# PENANGANAN KETIDAKSEIMBANGAN DATA PADA KLASIFIKASI PENGADUAN MASYARAKAT

**I Gusti Ngurah Ady Kusuma [1)], I Made Pradipta [2)], I Made Ari Santosa[3)]**
**I Komang Dharmendra[4)]**
Program Studi Sistem Komputer[1)2)3)] Program Studi Sistem Informasi[4)]
Fakultas Informatika dan Komputer, Fakultas Bisnis dan Vokasi
ITB STIKOM Bali, Denpasar, Bali
ady_kusuma@stikom-bali.ac.id[1)] madepradipta@stikom-bali.ac.id[2)] arisantosa@stikom-bali.ac.id[3)]
dharmendra@stikom-bali.ac.id[4)]

***ABSTRACT***

*Public complaints play a crucial role in improving the quality of institutional services. However, in processing complaint data, an imbalance issue often arises where the number of complaints in each class is not balanced. This research aims to address data imbalance in complaint classification using public complaint data from Denpasar City. Data imbalance can have a negative impact on classification, as models tend to be biased towards the majority class. To address this, oversampling models utilizing the SMOTE (Synthetic Minority Over-sampling Technique) and ADASYN (Adaptive Synthetic Sampling) techniques are employed. SMOTE and ADASYN are used to generate synthetic samples from the minority class in the dataset. Classification is performed using Naive Bayes Classifier (NBC), Support Vector Machine (SVM), and random forest algorithms. To evaluate model performance, evaluation metrics such as accuracy, precision, recall, and F1-score are utilized. Evaluation helps understand how well the models can accurately classify public complaints, particularly in the context of data imbalance. In addition to evaluation metrics, the processing time of each model is also calculated to assess the time required by the models. The results show that the use of SMOTE and ADASYN improves accuracy in the SVM and random forest algorithms. However, for the NBC algorithm, the use of sampling models actually decreases accuracy. Processing time is also an important factor in algorithm selection, with SVM having the longest processing time, NBC having the shortest processing time, and random forest falling in between the two.*

***Keywords:*** *SMOTE (Synthetic Minority Over-sampling Technique); ADASYN (Adaptive Synthetic Sampling); NBC (Naive Bayes Classifier); SVM (Support Vector Machine); random forest Classification.*

**ABSTRAK**

Pengaduan masyarakat memiliki peran penting dalam meningkatkan kualitas layanan lembaga. Namun, dalam pengolahan data pengaduan, sering terjadi ketidakseimbangan dimana jumlah pengaduan setiap kelas tidak seimbang. Penelitian ini bertujuan mengatasi ketidakseimbangan data dalam klasifikasi pengaduan dengan menggunakan data pengaduan masyarakat Kota Denpasar. Ketidakseimbangan data dapat berdampak negatif pada klasifikasi, model cenderung menjadi bias terhadap kelas mayoritas. Untuk mengatasinya dapat menggunakan model *oversampling* menggunakan teknik SMOTE (*Synthetic Minority Over-sampling Technique*) dan ADASYN (*Adaptive Synthetic Sampling*). SMOTE dan ADASYN digunakan untuk menghasilkan sampel sintetis dari kelas minoritas dalam dataset. Klasifikasi menggunakan NBC (*Naive Bayes Classifier*), SVM (*Support Vector Machine*), dan *random forest*. Untuk mengevaluasi performa model, digunakan matriks evaluasi akurasi, presisi, *recall*, dan *F1-Score*. Evaluasi membantu dalam memahami sejauh mana model-model dapat mengklasifikasikan pengaduan masyarakat dengan tepat, terutama dalam ketidakseimbangan data. Selain matriks evaluasi, juga dihitung waktu dari setiap model untuk mengetahui waktu proses yang dibutuhkan oleh model. Hasil menunjukkan penggunaan SMOTE dan ADASYN meningkatkan nilai akurasi pada algoritma SVM dan *random forest*. Namun, algoritma NBC, penggunaan model sampling justru menurunkan akurasi, waktu proses juga menjadi faktor penting dalam pemilihan algoritma. SVM memiliki waktu proses yang paling lama, NBC memiliki waktu proses yang paling pendek, dan *random forest* berada di antara keduanya.

