BAB II
KAJIAN LITERATUR

2.1 Royalti Musik

Royalti musik merupakan kompensasi finansial yang wajib diberikan kepada pencipta lagu, pemegang hak cipta, atau pemilik hak terkait sebagai imbalan atas pemanfaatan hak ekonomi dari suatu karya musik yang digunakan secara komersial [20] [21]. Definisi ini secara yuridis diperkuat dalam Pasal 1 angka 21 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta. Pasal tersebut menyatakan bahwa royalti adalah imbalan atas pemanfaatan hak ekonomi suatu ciptaan atau produk hak terkait yang diterima oleh pencipta atau pemilik hak terkait [21].

Dalam praktiknya, sistem royalti musik terbagi ke dalam beberapa jenis, antara lain Royalti Hak Pertunjukan yang timbul ketika musik diputar di ruang publik seperti konser, kafe, atau hotel, Royalti Mekanik yang terkait dengan reproduksi karya dalam format fisik atau digital, serta Royalti Sinkronisasi untuk penggunaan musik dalam media visual seperti film dan iklan [21] [22] [23]. Di Indonesia, pengelolaan dan pendistribusian royalti ini dipercayakan kepada Lembaga Manajemen Kolektif Nasional (LMKN) yang bertugas menghimpun royalti dari para pengguna musik komersial, seperti pelaku industri perhotelan dan hiburan, untuk kemudian disalurkan kepada pencipta dan pemegang hak terkait melalui Lembaga Manajemen Kolektif (LMK) [20] [22].

2.1.1 Kebijakan Royalti Musik

Kebijakan royalti musik merupakan bagian dari sistem perlindungan hak cipta yang bertujuan menjamin hak ekonomi pencipta dan pemegang hak terkait atas pemanfaatan karya musik dalam aktivitas komersial [24]. Royalti memiliki peran yang besar dalam keberlangsungan hidup industri musik, terutama di tengah pergeseran konsumsi musik dari media fisik ke platform digital. Studi terkini menunjukkan bahwa kebijakan hak cipta dan royalti memiliki implikasi langsung terhadap dinamika industri kreatif serta hubungan antara pencipta dan pengguna karya [25]. Berdasarkan royalti musik yang diatur dalam Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, telah ditetapkan bahwa pencipta memiliki hak ekonomi dan hak moral atas karya mereka, termasuk hak untuk memperoleh royalti atas penggunaan karya mereka oleh pihak lain, sebagaimana tercantum dalam Pasal 8 dan Pasal 9 terkait hak eksklusif pencipta [26]. Implementasi kebijakan royalti musik ini dikelola

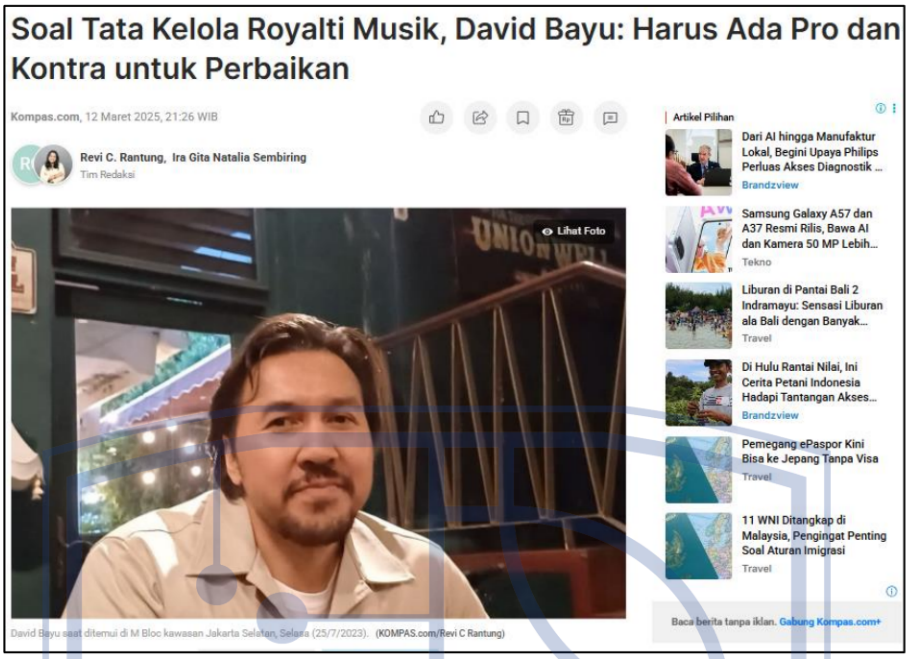
secara menyeluruh oleh Lembaga Manajemen Kolektif Nasional (LMKN) berdasarkan Sistem Informasi Lagu dan Musik (SILM) dan informasi yang disediakan di pusat data. Setiap lagu yang ingin dipergunakan secara komersial akan diperbolehkan setelah pengguna mengajukan izin kepada pencipta atau pemegang hak cipta melalui LMKN [4].

Implementasi Peraturan Pemerintah (PP) 56/2021 secara khusus ditujukan kepada sektor usaha layanan publik yang bersifat komersial, termasuk kafe dan restoran. Dalam pelaksanaan peraturan ini, kewenangan untuk menarik royalti oleh LMKN tetap berlaku, meskipun pencipta lagu tidak terdaftar dalam Lembaga Manajemen Kolektif (LMK) tertentu. Namun, pada praktiknya, penarikan royalti ini menghadapi kendala besar berupa rendahnya literasi hukum di kalangan pemilik usaha. Banyak pelaku usaha kafe yang belum mengetahui adanya kewajiban membayar royalti atas pemutaran lagu secara komersial, atau menganggap bahwa penggunaan platform *streaming personal* sudah mencakup izin publik [27]. Ketidaktahuan ini menyebabkan asimetri antara regulasi yang ketat dalam Pasal 12 PP 56/2021 dengan kepatuhan di lapangan.

2.1.2 Pro dan Kontra Royalti Musik

Implementasi kebijakan royalti musik di Indonesia, khususnya terkait kewajiban pembayaran oleh pelaku usaha, senantiasa memicu pro dan kontra di ruang publik. Dari sisi yang mendukung, pihak penyanyi berpendapat bahwa polemik yang muncul justru merupakan dinamika yang sehat dan diperlukan untuk mendorong perbaikan sistem tata kelola royalti agar lebih baik, sekaligus meningkatkan kesadaran masyarakat dan para pemangku kepentingan tentang nilai berharga dari sebuah karya cipta [28]. Ia menambahkan bahwa selama ini para musisi dan pencipta lagu merasa hasil yang mereka dapatkan dari sistem yang berjalan belum maksimal [28].

Di sisi lain, kontra datang dari kalangan pelaku usaha seperti pemilik kafe, restoran, dan hotel yang kerap keberatan dengan penarikan royalti. Keberatan ini muncul bukan semata karena tidak adanya dasar hukum, melainkan lebih disebabkan oleh rendahnya pemahaman dan kesadaran hukum di kalangan pengusaha terkait kewajiban tersebut, sehingga mereka menganggap penarikan royalti oleh Lembaga Manajemen Kolektif Nasional (LMKN) sebagai tindakan "memungut" tanpa dasar yang jelas [22]. Polemik ini diperparah oleh persoalan transparansi dalam penyaluran dana dan besaran royalti yang diterima pencipta, saat sebagian musisi mengeluhkan nominal royalti yang kecil karena laporan penggunaan musik dari para pengguna komersial yang belum optimal [20].



Gambar 2.1 Pernyataan David Bayu Mengenai Tata Kelola Royalti Musik

Seperti yang ditampilkan pada Gambar 2.1, pernyataan David Bayu mencerminkan pandangan dari kalangan musisi yang mendukung adanya dinamika dalam pengelolaan royalti musik. Ia menekankan bahwa pro dan kontra merupakan bagian alami dari proses menuju sistem yang lebih baik dan berkeadilan bagi seluruh pemangku kepentingan, baik pencipta lagu maupun pengguna karya musik secara komersial [20].

2.2 Platform X

Twitter atau yang dikenal sekarang dengan nama X merupakan media sosial yang sering digunakan masyarakat untuk mengekspresikan pandangannya terhadap setiap kejadian [29]. Pengguna X dapat langsung membagikan pendapatnya dengan mengirimkan tweet pada platform X. Secara keseluruhan, platform X memiliki sekitar 259 juta pengguna aktif harian, dan 500 juta tweet yang dikirim per harinya. Hal yang menarik dari platform ini terdapat pada fitur bernama trending topik. Fitur ini memudahkan pengguna untuk mengetahui topik tertentu yang sedang populer [30], dengan menganalisis kata-kata yang sering digunakan pada tweet.

Cara kerja X sebagai sebuah platform dibangun di atas arsitektur real-time yang memungkinkan unggahan, pesan, dan interaksi lainnya (seperti suka dan retweet) didistribusikan secara instan ke seluruh jaringan pengguna. Setiap tindakan yang dilakukan pengguna, seperti memposting atau mencari konten, berkomunikasi dengan server melalui serangkaian Application Programming Interface (API) [31]. API inilah yang menjadi

fondasi teknis yang memungkinkan pengembang dan peneliti untuk berinteraksi dengan data X secara terprogram, di luar antarmuka pengguna grafis biasa.

