ISSN:2302-8149
e-ISSN:2540-9719

Klasifikasi Genre Novel berdasarkan Sinopsis menggunakan Algoritma *Random Forest*

Novel Genre Classification based on Synopsis using the Random Forest Algorithm

[1]Prananing Mahanani*, [2]Charitas Fibriani
[1,2]Program Studi Sistem Informasi, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana
[1,2]Jl. Dr. O. Notohamidjojo No. 1-10, Blotongan, Sidorejo, Salatiga, Jawa Tengah 501715 Indonesia
*e-mail: charitas.fibriani@uksw.edu

(***received*:** 29 October 2025, ***revised*:** 27 November 2025, ***accepted*:** 28 November 2025)

Abstrak

Klasifikasi genre novel berdasarkan sinopsis merupakan tantangan utama dalam pengolahan teks karena setiap genre memiliki karakteristik kosakata yang unik. Penelitian ini mengevaluasi kinerja algoritma Random Forest dalam mengklasifikasikan genre novel pada kondisi distribusi data yang tidak seimbang. Tahapan penelitian meliputi *preprocessing* (case folding, tokenisasi, stopword removal, dan stemming), ekstraksi fitur menggunakan TF-IDF, dan pelatihan model Random Forest. Selain itu, dilakukan penyeimbangan data secara manual dengan menambah sampel pada kelas minoritas melalui *oversampling* sederhana. Model diuji menggunakan metrik akurasi dan analisis *confusion matrix*. Hasil menunjukkan bahwa Random Forest mampu mengidentifikasi sebagian besar genre dengan tingkat ketepatan sedang, terutama pada kelas dengan jumlah data lebih besar. Akurasi model awal sebesar 42,11% meningkat menjadi 46,67% setelah balancing diterapkan. Kesalahan prediksi terutama terjadi pada genre dengan jumlah sampel terbatas yang memiliki kemiripan kosakata dengan genre dominan. Hasil penelitian ini memperlihatkan bahwa Random Forest tetap dapat diterapkan untuk klasifikasi genre novel berbasis sinopsis tanpa bergantung sepenuhnya pada teknik balancing. Performa belum merata pada seluruh kelas sehingga analisis per-genre diperlukan untuk memperoleh gambaran evaluasi yang lebih menyeluruh.

Kata kunci: klasifikasi, data tidak seimbang, genre novel, random forest

Abstract

Novel genre classification based on synopses presents a significant challenge in text processing, as each genre exhibits distinct lexical characteristics. This study evaluates the performance of the Random Forest algorithm in classifying novel genres under conditions of imbalanced data distribution. The research stages include text preprocessing—comprising case folding, tokenization, stopword removal, and stemming—feature extraction using Term Frequency–Inverse Document Frequency (TF-IDF), and model training with Random Forest. In addition, manual data balancing was applied by increasing samples in minority classes through simple oversampling. The model was evaluated using accuracy metrics and confusion matrix analysis. The results indicate that Random Forest is able to identify most genres with moderate accuracy, particularly for classes with larger data volumes. The initial model achieved an accuracy of 42.11%, which increased to 46.67% after the application of data balancing. Misclassification primarily occurred in genres with limited samples that share similar vocabulary with dominant genres. These findings demonstrate that Random Forest can still be applied to synopsis-based novel genre classification without fully relying on balancing techniques. However, performance remains uneven across classes, highlighting the need for per-genre analysis to obtain a more comprehensive evaluation.

Keywords: *classification, imbalanced data, novel genre, random forest*

1 Pendahuluan

Perkembangan platform literasi digital mendorong peningkatan jumlah karya sastra yang tersedia secara daring. Pertumbuhan ini menuntut adanya sistem pencarian dan rekomendasi yang dapat

 ISSN:2302-8149
e-ISSN:2540-9719

mengelompokkan novel secara otomatis berdasarkan genre melalui teks sinopsis. Pendekatan ini penting karena sinopsis memuat informasi inti mengenai tema, alur, dan karakteristik cerita yang membedakan satu genre dengan genre lainnya, sehingga dapat mendukung personalisasi rekomendasi dan keterjangkauan informasi literasi digital [1].

Klasifikasi berbasis teks menghadapi tantangan utama pada distribusi kelas yang tidak merata. Beberapa genre memiliki jumlah data yang jauh lebih banyak dibandingkan genre lain, sehingga algoritma pembelajaran mesin cenderung mengenali pola kelas mayoritas dengan baik tetapi kesulitan membedakan karakteristik kosakata kelas minoritas. Kondisi ini berdampak pada penurunan precision dan recall pada genre minoritas meskipun akurasi keseluruhan tampak tinggi [2][3].

Random Forest (RF) merupakan salah satu algoritma yang banyak digunakan untuk klasifikasi berbasis teks karena mekanisme ensemble-nya yang mampu mengurangi overfitting dan menghasilkan prediksi yang stabil pada data berdimensi tinggi seperti TF-IDF. Penelitian terkini menunjukkan bahwa Random Forest mampu mempertahankan akurasi dan ketahanan prediksi pada kondisi ketidakseimbangan kelas melalui proses bagging serta pemilihan fitur acak dalam pembentukan pohon keputusan [4][5].

Rumusan masalah dalam penelitian ini berfokus pada bagaimana melakukan klasifikasi genre novel dengan kelas tidak seimbang menggunakan algoritma Random Forest. Penelitian ini diharapkan dapat memberikan kontribusi signifikan terhadap penerapan Random Forest dalam klasifikasi genre novel pada dataset nyata yang memiliki distribusi kelas tidak merata. Selain itu, hasil penelitian ini diharapkan memperkuat temuan sebelumnya mengenai efektivitas Random Forest dalam menangani ketidakseimbangan kelas tanpa penerapan teknik penyeimbangan tambahan.