***Kata Kunci:*** *SMOTE (Synthetic Minority Over-sampling Technique); ADASYN (Adaptive Synthetic Sampling); NBC (Naive Bayes Classifier); SVM (Support Vector Machine); random forest Classification*

## PENDAHULUAN

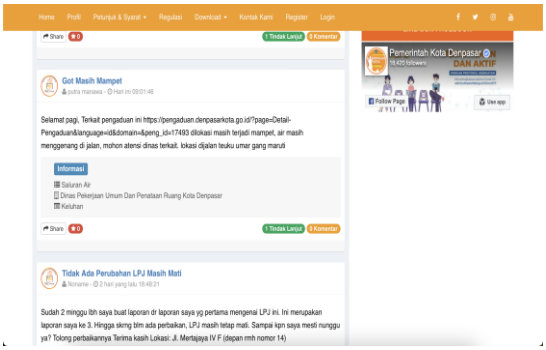
Pengaduan masyarakat memainkan peran penting dalam meningkatkan kualitas pelayanan dan produk yang diberikan oleh lembaga atau organisasi. Namun, dalam konteks pengolahan data pengaduan masyarakat, seringkali terjadi ketidakseimbangan data, di mana jumlah pengaduan pada setiap kelas atau label tidak seimbang. penelitian ini berfokus pada pengaduan masyarakat di Kota Denpasar, dengan tujuan untuk mengatasi masalah ketidakseimbangan data dalam klasifikasi pengaduan.

Ketidakseimbangan data dapat memiliki dampak negatif pada klasifikasi, di mana model cenderung menjadi bias terhadap kelas mayoritas[1]. Untuk mengatasi masalah ini, diterapkan model *oversampling* dengan menggunakan teknik SMOTE (*Synthetic Minority Over-sampling Technique*) dan ADASYN (*Adaptive Synthetic Sampling*). SMOTE dan ADASYN digunakan untuk menghasilkan sampel sintetis dari kelas minoritas dalam dataset pengaduan masyarakat[2], menciptakan keseimbangan yang lebih baik antara kelas mayoritas dan minoritas[3].

Setelah melakukan *oversampling*, dilanjutkan dengan klasifikasi menggunakan algoritma NBC (*Naive Bayes Classifier*)[4], SVM (*Support Vector Machine*), dan *random forest*[5]. Algoritma-algoritma ini dipilih karena telah terbukti efektif dalam klasifikasi pada berbagai tugas dan dataset menggunakan data pengaduan masyarakat yang dikumpulkan dari halaman website pengaduan Kota Denpasar pada halaman pengaduan.denpasarkota.go.id[6] sebagai dataset.

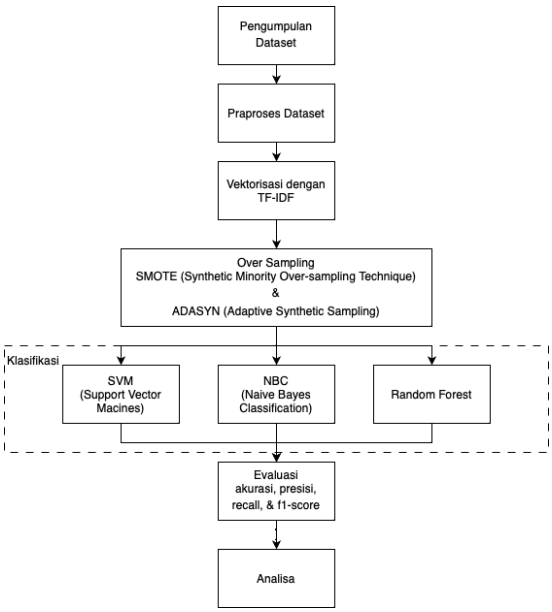
Untuk mengevaluasi performa model klasifikasi yang dikembangkan, digunakan matriks evaluasi yang meliputi akurasi, presisi, *recall*, dan *F1-Score*. Evaluasi ini membantu kami dalam memahami sejauh mana model-model tersebut dapat mengklasifikasikan dengan tepat pengaduan masyarakat, terutama dalam konteks ketidakseimbangan data. selain matrik evaluasi juga dihitung waktu dari setiap model klasifikasi untuk mengetahui waktu proses yang dibutuhkan setiap model klasifikasi[7].

Melalui penelitian ini, diharapkan dapat memberikan pemahaman yang lebih baik tentang penggunaan model sampling, khususnya kombinasi SMOTE dan ADASYN, dalam mengatasi ketidakseimbangan data pada pengaduan masyarakat. Selain itu, penelitian ini juga ingin membandingkan performa dari algoritma NBC, SVM, dan *random forest* dalam klasifikasi pengaduan masyarakat. Hasil penelitian ini diharapkan dapat memberikan wawasan yang berharga bagi pengembangan sistem pengaduan masyarakat yang lebih efektif dan efisien.




**Gambar 1.** Halaman Pengaduan Kota Denpasar

## METODE PENELITIAN




**Gambar 2.** Metode Penelitian

Adapun metode yang digunakan dalam pelaksanaan penelitian ini, dapat dilihat pada Gambar. 2.

a. Tahap awal penelitian dilakukan pengumpulan data pengaduan yang disampaikan oleh masyarakat kota denpasar secara publik pada halaman https://pengaduan.denpasarkota.go.id[6].
Data yang digunakan adalah data pengaduan teks dengan jumlah data 10306, yang terbagi menjadi kelas Keluhan, Usul / Saran, Pertanyaan, dan Informasi. Ratio dataset dapat dilihat pada tabel 1.

