STUDI EKSPLORASI METODE PENGENALAN EMOSI UNTUK DETEKSI MULTI-EMOSI DALAM ENTRI JURNAL



Disusun Oleh:

N a m a : Aldi Sri Laksono
NIM : 21523008

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
2025

HALAMAN PENGESAHAN DOSEN PEMBIMBING

STUDI EKSPLORASI METODE PENGENALAN EMOSI UNTUK DETEKSI MULTI-EMOSI DALAM ENTRI JURNAL

TUGAS AKHIR

Disusun Oleh:

N a m a : Aldi Sri Laksono

NIM : 21523008

Yogyakarta, 21 Juli 2025

Pembimbing,

(Arrie Kurniawardhani, S.Kom., M.Kom)

HALAMAN PENGESAHAN DOSEN PENGUJI

STUDI EKSPLORASI METODE PENGENALAN EMOSI UNTUK DETEKSI MULTI-EMOSI DALAM ENTRI JURNAL

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 23 Juli 2025

Tim Penguji

Arrie Kurniawardhani, S.Kom., M.Kom _____

Anggota 1

Elyza Gustri Wahyuni, S.T., M.Cs. _____

Anggota 2

Kholid Haryono, S.T., M.Kom. _____

Mengetahui,
Ketua Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia

(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Aldi Sri Laksono

NIM : 21523008

Tugas akhir dengan judul:

STUDI EKSPLORASI METODE PENGENALAN EMOSI UNTUK DETEKSI MULTI-EMOSI DALAM ENTRI JURNAL

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 23 Juli 2025



(Aldi Sri Laksono)

HALAMAN PERSEMBAHAN

Segala puji dan syukur penulis panjatkan atas limpahan rahmat dan karunia dari Allah SWT yang telah memberikan kelancaran sehingga laporan tugas akhir ini dapat terselesaikan dengan baik. Shalawat serta salam senantiasa tercurahkan kepada junjungan Nabi besar Muhammad SAW sebagai suri teladan bagi seluruh umat manusia.

Dengan penuh rasa hormat dan syukur, laporan tugas akhir ini penulis persembahkan secara istimewa kepada kedua orang tua saya, Bapak dan Ibu, serta kepada seluruh anggota keluarga saya yang senantiasa memberikan doa, dukungan tulus, dan kesabaran tanpa batas dalam setiap langkah penulis selama masa perkuliahan.

Pada kesempatan ini, penulis juga mengucapkan terima kasih yang setulus-tulusnya kepada Ibu Arrie Kurniawardhani, S.Kom., M.Kom., selaku dosen pembimbing yang dengan sabar memberikan arahan, motivasi, dan ilmu yang sangat berharga. Ucapan terima kasih juga ditujukan kepada sahabat dan teman-teman seperjuangan yang selalu mendampingi, memberikan semangat, dan menjadi pendorong dalam proses penyelesaian tugas akhir ini. Tak lupa, penulis juga menyampaikan terima kasih kepada seluruh civitas akademika Universitas Islam Indonesia yang telah memberikan kesempatan, fasilitas, dan lingkungan belajar yang mendukung penulis untuk terus berkembang.

Semoga segala bentuk kebaikan, motivasi, dorongan, dan doa yang diberikan oleh seluruh pihak menjadi amal yang diberkahi dan mendatangkan kebahagiaan yang berlipat ganda. Aamiin ya Rabbal ‘Alamin.

HALAMAN MOTO

I am a fool in many ways, chasing things not meant for me, ignoring the signs, and overthinking small things. But somehow, even if I don’t always choose the right path, I hope there’s something beautiful and good in the way I try. - Mr. A

Maka, nikmat Tuhanmu manakah yang kamu dustakan? – Q.S. Ar-Rahman 55:13

KATA PENGANTAR

Assalamu'alaikum warahmatullahi wabarakatuh.

Puji syukur kehadirat Allah SWT atas segala rahmat, hidayah, dan karunia-Nya, sehingga penulis dapat menyelesaikan penyusunan skripsi yang berjudul "Studi Eksplorasi Metode Pengenalan Emosi untuk Deteksi Multi-Emosi dalam Entri Jurnal" dengan baik dan tepat waktu. Shalawat serta salam semoga senantiasa tercurah kepada junjungan kita, Nabi Muhammad SAW, beserta keluarga, sahabat, dan para pengikutnya hingga akhir zaman. Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer pada Program Studi Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.

Dalam proses penyusunan skripsi ini, penulis menyadari bahwa keberhasilannya tidak terlepas dari bimbingan, dukungan, dan semangat dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis ingin menyampaikan ucapan terima kasih yang tulus kepada:

1. Kedua orang tua tercinta, Sriyono dan Umi Soliah, terima kasih atas setiap doa yang tak pernah putus, pengorbanan yang tak ternilai, serta dukungan moril dan materiil yang selalu menyertai.
2. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc., sebagai Kepala Jurusan Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
3. Bapak Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D., selaku Ketua Program Studi Informatika Universitas Islam Indonesia.
4. Ibu Arrie Kurniawardhani, S.Kom., M.Kom., sebagai dosen pembimbing yang dengan sabar memberikan arahan, bimbingan, dan motivasi sepanjang penelitian ini.
5. Seluruh Bapak dan Ibu Dosen Program Studi Informatika yang telah berbagi ilmu dan pengetahuan yang sangat bermanfaat selama masa perkuliahan.
6. Sahabat dan teman-teman yang telah meluangkan waktu untuk menemani dan memberikan semangat dalam proses pengerjaan tugas akhir ini.
7. Semua pihak yang telah mendukung dan membantu, yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari sepenuhnya bahwa skripsi ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang bersifat membangun sangat penulis harapkan untuk perbaikan di masa mendatang. Semoga skripsi ini dapat memberikan manfaat dan menambah khazanah ilmu pengetahuan bagi para pembaca.

Wassalamu'alaikum warahmatullahi wabarakatuh.

Yogyakarta, 23 Juli 2025



(Aldi Sri Laksono)

SARI

Penelitian ini bertujuan untuk mengatasi tantangan dalam deteksi emosi berbasis teks, khususnya pada kasus multi-emosi atau dalam metodenya disebut multilabel emosi di mana satu teks dapat mengekspresikan kombinasi emosi yang kompleks, serta masalah ketidakseimbangan label minoritas dalam data. Untuk mencapai tujuan ini, penelitian melakukan eksplorasi mendalam terhadap karakteristik dan kinerja berbagai model *machine learning* dan *deep learning* berbasis *transformer*. Metodologi penelitian dimulai dengan pengumpulan data dari dataset publik "*Journal Entries with Labelled Emotions*" yang berisi 1473 entri dengan 18 kategori emosi. Pada penelitian ini, dilakukan *generative oversampling* menggunakan *LLM GPT-4o* yang menghasilkan 500 data sintetis, untuk mengatasi ketimpangan data, sehingga total menjadi 1.973 data. Penelitian membandingkan performa empat model *machine learning* (*Naïve Bayes, Logistic Regression, SVM, RFC*) berbasis TF-IDF dan empat model *deep learning* berbasis *transformer* (*BERT, BERT-LSTM, RoBERTa, DistilBERT*). Evaluasi model dilakukan secara komprehensif dengan berbagai metrik akurasi (*Jaccard Accuracy, Subset Accuracy, Label-wise Accuracy*), *Precision*, *Recall*, *F1-Score, Confusion Matrix*, dan *Inference* Time. Pada kategori *machine learning* dan *deep learning* menggunakan data asli, seluruh model menunjukkan performa yang kurang baik pada data uji. Hasil teknik *generative oversampling* secara signifikan meningkatkan kinerja seluruh model, baik *machine learning* maupun *deep learning*, karena model menjadi lebih baik dalam mengatasi ketidakseimbangan kelas dan memprediksi label minoritas secara lebih merata. Dalam kategori *machine learning*, model *SVM* menunjukkan performa paling unggul, dengan *Jaccard Accuracy 0.45* dan mencapai *Macro F1-Score* tertinggi 0.64 setelah *oversampling*, dan juga relatif cepat dalam membuat prediksi (4-12 milidetik). Untuk kategori *deep learning*, semua model menunjukkan peningkatan kinerja setelah *oversampling*, dengan *Macro F1-Score* berkisar antara 0.67-0.69. Secara spesifik, *RoBERTa* dengan *Jaccard Accuracy* 0.50 unggul dalam pemerataan prediksi emosi, termasuk label minoritas yang sulit dikenali, sedangkan *DistilBERT* menjadi yang tercepat dalam waktu inferensi (sekitar 15 milidetik).

Kata kunci: Deteksi Emosi, Multilabel, Oversampling, Machine Learning, Deep Learning.

GLOSARIUM

Dataset	Kumpulan data yang digunakan sebagai sumber informasi dalam proses pelatihan dan pengujian model. Dataset ini berisi entri teks beserta label.
Deep Learning	Cabang dari machine learning yang menggunakan jaringan saraf tiruan (*neural networks*) dengan banyak lapisan (*deep*) untuk mempelajari representasi fitur secara otomatis dari data dalam jumlah besar.
Emosi	Respons psikologis yang dihasilkan dari interpretasi suatu peristiwa atau teks.
Evaluasi	Proses pengukuran kinerja model menggunakan metrik tertentu terhadap data uji.
Machine Learning	Bidang kecerdasan buatan yang memungkinkan komputer untuk belajar dari data dan membuat prediksi atau keputusan tanpa diprogram secara eksplisit.
Multilabel	Pendekatan klasifikasi di mana satu entri data atau teks dapat memiliki lebih dari satu label atau kategori secara bersamaan.
Overfitting	Kondisi di mana model machine learning atau deep learning mempelajari data latih terlalu baik, sehingga performanya menurun saat dihadapkan pada data baru atau data uji.
Oversampling	Teknik penyeimbangan data yang dilakukan dengan cara menambahkan data tiruan pada kelas minoritas agar distribusi label menjadi lebihseimbang dan model tidak bias terhadap kelas mayoritas.
Pelatihan	Proses di mana model machine learning mempelajari pola dari data latih (training data) dengan tujuan mengoptimalkan prediksi terhadap data baru.
Uji	Tahap pengujian model terhadap data yang belum pernah dilihat sebelumnya untuk menilai generalisasi model terhadap data nyata.
Validasi	Tahap dalam pelatihan model deep learning yang digunakan untuk menyetel parameter dan mencegah overfitting dengan memeriksa kinerja model terhadap subset data yang tidak dilatih.

DAFTAR ISI

HALAMAN JUDUL i
HALAMAN PENGESAHAN DOSEN PEMBIMBING ii
HALAMAN PENGESAHAN DOSEN PENGUJI iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR iv
HALAMAN PERSEMBAHAN v
HALAMAN MOTO vi
KATA PENGANTAR vii
SARI ix
GLOSARIUM x
DAFTAR ISI xi
DAFTAR TABEL xiv
DAFTAR GAMBAR xv
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Batasan Masalah 3
1.4 Tujuan Penelitian 4
1.5 Manfaat Penelitian 4
1.6 Sistematika Penulisan 4
BAB II KAJIAN PUSTAKA 6
2.1 Deteksi Emosi pada Teks 6
2.2 Klasifikasi Multilabel 6
2.3 Generatif Oversampling 6
2.4 Model Machine Learning 7
2.4.1 Naïve Bayes 8
2.4.2 Logistic Regression 9
2.4.3 Support Vector Machine 9
2.4.4 Random Forest Classifier 10
2.5 Model Deep Learning 11
2.5.1 BERT 12
2.5.2 BERT-LSTM 14
2.5.3 RoBERTa 15

2.5.4 DistilBERT 16
2.6 Hyperparameter 16
2.7 Metrik Evaluasi 17
2.7.1 Akurasi 18
2.7.2 Precision 19
2.7.3 Recall 20
2.7.4 F1-Score 21
2.7.5 Confusion Matrix 21
2.7.6 Waktu Inferensi 22
2.8 Kajian Literatur 22
BAB III METODOLOGI PENELITIAN 26
3.1 Tahapan Penelitian 26
3.2 Pengumpulan Data 26
3.3 Oversampling Data 27
3.4 Pembagian Data 28
3.5 Pembuatan Model 28
3.6 Evaluasi Model 31
BAB IV HASIL DAN PEMBAHASAN 32
4.1 Hasil Pengumpulan Data 32
4.2 Hasil Oversampling Data 35
4.3 Hasil Pembagian Data 37
4.4 Hasil Pembuatan Model 41
4.4.1 Hasil Model Machine Learning 41
4.4.2 Hasil Model Deep Learning 42
4.5 Hasil Evaluasi Model 46
4.5.1 Hasil Evaluasi Model Machine Learning 48
4.5.2 Hasil Evaluasi Model Deep Learning 56
4.6 Pembahasan 64
4.6.1 Pembahasan Karakteristik Data 64
4.6.2 Pembahasan Model Machine Learning 66
4.6.3 Pembahasan Model Deep Learning 66
BAB V KESIMPULAN DAN SARAN 68
5.1 Kesimpulan 68
5.2 Saran 70

DAFTAR PUSTAKA.....71
LAMPIRAN 75

DAFTAR TABEL

Tabel 2.1 Penelitian terdahulu 22
Tabel 3.1 Hyperparameter 30
Tabel 4.1 Dataset Journal Diary.....34
Tabel 4.2 Contoh perbedaan data sintetis dan data asli 36
Tabel 4.3 Tabel pembagian dataset asli model Machine Learning.....38
Tabel 4.4 Tabel pembagian dataset asli model Deep Learning 39
Tabel 4.5 Tabel pembagian dataset hasil oversampling model Machine Learning.....39
Tabel 4.6 Tabel pembagian dataset hasil oversampling model Deep Learning 40
Tabel 4.7 Tabel hasil pembuatan model machine learning menggunakan data asli.....41
Tabel 4.8 Tabel hasil pembuatan model machine learning menggunakan data hasil oversampling.....41
Tabel 4.9 Tabel hasil pembuatan model deep learning menggunakan data asli.....45
Tabel 4.10 Tabel hasil pembuatan model deep learning menggunakan data hasil oversampling 46
Tabel 4.11 Tabel hasil evaluasi model machine learning.....54
Tabel 4.12 Tabel hasil skenario pengujian model machine learning.....55
Tabel 4.13 Tabel hasil evaluasi model deep learning.....62
Tabel 4.14 Tabel hasil skenario pengujian model deep learning.....62

DAFTAR GAMBAR

Gambar 2.1 Cara kerja Naïve Bayes 8
Gambar 2.2 Linear Regression dan Logistic Regression 9
Gambar 2.3 Model Support Vector Machine (SVM) 10
Gambar 2.4 Model Random Forest Classifier 11
Gambar 2.5 Model BERT 13
Gambar 2.6 Model BERT-LSTM 15
Gambar 2.7 Model DistilBERT 16
Gambar 3.1 Diagram Alir Tahapan Penelitian. 26
Gambar 4.1 Persebaran Label Emosi Pada Dataset *Journal Entries with Labelled Emotions*. 35
Gambar 4.2 Persebaran label emosi pada dataset hasil oversampling. 37
Gambar 4.3 Kode program untuk splitting data pada model Machine Learning. 38
Gambar 4.4 Kode program untuk splitting data pada model Deep Learning. 38
Gambar 4.5 Visualisasi hasil pelatihan akurasi dan loss model deep learning pada data asli. 43
Gambar 4.6 Visualisasi hasil traning akurasi dan loss model deep learning pada data hasil oversampling. 44
Gambar 4.7 Confussion matrix model naive bayes pada data asli dan data hasil oversampling. 49
Gambar 4.8 Confussion matrix model logistic regression pada data asli dan data hasil oversampling. 50
Gambar 4.9 Confussion matrix model SVM pada data asli dan data hasil oversampling. 51
Gambar 4.10 Confussion matrix model RFC pada data asli dan data hasil oversampling 52
Gambar 4.11 Confusion matrix model BERT pada data asli dan data hasil oversampling. 57
Gambar 4.12 Confusion matrix model BERT-LSTM pada data asli dan data hasil oversampling. 58
Gambar 4.13 Confusion matrix model RoBERTa pada data asli dan data hasil oversampling. 59
Gambar 4.14 Confusion matrix model distilBERT pada data asli dan data hasil oversampling. 60

BAB I
PENDAHULUAN

1.1 Latar Belakang

Emosi merupakan aspek penting dalam interaksi manusia, yang memengaruhi pengambilan keputusan, perilaku sosial, dan kesejahteraan psikologis. Oleh karena itu, analisis emosi berbasis teks menjadi salah satu topik yang mendapatkan perhatian dikalangan penelitian saat ini. Hal ini tidak mengherankan, mengingat fakta bahwa emosi adalah aspek fundamental dari interaksi manusia yang berdampak signifikan pada perilaku, pengambilan keputusan, dan kondisi psikologis, dengan aplikasi penting di ranah media sosial, layanan pelanggan, hingga kesehatan mental (L. Li, 2022).

Meskipun demikian, banyak model analisis emosi awal mengasumsikan bahwa setiap teks hanya mewakili satu emosi utama, sebuah pendekatan yang dikenal sebagai klasifikasi *single label*. Pendekatan ini, meskipun berguna untuk kasus-kasus sederhana, memiliki kelemahan mendasar ketika dihadapkan pada kompleksitas ekspresi manusia yang sesungguhnya. Dalam praktiknya, asumsi bahwa sebuah teks hanya mengekspresikan satu emosi tunggal seringkali tidak valid atau tidak sesuai dengan kenyataan. Sebagai contoh, ketika sebuah teks mengandung beberapa emosi misalnya, rasa lega yang bercampur dengan kebahagiaan setelah melewati tantangan berat klasifikasi *single label* dipaksa untuk memilih hanya satu, sehingga mengabaikan emosi lainnya yang mungkin sama pentingnya. Hal ini tidak hanya menyederhanakan realitas secara berlebihan, tetapi juga dapat menyebabkan misrepresentasi keadaan emosional penulis yang sebenarnya.

Sebaliknya, teks sering kali menjadi wadah bagi kombinasi emosi (multi-emosi) yang kompleks dan muncul bersamaan, sebuah fenomena yang dikenal sebagai emosi multilabel yang mana satu teks dapat memiliki lebih dari satu label emosi (Rahman, Ta, Najjar, Azadmanesh, & Gönul, 2024). Sebagai ilustrasi, pertimbangkan kalimat I was able to spend some time with my family. *I helped out my grandmother, and spent time alone with each kid to just talk.*" Kalimat ini dapat mengandung emosi "*calm*", "*happy*", dan "*satisfied*" (Malhotra, 2021). Korelasi antar label emosi, seperti antara "*calm*", "*happy*", dan "*satisfied*", membuat tugas model yang diinginkan menjadi semakin kompleks, karena model tidak hanya dituntun untuk mengenali satu emosi dominan maupun label sentimen seperti postif, negatif, dan netral, tetapi juga harus mampu memahami interaksi antar berbagai emosi dalam satu konteks teks (Bashynska, Sarafanov, & Manikaeva, 2024).

Tantangan utama dalam klasifikasi multilabel bersumber dari karakteristik data tekstual yang kompleks, mencakup struktur kalimat yang rumit hingga masalah label minoritas atau label yang sedikit muncul pada data. Struktur kalimat yang rumit tidak hanya mengenai campuran panjang dan pendek kalimat antara data melainkan juga isi atau makna dari kalimat yang biasanya terdapat kontradiksi antar katanya sehingga membuat model harus mampu memahami hubungan antar klausa ini untuk mengenali kedua emosi, bukan hanya salah satunya. Selain itu, masalah pada label minoritas dikarenakan jumlah label yang banyak sehingga terdapat kondisi di mana distribusi label emosi dalam dataset tidak seimbang. Kinerja yang buruk pada label minoritas akan menyebabkan model menjadi bias, yang akan cenderung meninggikan prediksi pada label mayoritas. Akibatnya, kemampuan model untuk mendeteksi spektrum emosi yang luas menjadi sangat terbatas. Kekompleksan ini menjadi urgensi karena emosi manusia sendiri pada dasarnya bersifat majemuk dan seringkali tidak dapat diwakili oleh satu kategori tunggal (Coppini, Lucifora, Vicario, & Gangemi, 2023). Oleh karena itu, diperlukan model yang efektif harus mampu memahami hubungan antar label, mengerti makna kata sesuai konteks, dan menyaring informasi yang paling relevan dengan emosi dari seluruh teks (W. Liu, Pang, Li, Zhou, & Yue, 2021)

Untuk menjawab tantangan dalam memahami ekspresi perasaan yang kompleks ini, penelitian ini berfokus pada fenomena multi emosi dengan memandang klasifikasi multilabel emosi sebagai pendekatan utama. Untuk menangani tugas ini, tujuan penelitian bukan sekadar memperbandingkan machine learning dan deep learning secara langsung, melainkan melakukan eksplorasi komparatif terhadap berbagai model di dalam masing-masing lingkup. Pada ranah machine learning, model-model seperti *Naïve Bayes, Logistic Regression, Support Vector Machine*, atau *Random Forest* dapat menawarkan arsitektur sederhana dan efisiensi. Sementara pada ranah deep learning, dengan arsitektur *transformer* seperti BERT yang memiliki kemampuan menangkap konteks kata yang lebih dalam namun membutuhkan sumber daya komputasi yang lebih tinggi dan arsitektur yang lebih kompleks. Dengan cara ini, penelitian tidak hanya menggali fenomena multi-emosi, tetapi juga mengevaluasi bagaimana karakteristik masing-masing model, baik dalam keluarga machine learning maupun deep learning, memengaruhi performa dalam menghadapi kompleksitas data multilabel. Penelitian ini diharapkan dapat memberikan pemahaman yang lebih komprehensif mengenai model mana yang lebih sesuai dengan karakteristik data jurnal berlabel emosi, sekaligus menjadi acuan bagi penelitian lanjutan dalam pengembangan sistem analisis emosi berbasis teks.

Relevansi penelitian ini diperkuat oleh penggunaan dataset '*Journal Entries with Labelled Emotions*', yang secara spesifik terdiri dari entri jurnal. Entri jurnal adalah catatan pribadi mengenai pemikiran, perasaan, dan pengalaman seseorang. Tipe data ini sangat cocok untuk penelitian multi emosi karena karakteristiknya yang berupa teks campuran baik panjang, pendek, dan naratif, yang secara alami mengandung ekspresi emosi yang beragam, berbeda dengan teks pendek yang umum di media sosial. Dataset ini juga memiliki sifat multi-emosi atau multilabel yang dimana memiliki 18 emosi yang bergama dari *happy, anxious*, hingga *sad,* dan berbagai macam label emosi lainnya. Karakteristik dataset ini membuka peluang untuk mengevaluasi secara rinci karakteristik dari masing-masing metode terhadap dataset tersebut, serta mencari metode mana yang paling efektif dalam menangani variasi multi-emosi atau multilabel emosi yang kompleks.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, penelitian ini dirumuskan untuk menjawab beberapa pertanyaan berikut:

a. Apa perbedaan karakteristik masing-masing model berdasarkan evaluasi performa dari masing-masing model baik model machine learning maupun model deep learning dalam tugas deteksi emosi multilabel berbasis teks?
b. Model machine learning dan deep learning manakah yang paling sesuai untuk deteksi emosi teks multilabel, dengan mempertimbangkan berbagai hasil evaluasi, skenario pengujian, dan karakteristik model?

1.3 Batasan Masalah

Agar penelitian ini lebih terarah dan fokus, beberapa batasan yang diterapkan dalam penelitian ini adalah sebagai berikut:

a. Model yang digunakan berfokus pada penggunaan dan optimasi dari model machine learning dan deep learning berbasis *transformer.*
b. Data yang digunakan pada penelitian ini menggunakan bahasa inggris yang berasal dari sumber dataset Journal Entries with Labelled Emotions.
c. Label emosi yang digunakan pada penelitian ini terbatas pada 18 kategori emosi, yaitu *Happy, Satisfied, Calm, Proud, Excited, Frustrated, Anxious, Surprised, Nostalgic, Bored, Sad, Angry, Confused, Disgusted, Afraid, Embarrassed, Awkward*, dan *Envious.*

d. Evaluasi performa model difokuskan pada metrik akurasi, *precision*, *recall*, *f1-score*, *confussion matrix,* dan *inference time*

1.4 Tujuan Penelitian

Selaras dengan rumusan masalah yang telah diuraikan, penelitian ini memiliki beberapa tujuan utama sebagai berikut:

a. Mengeksplorasi perbedaan karakteristik antara masing-masing model machine learning dan model deep learning dalam tugas deteksi emosi teks multilabel berdasarkan hasil evaluasi performa.
b. Menentukan dan memberikan rekomendasi dari masing-masing model machine learning atau deep learning yang paling sesuai untuk deteksi emosi teks multilabel, berdasarkan analisis terhadap berbagai hasil evaluasi, skenario pengujian, dan karakteristik model.

1.5 Manfaat Penelitian

Penelitian ini berperan penting dalam pengembangan teknologi pemrosesan bahasa alami atau *Natural Languange Processing* (NLP), khususnya untuk deteksi emosi multilabel. Hasil dari penelitian ini diharapkan dapat memberikan rekomendasi model terbaik antara masing-masing model machine learning dan deep learning, guna meningkatkan kemampuan sistem dalam memahami berbagai emosi yang kompleks dalam teks. Dengan demikian, temuan ini dapat diterapkan secara praktis dalam berbagai bidang seperti analisis umpan balik pelanggan, pengembangan aplikasi kesehatan mental yang lebih sensitif, dan pemantauan opini publik di media sosial, dengan tujuan akhir untuk memahami berbagai macam emosi yang dirasakan seseorang secara bersamaan dengan lebih akurat menggunakan masing-masing model tersebut.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penelitian ini akan dibagi menjadi lima bab utama, dengan rincian sebagai berikut:

a. BAB I PENDAHULUAN

   Pada bab ini akan diuraikan latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, serta sistematika penulisan yang memberikan gambaran umum mengenai keseluruhan alur penelitian.

b. BAB II KAJIAN PUSTAKA

Bab ini membahas teori-teori dasar yang relevan dengan penelitian ini, serta mengkaji penelitian-penelitian terdahulu yang bertujuan untuk mengetahui perbedaan dan pembaruan penelitian ini berdasarkan dengan penelitian yang telah ada.

c. BAB III METODE PENELITIAN

Bab ini memaparkan langkah-langkah metodologis yang dilakukan dalam penelitian. Tahapan tersebut meliputi proses pengumpulan data, oversampling data, pembagian data, perancangan arsitektur model machine learning dan deep learning, skenario pengujian, serta metrik evaluasi yang digunakan untuk mengukur dan membandingkan performa setiap model.

d. BAB IV HASIL DAN PEMBAHASAN

Pada bab ini disajikan hasil dari eksperimen yang telah dilakukan. Bab ini akan memaparkan hasil kinerja dari setiap model machine learning dan deep learning, melakukan analisis perbandingan secara mendalam, serta membahas temuan-temuan penting terkait keunggulan dan kelemahan masing-masing pendekatan dalam konteks deteksi multi-emosi.

e. BAB V PENUTUP

Bab ini berisi kesimpulan yang menjawab rumusan masalah dari keseluruhan hasil penelitian, serta menyajikan saran yang dapat digunakan untuk pengembangan penelitian di masa mendatang.