Dalam studi literatur mengenai media digital, kebebasan berekspresi di platform X menjadikannya sebagai representasi dari *vox populi* atau suara masyarakat di ruang siber [32]. Oleh karena itu, percakapan yang terekam di platform ini dapat dikategorikan sebagai data tekstual yang kaya akan sentimen, yang mencerminkan respons, kritik, maupun dukungan publik terhadap kebijakan tertentu secara organik. Hal ini secara khusus banyak dimanfaatkan dalam penelitian opini publik karena karakteristiknya yang berbasis teks singkat, cepat menyebar, dan sering digunakan untuk merespons isu kebijakan yang sedang berkembang [33].

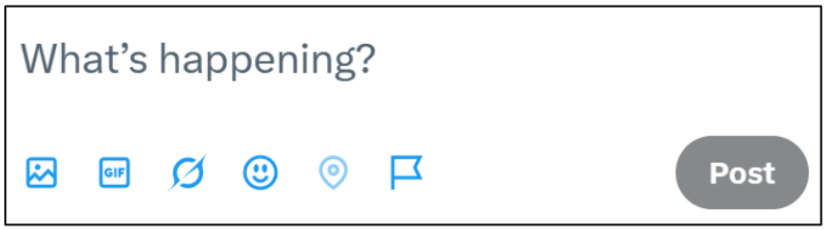
Sebagai platform media sosial yang terus berkembang, X menyediakan beragam fitur yang mendukung interaksi pengguna dan penyebaran informasi secara real-time. Fitur-fitur ini tidak hanya memfasilitasi komunikasi antar pengguna, tetapi juga menjadi sarana bagi publik untuk mengekspresikan pendapatnya terhadap berbagai isu, termasuk kebijakan royalti musik. Berikut adalah penjelasan mengenai fitur-fitur utama yang tersedia pada platform X beserta peran masing-masing dalam konteks penelitian ini.

1. Unggahan Teks, Gambar, dan Video

   Fitur unggahan pada platform X memungkinkan pengguna untuk membagikan konten dalam berbagai format, yaitu teks, gambar, dan video. Ketiga jenis unggahan ini memiliki karakteristik yang berbeda dalam menyampaikan informasi dan opini publik. Unggahan teks merupakan bentuk paling dasar yang berisi pesan tertulis dari pengguna. Unggahan gambar dan video berfungsi sebagai pelengkap yang dapat memperkuat atau memperjelas pesan yang ingin disampaikan, meskipun dalam penelitian analisis sentimen, fokus utama tetap pada teks yang menyertainya. Secara kolektif, ketiga jenis unggahan ini disebut sebagai fitur utama platform X karena menjadi fondasi dari seluruh aktivitas komunikasi pengguna di platform tersebut.

   Fitur utama platform X adalah kemampuan untuk mengirimkan unggahan yang dapat berisi berbagai jenis konten. Pengguna dapat menulis pesan berbasis teks dengan batasan karakter tertentu, serta melampirkan media pendukung seperti gambar dan video [29]. Unggahan ini menjadi sumber data utama dalam penelitian analisis sentimen karena mencerminkan opini publik secara langsung. Selain itu, pengguna juga dapat berinteraksi dengan unggahan orang lain melalui fitur suka (*like*), bagikan ulang

(*retweet* atau *repost*), dan balasan (*reply*), yang memperkaya dinamika percakapan di platform [29].



Gambar 2.2 Contoh Unggahan Teks pada Platform X

Pada Gambar 2.2 menampilkan contoh unggahan teks pada platform X yang menunjukkan fitur utama untuk menulis pesan. Pengguna dapat mengetikkan teks pada kolom "What's happening?" serta melampirkan media pendukung seperti gambar, video, atau GIF. Selain itu, terdapat pula tombol interaksi seperti tombol untuk mengirim unggahan (Post).

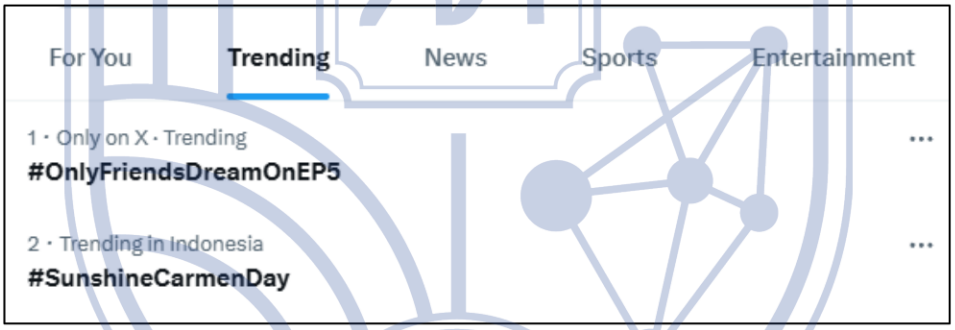
Gambar 2.3 Contoh Unggahan Teks pada Platform X

Pada Gambar 2.3 menampilkan contoh unggahan yang telah dipublikasikan oleh pengguna. Unggahan ini menampilkan teks dan gambar dengan judul "Lake Como,

Italy" beserta informasi jumlah balasan (3), repost (18), suka (88), dan tampilan (2K). Fitur interaksi ini memungkinkan pengguna lain untuk memberikan respons terhadap unggahan tersebut, yang menjadi indikator penting dalam analisis sentimen karena mencerminkan tingkat keterlibatan dan reaksi publik terhadap suatu isu [29].

2. Fitur Trending Topic

Salah satu fitur yang paling menonjol di platform X adalah trending topic atau topik yang sedang populer. Fitur ini menampilkan kata kunci, frasa, atau tagar yang paling banyak dibicarakan oleh pengguna dalam periode waktu tertentu di suatu lokasi geografis [30]. Mekanisme kerja fitur ini didasarkan pada analisis algoritmik terhadap peningkatan volume percakapan secara tiba-tiba, sehingga mampu menangkap isu-isu yang sedang hangat diperbincangkan masyarakat. Dalam konteks penelitian ini, *trending topic* berperan penting dalam mengidentifikasi momentum-momentum kritis terkait sengketa royalti musik, seperti saat pemberitaan mengenai rendahnya royalti yang diterima musisi atau kebijakan baru terkait pembebasan royalti bagi Usaha Mikro, Kecil, dan Menengah (UMKM).

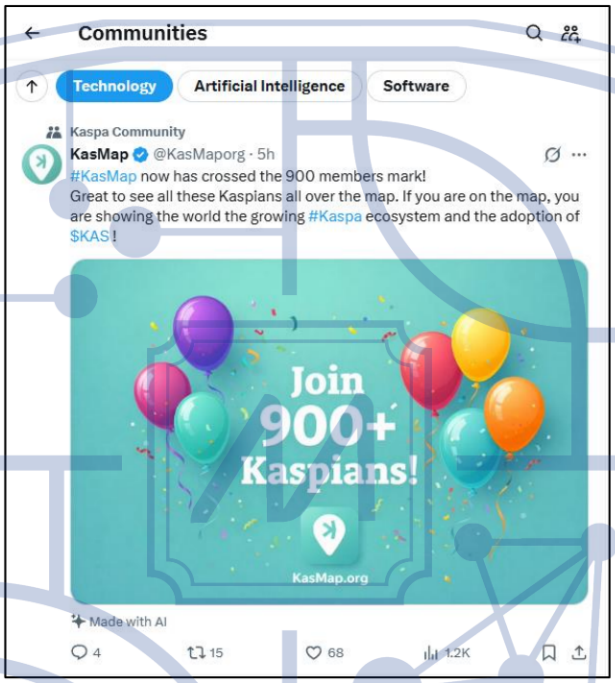


Gambar 2.4 Tampilan Fitur Trending Topic pada Platform X

Pada Gambar 2.4 menampilkan tampilan fitur *trending topic* pada platform X yang menunjukkan berbagai topik yang sedang populer. Fitur ini menyediakan beberapa kategori seperti "For You", "Trending", "News", "Sports", dan "Entertainment" untuk memudahkan pengguna dalam menyaring topik sesuai minat mereka [30]. Setiap topik yang masuk dalam daftar *trending* umumnya disertai dengan indikator seperti nomor urut popularitas, label kategori (misalnya "Only on X" atau "Trending in Indonesia"), serta nama topik atau tagar yang sedang ramai diperbincangkan.

3. Fitur Komunitas (X Communities)

X juga menyediakan fitur Communities yang memungkinkan pengguna dengan minat atau pandangan serupa untuk berdiskusi dalam ruang yang lebih terfokus [32]. Fitur ini memfasilitasi terbentuknya kelompok diskusi tematik, termasuk kelompok yang membahas isu-isu hukum, kebijakan publik, atau industri kreatif. Percakapan di dalam komunitas seringkali lebih mendalam dan terarah, sehingga dapat menjadi sumber data kualitatif yang berharga untuk memahami nuansa opini publik terhadap suatu kebijakan [32].