**Tabel 1.** Ratio Dataset

| **Kelas** | **Jumlah** |
|---|---|
| Keluhan | 8256 |
| Usul / Saran | 794 |
| Pertanyaan | 700 |
| Informasi | 552 |
| **Total** | **10306** |

b. Dilanjutkan dengan praproses pada dataset, Tahap *text processing* diperlukan untuk membersihkan sumber data dari data yang tidak diperlukan[8]. Proses ini bertujuan agar data yang digunakan nantinya bersih dari sesuatu yang tidak berpengaruh atau noise[9]. *Text processing* juga bertujuan agar data yang digunakan memiliki dimensi lebih kecil sehingga bisa diolah lebih lanjut[10]. Tahapan *text processing* yang dilakukan yaitu *case folding, cleansing, slangword correction, tokenizing, stopword removal*, dan *stemming*. Tabel 2 menunjukkan kalimat pengaduan sebelum prapraoses dan setelah praproses.

**Tabel 2. Tabel Perbandingan Praproses Data**

| Jenis Pengaduan | Keluhan |
|---|---|
| Data Sebelum praproses | |
| Saluran DSDP Mampet, air pembuangan tidak dapat mengalir sehingga meluap di halaman rumah | |
| | |
| Data Setelah praproses | |
| salur dsdp mampet air buang alir luap halaman rumah | |

c. Setelah praproses dilanjutkan dengan melakukan vektorisasi menggunakan TF-IDF, vektorisasi adalah proses mengubah teks menjadi representasi vektor yang dapat digunakan dalam pemrosesan komputasional[11][12]. Vektorisasi teks bertujuan untuk mengubah teks yang tidak terstruktur menjadi bentuk yang dapat dimengerti oleh model pembelajaran mesin[13]. Ini memungkinkan model untuk mengambil manfaat dari metode pembelajaran mesin yang berbasis numerik. Salah satu teknik vektorisasi yang umum digunakan dalam teks mining adalah TF-IDF (*Term Frequency-Inverse Document Frequency*). Dalam TF-IDF, teks dianalisis berdasarkan frekuensi kemunculan kata-kata individual dalam data pengaduan[14]. Dilanjutkan dengan membagi dataset dengan ratio 80% data latih, 20% data uji yang dapat dilihat pada tabel 3.

**Tabel 3.** Ratio Data

| **Ratio Data** | **Jumlah** |
|---|---|
| Data latih (80 %) | 8256 |
| Data uji (20 %) | 2064 |
| **Total** | **10306** |

d. Untuk menangani ketidakseimbangan data pada kelas keluhan, akan diterapkan metode *oversampling* SMOTE (*Synthetic Minority Over-sampling Technique*) untuk menangani ketidakseimbangan tersebut. Akan dibuat *instance* SMOTE dan menerapkannya pada subset data pelatihan dan label kelas yang telah dibagi sebelumnya. SMOTE adalah teknik *oversampling* yang menghasilkan sampel sintetis dari kelas minoritas[15]. Digunakan untuk mendapatkan set data latihan yang seimbang secara sintetis atau hampir seimbang antara kelas-kelasnya, yang kemudian digunakan untuk melatih klasifikasi. Sampel SMOTE adalah kombinasi linear dari dua sampel yang serupa dari kelas minoritas (**x** dan $\mathbf{x^R}$) dan didefinisikan pada persamaan 1.

$$s = x + u \cdot (x^R - x) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

dengan $\mathbf{0 \le u \le 1;}$ $\mathbf{x^R}$ dipilih secara acak dari 5 kelas terdekat kelas minoritas dari **x**.
Dilanjutkan dengan menerapkan metode *oversampling* ADASYN (*Adaptive Synthetic Sampling*) pada hasil *oversampling* SMOTE sebelumnya untuk mempertimbangkan ketidakseimbangan kelas yang mungkin masih ada. Hal ini akan menghasilkan sampel sintetis tambahan berdasarkan tingkat ketidakseimbangan dalam setiap fitur.

e. Setelah proses *oversampling* akan dilanjutkan dengan membangun model dengan menggunakan algoritma SVM, NBC, dan *random forest*, dimana akan dibangun 3 model yang berbeda untuk mengetahui perbandingan hasil *oversampling* dalam setiap model dan juga dibandingkan dengan model yang dibangun tanpa menggunakan data hasil *oversampling*.