2 Tinjauan Literatur

Klasifikasi teks merupakan pendekatan NLP untuk mengelompokkan dokumen berdasarkan pola linguistik. Pada domain literasi digital, sinopsis novel terbukti mengandung informasi tematik yang memadai untuk mengidentifikasi genre secara otomatis menggunakan pembelajaran mesin. Representasi sinopsis mendukung klasifikasi genre buku dan meningkatkan akurasi rekomendasi bacaan [1]. Pada konteks yang lebih spesifik, klasifikasi genre berbasis sinopsis terbukti efektif dalam mengelompokkan novel ke dalam kategori seperti romance, thriller, dan fantasy dengan memanfaatkan pemodelan teks menggunakan algoritma pembelajaran mesin [6].

Distribusi kelas yang tidak merata (imbalanced) merupakan masalah umum pada corpus teks, termasuk pada kategori novel. Model pembelajaran mesin cenderung mempelajari kosakata kelas mayoritas dan mengabaikan pola kosakata kelas minoritas. Ketidakseimbangan ini menghasilkan akurasi keseluruhan yang tinggi, tetapi kinerja pada kelas minoritas rendah [2][3].

TF-IDF menonjolkan kata-kata yang memiliki makna spesifik pada suatu dokumen tetapi jarang muncul pada keseluruhan koleksi dokumen. Representasi ini stabil dan efisien pada dataset berdimensi tinggi baik seimbang maupun tidak seimbang [5]. Penggunaan TF-IDF tetap relevan pada domain klasifikasi berbasis sinopsis novel.

Random Forest merupakan algoritma ensemble berbasis bagging dan pemilihan fitur acak yang memberikan performa konsisten pada berbagai tugas NLP. Penelitian terkini menunjukkan bahwa Random Forest mempertahankan akurasi dan stabilitas prediksi pada kondisi distribusi kelas tidak seimbang melalui pembelajaran berbasis banyak pohon keputusan [5]. Model Random Forest tetap memiliki bias terhadap kelas mayoritas ketika distribusi kelas sangat timpang. Jumlah pohon keputusan yang terlatih pada kelas minoritas menjadi jauh lebih sedikit sehingga recall menurun meskipun akurasi keseluruhan terlihat tinggi[2][4].

Berbagai pendekatan balancing seperti oversampling, undersampling, dan pembobotan kelas telah diterapkan untuk meningkatkan sensitivitas model terhadap kelas minoritas. Penelitian melaporkan bahwa pendekatan tersebut dapat meningkatkan performa, tetapi dapat menimbulkan risiko hilangnya informasi atau peningkatan waktu pelatihan [3]. Efektivitas balancing pada klasifikasi genre novel berbasis sinopsis masih memerlukan kajian lebih mendalam.

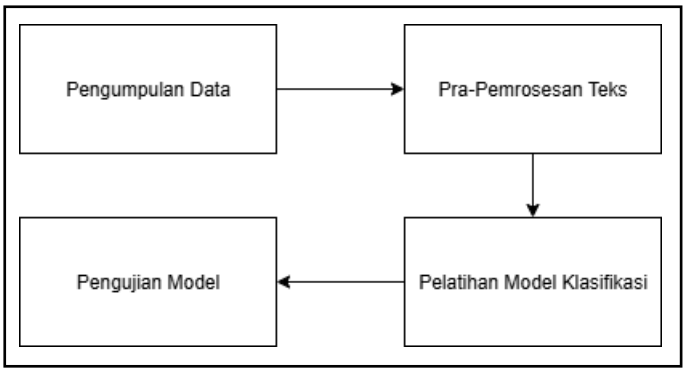
Penelitian terdahulu pada klasifikasi teks umumnya berfokus pada peningkatan performa melalui teknik balancing, seperti oversampling, undersampling, atau pembobotan kelas. Pendekatan tersebut dilaporkan mampu meningkatkan recall pada kelas minoritas, tetapi juga berpotensi mengubah distribusi data asli atau menambah kompleksitas komputasi[3]. Literatur menunjukkan bahwa

 ISSN:2302-8149
e-ISSN:2540-9719

Random Forest memiliki stabilitas prediksi yang baik pada berbagai karakteristik dataset, termasuk kondisi ketidakseimbangan kelas, namun temuan tersebut sebagian besar diperoleh pada studi yang tetap menggunakan balancing tambahan untuk memperbaiki distribusi data [2]. Kondisi ini menyisakan ruang penelitian untuk mengevaluasi performa Random Forest secara langsung pada dataset sinopsis novel tanpa penerapan balancing untuk menentukan sejauh mana model mampu menangani ketidakseimbangan kelas berdasarkan distribusi data asli.

3 Metode Penelitian

Penelitian ini bertujuan untuk mengevaluasi kemampuan algoritma Random Forest (RF) dalam mengklasifikasikan genre novel berdasarkan teks sinopsis menggunakan dataset asli yang memiliki distribusi kelas tidak seimbang. Pendekatan penelitian ini bersifat kuantitatif eksperimental, di mana model pembelajaran mesin dilatih dan diuji untuk memperoleh hasil kinerja secara empiris. Proses penelitian terdiri dari empat tahap utama, yaitu pengumpulan data, pra-pemrosesan teks, pelatihan model klasifikasi, dan pengujian model. Alur keseluruhan penelitian ditunjukkan pada Gambar 1.



Gambar 1 Alur penelitian

3.1 Pengumpulan Data

Dataset yang digunakan dalam penelitian ini diperoleh dari platform Kaggle dengan judul *Book Genre Prediction using NLP* yang dibuat oleh Prathamesh Gadekar (2022). Dataset ini berisi sebanyak 4.657 baris data dan terdiri dari empat kolom utama, yaitu *index*, *title*, *genre*, dan *summary*. Kolom *index* berisi nomor urut unik untuk setiap entri data, kolom *title* berisi judul novel dalam bentuk teks, kolom *genre* menunjukkan kategori atau jenis novel seperti *fantasy*, *romance*, atau *mystery*, sedangkan kolom *summary* berisi sinopsis atau ringkasan isi novel dalam bentuk teks panjang.