BAB II
KAJIAN PUSTAKA

2.1 Deteksi Emosi pada Teks

Emosi adalah bagian esensial dari diri manusia dan cara kita bertahan hidup. Emosi didefinisikan sebagai keadaan perasaan kompleks yang menghasilkan reaksi fisik dan psikologis, memengaruhi baik pemikiran maupun perilaku. Dalam model *Hourglass of Emotions*, emosi pada dasarnya 24 emosi seperti *happy, angry, sad, calm, surprise,* dan sebagainya (Cambria, Livingstone, & Hussain, n.d.). Untuk mengenali semua emosi tersebut maka diperlukan sebuah deteksi emosi.

Deteksi emosi adalah proses untuk mengenali jenis emosi yang dirasakan atau diungkapkan oleh seseorang, seperti senang, sedih, marah, takut, dan sebagainya. Dalam dunia teknologi, deteksi emosi menjadi salah satu topik penting, karena emosi sangat berperan dalam cara manusia berpikir, mengambil keputusan, dan berinteraksi. Salah satu bentuk penerapan dari deteksi emosi adalah melalui teks. Deteksi emosi pada teks termasuk bagian dari bidang NLP, dan sering dianggap sebagai langkah lanjutan dari analisis sentimen. Jika analisis sentimen hanya mengklasifikasikan tulisan sebagai positif, negatif, atau netral, maka deteksi emosi mencoba menggali lebih dalam untuk mengenali jenis emosi yang lebih spesifik (Seyeditabari, Tabari, & Zadrozny, 2018).

2.2 Klasifikasi Multilabel

Teks multilabel merujuk pada jenis data teks yang memiliki lebih dari satu label atau kategori yang relevan dalam satu unit data. Artinya, sebuah kalimat, paragraf, atau dokumen tidak hanya merepresentasikan satu makna, tema ataupun label tunggal, tetapi bisa mencerminkan berbagai topik, emosi, atau konteks secara bersamaan. Hal ini mencerminkan kompleksitas bahasa manusia yang sering kali tidak bersifat tunggal atau terpisah-pisah, melainkan saling tumpang tindih (Tsoumakas & Katakis, 2009).

2.3 Generatif Oversampling

Dalam ranah pembelajaran mesin, tantangan umum yang sering dihadapi adalah ketersediaan data yang tidak memadai atau tidak seimbang, terutama untuk kelas minoritas atau label yang memiliki sedikit data. Data yang tidak seimbang dapat menyebabkan model cenderung memihak pada kelas mayoritas atau label yang memiliki lebih banyak data,

menghasilkan kinerja yang buruk pada kelas minoritas, dan mengurangi kemampuan generalisasi model. Untuk mengatasi hal ini, teknik oversampling telah lama menjadi strategi yang efektif, dengan cara meningkatkan representasi kelas minoritas dalam dataset.

Oversampling data adalah serangkaian teknik yang digunakan untuk menyeimbangkan distribusi kelas dalam sebuah dataset dengan cara memperbanyak jumlah sampel pada kelas minoritas. Tujuannya adalah untuk memberikan bobot atau data yang lebih setara bagi kelas minoritas selama proses pelatihan model, sehingga model dapat belajar mengenali pola dari kedua kelas secara lebih efektif. Metode tradisional seperti SMOTE (Synthetic Minority Over-sampling Technique) dan ADASYN (Adaptive Synthetic Sampling) telah digunakan untuk menghasilkan sampel sintetis melalui interpolasi linier atau adaptif dari tetangga terdekat. Namun, metode ini memiliki keterbatasan dalam menciptakan keragaman dan realisme data yang memadai (Wibowo & Fatichah, 2021).

Munculnya Large Language Models (LLMs) revolusioner, seperti GPT-4, Gemini, dan LLaMA-2, telah membuka peluang baru untuk mengotomatisasi proses anotasi dan sintesis data yang rumit. LLM dapat menghasilkan data sintetis yang relevan secara kontekstual, dan meniru distribusi data asli. Ini menjadi dasar bagi generative oversampling, di mana LLM dimanfaatkan untuk menghasilkan sampel tambahan bagi kelompok data atau label yang kurang terwakili (Nakada, Xu, Li, & Zhang, 2025).

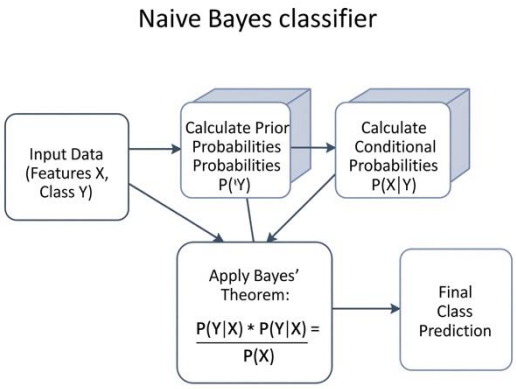
2.4 Model Machine Learning

Machine Learning merupakan cabang dari kecerdasan buatan (*Artificial Intelligence*) yang memungkinkan sistem untuk belajar secara otomatis dari data tanpa perlu diprogram secara eksplisit. Pada dasarnya, machine learning bekerja dengan membangun model matematis berdasarkan data yang diberikan, kemudian menggunakan model tersebut untuk melakukan prediksi atau pengambilan keputusan pada data baru. Proses ini melibatkan dua tahap utama, yaitu pelatihan (*training*) dan pengujian (*testing*). Pada tahap pelatihan, algoritma machine learning akan menganalisis pola dari data masukan dan labelnya untuk mempelajari hubungan di antara keduanya. Setelah model terbentuk, tahap pengujian dilakukan untuk mengevaluasi kemampuan model dalam memprediksi label dari data yang belum pernah dilihat sebelumnya. Machine learning memiliki beberapa pendekatan, salah satunya adalah *supervised learning*, yang digunakan dalam penelitian ini, di mana data dilengkapi dengan label sebagai acuan untuk pelatihan model. Keunggulan utama machine learning terletak pada kemampuannya untuk mengolah data dan menemukan pola kompleks yang sulit ditangkap oleh

pendekatan pemrograman manual yang mengandalkan aturan-aturan yang ditentukan secara eksplisit oleh manusia (Parsanna, 2025).

2.4.1 Naïve Bayes

Naive Bayes Classifier adalah sebuah metode klasifikasi probabilistik yang bekerja berdasarkan prinsip Teorema Bayes, dengan sebuah asumsi dasar yang kuat yang memberinya nama "naive" atau lugu. Asumsi ini menyatakan bahwa semua fitur atau variabel prediktor dalam data bersifat independen, artinya keberadaan atau nilai satu fitur tidak dianggap memengaruhi fitur lainnya dalam menentukan hasil kelas (Rish, 2001). Meskipun asumsi independensi ini jarang benar dalam konteks dunia nyata, pendekatan ini tetap menunjukkan performa yang kompetitif dalam banyak kasus.

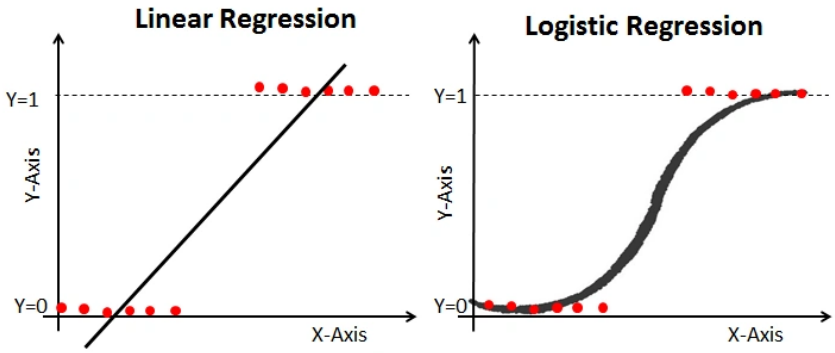


Gambar 2.1 Cara kerja Naïve Bayes

Cara kerja Naive Bayes dimulai dengan menghitung probabilitas prior dari setiap kelas dan probabilitas kondisional dari setiap fitur terhadap kelas tersebut. Kemudian, untuk data baru, model menghitung kemungkinan masing-masing kelas dan memilih kelas dengan probabilitas tertinggi sebagai hasil prediksi. Terdapat beberapa varian dari Naive Bayes, seperti Bernoulli Naive Bayes untuk fitur biner, Gaussian Naive Bayes untuk data kontinu, dan Multinomial Naive Bayes yang cocok untuk data teks berbasis frekuensi kata (H. Zhang, 2004).

2.4.2 Logistic Regression

Logistic Regression adalah salah satu algoritma supervised learning yang digunakan untuk tugas klasifikasi, terutama klasifikasi biner, meskipun dapat diperluas ke multikelas dan multilabel.



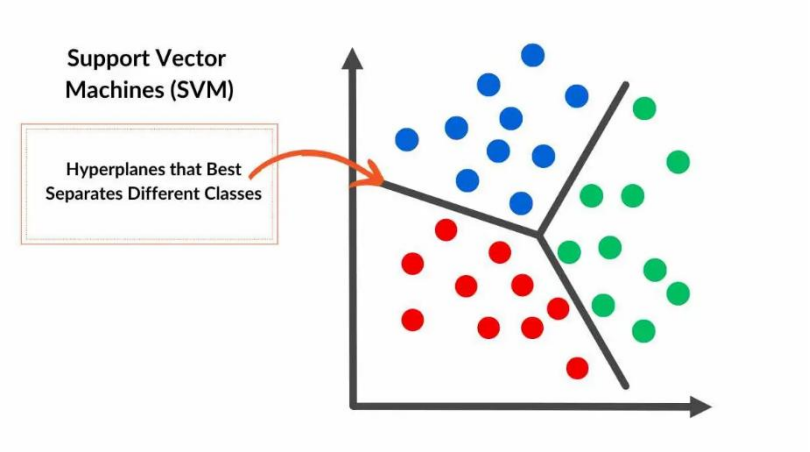
Gambar 2.2 Linear Regression dan Logistic Regression

Sumber: (Anshul, 2025)

Pada Gambar 2.2 dapat dilihat, perbedaan dengan linear regression yang digunakan untuk memprediksi nilai kontinu, logistic regression memodelkan hubungan antara variabel input dan probabilitas kejadian suatu kelas menggunakan fungsi logistik atau sigmoid. Fungsi sigmoid mengubah output linier dari kombinasi bobot dan fitur menjadi nilai antara 0 dan 1, yang kemudian ditafsirkan sebagai probabilitas suatu sampel termasuk dalam kelas tertentu. Jika probabilitas tersebut melebihi ambang batas tertentu (biasanya 0.5), maka sampel diklasifikasikan ke dalam kelas positif; jika tidak, ke kelas negatif. Logistic regression bekerja dengan meminimalkan fungsi log-loss atau binary cross-entropy untuk mengoptimalkan parameter model. Meskipun sederhana, logistic regression sangat efektif dalam kasus-kasus klasifikasi dengan data yang linier atau hampir linier, dan sering digunakan sebagai model baseline dalam berbagai masalah klasifikasi termasuk analisis sentimen, deteksi penyakit, serta klasifikasi teks (Hosmer, Lemeshow, & Sturdivant, 2013).

2.4.3 Support Vector Machine

Support Vector Machine (SVM) adalah salah satu algoritma klasifikasi yang populer dalam bidang machine learning, khususnya untuk tugas klasifikasi biner dan multikelas.



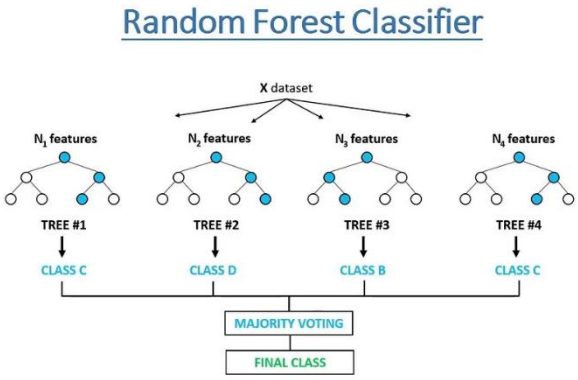
Gambar 2.3 Model Support Vector Machine (SVM)

Sumber: (Neri Van Otten, 2024)

SVM bekerja dengan mencari sebuah hyperplane optimal yang dapat memisahkan data ke dalam dua kelas berbeda dengan margin maksimum. Hyperplane ini adalah sebuah garis (pada data 2 dimensi) atau bidang (pada dimensi lebih tinggi) yang memisahkan dua kelompok data dengan jarak sejauh mungkin dari titik-titik data terdekat di masing-masing kelas, yang disebut sebagai support vectors. Keunggulan utama dari SVM adalah kemampuannya untuk bekerja dengan baik pada data berdimensi tinggi dan ketika jumlah fitur lebih banyak daripada jumlah data. Selain itu, SVM juga mendukung penggunaan kernel trick, yang memungkinkan pemetaan data non-linear ke dalam ruang berdimensi lebih tinggi agar dapat dipisahkan secara linear. Dengan kata lain, SVM tidak hanya terbatas pada data yang dapat dipisahkan secara linear, tetapi juga dapat menangani data yang kompleks secara non-linear menggunakan kernel seperti RBF (Radial Basis Function), polynomial, atau sigmoid (*threshold*) (Cortes & Vapnik, 1995).

2.4.4 Random Forest Classifier

Random Forest Classifier adalah algoritma machine learning berbasis *ensemble learning* yaitu pembelajaran mesin berdasarkan kumpulan dari banyak model perhitungan yang digunakan untuk tugas klasifikasi maupun regresi.



Gambar 2.4 Model Random Forest Classifier

Sumber: (Ankit Chauhan, 2021)

Algoritma ini bekerja dengan membentuk kumpulan (*ensemble*) dari banyak pohon keputusan (*decision trees*) yaitu model perhitungan yang dilatih secara independen menggunakan bagian data acak dari data latih dan fitur yang tersedia yaitu sebuah teknik yang dikenal sebagai bagging (*bootstrap aggregating*). *Bootstrap aggregating*, atau sering disingkat bagging, adalah teknik yang digunakan dalam pembelajaran mesin untuk meningkatkan akurasi dan kestabilan model dengan cara melatih banyak model pada data acak yang berbeda, lalu menggabungkan hasilnya (Breiman, 2001).

Dalam penerapannya pada klasifikasi multilabel, probabilitas prediksi untuk setiap label dapat dihitung berdasarkan proporsi pohon yang memilih label tersebut. Nilai probabilitas ini kemudian diproses menggunakan threshold tertentu (misalnya 0.5) untuk menentukan apakah suatu label diklasifikasikan sebagai relevan (positif) atau tidak. Meskipun Random Forest tidak secara eksplisit menggunakan fungsi sigmoid, pendekatan thresholding tetap digunakan untuk mengonversi skor probabilitas menjadi output biner yang sesuai dengan skenario multilabel.

2.5 Model Deep Learning

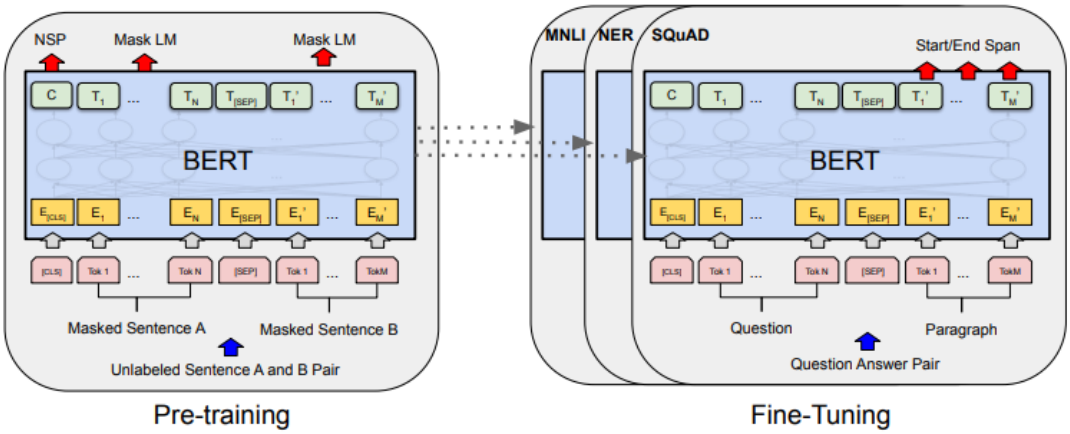
Deep Learning merupakan cabang dari machine learning yang berfokus pada penggunaan jaringan saraf tiruan atau *Artificial Neural Networks* (ANN) dengan banyak lapisan tersembunyi (*hidden layers*) yang berada di antara lapisan input dan output untuk memproses dan mengekstraksi fitur dari data secara otomatis. Terinspirasi oleh struktur dan fungsi otak manusia, setiap lapisan dalam model deep learning secara hierarkis belajar untuk mengenali fitur-fitur dari data pada tingkat abstraksi yang berbeda. Tidak seperti pendekatan machine

learning tradisional yang sering mengharuskan manusia mengekstrak fitur-fitur penting secara manual, deep learning mampu mempelajari sendiri fitur-fitur penting dari data mentah melalui proses pelatihan. Misalnya, dalam data teks, model deep learning dapat mengenali pola seperti struktur kalimat, makna kata dalam konteks tertentu, hingga hubungan antar kalimat, tanpa diberi aturan eksplisit. Kemampuan ini membuat deep learning sangat efektif dalam menangani data yang kompleks dan tidak terstruktur seperti teks, gambar, dan suara (Sarker, 2021).

Dalam bidang pemrosesan bahasa alami atau *Natural Languange Processing* (NLP), sebuah terobosan revolusioner dalam arsitektur deep learning, khususnya untuk pengolahan data sekuensial, adalah model arsitektur Transformer yang diperkenalkan oleh (Vaswani et al., 2023). Transformer mengandalkan sebuah mekanisme inti yang disebut *self-attention*, yaitu sebuah mekanisme yang memungkinkan model memperhatikan setiap kata dalam kalimat dan menentukan seberapa penting kata tersebut terhadap kata lain. Dengan cara ini, model dapat memahami konteks secara menyeluruh sabagai contoh, membedakan makna kata "bisa" dalam kalimat "saya bisa menyanyi" dan "ular itu berbisa". Berbeda dengan model-model sebelumnya seperti CNN maupun LSTM yang memproses urutan kata satu per satu, transformer dapat memproses seluruh kalimat secara paralel, membuatnya lebih cepat dan efisien. Arsitektur ini menjadi fondasi bagi berbagai model NLP modern seperti BERT (Bidirectional Encoder Representations from Transformers) dan seri GPT (Generative Pre-trained Transformer) yang unggul dalam memahami makna dalam suatu teks.

2.5.1 BERT

Bidirectional Encoder Representations from Transformers (BERT) adalah sebuah model representasi bahasa yang dikembangkan oleh Google dan diperkenalkan pada tahun 2019. BERT dibangun berdasarkan arsitektur Transformer, khususnya dengan memanfaatkan bagian *encoder* yang memungkinkan pemrosesan teks secara paralel dan efisien. Mekanisme inti yang menjadi kekuatan utama arsitektur ini adalah *self-attention*. Mekanisme *self-attention* memungkinkan model untuk menimbang tingkat kepentingan setiap kata dalam sebuah kalimat ketika memproses satu kata tertentu. Dengan kata lain, untuk memahami sebuah kata, model secara bersamaan memperhatikan semua kata lain dalam kalimat tersebut dan menentukan kata mana yang paling relevan untuk memberikan konteks.



Gambar 2.5 Model BERT

Sumber: (Devlin, Chang, Lee, & Toutanova, 2019)

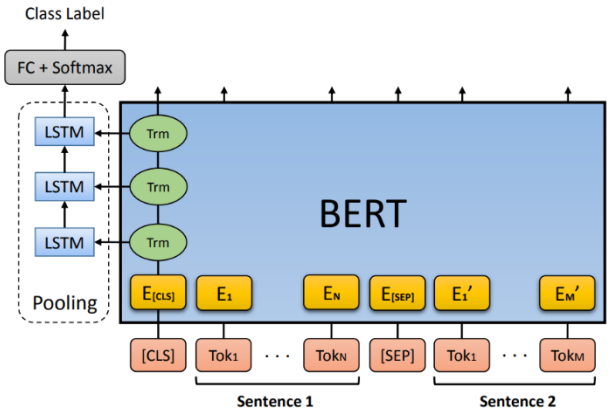
Pada Gambar 2.5 dapat dilihat ciri khas utama sekaligus keunggulan BERT adalah kemampuannya dalam memahami konteks kata secara dua arah (*bidirectional*), yaitu dengan memperhatikan kata-kata sebelum dan sesudah suatu kata dalam sebuah kalimat. Hal ini memungkinkan model untuk menangkap makna kata secara menyeluruh dan utuh pada keseluruhan teks. Untuk merepresentasikan masukan berupa satu atau dua kalimat, BERT selalu menempatkan token spesial [CLS] di awal setiap urutan masukan. Vektor tersembunyi akhir dari token [CLS] ini berfungsi sebagai representasi agregat seluruh urutan untuk tugas-tugas klasifikasi. Sementara itu, token [SEP] digunakan untuk memisahkan kalimat dalam pasangan kalimat, seperti antara pertanyaan dan paragraf, atau antara dua kalimat dalam tugas prediksi hubungan antar kalimat. Dalam fase pre-training, [CLS] digunakan untuk tugas Next Sentence Prediction (NSP), di mana model memprediksi apakah kalimat kedua adalah kelanjutan dari kalimat pertama, dengan format [CLS] kalimat A [SEP] kalimat B [SEP]. Saat fine-tuning untuk tugas-tugas downstream seperti klasifikasi tingkat kalimat, representasi dari token [CLS] (C) dimasukkan ke lapisan keluaran tambahan. Untuk tugas yang melibatkan pasangan teks, seperti tanya jawab, format masukannya juga menggunakan [CLS] di awal dan [SEP] sebagai pemisah antara pertanyaan dan bagian teks yang relevan, mirip dengan format pre-training. (Devlin et al., 2019).

Selain itu, untuk mencapai pemahaman kontekstual pada model BERT, dalam proses pelatihannya, BERT menggunakan dua teknik utama, yaitu *Masked Language Modeling* (MLM) dan *Next Sentence Prediction* (NSP). Pada teknik MLM, model secara acak menyembunyikan (*masking*) beberapa kata dalam kalimat. Pemilihan token yang akan di-

masking ini dilakukan sekali di awal dan tetap sama sepanjang pelatihan (*static masking*). Model kemudian dilatih untuk memprediksi kata yang hilang tersebut berdasarkan konteks di sekitarnya. Sebagai contoh, dalam kalimat "*I feel* [MASK] *today*," model akan mencoba menebak kata yang tersembunyi berdasarkan kata sebelum ("*feel*") dan sesudahnya ("*today*"). Sementara itu, pada teknik NSP, model diberikan dua kalimat dan dilatih untuk memutuskan apakah kalimat kedua merupakan lanjutan yang logis dari kalimat pertama. Teknik ini membuat BERT mampu memahami hubungan antar kalimat, yang sangat penting ketika sebuah informasi atau emosi tersebar dalam teks yang panjang (Chakkarwar, Tamane, & Thombre, 2023).

2.5.2 BERT-LSTM

BERT-LSTM adalah sebuah arsitektur gabungan antara BERT dan Long Short-Term Memory (LSTM) yang digunakan dalam pemrosesan teks. Penerapan LSTM pada keluaran dari BERT berfungsi untuk menangkap hubungan atau pola yang terjadi dalam urutan kata dan ketergantungan antar kata yang muncul dalam sebuah kalimat atau teks. Meskipun BERT sangat efektif dalam memahami konteks dari setiap kata dalam kalimat dengan memperhatikan kata-kata sebelumnya dan sesudahnya (dua arah), namun BERT sendiri tidak selalu optimal dalam memahami urutan kata yang panjang atau hubungan jangka panjang antara kata-kata yang jauh letaknya satu sama lain. Oleh karena itu, LSTM ditambahkan untuk mengatasi keterbatasan ini. LSTM, yang merupakan jenis jaringan saraf berbasis urutan, dirancang khusus untuk mempelajari dan mengingat informasi dari urutan data yang panjang, sehingga dapat menangkap pola temporal atau pola yang berhubungan dengan waktu dalam data tersebut (Tan, Lee, Anbananthen, & Lim, 2022).



Gambar 2.6 Model BERT-LSTM
Sumber: (Song et al., n.d.)

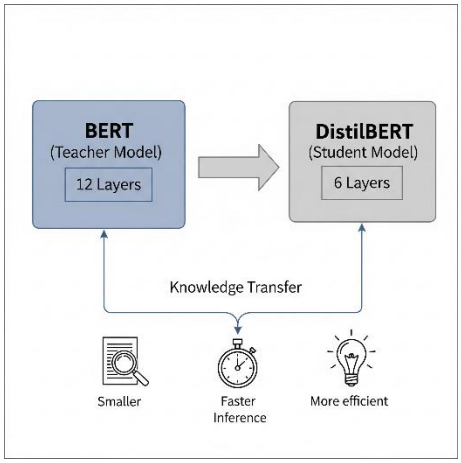
Pada Gambar 2.6, Penambahan LSTM pada BERT-LSTM bertujuan untuk mengambil informasi-informasi penting atau disebut representasi semantik yang tersembunyi di lapisan-lapisan tengah model BERT, terutama dari token khusus [CLS] (token yang mewakili seluruh kalimat). Biasanya, model BERT lain hanya memakai lapisan paling akhir untuk fine-tuning, tapi cara ini bisa membuat beberapa informasi penting jadi terlewat. Oleh karena itu, BERT-LSTM menganggap hidden states dari token [CLS] yang berasal dari setiap lapisan perantara sebagai sebuah urutan khusus yang bersifat abstrak hingga spesifik. Mengingat jaringan LSTM secara sifat memadai untuk memproses informasi yang sekuensial atau berurutan, sebuah jaringan LSTM digunakan untuk menghubungkan semua representasi perantara dari token [CLS] tersebut. Kemudian, keluaran dari sel LSTM terakhir digunakan sebagai representasi akhir untuk prediksi label (Song et al., n.d.). Dengan kata lain, LSTM memungkinkan model untuk mengingat informasi penting yang tersebar jauh dalam urutan kata atau kalimat yang panjang, yang mungkin tidak sepenuhnya diperhatikan oleh mekanisme *self-attention* BERT.

