PERBANDINGAN ALGORTIMA RANDOM FOREST DAN SVM UNTUK MULTI-LABEL KLASIFIKASI MOOD BUKU BERBASIS DESKRIPSI TEKS

Intan Sanu, Nur Rachmat
Informatika, Universitas Multi Data Palembang
Jl. Rajawali No.14, 9 Ilir, Palembang, Indonesia
intansanu34@gmail.com

ABSTRAK

Banyaknya buku dengan tema dan genre yang beragam sering menyulitkan pembaca dalam memilih bacaan yang sesuai dengan suasana hati, karena genre tidak selalu merepresentasikan nuansa emosional isi cerita. Permasalahan ini mendorong perlunya sistem klasifikasi *mood* buku berbasis teks deskripsi. Penelitian ini bertujuan untuk membandingkan kinerja algoritma Support Vector Machine (SVM) dan Random Forest dalam klasifikasi multi-label mood buku. Dataset yang digunakan adalah *7k Books* dari Kaggle, dengan atribut *title* dan *description*. Karena dataset tidak memiliki label emosi, dilakukan pelabelan otomatis menggunakan model emotion classifier berbasis Transformer (DistilRoBERTa) menjadi tujuh kategori mood, yaitu *joy, sadness, anger, fear, disgust, surprise,* dan *neutral*. Metode yang digunakan meliputi pra-pemrosesan teks, representasi fitur menggunakan TF-IDF (unigram dan bigram), serta pendekatan Binary Relevance dengan optimasi parameter menggunakan GridSearchCV. Evaluasi dilakukan menggunakan metrik macro-averaging, micro-averaging, dan Hamming Loss. Hasil penelitian menunjukkan bahwa Linear SVM memperoleh micro-averaging F1-score sebesar 0,7909, sedikit lebih tinggi dibandingkan Random Forest sebesar 0,7856, dengan Hamming Loss sekitar 0,11 untuk kedua model. Hasil ini menunjukkan bahwa Linear SVM lebih unggul secara keseluruhan, meskipun performa pada beberapa label masih dipengaruhi oleh ketidakseimbangan data.

Kata kunci : *Klasifikasi Multi-Label, Mood Buku, Random Forest, Support Vector Machine, TF-IDF*

1. PENDAHULUAN

Buku merupakan media penting dalam pengembangan pengetahuan dan pengalaman emosional pembaca [1]. Namun, banyaknya pilihan bacaan yang tersedia sering membuat pembaca kesulitan dalam memilih buku yang sesuai dengan preferensi dan suasana hati mereka. Sejumlah studi menunjukkan bahwa *mood* memiliki pengaruh signifikan terhadap preferensi membaca, di mana pembaca cenderung memilih bacaan yang sejalan dengan kondisi emosional yang sedang dialami [2]. Oleh karena itu, sistem rekomendasi berbasis *mood* menjadi pendekatan yang relevan untuk meningkatkan pengalaman membaca [3].

Mood didefinisikan sebagai keadaan afektif berdurasi relatif panjang dengan intensitas rendah yang dapat berubah secara bertahap [4]. Dalam konteks literatur, satu buku dapat memunculkan lebih dari satu *mood* secara bersamaan, sehingga pendekatan klasifikasi single-label kurang mampu merepresentasikan kompleksitas emosi dalam teks. Permasalahan ini menunjukkan perlunya pendekatan klasifiikasi multi-label dalam pengelompokan *mood* buku.

Penelitian sebelumnya telah mengembangkan klasifikasi dan rekomendasi buku berbasis mood menggunakan berbagai metode, mulai dari TF-IDF dan Logistic Regression hingga model berbasis Transformer seperti DistilRoBERTa dan BERT [3]. Meskipun model Transformer menunjukkan performa tinggi, metode klasik seperti Support Vector Machine (SVM) dan Random Forest tetap relevan karena stabilitas, interpretabilitas, serta performa kompetitifnya pada data teks berdimensi tinggi [5]. Namun, penelitian yang secara khusus membandingkan kedua algoritma tersebut dalam skenario klasifikasi multi-label *mood* buku masih terbatas.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk membandingkan performa algoritma Support Vector Machine dan Random Forest dalam klasifikasi multi-label *mood* buku berbasis deskripsi teks. Rencana penyelesaian masalah dilakukan dengan menerapkan representasi fitur TF-IDF (unigram dan bigram), pendekatan Binary Relevance sebagai metode problem transformation, serta optimasi parameter menggunakan GridSearchCV. Evaluasi performa model dilakukan menggunakan metrik macro-averaging, micro-averaging, dan Hamming Loss. Hasil penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem rekomendasi buku berbasis mood yang lebih akurat dan relevan bagi pengguna.

2. TINJAUAN PUSTAKA

Kajian pustaka disusun untuk memberikan dasar teoretis serta gambaran umum mengenai penelitian-penelitian relevan terkait klasifikasi teks, analisis *mood*, dan pendekatan multi-label. Proses ini dilakukan dengan menelaah berbagai sumber ilmiah seperti jurnal, prosiding, buku akademik, dan laporan penelitian yang membahas algoritma klasifikasi, representasi teks, serta studi-studi yang menyoroti *mood* dalam konteks literatur.