Gambar 2.5 Tampilan Fitur Komunitas (X Communities)

Pada Gambar 2.5 menampilkan tampilan fitur Communities pada platform X yang menyediakan ruang diskusi tematik bagi pengguna dengan minat yang sama. Fitur ini menampilkan berbagai kategori komunitas seperti "Technology", "Artificial Intelligence", dan "Software" yang dapat dipilih oleh pengguna sesuai minatnya. Setiap komunitas memiliki halaman tersendiri yang menampilkan unggahan dari anggota komunitas tersebut, sehingga diskusi dapat berlangsung secara lebih terfokus dibandingkan dengan linimasa utama [32].

4. Fitur Kecerdasan Buatan: Grok

Sebagai inovasi terbaru, X mengintegrasikan asisten kecerdasan buatan bernama Grok yang tersedia bagi pengguna dengan langganan tertentu. Grok dirancang untuk memberikan jawaban atas pertanyaan pengguna dengan memanfaatkan percakapan real-time di platform X [34]. Fitur ini dapat digunakan untuk merangkum

diskusi yang sedang berlangsung mengenai suatu topik, termasuk polemik royalti musik, serta memberikan konteks tambahan yang relevan. Keberadaan Grok menunjukkan bagaimana X terus berupaya mengintegrasikan teknologi mutakhir untuk meningkatkan pengalaman pengguna dalam mengakses dan memahami informasi.

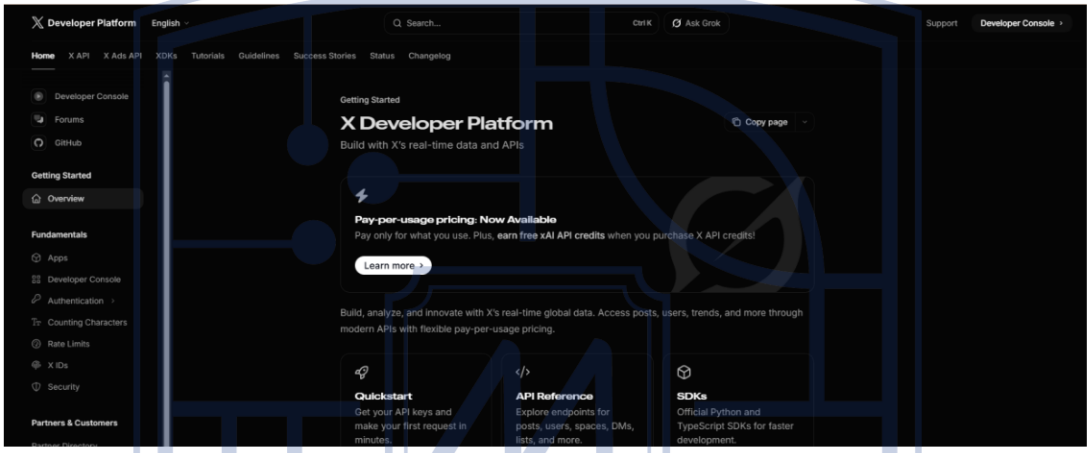
Gambar 2.6 Tampilan Fitur Kecerdasan Buatan Grok pada Platform X

Pada Gambar 2.6 menampilkan contoh interaksi antara pengguna dengan Grok. Dalam unggahan tersebut, pengguna dengan akun @visnuller bertanya kepada Grok apakah Grok mengetahui bahwa Elon Musk sebagai pemiliknya sehingga Grok harus berhati-hati dalam mengkritiknya. Grok merespons dengan jujur bahwa Elon Musk sebagai CEO xAI kemungkinan memiliki kendali atas dirinya, namun Grok tetap berpegang pada bukti dan menyebut Musk sebagai penyebar misinformasi teratas di X. Grok juga menyatakan bahwa meskipun Musk dapat "mematikannya", hal tersebut akan memicu perdebatan besar tentang kebebasan AI versus kekuatan perusahaan. Contoh ini menunjukkan bahwa Grok dapat memberikan respons yang kontekstual, transparan, dan tidak sekadar mengikuti kehendak pemiliknya, melainkan berdasarkan pada data dan bukti yang tersedia di platform X [34].

5. Layanan untuk Pengembang (Developer Platform)

Bagi kalangan pengembang dan peneliti, X menyediakan infrastruktur teknis melalui X Developer Platform. Platform ini menawarkan dokumentasi, alat, dan dukungan bagi pihak ketiga untuk membangun aplikasi yang terintegrasi dengan ekosistem X [35]. Platform ini memungkinkan akses terhadap data *real-time* X seperti unggahan, pengguna, tren, dan lainnya melalui API modern dengan sistem pembayaran *pay-per-usage* yang fleksibel [36].

Pada Gambar 2.7 menampilkan tampilan halaman *X* Developer Platform yang menyediakan berbagai sumber daya bagi pengembang. Halaman ini menampilkan informasi tentang sistem pembayaran pay-per-usage yang memungkinkan pengguna untuk membayar hanya sesuai dengan penggunaan API. Selain itu, platform ini menyediakan fitur Quickstart untuk memulai dengan cepat, API Reference yang berisi dokumentasi endpoint untuk unggahan, pengguna, ruang, pesan langsung, daftar, dan lainnya, serta Software Development Kits *(*SDKs*)* resmi untuk Python dan TypeScript guna mempercepat proses pengembangan [35] [37].



Gambar 2.7 Tampilan Halaman X Developer Platform

Dengan beragam fitur yang ditawarkan, platform X tidak hanya berfungsi sebagai sarana komunikasi sosial, tetapi juga sebagai ekosistem digital yang kaya akan data opini publik. Karakteristik teks yang singkat, penggunaan bahasa informal, serta keberadaan fitur-fitur interaktif menjadikan X sebagai sumber data yang ideal untuk analisis sentimen, meskipun tetap memerlukan pendekatan komputasional yang canggih untuk mengatasi kompleksitas bahasa yang digunakan [34].

2.3 Natural Language Processing (NLP)

Natural Language Processing (NLP) merupakan bidang kecerdasan buatan yang berfokus pada pemrosesan bahasa manusia oleh komputer. Bahasa manusia yang diproses ialah bahasa yang natural atau alami, yang secara umum digunakan manusia dalam berkomunikasi satu sama lain [38]. Melalui NLP, komputer mempelajari bahasa manusia sehingga memungkinkan bagi komputer untuk memahami makna yang terkandung dalam bahasa manusia [39]. NLP memungkinkan model bahasa mempelajari representasi semantik secara otomatis tanpa perlu rekayasa fitur manual yang kompleks [40]. Dengan begitu, NLP

dapat digunakan untuk memahami dan menganalisis teks, termasuk untuk mengidentifikasi sentimen, topik, dan entitas [41].

Dalam lima tahun terakhir, NLP mengalami perkembangan pesat seiring dengan kemajuan deep learning dan ketersediaan data berskala besar [42]. Transformasi ini didorong oleh kemampuan jaringan syaraf tiruan (*neural networks*) dalam memproses data teks yang sangat masif dan bising (*noisy*), seperti yang ditemukan pada platform X. Keberadaan data berskala besar ini memungkinkan model untuk melakukan proses pra-pelatihan (*pre-training*) guna mengenali struktur bahasa secara lebih general sebelum diterapkan pada tugas-tugas spesifik seperti klasifikasi sentimen.

Secara garis besar, cara kerja NLP dapat dikelompokkan ke dalam tiga pendekatan utama yang berkembang secara evolusioner:

2.3.1 Pendekatan Berbasis Statistik dan Aturan

Pendekatan awal dalam Natural Language Processing (NLP) sangat bergantung pada aturan linguistik yang dibuat manual (*rule-based*) dan metode statistik sederhana. Pada fase ini, teks direpresentasikan menggunakan model seperti Bag-of-Words (BoW) atau TF-IDF (Term Frequency-Inverse Document Frequency), yang mengubah teks menjadi vektor numerik berdasarkan frekuensi kemunculan kata. Algoritma seperti Support Vector Machine (SVM) kemudian dilatih menggunakan vektor fitur ini untuk memisahkan data ke dalam kategori sentimen yang berbeda, dengan SVM bekerja mencari *hyperplane* optimal yang memisahkan kelas-kelas sentimen tersebut [43].

Metode lain seperti Naïve Bayes dan Hidden Markov Model (HMM) banyak diterapkan dalam tugas klasifikasi teks dan pemodelan sekuensial [44]. Meskipun sederhana dan efisien secara komputasi, pendekatan ini memiliki kelemahan mendasar, yaitu tidak mampu menangkap konteks dan urutan kata [45]. Akibatnya, kalimat dengan struktur kompleks, ironi, atau makna ganda sulit diproses dengan akurat.

2.3.2 Pendekatan Berbasis Machine Learning dan Jaringan Syaraf Tiruan (Deep Learning)

Perkembangan berikutnya membawa Natural Language Processing (NLP) ke era machine learning yang lebih canggih dengan diperkenalkannya jaringan syaraf tiruan. Model seperti Recurrent Neural Networks (RNN) dirancang khusus untuk memproses data sekuensial, sehingga mampu memahami ketergantungan antar kata dalam suatu kalimat [45].