2.5.3 RoBERTa

A Robustly Optimized BERT Pretraining Approach (RoBERTa) adalah model bahasa yang dikembangkan oleh Facebook AI Research sebagai penyempurnaan dari BERT. RoBERTa menggunakan arsitektur dasar yang sama dengan BERT, namun dilatih dengan strategi yang lebih optimal. Perbedaan utamanya terletak pada penghapusan Next Sentence Prediction (NSP) dan penerapan dynamic masking, yaitu teknik di mana token yang di-masking dilakukan secara acak dan berubah-ubah di setiap epoch pelatihan. Hal ini memungkinkan model belajar dari konteks yang lebih bervariasi. Selain itu, RoBERTa dilatih menggunakan lebih dari 160 GB data dari BookCorpus, CC-News, OpenWebText, dan Stories, jauh lebih besar dibandingkan BERT yang hanya menggunakan sekitar 16 GB data. RoBERTa juga dilatih dengan batch size sebesar 8.000 selama 500.000 langkah pelatihan Dengan optimasi tersebut, RoBERTa sangat efektif untuk berbagai tugas NLP seperti klasifikasi teks, analisis sentimen, dan ekstraksi informasi (Y. Liu et al., 2019).

2.5.4 DistilBERT

Distilled version of BERT (DistilBERT) adalah versi ringan dan efisien dari BERT yang dikembangkan oleh tim Hugging Face pada tahun 2020. Model ini dibuat melalui proses knowledge distillation, yaitu teknik kompresi model di mana model yang lebih kecil dilatih untuk meniru perilaku model yang lebih besar dan kompleks, dalam hal ini BERT. Tujuan utama dari DistilBERT adalah untuk mempertahankan sebagian besar performa BERT sambil mengurangi ukuran model dan waktu inferensi secara signifikan.



Gambar 2.7 Model DistilBERT

DistilBERT memiliki sekitar 66 juta parameter, dibandingkan dengan BERT-base yang memiliki 110 juta parameter, dan dapat berjalan lebih cepat dalam proses inferensi maupun pelatihan. Meskipun lebih ringan, DistilBERT tetap mampu mencapai performa yang mendekati BERT pada berbagai tugas pemrosesan bahasa alami. Model ini tidak menggunakan komponen *Next Sentence Prediction* dalam proses pretraining-nya, namun tetap menggunakan *Masked Language Modeling* sebagai teknik utama untuk memahami konteks kata dalam teks. Dengan efisiensi dan kecepatan yang lebih tinggi, DistilBERT menjadi alternatif untuk penerapan NLP di lingkungan dengan keterbatasan sumber daya komputasi (Sanh, Debut, Chaumond, & Wolf, 2020).

2.6 Hyperparameter

Dalam proses pelatihan model Deep Learning, sejumlah hyperparameter digunakan untuk mengatur perilaku dan performa model selama training. Hyperparameter adalah parameter yang tidak dipelajari langsung oleh model selama proses pelatihan, melainkan ditentukan

sebelum pelatihan dimulai. Berbeda dengan parameter model seperti bobot atau bias, yang diperbarui selama proses pelatihan, hyperparameter menentukan aspek penting dalam struktur dan fungsi pelatihan itu sendiri (Bergstra & Bengio, 2012).

Hyperparameter pertama adalah MAX_LEN, yang menentukan panjang maksimum token input untuk setiap teks. Nilai ini penting untuk menjaga konsistensi input dan efisiensi komputasi, terutama pada model berbasis transformer. Selanjutnya, BATCH_SIZE digunakan untuk menentukan jumlah data yang diproses dalam satu iterasi pelatihan. Semakin besar nilai batch, semakin cepat proses pelatihan, namun membutuhkan memori yang lebih besar. OPTIMIZER digunakan pada pelatihan model transformer agar hasil prediksi model semakin akurat seiring berjalannya pelatihan. Untuk mengatur laju pembelajaran, digunakan nilai LEARNING_RATE yang disesuaikan agar model dapat mencapai konvergensi secara stabil tanpa melompat terlalu jauh dari optimum lokal. EPOCHS digunakan untuk menentukan jumlah pengulangan penuh terhadap seluruh dataset selama proses pelatihan. Dalam penelitian terkait multilabel, Salah satu hyperparameter penting lainnya adalah THRESHOLD yang digunakan untuk menentukan batas nilai probabilitas yang memisahkan prediksi positif dan negatif (Kim, 2016).

2.7 Metrik Evaluasi

Metrik evaluasi adalah alat ukur kuantitatif yang digunakan untuk menilai seberapa baik sebuah model machine learning atau deep learning bekerja. Metrik ini sangat penting karena membantu kita mengetahui apakah model sudah cukup baik atau masih perlu diperbaiki. Metrik ini juga berguna untuk membandingkan berbagai model yang dibuat, sehingga kita bisa memilih model yang paling cocok dan paling efektif untuk tugas tertentu (Hripcsak & Albers, 2022).

Berbeda dengan klasifikasi single label, pada klasifikasi multi label diperlukan metrik khusus yang mampu mengevaluasi prediksi secara komprehensif, seperti Jaccard Accuracy, Subset Accuracy, Label-wise Accuracy, Hamming Loss, dan juga ketiga metrik umum yaitu Precision, Recall, dan F1-Score yang dihitung dengan metode rata-rata (macro). Metrik-metrik ini digunakan untuk mengukur sejauh mana label yang diprediksi model sesuai dengan label sebenarnya, baik secara keseluruhan maupun per label. Penggunaan kombinasi metrik ini memungkinkan analisis yang lebih mendalam terhadap kemampuan model, tidak hanya dalam memprediksi label yang benar tetapi juga dalam menangani kompleksitas prediksi ganda pada setiap data teks (Tsoumakas & Katakis, 2009).

2.7.1 Akurasi

Akurasi secara mendasar adalah metrik evaluasi yang paling dasar dalam machine learning, yang mengukur proporsi prediksi yang benar dari total keseluruhan prediksi yang dibuat oleh model. Dalam bentuknya yang paling sederhana, akurasi dihitung dengan membagi jumlah prediksi yang benar dengan jumlah total data (Tharwat, 2021). Namun, dalam kasus klasifikasi emosi pada teks multilabel, di mana setiap data bisa memiliki lebih dari satu label yang relevan, pengukuran akurasi menjadi lebih rumit. Metrik akurasi yang biasa digunakan tidak cukup lagi, sehingga dibutuhkan pendekatan lain yang dapat mengevaluasi kinerja model dengan lebih detail. Dalam konteks penelitian ini, untuk mengukur performa model dalam tugas klasifikasi multilabel, digunakan beberapa jenis perhitungan akurasi yang lebih spesifik, yaitu Jaccard Accuracy, Subset Accuracy, dan Label-wise Accuracy (M.-L. Zhang & Zhou, 2014).

a. Jaccard Accuracy

Jaccard Accuracy menghitung rasio antara jumlah label yang diprediksi dengan benar (irisan) dengan jumlah total label unik antara himpunan prediksi dan himpunan sebenarnya (gabungan). Metrik ini mengevaluasi kemiripan antara set label prediksi dan set label asli untuk setiap instance, lalu merata-ratakannya untuk seluruh data. Metrik ini menawarkan pendekatan yang lebih fleksibel dibandingkan Subset Accuracy. Rumus Jaccard Accuracy dapat dilihat pada persamaan (2.1)

$$\text{Jaccard Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|Y_i \cup \hat{Y}_i\right|}{\left|Y_i \cap \hat{Y}_i\right|} \quad (2.1)$$

N= Jumlah total sampel data.

Y_i= himpunan label sebenarnya untuk sampel ke-i.

$\hat{Y}_i$= himpunan label yang diprediksi untuk sampel ke-i.

$\left|Y_i \cap \hat{Y}_i\right|$ = jumlah label yang sama antara himpunan sebenarnya dan himpunan prediksi.

$\left|Y_i \cap \hat{Y}_i\right|$ = jumlah total label unik dalam gabungan himpunan sebenarnya dan himpunan prediksi.

b. Subset Accuracy

Metrik ini adalah yang paling ketat dalam evaluasi multilabel. Ia mengukur persentase prediksi di mana himpunan label yang diprediksi oleh model sama persis dengan himpunan label yang sebenarnya. Metrik ini hanya memberikan nilai "benar" (1) jika semua label

untuk sebuah instance data berhasil diprediksi dengan sempurna, dan "salah" (0) jika ada satu saja kesalahan, baik itu kelebihan atau kekurangan label. Rumus Subset Accuracy dapat dilihat pada persamaan (2.2)

$$\text{Subset Accuracy} = \frac{1}{N}\sum_{i=1}^{N} I\left(Y_i = \hat{Y}_i\right) \quad (2.2)$$

N= Jumlah total sampel data.

Y_i= himpunan label sebenarnya untuk sampel ke-i.

$\hat{Y}_i$= himpunan label yang diprediksi untuk sampel ke-i.

$I(x)$= fungsi indikator yang bernilai 1 jika kondisi x benar, dan 0 jika salah.

c. Label-wise Accuracy

Pendekatan ini memecah masalah multilabel menjadi beberapa masalah klasifikasi biner, satu untuk setiap label yang ada. Label-wise Accuracy ini mengukur akurasi untuk setiap label individual secara terpisah dan kemudian menghitung rata-rata akurasinya. Rumus Subset Accuracy dapat dilihat pada persamaan (2.3)

$$\text{Label} - \text{wise Accuracy} = \frac{1}{L}\sum_{j=1}^{L} \frac{TP_j + TN_j}{TP_j + TN_j + FP_j + FN_j} \quad (2.3)$$

L= Jumlah label.

j= setiap label untuk TP (True Positive), TN (True Negative), FP (False Positive), dan FN (False Negative).

2.7.2 Precision

Precision adalah salah satu metrik evaluasi yang digunakan untuk mengukur akurasi dari prediksi positif yang dibuat oleh model. Precision mengukur seberapa banyak prediksi yang benar-benar positif dibandingkan dengan semua prediksi yang dihasilkan oleh model sebagai positif. Di mana True Positives (TP) adalah jumlah prediksi yang benar-benar positif, akan dibagi dengan jumlah dari TP dan False Positives (FP) adalah jumlah prediksi yang salah dikategorikan sebagai positif (Sokolova & Lapalme, 2009).

Namun, dalam penelitian ini yang berfokus pada klasifikasi multilabel emosi, digunakan Macro Precision untuk mengevaluasi kinerja model. Macro Precision adalah metrik yang

menghitung precision untuk setiap label secara terpisah dan kemudian mengambil rata-rata dari semua precision label tersebut. Pendekatan ini tidak memperhitungkan ketidakseimbangan antar label, sehingga memberikan penilaian yang lebih seimbang terhadap setiap label dalam klasifikasi multilabel. Rumus Macro Precision dapat dilihat pada persamaan (2.4)

$$\text{Macro Precision} = \frac{1}{L}\sum_{l=1}^{L}\frac{TP_l}{TP_l + FP_l} \quad (2.4)$$

L= Jumlah label.

TP_l, FP_l= jumlah true positives dan false positives untuk label ke-l.

2.7.3 Recall

Recall adalah metrik evaluasi yang mengukur seberapa baik model dalam menemukan semua contoh positif yang ada di dataset. Recall menghitung proporsi dari jumlah True Positives (TP) yang berhasil diprediksi oleh model dibandingkan dengan jumlah keseluruhan data yang seharusnya diklasifikasikan sebagai positif, yang mencakup baik True Positives (TP) maupun False Negatives (FN). Di mana True Positives (TP) adalah jumlah prediksi yang benar-benar positif, akan dibagi dengan jumlah dari TP dan False Negatives (FN) yang adalah jumlah data yang seharusnya diprediksi positif tetapi dikategorikan sebagai negatif (Sokolova & Lapalme, 2009).

Sama seperti precision, penelitian ini menggunakan pendekatan Macro Recall untuk evaluasi. Pendekatan ini bekerja dengan cara menghitung nilai recall untuk setiap label emosi secara terpisah, lalu merata-ratakan hasil tersebut tanpa memandang jumlah sampel pada setiap kelas. Dengan memberikan bobot yang sama pada setiap kelas, Macro Recall memastikan bahwa kemampuan model untuk mendeteksi kelas minoritas (emosi yang jarang muncul) dievaluasi secara setara dengan kelas mayoritas. Rumus Macro Recall dapat dilihat pada persamaan (2.5)

$$\text{Macro Recall} = \frac{1}{L}\sum_{l=1}^{L}\frac{TP_l}{TP_l + FN_l} \quad (2.5)$$

L= Jumlah label.

TP_l, FP_l= jumlah true positives dan false negatives untuk label ke-l.

2.7.4 F1-Score

F1 Score adalah metrik evaluasi yang digunakan untuk mengukur keseimbangan antara precision dan recall dalam model klasifikasi. F1 Score dihitung sebagai harmonik rata-rata antara precision dan recall, yang memberikan gambaran lebih seimbang mengenai kinerja model, terutama ketika terdapat ketidakseimbangan antara prediksi positif dan negatif (Goutte & Gaussier, 2005). Macro F1 Score menghitung F1 Score untuk setiap label secara terpisah, kemudian mengambil rata-rata dari F1 Score yang dihasilkan untuk setiap label. Pendekatan ini memberikan bobot yang sama pada setiap label, tanpa mempengaruhi distribusi jumlah label dalam dataset. Metrik ini sangat berguna karena memberikan gambaran yang seimbang mengenai performa model, terutama ketika berhadapan dengan dataset yang tidak seimbang (imbalanced dataset), di mana akurasi saja bisa tidak sesuai (Opitz & Burst, 2021). Rumus Macro Recall dapat dilihat pada persamaan (2.6)

$$\text{Macro Recall} = \frac{1}{L}\sum_{l=1}^{L}\frac{2 \times Precision_l \times Recall_l}{Precision_l + Recall_l} \quad (2.5)$$

L= Jumlah label.

$Precision_l$ dan $Recall_l$ = nilai precision dan recall untuk label ke-l.

2.7.5 Confusion Matrix

Confusion Matrix adalah tabel visualisasi yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan membandingkan prediksi model terhadap nilai sebenarnya atau aktual. Matriks ini terdiri dari empat elemen utama yang menggambarkan hasil prediksi model: True Positives (TP), False Positives (FP), True Negatives (TN), dan False Negatives (FN). True Positives (TP) adalah jumlah data yang benar-benar positif dan diprediksi sebagai positif, sementara False Positives (FP) adalah data yang seharusnya negatif tetapi diprediksi sebagai positif. True Negatives (TN) mengacu pada data yang benar-benar negatif dan diprediksi sebagai negatif, dan False Negatives (FN) adalah data yang seharusnya positif tetapi diprediksi sebagai negatif. Confusion Matrix memberikan gambaran yang lebih jelas mengenai kesalahan-kesalahan yang dibuat oleh model dan memungkinkan perhitungan metrik evaluasi lainnya seperti precision, recall, F1 score, dan accuracy (Tharwat, 2021).

2.7.6 Waktu Inferensi

Waktu inferensi atau *inference time* adalah metrik evaluasi yang mengukur kecepatan sebuah model machine learning atau deep learning yang sudah dilatih, untuk melakukan prediksi pada data baru yang belum pernah dilihat sebelumnya. Waktu inferensi mengukur efisiensi model dalam membuat keputusan berdasarkan data baru yang diberikan, dan sangat penting dalam aplikasi yang memerlukan *respons real-time* atau dalam sistem dengan sumber daya terbatas. Dalam banyak kasus, model yang lebih kompleks atau lebih besar cenderung memiliki waktu inferensi yang lebih lama karena melibatkan komputasi yang lebih intensif. Oleh karena itu, waktu inferensi menjadi salah satu pertimbangan penting dalam pemilihan model, terutama untuk aplikasi seperti deteksi emosi pada teks, yang memerlukan kecepatan dalam memberikan feedback kepada pengguna (He, Zhang, Ren, & Sun, 2016).

2.8 Kajian Literatur

Kajian literatur ini bertujuan untuk mengidentifikasi dan menganalisis berbagai penelitian terdahulu yang relevan dengan topik deteksi pada teks multilabel dan diutamakan mengenai emosi. Melalui kajian ini, penulis berupaya memahami pendekatan, metode, serta hasil yang telah dicapai oleh studi-studi sebelumnya sebagai gambaran dan acauan untuk mengidentifikasi celah atau keterbatasan yang masih ada dalam studi-studi terdahulu, sehingga penelitian ini diharapakan dapat memberikan kontribusi baru yang belum ada pada penelitian sebelumnya.

Tabel 2.1 Penelitian terdahulu

Literatur	Masalah	Dataset	Model	Kinerja Model
SSN_MLRG1@LT-EDI-ACL2022: Multi-Class Classification using BERT models for Detecting Depression Signs from Social Media Text (Anantharaman, Rajalakshmi, Angel Deborah, Saritha, & Sakaya Milton, 2022)	Deteksi tanda-tanda depresi dari postingan media sosial, mengklasifikasikan menjadi "tidak depresi", "depresi sedang", dan "depresi parah	DepSign-LT-EDI@ACL-2022 (8891 entri pelatihan, 4496 pengembangan, 3245 pengujian)	BERT	pada data tes Akurasi 58.5%, Macro F1-Score 0.412, Macro Recall 0.403, Macro Precision 0.436
Deep learning for extreme multi-label text classification (J.	Mengekstrak informasi dari data yang sangat besar dengan kontekstual	Dataset Zhihu Corpora (3 juta pertanyaan, 1999 kelas),	CNN, RNN, LSTM, FastText, TextCNN,	Koefisien harmonik dari presisi dan recall (score). Model terbaik

Liu, Chang, Wu, & Yang, 2017)	dan komponen kunci secara otomatis tanpa fitur rancangan manusia	dataset teks publik terbesar dalam bahasa Mandarin	dan Contextual-RNN (C-RNN)	adalah C-RNN mencapai score 0.4136
Multi-label emotion classification in texts using transfer learning (Ameer et al., 2023)	Klasifikasi emosi multi-label dari teks pendek di media sosial, kurangnya penggunaan mekanisme *attention* berganda atau *Transformer Networks*	Dataset SemEval-2018 Task-1C (Inggris, 11 emosi + netral) dan Ren-CECps (Mandarin, 8 emosi dasar)	LSTM, Bi-LSTM, XLNet-MA, DistilBERT-MA, RoBERTa-MA	RoBERTa-MA mencapai akurasi 62.4% untuk Inggris. XLNet-MA mencapai akurasi 45.6% untuk Mandarin.
DepressionEmo: A novel dataset for multilabel classification of depression emotions (Rahman et al., 2024)	Deteksi 8 emosi terkait depresi dari postingan pengguna Reddit yang panjang. Kebutuhan dataset spesifik untuk emosi depresi dalam teks panjang.	Dataset Reddit DepressionEmo	SVM, XGBoost, LightGBM, ERT, BART, GAN-BERT, T5	BERT-uncased dan BART mencapai F1 Macro 0.76 dan F1 Micro 0.80
Research on Multi-label Text Classification Method Based on tALBERT-CNN (W. Liu et al., 2021)	kesulitan menangkap fitur semantik dari berbagai tingkat seperti dokumen panjang dengan semantik kompleks	arXiv Academic Paper Dataset (AAPD), Internet Movie Database (IMDB), Reuters Corpus Volume I (RCV1)	tALBERT-CNN	Subset akurasi pada ketiga dataset AAPD = 0.3806, IMDB = 0.4825, RCV1 = 0.4868
Large scale multi-label text classification of a hierarchical dataset using Rocchio algorithm (Sowmya B J, Chetan, & Srinivasa, 2016)	Klasifikasi teks multi-label untuk dataset hierarkis skala besar (misal: Wikipedia), dengan tantangan volume data, jumlah kelas yang meningkat, dan sparsity	Dataset dari Kaggle challenge 'Large Scale Hierarchical	Rocchio (centroid-based classifier), KNN	Perhitungan skor berdasarkan Koefisien harmonik dari macro presisi dan macro recall Rocchio = 0.28931, KNN =0.23088
Hybrid Feature Extraction for Multi-Label Emotion Classification in English Text Messages (Ahanin, Ismail, Singh, & AL-Ashmori, 2023)	Klasifikasi emosi multi-label dalam pesan teks (Twitter), di mana satu teks dapat memiliki banyak emosi.	Dua dataset benchmark: SemEval-2018 (Twitter, 11 emosi + netral) dan GoEmotions (Reddit, 27 emosi + netral)	Bi-LSTM dan BERT dengan penerapan fitur hibrida (HEF+DLF)	Akurasi Jaccard tertinggi: 68.40% pada SemEval-2018 (Bi-LSTM+DLF+HEF) dan 53.45% pada GoEmotions (BERT+DLF+HEF)

Tabel 2.1 menyajikan rangkuman dari sejumlah penelitian terdahulu yang berfokus pada klasifikasi teks multilabel, khususnya dalam domain deteksi emosi dan depresi. Beberapa penelitian secara spesifik menangani deteksi depresi dari teks media sosial. Sebagai contoh, (Anantharaman et al., 2022) menggunakan model BERT pada dataset DepSign-LT-EDI@ACL-2022 yang berisi 8891 data latih untuk mengklasifikasikan tingkat depresi, dan berhasil mencapai akurasi 58.5% dengan Macro F1-Score 0.412. Serupa dengan itu, (Rahman et al., 2024) menggunakan dataset Reddit Depression Emo untuk mendeteksi 8 emosi terkait depresi dan melaporkan bahwa model BART serta BERT-uncased memberikan kinerja terbaik dengan F1 Macro sebesar 0.76 dan F1 Micro 0.80.

Pada bidang klasifikasi emosi multi-label yang lebih umum, (Ameer et al., 2023) menerapkan berbagai model seperti RoBERTa-MA dan XLNet-MA pada dataset SemEval-2018 Task-1C (Inggris) dan Ren-CECps (Mandarin). Kinerja terbaik untuk bahasa Inggris dicapai oleh RoBERTa-MA dengan akurasi 62.4%, sementara untuk bahasa Mandarin, XLNet-MA mencapai akurasi 45.6%. (Ahanin et al., 2023) juga bekerja pada klasifikasi emosi multi-label menggunakan dataset SemEval-2018 dan GoEmotions, di mana pendekatan mereka yang menggunakan Bi-LSTM dan BERT dengan fitur hibrida menghasilkan Jaccard Accuracy tertinggi sebesar 68.40% pada dataset SemEval-2018.

Penelitian lain mencakup tantangan klasifikasi teks multilabel yang lebih luas. (J. Liu et al., 2017) mengatasi klasifikasi multi-label ekstrem pada dataset masif Zhihu Corpora berbahasa Mandarin dengan 3 juta pertanyaan, di mana model C-RNN mereka mencapai skor 0.4136. (W. Liu et al., 2021) mengembangkan model tALBERT-CNN untuk menangani teks panjang pada dataset akademis arXiv (AAPD), film IMDB, dan berita RCV1, dengan perolehan subset akurasi tertinggi 0.4868 pada RCV1. Terakhir, (Sowmya B J et al., 2016) menerapkan algoritma klasik seperti Rocchio dan KNN pada dataset hierarkis skala besar dari Kaggle, dengan Rocchio menunjukkan kinerja lebih baik melalui skor 0.28931.

Berbagai penelitian sebelumnya, telah berhasil mengklasifikasikan teks multi-label dan mendeteksi emosi menggunakan berbagai model, termasuk arsitektur transformer, dengan fokus pada deteksi tanda-tanda depresi dari media sosial, klasifikasi emosi multi-label dari teks pendek media sosial, atau penanganan klasifikasi teks multi-label ekstrem pada dataset masif atau teks panjang akademis dan berita. Meskipun demikian, terdapat celah dalam eksplorasi komprehensif mengenai deteksi multi-emosi (klasifikasi multi-label emosi) dalam entri jurnal

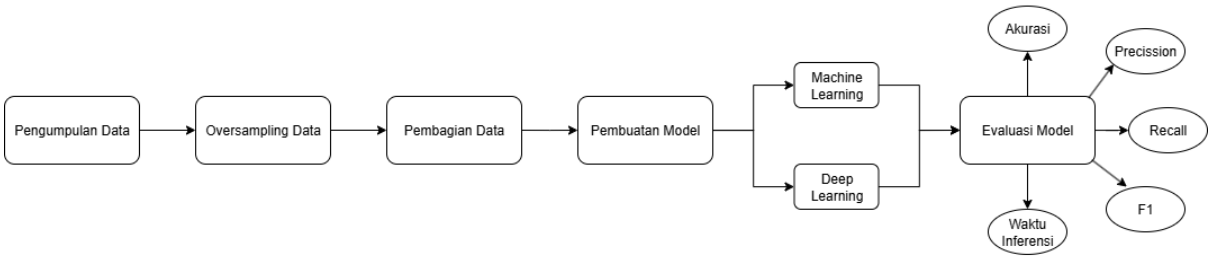
yang bersifat naratif dan memiliki kombinasi emosi yang kompleks, dengan karakteristik teks campuran baik panjang maupun pendek.