Penelitian terdahulu menunjukkan bahwa klasifikasi *mood* pada teks memiliki tantangan yang

kompleks, terutama karena satu dokumen dapat memunculkan lebih dari satu emosi secara bersamaan. Hal ini menempatkan pendekatan klasifikasi multi-label sebagai metode yang lebih sesuai dibandingkan klasifikasi tunggal. Berbagai teknik representasi teks, seperti TF-IDF, telah digunakan secara luas dalam analisis teks, bersama dengan algoritma Support Vector Machine dan Random Forest untuk tugas klasifikasi emosi dan genre [6].

Terdapat beberapa penelitian yang menekankan pentingnya memahami hubungan antara *mood* dan pengalaman membaca. Sistem rekomendasi berbasis *mood* terbukti dapat meningkatkan relevansi rekomendasi buku melalui penyesuaian terhadap kondisi emosional pengguna [3]. Namun, sebagian besar penelitian sebelumnya masih menggunakan pendekatan single-label, sehingga tidak sepenuhnya menangkap keberagaman *mood* yang dapat muncul pada deskripsi buku maupun teks sastra [5].

Sejalan dengan itu, penelitian dalam domain emotion analysis modern menunjukkan bahwa pendekatan berbasis Transformer memberikan performa lebih baik dalam memahami konteks semantik dibandingkan metode klasik seperti TF-IDF dan SVM [7]. Namun demikian, algoritma tradisional seperti SVM dan Random Forest tetap menjadi baseline yang kuat dan kompetitif dalam berbagai studi multi-label, terutama ketika digunakan bersama teknik problem transformation seperti Binary Relevance [8].

Berdasarkan temuan-temuan tersebut, kajian pustaka ini memberikan landasan teoretis yang diperlukan untuk membandingkan performa algoritma Random Forest dan SVM dalam klasifikasi multi-label *mood* buku berbasis deskripsi teks menggunakan representasi TF-IDF dan pendekatan Binary Relevance.

2.1. Machine Learning

Machine Learning adalah cabang kecerdasan buatan yang memungkinkan komputer belajar dari data untuk mengenali pola dan membuat prediksi tanpa pemrograman eksplisit [9]. Dalam klasifikasi teks, algoritma machine learning memanfaatkan representasi fitur seperti TF-IDF untuk mengukur pentingnya kata dalam dokumen [10]. Berbagai algoritma seperti Naïve Bayes, Support Vector Machine, Decision Tree, dan Random Forest terbukti efektif untuk tugas klasifikasi dengan tingkat kompleksitas yang berbeda [11].

2.2. Sistem Rekomendasi

Sistem rekomendasi membantu pengguna memilih item sesuai preferensi mereka. Dalam konteks literatur, sistem rekomendasi berbasis *mood* dapat meningkatkan pengalaman membaca dengan menyesuaikan buku dengan kondisi emosional pembaca [3]. Penelitian menunjukkan bahwa pembaca cenderung memilih buku yang sejalan dengan suasana hati mereka karena memberikan kedalaman refleksi emosional [2].

2.3. *Mood* dalam Membaca

Mood merupakan keadaan afektif berdurasi lebih panjang namun berintensitas rendah, yang dapat memengaruhi perilaku dan preferensi membaca. Pembaca dengan *mood* positif cenderung memilih bacaan ringan, sedangkan *mood* negatif sering mendorong pemilihan bacaan reflektif dan emosional [4].

2.4. *Emotion Classifiers*

Klasifikasi emosi pada teks dilakukan melalui tahap praproses, ekstraksi fitur, dan klasifikasi [12]. Terdapat tiga pendekatan utama:

a. Lexicon-based, menggunakan kamus emosi seperti NRC dan WordNet-Affect.
b. Machine Learning-based, menggunakan algoritma SVM, Naïve Bayes, Decision Tree, Random Forest dengan representasi TF-IDF atau n-gram.
c. Deep Learning-based, menggunakan CNN, RNN, dan Transformer seperti BERT dan RoBERTa.

2.5. Representasi Teks (TF-IDF)

TF-IDF merupakan metode untuk menilai pentingnya kata dalam dokumen dengan mengkombinasikan frekuensi kata (TF) dan kelangkaannya dalam keseluruhan dokumen (IDF).
Rumus Perhitungan:

a. Term Frequency (TF):
$$TF(t,d) = \frac{f_{t,d}}{\sum_k f_{k,d}} \quad (1)$$
b. Inverse Document Frequency (IDF):
$$IDF(t) = \log\left(\frac{N}{df_t}\right) \quad (2)$$
c. TF-IDF
$$TF - IDF(t,d) = TF(t,d) \cdot IDF(t) \quad (3)$$

2.6. Multi-Label Classification

Klasifikasi multi-label memungkinkan satu sampel memiliki lebih dari satu label sekaligus. Salah satu pendekatan adalah Binary Relevance, yang mengubah setiap label menjadi tugas klasifikasi biner sehingga algoritma standar dapat digunakan untuk setiap label secara independen.
Rumus Perhitungan:

a. Untuk setiap label y_j, dibentuk dataset biner:
$$D_j = \{(x_i, \varphi(Y_i, y_j)) \mid 1 \le i \le m\} \quad (4)$$
b. Dengan fungsi label:
$$\varphi(Yi, yj) = \begin{cases} +1, & jika\ y_j \in Y_i \\ -1, & jika\ y_j \notin Y_i \end{cases} \quad (5)$$
c. Setiap D_j dilatih menggunakan algortima dasar B sehingga diperoleh *classifier* :
$$g_j \leftarrow B(D_j) \quad (6)$$
d. Prediksi untuk instance baru x dilakukan dengan memilih semua label yang memiliki keluaran positif:

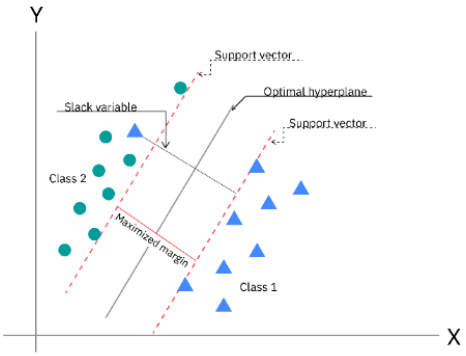
$$Y = \{y_j | g_j(x) > 0, 1 \leq j \leq q\} \quad (7)$$

e. Jika tidak ada label yang memenuhi, maka dipilih label dengan skor terbesar:

$$Y = \left\{ y_j \middle| g_j(x) > 0 \right\} \cup \{y_j * | j* = args \max_{1 \leq j \leq 1} g_j(x)\} \quad (8)$$

2.7. Support Vector Machine

SVM merupakan algoritma supervised learning yang mencari hyperplane terbaik untuk memisahkan kelas dengan margin maksimum [6].



Gambar 1. Hyperplane support vector machine

Secara umum, SVM terdiri dari dua jenis: SVM Linear, yang digunakan ketika data dapat dipisahkan secara linear, dan SVM Nonlinear, yang menerapkan *kernel trick* untuk memproyeksikan data ke ruang fitur berdimensi lebih tinggi agar pemisahan linear dapat dicapai.

Rumus Perhitungan:

a. Untuk sebuah model SVM linear, fungsi keputusan:

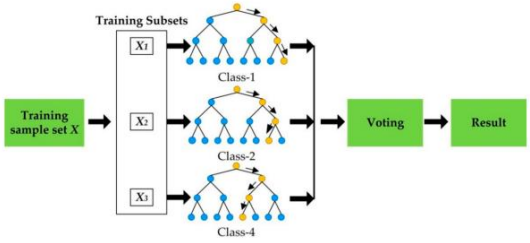
$$f(x) = w^T x + b \quad (8)$$

b. Dengan tujuan optimasi:

$$\min_{w,b} \frac{1}{2} \|w\|^2 \quad (9)$$

2.8. Random Forest

Random Forest adalah metode ensemble yang menggabungkan banyak decision tree untuk meningkatkan akurasi dan stabilitas prediksi [13]. etiap pohon dibangun melalui bootstrap sampling dan pemilihan fitur secara acak untuk meningkatkan keragaman model.



Gambar 2. Random forest

Secara umum, hasil prediksi diperoleh melalui mekanisme *majority voting*, yaitu memilih kelas yang paling banyak diprediksi oleh seluruh pohon dalam *forest*. Pendekatan ini membuat Random Forest lebih *robust* terhadap *overfitting* dan lebih stabil dibandingkan single decision tree.

Rumus Perhitungan:

a. Untuk sebuah model SVM linear, fungsi keputusan:

$$\check{y} = mode\{h_1(x), h_2(x), \ldots, h_k(x)\} \quad (10)$$

2.9. GridSearchCV

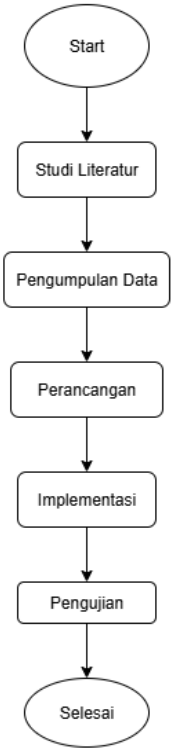
GridSearchCV digunakan untuk melakukan pencarian kombinasi hyperparameter terbaik dengan evaluasi melalui cross-validation [14]. Pada kasus multi-label, tuning hyperparameter seperti C pada SVM atau n_estimators pada Random Forest yang diperlukan untuk mengoptimalkan performa tiap classifier [15].

Rumus skor rata-rata dari k-fold cross-validation :

$$CV_{mean} = \frac{1}{k} \sum_{i=1}^{k} S_i \quad (11)$$

3. METODE PENELITIAN

Dalam klasifikasi *mood* buku yang dikembangkan dalam penelitian ini dapat digambarkan seperti berikut:



Gambar 3. Tahapan metodelogi penelitian

Metodologi penelitian pada klasifikasi *mood* buku ini terdiri atas lima tahapan utama beserta subproses yang menyusunnya. Rangkaian tahapan tersebut digambarkan sebagai berikut.