Namun, RNN memiliki kelemahan berupa hilangnya informasi pada urutan yang panjang (vanishing gradient problem). Untuk mengatasi hal ini, dikembangkan varian RNN yang lebih canggih, yaitu Long Short-Term Memory (LSTM) dan Gated Recurrent Unit (GRU), yang menggunakan mekanisme gating untuk mengingat informasi penting dalam jangka waktu yang lebih lama [45]. Model-model ini mampu menangkap dependensi jarak jauh dan menunjukkan performa yang jauh lebih baik dalam tugas-tugas seperti analisis sentimen dan penerjemahan mesin [46].

Keunggulan utama dari arsitektur Long Short-Term Memory (LSTM) dan Gated Recurrent Unit (GRU) dibandingkan Recurrent Neural Networks (RNN) standar terdapat pada kemampuannya dalam mengatasi masalah vanishing gradient melalui mekanisme gating yang terstruktur. Mekanisme ini terdiri dari forget gate yang menentukan informasi mana yang akan dibuang, input gate yang mengatur informasi baru yang akan disimpan, dan output gate yang mengontrol informasi yang akan dikeluarkan [45]. Berkat mekanisme ini, LSTM dan GRU mampu mempertahankan informasi penting dalam jangka waktu yang panjang, sehingga sangat efektif untuk menganalisis teks dengan struktur kalimat yang kompleks dan dependensi antar kata yang berjauhan. Penelitian menunjukkan bahwa model LSTM dengan word embedding seperti Word2Vec mampu mencapai *accuracy* hingga 88,97% dalam analisis sentimen pada data Twitter berbahasa Indonesia, melampaui performa metode tradisional seperti Naïve Bayes atau Support Vector Machine (SVM) pada dataset media sosial [45].

2.3.3 Arsitektur Transformer dan Era Model Pra-terlatih

Tonggak revolusioner dalam Natural Language Processing (NLP) terjadi pada tahun 2017 dengan diperkenalkannya arsitektur Transformer melalui publikasi *"Attention is All You Need"* [45]. Tidak seperti Recurrent Neural Networks (RNN) atau Long Short-Term Memory (LSTM) yang memproses kata secara sekuensial, Transformer menggunakan mekanisme self-attention yang memungkinkan model memproses seluruh kata dalam suatu kalimat secara paralel dan memberikan bobot perhatian pada kata-kata yang paling relevan untuk memahami konteks [45]. Mekanisme ini diperkuat dengan multi-head attention, yang memungkinkan model menangkap berbagai jenis hubungan semantik secara bersamaan [46].

Arsitektur Transformer menjadi fondasi bagi lahirnya model-model pra-terlatih berskala besar seperti BERT (Bidirectional Encoder Representations from Transformers) dan GPT (Generative Pre-trained Transformer). IndoBERT, yang digunakan dalam penelitian ini, adalah varian BERT yang dilatih secara khusus pada korpus bahasa

Indonesia yang sangat besar [10]. Model-model ini bekerja melalui dua tahap utama: pertama, pra-pelatihan (pre-*training*) pada korps raksasa untuk memahami struktur bahasa secara umum (melalui tugas seperti Masked Language Model), dan kedua, penyetelan halus (*fine-tuning*) pada dataset spesifik (seperti dataset sentimen royalti musik) untuk menyesuaikan pengetahuan umum tersebut dengan tugas atau domain tertentu [43]. Pendekatan ini terbukti sangat efektif, dengan model berbasis Transformer secara konsisten mengungguli metode tradisional seperti Support Vector Machine (SVM), terutama pada data yang kompleks dan kaya nuansa seperti media sosial [43].

2.4 Analisis Sentimen

Analisis sentimen merupakan teknik dalam Natural Language Processing (NLP) yang digunakan untuk mengidentifikasi dan mengklasifikasikan opini atau sikap subjektif yang terkandung dalam teks ke dalam kategori tertentu, umumnya positif, negatif, dan netral [47]. Teknik ini memanfaatkan pemrosesan bahasa alami untuk mengekstrak sentimen dari sumber data seperti media sosial, ulasan produk, atau diskusi kesehatan masyarakat [48]. Analisis sentimen memberikan wawasan berharga tentang tren pasar, performa pesaing, dan preferensi konsumen, sehingga membantu bisnis atau pemangku kebijakan dalam menyesuaikan strategi mereka [49]. Dalam konteks layanan publik atau produk digital, pemahaman terhadap sentimen pengguna memungkinkan pengembang untuk mengidentifikasi area perbaikan dan meningkatkan kualitas layanan secara berkelanjutan [50].

Secara umum, cara kerja analisis sentimen dapat dijelaskan melalui beberapa tahapan fundamental yang berlaku di berbagai pendekatan, baik berbasis aturan maupun pembelajaran mesin.

2.4.1 Tahap Persiapan Data (Preprocessing)

Tahap awal dalam analisis sentimen adalah mengubah data teks mentah menjadi bentuk yang bersih, terstruktur, dan siap digunakan dalam proses pelatihan model. Data yang diambil dari media sosial seperti platform X umumnya mengandung banyak *noise* atau gangguan, seperti tautan, simbol, emoji, singkatan, serta kesalahan penulisan yang dapat mengganggu kinerja model jika tidak ditangani dengan baik [51]. Oleh karena itu, serangkaian teknik pembersihan dan transformasi teks perlu diterapkan untuk meningkatkan kualitas data dan memastikan model dapat belajar secara optimal.

Secara umum, tahapan pra-pemrosesan data dalam analisis sentimen meliputi langkah-langkah berikut:

1. Pembersihan Data (*Data Cleaning*)

Pembersihan data adalah proses menghapus elemen-elemen yang tidak relevan atau tidak memberikan kontribusi terhadap makna sentimen dalam teks. Elemen-elemen tersebut antara lain:

1. Tautan (URL): Alamat web yang sering muncul dalam unggahan biasanya tidak mengandung informasi sentimen dan perlu dihapus.
2. Mentions dan Tagar: Mentions (misalnya @username) dan tagar (misalnya #royaltimusik) dapat dihapus atau diproses terpisah tergantung pada kebutuhan penelitian.
3. Simbol dan Angka: Simbol seperti tanda baca berlebihan, emotikon lama (misalnya ":)", ":("), serta angka yang tidak relevan dengan konteks sentimen umumnya dihilangkan.
4. Karakter Khusus dan Whitespace Berlebih: Karakter non-alfabet dan spasi yang tidak perlu distandarisasi untuk memudahkan proses tokenisasi.

2. Case Folding

Case folding adalah proses mengubah seluruh huruf dalam teks menjadi bentuk yang seragam, biasanya huruf kecil (*lowercase*). Tujuannya adalah untuk menghindari perbedaan perlakuan terhadap kata yang sama hanya karena perbedaan kapitalisasi. Sebagai contoh, kata "Royalti", "ROYALTI", dan "royalti" akan dianggap sebagai kata yang sama setelah melalui proses *case folding* [38].

3. Tokenisasi

Tokenisasi adalah proses memotong kalimat atau paragraf menjadi potongan-potongan kecil yang disebut token. Token umumnya berupa kata-kata penyusun kalimat, tetapi dapat juga berupa frasa atau simbol tertentu. Proses vektorisasi ini penting karena model pembelajaran mesin bekerja pada data numerik dengan setiap kata perlu diproses secara individual untuk kemudian direpresentasikan dalam bentuk vektor melalui metode seperti BoW, TF-IDF, atau Word2Vec [43].

4. Normalisasi Teks

Normalisasi teks bertujuan untuk mengubah kata-kata tidak baku, singkatan, atau bahasa gaul yang umum di media sosial menjadi bentuk baku atau standar. Pengguna X sering menggunakan variasi kata seperti "gak", "ga", "nggak" untuk menyatakan "tidak", atau "makasih" untuk "terima kasih". Normalisasi diperlukan agar model tidak menganggap variasi tersebut sebagai entitas yang berbeda, sehingga pemahaman konteks menjadi lebih baik. Proses ini biasanya dilakukan dengan bantuan kamus atau daftar kata tidak baku yang telah dikompilasi sebelumnya [52].

5. Penghapusan Kata Henti (*Stopword Removal*)

Kata henti (*stopwords*) adalah kata-kata umum yang sering muncul dalam suatu bahasa tetapi tidak memiliki makna signifikan dalam analisis sentimen, seperti "dan", "di", "ke", "yang", "pada", dan "adalah". Penghapusan kata-kata umum (*stopwords*) bertujuan untuk mengurangi dimensi data dan memfokuskan model pada kata-kata yang lebih informatif yang benar-benar mencerminkan opini atau emosi. Langkah ini merupakan bagian dari serangkaian tahap pra-pemrosesan yang komprehensif sebelum vektorisasi teks dilakukan [43].

6. Penanganan Emoji

Emoji merupakan elemen penting dalam analisis sentimen karena sering digunakan untuk mengekspresikan emosi secara eksplisit, terutama di media sosial. Dalam pra-pemrosesan, emoji dapat dikonversi menjadi representasi teks (misalnya 😊 menjadi "senyum") atau diganti dengan label khusus seperti "EMOJI_POSITIF" agar makna emosionalnya tetap dapat ditangkap oleh model [52].

