

# SENASTIKA Universitas Malikussaleh

---

## KLASIFIKASI CYBERBULLYING DI TWITTER MENGGUNAKAN METODE GRADIENT BOOSTING

Rofid Sazama<sup>\*1</sup>, Agil Muttaqin<sup>2</sup>, Riska Ananda<sup>3</sup>, Ririn Julianansa<sup>4</sup>, Munirul Ula, S.T., M.Eng, Ph.D<sup>5</sup>

Teknik Informatika, Universitas Malikussaleh, Lhokseumawe, Aceh

Email: <sup>1</sup> [rofid.220170167@mhs.unimal.ac.id](mailto:rofid.220170167@mhs.unimal.ac.id), <sup>2</sup> [agil.220170133@mhs.unimal.ac.id](mailto:agil.220170133@mhs.unimal.ac.id), <sup>3</sup>

[riska.210170267@mhs.unimal.ac.id](mailto:riska.210170267@mhs.unimal.ac.id), <sup>4</sup> [ririn.210170242@mhs.unimal.ac.id](mailto:ririn.210170242@mhs.unimal.ac.id), <sup>5</sup> [munirulula@unimal.ac.id](mailto:munirulula@unimal.ac.id)

### Abstrak

Cyberbullying merupakan salah satu masalah sosial yang berkembang pesat seiring dengan meningkatnya penggunaan media sosial. Deteksi dan klasifikasi kasus cyberbullying menjadi tantangan penting dalam meminimalisir dampak negatif terhadap korban. Penelitian ini bertujuan untuk mengklasifikasikan teks yang mengandung unsur cyberbullying dengan menggunakan metode *Gradient Boosting*. Proses penelitian dimulai dengan mengimpor pustaka yang diperlukan dan memuat *dataset* yang berisi komentar-komentar online. Kemudian, dilakukan *preprocessing* teks, seperti penghapusan *stop words* dan *tokenisasi*, untuk memastikan data siap digunakan. Selanjutnya, *dataset* dibagi menjadi data pelatihan dan data pengujian. Model *Gradient Boosting* diinisialisasi dan dilatih menggunakan data pelatihan, kemudian dilakukan prediksi pada data pengujian. Evaluasi model dilakukan dengan menggunakan metrik akurasi, presisi, *recall*, dan *F1-score* untuk mengukur kinerja klasifikasi. Hasil penelitian menunjukkan bahwa metode *Gradient Boosting* mampu mengklasifikasikan teks cyberbullying dengan akurasi yang memuaskan. Selain itu, visualisasi hasil evaluasi model membantu dalam memahami performa model secara lebih mendalam.

**Keywords:** *Cyberbullying, Klasifikasi Teks, Gradient Boosting, Preprocessing Teks, Evaluasi Model, Machine Learning, Visualisasi Hasil*

---

## 1. PENDAHULUAN

Cyberbullying adalah fenomena penyalahgunaan teknologi komunikasi, terutama media sosial, untuk mengintimidasi, menghina, atau menyakiti individu atau kelompok secara psikologis. Dengan semakin meluasnya penggunaan internet, kasus cyberbullying semakin meningkat dan menjadi ancaman serius, terutama bagi remaja dan kelompok rentan. Dampak dari cyberbullying tidak hanya terbatas pada kesehatan mental, tetapi juga dapat mempengaruhi performa akademik, interaksi sosial, dan bahkan menyebabkan tindakan yang lebih ekstrem, seperti depresi dan bunuh diri.

Di era digital ini, teknologi kecerdasan buatan (AI) dan pembelajaran mesin (*machine learning*) menawarkan solusi untuk mendeteksi cyberbullying secara otomatis. Algoritma pembelajaran mesin memungkinkan komputer untuk menganalisis pola dalam teks dan membuat keputusan berdasarkan data yang telah dilatih. Salah satu algoritma yang efektif dalam tugas klasifikasi teks adalah *Gradient Boosting*. Algoritma ini bekerja dengan membangun serangkaian model secara bertahap, di mana setiap model baru berfokus pada kesalahan prediksi yang dibuat oleh model sebelumnya. Hal ini memungkinkan *Gradient Boosting* untuk menghasilkan model yang lebih akurat dibandingkan dengan metode lainnya.