3.1. Studi Literatur

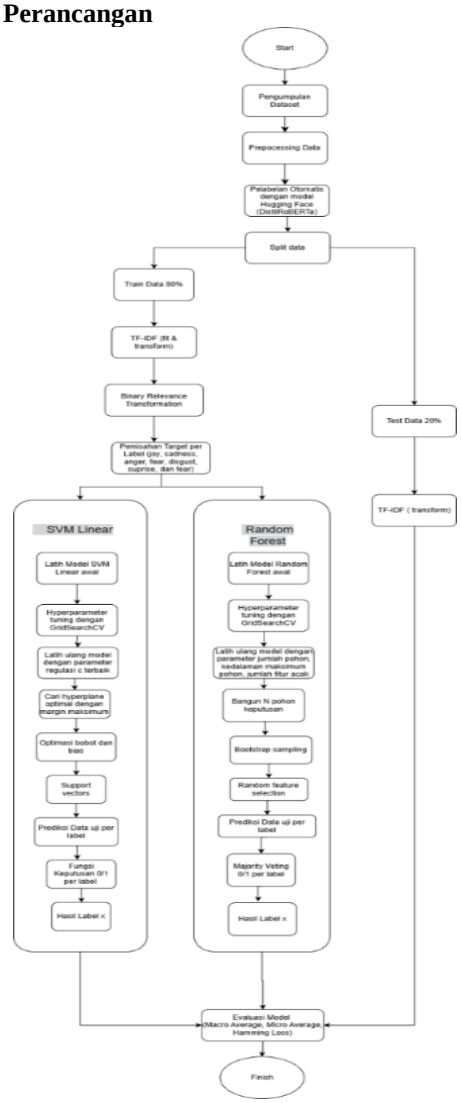
Tahap ini diawali dengan studi literatur terhadap jurnal dan buku yang membahas perbandingan

algoritma SVM dan Random Forest dalam klasifikasi multi-label berbasis deskripsi teks. Literatur ini mencakup teori sistem rekomendasi buku berbasis *mood*, representasi teks TF-IDF, dan pendekatan Binary Relevance, serta prinsip kerja dan parameter utama kedua algoritma. Selain itu, ditinjau pula penelitian terdahulu untuk memahami kelebihan, keterbatasan, dan metrik evaluasi yang digunakan. Hasil studi tersebut menjadi dasar perancangan penelitian ini.

3.2. Pengumpulan Dataset

Pada tahap ini dilakukan pengumpulan dan penyiapan data menggunakan *7k Books Dataset* dari Kaggle yang berisi metadata dan deskripsi buku. Dataset kemudian dilabeli secara otomatis menggunakan model Hugging Face *(michellejieli/emotion_text_classifier)* untuk menghasilkan tujuh kategori *mood*. Hasil pelabelan menghasilkan dataset *books_with_mood_7labels.csv* yang digunakan sebagai data utama untuk pelatihan dan pengujian model.

3.3. Perancangan



Gambar 4. Skema perancangan

Pada tahap *preprocessing*, dilakukan serangkaian proses pembersihan teks untuk memastikan bahwa data siap digunakan pada tahap pemodelan. Langkah-langkah yang diterapkan meliputi penghapusan nilai kosong dan duplikasi pada kolom deskripsi, *case folding* untuk penyeragaman huruf, *tokenization*, serta normalisasi teks. Selain itu, dilakukan pula koreksi kesalahan ketik menggunakan library Python seperti SpellChecker atau TextBlob untuk memperbaiki *typo* yang umum muncul. Proses normalisasi dan koreksi ini membantu menetralkan kata tidak baku, kata berulang, serta karakter nonrelevan agar tidak menjadi fitur TF-IDF yang dapat mengganggu performa model. Pembersihan karakter non-teks seperti emotikon dan simbol juga dilakukan untuk mengurangi *noise* sehingga representasi teks lebih bersih dan informatif sebelum memasuki tahap pembentukan fitur.. Hasil pelabelan menghasilkan dataset *books_with_mood_7labels.csv* yang digunakan sebagai data utama untuk pelatihan dan pengujian model.

Tahap berikutnya adalah pelabelan otomatis menggunakan model pra-latih dari Hugging Face, yaitu michellejieli/emotion_text_classifier yang berbasis arsitektur DistilRoBERTa. Model ini digunakan untuk memprediksi emosi pada setiap deskripsi buku berbahasa Inggris dan menghasilkan label multi-label berdasarkan tujuh kategori, yaitu *joy*, *sadness*, *anger*, *fear*, *disgust*, *surprise*, dan *neutral*. Tahap ini menghasilkan dataset baru dengan kolom label emosi yang kemudian digunakan sebagai dasar dalam proses klasifikasi.

Dataset selanjutnya dibagi menjadi 80% data latih dan 20% data uji. Representasi teks dilakukan menggunakan Term Frequency–Inverse Document Frequency (TF-IDF) dengan kombinasi unigram dan bigram untuk mengubah teks ke dalam bentuk vektor numerik. Pada data latih, dilakukan proses *fit* dan *transform* agar model mempelajari kosakata serta bobot IDF. Sementara itu, pada data uji hanya dilakukan *transform* menggunakan parameter hasil pembelajaran dari data latih untuk menjaga konsistensi dan menghindari *data leakage*.