Penelitian-penelitian terdahulu juga sudah menerapkan berbagai model, baik machine learning maupun deep learning berbasis transformer untuk menangani karakteristik unik dari berbagai jenis data multilabel secara menyeluruh. Meskipun demikian, terdapat celah yang signifikan karena belum ada penelitian yang secara komprehensif melakukan studi komparatif langsung antara berbagai metode machine learning dan deep learning. Kontribusi penelitian saat ini adalah untuk mengisi celah tersebut dengan melakukan eksplorasi komparatif mendalam terhadap berbagai metode machine learning (Naïve Bayes, Logistic Regression, SVM, RFC) dan Deep Learning berbasis transformer (BERT, BERT-LSTM, RoBERTa, DistilBERT) bertujuan untuk memberikan rekomendasi model terbaik yang paling sesuai untuk deteksi emosi teks multi-label dalam konteks entri jurnal.

BAB III
METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Tahapan pada penelitian ini melibatkan langkah-langkah seperti pengumpulan data, oversampling data, pembagian data, pembuatan model, dan evaluasi model.



Gambar 3.1 Diagram Alir Tahapan Penelitian.

Berdasarkan Gambar 3.1, Penelitian ini dilaksanakan melalui serangkaian tahapan yang diawali dengan proses pengumpulan data yang relevan. Data yang telah terkumpul kemudian menggunakan teknik oversampling guna untuk mengatasi masalah ketidakseimbangan label pada data asli, yang nantinya akan menjadi bahan eksplorasi mengenai kemampuan model dalam menangani ketidakseimbangan label dengan hasil evaluasi antara data hasil oversampling dengan data asli. Setelah itu, data dibagi menjadi data latih dan data uji untuk persiapan pembuatan model. Pada tahap inti, model dikembangkan menggunakan dua pendekatan utama, yaitu Machine Learning dan Deep Learning. Pada pendekatan Machine Learning, model yang digunakan antara lain: Support Vector Machine, Naïve Bayes, Random Forest Classifier, dan Logistic Regression. Sedangkan, pada pendekatan Deep Learning, model yang diguanakan adalah BERT, BERT-LSTM, DistilBERT, dan RoBERTa. Kinerja dari setiap model yang dihasilkan kemudian dievaluasi secara menyeluruh berdasarkan beberapa metrik evaluasi, yang mencakup akurasi, precision, recall, F1-score, dan waktu inferensi untuk menentukan model yang paling efektif dan efisien.

3.2 Pengumpulan Data

Pada penelitian ini, data yang digunakan bersumber dari dataset publik berjudul "Journal Entries with Labelled Emotions" yang dipublikasikan oleh Madhav Malhotra di Kaggle (Malhotra, 2021). Dataset ini berisi entri jurnal harian yang telah diberi label emosi secara

manual dan mencakup berbagai ekspresi emosional dari para penulisnya. Berdasarkan penelitian yang dilakukan oleh (X. A. Li & Parikh, 2019), data tersebut diperoleh dengan cara merekrut 500 subjek anonim melalui platform Amazon Mechanical Turk (AMT). Setiap subjek diinstruksikan untuk menulis entri jurnal yang mendeskripsikan hari mereka sebelumnya sebagai respons terhadap pertanyaan pemicu: "*What were salient aspects of your day yesterday? How did you feel about them?*". Setiap entri dapat berisi hingga tiga sampel teks yang menjelaskan aspek berbeda dari hari-hari mereka. Para subjek tersebut juga diminta untuk memberikan anotasi pada setiap *sub-entry* dengan memilih topik dan emosi yang relevan dari daftar yang telah disediakan, yang kemudian berfungsi sebagai label yang sebenarnya (*ground truth*) untuk evaluasi model.

3.3 Oversampling Data

Berdasarkan analisis dan eksplorasi pada pengumpulan data, teridentifikasi adanya masalah ketidakseimbangan kelas atau label pada dataset asli yang berpotensi menyebabkan performa model menjadi bias terhadap kelas mayoritas. Untuk mengatasi tantangan tersebut, penelitian ini menerapkan teknik oversampling. Proses oversampling data pada penelitian ini menggunakan LLM GPT-4o yang berfungsi sebagai generator data sintetis atau data hasil oversampling, dengan cara melalui proses prompting langsung. Pendekatan ini bertujuan untuk mengaugmentasi atau bahkan menambahkan data baru pada dataset nyata atau asli, terutama dalam skenario di mana data berlabel langka, minoritas, atau sedikit. Berbagai strategi prompting dapat diterapkan untuk menghasilkan data sintetis yang relevan dengan tugas.

Proses prompting dimulai dengan *One-shot* generation melibatkan penyediaan deskripsi tugas dan satu contoh pasangan input-output untuk mengilustrasikan format, yang dimana penelitin melakukan input data asli selajutnya LLM melakukan oversampling berdasarkan hasil pertama kali prompting. Melangkah lebih jauh, *few-shot* generation menggunakan beberapa contoh misalnya, 3-5 prompt untuk memandu LLM menghasilkan output berkualitas lebih tinggi dan spesifik tugas dengan meniru pola-pola yang diberikan, menjadikannya sangat berguna untuk tugas dengan struktur atau ruang label tertentu, meskipun juga berisiko menghasilkan duplikat atau frasa yang sangat mirip. Untuk meningkatkan keragaman data sintetis, terutama ketika mengandalkan data sintetis secara ekstensif, strategi *topic/controlled generation* digunakan atau bisa disebut sebagai pengenalan topik dari dataset tersebut kedapa model LLM (Nadas, Diosan, & Tomescu, 2025). Ini melibatkan penggunaan LLM untuk terlebih dahulu menghasilkan daftar topik atau skenario yang relevan untuk suatu tugas, lalu

menggunakan topik yang dipilih untuk mem-prompt LLM agar menghasilkan contoh spesifik, sehingga dapat memahami keragaman yang terkontrol ke dalam dataset. Data sintetis yang dihasilkan kemudian dapat digabungkan dengan data nyata untuk melatih model, dengan tujuan meningkatkan kinerja, terutama di rezim data rendah.

3.4 Pembagian Data

Pada tahap pembagian dataset, dataset asli dan dataset yang telah melalui proses oversampling akan dibagi dengan rasio 70:30 untuk masing-masing dataset tersebut. Rasio tersebut digunakan berdasarkan referensi dari jurnal (Muraina, 2022), yang dimana rasio tersebut adalah paling optimal untuk dataset dengan jumlah data disekitar 100 hingga 1000000 data. Karakteristik dataset yang digunakan dalam penelitian ini memiliki kesamaan dengan dataset yang digunakan oleh Muraina, yaitu sama-sama berada pada skala kecil hingga menengah dan diperuntukkan bagi tugas klasifikasi dan juga memiliki sifat multilabel yang dimana pada dataset tersebut memiliki 7 label.

Meskipun dataset Journal Entries with Labelled Emotions berbentuk teks naratif multilabel dengan variasi panjang dan distribusi label yang tidak seimbang, tantangan utamanya serupa, yaitu memastikan model mampu mempelajari pola yang representatif meski data terbatas dan label tidak merata. Oleh karena itu, pembagian data dengan rasio 70:30 juga dinilai relevan dalam konteks penelitian ini. Sedangkan pada kelompok model deep learning, 70 persen data digunakan untuk pelatihan, dan 30 persen sisanya dibagi merata menjadi 15 persen data validasi serta 15 persen data uji. Skema ini sesuai dengan kebutuhan deep learning yang memerlukan data validasi khusus untuk proses hyperparameter tuning sekaligus pencegahan overfitting.

3.5 Pembuatan Model

Pada tahap pembuatan model, penelitian ini mengeksplorasi dua pendekatan utama, yaitu Machine Learning dan Deep Learning untuk klasifikasi emosi multilabel. Pada Machine Learning (SVM, Naïve Bayes, Random Forest, Logistic Regression), data latih (70%) diubah menjadi vektor numerik dengan Term Frequency-Inverse Document Frequency (TF-IDF), lalu digunakan untuk melatih model, sedangkan kinerjanya dievaluasi dengan data uji (30%). Sementara itu, pada Deep Learning (BERT, BERT-LSTM, RoBERTa, DistilBERT), data latih (70%) diproses melalui tokenisasi, dan juga pengaturan hyperparameter. Setelah itu, model akan dilatih dalam beberapa epoch dengan pemantauan melalui data validasi (15%)

menggunakan mekanisme early stopping. Setelah pelatihan, model terbaik dievaluasi menggunakan data uji (15%) untuk mengukur performa secara objektif.

a. Model Machine Learning

Pada model Machine Learning, akan digunakan pemanggilan pada keseluruhan model dari pustaka *scikit-learn* dan tambahan masukan fitur *MultiOutputClassifier* yang juga berasal dari pustaka *scikit-learn* untuk setipa model Machine Learning. Sebelum data dimasukkan ke dalam model, dilakukan proses ekstraksi fitur untuk mengubah data teks menjadi format numerik (*vectorizer*) menggunakan Term Frequency-Inverse Document Frequency (TF-IDF). Fungsi utama dari TF-IDF adalah untuk mengukur pentingnya sebuah kata dalam sebuah dokumen terhadap keseluruhan koleksi dokumen (korpus). Metode ini memberikan bobot yang tinggi pada kata-kata yang sering muncul dalam satu teks atau relevan untuk teks tersebut tetapi jarang muncul pada teks-teks lainnya (*unique*), sehingga membantu model mengenali kata kunci yang paling signifikan dan representatif.

Selanjutnya, setiap model Machine Learning yang telah menerima masukan fitur dari TF-IDF akan dibungkus menggunakan MultiOutputClassifier. MultiOutputClassifier merupakan pendekatan yang digunakan oleh model Machine Learning dalam kasus klasifikasi multilabel. MultiOutputClassifier bekerja dengan melatih satu model biner untuk setiap label secara independen. Meskipun pendekatan ini tidak mempertimbangkan korelasi antar label, ia tetap efektif dan fleksibel untuk diterapkan pada kasus multi-label classification seperti deteksi emosi dalam teks, di mana satu teks dapat merepresentasikan lebih dari satu emosi secara bersamaan.

b. Model Deep Learning

Pembentukan model Deep Learning dalam penelitian ini dilakukan menggunakan pustaka *pytorch*, dengan pendekatan *multilabel classification* yang memungkinkan satu teks diklasifikasikan ke dalam lebih dari satu label emosi secara bersamaan. Seluruh model menggunakan arsitektur transformer sebagai basis utama, di mana setiap model secara otomatis memanfaatkan *vectorizer embedding* yang merupakan bawaan dari arsitektur tersebut. Proses ini mengubah teks yang dimasukkan menjadi bentuk vektor numerik yang mengandung informasi penting tentang makna teks tersebut. Vektor-vektor ini kemudian diproses lebih lanjut oleh lapisan-lapisan transformer, yang merupakan bagian dari model, melalui tahapan penyesuaian atau fine-tuning agar model bisa lebih spesifik memahami dan memprediksi berdasarkan data yang diberikan. Fine-tuning dilakukan dengan melatih model yang sudah dilatih sebelumnya dengan data baru untuk meningkatkan kinerjanya pada tugas tertentu.

Secara umum, setiap model memiliki komponen akhir berupa lapisan *dense* (*linear*) dengan jumlah *neuron* yang disesuaikan dengan jumlah label, serta fungsi aktivasi sigmoid (*threshold*) untuk menghasilkan probabilitas pada masing-masing label. Fungsi *loss* yang digunakan adalah *BCEWithLogitsLoss* dari pustaka *pytorch*, yang sesuai untuk klasifikasi multilabel. *BCEWithLogitsLoss* sendiri merupakan fungsi *loss* dari *PyTorch* yang menggabungkan fungsi *Sigmoid* dan *Binary Cross-Entropy Loss* menjadi satu langkah efisien untuk menangani klasifikasi multilabel langsung dari output mentah model.

c. Hyperparamater

Seperti yang dijelaskan pada Sub Bab 2.6, pada pembuatan model deep learning akan digunakan beberapa hyperparamater seperti yang dilihat pada Tabel 3.1 berikut. Nilai-nilai hyperparameter ini diambil berdasarkan rekomendasi yang ada dalam jurnal BERT (Devlin et al., 2019).

Tabel 3.1 Hyperparameter

Nama	**Nilai**
MAX_LEN	256
TRAIN_BATCH_SIZE	32
VALID_BATCH_SIZE	32
TEST_BATCH_SIZE	32
OPTIMIZER	AdamW
LEARNING_RATE	1e-05
THRESHOLD	0.5
LOSS_THRESHOLD	0.02
EPOCHS	150

Beberapa hyperparameter penting dalam pelatihan model ini adalah MAX_LEN (256), yang menentukan panjang maksimum input teks agar tetap konsisten dan efisien. TRAIN_BATCH_SIZE, VALID_BATCH_SIZE, dan TEST_BATCH_SIZE (32) yang mengatur jumlah data yang diproses dalam satu iterasi, memberikan keseimbangan antara efisiensi pelatihan dan penggunaan memori. OPTIMIZER yang digunakan adalah AdamW, yang efektif dalam melatih model transformer dengan memberikan pembaruan parameter yang stabil. LEARNING_RATE (1e-05) dipilih untuk memastikan konvergensi yang stabil tanpa melompat terlalu jauh (Devlin et al., 2019). Selanjutnya, sebagai penyesuain terhadap deteksi emosi teks mltilabel, maka diterapkan THRESHOLD (0.5) digunakan untuk memutuskan

apakah sebuah label prediksi dianggap positif atau negatif. LOSS_THRESHOLD (0.02) berfungsi untuk menghentikan pelatihan lebih awal jika perubahan loss sangat kecil, mencegah overfitting. Terakhir, EPOCHS (150) dipilih untuk memberi cukup waktu pelatihan, dengan fitur early stopping diterapkan agar pelatihan berhenti jika model tidak menunjukkan peningkatan pada loss validasi. Semua nilai hyperparameter yang tercantum dalam tabel ini akan disamakan untuk semua model deep learning yang digunakan dalam penelitian ini. Pendekatan ini bertujuan untuk memastikan konsistensi dan keadilan dalam evaluasi model, sehingga perbandingan kinerja antar model dapat dilakukan dengan lebih objektif tanpa pengaruh perubahan hyperparameter yang tidak konsisten.

3.6 Evaluasi Model

Pada tahap evaluasi model, dilakukan pengujian terhadap seluruh model Machine Learning dan Deep Learning menggunakan data uji yang telah dipisahkan sebelumnya. Evaluasi dilakukan dengan menggunakan sejumlah metrik yaitu akurasi, *precision*, *recall*, *F1-score*, *confusion matrix*, dan *inference time*. Seluruh metrik evaluasi dihitung menggunakan pustaka *scikit-learn* yang diterapkan pada keseluruhan model, baik untuk Machine Learning maupun Deep Learning. Khusus untuk pengukuran *inference time*, digunakan pustaka *time* untuk mencatat lama waktu yang dibutuhkan oleh model dalam memproses dan menghasilkan prediksi terhadap masukan data teks. Selain menggunakan evaluasi pada matrik-matrik perhitungan, dilakukan juga skenario pengujian terhadap beberapa data uji secara langsung untuk mengamati hasil prediksi model terhadap teks dengan berbagai konteks emosi.

BAB IV
HASIL DAN PEMBAHASAN

Bab ini akan memaparkan secara rinci hasil dan pembahasan dari serangkaian eksperimen yang telah dilakukan. Seluruh proses perancangan, pelatihan, dan pengujian model dalam penelitian ini diimplementasikan menggunakan platform Google Colaboratory (Colab) Notebook, dengan memanfaatkan dukungan akselerator perangkat keras berupa T4 GPU untuk mempercepat komputasi, khususnya pada model deep learning. Untuk menunjang implementasi tersebut, penelitian ini menggunakan beberapa pustaka (*library*) utama dari Python. Pustaka *Pandas* dan *NumPy* digunakan sebagai perangkat esensial untuk tahap eksplorasi dan pengolahan data. Dalam pembuatan model, pustaka *PyTorch* menjadi fondasi utama untuk merancang arsitektur dan melatih model berbasis deep learning. Sementara itu, model machine learning dibangun menggunakan pustaka *Scikit-learn*. Lebih lanjut, *Scikit-learn* juga digunakan sebagai perangkat evaluasi untuk mengukur performa keseluruhan model yang diuji, baik dari model machine learning maupun deep learning.

4.1 Hasil Pengumpulan Data

Total keseluruhan data yang diperoleh sumber (Malhotra, 2021) tersebut berjumlah 1473 data yang terdiri atas 18 label emosi, yaitu *afraid* (AF), *angry* (AG), *anxious* (AX), *ashamed* (AS), *awkward* (AW), *bored* (BR), *calm* (CA), *confused* (CO), *disgusted* (DS), *excited* (EX), *frustrated* (FR), *happy* (HP), *jealous* (JL), *nostalgic* (NS), *proud* (PR), *sad* (SD), *satisfied* (ST), dan *surprised* (SR). Sebanyak 18 emosi yang digunakan sebagai label telah ditentukan pada penelitian (X. A. Li & Parikh, 2019), daftar 18 emosi ini dikurasi dari studi oleh (Cambria et al., n.d.) yang menggambarkan model *hourglass of emotion*, dan juga berdasarkan penilaian yang dilakukan oleh peneliti mengenai emosi yang mungkin dialami sehari-hari. Untuk memverifikasi kelengkapan daftar emosi ini, peneliti melakukan studi terhadap manusia dengan menanyakan 100 subjek AMT (Amazon Mechanical Turk) apakah ada emosi yang ingin mereka bicarakan yang hilang dari daftar, dan seluruh 100 subjek menyatakan bahwa daftar tersebut sudah memadai. Emosi-emosi yang telah ditentukan ini, kemudian berfungsi sebagai label kebenaran (*ground truth*) yang digunakan oleh subjek untuk menganotasi setiap sub-entri jurnal dengan memilih label emosi yang relevan dari daftar yang disediakan.

Pada data ini, label emosi ditunjukkan dengan nilai 1 dan 0 yang berarti 1 bernilai TRUE dan 0 bernilai FALSE nilai tersebut bisa terdapat lebih dari 1 pada masing-masing teks yang

berarti teks tersebut memiliki banyak emosi atau multilabel. Selanjutnya, Sebagai bagian dari pembersihan dataset, dilakukan juga perubahan nama pada *header* tabel. Kolom yang pada dataset asli menggunakan nama seperti Answer, Answer.f1.afraid.raw, dan Answer.f1.angry.raw, dan kolom emosi lainnya, diubah menjadi text, afraid, dan angry dan kolom emosi lainnya. Hal ini bertujuan untuk mempermudah dataset asli agar label pada data dapat diproses oleh model sehingga label emosi yang diolah menjadi lebih bersih.

Selain itu, secara keseluruhan data teks yang didapatkan sudah cukup bersih dan tidak ada data yang kosong, sehingga tidak diperlukan pengolahan preprocessing atau cleaning lebih lanjut. Hal ini juga berguna untuk menghindari hilangnya pemahaman konteks pada data ketika melakukan pelatihan model, sehingga model dapat memprediksi emosi dengan lebih baik.. Hal itu bertujuan untuk memperkaya variasi ekspresi emosi dalam teks serta mendukung analisis yang lebih akurat dalam tugas deteksi multi emosi berbasis klasifikasi teks multilabel. Contoh tampilan dataset dapat dilihat pada Tabel 4.1 dan persebaran label emosi pada data dapat dilihat pada Gambar 4.1.

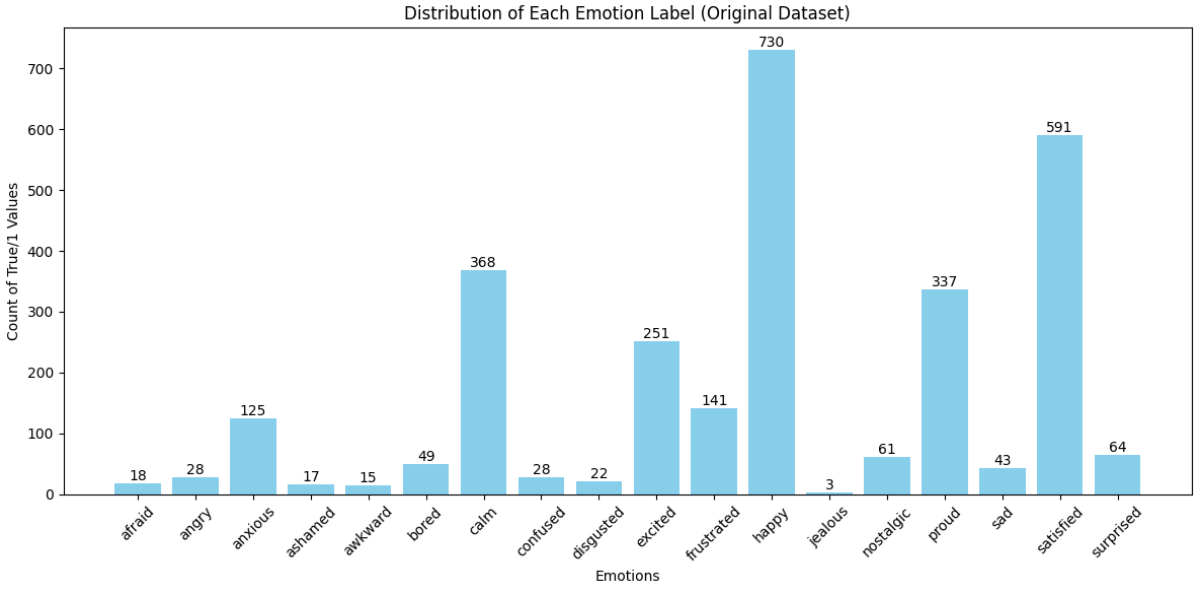
Tabel 4.1 Dataset Journal Diary

Text	AF	AG	AX	AS	AW	BR	CA	CO	DS	EX	FR	HP	JL	NS	PR	SD	ST	SR
I was very happy to be out enjoying some time off from work, and was able to buy some things I had been looking forward to for my collecting hobby. It was the highlight of my week thus far.	0	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	1	0
Yesterday, I visited my parents and had dinner with them. I hadn't seen them in a few weeks, so it was wonderful to see them and catch up on things.	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
I wanted to get a lot of work done in a small amount of time that I had.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Seperti yang terlihat pada Tabel 4.1, contoh data dari Journal Diary Dataset yang digunakan dalam penelitian ini. Setiap baris terdiri dari sebuah teks jurnal harian dan 18 label emosi. Dataset ini bersifat multilabel, artinya satu teks dapat memiliki lebih dari satu label emosi yang bernilai 1 atau TRUE secara bersamaan, seperti yang terlihat pada data baris pertama dan kedua dimana dari teks tersebut terdapat label emosi *calm* (CA)*, excited* (EX)*, happy* (HP)*,* dan *satisfied* (ST). namun, ada beberapa data juga yang hanya memiliki satu label seperti pada baris ketiga yaitu *anxious* (AX). Hal tersebut menunjukkan bahwasannya kelengkapan data ini yang tidak hanya memiliki data multilabel tetapi juga data *single label*. Selain itu, ada beberapa temuan dari data tersebut yaitu memiliki rata-rata kata sekitar 39-50 kata, dengan teks terpanjang dengan 111 kata dan teks terpendek dengan 3 kata.

Sebelum disajikan visualisasi persebaran label emosi, penting untuk mengetahui bahwa apakah dataset asli menunjukkan adanya ketidakseimbangan label (imbalance) yang signifikan. Hal ini berarti beberapa label emosi memiliki frekuensi kemunculan yang sangat dominan, sementara sebagian besar lainnya muncul sangat jarang. Kondisi ketidakseimbangan ini

berpotensi menyebabkan model bias terhadap kelas mayoritas, sehingga menjadi tantangan utama dalam pelatihan dan evaluasi model.



Gambar 4.1 Persebaran Label Emosi Pada Dataset *Journal Entries with Labelled Emotions*.

Berdasarkan Gambar 4.1, persebaran data pada dataset asli menunjukkan adanya ketidakseimbangan label (imbalance) yang signifikan, di mana beberapa label emosi memiliki frekuensi kemunculan yang sangat dominan sementara sebagian besar lainnya muncul sangat jarang. Emosi yang paling dominan adalah *happy* dengan 730 kemunculan dan *satisfied* (591), sedangkan di sisi lain, emosi seperti *jealous* (3), *ashamed* (15), dan *afraid* (18) menjadi kelas minoritas dengan frekuensi sangat rendah. Ketidakseimbangan yang tajam ini menjadi tantangan utama dalam pelatihan model, karena berpotensi menyebabkan bias di mana model akan cenderung lebih baik dalam mengenali emosi mayoritas dibandingkan emosi minoritas.

4.2 Hasil Oversampling Data

Pada tahap ini, telah dilakukan proses generatif oversampling menggunakan model LLM GPT-4o dengan tujuan untuk menyeimbangkan distribusi label emosi dalam data pelatihan. Proses ini menghasilkan 500 data sintetis, sehingga jumlah total data meningkat dari 1.473 menjadi 1.973. Oversampling difokuskan pada label-label minoritas yang memiliki jumlah data jauh lebih sedikit dibanding label lainnya, yakni: afraid, angry, ashamed, awkward, bored, confused, disgusted, jealous, nostalgic, sad, dan surprised. Dengan penambahan data sintetis ini, diharapkan distribusi label menjadi lebih seimbang dan performa model dalam mengenali

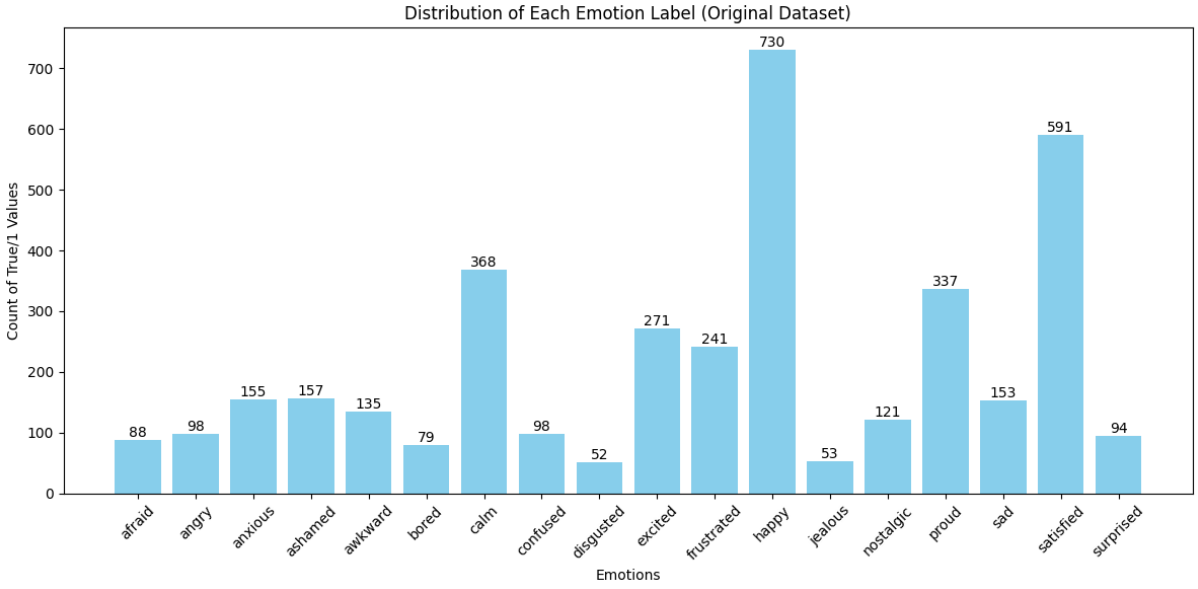
emosi-emosi minoritas meningkat. Contoh perbedaan data antara data asli dan data hasil oversampling dapat dilihat pada Tabel 4.2, sedangkan persebaran visual data setelah proses oversampling ditampilkan pada Gambar 4.2.