SVM (*Support Vector Machine*) adalah algoritma pembelajaran mesin yang digunakan untuk melakukan klasifikasi[16]. Dalam konteks pengaduan, SVM dapat digunakan untuk mengklasifikasikan pengaduan menjadi beberapa kategori atau label yang relevan[17]. SVM bekerja dengan membangun sebuah model yang dapat memisahkan pengaduan menjadi dua atau lebih kelas yang berbeda. Model SVM mencari *hyperplane* (bidang pemisah) yang

optimal, yang dapat memaksimalkan margin (jarak) antara pengaduan dari kelas yang berbeda. Margin ini adalah jarak antara *hyperplane* dan titik pengaduan terdekat dari setiap kelas.

Dalam konteks pengaduan, data pengaduan biasanya direpresentasikan sebagai fitur-fitur yang relevan, seperti teks pengaduan, kategori pengaduan, atau atribut lain yang berkaitan. SVM kemudian menggunakan fitur-fitur ini untuk membangun model yang dapat memisahkan pengaduan menjadi kelas-kelas yang sesuai. Pada tahap pelatihan, SVM memilih sejumlah pengaduan sebagai support vectors, yaitu pengaduan-pengaduan yang berada paling dekat dengan *hyperplane*. Support vectors ini merupakan titik-titik penting yang mempengaruhi pembentukan *hyperplane* dan menentukan keputusan klasifikasi. Persamaan 2 adalah persamaan untuk SVM.

$$y = f(x) = \text{sgn}(w \cdot x + b) \quad \ldots\ldots\ldots\ldots(2)$$

Di mana:

y : hasil klasifikasi (kelas yang diprediksi) dari SVM.
f : fungsi keputusan SVM.
x : vektor fitur masukan.
w : vektor bobot.
b : bias.
sgn : fungsi signum yang menghasilkan nilai +1 atau -1 tergantung pada tanda dari argumen yang diberikan.

Setelah model SVM terlatih, pengaduan baru dapat diklasifikasikan dengan menghitung posisi relatifnya terhadap *hyperplane*. Pengaduan akan diatributkan ke salah satu kelas berdasarkan posisinya terhadap *hyperplane*.

NBC (*Naive Bayes Classifier*) metode yang berbasis pada teorema Bayes dan mengasumsikan independensi kondisional dari setiap fitur dalam data. Dalam konteks klasifikasi pengaduan, NBC dapat digunakan untuk mengklasifikasikan pengaduan ke dalam beberapa kategori atau label yang relevan. NBC menggunakan probabilitas dan statistik untuk menghasilkan keputusan klasifikasi[18][4].

NBC bekerja dengan mempelajari distribusi probabilitas dari setiap fitur dalam data pengaduan yang terkait dengan kelas-kelas yang ada. Fitur-fitur ini dapat berupa atribut, kata-kata dalam teks pengaduan, atau kategori-kategori tertentu yang berkaitan dengan pengaduan. Selama proses pelatihan, NBC menghitung probabilitas masing-masing fitur berdasarkan kelas yang sesuai. Kemudian, saat melakukan klasifikasi, NBC menghitung probabilitas bahwa suatu pengaduan termasuk dalam setiap kelas berdasarkan fitur-fitur yang diamati.

NBC menggunakan teorema *Bayes* untuk menghitung probabilitas *posterior*, yaitu probabilitas bahwa suatu pengaduan termasuk dalam suatu kelas tertentu, berdasarkan probabilitas prior dan probabilitas likelihood. Probabilitas prior adalah probabilitas awal suatu pengaduan termasuk dalam suatu kelas sebelum fitur-fitur diamati, sedangkan probabilitas *likelihood* adalah probabilitas bahwa fitur-fitur diamati muncul dalam kelas tersebut. Persamaan 3 adalah persamaan untuk NBC.

$$y = c \in C\big(P(c) \cdot P(x \in X \mid c)(x)\big) \ldots(3)$$

Di mana:

y : hasil klasifikasi (kelas yang diprediksi) dari NBC.
c : kelas yang mungkin untuk pengaduan.
C : himpunan semua kelas yang mungkin.
P(c) : probabilitas prior dari kelas c.
x : vektor fitur masukan.
X : ruang fitur yang mungkin.
P(x | c) : probabilitas likelihood dari fitur x terkait dengan kelas c.
argmax : fungsi yang menghasilkan kelas dengan probabilitas tertinggi.

NBC mengasumsikan bahwa fitur-fitur dalam pengaduan adalah independen satu sama lain, meskipun dalam kenyataannya mungkin ada korelasi antara fitur-fitur tersebut. Meskipun demikian, NBC masih sering digunakan karena kecepatan dan keandalannya dalam klasifikasi, terutama pada dataset dengan dimensi fitur yang besar.

*Random forest* (Hutan Acak) merupakan gabungan dari beberapa pohon keputusan yang bekerja secara independen dan menghasilkan keputusan klasifikasi berdasarkan mayoritas suara dari pohon-pohon tersebut[14][5]. Dalam konteks klasifikasi pengaduan, *random forest* dapat digunakan untuk mengklasifikasikan pengaduan ke dalam berbagai kategori atau label yang relevan. *Random forest* menggabungkan prediksi dari setiap pohon keputusan yang terbentuk secara acak untuk menghasilkan keputusan klasifikasi akhir.