Untuk menangani sifat multi-label pada data, penelitian ini menggunakan pendekatan *Problem Transformation* melalui teknik Binary Relevance. Pada teknik ini, setiap label emosi diubah menjadi tugas klasifikasi biner yang terpisah sehingga model hanya mempelajari apakah sebuah dokumen memiliki atau tidak memiliki suatu label tertentu. Pendekatan ini memungkinkan algoritma single-label seperti Support Vector Machine dan Random Forest digunakan tanpa modifikasi khusus pada arsitektur dasarnya. Dengan demikian, masalah multi-label yang kompleks dapat dipecah menjadi beberapa masalah klasifikasi sederhana yang lebih mudah dioptimalkan.

Penelitian ini menggunakan dua algoritma klasifikasi, yaitu SVM ber-kernel linear dan Random Forest. Pada SVM linear, proses pelatihan dilakukan dengan mencari *hyperplane* terbaik yang mampu

memisahkan dokumen positif dan negatif berdasarkan representasi TF-IDF. Model mengoptimalkan bobot dan bias untuk memaksimalkan margin pemisahan antar kelas, di mana dokumen terdekat dengan *hyperplane* berfungsi sebagai *support vectors*. Sebaliknya, pada Random Forest, proses pelatihan dilakukan dengan membangun banyak pohon keputusan menggunakan *bootstrap sampling* serta pemilihan subset fitur secara acak pada setiap percabangan, sehingga setiap pohon memiliki karakteristik yang unik. Prediksi akhir diperoleh melalui mekanisme *majority voting* antar pohon untuk menentukan apakah suatu dokumen termasuk ke dalam kategori label tertentu.

Untuk meningkatkan performa model, penelitian ini menerapkan GridSearchCV sebagai metode optimasi hiperparameter. Teknik ini dipilih karena jumlah parameter yang diuji relatif terbatas dan ruang pencarian terkontrol, sehingga grid search dapat bekerja secara efisien. Pada algoritma SVM, parameter yang dioptimalkan adalah nilai regulasi C untuk menyeimbangkan trade-off antara margin dan kesalahan klasifikasi. Sedangkan pada Random Forest, parameter seperti jumlah pohon, kedalaman maksimum pohon, dan jumlah fitur pada setiap node dievaluasi untuk meningkatkan stabilitas dan kemampuan generalisasi model pada masing-masing label *mood*.

3.4. Implementasi

Pada tahapan ini dilakukan implementasi dari sistem yang telah dirancang sebelumnya agar sistem dapat mengenali dan melakukan klasifikasi terhadap data *training* yang telah diolah sebelumnya.

3.5. Pengujian

Pada tahap ini, sistem yang telah dibangun diuji menggunakan data uji untuk memperoleh performa model. Setelah proses pengujian dilakukan, tingkat keberhasilan metode dievaluasi menggunakan metrik Macro-averaging, Micro-averaging, dan Hamming Loss.

Rumus Perhitungan:

a. Macro-averaging :

$$B_{macro}(h) = \frac{1}{q}\sum_{j=1}^{q} B(TP_J, FP_J, TN_{J,} FN_J) \quad (12)$$

b. Micro-averaging :

$$B_{micro}(h) = B\left(\sum_{j=1}^{q} TP_J, \sum_{j=1}^{q} FP_J, \sum_{j=1}^{q} TN_J, \ \sum_{j=1}^{q} FN_J\right) (13)$$

c. Hamming Loss:

$$Hamming\ Loss(h) = \frac{1}{P}\sum_{i=1}^{P} |h(x_i)\Delta Yi| \quad (14)$$

4. HASIL DAN PEMBAHASAN

4.1. Pengambilan Data

Pengambilan Pengambilan data dilakukan menggunakan dataset *7k Books* dari Kaggle. Dari 12 atribut awal, penelitian ini hanya menggunakan *title* dan *description*, di mana deskripsi menjadi sumber fitur teks dan judul sebagai informasi pendukung. Setiap deskripsi dilabeli otomatis menggunakan model pra-latih DistilRoBERTa menjadi tujuh kategori mood, yaitu *joy*, *sadness*, *anger*, *fear*, *disgust*, *surprise*, dan *neutral*. Hasilnya disimpan dalam file *books_with_mood_7labels.csv*, yang digunakan sebagai dataset utama untuk pelatihan dan pengujian model klasifikasi multi-label.

Tabel 1. Dataset books with mood 7labels

No.	title	description	mood
1.	The One Tree	Volume Two of Stephen Donaldson's acclaimed second trilogy featuing the compelling anti-hero Thomas Covenant.	['joy', 'neutral']
2.	Master of the Game	Kate Blackwell is an enigma and one of the most powerful women in the world. But at her ninetieth birthday celebrations there are ghosts of absent friends and absent enemies.	['sadness', 'neutral']
3.	Murder in LaMut	Available in the U.S. for the first time, this is the second volume in the exceptional Legends of the Riftwar series from "New York Times"-bestselling authors Feist and Rosenberg.	['joy', 'neutral']
4.	Mystical Paths	1968 finds Nicholas Darrow wrestling with personal problems. How can he marry Rosalind when he is unable to avoid promiscuity? How can he become a priest when he finds it so difficult to live as one? And can he break his dangerous dependence on his father?	['surprise', 'neutral']
5.	Miss Marple	Miss Marple featured in 20 short stories, published in a number of different collections in Britain and America. Presented here in their order of publication, Miss Marple uses her unique insight to deduce the truth about a series of unsolved crimes.	['joy', 'neutral']

Berdasarkan tabel 1, dataset ini memuat tiga kolom utama, yaitu *title*, *description*, dan *mood*. Kolom *title* berisi judul buku, kolom *description* berisi ringkasan isi buku, dan kolom *mood* berisi label kategori emosi hasil pelabelan otomatis. Kolom-kolom ini menjadi dasar bagi proses pelatihan dan evaluasi model klasifikasi multi-label.