Tabel 4.2 Contoh perbedaan data sintetis dan data asli

Jenis Data	Teks	Label Emosi
Data sintetis	I was asked a question during the meeting, and I had no idea how to answer. My mind went blank, and I just stuttered out something that made no sense. I felt both confused and embarrassed.	[ashamed, confused]
Data asli	I was productive at work yesterday, and it made me feel good. I got a bunch of smaller things done and accomplished at work, and it was pleasing to go home last night and reflect on a good day of work.	[proud, satisfied]
Data sintetis	I bumped into my ex unexpectedly today, laughing with someone else. It was weird. I thought I had moved on, but a pang of jealousy hit me. At the same time, I felt kind of ashamed for still caring.	[ashamed, jealous]
Data asli	I didn't lose my appetite. But I can only eat soft foods, especially the first day. So I was hungry and jealous of everyone around me eating normally.	[afraid, jealous]

Berdasarkan hasil observasi terhadap data sintetis yang dihasilkan oleh GPT-4o, terlihat bahwa data tersebut memiliki struktur, gaya bahasa, dan konteks emosional yang serupa dengan data asli. Misalnya, pada data sintetis dengan label ashamed dan confused, narasi situasi yang memicu emosi digambarkan dengan jelas dan relevan. Begitu pula dengan data sintetis berlabel ashamed dan jealous, yang menggambarkan reaksi emosional terhadap pertemuan tak terduga dengan mantan, memiliki kemiripan konteks dengan data asli berlabel jealous, yang mengungkapkan rasa iri terhadap orang lain yang dapat makan secara normal. Namun, data sintetis memiliki kekurangan dimana data tersebut terkadang kaku dan langsung menjelaskan emosi yang dirasakan oleh penulis pada text. Disisi lain, beberapa text pada data asli terkadang tidak langsung menjelaskan emosi yang sedang dirasakan pada teks tetapi bisa juga melalui tentang hal-hal yang dilakukan oleh penulis. Tetapi secara keseluruhan, kesamaan struktur, gaya bahasa, dan konteks emosional ini menunjukkan bahwa data sintetis memiliki kualitas

semantik dan emosional yang dapat diterima, sehingga layak digunakan dalam pelatihan model untuk meningkatkan representasi label minoritas.



Gambar 4.2 Persebaran label emosi pada dataset hasil oversampling.

Gambar 4.2 menampilkan persebaran data label setelah melalui strategi penyeimbangan, yang menggunakan generatif teknik oversampling untuk menambah data pada kelas minoritas. Terlihat jelas bahwa jumlah sampel untuk label minoritas seperti *jealous* (meningkat dari 3 menjadi 53) *ashamed* (dari 17 menjadi 157), dan beberapa label minoritas lainnya yang telah ditingkatkan. Hasil akhirnya adalah distribusi data yang jauh lebih seimbang, yang dirancang agar model dapat belajar mengenali setiap emosi secara lebih adil tanpa menjadi bias, sehingga meningkatkan performa prediksinya terhadap label-label minoritas.

4.3 Hasil Pembagian Data

Implementasi pembagian data ini dilakukan secara acak namun tetap tertata distribusi labelnya dengan menggunakan fungsi *train_test_split* dari pustaka *sklearn.mode._selection* dengan parameter *shuffle=True* untuk menjaga keseimbangan distribusi label pada setiap pembagian data baik data latih, validasi, maupun uji. Kode untuk pembagian data baik pada model Machine Learing dan juga model Deep Learning dapat dilihat pada Gambar 4.3 dan Gambar 4.4.

```
from sklearn.model_selection import train_test_split
```

```
df_train, df_test = train_test_split(df, random_state=77, test_size=0.30,
shuffle=True)
```

Gambar 4.3 Kode program untuk splitting data pada model Machine Learning.

```
from sklearn.model_selection import train_test_split
df_train, df_test = train_test_split(df, random_state=77, test_size=0.30,
shuffle=True)
df_test, df_valid = train_test_split(df_test, random_state=88,
test_size=0.50, shuffle=True)
```

Gambar 4.4 Kode program untuk splitting data pada model Deep Learning.

Pada dataset asli, hasil dari proses pembagian data ini menghasilkan data latih sebanyak 1031 entri data data uji sebanyak 442 entri untuk data pada model Machine Learning. Sedangkan pada model Deep Learning, menghasilkan data latih sebanyak 1031 entri, data validasi sebanyak 221, dan data uji sebanyak 221 entri. Pada dataset hasil oversampling, hasil dari proses pembagian data ini menghasilkan data latih sebanyak 1381 entri data data uji sebanyak 592 entri untuk data pada model Machine Learning. Sedangkan pada model Deep Learning, menghasilkan data latih sebanyak 1031 entri, data validasi sebanyak 296, dan data uji sebanyak 296 entri. Persebaran label pada masing-masing data baik data latih, uji, dan validasi dapat dilihat pada Tabel 4.3, Tabel 4.4, Tabel 4.5, dan Tabel 4.6

Tabel 4.3 Tabel pembagian dataset asli model Machine Learning

Label Emosi	Latih	Uji
afraid	10	8
angry	23	5
anxious	94	31
ashamed	13	4
awkward	9	6
bored	30	19
calm	256	112
confused	18	10
disgusted	17	5
excited	169	82
frustrated	98	43
happy	505	225
jealous	1	2
nostalgic	43	18
proud	213	124
sad	27	16
satisfied	417	174
surprised	42	22

Tabel 4.4 Tabel pembagian dataset asli model Deep Learning

Label Emosi	Latih	Validasi	Uji
afraid	10	3	5
angry	23	3	2
anxious	94	16	15
ashamed	13	1	3
awkward	9	4	2
bored	30	8	11
calm	256	51	61
confused	18	6	4
disgusted	17	2	3
excited	169	43	39
frustrated	98	18	25
happy	505	109	116
jealous	1	0	2
nostalgic	43	9	9
proud	213	71	53
sad	27	6	10
satisfied	417	89	85
surprised	42	15	7

Tabel 4.5 Tabel pembagian dataset hasil oversampling model Machine Learning

Label Emosi	Latih	Uji
afraid	65	23
angry	78	20
anxious	110	45
ashamed	113	44
awkward	98	37
bored	58	21
calm	253	115
confused	65	33
disgusted	38	14
excited	178	93
frustrated	117	64
happy	508	222
jealous	39	14
nostalgic	90	31
proud	225	112
sad	106	47
satisfied	399	192
surprised	68	26

Tabel 4.6 Tabel pembagian dataset hasil oversampling model Deep Learning

Label Emosi	Latih	Validasi	Uji
afraid	65	11	12
angry	78	11	9
anxious	110	21	24
ashamed	113	26	18
awkward	98	22	15
bored	58	10	11
calm	253	56	59
confused	65	16	17
disgusted	38	4	10
excited	178	38	55
frustrated	117	38	26
happy	508	100	122
jealous	39	6	8
nostalgic	90	22	9
proud	225	64	48
sad	106	28	19
satisfied	399	95	97
surprised	68	11	15

Berdasarkan perbandingan tabel-tabel diatas, perbedaan mendasar antara pembagian dataset asli dengan dataset hasil oversampling terletak pada penanganan ketidakseimbangan label. Pada dataset asli yaitu Tabel 4.3 dan Tabel 4.4, terlihat adanya ketimpangan jumlah data yang sangat signifikan antar label emosi. Kelas mayoritas seperti happy (505 data latih), satisfied (417 data latih), dan calm (256 data latih) memiliki jumlah sampel yang jauh lebih banyak dibandingkan kelas minoritas seperti jealous (hanya 1 data latih), awkward (9 data latih), dan disgusted (17 data latih). Perbedaan jumlah yang sangat jauh ini menjadi masalah pada data validasi di model Deep Learning, di mana emosi *jealous* bahkan tidak memiliki data sama sekali (0 data). Akibatnya, kemampuan model untuk mengenali emosi tersebut tidak bisa optimal selama proses pelatihan.

Sementara itu, pada dataset hasil oversampling yaitu Tabel 4.5 dan Tabel 4.6, jumlah data untuk kelas-kelas minoritas telah ditingkatkan secara drastis, terutama pada set data latih. Sebagai contoh, data untuk emosi *jealous* meningkat dari 1 menjadi 39, ashamed dari 13 menjadi 113, dan sad dari 27 menjadi 106. Sebaliknya, jumlah data untuk label mayoritas seperti happy dan satisfied cenderung tidak berubah secara signifikan. Perbaikan penting juga terlihat pada data validasi dan data uji pada model Deep Learning, di mana semua emosi kini memiliki data dan ada beberapa peningkatan pada label yang sebelumnya jarang muncul bahkan tidak ada data seperti *jealous* yang sebelumnya tidak memiliki data pada validasi

sekarang mempunyai 6 data setelah melakukan oversampling. Proses ini bertujuan untuk menciptakan distribusi data yang lebih seimbang di setiap label emosi, sehingga model Machine Learning dan Deep Learning dapat mempelajari pola dari setiap emosi secara lebih merata terhadap label yang jarang muncul.

4.4 Hasil Pembuatan Model

Hasil dari pembuatan model dibagi menjadi dua bagian utama yaitu machine learning dan deep learning. Pada bagian machine learning, akan dipaparkan tabel perbandingan akurasi antara data asli dan data hasil oversampling dari keempat model. Sementara itu, untuk bagian deep learning, pembahasan akan diawali dengan visualisasi grafik performa akurasi dan loss selama pelatihan, yang kemudian dilanjutkan dengan tabel perbandingan akurasi model pada data asli dan data hasil oversampling untuk membandingkan performa keempat model deep learning pada kedua jenis data tersebut.

4.4.1 Hasil Model Machine Learning

Hasil pembuatan model machine learning telah diterapkan pada kedua jenis dataset, yaitu dataset asli dan dataset hasil oversampling. Pada hasil model machine learning ini, akan ditampilkan performa dari empat model machine learning yang digunakan, yaitu Naive Bayes, Logistic Regression, Support Vector Machine (SVM), dan Random Forest Classifier (RFC). Masing-masing model akan dibagi menjadi dua pembagain data yaitu data latih sebesar 70% dan data uji sebesar 30% pada masing-masing dataset. Kemudian, akan dievaluasi berdasarkan empat metrik utama: Jaccard Accuracy (JA), Subset Accuracy (SA), Label-wise Accuracy (LA), dan Loss.

Tabel 4.7 Tabel hasil pembuatan model machine learning menggunakan data asli

Model	**Data Latih (70%)**				**Data Uji (30%)**			
	JA	***SA***	***LA***	***Loss***	***JA***	***SA***	***LA***	***Loss***
Naïve Bayes	0.3342	0.1823	0.9224	3.880	0.1934	0.0723	0.8978	5.608
Logistic Regression	0.3770	0.2075	0.9275	2.042	0.2202	0.0791	0.8993	2.868
SVM	0.6621	0.4723	0.9599	1.227	0.2744	0.1063	0.9005	2.763
RFC	0.9994	0.9990	0.9999	0.577	0.2439	0.0950	0.9027	4.527

Tabel 4.8 Tabel hasil pembuatan model machine learning menggunakan data hasil oversampling

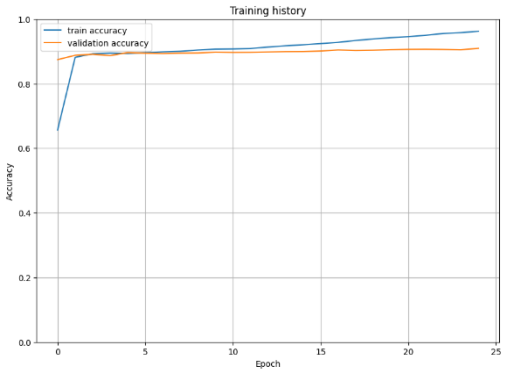
Model	**Data Latih (70%)**	**Data Uji (30%)**

	JA	***SA***	***LA***	***Loss***	***JA***	***SA***	***LA***	***Loss***
Naïve Bayes	0.4345	0.3410	0.9352	2.917	0.2813	0.2280	0.9169	4.193
Logistic Regression	0.5165	0.3997	0.9430	1.693	0.3666	0.2787	0.9237	2.291
SVM	0.7950	0.6560	0.9751	0.796	0.4561	0.3226	0.9265	2.274
RFC	1.0	1.0	1.0	0.429	0.3735	0.2905	0.9224	2.942

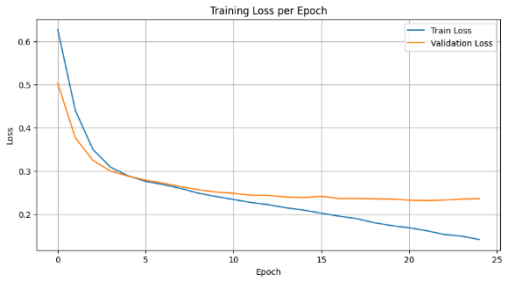
Pada Tabel 4.7, yang menggunakan data asli, seluruh model mengalami penurunan performa pada data uji, yang disebabkan oleh ketidakseimbangan dataset, mengakibatkan label pada data uji lebih sedikit dan membuat model kesulitan dalam melakukan prediksi. Naïve Bayes, Logistic Regression, SVM, dan RFC menunjukkan penurunan signifikan pada data uji, dengan RFC cenderung overfit pada data latih. Sementara itu, pada Tabel 4.8 yang menggunakan data hasil oversampling, performa model mengalami peningkatan, terutama pada Naïve Bayes dan Logistic Regression, yang lebih mampu menangani data minoritas yang seimbang. SVM dan RFC juga menunjukkan perbaikan, meskipun masih menghadapi penurunan kinerja pada data uji, yang mengindikasikan adanya tantangan dalam mengaplikasikan model ke data yang baru (data uji). Namun, hasil dari keseluruhan akurasi pada data latih hasil oversampling menunjukkan bahwa model SVM dan juga RFC berpotensi untuk melakukan tugas prediksi multilabel meskipun masih merupakan model machine learning konvensional.

4.4.2 Hasil Model Deep Learning

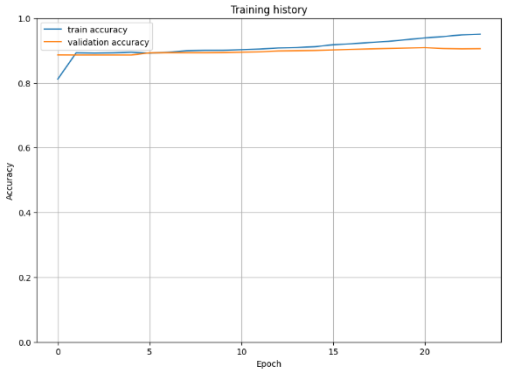
Hasil pembuatan model deep learning telah diterapkan pada kedua jenis dataset, yaitu dataset asli dan dataset hasil oversampling. Pada hasil model deep learning ini, akan ditampilkan performa dari empat model deep learning yang digunakan, yaitu BERT, BERT-LSTM, RoBERTa, dan DistilBERT. Masing-masing model akan dibagi menjadi tiga bagian data, yaitu 70% untuk data latih, 15% untuk data validasi, dan 15% untuk data uji pada masing-masing dataset. Evaluasi performa model akan dilakukan dengan mengamati hasil training dan loss per epoch dari setiap model, yang akan divisualisasikan dalam grafik untuk memberikan gambaran yang jelas tentang perkembangan performa selama proses pelatihan. Kemudian, setiap pembagian data akan dievaluasi berdasarkan ketiga metriks akurasi seperti yang dijelaskan pada Sub Bab 2.7 yaitu *Jaccard Accuracy (JA)*, *Subset Accuracy (SA)*, *Label-wise Accuracy (LA)*, dan metriks Loss.



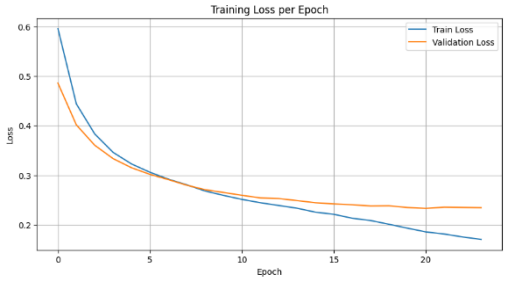
(a) Grafik akurasi pelatihan BERT



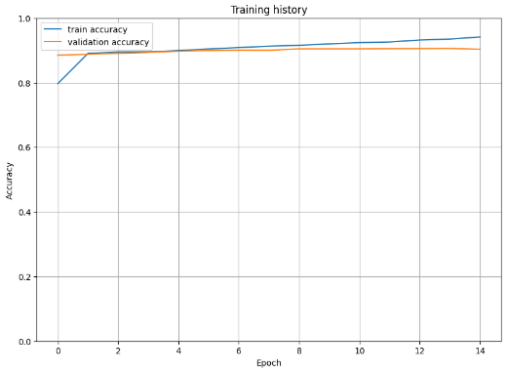
(b) Grafik loss pelatihan BERT



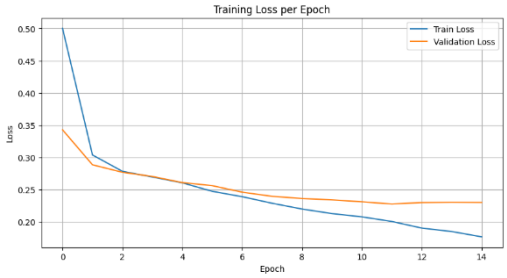
(c) Grafik akurasi pelatihan BERT-LSTM



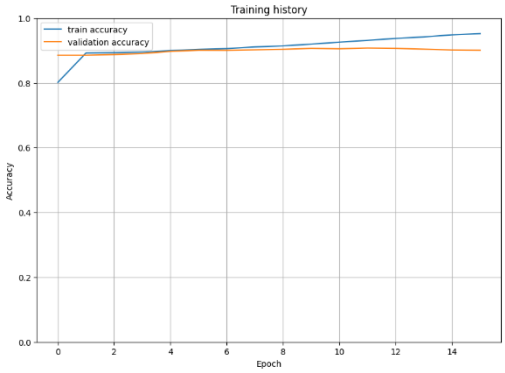
(d) Grafik loss pelatihan BERT-LSTM



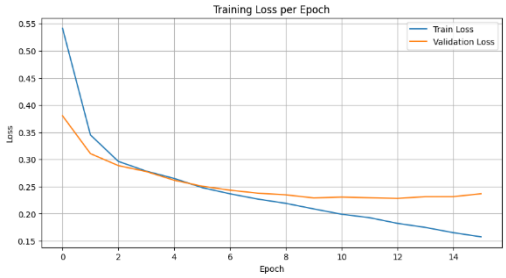
(e) Grafik akurasi pelatihan RoBERTa



(f) Grafik loss pelatihan RoBERTa

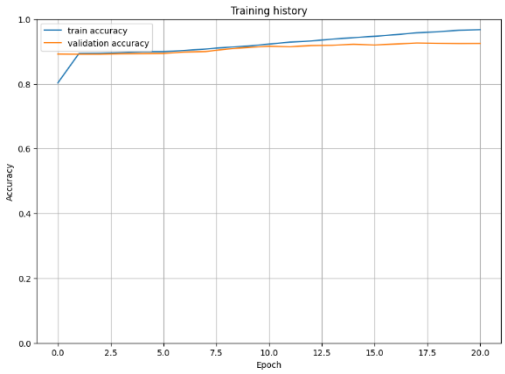


(g) Grafik akurasi pelatihan DistilBERT

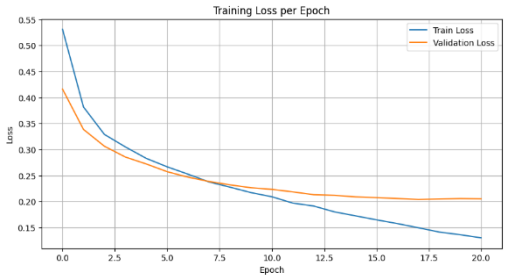


(h) Grafik loss pelatihan DistilBERT

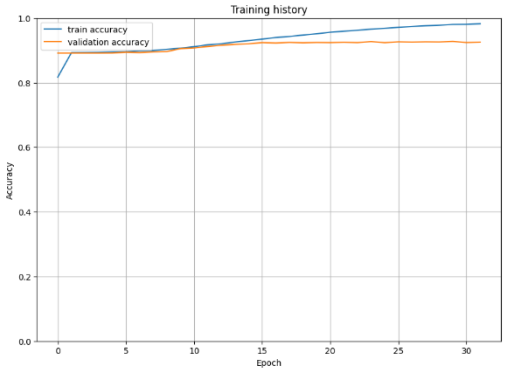
Gambar 4.5 Visualisasi hasil pelatihan akurasi dan loss model deep learning pada data asli.



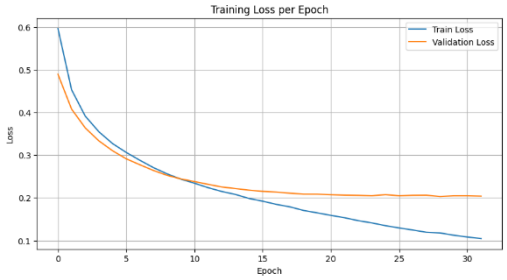
(a) Grafik akurasi pelatihan BERT



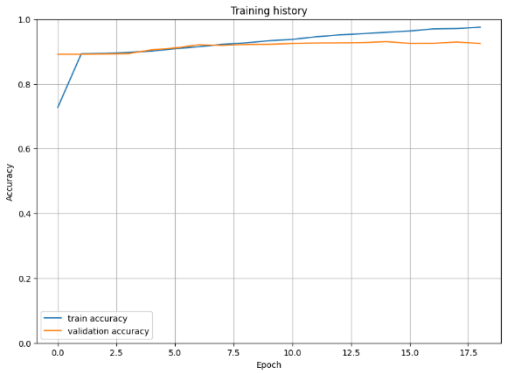
(b) Grafik loss pelatihan BERT



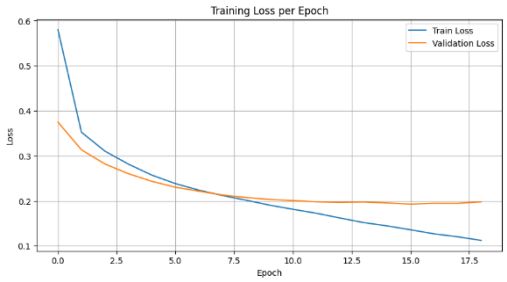
(c) Grafik akurasi pelatihan BERT-LSTM



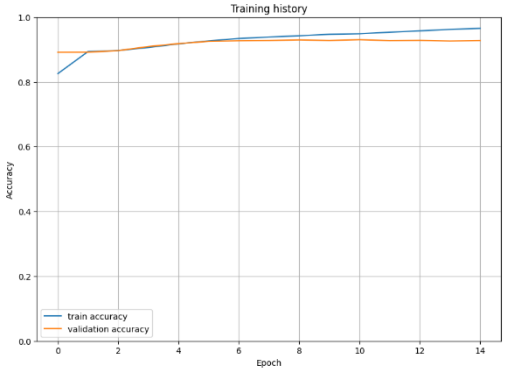
(d) Grafik loss pelatihan BERT-LSTM



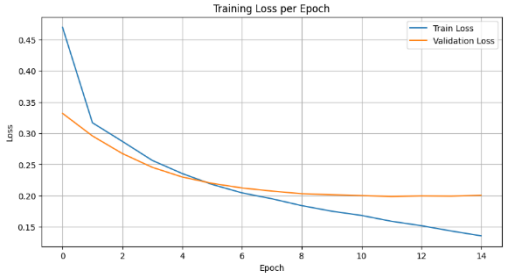
(e) Grafik akurasi pelatihan RoBERTa



(f) Grafik loss pelatihan RoBERTa



(g) Grafik akurasi pelatihan DistilBERT



(h) Grafik loss pelatihan DistilBERT

Gambar 4.6 Visualisasi hasil traning akurasi dan loss model deep learning pada data hasil oversampling.

Seperti yang terlihat pada Gambar 4.5 dan Gambar 4.6, grafik di sebelah kiri menunjukkan tingkat ketepatan (akurasi) model, sementara grafik di kanan menunjukkan tingkat kesalahannya (loss). Seperti yang terlihat, seiring berjalannya waktu, semua model menjadi semakin pintar. Ini ditandai dengan grafik akurasi yang terus naik dan grafik kesalahan yang terus turun, yang merupakan pertanda baik bahwa proses pelatihan berjalan sukses. Garis biru (data latihan) dan oranye (data valdiasi) yang saling berdekatan menunjukkan bahwa model tidak hanya menghafal, tetapi benar-benar memahami pola data.

Model yang akan digunakan pada pelatihan ini adalah model pada tiga epoch terakhir belum terjadi. Hal tersebut dilakukan karena, untuk mengetahui puncak performa dari model tersebut sebelum adanya overfitting yang diindikasikan dengan tidak adanya kenaikan yang signifikan pada akurasi serta terjadi kenaikan angka pada loss. Sebagai contoh pada model RoBERTa Gambar 4.5, dapat dilihat bahwa model akhir yang akan digunakan bukanlah model dari epoch ke-15, melainkan model yang disimpan pada titik performa puncaknya sebelum overfitting menjadi signifikan yaitu pada epoch ke-13. Salah satu cara untuk mengimplementasikannya adalah dengan mekanisme *early stopping* atau pemberhentian awal, di mana pelatihan dihentikan jika tidak ada perbaikan pada metrik validasi (misalnya, validation loss) selama beberapa epoch berturut-turut (misalnya, tiga epoch).