Pohon keputusan dalam *Random forest* adalah struktur pemutusan keputusan yang menggambarkan alur logika berdasarkan fitur-fitur pengaduan. Pohon-pohon tersebut dibentuk secara acak dari subset acak dari fitur-fitur dan pengaduan yang diambil

dengan penggantian (*bootstrap*). Setiap pohon dalam *Random forest* melakukan klasifikasi berdasarkan aturan-aturan yang terbentuk pada setiap simpul dan daun pohon seperti yang ditunjukkan pada persamaan 4.

$$y = c \in C(f(x)) \quad \ldots\ldots\ldots\ldots\ldots\ldots(4)$$

Di mana:

y : hasil klasifikasi (kelas yang diprediksi) dari *Random forest*.

c : kelas yang mungkin untuk pengaduan.

C : himpunan semua kelas yang mungkin.

f : fungsi prediksi *Random forest*.

x : vektor fitur masukan.

Selama proses pelatihan, *Random forest* membentuk banyak pohon keputusan dengan menggunakan teknik resampling dan pembentukan pohon yang acak. Pada saat klasifikasi, setiap pohon memberikan suara untuk kategori kelas yang relevan berdasarkan fitur-fitur yang diamati. Keputusan klasifikasi akhir diambil berdasarkan mayoritas suara dari semua pohon dalam *Random forest*.

f. Untuk menganalisis hasil dan memastikan bahwa ketidakseimbangan kelas telah ditangani dengan baik, akan digunakan beberapa metrik evaluasi klasifikasi yang relevan, seperti akurasi, presisi, *recall*, dan *F1-Score*.

akurasi dilakukan dengan mengukur sejauh mana model mampu mengklasifikasikan dengan benar semua kelas, dinyatakan sebagai rasio prediksi yang benar (TP + TN) dengan total sampel yang ditunjukkan oleh persamaan 5.

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

Presisi Mengukur sejauh mana pengaduan yang diklasifikasikan sebagai positif (kelas yang benar) benar-benar positif, dengan persamaan 6 menunjukkan proses penentuan nilai presisi.

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

*Recall* dilakukan dengan mengukur sejauh mana model mampu mendeteksi pengaduan yang sebenarnya positif (kelas yang benar). Persamaan untuk menentukan *recall* dapat dilihat pada persamaan 7.

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

*F1-Score* dilakukan dengan menggabungkan presisi dan *recall* menjadi satu skor yang mencerminkan keseimbangan antara keduanya. Persamaan 8 menampilkan persamaan untuk menampilkan *F1-Score*.

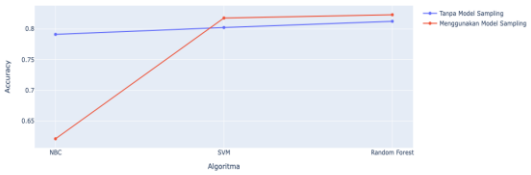
$$F^{\wedge}3 = 2 \text{ Precision} Recall \; / \; (\text{Precision} + Recall) \; ) \quad \ldots\ldots\ldots\ldots\ldots\ldots(8)$$

## HASIL DAN PEMBAHASAN

Pada penelitian ini dibangun 6 model dengan 3 algortima dan 2 skenario, dimana algoritma yang digunakan adalah SVM, NBC, dan *random forest*. Sedangkan untuk skenario yang digunakan adalah model tanpa menggunakan metode sampling dan menggunakan metode sampling (SMOTE dan ADASYN). Untuk menguji model yang dibangun akan digunakan matrix evaluasi akurasi, presisi, *recall*, dan *F1-Score*.

### 1. Akurasi

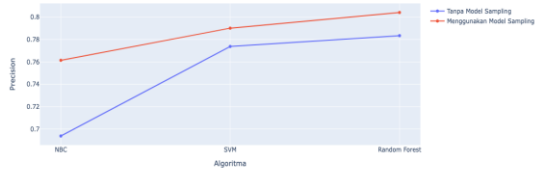
Pada gambar 3 menampilkan akurasi dari model yang tidak menggunakan model sampling dan menggunakan model sampling, Nilai akurasi tertinggi dihasilkan pada algoritma *Random forest* yang menggunakan model sampling dengan nilai akurasi sebesar 0.823159, dan nilai akurasi terendah didapatkan pada algoritma NBC yang menggunakan model sampling dengan nilai akurasi sebesar 0.621124.