Penelitian bertujuan mengembang model klasifikasi teks berbasis *Gradient Boosting* untuk mendeteksi komentar atau konten yang mengandung cyberbullying. Proses pengembangan model ini terdiri dari beberapa tahapan, mulai dari *import* pustaka yang diperlukan, pemuatan *dataset*, *preprocessing* teks untuk membersihkan dan menyiapkan data, membagi *dataset* menjadi bagian pelatihan dan pengujian, melatih model, melakukan prediksi pada data pengujian, serta mengevaluasi kinerja model menggunakan metrik evaluasi seperti akurasi, presisi, *recall*, dan *F1-score*. Selain itu, visualisasi hasil evaluasi akan memberikan pemahaman yang lebih mendalam mengenai kinerja model.

Dengan hasil yang diharapkan dari penelitian ini, model klasifikasi yang dihasilkan dapat menjadi alat yang berguna dalam membantu platform media sosial dan pengembang aplikasi untuk secara otomatis mengidentifikasi dan menindaklanjuti kasus cyberbullying. Implementasi sistem semacam ini tidak hanya akan meningkatkan keamanan dunia maya, tetapi juga dapat berperan dalam menciptakan lingkungan digital yang lebih positif dan bebas dari kekerasan verbal.

## 2. METODE PENELITIAN

Penelitian ini dilakukan dengan mengikuti beberapa langkah sistematis yang mencakup pengumpulan data, preprocessing, pengembangan model, serta evaluasi model.

### 2.1. Pengumpulan Data

Dataset yang digunakan dalam penelitian ini diambil dari sumber komentar online yang tersedia secara publik. Data ini mencakup komentar yang mengandung unsur cyberbullying serta komentar yang tidak mengandung unsur tersebut. Dataset kemudian dibersihkan untuk menghilangkan entri yang tidak relevan.

### 2.2. Preprocessing Teks

Tahap preprocessing sangat penting untuk mempersiapkan data agar siap digunakan oleh model.

- **Tokenisasi:** Memecah input teks menjadi kata atau token.
- **Penghapusan Stop Words:** Menghapus kata umum tidak bermakna (misalnya: "dan", "atau", "dari").
- **Stemming:** Mengubah kata ke bentuk dasarnya untuk menyederhanakan variasi kata.
- **Representasi Teks:** Menggunakan representasi teks seperti TF-IDF (Term Frequency-Inverse Document Frequency) yang digunakan mengubah teks menjadi vektor numerik yang dapat digunakan oleh algoritma pembelajaran mesin.

### 2.3. Pembagian Dataset

Dataset dibagi menjadi data pelatihan dan data pengujian, dengan perbandingan 80:20. Data pelatihan digunakan melatih model, sedangkan data pengujian digunakan mengevaluasi kinerja model yang telah dilatih.

### 2.4. Pengembangan Model

- **Inisialisasi Model:** Menggunakan pustaka yang sesuai untuk menginisialisasi model Gradient Boosting.
- **Pelatihan Model:** Dilatih dengan menggunakan data pelatihan yang telah disiapkan. Proses pelatihan mencakup pengaturan parameter model untuk mencapai performa optimal.
- **Prediksi:** Setelah model dilatih, dilakukan prediksi terhadap data pengujian untuk mengklasifikasikan apakah komentar tersebut termasuk dalam kategori cyberbullying atau tidak.