Secara keseluruhan, berdasarkan kedua gambar tersebut, baik dari data asli dan data hasil oversampling, tidak ada suatu perubahan yang signifikan semasa pelatihan model dari kedua data tersebut sehingga dapat di asumsikan bahwa model dapat belajar dan mengenali pola pada data dengan sangat baik untuk kedua data tersebut.

Tabel 4.9 Tabel hasil pembuatan model deep learning menggunakan data asli

Model	**Latih**				**Validasi**				**Uji**				**Epoch**
	JA	***SA***	***LA***	***Loss***	***JA***	***SA***	***LA***	***Loss***	***JA***	***SA***	***LA***	***Loss***	
BERT	0.75	0.63	0.96	0.14	0.35	0.10	0.90	0.23	0.32	0.11	0.90	0.22	22
BERT-LSTM	0.68	0.51	0.95	0.17	0.34	0.08	0.90	0.23	0.33	0.09	0.90	0.25	21
RoBERTa	0.61	0.42	0.95	0.15	0.32	0.08	0.90	0.23	0.34	0.09	0.89	0.23	13
DistilBERT	0.61	0.40	0.94	0.17	0.35	0.09	0.90	0.23	0.37	0.12	0.90	0.22	12

Tabel 4.10 Tabel hasil pembuatan model deep learning menggunakan data hasil oversampling

Model	Latih				Validasi				Uji				Epoch
	JA	*SA*	*LA*	*Loss*	*JA*	*SA*	*LA*	*Loss*	*JA*	*SA*	*LA*	*Loss*	
BERT	0.86	0.76	0.96	0.13	0.46	0.29	0.92	0.20	0.49	0.31	0.92	0.20	18
BERT-LSTM	0.91	0.82	0.98	0.10	0.48	0.31	0.92	0.20	0.50	0.32	0.92	0.19	29
RoBERTa	0.87	0.77	0.97	0.11	0.52	0.31	0.92	0.19	0.50	0.29	0.92	0.19	16
DistilBERT	0.78	0.63	0.96	0.13	0.48	0.29	0.92	0.20	0.49	0.30	0.92	0.20	12

Berdasarkan Tabel 4.9, evaluasi performa model deep learning yang menggunakan data asli menunjukkan sebuah kemampuan dasar dari setiap model. Pada tahap pengujian (Uji), model-model deep learning tersebut menunjukkan nilai Jaccard Accuracy (JA) sekitar 0.32 hingga dengan nilai tertinggi 0.37 yang diraih oleh model DistilBERT. Namun, skor Subset Accuracy (SA) yang dicapai relatif rendah untuk semua model, dengan nilai tertinggi hanya 0.11. Rendahnya skor SA ini mengindikasikan bahwa meskipun model cukup baik dalam mengenali label secara individual Label-wise Accuracy (LA) sekitar 0.90, model masih kesulitan untuk memprediksi seluruh kombinasi label dengan tepat untuk satu data.

Selanjutnya, Tabel 4.10 menyajikan hasil setelah data latih diproses menggunakan teknik oversampling. Pada tabel ini, terlihat adanya peningkatan performa yang sangat jelas dan konsisten di semua model. Sebagai contoh, model BERT-LSTM pada tahap pengujian mengalami lonjakan performa yang signifikan, dengan skor JA meningkat menjadi 0.50 dan skor SA naik hampir tiga kali lipat menjadi 0.32. Peningkatan ini juga didukung oleh penurunan nilai Loss dari 0.25 menjadi 0.19, yang menandakan prediksi model menjadi lebih akurat. Perbandingan antara kedua tabel tersebut secara menunjukkan bahwa penerapan teknik oversampling merupakan strategi yang sangat efektif untuk meningkatkan kinerja model pada penelitian multilabel ini.

4.5 Hasil Evaluasi Model

Pada sub bab ini, disajikan hasil evaluasi dari seluruh model Machine Learning dan Deep Learning yang telah diuji menggunakan data uji. Pemaparan hasil evaluasi diawali dengan analisis terhadap pola kesalahan klasifikasi melalui visualisasi confusion matrix untuk masing-masing model. Selanjutnya, dipaparkan kinerja kuantitatif setiap model secara lebih detail berdasarkan metrik evaluasi yang mencakup precision, recall, dan F1-score. Sebagai tahap akhir, sub bab ini menyajikan hasil dari skenario pengujian, di mana prediksi model terhadap

beberapa teks dengan berbagai konteks emosi diamati secara langsung. Pada setiap skenario pengujian ini, disertakan pula pengukuran *inference time* untuk menganalisis efisiensi dan kecepatan komputasi model dalam menghasilkan sebuah prediksi.

Pada skenario pengujian prediksi model terhadap teks dari data uji, akan diberikan 5 teks dengan label asli (*ground label*) dan masing-masing teks memiliki karakteristik yang berbeda sebagai berikut:

1. Teks dengan banyak label emosi

   "I was very happy to be out enjoying some time off from work, and was able to buy some things I had been looking forward to for my collecting hobby. It was the highlight of my week thus far."

   Dengan label asli yaitu: Calm (CA), Excited (EX), Happy (HP), Satisfied (ST).
2. Teks dengan label paling sedikit muncul (minor)

   "My friend recently told me about a promotion he'd gotten. I was proud and happy for him, but I was also jealousy. I've been working on a promotion myself for a year now, and there's no sign it's coming any time soon."

   Dengan label asli yaitu: Jealous (JL) dan Proud (PR).
3. Teks dengan satu label

   "Praying for me is not so much about asking God for things but instead pray for stength to do whats right to be the man I am intended to be. To be a good father and husband and example. To hopefully plant the seeds of faith in my children that our life has a purpose beyond just the physical observable realm"

   Dengan label asli yaitu: Calm (CA).
4. Teks dengan panjang teks terpendek

   "i slept 8 hours."

   Dengan label asli yaitu: Happy (HP).
5. Teks dengan panjang teks terpanjang

   "I had a good selection of tasks yesterday, and they exceeded my goals (in terms of hours worked and earnings.) Knowing that the weather had turned cold and miserable, and that at the end of the week my work day's were going to be interrupted by other responsibilities, it was that much more satisfying that I could get more than the average amount of work accomplished yesterday. I was also able to concentrate on/organize some upcoming chores and personal projects, especially some photography I want to

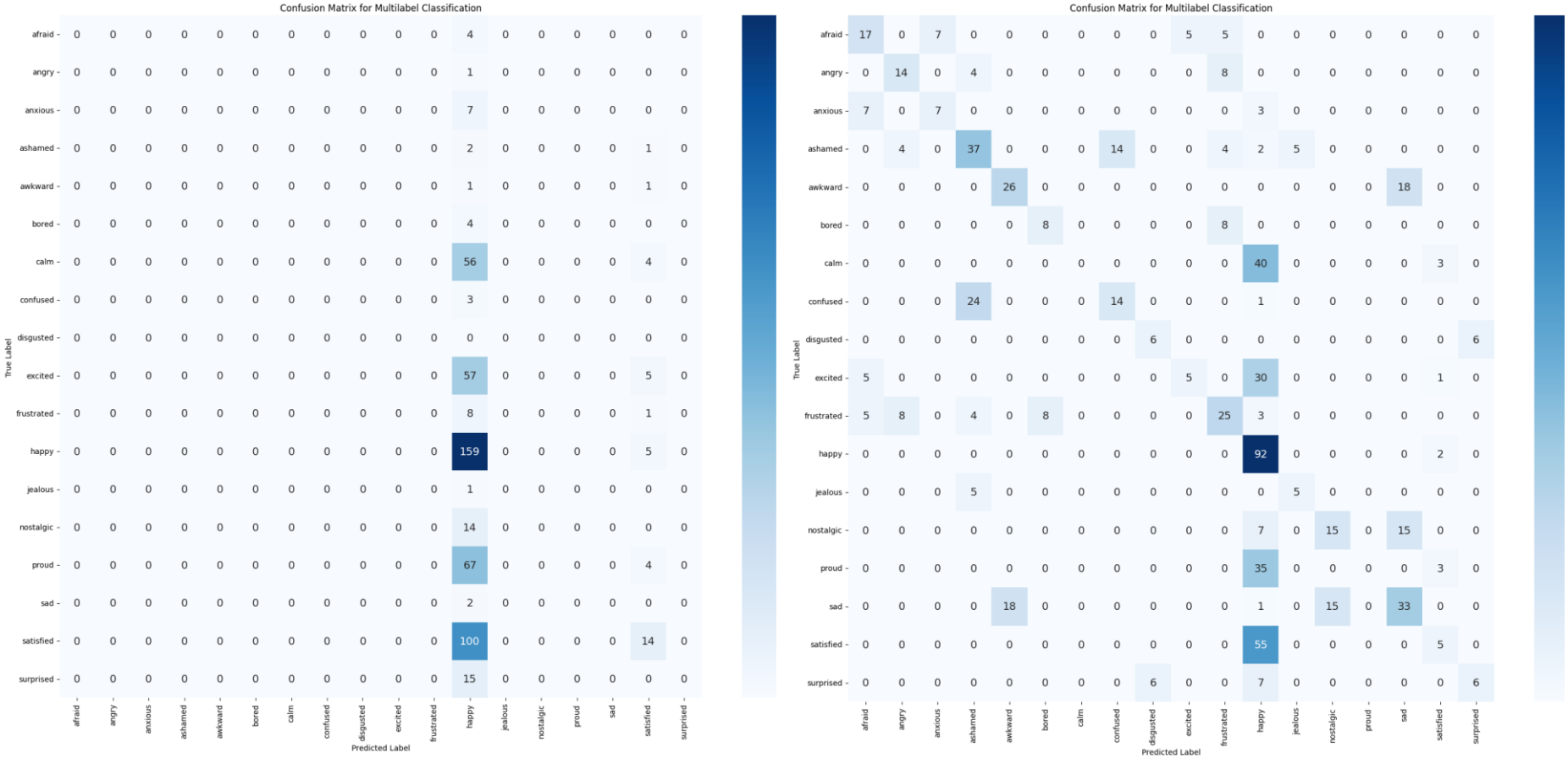
accomplish. Earning more money now puts me at ease about missing some work time later this month when I pursue these types of personal projects."

Dengan label asli yaitu: Calm (CA), Happy (HP), Satisfied (ST).

4.5.1 Hasil Evaluasi Model Machine Learning

Evaluasi model machine learning pada data uji dilakukan melalui beberapa tahap. Pertama, pola kesalahan dianalisis menggunakan confusion matrix. Selanjutnya, kinerja model diukur secara kuantitatif dengan metrik makro untuk masing-masing precision, recall, dan F1-score. Terakhir, dilakukan pengujian pada beberapa sampel teks secara langsung serta diukur waktu inferensi untuk menilai kecepatan dan efisiensi model.

Naïve Bayes

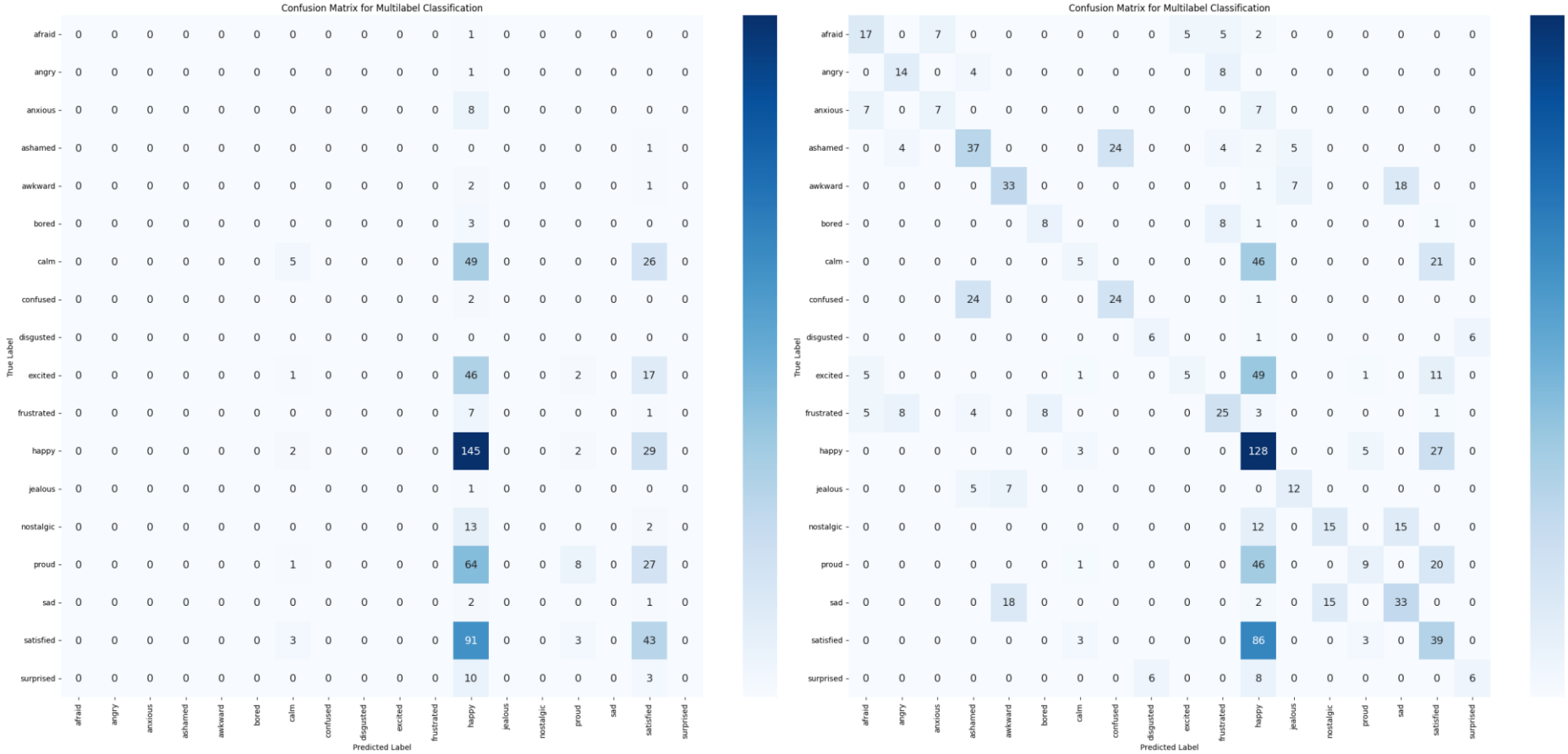


(a) Data asli

(b) Data hasil oversampling

Gambar 4.7 Confussion matrix model naive bayes pada data asli dan data hasil oversampling.

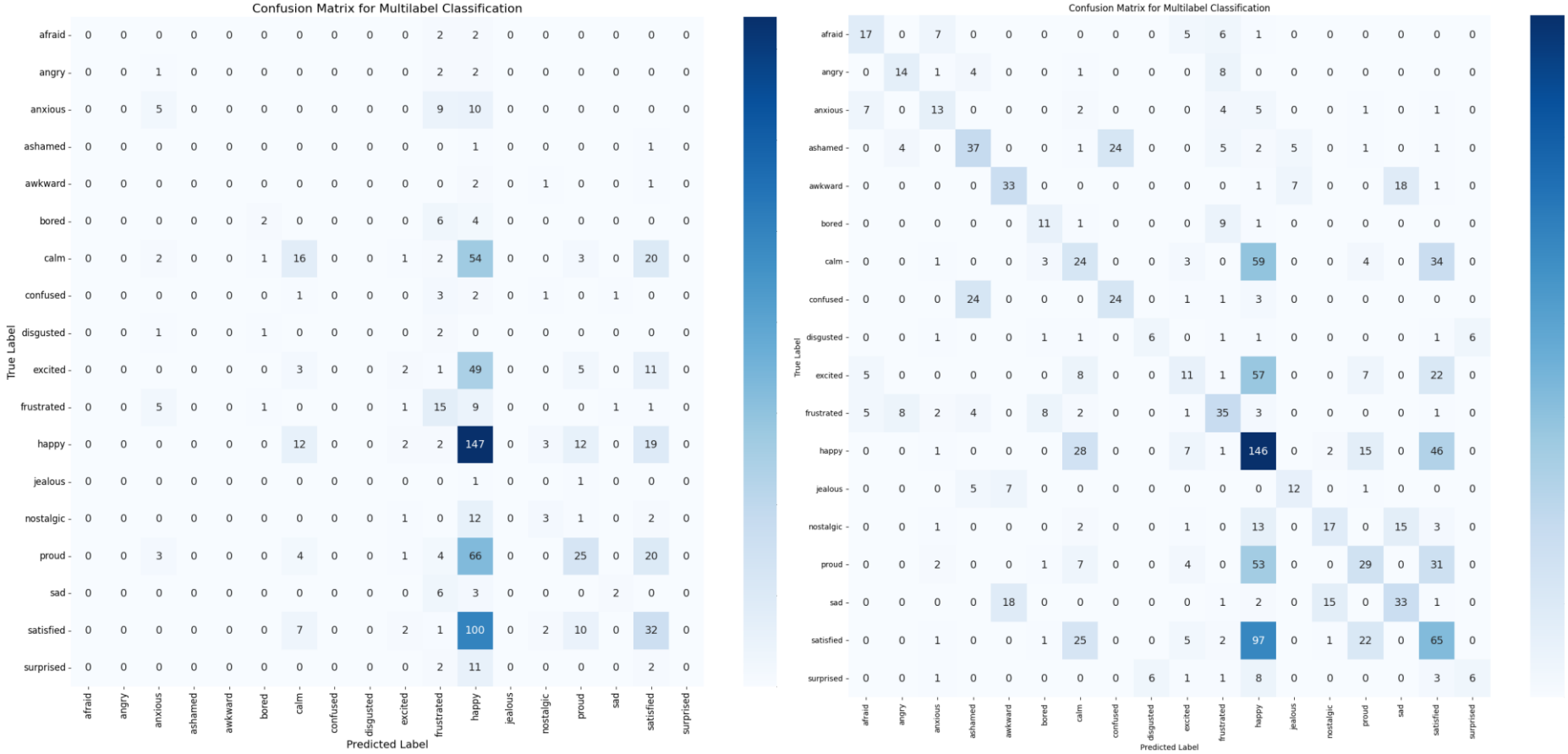
Logistic Regression



(a) Data asli (b) Data hasil oversampling

Gambar 4.8 Confussion matrix model logistic regression pada data asli dan data hasil oversampling.

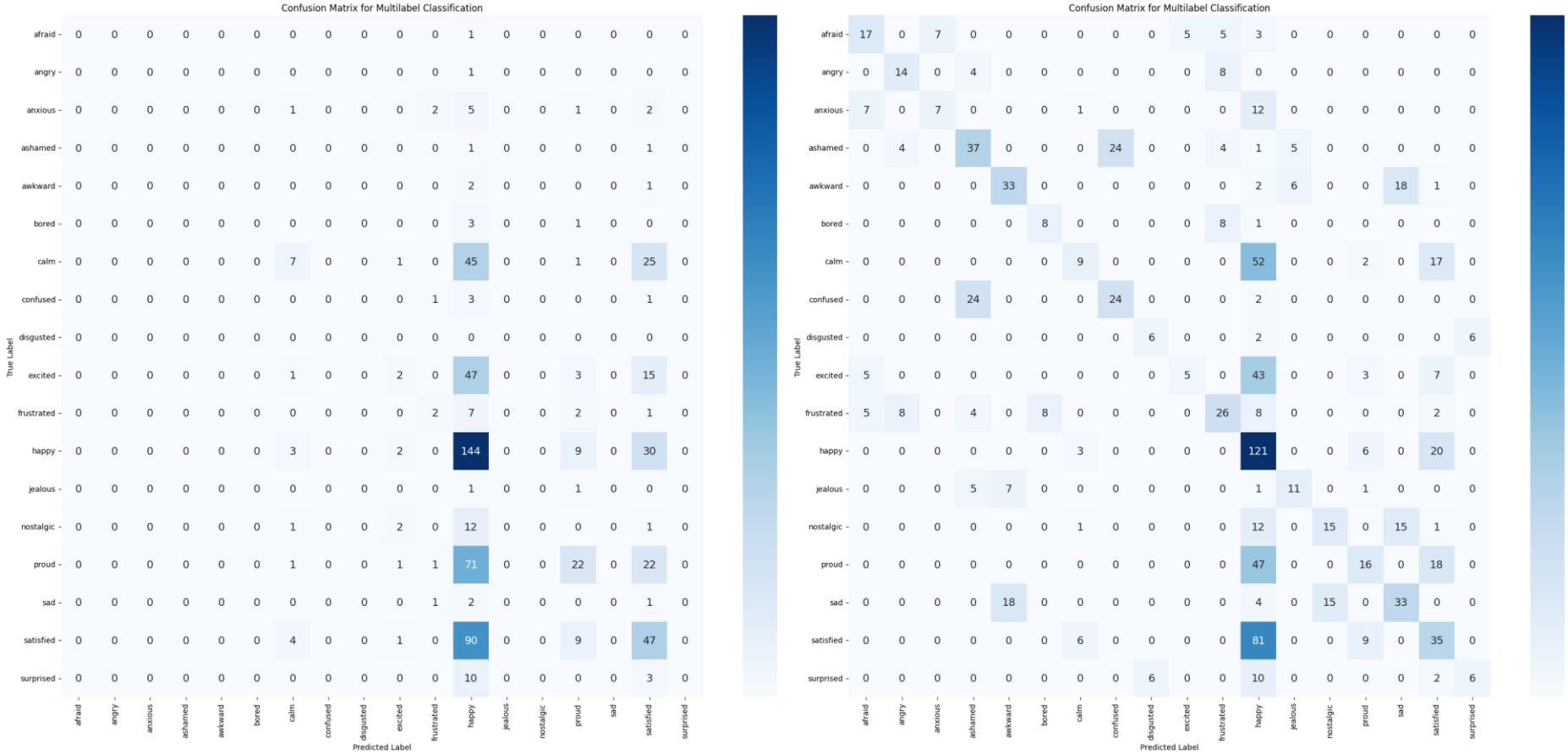
Support Vector Machine



(a) Data asli (b) Data hasil oversampling

Gambar 4.9 Confussion matrix model SVM pada data asli dan data hasil oversampling.

Random Forest Classifier



(a) Data asli (b) Data hasil oversampling

Gambar 4.10 Confussion matrix model RFC pada data asli dan data hasil oversampling

Pada Gambar 4.7 hingga Gambar 4.10, dapat dilihat bahwa secara konsisten keseluruhan model menunjukkan peningkatan performa yang baik setelah penerapan teknik oversampling. Sebelum oversampling, confusion matrix pada data asli menunjukkan bahwa keseluruhan model cenderung melakukan prediksi emosi kepada label-label yang mayoritas. Total nilai yang muncul pada confusion matrix ini adalah 536 sedangkan, total data pada data uji adalah 442 seperti pada Sub BAB 4.3. Perbedaan nilai ini menjadi sifat multilabel yang menunjukkan bahwa satu data text dapat memiliki satu atau lebih label emosi yang di mana akan menjadi nilai pada confusion matrix tersebut. Untuk memahaminya pada label emosi mana kah yang bernilai benar atau sesuai prediksi seperti pada label asli, kita perlu membaca perpotongan antara Label Sebenarnya (sumbu vertikal) dan Label Prediksi (sumbu horizontal), di mana angka pada diagonal adalah prediksi yang benar, sedangkan diluar diagonal tersebut maka label tersebut salah prediksi dan tidak seperti pada label aslinya (*ground truth*).

Sebagai contoh, pada model Naïve Bayes (Gambar 4.7a) pada data asli berhasil memprediksi 159 data *happy* dan 14 data *satisfied* dengan benar. Namun, bias ini menyebabkan model secara keliru mengklasifikasikan mayoritas emosi lain sebagai *happy*, seperti 100 data *satisfied* dan 56 data *calm* yang salah diprediksi. Akibatnya, banyak emosi minoritas seperti *afraid* dan *angry* tidak memiliki prediksi benar sama sekali (nilai diagonal 0). Kecenderungan serupa juga terlihat pada model lain seperti SVM dan RFC, meskipun variasi label yang muncul pada keduanya sudah sedikit lebih banyak.

Setelah penerapan oversampling, seluruh model menunjukkan peningkatan yang tercermin dari semakin bervariasinya label emosi yang terdeteksi. Pada Naïve Bayes (Gambar 4.7b), matriks setelah oversampling menunjukkan distribusi prediksi yang jauh lebih merata. Model kini mampu mengenali emosi yang sebelumnya gagal total, seperti menghasilkan 17 prediksi benar untuk *afraid*, 37 untuk *ashamed*, dan 40 untuk calm. Peningkatan ini menegaskan bahwa model tidak lagi hanya bergantung pada label mayoritas, terbukti dari naiknya prediksi benar untuk *satisfied* menjadi 55 dan turunnya prediksi *happy* menjadi 92. Penurunan pada label *happy* ini justru merupakan pertanda positif yang menandakan model telah berhasil mempelajari fitur emosi yang lebih luas. Meskipun label mayoritas seperti *happy* dan *satisfied* masih dominan, distribusi nilai emosi mulai merata di seluruh label. Begitu juga pada model Logistic Regression, SVM, dan RFC, meskipun ada beberapa label emosi yang dominan, munculnya variasi label semakin jelas pada seluruh model.