Setiap baris dalam dataset merepresentasikan satu novel dengan informasi lengkap mengenai judul, genre, dan isi sinopsisnya. Kolom *genre* digunakan sebagai variabel target dalam proses klasifikasi, sementara kolom *summary* menjadi fitur utama yang diolah menggunakan metode TF-IDF untuk menghasilkan representasi numerik dari teks. Penelitian ini mengambil subset sebanyak 1.000 data untuk keperluan pelatihan dan pengujian model dari total 4.657 data yang tersedia.

3.2 Pra-Pemrosesan Data

Tahap pra-pemrosesan dilakukan untuk mengubah data teks pada kolom *summary* menjadi representasi numerik yang dapat dipahami oleh algoritma *machine learning*. Proses ini melibatkan beberapa langkah utama, yaitu pembersihan teks, tokenisasi, penghapusan *stopwords*, *stemming*, dan pembobotan kata menggunakan metode TF-IDF.

Proses tokenisasi dilakukan dengan fungsi unnest_tokens() dari library tidytext untuk memecah setiap sinopsis novel menjadi kumpulan kata tunggal. Tahapan ini bertujuan untuk mengubah setiap dokumen teks $D_i$menjadi sekumpulan token sehingga teks dapat direpresentasikan dalam bentuk *bag of words* dengan rumus (1):

$$w_1, w_2, \dots, w_n \quad (1)$$

Langkah setelah tokenisasi, dilakukan penghapusan *stopwords* menggunakan fungsi anti_join(stop_words) untuk menghilangkan kata-kata umum seperti *the*, *is*, dan *and* yang tidak memiliki makna informatif dalam proses klasifikasi. Secara matematis, proses ini dapat digambarkan dengan rumus (2):

 ISSN:2302-8149
e-ISSN:2540-9719

$$W' = W - S \quad (2)$$

W adalah himpunan semua kata dalam dokumen dan S adalah himpunan kata-kata yang termasuk dalam daftar *stopwords*.

Langkah berikutnya adalah proses *stemming* menggunakan fungsi wordStem() dari library SnowballC, yang bertujuan untuk mengembalikan setiap kata ke bentuk dasarnya. Contohnya, kata *running*, *runs*, dan *runner* akan diubah menjadi *run* agar model tidak memperlakukan ketiganya sebagai kata berbeda.

Setelah teks dibersihkan dan dinormalisasi, dilakukan pembobotan menggunakan metode Term Frequency–Inverse Document Frequency (TF-IDF) dengan fungsi bind_tf_idf(). Pembobotan ini digunakan untuk mengukur seberapa penting sebuah kata terhadap suatu dokumen dibandingkan dengan seluruh dokumen dalam dataset. Rumus *Term Frequency* (TF) dapat digambarkan dengan rumus (3):

$$TF(t,d) = \frac{tf}{\max(tf)} \quad (3)$$

Nilai *t f* menunjukkan frekuensi kemunculan term t dalam dokumen d, dan $\max(tf)$ merupakan jumlah kemunculan term terbanyak dalam dokumen yang sama.

Inverse Document Frequency (IDF) merupakan ukuran yang menunjukkan tingkat kekhususan suatu term di seluruh dokumen dalam dataset. Bobot IDF akan bernilai tinggi ketika sebuah term jarang muncul, dan bernilai rendah ketika term tersebut muncul di banyak dokumen. Nilai IDF dihitung menggunakan persamaan (4):

$$IDF(t) = \log\left(\frac{N}{df_t}\right) \quad (4)$$

Pada persamaan tersebut, N menyatakan jumlah keseluruhan dokumen dalam dataset, sedangkan df_t menyatakan jumlah dokumen yang mengandung term t. $IDF(t)$ menggambarkan bobot *inverse document frequency* untuk term t, yaitu seberapa penting term tersebut dalam proses pembobotan berdasarkan tingkat kemunculannya pada seluruh dokumen.

3.3 Pelatihan Model

Pelatihan model dilakukan menggunakan Random Forest (RF) sebagai model utama, yang diimplementasikan dengan library randomForest di R. Random Forest dipilih karena kemampuannya menangani ketidakseimbangan kelas secara alami dan mengurangi overfitting.

Pada Random Forest, setiap pohon keputusan dilatih pada subset acak data melalui mekanisme bagging, dan subset fitur juga dipilih secara acak untuk setiap split. Prediksi akhir untuk setiap dokumen ditentukan melalui voting mayoritas antar pohon, yang secara matematis dapat ditunjukkan pada rumus (5)

$$\hat{y} = mode\{h_1(x), h_2(x), \ldots, h_n(x)\} \quad (5)$$

di mana $\hat{y}$ adalah prediksi akhir, $h_i(x)$adalah prediksi pohon ke-i, dan n adalah jumlah pohon dalam hutan.

Pra-pemrosesan teks yang meliputi tokenisasi, penghapusan stopwords, stemming, dan perhitungan TF-IDF dilakukan dengan memanfaatkan library tidytext, dplyr, tidyr, SnowballC, dan stringr, sehingga setiap dokumen direpresentasikan dalam bentuk vektor numerik yang konsisten untuk pelatihan model.

3.4 Pengujian Model

Tahap pengujian dilakukan untuk mengevaluasi kemampuan model dalam mengklasifikasikan genre novel berdasarkan data yang belum pernah digunakan selama pelatihan. Pengujian model bertujuan menilai efektivitas algoritma dalam mengenali pola teks dan menentukan sejauh mana hasil klasifikasi dapat digeneralisasikan terhadap data baru.

Proses pengujian dilakukan dengan membagi dataset menjadi dua bagian, yaitu 80% data pelatihan (*training data*) dan 20% data pengujian (*testing data*). Pembagian dilakukan secara acak menggunakan fungsi createDataPartition() dari library caret agar setiap kelas genre tetap terdistribusi proporsional. Data pelatihan digunakan untuk membangun model, sedangkan data pengujian digunakan untuk menguji performa model terhadap sampel yang belum pernah dipelajari sebelumnya.

 ISSN:2302-8149
e-ISSN:2540-9719

Evaluasi model dilakukan menggunakan fungsi confusionMatrix() dari library caret, yang menghasilkan berbagai metrik pengujian seperti Accuracy, F1-Score, dan Cohen's Kappa. Metrik-metrik ini digunakan untuk menilai ketepatan prediksi model dan kemampuan algoritma dalam mengklasifikasikan kelas minoritas maupun mayoritas.