### 2.5. Evaluasi Model

- **Akurasi:** Persentase prediksi benar dari total prediksi.
- **Presisi:** Rasio prediksi positif benar dibandingkan dengan total prediksi positif.
- **Recall:** Rasio prediksi positif benar dibandingkan dengan total kasus positif yang ada.
- **F1-score:** Rata Rata harmonis dari presisi dan recall yang memberikan gambaran kinerja

### 2.6. Visualisasi Hasil

Hasil evaluasi model divisualisasikan untuk memberikan pemahaman lebih baik tentang kinerja model. Visualisasi mencakup grafik confusion matrix, kurva ROC, dan metrik evaluasi lainnya.

## 3. HASIL DAN PEMBAHASAN

### 3.1. Import Libraries and Load Dataset

Proses penelitian dimulai dengan mengimpor pustaka untuk analisis data dan pengembangan model. Setelah semua pustaka diimpor, langkah selanjutnya adalah memuat dataset yang telah disiapkan. Dataset berisi komentar online yang dilabeli sebagai mengandung unsur cyberbullying atau tidak.

- **Input:** File CSV dengan tweet dan tipe cyberbullying.
  - **Proses:** Membaca file CSV ke dalam DataFrame.
- ```
#Import Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.feature_extraction.text import TfidfVectorizer
```

```

from sklearn.pipeline import make_pipeline
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix

# Load dataset
data_path = '/content/cyberbullying_tweets.csv'
df = pd.read_csv(data_path)

# Display dataframe
df.head(10)

```

- **Output:** DataFrame berisi data dari file CSV

| Data | Keterangan        |
|------|-------------------|
| 0    | not cyberbullying |
| 1    | not cyberbullying |
| 2    | not cyberbullying |
| 3    | not cyberbullying |
| 4    | not cyberbullying |
| 5    | not cyberbullying |
| 6    | not cyberbullying |
| 7    | not cyberbullying |
| 8    | not cyberbullying |
| 9    | not cyberbullying |

### 3.2. Preprocessing Teks

Setelah dataset dimuat, tahap berikutnya adalah preprocessing teks. Proses ini bertujuan untuk menyiapkan data agar dapat digunakan oleh model pembelajaran mesin.

- **Input:** Teks tweet kolom `tweet_text` dan label dari kolom `cyberbullying_type` dalam DataFrame `df`.

| Data | Keterangan        |
|------|-------------------|
| 0    | not cyberbullying |
| 1    | not cyberbullying |
| 2    | not cyberbullying |
| 3    | not cyberbullying |
| 4    | not cyberbullying |
| 5    | not cyberbullying |
| 6    | not cyberbullying |
| 7    | not cyberbullying |
| 8    | not cyberbullying |
| 9    | not cyberbullying |

- **Proses:** Menggunakan `TfidfVectorizer` untuk mengubah teks menjadi fitur numerik (TF-IDF). Menetapkan target variable untuk klasifikasi.

```

# Ubah data teks menjadi fitur numerik menggunakan TF-IDF Vectorizer
vectorizer = TfidfVectorizer(max_features=5000)
X = vectorizer.fit_transform(df['tweet_text'])

```

```

# Asumsikan kolom target bernama 'label'
y = df['cyberbullying_type']

```

- **Output:** `x`: Matriks sparse fitur TF-IDF dari teks tweet.  
`y`: Series dari label tipe cyberbullying yang akan digunakan untuk pelatihan model klasifikasi.

### 3.3. Split Dataset

Dataset kemudian dibagi menjadi: 80% untuk data pelatihan dan 20% untuk data pengujian. Pembagian penting memastikan bahwa model dilatih dengan cukup data dan dapat diuji menggunakan data yang belum pernah dilihat sebelumnya.