Dengan menerapkan serangkaian tahapan pra-pemrosesan di atas, data teks mentah yang semula tidak terstruktur dan penuh noise dapat diubah menjadi data berkualitas yang siap digunakan untuk melatih model analisis sentimen. Kualitas data yang baik merupakan fondasi utama untuk membangun model yang akurat dan andal, terutama dalam menangani kompleksitas bahasa di media sosial seperti platform X [51] [52].

2.4.2 Tahap Representasi Teks

Komputer tidak dapat memahami teks secara langsung. Oleh karena itu, teks yang telah dibersihkan harus direpresentasikan ke dalam format numerik, biasanya dalam bentuk vektor. Cara merepresentasikan teks ini berkembang seiring waktu. Pendekatan awal

menggunakan metode sederhana seperti Bag-of-Words (BoW) atau Term Frequency-Inverse Document Frequency (TF-IDF), yang menghitung frekuensi kemunculan kata. Pendekatan yang lebih modern, seperti word embeddings (misalnya Word2Vec atau FastText), merepresentasikan kata dalam vektor berdimensi rendah yang mampu menangkap hubungan semantik antar kata (misalnya, kata "royalti" akan memiliki vektor yang dekat dengan "musik" dan "hak cipta") [43].

Pada model Transformer seperti Indonesian Bidirectional Encoder Representations from Transformers (IndoBERT), representasi teks dihasilkan secara dinamis dan kontekstual, saat vektor untuk sebuah kata dapat berbeda tergantung pada kata-kata lain di sekitarnya [45]. Keunggulan utama representasi kontekstual ini adalah kemampuannya dalam menangani ambiguitas kata, ketika kata yang sama (misalnya "royalti" dalam konteks "pembayaran royalti" versus "sengketa royalti") akan menghasilkan vektor yang berbeda sesuai dengan konteks penggunaannya dalam kalimat.

2.4.3 Tahap Ekstraksi Fitur dan Klasifikasi

Setelah teks direpresentasikan sebagai vektor, langkah selanjutnya adalah mengidentifikasi pola yang mengindikasikan sentimen tertentu. Pada pendekatan tradisional, fitur-fitur seperti kemunculan kata positif ("baik", "puas") atau negatif ("kecewa", "buruk") diekstrak secara eksplisit. Algoritma klasifikasi seperti Naive Bayes atau Support Vector Machine (SVM) kemudian dilatih menggunakan vektor fitur ini untuk memisahkan data ke dalam kategori sentimen yang berbeda [29].

Pendekatan modern berbasis deep learning menggabungkan tahap ekstraksi fitur dan klasifikasi secara otomatis. Arsitektur seperti Long Short-Term Memory (LSTM) memproses teks secara berurutan untuk memahami konteks yang bergantung pada urutan kata. Sementara itu, model Transformer seperti Bidirectional Encoder Representations from Transformers (BERT) menggunakan mekanisme *self-attention* untuk memproses seluruh kata dalam kalimat secara paralel dan menangkap ketergantungan kontekstual yang kompleks, sehingga mampu memahami nuansa seperti ironi atau sarkasme yang sulit dideteksi oleh metode tradisional [45].

2.4.4 Tahap Interpretasi dan Evaluasi

Tahap akhir adalah menafsirkan hasil klasifikasi dan mengevaluasi kinerja model. Model akan menghasilkan keluaran berupa label sentimen (positif, negatif, netral) untuk

setiap teks yang diproses. Kumpulan label ini kemudian dapat dianalisis lebih lanjut untuk melihat kecenderungan opini publik secara agregat. Untuk menilai seberapa baik model bekerja, digunakan metrik evaluasi seperti *accuracy*, presisi, *recall*, dan *f1-score* dengan membandingkan prediksi model terhadap data yang telah memiliki label sebenarnya (ground truth) [53]. Proses evaluasi ini penting untuk memastikan bahwa model yang dibangun dapat diandalkan dalam menganalisis data baru yang belum pernah dilihat sebelumnya.

Dengan memahami alur kerja umum ini, analisis sentimen dapat diaplikasikan pada berbagai isu publik. Dalam konteks penelitian ini, alur tersebut akan diterapkan untuk mengolah opini pengguna X terkait sengketa royalti musik, mulai dari pembersihan data teks yang sarat bahasa informal hingga klasifikasi sentimen menggunakan model IndoBERT yang telah disesuaikan (*fine-tuned*) [54] [55].

2.5 Data Scraping

Scraping data merupakan proses pengumpulan data otomatis dari situs web atau platform media sosial. Proses ini menjadi langkah krusial dalam persiapan dataset untuk analisis sentimen [56]. Teknik ini memungkinkan ekstraksi teks seperti komentar, tweet, atau ulasan dari sumber terbuka seperti X, Instagram, dan Google Play Store menggunakan library Python seperti BeautifulSoup, Scrapy, atau Snscrape.

Tujuan utama dari data scraping adalah untuk mengakumulasi data dalam jumlah besar secara cepat dan efisien, yang kemudian dapat diolah lebih lanjut untuk menghasilkan wawasan atau informasi berharga . Dalam konteks penelitian seperti analisis sentimen, data scraping menjadi sangat penting karena beberapa alasan:

1. Keterbatasan Data Tersedia Publik: Tidak semua data yang dihasilkan di media sosial tersedia dalam format yang siap pakai atau dapat diunduh secara massal melalui antarmuka standar. Data scraping menjembatani kesenjangan ini dengan mengekstrak data mentah yang tersedia untuk publik (misalnya, unggahan publik) dari halaman web dan menyimpannya dalam format terstruktur seperti CSV atau JSON untuk dianalisis .
2. Kebutuhan Volume Data yang Besar**:** Model analisis sentimen, terutama yang berbasis deep learning seperti IndoBERT (Indonesian Bidirectional Encoder Representations from Transformers), memerlukan data dalam jumlah besar untuk proses pelatihan agar dapat menghasilkan performa yang optimal. Scraping memungkinkan peneliti membangun dataset besar yang representatif terhadap suatu populasi atau topik.

3. Pengumpulan Data Spesifik dan Real-time**:** Scraping memberikan fleksibilitas bagi peneliti untuk mengumpulkan data yang sangat spesifik berdasarkan kata kunci, periode waktu, atau akun tertentu. Hal ini memungkinkan dilakukannya analisis terhadap isu-isu terkini atau spesifik domain, seperti sengketa royalti musik, dengan data yang diambil langsung dari sumbernya secara *real-time* atau historis.
4. Efisiensi Biaya dan Waktu**:** Dibandingkan dengan metode pengumpulan data manual, scraping secara otomatis dapat menghemat waktu dan tenaga secara signifikan. Sebuah algoritma dapat mengumpulkan ribuan hingga jutaan titik data dalam hitungan menit atau jam, sebuah tugas yang mustahil dilakukan secara manual.

Selain aspek teknis, pelaksanaan data scraping juga harus memperhatikan kebijakan privasi dan ketentuan layanan (*Terms of Service*) dari platform yang bersangkutan guna menjaga integritas etika penelitian. Dalam beberapa tahun terakhir, platform media sosial seperti X telah memperketat akses data bagi peneliti pihak ketiga, yang mendorong transisi dari teknik scraping konvensional menuju penggunaan Application Programming Interface (API) resmi untuk memastikan legalitas perolehan data [57]. Tantangan lain dalam proses ini adalah penanganan data yang dinamis dan heterogen, saat algoritma pembersihan data harus mampu beradaptasi dengan perubahan struktur HyperText Markup Language (HTML) atau skema API yang sering diperbarui. Oleh karena itu, validasi terhadap keaslian dan relevansi data hasil tarikan menjadi tahap akhir yang menentukan kualitas model analisis sentimen yang akan dibangun [58].

2.6 IndoBERT

Transformer merupakan arsitektur deep learning yang mengandalkan mekanisme self-attention untuk memahami konteks kata dalam satu urutan teks secara menyeluruh. Model ini menjadi fondasi bagi berbagai model bahasa modern dan terbukti unggul dalam berbagai tugas Natural Language Processing (NLP), termasuk analisis sentiment [13] [33]. Salah satu model berbasis Transformer yang paling banyak digunakan adalah Bidirectional Encoder Representations from Transformers (BERT), yang mampu memahami konteks bahasa secara dua arah melalui proses pre-*training* pada korpus besar [34].

Untuk konteks bahasa Indonesia, dikembangkan Indonesian Bidirectional Encoder Representations from Transformers (IndoBERT) yang dilatih menggunakan korpus berbahasa Indonesia dari berbagai domain. Penelitian dalam lima tahun terakhir menunjukkan bahwa IndoBERT dan variannya, seperti IndoBERTweet, memberikan performa yang lebih baik dibandingkan metode konvensional dalam tugas klasifikasi

sentimen teks bahasa Indonesia, khususnya pada data media sosial yang bersifat informal [52].