**Gambar 3.** Akurasi Pengujian

### 2. Presisi

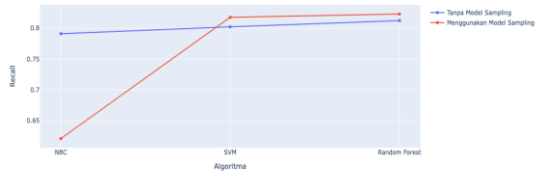
Pada matrix evaluasi presisi menunjukkan pada gambar 4 menunjukkan model dengan menggunakan model sampling menghasilkan nilai presisi yang lebih tinggi dibandingkan dengan model tanpa menggunakan model sampling, *Random forest* dengan menggunakan model sampling menghasilkan nilai presisi tertinggi sebesar 0.804127, sedangkan model NBC tanpa menggunakan model sampling menghasilan nilai presisi terendah sebesar 0.693820.




**Gambar 4.** Presisi Pengujian

### 3. *Recall*

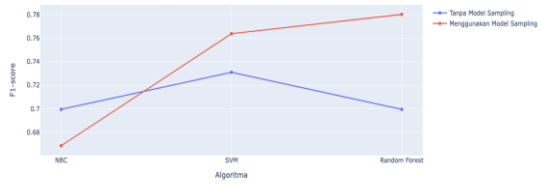
pada Matrix Evaluasi *recall* yang ditujukkan pada gambar 5 menampikan grafik hasil yang mirip dengan akurasi dimana nilai *recall* sebesar 0.823159 didapatkan oleh *Random forest* dengan menggunakan model sampling, sedangkan *Random forest* tanpa menggunakan model sampling menghasilkan nilai *recall* sebesar 0.812500 yang masih nilai tertinggi jika dibandingkan dengan model NBC dan SVM yang sama-sama tidak menggunakan model sampling.




**Gambar 5.** *Recall* Pengujian

### 4. *F1-Score*

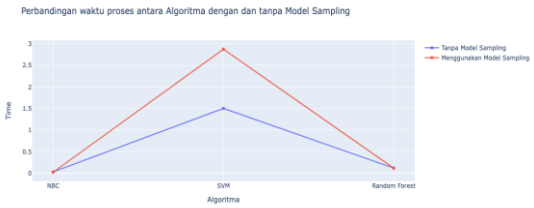
Pada *F1-Score* yang ditampilkan pada gambar 6 menunjukkan hasil yang berbeda dengan matrix evaluasi yang lainnya, dimana pada model *random forest* tanpa menggunakan model sampling mendapatkan nilai akurasi sebesar 0.699430, nilai yang sama dengan model NBC tanpa model sampling. Meskipun nilai terendah dihasilkan oleh model NBC dengan model sampling dengan nilai 0.668442, namun nilai *F1-Score* pada model *random forest* tanpa menggunakan model sampling lebih rendah dibandingkan nilai matrix evaluasi pada model yang sama.




**Gambar 6.** *F1-Score* Pengujian

### 5. Waktu Proses

Gambar 7 menunjukkan perbedaan waktu proses antara algoritma SVM, NBC, dan *Random forest* menunjukkan SVM memiliki waktu proses yang lebih lama pada semua model, SVM memiliki kompleksitas waktu yang tinggi, terutama saat menghitung pemisah maksimal (*hyperplane*) antara kelas-kelas data.




**Gambar 7.** Waktu Proses

Terutama ketika jumlah fitur atau dimensi data meningkat, sehingga pada penerapan model sampling waktu proses mengalami peningkatan sebesar 1.37 detik.

Pada NBC memiliki waktu proses yang paling pendek dikarenakan NBC adalah algoritma yang sederhana dan memiliki kompleksitas waktu yang rendah. Ini karena NBC hanya menghitung probabilitas dasar dan mengalikan probabilitas fitur untuk menghasilkan probabilitas kelas. Ini ditunjukkan dengan waktu proses pada model NBC tanpa menggunakan model sampling paling kecil dengan durasi 0.030465 detik. Tabel 4 menunjukkan waktu proses semua model yang dibuat pada penelitian.

### 6. Analisa

Tabel matrix evaluasi pada tabel 4 menunjukkan bahwa *random forest* menunjukkan waktu proses yang berada diantara SVM dan NBC, baik pada model tanpa penggunaan model sampling dengan durasi 0.118481 detik, dan pada model yang menggunakan model sampling didapatkan durasi sebesar 0.114090 detik, dimana dengan penggunaan model sampling mengalami peningkatan waktu proses sebesar 0.0043907. Perbedaan durasi waktu ini disebabkan karena pada *Random forest* melibatkan pembangunan banyak pohon keputusan (decision tree) dan penggabungan hasil prediksi dari masing-masing pohon. Waktu proses *Random forest* dapat dipengaruhi oleh jumlah pohon yang dibangun, kedalaman pohon, dan jumlah fitur yang digunakan.