4.2. Pra-Pemrosesan Data

Pengambilan Tahap pra-proses dilakukan untuk memastikan kualitas data sebelum dimasukkan ke dalam model. Langkah-langkah yang dilakukan meliputi:

a. Pembersihan teks, dengan menghapus karakter non-alfabet, URL, dan whitespace berlebih.
b. Tokenisasi dan *case folding*, yaitu memecah teks menjadi token kata dan menyamakan huruf menjadi lowercase.
c. Koreksi *typo*, dengan menggunakan *library* SpellChecker untuk memperbaiki kesalahan ketik pada deskripsi buku.

```
=== Statistik Sebelum Preprocessing ===
count    6810.000000
mean       64.214684
std        66.604150
min         1.000000
25%        26.000000
50%        39.000000
75%        79.000000
max       920.000000
Name: len_before, dtype: float64

=== Statistik Setelah Preprocessing ===
count    6810.000000
mean       65.464317
std        67.794383
min         0.000000
25%        26.000000
50%        39.500000
75%        80.000000
max       930.000000
Name: len_after, dtype: float64

Jumlah token unik sebelum preprocessing: 60396
Jumlah token unik setelah preprocessing : 26976
```

Gambar 5. Statistik prepocessing

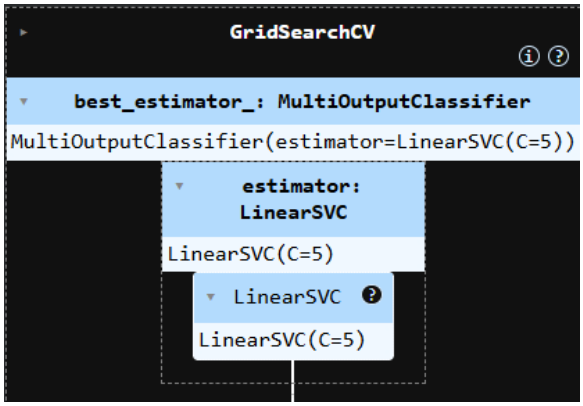
Gambar 5 menyajikan statistik deskriptif panjang teks sebelum dan sesudah pra-pemrosesan, yang mencakup nilai *count* (jumlah data), *mean* (rata-rata panjang teks), *std* (variasi panjang teks), *min* dan *max* (panjang terpendek dan terpanjang), serta kuartil 25%, 50% (median), dan 75% yang menggambarkan sebaran data pada tiga titik pembagi. Statistik ini digunakan untuk menilai pengaruh preprocessing terhadap distribusi teks, dan hasilnya menunjukkan bahwa distribusi panjang teks tetap stabil. Kondisi ini penting dalam konteks multilabel karena kualitas dan konsistensi teks berpengaruh langsung terhadap representasi fitur TF-IDF dan performa model klasifikasi. Setelah pra-proses, dataset dibagi menjadi 80% data latih dan 20% data uji, kemudian setiap deskripsi direpresentasikan dalam bentuk vektor TF-IDF dengan kombinasi unigram dan bigram untuk menangkap konteks kata.

4.3. Pembuatan Model

Penelitian ini menggunakan pendekatan multi-label dengan Binary Relevance, di mana setiap label mood diperlakukan sebagai tugas klasifikasi biner terpisah. Pendekatan ini memungkinkan setiap label dipelajari secara independen, sehingga pola kata yang relevan untuk satu mood tidak terpengaruh oleh label lain. Implementasi Binary Relevance dilakukan menggunakan MultiOutputClassifier, yang membungkus *classifier* dasar seperti Linear Support Vector Machine atau Random Forest. Setiap *classifier* dasar dilatih untuk memprediksi satu label mood secara biner (1 = ada *mood*, 0 = tidak ada *mood*). Output dari semua *classifier* digabungkan menjadi vektor multi-label untuk setiap buku.

4.4. Support Vector Machine

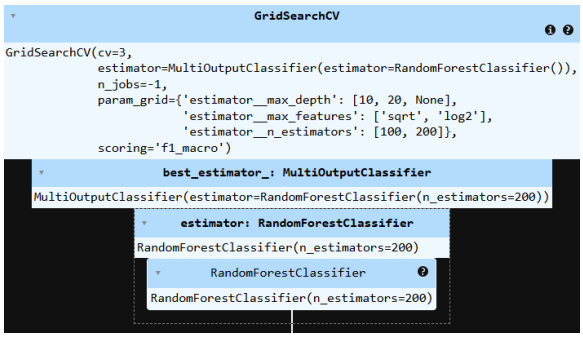
Linear SVM digunakan untuk mencari *hyperplane* optimal yang dapat memisahkan data positif dan negatif pada setiap label. Parameter regulasi, seperti **C**, dioptimalkan menggunakan GridSearchCV untuk meningkatkan kemampuan generalisasi model.