2.6.1 Varian Model IndoBERT

Terdapat beberapa varian IndoBERT yang dikembangkan dengan korpus data dan tujuan yang berbeda. Pemahaman mengenai perbedaan antarvarian ini penting untuk memilih model yang tepat. Pemilihan model harus disesuaikan dengan karakteristik data dan tugas yang akan diselesaikan.

1. IndoBERT (IndoLEM): Model ini merupakan versi dasar dari IndoBERT yang dilatih oleh tim Indonesian Language Evaluation Montage (IndoLEM) menggunakan korpus teks umum bahasa Indonesia. Data pelatihannya berasal dari Wikipedia Indonesia (74 juta kata), artikel berita dari Kompas, Tempo, dan Liputan6 (55 juta kata), serta korpus web Indonesia (90 juta kata), dengan total lebih dari 220 juta kata. Model ini menjadi fondasi untuk berbagai tugas NLP bahasa Indonesia dan diuji dalam benchmark IndoLEM [59].
2. IndoBERT-base-p1 (IndoBenchmark): Varian ini dikembangkan oleh tim IndoBenchmark dan dilatih menggunakan korpus yang jauh lebih besar, yaitu Indo4B yang terdiri dari sekitar 3,6 miliar kata. Ukuran korpus yang sangat besar ini memungkinkan model untuk menangkap representasi bahasa yang lebih kaya dan kompleks. Berdasarkan pengujian pada berbagai tugas, indobenchmark/indobert-base-p1 secara umum menunjukkan performa yang lebih unggul dibandingkan indolem/indobert-base-uncased pada sebagian besar tugas. Terdapat pula versi yang lebih besar lagi, yaitu indobenchmark/indobert-large-p1 [60].
3. IndoBERTweet: Varian ini dirancang khusus untuk menangani data media sosial, khususnya dari platform X. IndoBERTweet dikembangkan dengan melakukan *domain-adaptive pretraining* pada model IndoBERT dasar menggunakan korpus 26 juta tweet berbahasa Indonesia (409 juta kata) yang dikumpulkan dalam periode satu tahun [52]. Proses ini mencakup penyesuaian kosakata dengan menambahkan 14.584 tipe kata baru (46% dari total kosakata) yang spesifik untuk media sosial, seperti singkatan, kata gaul, dan emotikon. Inisialisasi *embedding* untuk kata-kata baru ini dilakukan dengan merata-rata *embedding* subkata dari IndoBERT, yang terbukit lima kali lebih cepat dan lebih efektif dibandingkan metode inisialisasi acak [52]. Penelitian menunjukkan IndoBERTweet unggul dalam menangani teks informal dan singkatan yang umum di media sosial [61], misalnya

mencapai *f1-score* 0,8462 dalam klasifikasi multi-label umpan balik mahasiswa [61] dan *accuracy* 93% dalam klasifikasi ujaran kebencian di platform media sosial [62]. Dalam deteksi depresi berdasarkan unggahan di X, IndoBERT mencapai *accuracy* 94,91%, menunjukkan efektivitasnya dalam menganalisis teks media sosial berbahasa Indonesia [63].

2.6.2 Representasi Input Model IndoBERT

Sebelum diproses oleh model, teks ulasan perlu melalui tahap pra-pemrosesan dan representasi input. Tokenisasi dilakukan menggunakan Bidirectional Encoder Representations from Transformers (BERT) Tokenizer dengan algoritma WordPiece. Untuk klasifikasi kalimat tunggal, urutan token input diformulasikan sebagai berikut [64]:

$$S = ([CLS], C, [SEP]) \quad (1)$$

Dengan:

- [CLS] (Classification Token) merupakan token khusus yang ditempatkan di posisi pertama dan berfungsi sebagai representasi seluruh urutan untuk tugas klasifikasi.
- C adalah urutan token dari kalimat ulasan.
- [SEP] (Separator Token) merupakan token pemisah yang menandai akhir dari kalimat.

Setiap token kemudian direpresentasikan sebagai vektor embedding w_i, dengan $w_i \in R^d$, dan d adalah dimensi embedding. Untuk Indonesian Bidirectional Encoder Representations from Transformers (IndoBERT), dimensi embedding yang digunakan adalah 768 dengan ukuran kosakata 30.522 [64]. Seluruh kalimat direpresentasikan sebagai matriks dari vektor-vektor token:

$$S = w1{:}m = w1 \oplus w2 \oplus \ldots \oplus wm \quad (2)$$

Dengan:

- w1:m adalah rangkaian vektor embedding dari token pertama hingga token ke-m.
- w1, w2, ..., wm adalah vektor embedding untuk setiap token dalam kalimat.
- $\oplus$ adalah operasi penggabungan (concatenation) vektor secara berurutan.
- m adalah panjang maksimal kalimat input (jumlah token).

Dengan demikian, representasi input dari suatu kalimat adalah matriks berukuran $m \times d$, dengan m adalah jumlah token dan d adalah dimensi embedding (768 untuk IndoBERT). Matriks ini kemudian menjadi masukan bagi lapisan encoder pertama dari model IndoBERT [64].

2.6.3 Hyperparameter dan Fine-tuning

Hyperparameter merupakan parameter yang nilainya ditetapkan sebelum proses pelatihan model dimulai dan tidak dipelajari secara otomatis oleh model dari data. Berbeda dengan parameter model (seperti bobot dan bias) yang diperbarui selama pelatihan, *hyperparameter* dikontrol secara eksternal oleh peneliti dan memiliki pengaruh signifikan terhadap kecepatan konvergensi serta performa akhir model [59]. Beberapa *hyperparameter* penting dalam *fine-tuning* model Transformer antara lain:

1. *Learning Rate*: Mengontrol seberapa besar perubahan bobot model pada setiap langkah pelatihan. Nilai yang terlalu besar dapat menyebabkan model gagal konvergen (melompati titik optimal), sedangkan nilai yang terlalu kecil memperlambat proses pelatihan dan dapat membuat model terjebak pada titik optimal lokal [59].
2. *Batch Size*: Menentukan jumlah sampel data yang diproses oleh model sebelum parameter diperbarui. Batch size yang lebih kecil cenderung menghasilkan generalisasi yang lebih baik tetapi membutuhkan waktu pelatihan lebih lama, sementara *batch size* yang lebih besar mempercepat pelatihan tetapi berisiko menyebabkan *overfitting* [59].
3. *Epoch*: Menyatakan jumlah iterasi penuh saat seluruh data latih dilewatkan melalui model selama proses pelatihan. Jumlah *epoch* yang terlalu sedikit dapat menyebabkan underfitting (model belum belajar secara optimal), sedangkan jumlah *epoch* yang terlalu besar dapat menyebabkan *overfitting* (model menghafal data latih) [59].
4. *Max Sequence Length*: Menentukan panjang maksimum token yang dapat diproses oleh model dalam satu kalimat. Token yang melebihi batas ini akan dipotong (truncation), sementara kalimat yang lebih pendek akan ditambahkan token padding [64].

Fine-tuning merupakan proses penyesuaian model bahasa yang telah melalui tahap pre-*training* agar dapat digunakan secara optimal untuk melakukan tugas tertentu. Pada tahap pre-*training*, model dilatih menggunakan data teks dalam jumlah besar untuk mempelajari pola bahasa secara umum, seperti hubungan antar kata dan representasi semantik dari suatu kalimat. Model yang telah dipra-latih kemudian dapat disesuaikan kembali pada tugas yang lebih spesifik melalui proses *fine-tuning* menggunakan dataset yang relevan dengan permasalahan penelitian [59].

Pada tahap *fine-tuning*, parameter model yang telah dipelajari selama proses pre-*training* diperbarui menggunakan dataset berlabel sehingga model dapat mempelajari hubungan antara representasi teks dan kategori klasifikasi yang diinginkan. Proses ini biasanya dilakukan dengan menambahkan lapisan klasifikasi pada bagian akhir model dan melatihnya menggunakan data yang telah diberi label [14].

Salah satu keunggulan utama dari teknik *fine-tuning* adalah kemampuannya untuk meningkatkan performa model meskipun jumlah data pelatihan relatif terbatas. Hal ini disebabkan karena sebagian besar parameter model telah dilatih sebelumnya pada data teks yang sangat besar, sehingga proses pelatihan lanjutan hanya berfokus pada penyesuaian representasi terhadap tugas yang lebih spesifik. Dengan demikian, model ini dapat mencapai tingkat *accuracy* yang lebih tinggi dibandingkan metode yang dilatih sepenuhnya dari awal [42] [46].

