Martinez, C. (2016). Ordinal Regression Methods: Survey and Experimental Study. *IEEE Transactions on Knowledge and Data Engineering*, *28*(1), 127–146. https://doi.org/10.1109/TKDE.2015.2457911

Huang, Y., Dai, X., Yu, J., & Huang, Z. (2023). SA-SGRU: Combining Improved Self-Attention and Skip-GRU for Text Classification. *Applied Sciences*, *13*(3), 1296. https://doi.org/10.3390/app13031296

Katsir, I. bin 'Umar. (2004). *Tafsir Al-Qur'an Al-'Azhim* (2 ed.). Dar Taibah.

Kementerian Agama RI. (2019). *Al-Qur'an dan Tafsirnya (Edisi Penyempurnaan)*. Lajnah Pentashihan Mushaf Al-Qur'an Badan Litbang dan Diklat Kementerian Agama RI. https://quran.kemenag.go.id/

Kemp, S. (2025). *5 billion social media users — DataReportal – Global Digital Insights*. https://datareportal.com/reports/digital-2024-deep-dive-5-billion-social-media-users

Kohlschütter, C., Fankhauser, P., & Nejdl, W. (2010). Boilerplate detection using shallow text features. *Web Search and Data Mining*. https://api.semanticscholar.org/CorpusID:10739339

McCullagh, P. (1980). Regression Models for Ordinal Data. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *42*(2), 109–127. https://doi.org/10.1111/j.2517-6161.1980.tb01109.x

Meyes, R., Lu, M., de Puiseau, C. W., & Meisen, T. (2019). *Ablation Studies in Artificial Neural Networks*. http://arxiv.org/abs/1901.08644

Mowery, D., Bryan, C., & Conway, M. (2015). Towards Developing an Annotation Scheme for Depressive Disorder Symptoms: A Preliminary Study using Twitter Data. *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, 89–98. https://doi.org/10.3115/v1/W15-1211

Naseem, U., Dunn, A. G., Kim, J., & Khushi, M. (2022). Early Identification of Depression Severity Levels on Reddit Using Ordinal Classification. *Proceedings of the ACM Web Conference 2022*, 2563–2572. https://doi.org/10.1145/3485447.3512128

Palomino, M. A., & Aider, F. (2022). Evaluating the Effectiveness of Text Pre-Processing in Sentiment Analysis. *Applied Sciences (Switzerland)*, *12*(17). https://doi.org/10.3390/app12178765

Pukelsheim, F. (1994). The Three Sigma Rule. *The American Statistician*, *48*(2), 88–91. https://doi.org/10.1080/00031305.1994.10476030

Reddit. (2025, Oktober 30). *Reddit Announces Third Quarter 2025 Results*. https://investor.redditinc.com/news-events/news-releases/news-details/2025/Reddit-Announces-Third-Quarter-2025-Results/

Shukla, D., & Dwivedi, S. K. (2024). The study of the effect of preprocessing techniques for emotion detection on Amazon product review dataset. *Social Network Analysis and Mining*, *14*(1), 191. https://doi.org/10.1007/s13278-024-01352-4

Siino, M., Tinnirello, I., & La Cascia, M. (2024). Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers. *Information Systems*, *121*, 102342. https://doi.org/10.1016/j.is.2023.102342

Tejaswini, V., Sathya Babu, K., & Sahoo, B. (2024). Depression Detection from Social Media Text Analysis using Natural Language Processing Techniques and Hybrid Deep Learning Model. *ACM Transactions on Asian and Low-Resource Language Information Processing*, *23*(1), 1–20. https://doi.org/10.1145/3569580

Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. *Information Processing & Management*, *50*(1), 104–112. https://doi.org/10.1016/j.ipm.2013.08.006

Vargas, V. M., Gutiérrez, P. A., & Hervás-Martínez, C. (2020). Cumulative link models for deep ordinal classification. *Neurocomputing*, *401*, 48–58. https://doi.org/10.1016/j.neucom.2020.03.034

Wong, T.-T., & Yeh, P.-Y. (2020). Reliable Accuracy Estimates from *k* -Fold Cross Validation. *IEEE Transactions on Knowledge and Data Engineering*, *32*(8), 1586–1594. https://doi.org/10.1109/TKDE.2019.2912815

World Health Organization. (2025a). *Depressive disorder (depression)*. https://www.who.int/news-room/fact-sheets/detail/depression

World Health Organization. (2025b). *Suicide worldwide in 2021: global health estimates*. https://www.who.int/publications/i/item/9789240110069

Xu, Y., & Goodacre, R. (2018). On Splitting Training and Validation Set: A Comparative Study of Cross-Validation, Bootstrap and Systematic Sampling for Estimating the Generalization Performance of Supervised Learning. *Journal of Analysis and Testing*, *2*(3), 249–262. https://doi.org/10.1007/s41664-018-0068-2

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016). *Hierarchical Attention Networks for Document Classification*.

Zhang, Z., Zhu, J., Guo, Z., Zhang, Y., Li, Z., & Hu, B. (2024). Natural Language Processing for Depression Prediction on Sina Weibo: Method Study and Analysis. *JMIR Mental Health*, *11*. https://doi.org/10.2196/58259

Zogan, H., Razzak, I., Jameel, S., & Xu, G. (2021). DepressionNet: Learning Multi-modalities with User Post Summarization for Depression Detection on Social Media. *SIGIR 2021 - Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 133–142. https://doi.org/10.1145/3404835.3462938