Nilai akurasi dihitung dengan rumus (6):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$

TP= True Positive, *TN= True Negative*, *FP= False Positive*, dan *FN= False Negative*.

Nilai F1-Score digunakan untuk menilai keseimbangan antara *precision* dan *recall*, dengan rumus (7):

$$F1 = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \quad (7)$$

Cohen's Kappa digunakan untuk mengukur tingkat kesepakatan antara hasil prediksi model dan label sebenarnya, dengan rumus (8):

$$\kappa = \frac{p_o - p_e}{1 - p_e} \quad (8)$$

Nilai $p_o$adalah akurasi observasi dan $p_e$adalah akurasi yang diharapkan secara acak. Evaluasi ini menjadi dasar untuk menilai efektivitas algoritma Random Forest.

4 Hasil dan Pembahasan (or Results and Analysis)

4.1 Hasil Pengumpulan Data

Dataset yang digunakan pada penelitian ini diambil dari platform Kaggle dengan judul *Book Genre Prediction using NLP* yang disusun oleh Prathamesh Gadekar. Dataset tersebut berisi 4.657 baris data dengan dua kolom utama, yaitu *summary* yang berisi sinopsis novel dan *genre* sebagai label kategorinya. Data memiliki karakteristik yang tidak seimbang karena jumlah novel pada tiap genre berbeda cukup signifikan. Ketidakseimbangan ini terlihat dari distribusi kelas yang menunjukkan dominasi beberapa genre tertentu dibandingkan lainnya.

Sebanyak 1.000 baris data dipilih secara acak untuk digunakan dalam penelitian. Pemilihan sampel dilakukan untuk menjaga efisiensi proses komputasi serta memastikan data yang digunakan tetap representatif terhadap keseluruhan dataset. Proses pengambilan dilakukan menggunakan fungsi sample_n() dari library dplyr pada bahasa pemrograman R, sebagaimana ditunjukkan pada potongan kode berikut:

```
#SAMPLING 1000 DATA
set.seed(123)
buku_1000 <- buku_raw %>%
  group_by(genre) %>%
  sample_n(size = min(1000, n()), replace = FALSE) %>%
  ungroup() %>%
  sample_n(1000)
```

Gambar 2 Kode Sampling Data

Tabel 1 menampilkan contoh data hasil pengambilan sampel. Setiap baris merepresentasikan satu novel dengan potongan sinopsis dan label genre yang sesuai.

Tabel 1 Hasil Sampling Data

No	Title	Genre	Summary
1	*The Steerswoman*	Fantasy	*The story begins with the Steerswoman Rowan investigating the origins of a number…*
2	*Inkspell*	Fantasy	*After the events of Inkheart, Meggie and her family discover that the world of books…*
3	*The Twelfth Card*	Thriller	*A young woman becomes the target of a dangerous man seeking a mysterious document…*
4	*The Boy in the Striped Pyjamas*	History	*Set during World War II, an innocent friendship forms between two boys separated by…*

 ISSN:2302-8149
e-ISSN:2540-9719

No	Title	Genre	Summary
5	*The Running Dream*	Sports	*When a track star loses her leg in an accident, she must find the strength to run again…*

4.2 Hasil Pra-pemrosesan Teks

Tahap pra-pemrosesan dilakukan untuk mengubah teks pada kolom *summary* menjadi bentuk numerik yang dapat dipahami oleh algoritma machine learning. Langkah-langkah yang dilakukan meliputi case folding, tokenisasi, penghapusan stopwords, stemming, dan pembobotan kata menggunakan TF-IDF.

Proses ini dijalankan dengan fungsi-fungsi utama dari library tidytext, SnowballC, dan dplyr. Tahap tokenisasi dilakukan menggunakan unnest_tokens() untuk memecah teks sinopsis menjadi token kata tunggal, sementara penghapusan stopwords menggunakan anti_join(stop_words) agar kata umum seperti *the*, *is*, dan *and* dihilangkan. Setelah itu, kata-kata dikembalikan ke bentuk dasarnya menggunakan wordStem() dari library SnowballC.

```
buku_tokens <- buku_1000 %>%
  mutate(doc_id = row_number()) %>%
  unnest_tokens(word, summary) %>%
  anti_join(stop_words, by = "word") %>%
  mutate(word = wordStem(word)) %>%
  count(doc_id, genre, word) %>%
```

Gambar 3 Kode Pra-pemrosesan Teks Sinopsis

Contoh hasil pra-pemrosesan sinopsis ditunjukkan pada Tabel 2. Proses ini menghasilkan teks yang lebih bersih dan ringkas sehingga model dapat mengenali kata-kata yang relevan dengan karakteristik setiap genre.

Tabel 2 Hasil Pra-pemrosesan Teks Sinopsis

Tahap	Hasil
Teks Asli	She is a young woman who moves to a small town and finds love in unexpected places while facing her painful past.
Case Folding	she is a young woman who moves to a small town and finds love in unexpected places while facing her painful past.
Tokenisasi	she, is, a, young, woman, who, moves, to, a, small, town, and, finds, love, in, unexpected, places, while, facing, her, painful, past
Stopwords Removal	young, woman, moves, small, town, finds, love, unexpected, places, facing, painful, past
Stemming	young, woman, move, small, town, find, love, unexpect, place, face, pain, past

Setelah tahap pembersihan teks selesai, pembobotan dilakukan menggunakan fungsi bind_tf_idf() untuk menghasilkan nilai TF-IDF (Term Frequency-Inverse Document Frequency). Nilai ini digunakan untuk mengukur tingkat kepentingan suatu kata terhadap sebuah dokumen dibandingkan dengan seluruh dokumen lain dalam korpus. Proses ini dijalankan dengan kode berikut:

```
# TF-IDF
buku_tfidf <- buku_tokens %>%
  bind_tf_idf(word, doc_id, n)
```