- **Input:** `X`: Matriks fitur TF-IDF dari teks tweet  
`Y`: Series label tipe cyberbullying

- **Proses:**

1. Membagi dataset menjadi data pelatihan dan data pengujian menggunakan `train_test_split`.
2. `test_size=0.2`: 20% data digunakan untuk pengujian, 80% data digunakan untuk pelatihan.
3. `random_state=42`: Seed untuk memastikan hasil pemisahan yang konsisten.

```
# Pisahkan kumpulan data menjadi kumpulan pelatihan dan pengujian
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

- **Output:** `X_train`: Subset dari X untuk pelatihan (80% data fitur)  
`X_test`: Subset dari X untuk pengujian (20% data fitur)  
`Y_train`: Subset dari y untuk pelatihan (80% data target)  
`Y_test`: Subset dari y untuk pengujian (20% data target)

### 3.4. Initialize dan Train Model Gradient Boosting

Setelah data dibagi, langkah berikutnya adalah menginisialisasi dan melatih model Gradient Boosting. Model diinisialisasi menggunakan pustaka `sklearn`, dan parameter default digunakan pada tahap awal. Selama proses pelatihan, model dilatih dengan data pelatihan yang telah diproses.

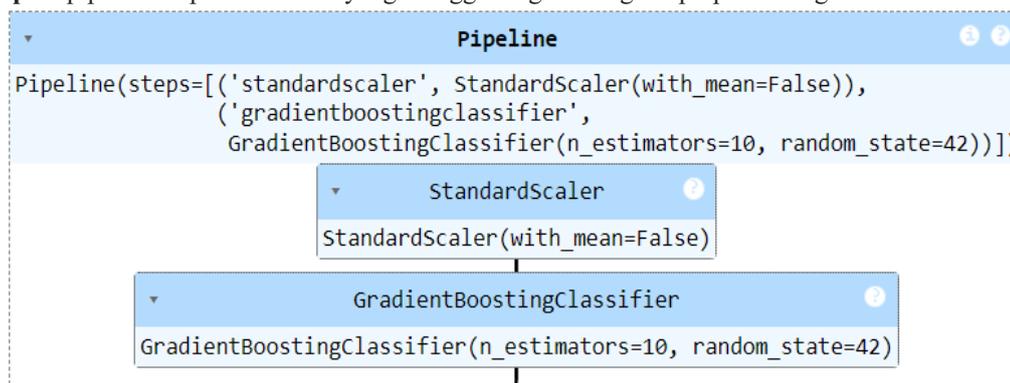
- **Input:** `X_train`: Matriks fitur TF-IDF dari teks tweet untuk pelatihan  
`Y_train`: Label tipe cyberbullying untuk pelatihan
- **Proses:** `StandardScaler` (`with_mean=False`):
  - Menstandarisasi fitur dengan menghapus rata-rata dan menskalakan ke unit varians, tanpa mengurangi rata-rata karena matriks sparse.

```
GradientBoostingClassifier(n_estimators=10, random_state=42):
```

- Membangun model ensemble dari 10 pohon keputusan yang lemah.
- Melatih model menggunakan data yang telah distandarisasi.

```
# Inisialisasi dan latih model Gradient Boosting menggunakan pipeline dengan StandardScaler
pipeline = make_pipeline(StandardScaler(with_mean=False), GradientBoostingClassifier(n_estimators=10,
random_state=42))
pipeline.fit(X_train, y_train)
```

- **Output:** `pipeline`: Pipeline terlatih yang menggabungkan langkah preprocessing dan model.



### 3.5. Predict on Testing Set

Setelah model dilatih, langkah berikutnya adalah melakukan prediksi pada data pengujian. Model yang telah terlatih diterapkan untuk mengklasifikasikan komentar dalam dataset pengujian. Model menghasilkan prediksi yang menunjukkan apakah komentar tersebut mengandung unsur cyberbullying atau tidak.

- **Input:** `X_test`: Matriks fitur TF-IDF dari teks tweet untuk pengujian  
`pipeline`: Pipeline terlatih yang menggabungkan `StandardScaler` dan `GradientBoostingClassifier`
- **Proses:** `pipeline.predict(X_test)`:

- StandardScaler: Menstandarisasi X\_test berdasarkan mean dan variance dari X\_train.
- GradientBoostingClassifier: Memprediksi label target untuk data yang telah distandarisasi.