Pada penggunaan model sampling untuk algoritma SVM dan *Random forest* mengalami peningkatan nilai akurasi, kecuali pada algoritma NBC yang mengalami penurunan nilai akurasi, pada pengujian yang telah diterapkan menunjukkan secara keseluruhan matrix evaluasi menunjukkan model *random forest* menggunakan model sampling menunjukkan hasil tertinggi dibandingkan dengan model lain yang dibuat. Algoritma *Random forest* menunjukkan performa yang baik dengan hasil tertinggi pada 2 skenario yang diterapkan dengan waktu proses yang berada diantara SVM dan NBC.

**Tabel 4. Tabel Hasil Pengujian**

| **Algorithm** | ***Accuracy*** | ***Precision*** | ***Recall*** | ***F1-Score*** | **Waktu Proses** |
|---|---|---|---|---|---|
| Tanpa Menggunakan Model Sampling | | | | | |
| NBC | 0.791182 | 0.693820 | 0.791182 | 0.699430 | **0.030465** |
| SVM | 0.802326 | 0.773789 | 0.802326 | 0.731031 | 1.50152 |
| *Random forest* | 0.812500 | 0.783430 | 0.812500 | 0.699430 | 0.118481 |
| Menggunakan Model Sampling | | | | | |
| NBC | 0,621124 | 0,761407 | 0,621124 | 0,668442 | **0.022646** |
| SVM | 0,817829 | 0,790129 | 0,817829 | 0,763913 | 2.874124 |
| *Random forest* | **0,823159** | **0,804127** | **0,823159** | **0,780354** | 0.11409 |

## SIMPULAN

Penggunaan model sampling seperti SMOTE dan ADASYN, dapat memberikan dampak yang berbeda terhadap kinerja algoritma klasifikasi. Pada penelitian ini, pembangunan model tanpa menggunakan model sampling dan menggunakan model sampling memberikan perubahan tingkat akurasi. Pada penelitian berikutnya dapat digunakan algoritma selain SVM, NBC dan *random forest* untuk lebih mengetahui pengaruh penggunaan model sampling pada penanganan ketidakseimbangan dataset

Penggunaan model sampling tidak selalu meningkatkan kinerja algoritma, seperti pada model NBC, penggunaan model sampling justru menurunkan kinerja algoritma. Dimana Model NBC yang menggunakan model sampling justru menurunkan akurasi. Selain SMOTE dan ADASYN yang termasuk dalam Random *Oversampling*, bisa juga diterapkan model sampling lain, seperti Random Undersampling, atau Combination Sampling yang merupakan gabungan dari Random *Oversampling* dan Random Undersampling.

Selain akurasi, faktor lain seperti waktu proses juga perlu dipertimbangkan dalam pemilihan algoritma. Dalam penelitian ini, algoritma SVM (*Support Vector Machine*) menunjukkan waktu proses yang paling lama, NBC memiliki waktu proses yang paling pendek, dan *Random forest* berada di antara keduanya. Perbedaan waktu proses ini dapat dipengaruhi oleh kompleksitas algoritma, volume data, dan efisiensi implementasi

## DAFTAR PUSTAKA

[1] F. Abdulloh, A. Aminuddin, M. Rahardi, and S. Anggita, "Observation of Imbalance Tracer Study Data for Graduates Employability Prediction in Indonesia," International Journal of Advanced Computer Science and Applications, vol. 13, pp. 169–174, Sep. 2022, doi: 10.14569/IJACSA.2022.0130820.

[2] E. M. O. N. Haryanto, A. K. A. Estetikha, and R. A. Setiawan, "IMPLEMENTASI SMOTE UNTUK MENGATASI IMBALANCED DATA PADA SENTIMEN ANALISIS SENTIMEN HOTEL DI NUSA TENGGARA BARAT DENGAN MENGGUNAKAN ALGORITMA SVM," Informasi Interaktif, vol. 7, no. 1, Art. no. 1, Jan. 2022.

[3] F. Last, G. Douzas, and F. Bacao, "Oversampling for Imbalanced Learning Based on K-Means and SMOTE," Information Sciences, vol. 465, pp. 1–20, Oct. 2018, doi: 10.1016/j.ins.2018.06.056.

[4] R. Ardianto, T. Rivanie, Y. Alkhalifi, F. S. Nugraha, and W. Gata, "SENTIMENT ANALYSIS ON E-SPORTS FOR EDUCATION CURRICULUM USING NAIVE BAYES AND SUPPORT VECTOR MACHINE," Jurnal Ilmu Komputer dan Informasi, vol. 13, no. 2, pp. 109–122, Jul. 2020, doi: 10.21609/jiki.v13i2.885.