Gambar 4 Kode fungsi bind

Fungsi bind_tf_idf() menghitung nilai frekuensi kata (*term frequency*) dan seberapa jarang kata tersebut muncul di seluruh dokumen (*inverse document frequency*), lalu mengalikannya untuk

 ISSN:2302-8149
e-ISSN:2540-9719

menghasilkan bobot TF-IDF. Nilai yang lebih tinggi menunjukkan bahwa kata tersebut memiliki makna yang lebih spesifik terhadap dokumen tertentu. Selanjutnya, untuk mengurangi kompleksitas fitur, hanya 500 kata dengan bobot TF-IDF tertinggi yang dipilih menggunakan kode berikut:

```
# 500 terms
top_terms <- buku_tfidf %>%
  group_by(word) %>%
  summarise(total_tfidf = sum(tf_idf)) %>%
  arrange(desc(total_tfidf)) %>%
  head(500)
```

Gambar 5 Kode proses pemilihan 500 kata dengan nilai TF-IDF tertinggi

Hasil perhitungan TF-IDF menghasilkan daftar kata dengan bobot tertinggi yang dianggap paling representatif dalam membedakan genre novel. Nilai TF-IDF yang besar menunjukkan bahwa kata tersebut sering muncul dalam dokumen tertentu namun jarang ditemukan pada dokumen lain, sehingga memiliki kontribusi yang kuat terhadap proses klasifikasi. Sepuluh kata dengan nilai TF-IDF tertinggi ditampilkan pada Tabel 3, yang menunjukkan bahwa kata-kata tersebut memiliki korelasi kuat dengan tema atau karakteristik genre novel.

Tabel 3 Sepuluh kata dengan nilai TF-IDF tertinggi

No	Kata	Total TF-IDF	No	Kata	Total TF-IDF
1	magic	0.0875	6	mystery	0.0759
2	kingdom	0.0851	7	love	0.0738
3	murder	0.0813	8	dream	0.0725
4	school	0.0784	9	war	0.0716
5	dragon	0.0772	10	family	0.0708

Tabel 3 menunjukkan bahwa terdapat sejumlah kata yang memberikan kontribusi terbesar terhadap keputusan klasifikasi. Kata *magic*, *kingdom*, dan *dragon* paling banyak muncul pada novel bergenre fantasy sehingga menjadi penanda kuat bagi model untuk mengenali genre tersebut. Kata *murder* dan *mystery* banyak ditemukan pada cerita yang berfokus pada penyelidikan dan kriminal, sehingga mendorong model untuk memprediksi thriller atau crime. Kata *love*, *war*, dan *family* juga memiliki bobot tinggi karena sering muncul pada novel romance dan history yang menampilkan hubungan emosional dan latar konflik.

Kata-kata dengan bobot tinggi membantu model membedakan genre yang memiliki ciri kosakata yang jelas, sedangkan genre dengan jumlah data sedikit atau kosakata yang mirip dengan genre lain tidak memiliki kata kunci yang cukup kuat sebagai pembeda. Keadaan ini menyebabkan genre seperti psychology dan sports cenderung keliru diprediksi sebagai thriller atau fantasy. Temuan tersebut menunjukkan bahwa akurasi tidak hanya dipengaruhi oleh algoritma, tetapi juga oleh seberapa jelas ciri bahasa pada tiap genre.

4.3 Training Random Forest

Model utama yang digunakan dalam penelitian ini adalah Random Forest (RF), yang dilatih menggunakan data hasil pra-pemrosesan dengan representasi TF-IDF. Proses pelatihan dilakukan menggunakan fungsi randomForest() dari library *randomForest* dengan jumlah pohon keputusan sebanyak 300. Parameter ntree = 300 dipilih agar model memperoleh stabilitas hasil tanpa menyebabkan waktu komputasi berlebih.

```
# Training Random Forest
set.seed(123)
model_rf <- randomForest(genre ~ ., data = train_manual_balanced, ntree = 300)
```

Gambar 6 Kode training random forest

Setiap pohon dalam model dilatih pada subset data yang dipilih secara acak melalui mekanisme *bagging*. Selama pelatihan, algoritma juga memilih subset fitur secara acak untuk setiap pemisahan cabang (*split*), sehingga menghasilkan variasi antar pohon dan mengurangi risiko

 ISSN:2302-8149
e-ISSN:2540-9719

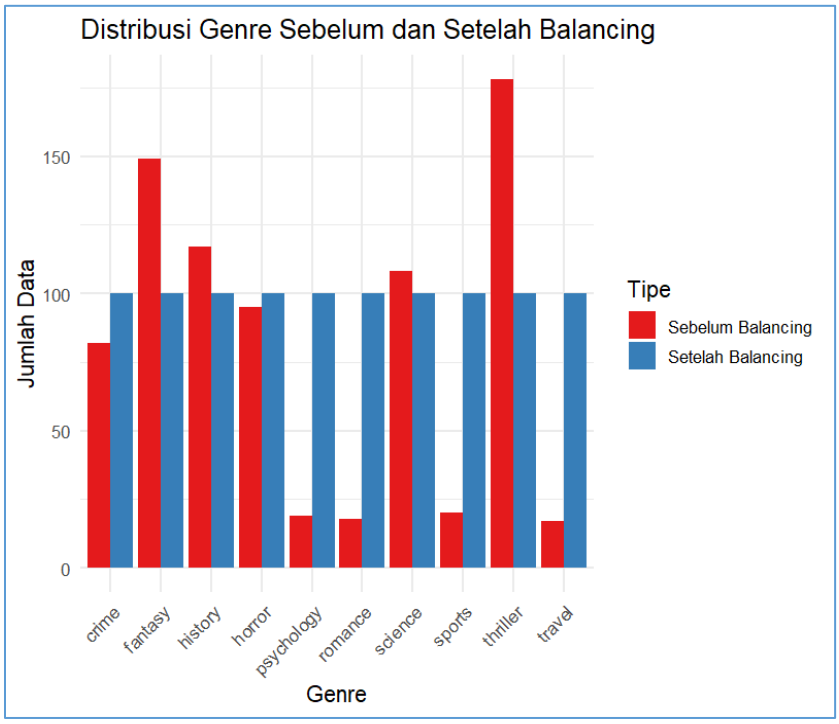
overfitting. Hasil akhir prediksi pada setiap dokumen ditentukan melalui proses *voting* mayoritas antar pohon dalam hutan keputusan. Pelatihan model berjalan dengan baik dan menghasilkan model dengan stabilitas prediksi yang tinggi terhadap data uji.