```
# Prediksi set pengujian
y_pred = pipeline.predict(X_test)
```

- **Output:** y\_pred: Array prediksi label untuk data pengujian X\_test.

### 3.6. Evaluate the Model

Kinerja model menggunakan beberapa metrik, yaitu akurasi, presisi, recall, dan F1-score.

- **Input:** y\_test: Label sebenarnya untuk data pengujian  
y\_pred: Label yang diprediksi oleh model untuk data pengujian
- **Proses:** accuracy\_score(y\_test, y\_pred):
  - Menghitung akurasi prediksi.
 classification\_report(y\_test, y\_pred):
  - Menghasilkan laporan terperinci precision, recall, F1-score, dan support untuk setiap kelas.
 confusion\_matrix(y\_test, y\_pred):
  - Menghasilkan matriks kebingungan yang menunjukkan jumlah prediksi benar dan salah

```
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy:.2f}")

# Detailed classification report
print("\nLogistic Regression Classification Report:")
print(classification_report(y_test, y_pred))

# Confusion Matrix
print("\nLogistic Regression Confusion Matrix:")
print(confusion_matrix(y_test, y_pred))
```

- **Output:** Accuracy:
  - Nilai akurasi model yang ditampilkan dalam format dua digit desimal.
 Classification Report:
  - Laporan terperinci tentang performa klasifikasi.
 Confusion Matrix:
  - Matriks kebingungan menunjukkan distribusi prediksi benar dan salah untuk masing kelas.

Accuracy: 0.78

| Logistic Regression | Classification Report: |        |          |         |
|---------------------|------------------------|--------|----------|---------|
|                     | precision              | recall | f1-score | support |
| age                 | 0.94                   | 0.98   | 0.96     | 1603    |
| ethnicity           | 0.98                   | 0.94   | 0.96     | 1603    |
| gender              | 0.96                   | 0.70   | 0.81     | 1531    |
| not_cyberbullying   | 0.68                   | 0.30   | 0.42     | 1624    |
| other_cyberbullying | 0.48                   | 0.93   | 0.63     | 1612    |
| religion            | 0.97                   | 0.85   | 0.91     | 1566    |

|              |      |      |      |      |
|--------------|------|------|------|------|
| accuracy     |      |      | 0.78 | 9539 |
| macro avg    | 0.84 | 0.78 | 0.78 | 9539 |
| weighted avg | 0.83 | 0.78 | 0.78 | 9539 |

Logistic Regression Confusion Matrix:

```
[[1578  2  0  4  19  0]
 [  11 1502  2  13  69  6]
 [   3  5 1075 111 336  1]
 [  70  5  11 492 1013 33]
 [   6  4  14  95 1492  1]
 [   6  7  12  7  204 1330]]
```

### 3.7. Visualisasi Hasil

Visualisasi hasil evaluasi model dilakukan untuk memberikan gambaran tentang kinerja model. Salah satu alat visualisasi yang digunakan adalah confusion matrix, yang menunjukkan perbandingan antara prediksi model dan label sebenarnya.

- **Input:** `y_test`: Label sebenarnya untuk data pengujian  
`y_pred`: Label yang diprediksi oleh model untuk data pengujian  
`pipeline`: Pipeline terlatih yang menggabungkan `StandardScaler` dan `GradientBoostingClassifier`  
`vectorizer`: TF-IDF Vectorizer untuk mengubah teks tweet menjadi fitur numeric  
`df`: DataFrame yang berisi data teks (tweet) dan label tipe cyberbullying
- **Proses:** `confusion_matrix(y_test, y_pred)`:
  - Menghasilkan matriks kebingungan untuk mengevaluasi performa model.`sns.heatmap()`:
  - Menghasilkan heatmap untuk matriks kebingungan.`sns.countplot()`:
  - Menghitung dan menampilkan distribusi kelas yang diprediksi.`feature_importances`:
  - Menghitung dan menampilkan pentingnya fitur model `GradientBoostingClassifier`.`sns.barplot()`:
  - Menampilkan plot bar untuk fitur-fitur paling penting.