Tabel 4.11 Tabel hasil evaluasi model machine learning

Model	Data asli			Data hasil oversampling		
	Macro Precission	**Macro Recall**	**Macro F1-Score**	**Macro Precission**	**Macro Recall**	**Macro F1-Score**
Naïve Bayes	0.08	0.04	0.05	0.84	0.39	0.50
Logistic Regression	0.17	0.06	0.07	0.94	0.47	0.58
SVM	0.40	0.12	0.16	0.87	0.53	0.64
RFC	026	0.07	0.09	0.94	0.47	0.58

Tabel 4.11 Tabel hasil evaluasi model machine learningmenunjukkan hasil evaluasi performa model machine learning pada data asli dan data hasil oversampling, dengan metrik yang mencakup Macro Precision, Macro Recall, dan Macro F1-Score. Pada data asli, model-model seperti Naïve Bayes dan Logistic Regression menunjukkan performa yang sangat rendah, dengan nilai Macro Precision, Macro Recall, dan Macro F1-Score yang hanya sekitar 0.08, 0.04, dan 0.05 untuk Naïve Bayes, dan 0.17, 0.06, dan 0.07 untuk Logistic Regression. Namun, setelah dilakukan oversampling, terjadi peningkatan yang signifikan pada kedua model tersebut, dengan Naïve Bayes mencapai 0.84 untuk Macro Precision, 0.39 untuk Macro Recall, dan 0.50 untuk Macro F1-Score. Begitu pula dengan Logistic Regression yang meningkat menjadi 0.94 pada Macro Precision, 0.47 pada Macro Recall, dan 0.58 pada Macro F1-Score.

Pada model SVM, meskipun sudah menunjukkan performa yang lebih baik pada data asli (Macro Precision 0.40, Macro Recall 0.12, Macro F1-Score 0.16), oversampling memberikan peningkatan yang cukup signifikan, dengan nilai Macro Precision mencapai 0.87, Macro Recall 0.53, dan Macro F1-Score 0.64. Model RFC juga mengalami peningkatan setelah oversampling, meskipun nilai awal pada data asli lebih rendah dibandingkan dengan SVM (Macro Precision 0.26, Macro Recall 0.07, Macro F1-Score 0.09). Setelah oversampling, RFC menunjukkan peningkatan menjadi 0.94 pada Macro Precision, 0.47 pada Macro Recall, dan 0.58 pada Macro F1-Score.

Secara keseluruhan, penerapan teknik oversampling memberikan dampak yang baik terhadap kemampuan model dalam mengenali variasi label emosi, mengurangi ketidakseimbangan kelas pada data, serta meningkatkan kinerja prediksi model agar bisa lebih akurat dalam memprediksikan emosi.

Tabel 4.12 Tabel hasil skenario pengujian model machine learning

Model	Teks 1		Teks 2		Teks 3		Teks 4		Teks 5	
	Label	Time (ms)	Label	Time (ms)	Label	Time (ms)	Label	Time (ms)	Label	Time (ms)
Data asli										
Naïve Bayes	HP	9.5	HP	11	-	12.3	-	10.8	-	10.5
Logistic Regression	HP	4.7	HP	3.3	-	4.2	-	3.4	-	4
SVM	HP	7.8	PR, HP	6.5	CA	6.4	-	4.2	HP	7.6
RFC	HP	108.8	HP, PR	131.1	HP	113.7	-	140.8	-	126.6
Data hasil oversampling										
Naïve Bayes	-	5.2	-	5.6	HP	8.4	-	10.8	-	7.3
Logistic Regression	HP, ST	3.5	-	2.5	-	5.2	-	3.3	-	4.5
SVM	HP, ST	8.4	PR	8.1	CA	8.5	-	7	ST	10.3
RFC	HP	121	PR, HP	134.3	-	144.7	HP	134.8	-	129

Seperti yang dijelaskan pada Hasil Evaluasi Model 4.5, Analisis skenario pengujian pada Tabel 4.11 Tabel hasil evaluasi model machine learning menunjukkan perbandingan kinerja antara model yang dilatih menggunakan data asli dengan data hasil oversampling. Secara keseluruhan, oversampling terbukti meningkatkan kualitas prediksi model, terutama pada kasus-kasus yang kompleks.

Sebagai contoh, pada Teks 1 yang memiliki banyak label, model SVM dan Logistic Regression yang dilatih dengan data oversampling mampu mengenali label tambahan ST (Satisfied), sebuah peningkatan dari model pada data asli yang hanya mampu memprediksi HP (Happy). Peningkatan paling signifikan terlihat pada Teks 2 yang mengandung label minoritas, di mana model SVM pada data oversampling memberikan prediksi yang lebih presisi yaitu PR (Proud), dibandingkan dengan model pada data asli yang keliru menambahkan label mayoritas HP. Meskipun demikian, baik pada data asli maupun hasil oversampling, sebagian besar model masih kesulitan memberikan prediksi akurat untuk teks yang sangat pendek (Teks 4) dan sangat panjang (Teks 5), yang menunjukkan adanya batasan pada kedua kondisi data.

Dari sisi kecepatan komputasi, perbandingan menunjukkan perbedaan yang sangat besar antar arsitektur model itu sendiri. Model Random Forest (RFC) secara konsisten sangat lambat yaitu di atas 100 ms, model svm dan naive bayes menunjukkan performa yang cepat dan konsisten yaitu dikisaran 12 hingga 4 ms sementara Logistic Regression sangat cepat yaitu di

bawah 5 ms, terlepas dari apakah menggunakan data asli atau hasil oversampling. Hal ini menggarisbawahi adanya perbedaan antara akurasi dan efisiensi komputasi yang melekat pada pemilihan model yang menjadi karakteristik dari model-model tersebut pada penelitian ini.

4.5.2 Hasil Evaluasi Model Deep Learning

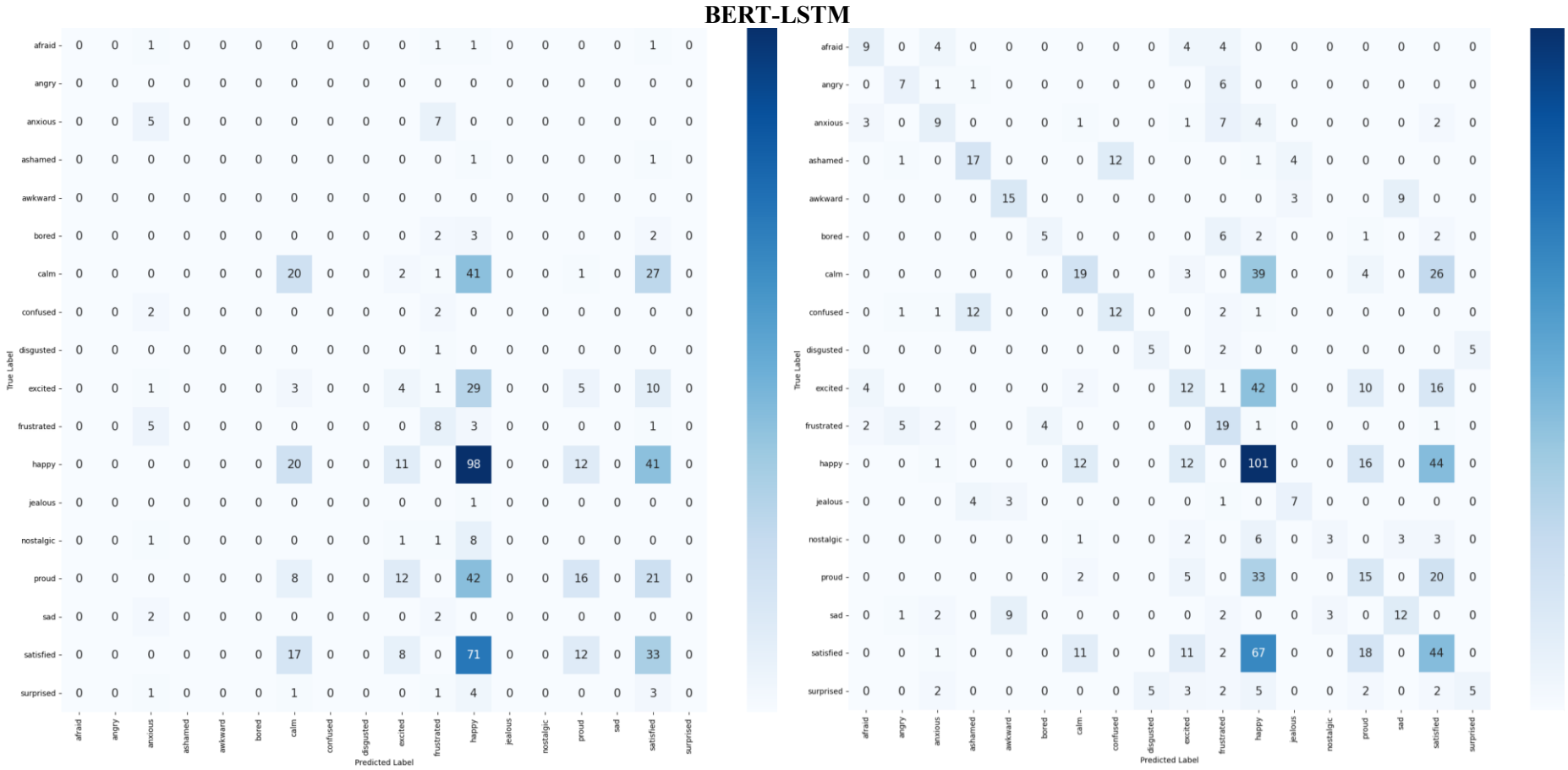
Evaluasi model deep learning pada data uji dilakukan melalui beberapa tahap. Pertama, pola kesalahan dianalisis menggunakan confusion matrix. Selanjutnya, kinerja model diukur secara kuantitatif dengan metrik makro untuk masing-masing precision, recall, dan F1-score. Terakhir, dilakukan pengujian pada beberapa sampel teks secara langsung serta diukur waktu inferensi untuk menilai kecepatan dan efisiensi model.

BERT



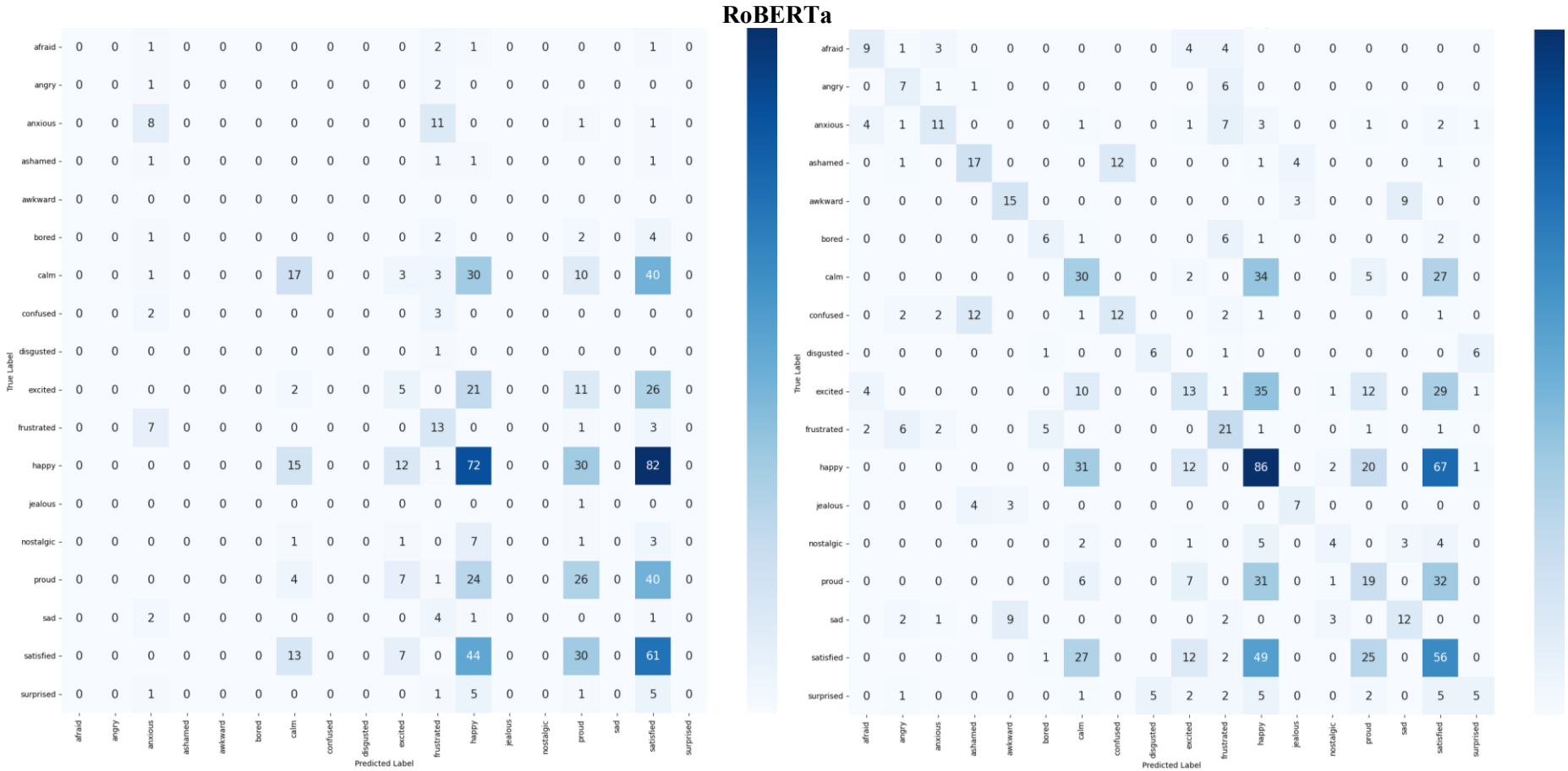
(a) Data asli (a) Data hasil oversampling

Gambar 4.11 Confusion matrix model BERT pada data asli dan data hasil oversampling.



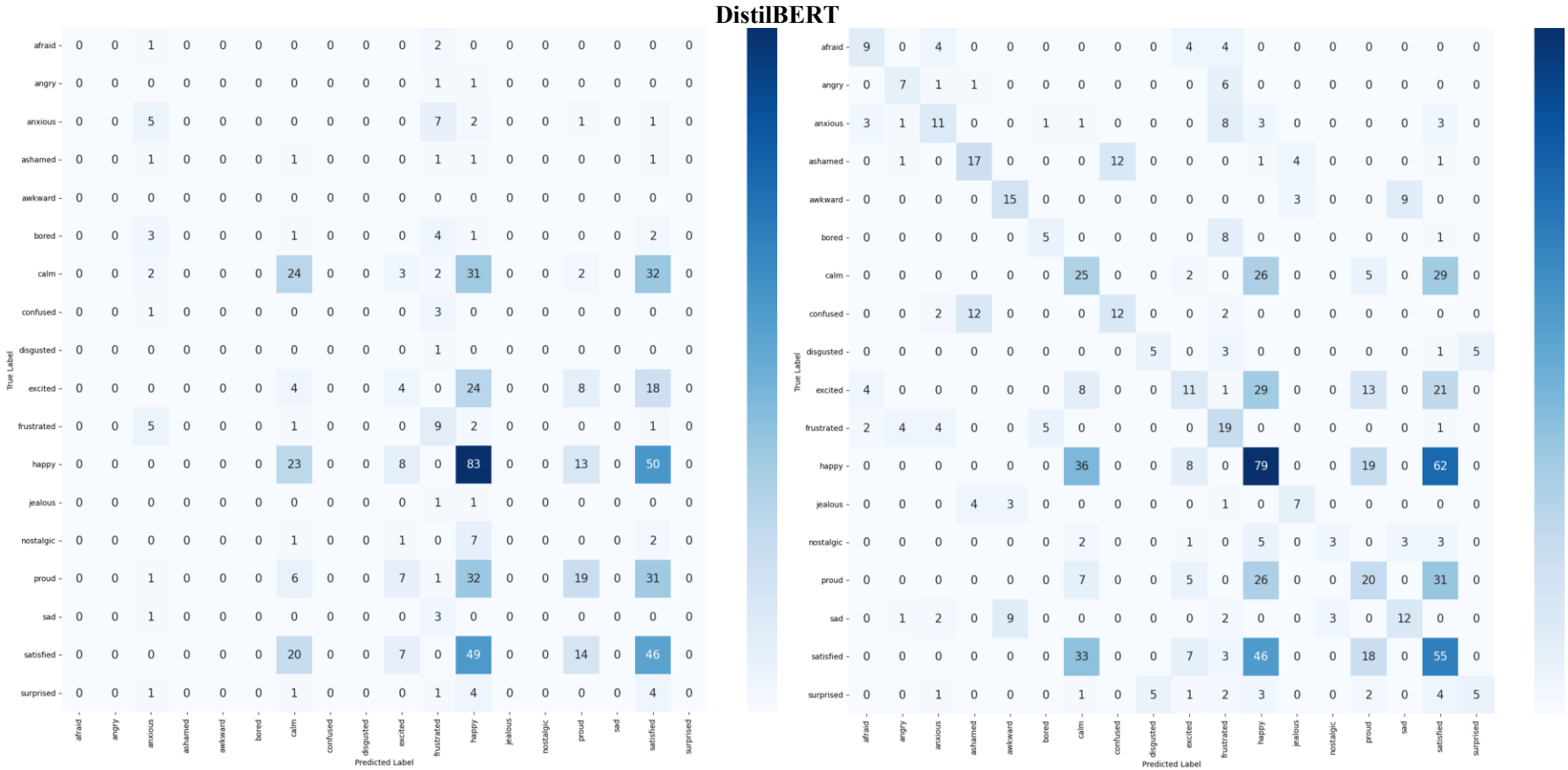
(a) Data asli (b) Data hasil oversampling

Gambar 4.12 Confusion matrix model BERT-LSTM pada data asli dan data hasil oversampling.



(a) Data asli (b) Data hasil oversampling

Gambar 4.13 Confusion matrix model RoBERTa pada data asli dan data hasil oversampling.



(a) Data asli (b) Data hasil oversampling

Gambar 4.14 Confusion matrix model distilBERT pada data asli dan data hasil oversampling.

Pada Gambar 4.11 hingga Gambar 4.14, dapat dilihat bahwa perbandingan confusion matrix antara data asli dengan data hasil oversampling menunjukkan performa keseluruhan model dalam menangkap label emosi menjadi semakin baik. Berbeda dengan model machine learning yang hanya menangkap label emosi mayor yaitu happy dan satisfied, pada data asli keseluruhan model deep learning sudah dapat melakukan prediksi pada emosi yang lebih beragam. Untuk memahami performanya, kita dapat membaca perpotongan antara Label Sebenarnya (sumbu vertikal) dan Label Prediksi (sumbu horizontal), di mana angka pada diagonal menunjukkan prediksi yang benar sesuai dengan label asli (*ground truth*).

Sebagai contoh, pada model DistilBERT (Gambar 4.14a) mampu memprediksi dengan baik beberapa label seperti *happy* (83), *satisfied* (46), *proud* (32), dan *sad* (31). Meskipun demikian, model masih menunjukkan kecenderungan untuk salah memprediksi beberapa emosi sebagai label mayoritas. Sebagai contoh, sebanyak 31 data emosi *calm* dan 24 data *excited* salah diprediksi sebagai *happy*. Selain itu, beberapa label minoritas seperti *afraid*, *angry*, *awkward*, dan *jealous* belum memiliki prediksi yang benar sama sekali (nilai diagonal 0), yang menunjukkan bahwa bias terhadap kelas mayoritas masih ada.

Setelah penerapan teknik oversampling, dapat dilihat bahwa model DistilBERT (Gambar 4.14b) mampu melakukan prediksi label yang lebih merata dan akurat. Matriks pada data hasil oversampling menunjukkan peningkatan signifikan pada kemampuan model untuk mengenali label-label minoritas yang sebelumnya gagal terdeteksi. Contohnya, prediksi yang benar untuk *afraid* meningkat dari 0 menjadi 9, *ashamed* dari 1 menjadi 17, dan *frustrated* dari 9 menjadi 19. Peningkatan ini menegaskan bahwa model tidak lagi terlalu bergantung pada label mayoritas dan telah berhasil mempelajari fitur-fitur unik dari setiap emosi. Hal ini juga tercermin dari distribusi yang lebih seimbang pada kelas-kelas yang sebelumnya sudah baik, seperti naiknya prediksi benar untuk *satisfied* dari 46 menjadi 55.

Secara keseluruhan, semua model dapat melakukan prediksi pada label minoritas seperti *nostalgic*, *bored*, *jealous*, dan *disgusted* yang sebelumnya memiliki nilai 0 pada prediksi yang benar sesuai dengan label asli. Diantara ke-empat model tersebut, dapat dilihat bahwa RoBERTa (Gambar 4.133b) memiliki performa yang lebih unggul terhadap pemerataan label, hal ini dapat dilihat pada label minoritas yang dapat diprediksi oleh model robert memiliki nilai lebih banyak dari pada model-model lainnya yaitu pada *nostalgic* (4), *bored* (6), *jealous* (7), dan *disgusted* (6).

Tabel 4.13 Tabel hasil evaluasi model deep learning

Model	Data asli			Data hasil oversampling		
	Macro Precission	Macro Recall	Macro F1-Score	Macro Precission	Macro Recall	Macro F1-Score
BERT	0.26	0.14	0.17	0.86	0.59	0.68
BERT-LSTM	0.23	0.15	0.17	0.86	0.49	0.67
RoBERTA	0.24	0.18	0.20	0.80	0.63	0.69
DistilBERT	0.23	0.17	0.19	0.85	0.60	0.68

Tabel 4.13 menunjukkan hasil evaluasi model deep learning menggunakan data asli dan data hasil oversampling dengan metrik Macro Precision, Macro Recall, dan Macro F1-Score. Pada data asli, model-model BERT, BERT-LSTM, RoBERTA, dan DistilBERT menunjukkan kinerja yang cukup rendah dengan nilai Macro Precision, Macro Recall, dan Macro F1-Score yang berada di kisaran 0.23 hingga 0.26 untuk Precision, 0.14 hingga 0.18 untuk Recall, dan 0.17 hingga 0.20 untuk F1-Score. Meskipun nilai ini terbilang rendah, RoBERTA sedikit menunjukkan performa yang lebih baik di antara model-model lainnya dengan nilai Macro Recall 0.18 dan Macro F1-Score 0.20.

Setelah dilakukan oversampling, seluruh model mengalami peningkatan pada ketiga metrik. Model BERT, yang awalnya memiliki Macro Precision 0.26 dan Macro Recall 0.14 pada data asli, meningkat menjadi 0.86 untuk Precision dan 0.59 untuk Recall, dengan Macro F1-Score yang mencapai 0.68. Model BERT-LSTM juga menunjukkan peningkatan yang serupa, dengan nilai Macro Precision 0.86, Macro Recall 0.49, dan Macro F1-Score 0.67 setelah oversampling. RoBERTA dan DistilBERT juga menunjukkan peningkatan yang signifikan, dengan RoBERTA mencapai 0.80 pada Precision, 0.63 pada Recall, dan 0.69 pada F1-Score, sementara DistilBERT mencatatkan nilai 0.85 pada Precision, 0.60 pada Recall, dan 0.68 pada F1-Score.

Secara keseluruhan, penerapan oversampling memberikan dampak yang baik terhadap performa model deep learning ini. Semua model menunjukkan peningkatan yang signifikan pada Precision, Recall, dan F1-Score, yang menunjukkan bahwa teknik oversampling efektif dalam membantu model-model tersebut mengenali variasi emosi dalam data yang lebih seimbang, meningkatkan akurasi, serta meningkatkan kemampuan model dalam menghasilkan prediksi yang lebih baik.

Tabel 4.14 Tabel hasil skenario pengujian model deep learning

Model	**Teks 1**	**Teks 2**	**Teks 3**	**Teks 4**	**Teks 5**

	Label	Time (ms)	Label	Time (ms)	Label	Time (ms)	Label	Time (ms)	Label	Time (ms)
Data asli										
BERT	HP	30.3	-	31.6	CA, HP, ST	28.3	-	29.1	ST	30.6
BERT-LSTM	HP	34.4	HP	34.1	CA, HP, ST	35.2	HP	33.2	CA, ST	34.8
RoBERTA	EX, HP, PR, ST	30.3	-	28.8	PR, ST	27.7	-	28.1	ST	29
DistilBERT	HP	15.5	HP, PR	15.5	CA, ST	16.6	-	15.5	CA, PR, ST	16.6
Data hasil oversampling										
BERT	HP	30.2	-	28.4	HP, ST	28.3	HP	27.5	CA, ST	29.7
BERT-LSTM	HP	21.8	-	35.5	CA, HP, ST	31.8	HP	30.8	CA, HP, ST	30.4
RoBERTA	CA, HP, ST	28.4	-	25.6	CA, HP, PR, ST	29	HP	28.2	CA, ST	29.3
DistilBERT	HP, ST	15.9	-	15.5	CA, ST	15.3	-	14.8	CA, ST	16.7

Analisis skenario pengujian pada Tabel 4.14 menunjukkan bahwa perbandingan antara model deep learning yang dilatih menggunakan data asli dengan data hasil oversampling memberikan dampak yang beragam pada akurasi prediksi, sementara waktu inferensi lebih ditentukan oleh arsitektur model. Di satu sisi, oversampling terbukti mampu meningkatkan kualitas prediksi secara signifikan pada kasus tertentu.

Pada Teks 5 degan label asli Calm (CA), Happy (HP), Satisfied (ST), pada data asli, sebagian besar model hanya mampu mengenali sebagian kecil emosi. Seperti pada BERT dan RoBERTa hanya memprediksi ST, sementara BERT-LSTM sedikit lebih baik dengan prediksi CA, ST. Setelah oversampling, performa BERT-LSTM meningkat drastis menjadi sempurna dengan prediksi CA, HP, ST. Peningkatan juga terlihat pada Teks 4 dengan label asli Happy (HP), di mana pada data asli, hanya BERT-LSTM yang berhasil memberikan prediksi HP yang benar, sementara model lain gagal. Setelah oversampling, BERT dan RoBERTa juga menjadi mampu memberikan prediksi HP yang benar.

Namun, di sisi lain, oversampling justru dapat menurunkan performa, terutama pada Teks 2 dengan label asli Jealous (JL) dan Proud (PR). Pada data asli, hanya DistilBERT yang

menunjukkan keberhasilan parsial dengan mampu mengenali PR (dalam prediksinya HP, PR), sementara model lain gagal total atau salah prediksi. Namun, setelah dilatih dengan data yang hasil oversampling, kemampuan DistilBERT ini justru hilang dan gagal memberikan prediksi sama sekali.