[5] V. A. Fitri, R. Andreswari, and M. A. Hasibuan, "Sentiment Analysis of Social Media Twitter with Case of Anti-LGBT Campaign in Indonesia using Naïve Bayes, Decision Tree, and Random Forest Algorithm," Procedia Computer Science, vol. 161, pp. 765–772, Jan. 2019, doi: 10.1016/j.procs.2019.11.181.

[6] "Pro Denpasar." https://pengaduan.denpasarkota.go.id/ (accessed Apr. 17, 2023).

[7] V. I. Santoso, G. Virginia, and Y. Lukito, "Penerapan Sentiment Analysis Pada Hasil Evaluasi Dosen Dengan Metode Support Vector Machine," Jurnal Transformatika, vol. 14, no. 2, p. 72, Jan. 2017, doi: 10.26623/transformatika.v14i2.439.

[8] komang dharmendra, K. O. Saputra, and I. N. Pramaita, "Analisa Sentiment Untuk

Opini Alumni Perguruan Tinggi," Majalah Ilmiah Teknologi Elektro, vol. 18, no. 2, pp. xxxx–xxxx, Jul. 2019, doi: 10.24843/MITE.2019.V18I02.P11.

[9] I. M. A. Agastya, "Pengaruh Stemmer Bahasa Indonesia Terhadap Peforma Analisis Sentimen Terjemahan Ulasan Film," Jurnal Tekno Kompak, vol. 12, no. 1, pp. 18–23, Feb. 2018, doi: 10.33365/JTK.V12I1.70.

[10] I. K. Dharmendra, N. N. U. Januhari, I. P. Ramayasa, and I. M. A. W. Putra, "Uji Komparasi Sentiment Analysis Pada Opini Alumni Terhadap Perguruan Tinggi," Jurnal Teknik Informatika UNIKA Santo Thomas, pp. 1–6, May 2022, doi: 10.54367/jtiust.v7i1.1748.

[11] M. P. Simatupang and D. P. Utomo, "ANALISA TESTIMONIAL DENGAN MENGGUNAKAN ALGORITMA TEXT MINING DAN TERM FREQUENCY-INVERSE DOCUMENT FREQUENCE (TF-IDF) PADA TOKO ALLMEEART," KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer), vol. 3, no. 1, Art. no. 1, Dec. 2019, doi: 10.30865/komik.v3i1.1697.

[12] U. Krzeszewska, A. Poniszewska-Marańda, and J. Ochelska-Mierzejewska, "Systematic Comparison of Vectorization Methods in Classification Context," Applied Sciences, vol. 12, no. 10, Art. no. 10, Jan. 2022, doi: 10.3390/app12105119.

[13] F. C. Kasih, A. Puspaningrum, and A. Ghozali, "RANCANG BANGUN APLIKASI WEB UNTUK ANALISIS SENTIMEN PENGGUNA TWITTER TERHADAP KINERJA PEMERINTAH PROVINSI JAWA BARAT MENGGUNAKAN NAIVE BAYES CLASSIFIER," SEMINAR TEKNOLOGI TERAPAN, vol. 1, no. 1, Art. no. 1, 2021, Accessed: Jun. 20, 2023. [Online]. Available: https://prosiding.polindra.ac.id/index.php/semitera/article/view/149-157

[14] B. B. Baskoro, I. Susanto, and S. Khomsah, "Analisis Sentimen Pelanggan Hotel di Purwokerto Menggunakan Metode Random Forest dan TF-IDF (Studi Kasus: Ulasan Pelanggan Pada Situs TRIPADVISOR)," INISTA (Journal of Informatics Information System Software Engineering and Applications), vol. 3, no. 2, Art. no. 2, Jun. 2021, doi: 10.20895/inista.v3i2.218.

[15] M. Umer et al., "Scientific papers citation analysis using textual features and SMOTE resampling techniques," Pattern Recognition Letters, vol. 150, pp. 250–257, Oct. 2021, doi: 10.1016/j.patrec.2021.07.009.

[16] R. Wati and S. Ernawati, "Analisis Sentimen Persepsi Publik Mengenai PPKM Pada Twitter Berbasis SVM Menggunakan Python," Jurnal Teknik Informatika UNIKA Santo Thomas, pp. 240–247, Nov. 2021, doi: 10.54367/JTIUST.V6I2.1465.

[17] R. Moraes, J. F. Valiati, and W. P. Gavião Neto, "Document-level sentiment classification: An empirical comparison between SVM and ANN," Expert Systems with Applications, vol. 40, no. 2, pp. 621–633, Feb. 2013, doi: 10.1016/J.ESWA.2012.07.059.

[18] T. Desyani, A. Saifudin, and Y. Yulianti, "Feature Selection Based on Naive Bayes for Caesarean Section Prediction," IOP Conf. Ser.: Mater. Sci. Eng., vol. 879, no. 1, p. 012091, Jul. 2020, doi: 10.1088/1757-899X/879/1/012091.