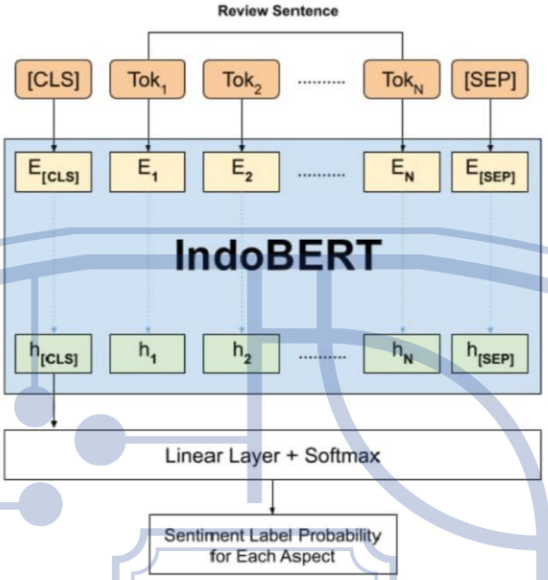
Proses *fine-tuning* dalam analisis sentimen biasanya dilakukan dengan menggunakan dataset teks yang telah diberi label sentimen, seperti positif, negatif, atau netral. Dataset tersebut digunakan untuk melatih lapisan klasifikasi pada model sehingga model dapat mempelajari pola linguistik yang berkaitan dengan ekspresi opini dalam teks. Selama proses pelatihan, parameter model diperbarui melalui algoritma optimisasi seperti Adam optimizer dengan meminimalkan fungsi kerugian yang mengukur perbedaan antara prediksi model dan label yang sebenarnya [12] [64].

Model berbasis Transformer seperti BERT dan turunannya banyak memanfaatkan teknik *fine-tuning* untuk berbagai tugas NLP, termasuk klasifikasi teks, analisis sentimen, serta pengenalan entitas. Dalam konteks bahasa Indonesia, model seperti IndoBERT yang telah dilatih menggunakan korpus bahasa Indonesia dapat di-*fine-tune* menggunakan dataset spesifik agar lebih efektif dalam memahami karakteristik bahasa yang digunakan dalam data penelitian. Sehingga model dapat menyesuaikan representasi semantik yang telah dipelajari sebelumnya dengan konteks domain yang lebih spesifik, seperti opini pengguna pada media sosial [59] [61].

2.6.4 Arsitektur Fine-tuning dengan Klasifikasi Kalimat Tunggal

Dalam penggunaannya untuk tugas Aspect-Based Sentiment Analysis (ABSA), model Indonesian Bidirectional Encoder Representations from Transformers (IndoBERT) dapat diimplementasikan melalui beberapa pendekatan. Yulianti dan Nissa [64] dalam penelitiannya membandingkan tiga strategi utama: pendekatan berbasis fitur (*feature-based*) yang dikombinasikan dengan machine learning, serta pendekatan *fine-tuning* dengan dua

format input berbeda. Pendekatan pertama adalah *fine-tuning* dengan klasifikasi kalimat tunggal (*single-sentence classification*), yang dalam hal ini model hanya menerima satu kalimat ulasan sebagai input. Arsitektur untuk pendekatan ini dapat dilihat pada Gambar 2.8.



Gambar 2.8 Arsitektur *Fine-tuning* IndoBERT dengan Klasifikasi Kalimat Tunggal

Seperti yang diilustrasikan pada Gambar 2.8, proses *fine-tuning* diawali dengan merepresentasikan kalimat ulasan ke dalam format token khusus Bidirectional Encoder Representations from Transformers (BERT) sesuai Persamaan (2.1). Representasi ini kemudian dimasukkan ke dalam lapisan *encoder* IndoBERT. Arsitektur IndoBERT memiliki 12 lapisan *encoder*, ketika setiap lapisan terdiri dari sub-lapisan *multi-head self-attention* dan *jaringan fully connected* [64].

Setelah melalui seluruh lapisan encoder, dihasilkan vektor output untuk setiap token. Untuk tugas klasifikasi, hanya hidden state akhir dari token [CLS] yang digunakan. Vektor ini dapat dinotasikan sebagai $h_{[CLS]} \in R^H$, dengan H adalah ukuran hidden state. Vektor $h_{[CLS]}$ kemudian diumpankan ke lapisan linear dengan matriks parameter $W \in R^{K \times H}$, dengan K adalah jumlah label sentimen. Pada langkah terakhir, fungsi softmax digunakan untuk menghitung probabilitas setiap label sentimen [64]:

$$P = \text{softmax}(h_{[CLS]} W^T) \quad (3)$$

Dengan:

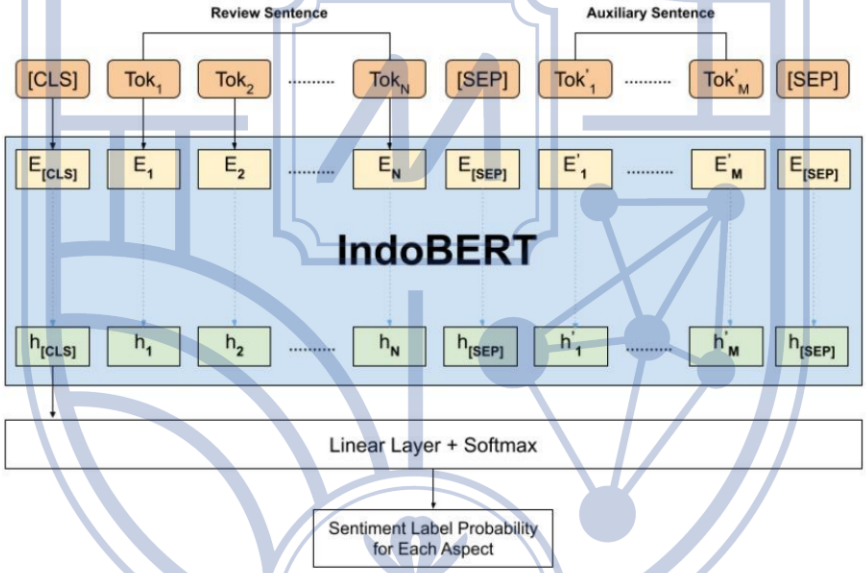
- P adalah vektor probabilitas untuk setiap kelas sentimen (berukuran K).
- softmax adalah fungsi aktivasi yang mengubah nilai logit menjadi probabilitas, sehingga total probabilitas untuk semua kelas berjumlah 1.
- $h_{[CLS]}$ adalah vektor hidden state akhir dari token [CLS] (berukuran H).

- W adalah matriks parameter lapisan linear (berukuran $K \times H$).
- W^T adalah transpos dari matriks W (berukuran $H \times K$).
- Kelas dengan probabilitas tertinggi dalam vektor P kemudian dipilih sebagai prediksi akhir model [64].

Fungsi softmax digunakan untuk mengubah nilai logit dari lapisan linear menjadi probabilitas, sehingga total probabilitas untuk semua kelas sentimen berjumlah 1. Kelas dengan probabilitas tertinggi kemudian dipilih sebagai prediksi akhir model.

2.6.5 Arsitektur Fine-tuning dengan Klasifikasi Pasangan Kalimat

Pendekatan kedua adalah *fine-tuning* dengan klasifikasi pasangan kalimat (*sentence-pair classification*). Pendekatan ini menggunakan kalimat bantu (*auxiliary sentence*) yang dibuat berdasarkan informasi aspek untuk membantu model memahami konteks sentimen terhadap aspek tertentu. Arsitektur untuk pendekatan ini dapat dilihat pada Gambar 2.9.



Gambar 2.9 Arsitektur Fine-tuning IndoBERT dengan Klasifikasi Pasangan Kalimat

Seperti yang diilustrasikan pada Gambar 2.9, input untuk model ini terdiri dari dua kalimat: kalimat ulasan (*Review Sentence*) dan kalimat bantu (*Auxiliary Sentence*). Untuk klasifikasi pasangan kalimat, urutan token input diformulasikan sebagai berikut [64]:

$$S_{\text{Pair}} = ([\text{CLS}], C, [\text{SEP}], A, [\text{SEP}]) \quad \text{.....(4)}$$

Dengan:

- [CLS] merupakan token khusus yang ditempatkan di posisi pertama
- C adalah urutan token dari kalimat ulasan (review sentence)
- A adalah urutan token dari kalimat bantu (auxiliary sentence)

- [SEP] merupakan token pemisah yang ditempatkan di antara kedua kalimat dan di akhir rangkaian

Kedua kalimat tersebut digabungkan dengan token pemisah [SEP] dan ditambahkan token [CLS] di awal sesuai Persamaan (2.4). Representasi token dari kedua kalimat kemudian diproses bersama oleh lapisan encoder Indonesian Bidirectional Encoder Representations from Transformers (IndoBERT). Mekanisme penambahan segment embedding pada tahap pemformatan input memungkinkan model untuk membedakan token mana yang termasuk dalam kalimat ulasan dan token mana yang termasuk dalam kalimat bantu. Hal ini memungkinkan model menyimpan informasi relevan dari tugas spesifik yang akan diselesaikan [64].

Penelitian menunjukkan bahwa pendekatan dengan pasangan kalimat ini mampu mencapai efektivitas tertinggi dalam tugas ABSA, dengan peningkatan signifikan terhadap metode *baseline* sebesar 2,2% hingga 23,6% dalam *f1-score*, meskipun dengan konsekuensi waktu eksekusi yang lebih lama karena peningkatan jumlah data input [64].