Gambar 6. Model support vector machine

Gambar ini menampilkan hasil optimasi parameter C pada Linear SVM menggunakan GridSearchCV. Dari proses ini diperoleh best_estimator_ berupa MultiOutputClassifier dengan LinearSVC(C=5) sebagai base estimator. Estimator ini kemudian digunakan untuk prediksi multi-label pada data uji. Output menunjukkan bahwa nilai C terbaik adalah 5, yang digunakan untuk memaksimalkan performa klasifikasi pada semua label *mood*.

Random Forest, membangun sejumlah pohon keputusan melalui bootstrap sampling dan pemilihan fitur secara acak. Parameter seperti jumlah pohon (n_estimators), kedalaman pohon (max_depth), dan jumlah fitur pada setiap node (max_features) juga dioptimalkan menggunakan GridSearchCV.



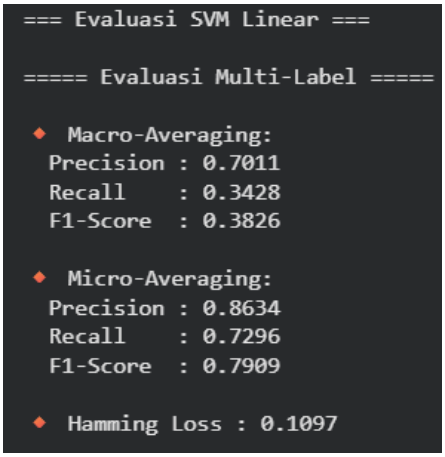
Gambar 7. Model random forest

Gambar ini menampilkan hasil optimasi parameter pada Random Forest menggunakan GridSearchCV. Dari proses ini diperoleh best_estimator_ berupa MultiOutputClassifier dengan RandomForestClassifier(n_estimators=200)

sebagai base estimator. Estimator ini kemudian digunakan untuk prediksi multi-label pada data uji. Output menunjukkan bahwa jumlah pohon terbaik adalah 200, yang digunakan untuk memaksimalkan performa klasifikasi pada semua label *mood*.

4.5. Evaluasi SVM Linear

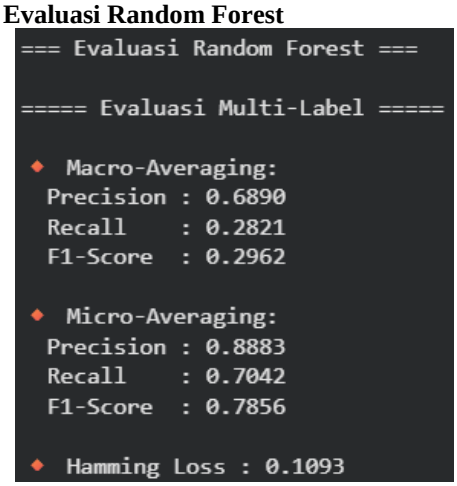
Evaluasi kinerja algoritma Linear SVM dilakukan untuk memprediksi multi-label mood pada deskripsi buku. Model dievaluasi menggunakan metrik macro-averaging, micro-averaging, dan Hamming Loss, yang dapat menunjukkan performa model secara keseluruhan maupun per label.



Gambar 8. Hasil evaluasi svm

Model Linear SVM memiliki precision macro 0,7011, recall 0,3428, dan F1-score 0,3826, menunjukkan rata-rata performa per label yang belum merata. Micro-averaging menghasilkan precision 0,8634, recall 0,7296, dan F1-score 0,7909, menandakan performa keseluruhan model cukup baik secara agregat. Hamming Loss sebesar 0,1097 menunjukkan kesalahan prediksi rata-rata sekitar 10,97%. Hasil ini menunjukkan bahwa Linear SVM mampu memprediksi *mood* buku dengan andal secara keseluruhan, meskipun performa pada beberapa label tertentu masih dapat ditingkatkan.

4.6. Evaluasi Random Forest



Gambar 9. Hasil evaluasi random forest

Model Random Forest menunjukkan precision macro 0,6890, recall 0,2821, dan F1-score 0,2962, yang mencerminkan performa rata-rata per label yang masih rendah pada beberapa kategori mood. Micro-averaging menghasilkan precision 0,8883, recall 0,7042, dan F1-score 0,7856, menandakan performa keseluruhan model cukup baik secara agregat. Hamming Loss sebesar 0,1093 menunjukkan kesalahan prediksi rata-rata sekitar 10,93%. Hasil ini menunjukkan bahwa Random Forest mampu memprediksi *mood* buku secara keseluruhan dengan andal, meskipun beberapa label tertentu masih memerlukan peningkatan performa.

5. KESIMPULAN DAN SARAN

Penelitian ini membandingkan dua algoritma, yakni Linear SVM dan Random Forest, dalam memprediksi mood buku pada dataset multi-label. Hasil evaluasi menunjukkan bahwa Linear SVM memiliki performa yang sedikit lebih baik secara keseluruhan dengan micro-averaging F1-score 0,7909 dibandingkan Random Forest 0,7856, meskipun kedua algoritma menunjukkan macro-averaging yang rendah, mengindikasikan prediksi pada beberapa label masih belum seimbang. Perbedaan performa ini kemungkinan disebabkan oleh karakteristik algoritma dan ketidakseimbangan label dalam dataset. Sebagai saran, penelitian selanjutnya dapat menerapkan teknik penyeimbangan kelas, optimasi hyperparameter lebih lanjut, atau metode ensemble learning, serta memperluas jumlah dataset agar evaluasi menjadi lebih representatif dan akurat dalam menangani multi-label *mood* buku.