Sementara itu, dari sisi waktu inferensi, perlakuan data tidak memberikan pengaruh, yang berarti perbedaan kecepatan murni disebabkan oleh kompleksitas atau arsitektur model itu sendiri. DistilBERT secara konsisten menjadi yang tercepat dengan waktu sekitar 15 ms, hampir dua kali lebih cepat dibandingkan BERT, BERT-LSTM, dan RoBERTa yang berada di rentang 25-35 ms pada kedua jenis data. Oversampling dapat memperbaiki deteksi multi-label pada beberapa teks dimana yang sebelumnya model tersebut tidak dapat memunculkan emosi pada teks tersebut menjadi bisa. Namun, masih terdapat pada kasus label minoritas seperti jealous (JL) yang masih tidak dapat terprediksi oleh model.

4.6 Pembahasan

Bagian pembahasan ini memaparkan tiga aspek utama yang menjadi inti dari hasil penelitian. Aspek pertama adalah karakteristik dataset Journal Entries with Labelled Emotions yang berperan penting dalam menjelaskan perbedaan performa pada setiap metode pemodelan. Aspek kedua adalah hasil evaluasi kelompok model machine learning yang terdiri dari Naïve Bayes, Logistic Regression, SVM, dan Random Forest Classifier. Pembahasan pada kelompok ini berfokus pada perbandingan performa antar model dalam lingkup machine learning dengan menggunakan pembagian data 70 persen untuk pelatihan dan 30 persen untuk pengujian. Aspek ketiga adalah hasil evaluasi kelompok model deep learning yang mencakup BERT, BERT-LSTM, DistilBERT, dan RoBERTa. Pada kelompok ini, pembahasan diarahkan untuk membandingkan performa antar arsitektur deep learning dengan pembagian data 70 persen untuk pelatihan, 15 persen untuk validasi, dan 15 persen untuk pengujian.

4.6.1 Pembahasan Karakteristik Data

Karakteristik dari dataset ‘Journal Entries with Labelled Emotions’ merupakan faktor fundamental yang menjelaskan mengapa setiap metode pemodelan menghasilkan tingkat performa yang berbeda. Sifat data ini tidak hanya menjadi tantangan, tetapi juga memberikan wawasan yang dapat digeneralisasi untuk penelitian pada konteks serupa. Terdapat tiga karakteristik utama yang memengaruhi hasil, yaitu ketidakseimbangan label yang ekstrem, sifat teks yang naratif dan bervariasi, serta struktur multilabel.

Ketidakseimbangan label adalah faktor paling dominan yang menyebabkan rendahnya performa model pada data asli. Dengan adanya emosi mayoritas seperti happy dan satisfied yang jumlahnya ratusan kali lipat dibanding emosi minoritas seperti jealous, model machine learning sederhana seperti Naïve Bayes dan Logistic Regression cenderung mengambil jalan pintas dengan menjadi bias. Model-model ini gagal mempelajari pola dari kelas minoritas, yang secara langsung tecermin pada nilai Macro F1-Score yang sangat rendah. Metrik ini sangat sensitif terhadap performa pada kelas minoritas, sehingga menjadi indikator kegagalan model dalam menangani data yang tidak seimbang.

Kedua, sifat teks yang naratif dengan panjang yang bervariasi (3 hingga 111 kata) memberikan keuntungan bagi model yang mampu memahami konteks, seperti arsitektur Transformer pada deep learning. Inilah mengapa, bahkan pada data asli, model deep learning mampu mengidentifikasi variasi emosi yang lebih beragam dibandingkan model machine learning.

Ketiga, struktur multilabel dari data menjelaskan performa yang tercermin pada metrik seperti Subset Accuracy (SA) dan Jaccard Accuracy (JA). Metrik SA, yang menuntut prediksi kombinasi label yang persis benar, secara konsisten menghasilkan skor yang rendah karena sifatnya yang sangat ketat. Oleh karena itu, Jaccard Accuracy (JA) menjadi metrik yang lebih sesuai dan informatif dalam konteks ini. JA tidak menghukum model secara penuh untuk kesalahan kecil, melainkan mengukur tumpang tindih antara label prediksi dan label sebenarnya, sehingga memberikan gambaran yang lebih adil tentang kemampuan model dalam mengidentifikasi sebagian besar emosi yang relevan.

Wawasan dari analisis ini dapat digeneralisasi untuk penelitian di masa depan yang menggunakan data dengan karakteristik serupa, seperti ulasan produk naratif, catatan klinis, maupun esai pribadi yang bersifat multilabel. Temuan penelitian ini juga memberikan beberapa catatan penting. Pertama, penanganan ketidakseimbangan data merupakan langkah krusial yang tidak dapat diabaikan, di mana teknik seperti oversampling terbukti cukup efektif. Kedua, pemilihan model sebaiknya mempertimbangkan kompleksitas teks yang dianalisis. Dengan demikian, penelitian selanjutnya disarankan untuk memprioritaskan strategi penyeimbangan data serta menggunakan metrik evaluasi yang lebih komprehensif, seperti Macro F1-Score dan Jaccard Accuracy, guna memperoleh hasil yang lebih valid dan akurat.

4.6.2 Pembahasan Model Machine Learning

Evaluasi pada kelompok model machine learning (Naïve Bayes, Logistic Regression, SVM, dan RFC) menunjukkan adanya perbedaan performa yang jelas antara penggunaan data asli dan data hasil oversampling. Pada data asli, seluruh model menunjukkan performa yang buruk pada data uji. Naïve Bayes dan Logistic Regression mencatatkan Jaccard Accuracy (JA) yang sangat rendah, yaitu 0.1934 dan 0.2202, serta Macro F1-Score yang hampir nol (0.05 dan 0.07). Hal ini menunjukkan bahwa kedua model tersebut sangat bias dan cenderung hanya memprediksi label-label mayoritas seperti "happy" dan "satisfied". Di antara model lainnya, SVM menunjukkan performa paling unggul pada data asli dengan JA data uji 0.2744 dan Macro F1-Score 0.16. SVM juga terbukti mampu memprediksi variasi label yang lebih luas, termasuk "*calm*", "*frustrated*", dan "*proud*".

Setelah penerapan teknik oversampling, terjadi peningkatan performa yang sangat signifikan pada semua model. Macro F1-Score pada Naïve Bayes melonjak menjadi 0.50 dan pada Logistic Regression menjadi 0.58. Peningkatan paling impresif kembali ditunjukkan oleh SVM, yang mencapai JA data uji sebesar 0.4561 dan Macro F1-Score 0.64. Angka ini menjadikan SVM sebagai model machine learning dengan performa terbaik secara keseluruhan. Meskipun demikian, model SVM masih menunjukkan kesulitan dalam memprediksi teks yang sangat pendek atau sangat panjang.

Dari sisi waktu inferensi, terdapat perbedaan kecepatan yang kontras. Random Forest Classifier (RFC) secara konsisten menjadi yang paling lambat (di atas 100 ms). Sebaliknya, SVM dan Naïve Bayes tergolong cepat (4-12 ms), sementara Logistic Regression menjadi yang tercepat (di bawah 5 ms). Kecepatan ini murni dipengaruhi oleh kompleksitas arsitektur masing-masing model, bukan oleh jenis data yang digunakan.

4.6.3 Pembahasan Model Deep Learning

Pada kategori model deep learning (BERT, BERT-LSTM, DistilBERT, dan RoBERTa), hasil model yang telah dievaluasi menggunakan data uji 15% menunjukkan kemampuan yang kuat dalam menangani tugas klasifikasi multi-emosi, terutama setelah masalah ketidakseimbangan data diatasi. Saat menggunakan data asli, semua model deep learning menunjukkan kemampuan dasar dalam prediksi emosi. Hal ini terlihat dari Label-wise Accuracy (LA) yang sudah baik (sekitar 0.90), namun Subset Accuracy (SA) masih relatif rendah (tertinggi 0.11), menandakan kesulitan dalam memprediksi seluruh kombinasi label secara tepat. Meskipun Macro F1-Score masih terbilang rendah (0.17-0.20), model-model ini

sudah mampu memprediksi variasi label emosi yang lebih beragam dibandingkan model machine learning pada data asli, seperti "anxious", "calm", "excited", "frustrated", "happy", "proud", dan "satisfied".

Penerapan data hasil oversampling membawa peningkatan performa yang sangat jelas dan konsisten di seluruh model. BERT-LSTM mengalami lonjakan paling signifikan, dengan JA meningkat menjadi 0.50 dan SA menjadi 0.32 pada data uji. Seluruh model deep learning mencapai Macro F1-Score yang sangat baik dan setara di rentang 0.67-0.69. Oversampling juga berhasil membuat model mampu memprediksi label minoritas yang sebelumnya sulit terdeteksi, seperti "nostalgic", "bored", "jealous", dan "disgusted". Dalam hal ini, RoBERTa menunjukkan keunggulan dalam pemerataan prediksi antar label. Selain itu, RoBERTa juga memiliki nilai JA tertinggi dengan nilai 0.50 serta nilai Macro F1-Score tertinggi baik pada data asli yaitu 0.20 dan juga pada data hasil oversampling yaitu 0.63. Sebuah temuan unik adalah pada DistilBERT, yang justru kehilangan kemampuan memprediksi label "proud" setelah proses oversampling.

Dari segi waktu inferensi, hampir semua model deep learning menunjukkan kinerja yang relatif serupa. Model BERT, BERT-LSTM, dan RoBERTa membutuhkan waktu sekitar 25 hingga 35 ms untuk melakukan prediksi pada data dalam skenario pengujian. Sebaliknya, DistilBERT secara konsisten tampil sebagai model tercepat dengan waktu inferensi rata-rata 15 ms, atau hampir dua kali lebih cepat dibandingkan model deep learning lainnya. Keunggulan ini tidak hanya disebabkan oleh jumlah parameter DistilBERT yang lebih ringkas, tetapi juga oleh efisiensinya dalam melakukan prediksi label emosi.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini telah berhasil mengeksplorasi dan membandingkan performa berbagai model machine learning dan deep learning dalam tugas deteksi emosi multilabel berbasis teks. Berdasarkan hasil penelitian yang telah dilakukan, diperoleh kesimpulan yaitu:

a. Berdasarkan hasil penelitian, terdapat perbedaan karakteristik pada model-model dalam lingkupnya masing-masing yang dipengaruhi oleh sifat data yang digunakan. Dataset yang naratif dengan panjang teks bervariasi serta struktur multilabel berbasis emosi yang beragam menjadi faktor utama yang memengaruhi performa setiap model. Perlu diketahui bahwa, karakteristik ini juga dapat digeneralisasi pada jenis data serupa seperti ulasan produk, catatan klinis, maupun esai pribadi yang memiliki sifat multilabel dan distribusi label tidak seimbang. Pada karakteristik untuk masing-masing model machine learning adalah sebagai berikut:
    1. Naïve bayes memiliki performa yang sangat rendah pada dataset asli dan hanya mampu melakukan prediksi kepada label mayor seperti happy dan satisfied, setelah menggunakan data hasil oversampling, terjadi peningkatan performa yang membuat model tersebut mampu memprediksi emosi yang sebelumnya gagal terdeteksi dan menunjukkan distribusi prediksi lebih merata namun masih sering melakukan prediksi pada label mayor dan memiliki waktu inferensi yang cepat sekitar 4 hingga 12 ms.
    2. Logistic Regression memiliki performa dan karakteristik yang menyerupai Naïve bayes namun, memiliki beberapa peningkatan dari segi akurasi dan juga metriks evaluasi. Model logistic regression memiliki karakteristik waktu inferensi yang tercepat diantara model machine learning lainnya yaitu dibawah 5ms pada keseluruhan hasil pengujian
    3. Support Vector Machine (SVM) memiliki performa pada data asli yang sedikit lebih baik dari model machine learning lainnya karena mampu memprediksi lebih banyak label emosi seperti *calm*, *frustrated*, dan *proud*. Setelah menggunakan data hasil oversampling, SVM bahkan mampu mendekati kinerja model deep learning dan tetap memiliki waktu inferensi yang cepat sekitar 4 hingga 12 ms.
    4. Random Forest Classifier (RFC) pada penelitian ini terjadi overfitting pada model ini, yang mengakibatkan model melakukan prediksi yang sempurna pada data latih namun

mengalami penurunan performa yang sangat jauh pada data uji, hal tersebut juga berlaku baik pada data asli maupun data hasil oversampling. Dari segi waktu inferensi, model ini memiliki waktu inferensi terlama yaitu sekitar 100ms.

Pada karakteristik untuk masing-masing model deep learning adalah sebagai berikut:

1. BERT sebagai model dasar deep learning pada penelitian ini masih menunjukkan performa yang belum optimal pada data asli. Meskipun demikian, model ini sudah mampu menangkap dan memprediksi beberapa emosi, terutama pada label mayor seperti happy dan satisfied. Setelah dilakukan oversampling, performanya meningkat signifikan dengan akurasi dan metrik evaluasi yang lebih baik, serta kemampuan memprediksi label minor yang sebelumnya gagal terdeteksi. Dari segi efisiensi, waktu inferensinya berada pada kisaran menengah yaitu 28 hingga 32 ms.
2. BERT-LSTM, pada data asli memiliki performa yang menyerupai model dasar nya yaitu BERT. Setelah menggunakan data hasil oversampling, memiliki peningkatan yang tinggi pada akurasi dan unggul dalam menangani konteks kompleks dan teks dengan banyak label. Dari segi waktu inferensi, model ini memiliki waktu inferensi yang sedikit lebih lama dari model BERT yaitu sekitar 28-36 ms.
3. RoBERTa memiliki keunggulan pada kemampuannya dalam memprediksi label-label minor. Pada data asli, model ini sudah dapat meyeimbangkan prediksi pada label-label emosi minor daripada model lainnya, dan performa itu terus meningkat setelah menggunakan data hasil oversampling. Waktu inferensi yang dimiliki model ini sekitar 25-30 ms, yang dimana lebih cepat dari model dasarnya yaitu BERT.
4. DistilBERT memiliki karakteristik utama sebagai model deep learning paling efisien karena memiliki jumlah parameter paling sedikit diantara model deep learning lainnya. Pada data asli, model ini dapat menyeimbangkan prediksi pada label-label emosi minor. Setelah menggunakan data hasil oversampling, kenaikan prediksi label-label minor pada model DistilBERT tidak sesignifikan seperti RoBERTa meskipun masih lebih unggul dari model dasarnya yaitu BERT. Waktu inferensi model ini adalah yang tercepat diantara model deep learning lain yaitu sekitar 15ms.

b. Berdasarkan hasil evaluasi, skenario uji, dan juga karakteristik dari semua model, sehingga didapatkan bahwa model yang paling sesuai untuk melakukan prediksi teks emosi multilabel ini pada model machine learning adalah SVM dan pada model deep learning adalah RoBERTa. SVM unggul karena memiliki akurasi tertinggi di antara model machine learning lainnya bahkan setelah menggunakan data hasil oversampling, sambil

tetap mempertahankan waktu inferensi yang sangat cepat dan efisien. RoBERTa menjadi model terbaik karena kemampuannya yang unggul dalam pemerataan prediksi emosi, menjadikannya model deep learning yang paling andal untuk mendeteksi spektrum emosi secara merata, termasuk label-label minoritas yang paling sulit dikenali.

5.2 Saran

Setelah melakukan penelitian mengenai analisis model machine learning dan deep learning pada deteksi emosi multilabel pada teks tersebut, diharapkan penelitan selanjutnya dapat diterapkan beberapa saran berikut:

a. Dapat menggunakan dataset yang lebih seimbang antara masing-masing labelnya sehingga hasil prediksi model dapat lebih akurat dan dalam prediksinya dapat memunculkan label emosi yang lebih beragam.
b. Model-model yang telah dianalisis diharapkan dapat diterapkan pada aplikasi dunia nyata seperti mobile ataupun web. Model-model tersebut diharapkan dapat sebagai alat bantu untuk memprediksi emosi terkait kesehatan mental maupun analisis lebih mendalam pada emosi di dalam teks.

DAFTAR PUSTAKA

Ahanin, Z., Ismail, M. A., Singh, N. S. S., & AL-Ashmori, A. (2023). Hybrid Feature Extraction for Multi-Label Emotion Classification in English Text Messages. *Sustainability (Switzerland)*, *15*(16). https://doi.org/10.3390/su151612539

Ameer, I., Bölücü, N., Siddiqui, M. H. F., Can, B., Sidorov, G., & Gelbukh, A. (2023). Multi-label emotion classification in texts using transfer learning. *Expert Systems with Applications*, *213*. https://doi.org/10.1016/j.eswa.2022.118534

Anantharaman, K., Rajalakshmi, S., Angel Deborah, S., Saritha, M., & Sakaya Milton, R. (2022). *SSN_MLRG1@LT-EDI-ACL2022: Multi-Class Classification using BERT models for Detecting Depression Signs from Social Media Text*.

Ankit Chauhan. (2021, February 24). Random Forest Classifier and its Hyperparameters.

Anshul. (2025, April 25). Logistic Regression in Machine Learning.

Bashynska, I., Sarafanov, M., & Manikaeva, O. (2024). Research and Development of a Modern Deep Learning Model for Emotional Analysis Management of Text Data. *Applied Sciences (Switzerland)*, *14*(5). https://doi.org/10.3390/app14051952

Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *The Journal of Machine Learning Research*, *13*, 281–305.

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Cambria, E., Livingstone, A., & Hussain, A. (n.d.). *LNCS 7403 - The Hourglass of Emotions*. Retrieved from http://sentic.net

Chakkarwar, V., Tamane, S., & Thombre, A. (2023). *A Review on BERT and Its Implementation in Various NLP Tasks*. https://doi.org/10.2991/978-94-6463-136-4_12

Coppini, S., Lucifora, C., Vicario, C. M., & Gangemi, A. (2023). Experiments on real-life emotions challenge Ekman's model. *Scientific Reports*, *13*(1), 9511. https://doi.org/10.1038/s41598-023-36201-5

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*.

Goutte, C., & Gaussier, E. (2005). *A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation*. https://doi.org/10.1007/978-3-540-31865-1_25

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. IEEE. https://doi.org/10.1109/CVPR.2016.90

Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied Logistic Regression*. Wiley. https://doi.org/10.1002/9781118548387

Hripcsak, G., & Albers, D. J. (2022). Evaluating Prediction of Continuous Clinical Values: A Glucose Case Study. *Methods of Information in Medicine*, *61*(S 01), e35–e44. https://doi.org/10.1055/s-0042-1743170

Kim, K. G. (2016). Book Review: Deep Learning. *Healthcare Informatics Research*, *22*(4), 351. https://doi.org/10.4258/hir.2016.22.4.351

Li, L. (2022). Emotion Analysis Method of Teaching Evaluation Texts Based on Deep Learning in Big Data Environment. *Computational Intelligence and Neuroscience*, *2022*, 1–8. https://doi.org/10.1155/2022/9909209

Li, X. A., & Parikh, D. (2019). *Lemotif: An Affective Visual Journal Using Deep Neural Networks*. Retrieved from http://arxiv.org/abs/1903.07766

Liu, J., Chang, W. C., Wu, Y., & Yang, Y. (2017). Deep learning for extreme multi-label text classification. *SIGIR 2017 - Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 115–124. Association for Computing Machinery, Inc. https://doi.org/10.1145/3077136.3080834

Liu, W., Pang, J., Li, N., Zhou, X., & Yue, F. (2021). Research on Multi-label Text Classification Method Based on tALBERT-CNN. *International Journal of Computational Intelligence Systems*, *14*(1). https://doi.org/10.1007/s44196-021-00055-4

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., … Stoyanov, V. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*.

Malhotra, M. (2021). Journal Entries with Labelled Emotions. Retrieved April 29, 2025, from Kaggle website: https://www.kaggle.com/datasets/madhavmalhotra/journal-entries-with-labelled-emotions/

Muraina, I. (2022). *IDEAL DATASET SPLITTING RATIOS IN MACHINE LEARNING ALGORITHMS: GENERAL CONCERNS FOR DATA SCIENTISTS AND DATA ANALYSTS*.

Nadas, M., Diosan, L., & Tomescu, A. (2025). *Synthetic Data Generation Using Large Language Models: Advances in Text and Code*. Retrieved from http://arxiv.org/abs/2503.14023

Nakada, R., Xu, Y., Li, L., & Zhang, L. (2025). *Synthetic Oversampling: Theory and A Practical Approach Using LLMs to Address Data Imbalance*. Retrieved from http://arxiv.org/abs/2406.03628

Neri Van Otten. (2024, May 6). Support Vector Machines (SVM) In Machine Learning Made Simple & How To Tutorial.

Opitz, J., & Burst, S. (2021). *Macro F1 and Macro F1*.

Parsanna, M. L. (2025). A Review on Various Machine Learning Algorithms. *INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, *09*(03), 1–9. https://doi.org/10.55041/IJSREM42692

Rahman, A. B. S., Ta, H. T., Najjar, L., Azadmanesh, A., & Gönul, A. S. (2024). DepressionEmo: A novel dataset for multilabel classification of depression emotions. *Journal of Affective Disorders*, *366*, 445–458. https://doi.org/10.1016/j.jad.2024.08.013

Rish, I. (2001). An Empirical Study of the Naïve Bayes Classifier. *IJCAI 2001 Work Empir Methods Artif Intell*, *3*.

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2020). *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*.

Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Computer Science*, *2*(6), 420. https://doi.org/10.1007/s42979-021-00815-1

Seyeditabari, A., Tabari, N., & Zadrozny, W. (2018). *Emotion Detection in Text: a Review*.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

Song, Y., Zhiyue, L., Wang, J., Liang, Z., Liu, Z., & Jiang, T. (n.d.). *Utilizing BERT Intermediate Layers for Aspect Based Sentiment Analysis and Natural Language Inference*. https://doi.org/10.48550/arXiv.2002.04815

Sowmya B J, Chetan, & Srinivasa, K. G. (2016). Large scale multi-label text classification of a hierarchical dataset using Rocchio algorithm. *2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, 291–296. IEEE. https://doi.org/10.1109/CSITSS.2016.7779373

Tan, K., Lee, C.-P., Anbananthen, K., & Lim, K. (2022). RoBERTa-LSTM: A Hybrid Model for Sentiment Analysis with Transformers and Recurrent Neural Network. *IEEE Access*, *10*, 1. https://doi.org/10.1109/ACCESS.2022.3152828

Tharwat, A. (2021). Classification assessment methods. *Applied Computing and Informatics*, *17*(1), 168–192. https://doi.org/10.1016/j.aci.2018.08.003

Tsoumakas, G., & Katakis, I. (2009). Multi-Label Classification: An Overview. *International Journal of Data Warehousing and Mining*, *3*, 1–13. https://doi.org/10.4018/jdwm.2007070101

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2023). *Attention Is All You Need*.

Wibowo, P., & Fatichah, C. (2021). An in-depth performance analysis of the oversampling techniques for high-class imbalanced dataset. *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, *7*(1), 63. https://doi.org/10.26594/register.v7i1.2206

Zhang, H. (2004). The Optimality of Naive Bayes. In *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2004* (Vol. 2).

Zhang, M.-L., & Zhou, Z.-H. (2014). A Review on Multi-Label Learning Algorithms. *IEEE Transactions on Knowledge and Data Engineering*, *26*(8), 1819–1837. https://doi.org/10.1109/TKDE.2013.39

LAMPIRAN

Dataset asli “Journal Entries with labelled emotions”
https://drive.google.com/file/d/1j-zoAADtAXjuKGO-WPJkWOBHFZDquTQz/view?usp=sharing

Dataset hasil oversampling
https://drive.google.com/file/d/1SK4eOuHpG7GkNXNMIctbmSAXhTV-X4C3/view?usp=sharing

Model Naive Bayes menggunakan data asli
https://colab.research.google.com/drive/1W7bdMW420OlZpnKM-nfeffBvFalw3UZ5?usp=sharing

Model Naive Bayes menggunakan data hasil oversampling
https://colab.research.google.com/drive/1F65QV1zGzuIz3mdV94gPdyUwbHYgqlnY?usp=sharing

Model Logistic Regression menggunakan data asli
https://colab.research.google.com/drive/1agGeFq0heoqRb9NjYbBxG3NWPuTFqpQw?usp=sharing

Model Logistic Regression menggunakan data hasil oversampling
https://colab.research.google.com/drive/1ZVwDeBkV6LmfAaW-xy9lONWfH0f_UN18?usp=sharing

Model Support Vector Machine menggunakan data asli
https://colab.research.google.com/drive/1dyvopOrda4-qpYWjspQZ3c5TNxPbEXJu?usp=sharing

Model Support Vector Machine menggunakan data hasil oversampling
https://colab.research.google.com/drive/1sItnlBVyP8ejwIKvAG5UEt5fYQy6f51X?usp=sharing

Model Random Forest Classifier menggunakan data asli
https://colab.research.google.com/drive/1x_8TwIOxJvNy9-LxrPpoSJXa-QAbM2wT?usp=sharing

Model Random Forest Classifier menggunakan data hasil oversampling
https://colab.research.google.com/drive/1re6qiP9ajJCK5jpl52fBPjRkl1k2PjxU?usp=sharing

Model BERT menggunakan data asli
https://colab.research.google.com/drive/1MJWE4SIdMJelGN0oZFM50ZV34iAuQPFY?usp=sharing

Model BERT menggunakan data hasil oversampling
https://colab.research.google.com/drive/1wUCzN-5CC6zZBpCzEDArwLKzOa9ME-SC?usp=sharing

Model BERT-LSTM menggunakan data asli
https://colab.research.google.com/drive/1BqCy_o4ChCQ1pOm5LrDZZQkSkLQq0YJQ?usp=sharing

Model BERT-LSTM menggunakan data hasil oversampling
https://colab.research.google.com/drive/1Nphq4hsq1PFPuYAGYDsH4x_-vd4y_n8J?usp=sharing

Model RoBERTa menggunakan data asli
https://colab.research.google.com/drive/1l_rHbfMC3YbPxVeWoucFw8PwYeWBTJmF?usp=sharing

Model RoBERTa menggunakan data hasil oversampling
https://colab.research.google.com/drive/1iW7_mkVrcNxiWENxKkkjGveAog_gf6OQ?usp=sharing

Model DistilBERT menggunakan data asli

https://colab.research.google.com/drive/1TaL05uOcisFG-L4Im9lhsyABKsr8p2g2?usp=sharing

Model DistilBERT menggunakan data hasil oversampling
https://colab.research.google.com/drive/1u0Or5oHShfeC8KfecvGih7Z-8XBn4uMZ?usp=sharing