4.4 Pengujian Model

Dataset yang digunakan memiliki distribusi kelas yang tidak merata karena jumlah data pada setiap genre berbeda secara signifikan. Kondisi ini dibiarkan apa adanya pada tahap utama penelitian untuk menilai sejauh mana algoritma Random Forest mampu menangani data tidak seimbang tanpa teknik penyeimbangan tambahan. Pengujian tambahan dengan data yang telah diseimbangkan tetap dilakukan sebagai analisis komparatif guna melihat perbedaan performa model antara kondisi data asli dan data hasil balancing. Proses penyeimbangan dilakukan menggunakan fungsi sample_n() dari library *dplyr*, dengan mengambil masing-masing 100 sampel dari setiap genre untuk membentuk data pelatihan yang seimbang.

```
# Manual Balancing
train_manual_balanced <- train_data %>%
  group_by(genre) %>%
  sample_n(size = 100, replace = TRUE) %>%
  ungroup()
```

Gambar 7 Kode manual balancing



Gambar 8 Distribusi Genre Sebelum dan Sesudah Balancing

Setelah proses pembentukan data pelatihan selesai, model diuji menggunakan data uji yang telah dipisahkan sebelumnya. Tahap ini bertujuan untuk mengevaluasi kemampuan algoritma Random Forest dalam memprediksi genre novel berdasarkan teks sinopsis yang belum pernah digunakan selama proses pelatihan. Pembagian data dilakukan menggunakan fungsi createDataPartition() dari library *caret* dengan proporsi 80% untuk pelatihan dan 20% untuk pengujian, sehingga distribusi setiap kelas tetap proporsional.

Evaluasi kinerja model dilakukan menggunakan fungsi evaluate_model(), yang digunakan untuk menghitung nilai Accuracy, Cohen's Kappa, dan F1-Score. Fungsi ini mengotomatisasi proses perhitungan dengan memanfaatkan confusionMatrix() dari library *caret*, sehingga hasil evaluasi antar model dapat dibandingkan secara konsisten. Fungsi ini menerima tiga parameter, yaitu hasil prediksi

 ISSN:2302-8149
e-ISSN:2540-9719

model (predictions), label sebenarnya (actual), dan nama model (model_name). Hasil keluaran berupa daftar (*list*) berisi nilai metrik evaluasi serta *confusion matrix* yang digunakan untuk analisis performa model pada tahap selanjutnya.

```
# Fungsi evaluasi
evaluate_model <- function(predictions, actual, model_name) {
  predictions <- factor(predictions, levels = levels(actual))
  conf_matrix <- confusionMatrix(predictions, actual, mode = "prec_recall")

  accuracy <- round(conf_matrix$overall['Accuracy'] * 100, 2)
  kappa <- round(conf_matrix$overall['Kappa'], 3)
  f1_scores <- conf_matrix$byClass[, 'F1']
  f1_mean <- round(mean(f1_scores, na.rm = TRUE), 3)

  return(list(
    model_name = model_name,
    accuracy = accuracy,
    kappa = kappa,
    f1_mean = f1_mean,
    conf_matrix = conf_matrix
  ))
}
```

Gambar 9 Kode fungsi evaluasi model

Model Random Forest (RF) diuji menggunakan data uji sebanyak 20% dari total dataset. Proses pengujian dilakukan dengan memanggil fungsi predict() untuk menghasilkan prediksi kelas pada data uji, kemudian hasilnya dievaluasi menggunakan fungsi evaluate_model().

```
#Random Forest
predictions_rf <- predict(model_rf, test_data)
results_rf <- evaluate_model(predictions_rf, test_data$genre, "Random Forest")
```

Gambar 10 Kode pengujian dan evaluasi random forest

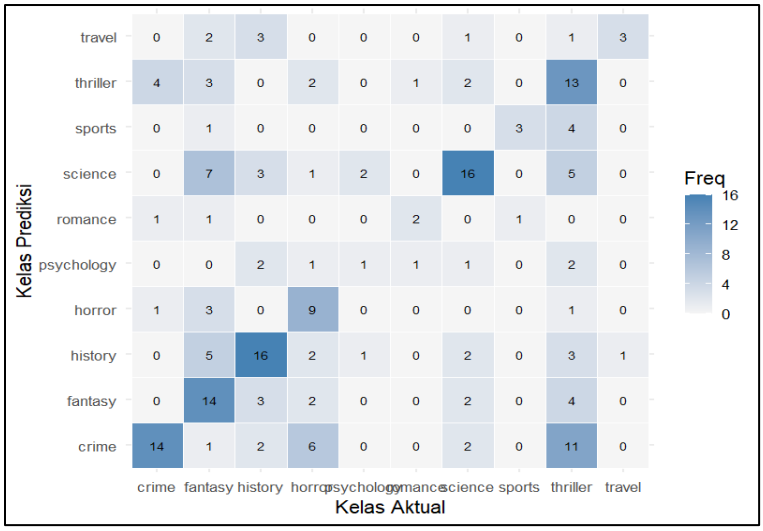
Hasil evaluasi dibedakan berdasarkan dua kondisi dataset, yaitu sebelum balancing dan setelah balancing. Ringkasan hasil evaluasi performa Random Forest disajikan pada Tabel 4.