DAFTAR PUSTAKA

[1] K. N. bin Z. Badri, “Factors that Promote Reading Culture and Its Impact on Society,” *JSSH: Jurnal Sains Sosial dan Humaniora*, vol. 6, pp. 35–44, 2022, doi: 10.30595/jssh.v6i1.13301.

[2] N. Currie, K. Wilkinson, and S. McGeown, “Reading Fiction and Psychological Well-being During Older Adulthood Positive.pdf,” 2025, *Reading Fiction and Psychological Well- being During Older Adulthood: Positive Affect, Connection and Personal Growth*. doi: 10.1002/rrq.605.

[3] Sarwath Unnisa and Akshaya N S, “Personalized Mood-Centric Book Recommendation Integrating Machine Learning with Content Based Filtering,” *International Journal of Information Technology, Research and Applications*, vol. 3, no. 3, pp. 15–22, 2024, doi: 10.59461/ijitra.v3i3.100.

[4] M. Naranowicz, “Mood effects on semantic processes: Behavioural and electrophysiological evidence,” *Front Psychol*, vol. 13, no. November, pp. 1–18, 2022, doi: 10.3389/fpsyg.2022.1014706.

[5] J. Wong, I. Sanu, and H. Irsyad, "Implementasi TF-IDF, Cosine Similarity, dan Logistic Regression Pada Rekomendasi Buku Berdasarkan Mood Pembaca Dengan Data Oversampling," *Device : Journal of Information System, Computer Science and Information Technology*, vol. 6, no. 1, pp. 142–154, 2025, doi: 10.46576/device.v6i1.6499.

[6] B. Falakhi, I. Cholissodin, and R. S. Perdana, "Klasifikasi Sinopsis Novel berdasarkan Jenis Genre menggunakan Multi-class Support Vector Machine dan Chi-square," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 1, pp. 192–202, 2023, [Online]. Available: http://j-ptiik.ub.ac.id

[7] Z. Sarif, M. S. Akhtar, A. Das, and D. Das, "Trans-Sent at SemEval-2025 Task 11: Text-based Multi-label Emotion Detection using Pre-Trained BERT Transformer Models," *Association for Computational Linguistics*, vol. Proceeding, pp. 893–898, 2025, [Online]. Available: https://aclanthology.org/2025.semeval-1.121/

[8] D. Kher and K. Passi, "Multi-label Emotion Classification using Machine Learning and Deep Learning Methods," *International Conference on Web Information Systems and Technologies, WEBIST - Proceedings*, vol. 2022-Octob, no. Webist, pp. 128–135, 2022, doi: 10.5220/0011532400003318.

[9] E. I. Sukmana and L. Nilawati, "Penerapan Machine Learning Pada Sistem Informasi Klasifikasi Informasi Penggalian Potensi Pajak," *Jutisi : Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, vol. 13, no. 3, pp. 2137–2148, 2024, doi: 10.35889/jutisi.v13i3.2323.

[10] A. Firizkiansah, A. Muhammad, and I. R. Maulana, "Optimasi Klasifikasi Data Teks Menggunakan Algoritma Logistic Regression dengan TF-IDF dan SMOTE," *Jurnal Ilmiah Ilmu Komputer dan Teknologi Informasi*, vol. 2, no. 1, pp. 3089–2996, 2025, [Online]. Available: https://ojs.sains.ac.id/index.php/Jikomti/article/download/97/119/355

[11] K. Machová, M. Szabóova, J. Paralič, and J. Mičko, "Detection of emotion by text analysis using machine learning," *Front Psychol*, vol. 14, pp. 1–14, 2023, doi: 10.3389/fpsyg.2023.1190326.

[12] S. Kusal, S. Patil, J. Choudrie, K. Kotecha, D. Vora, and I. Pappas, "A Review on Text-Based Emotion Detection -- Techniques, Applications, Datasets, and Future Directions," 2022, [Online]. Available: http://arxiv.org/abs/2205.03235

[13] M. R. Kurniawan, O. N. Pratiwi, and F. Hamami, "Klasifikasi Soal Menggunakan Multi-Label Problem Transformation Dengan Metode Random Forest Dan K-Nearest Neighbor," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 10, no. 1, pp. 367–380, 2025, doi: 10.29100/jipi.v10i1.5910.

[14] C. G. Siji George and B. Sumathi, "Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, pp. 173–178, 2020, doi: 10.14569/IJACSA.2020.0110920.

[15] E. C. Pratama, "Jurnal JPILKOM (Jurnal Penelitian Ilmu Komputer) Optimasi Algoritma Decision Tree dalam Memprediksi Penyakit Diabetes Menggunakan GridSearchCV," vol. 2, no. 4, pp. 187–196, 2025, [Online]. Available: https://jpilkom.org