```
# Confusion Matrix
```

```
conf_matrix = confusion_matrix(y_test, y_pred)
```

```
plt.figure(figsize=(8, 6))
```

```
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues', xticklabels=['Not Cyberbullying', 'Cyberbullying'], yticklabels=['Not Cyberbullying', 'Cyberbullying'])
```

```
plt.xlabel('Predicted Label')
```

```
plt.ylabel('True Label')
```

```
plt.title('Confusion Matrix')
```

```
plt.show()
```

```
# Distribution of Predicted Classes
```

```
pred_df = pd.DataFrame({'Predicted': y_pred})
```

```
plt.figure(figsize=(10, 6))
```

```
sns.countplot(data=pred_df, x='Predicted', order=df['cyberbullying_type'].unique())
```

```
plt.title('Distribution of Predicted Classes')
```

```
plt.xlabel('Class')
```

```
plt.ylabel('Frequency')
```

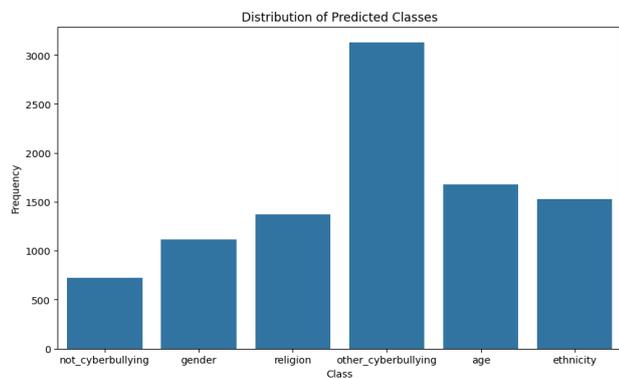
```
plt.show()