Tabel 4 Perbandingan hasil pengujian sebelum dan sesudah balancing

Model	Kondisi Data	Akurasi (%)	Kappa	F1-Score
Random Forest	Sebelum Balancing	42.11	0.31	0.36
Random Forest	Setelah Balancing	46.67	0.387	0.443

Berdasarkan hasil pada Tabel 4, peningkatan akurasi, Kappa, dan F1-Score setelah dilakukan balancing menunjukkan bahwa penyebaran jumlah data antar genre berpengaruh pada kinerja Random Forest. Saat data masih tidak seimbang, model lebih mudah mengenali genre yang jumlah datanya banyak dan kesulitan mengidentifikasi genre yang jumlahnya sedikit. Setelah balancing, model menjadi lebih sensitif terhadap genre minoritas sehingga hasil prediksi lebih merata. Analisis confusion matrix kemudian digunakan untuk melihat pola kesalahan prediksi secara lebih rinci, termasuk mengidentifikasi genre yang paling sering tertukar serta mencari kemungkinan alasan munculnya kesalahan tersebut.

 ISSN:2302-8149
e-ISSN:2540-9719



Gambar 11 Confusion matrix model random forest

Berdasarkan Gambar 11, hasil confusion matrix menunjukkan bahwa Random Forest mampu mengklasifikasikan sebagian besar genre novel dengan cukup baik. Pola diagonal yang dominan menandakan bahwa sebagian besar prediksi berada pada kelas yang benar, terutama pada genre dengan jumlah data relatif banyak seperti *crime*, *history*, *fantasy*, dan *thriller*. Meskipun demikian, tidak semua genre menunjukkan pola prediksi yang sama baiknya.

Beberapa genre memiliki tingkat kesalahan prediksi yang lebih tinggi dibandingkan genre lain. Genre seperti *psychology*, *sports*, dan *romance* menjadi kategori yang paling sering salah karena jumlah datanya lebih sedikit dan kosakatanya tidak cukup khas untuk menjadi pembeda. Ketiga genre tersebut kerap terklasifikasi sebagai *thriller* atau *fantasy*, yang merupakan genre dengan data lebih besar dan ciri kosakata yang lebih kuat. Kedekatan tema juga memengaruhi kesalahan prediksi, misalnya *psychology* dan *thriller* sama-sama memuat unsur konflik emosional dan ketegangan sehingga beberapa sinopsis mudah tertukar. Temuan ini menunjukkan bahwa performa model dalam mengenali suatu genre tidak hanya dipengaruhi oleh distribusi jumlah data, tetapi juga oleh kemiripan kosakata antar genre yang menyebabkan batas antar kategori menjadi kurang jelas bagi model.

4.5 Pengujian Model dengan Data Baru

Tahap pengujian tambahan dilakukan untuk menguji kemampuan generalisasi model Random Forest terhadap data baru yang belum pernah digunakan pada proses pelatihan maupun pengujian sebelumnya. Lima sinopsis novel dipilih secara acak dari dataset eksternal untuk melihat kemampuan model dalam mengenali genre berdasarkan isi teks. Proses prediksi dilakukan menggunakan fungsi predict() pada model Random Forest yang telah dilatih sebelumnya.

```
set.seed(123)
sampel_index <- sample(1:nrow(df_final_clean), 5)

prediksi_baru <- predict(model_rf, df_final_clean[sampel_index, -ncol(df_final_clean)])

hasil_uji <- data.frame(
  Judul = buku_1000$title[sampel_index],
  Sinopsis = substr(buku_1000$summary[sampel_index], 1, 120),
  Genre_Asli = df_final_clean$genre[sampel_index],
  Prediksi_RF = prediksi_baru
)
hasil_uji$Keterangan <- ifelse(hasil_uji$Genre_Asli == hasil_uji$Prediksi_RF, "Benar", "Salah")
print(hasil_uji)
```

Gambar 12 Kode pengujian dengan data baru

 ISSN:2302-8149
e-ISSN:2540-9719

Tabel 5 Hasil prediksi data baru oleh model random forest

No	Judul	Sinopsis (potongan)	Genre Asli	Prediksi RF	Keterangan
1	*Gladiator at Law*	The action takes place in and around a future Monmouth City, New Jersey. The city proper consists of luxurious GML bubbles...	science	science	Benar
2	*1984*	Among the seminal texts of the 20th century, Nineteen Eighty-Four is a rare work that grows more haunting as its futurism...	science	science	Benar
3	*206 Bones*	Tempe is still unsure whether to continue her romantic relationship with Andrew Ryan. Tempe and Ryan set out to Chicago...	crime	crime	Benar
4	*The City & the City*	Inspector Tyador Borlú, of the Extreme Crime Squad in the European city-state of Besźel, investigates the murder of Maha...	crime	crime	Benar
5	*Graceling*	The novel *Graceling* by Kristin Cashore follows the life of the 18-year-old Katsa. She is a Graceling, a person with a gr...	fantasy	history	Salah

5 Kesimpulan

Penelitian ini bertujuan mengevaluasi kemampuan algoritma Random Forest dalam mengklasifikasikan genre novel berdasarkan teks sinopsis pada kondisi distribusi data yang tidak seimbang. Hasil pengujian menunjukkan bahwa model dapat mengenali sebagian besar genre dengan tingkat ketepatan sedang, terutama pada kategori dengan jumlah data relatif lebih besar seperti *crime*, *history*, *fantasy*, dan *thriller*. Namun performa model belum merata pada seluruh kategori karena tingkat pengenalan genre menurun cukup tajam pada kelas dengan jumlah data sedikit. Confusion matrix mengungkap bahwa kesalahan prediksi terutama terjadi pada genre *psychology*, *sports*, dan *romance*. Ketiga genre tersebut memiliki kosakata yang tumpang tindih dengan genre dominan sehingga model kesulitan membedakan ciri bahasa yang menjadi penanda setiap kategori. Keterbatasan kosakata unik pada genre minoritas serta kemiripan tema antar genre memperkuat kemungkinan terjadinya *misclassification*. Pola ini menunjukkan bahwa performa klasifikasi tidak hanya dipengaruhi oleh ukuran dataset, tetapi juga oleh kekhasan istilah dan karakteristik bahasa pada masing-masing genre. Kontribusi ilmiah penelitian ini terletak pada penyajian bukti empiris mengenai kinerja Random Forest pada klasifikasi genre novel berbasis sinopsis dalam kondisi data yang tidak seimbang. Temuan menunjukkan bahwa model tetap dapat bekerja tanpa teknik balancing otomatis, tetapi kualitas prediksi perlu dianalisis secara per-genre karena akurasi keseluruhan belum mencerminkan kualitas prediksi pada kategori minoritas. Dengan demikian, evaluasi per-genre menjadi pendekatan penting untuk memahami performa algoritma secara lebih detail pada data tidak seimbang. Penelitian ini memiliki keterbatasan pada ukuran dataset yang belum cukup besar untuk mempelajari pola bahasa yang khas pada semua genre, serta pada representasi fitur yang hanya menggunakan TF-IDF sehingga makna kalimat belum sepenuhnya tertangkap oleh model. Berdasarkan hasil penelitian, kesalahan klasifikasi terutama dipengaruhi oleh kedekatan kosakata antar genre dan rendahnya jumlah data pada beberapa kategori. Oleh karena itu, penelitian lanjutan dapat mempertimbangkan penggunaan model embedding seperti Word2Vec, FastText, atau BERT untuk meningkatkan kemampuan model dalam memahami konteks bahasa. Selain itu, pendekatan ensemble yang dirancang khusus untuk data tidak seimbang, seperti Balanced Random Forest atau EasyEnsemble, dapat dieksplorasi untuk meningkatkan sensitivitas model terhadap kategori dengan proporsi data kecil tanpa mengubah distribusi kelas.