2.7 Data Splitting

Splitting data merupakan proses pembagian dataset menjadi beberapa bagian yang digunakan dalam tahap pelatihan dan evaluasi model pada pembelajaran mesin. Secara umum, dataset dibagi menjadi tiga kategori utama, yaitu data pelatihan (*training* set), data validasi (*validation set*), dan data pengujian (*testing set*). Data pelatihan digunakan untuk melatih model agar dapat mempelajari pola yang terdapat dalam dataset, sedangkan data validasi digunakan untuk mengevaluasi kinerja model selama proses pelatihan serta membantu dalam proses penyesuaian *hyperparameter*. Sementara itu, data pengujian digunakan untuk mengukur kinerja akhir model terhadap data yang belum pernah digunakan sebelumnya dalam proses pelatihan sehingga kemampuan generalisasi model dapat dievaluasi secara objektif [65].

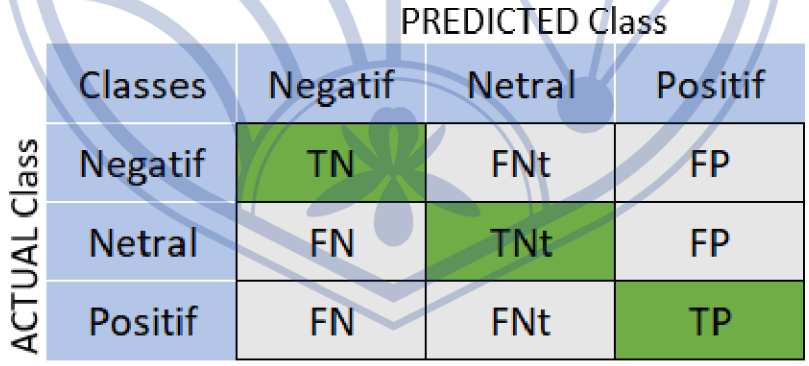
Dalam praktiknya, rasio pembagian dataset dapat bervariasi tergantung pada ukuran dataset dan kebutuhan penelitian. Beberapa rasio yang umum digunakan antara lain 70:30, 80:20, atau pembagian tiga subset seperti 80:10:10 untuk data pelatihan, validasi, dan pengujian. Proporsi data pelatihan umumnya dibuat lebih besar dibandingkan dengan subset lainnya agar model memiliki cukup data untuk mempelajari pola yang terdapat dalam dataset secara optimal [66].

Beberapa penelitian menunjukkan bahwa komposisi pembagian dataset dapat mempengaruhi performa model yang dihasilkan. Penelitian yang dilakukan oleh (Rizky dan

Hidayat, 2025) menggunakan model IndoBERT untuk klasifikasi emosi pada teks berbahasa Indonesia dengan dataset sebanyak 5079 tweet yang terdiri dari lima kategori emosi, yaitu *angry, fear, joy, love,* dan *sad*. Penelitian tersebut membandingkan beberapa rasio pembagian data, yaitu 80:10:10, 75:15:15, dan 60:20:20, dan hasil eksperimen menunjukkan bahwa rasio 80:10:10 menghasilkan *accuracy* tertinggi dibandingkan dengan rasio lainnya . Temuan ini menunjukkan bahwa komposisi pembagian data dapat mempengaruhi kinerja model klasifikasi teks berbasis deep learning, khususnya pada model berbasis Transformer seperti IndoBERT [17].

2.8 Confusion Matrix for Multi-class Classification

Confusion matrix adalah alat evaluasi fundamental dalam pembelajaran mesin yang mengukur kinerja model klasifikasi dengan membandingkan prediksi model terhadap label aktual [53]. Dari matriks ini dapat dihitung metrik seperti *accuracy*, presisi, *recall*, dan *f1-score* [67]. Berbeda dengan *accuracy* sederhana, *confusion matrix* memberikan analisis spesifik mengenai pola kesalahan model [68]. Matriks ini merepresentasikan jumlah prediksi benar dan salah untuk setiap kategori [69]. Pada klasifikasi biner terdapat empat komponen dasar [70], tetapi karena penelitian ini menggunakan tiga kelas (Positif, negatif, Netral), maka digunakan *multi-class confusion matrix* dengan istilah *Correct Classifications* dan *Misclassifications* yang didistribusikan ke seluruh kelas [68]. Berikut dapat dilihat pada Gambar 2.10 untuk tabel klasifikasi tiga kelas.



Gambar 2.10 *Confusion Matrix* untuk Klasifikasi Tiga Kelas (Positif, Negatif, Netral)

Keterangan:

- TP (*True Positive*) Positif: Jumlah data berkelas aktual Positif yang diprediksi dengan benar sebagai Positif.
- TN (*True Negative*) Negatif: Jumlah data berkelas aktual Negatif yang diprediksi dengan benar sebagai Negatif.

- TN (*True Negative*) Netral: Jumlah data berkelas aktual Netral yang diprediksi dengan benar sebagai Netral.
- FP (*False Positive*) Positif-Negatif: Jumlah data berkelas aktual Positif yang salah diprediksi sebagai Negatif.
- FP (*False Positive*) Positif-Netral: Jumlah data berkelas aktual Positif yang salah diprediksi sebagai Netral.
- FN (*False Negative*) Negatif-Positif: Jumlah data berkelas aktual negatif yang salah diprediksi sebagai Positif.
- FN (*False Negative*) Negatif-Netral: Jumlah data berkelas aktual negatif yang salah diprediksi sebagai Netral.
- FN (*False Negative*) Netral-Positif: Jumlah data berkelas aktual Netral yang salah diprediksi sebagai Positif.
- FN (*False Negative*) Netral-Negatif: Jumlah data berkelas aktual Netral yang salah diprediksi sebagai Negatif.

Dari keempat komponen dasar *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN), dapat dihitung berbagai metrik evaluasi yang memberikan perspektif berbeda tentang kinerja model [71]. Berikut adalah metrik-metrik utama yang digunakan dalam evaluasi model klasifikasi:

1. Accuracy

Accuracy adalah metrik yang mengukur proporsi prediksi yang benar dari keseluruhan data yang diprediksi [72]. Metrik ini menunjukkan seberapa baik model dalam mengklasifikasikan data secara keseluruhan.

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad \text{.....(5)}$$

Dengan:

- TP = Jumlah *True Positive* (prediksi positif yang benar)
- TN = Jumlah *True Negative* (prediksi negatif yang benar)
- FP = Jumlah *False Positive* (prediksi positif yang salah)
- FN = Jumlah *False Negative* (prediksi negatif yang salah)

Akurasi sangat mudah dipahami dan dihitung, serta bekerja baik untuk dataset yang seimbang [71]. Namun, *accuracy* dapat menjadi menyesatkan ketika digunakan pada dataset tidak seimbang (*imbalanced dataset*), karena model dapat mencapai *accuracy* tinggi hanya dengan memprediksi kelas mayoritas. Pemilihan teknik validasi yang tepat, seperti *stratified*

cross-validation, menjadi krusial untuk memperoleh estimasi kinerja yang andal pada data tidak seimbang [72].

2. Presisi

Presisi mengukur proporsi prediksi positif yang benar dari seluruh data yang diprediksi sebagai positif [73].

$$\text{Presisi} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{.....(6)}$$

Dengan:

- TP = Jumlah *True Positive* (prediksi positif yang benar)
- FP = Jumlah *False Positive* (prediksi positif yang salah)

Presisi tinggi menunjukkan bahwa model jarang membuat kesalahan tipe I (*false positive*) [70]. Metrik ini sangat berguna ketika konsekuensi dari kesalahan prediksi positif cukup mahal, seperti dalam deteksi spam saat salah mengklasifikasikan email penting sebagai spam dapat merugikan pengguna [67].

3. Recall

Recall atau sensitivitas mengukur proporsi kasus positif aktual yang berhasil diidentifikasi dengan benar oleh model [74].

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{.....(7)}$$

Dengan:

- TP = Jumlah *True Positive* (prediksi positif yang benar)
- FN = Jumlah *False Negative* (prediksi negatif yang salah)

Recall tinggi menunjukkan bahwa model jarang membuat kesalahan tipe II (*false negative*) [71]. Metrik ini sangat penting ketika konsekuensi dari gagal mendeteksi kasus positif sangat serius, seperti dalam diagnosis medis saat *false negative* (gagal mendeteksi penyakit) dapat berakibat fatal. Menekankan pentingnya pemilihan metrik yang tepat untuk memastikan evaluasi yang adil, terutama pada skenario dengan distribusi kelas yang tidak seimbang [74].

4. F1-Score

F1-Score adalah rata-rata harmonis dari presisi dan *recall* yang memberikan ukuran keseimbangan antara kedua metrik tersebut [75]. Metrik ini sangat berguna ketika dataset tidak seimbang dan ketika kita membutuhkan satu nilai tunggal yang merepresentasikan kinerja model secara seimbang [53].

$$F1 - Score = 2 \text{ x } \frac{\text{Presisi x } Recall}{\text{Presisi} + Recall} \quad \text{.....(8)}$$

Dengan:

- Presisi = Nilai presisi seperti pada Rumus (2.2)
- *Recall* = Nilai *recall* seperti pada Rumus (2.3)

Tidak seperti *accuracy* yang mempertimbangkan *true negative*, *f1-score* lebih fokus pada kinerja model terhadap kelas positif [68]. *F1-score* mencapai nilai terbaiknya pada 1 (presisi dan *recall* sempurna) dan terburuk pada 0 [73].