# Feature Importances
feature_importances = pipeline.named_steps['gradientboostingclassifier'].feature_importances_
features = vectorizer.get_feature_names_out()
importance_df = pd.DataFrame({'feature': features, 'importance': feature_importances})
importance_df = importance_df.sort_values(by='importance', ascending=False).reset_index(drop=True)
plt.figure(figsize=(12, 6))
sns.barplot(x='importance', y='feature', hue='feature', data=importance_df.head(10), palette='viridis',
            legend=False)
plt.title('Top 10 Important Features')
plt.xlabel('Importance')
plt.ylabel('Feature')
plt.show()
```

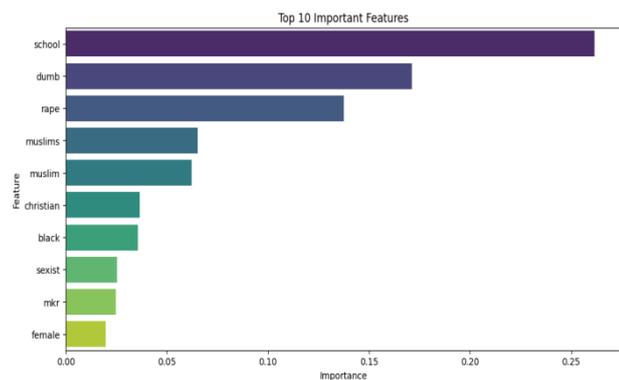
- **Output:** Heatmap dari Confusion Matrix



Grafik bar distribusi kelas yang diprediksi



Plot bar dari fitur-fitur paling penting



#### 4. DISKUSI

Berdasarkan hasil evaluasi model Gradient Boosting yang telah diimplementasikan, dapat dilihat bahwa model ini mampu memberikan hasil klasifikasi yang cukup baik dalam mendeteksi teks yang mengandung unsur cyberbullying. Nilai akurasi sebesar 78% menunjukkan bahwa metode ini efektif dalam mengidentifikasi pola-pola dalam teks yang berhubungan dengan cyberbullying. Namun, terdapat beberapa aspek penting yang perlu diperhatikan dari hasil evaluasi ini. Salah satu aspek yang patut dicermati adalah perbedaan nilai precision dan recall pada beberapa kategori, terutama kategori *other\_cyberbullying*. Meskipun recall untuk kategori ini tinggi (93%), yang menunjukkan bahwa model mampu mengidentifikasi sebagian besar kasus cyberbullying, nilai precision untuk kategori ini relatif rendah (48%).

Selain itu, confusion matrix menunjukkan bahwa sebagian besar kesalahan klasifikasi terjadi pada kelas *not\_cyberbullying* dan *other\_cyberbullying*. Hal ini bisa disebabkan oleh adanya kemiripan pola bahasa antara komentar yang netral dengan yang mengandung unsur cyberbullying, terutama dalam konteks penggunaan bahasa yang ambigu atau mengandung sarkasme. Penelitian di masa mendatang dapat mempertimbangkan penggunaan pendekatan berbasis *deep learning* yang mampu menangkap konteks bahasa secara lebih mendalam.

#### 5. KESIMPULAN

Penelitian ini telah mengimplementasikan metode Gradient Boosting untuk klasifikasi cyberbullying, dengan hasil yang menunjukkan tingkat akurasi sebesar 78%. Model ini terbukti efektif dalam mendeteksi beberapa kategori cyberbullying, khususnya pada kategori yang lebih eksplisit seperti *ethnicity* dan *age*. Namun, hasil evaluasi juga menunjukkan adanya tantangan dalam mengklasifikasikan kategori *other\_cyberbullying*, yang mengalami kesalahan klasifikasi lebih sering, terutama pada prediksi yang keliru antara komentar netral dan yang mengandung cyberbullying. Meski begitu, recall yang tinggi menunjukkan sensitivitas model dalam mendeteksi sebagian besar kasus cyberbullying, namun perlu ada peningkatan pada precision untuk mengurangi prediksi yang salah. Ini menjadi indikasi bahwa model Gradient Boosting dapat memberikan hasil yang baik, tetapi memiliki keterbatasan dalam menangkap konteks bahasa yang lebih rumit, terutama pada teks yang ambigu atau mengandung sarkasme.

Penggunaan pendekatan yang lebih maju dalam pemrosesan bahasa alami, seperti *word embeddings* atau model berbasis *deep learning* seperti *BERT*, dapat membantu meningkatkan performa model, terutama dalam memahami konteks bahasa yang lebih kompleks. Dengan demikian, meskipun metode Gradient Boosting cukup menjanjikan, penelitian di masa depan perlu mempertimbangkan teknik-teknik yang lebih canggih untuk mengatasi tantangan klasifikasi pada kategori yang lebih sulit dan meningkatkan akurasi keseluruhan.