Referensi

[1] A. Sethy, A. K. Rout, A. Uriti, and S. P. Yalla, "*Revue d ' Intelligence Artificielle A*

 ISSN:2302-8149
e-ISSN:2540-9719

Comprehensive Machine Learning Framework for Automated Book Genre Classifier," Vol. 37, No. 3, pp. 745–751, 2023.

[2] S. Nouas, L. Oukid, and F. Boumahdi, "ur l P re," *Data SCI. Manag.*, 2025, DOI: 10.1016/j.dsm.2025.03.001.

[3] C. Kaope and Y. Pristyanto, "*The Effect of Class Imbalance Handling on Datasets Toward Classification Algorithm Performance*," Vol. 22, No. 2, pp. 227–238, 2023, DOI: 10.30812/matrik.v22i2.2515.

[4] L. Dube and T. Verster, "*Enhancing Classification Performance in Imbalanced Datasets: A Comparative Analysis of Machine Learning Models*," *Data SCI. Financ. Econ.*, Vol. 3, No. 4, pp. 354–379, 2023, DOI: 10.3934/dsfe.2023021.

[5] N. Jalal, A. Mehmood, G. Sang, and I. Ashraf, "*A Novel Improved Random Forest for Text Classification using Feature Ranking and Optimal Number of Trees*," *J. King Saud Univ. - Comput. Inf. SCI.*, Vol. 34, No. 6, pp. 2733–2742, 2022, DOI: 10.1016/j.jksuci.2022.03.012.

[6] A. Agung, A. Witaradiani, I. G. Arta, and P. Praba, "Klasifikasi Genre Buku berdasarkan Sinopsis menggunakan *Naïve Bayes* dan *Logistic Regression*," Vol. 3, pp. 835–844, 2025.

[7] N. D. Primadya, A. Nugraha, and S. Y. Fahrezi, "*Optimizing Imbalanced Data Classification : Under Sampling Algorithm Strategy with Classification Combination*," No. April 2024, pp. 277–288.

[8] M. R. F. Rahmatullah, P. N. Andono, and M. A. Soeleman, "*Improving Random Forest Performance for Sentiment Analysis on Unbalanced Data using SMOTE and BoW Integration : PLN Mobile Application Case Study*," Vol. 12, No. 1, pp. 1–10, 2025, DOI: 10.15294/sji.v12i1.19295.

[9] A. Nawaz, A. Ahmad, and S. S. Khan, "*Beyond Rebalancing: Benchmarking Binary Classifiers Under Class Imbalance Without Rebalancing Techniques*," 2025, [Online]. Available: http://arxiv.org/abs/2509.07605

[10] S. Wang, Y. Dai, J. Shen, and J. Xuan, "*Research on Expansion and Classification of Imbalanced Data based on SMOTE Algorithm*," *SCI. Rep.*, Vol. 11, No. 1, pp. 1–11, 2021, DOI: 10.1038/s41598-021-03430-5.

[11] M. Altalhan, A. Algarni, and M. Turki-Hadj Alouane, "*Imbalanced Data Problem in Machine Learning: A Review*," *IEEE Access*, Vol. 13, pp. 13686–13699, 2025, DOI: 10.1109/ACCESS.2025.3531662.

[12] A. S. More and D. P. Rana, "*An Experimental Assessment of Random Forest Classification Performance Improvisation with Sampling and Stage Wise Success Rate Calculation*," *Procedia Comput. SCI.*, Vol. 167, No. Iccids 2019, pp. 1711–1721, 2020, DOI: 10.1016/j.procs.2020.03.381.

[13] T. Wongvorachan, S. He, and O. Bulut, "*A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining*," *Inf.*, Vol. 14, No. 1, 2023, DOI: 10.3390/info14010054.

[14] W. Chen, K. Yang, Z. Yu, Y. Shi, and C. L. P. Chen, *A Survey on Imbalanced Learning: Latest Research, Applications and Future Directions*, Vol. 57, No. 6. 2024. DOI: 10.1007/s10462-024-10759-6.

[15] D. Siswara, A. M. Soleh, and A. Hamim Wigena, "*Classification Modeling with RNN-based, Random Forest, and XGBoost for Imbalanced Data: A Case of Early Crash Detection in ASEAN-5 Stock Markets*," *Sci. J. Informatics*, Vol. 11, No. 3, pp. 569–582, 2024, DOI: 10.15294/sji.v11i3.4067.

[16] M. Imani, A. Beikmohammadi, and H. R. Arabnia, "*Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS under Varying Imbalance Levels*," *Technologies*, Vol. 13, No. 3, pp. 1–40, 2025, DOI: 10.3390/technologies13030088.