#### UCAPAN TERIMA KASIH

Dalam mengakhiri artikel ini, saya mengucapkan terima kasih kepada dosen pengampun pak Dr. Munirul Ula, ST. atas bimbingannya di tempat mengabdikan saya Universitas Malikussaleh dengan Matakuliah Riset Teknologi Informasi. Terima kasih juga yang sebesar-besarnya atas waktu dan perhatian kepada semua pembaca dalam membaca hasil penelitian ini. Semoga artikel ini bermanfaat dan dapat memberikan pemahaman yang lebih mendalam khususnya pada Metode Boosting.

#### DAFTAR PUSTAKA

- [1] L. P. Nasyuli, I. Lubis, and A. M. Elhanafi, "Penerapan Model Machine Learning Algoritma Gradient Boosting dan Linear Regression Melakukan Prediksi Harga Kendaraan Bekas Application Of Machine Learning Models and Gradient Boosting Algorithms Doing Linear Regression Vehicle Price Prediction Used," *J. Ilmu Komput. dan Sist. Informasi (JIRSI)*, vol. 2, no. 2, pp. 299–310, 2023, [Online]. Available: <https://jurnal.unity-academy.sch.id/index.php/jirsi/index%0Ahttps://creativecommons.org/licenses/by-sa/4.0/>
- [2] B. Suharmanto, S. Kurnia, H. G. Prabowo, and ..., "Klasifikasi Tweet Cyberbullying dengan Menggunakan Algoritma Random Forest," ... *Ilmu Komput. dan ...*, pp. 512–520, 2022, [Online]. Available: <https://conference.upnvj.ac.id/index.php/senamika/article/view/2207%0Ahttps://conference.upnvj.ac.id/index.php/senamika/article/download/2207/1683>
- [3] F. Farasalsabila, E. Utami, and H. Hanafi, "Deteksi Cyberbullying Menggunakan BERT dan Bi-LSTM," *J. Teknol.*, vol. 17, no. 1, pp. 1–6, 2024, doi: 10.34151/jurtek.v17i1.4636.
- [4] W. Athira Luqyana, I. Cholissodin, and R. S. Perdana, "Analisis Sentimen Cyberbullying pada Komentar Instagram dengan Metode Klasifikasi Support Vector Machine," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 11, pp. 4704–4713, 2018, [Online]. Available: <http://j-ptiik.ub.ac.id>

- [5] A. Handika Permana, F. Rakhmat Umbara, and F. Kasyidi, "Klasifikasi Penyakit Jantung Tipe Kardiovaskular Menggunakan Adaptive Synthetic Sampling dan Algoritma Extreme Gradient Boosting," *Build. Informatics, Technol. Sci.*, vol. 6, no. 1, pp. 499–508, 2024, doi: 10.47065/bits.v6i1.5421.
- [6] I. Wardhana, Musi Ariawijaya, Vandri Ahmad Isnaini, and Rahmi Putri Wirman, "Gradient Boosting Machine, Random Forest dan Light GBM untuk Klasifikasi Kacang Kering," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 1, pp. 92–99, 2022, doi: 10.29207/resti.v6i1.3682.
- [7] E. Fitri, "Analisis Perbandingan Metode Regresi Linier, Random Forest Regression dan Gradient Boosted Trees Regression Method untuk Prediksi Harga Rumah," *J. Appl. Comput. Sci. Technol.*, vol. 4, no. 1, pp. 58–64, 2023, doi: 10.52158/jacost.v4i1.491.
- [8] M. R. Kurniawanda and F. A. T. Tobing, "Analysis Sentiment Cyberbullying In Instagram Comments with XGBoost Method," *IJNMT (International J. New Media Technol.)*, vol. 9, no. 1, pp. 28–34, 2022, doi: 10.31937/ijnmt.v9i1.2670.
- [9] M. Ridwansyah and H. Zakaria, "Implementasi Algoritma Gradient Boosting Pada Aplikasi Hutang Piutang Perorangan Secara Berbasis Web Untuk Meningkatkan Akurasi Prediksi Pelunasan Hutang (Studi Kasus : PT Naila Kreasi Mandiri)," *JURIHUM J. Inov. dan Hum.*, vol. 1, no. 4, pp. 440–451, 2023, [Online]. Available: <https://jurnalmahasiswa.com/index.php/jurikum>
- [10] I Putu Ramayasa, I Gusti Ayu Desi Saryanti, I Komang Dharmendra, and Edwar, "Perbandingan Metode Vektorisasi Pada Analisa Sentiment, Studi Kasus : Cyberbullying Pada Komentar Instagram," *J. Teknol. Inf. dan Komput.*, vol. 9, no. 5, pp. 505–512, 2023, doi: 10.36002/jutik.v9i5.2645.