

Kolaborasi *Naïve Bayes* dan *AdaBoost* dalam Klasifikasi Bakteri *E.coli*

Ahlihi Masruro¹, Hastari Utama², Agung Triyadi³

^{1,2} Teknik Informatika, Universitas Amikom Yogyakarta

³ Informatika, Universitas Amikom Yogyakarta

¹ ahlihi.m@amikom.ac.id, ² utama@amikom.ac.id, ³ agung.triyadi@students.amikom.ac.id

INTISARI

Penelitian ini bertujuan untuk mengevaluasi kinerja algoritma *Naïve Bayes* yang di-boosting menggunakan *AdaBoost* dalam klasifikasi bakteri *E.coli*. *Naïve Bayes* sering dianggap sebagai *weak classifier* karena akurasi yang kurang optimal pada dataset yang kompleks, sehingga diperlukan metode *boosting* untuk meningkatkan performanya. *AdaBoost* dipilih karena kemampuannya memperkuat *weak classifiers* dengan menyesuaikan bobot pada data yang salah diklasifikasikan, sehingga hasil klasifikasi menjadi lebih akurat. Dataset *E.coli* yang digunakan dalam penelitian ini terdiri dari 7 fitur biologis yang menggambarkan karakteristik sinyal dan protein. Proses implementasi dilakukan menggunakan Python, Jupyter Notebook, serta library *scikit-learn* dan *imblearn*. Evaluasi kinerja dilakukan dengan menggunakan metode *10-fold cross-validation* dan *confusion matrix*. Hasil penelitian menunjukkan bahwa *Naïve Bayes* yang di-boosting dengan *AdaBoost* berhasil meningkatkan akurasi klasifikasi dari 76% menjadi 94%, serta meningkatkan rata-rata *cross-validation* dan *geometric mean*. Meskipun *AdaBoost* tidak selalu menjamin peningkatan kinerja pada setiap dataset, kolaborasi kedua algoritma ini terbukti mampu memberikan hasil yang lebih stabil dan akurat pada dataset *E.coli*. Penelitian ini menyimpulkan bahwa kombinasi *Naïve Bayes* dan *AdaBoost* efektif dalam meningkatkan performa klasifikasi dataset biologis yang kompleks.

Kata kunci: *AdaBoost*, *E.coli*, Klasifikasi, *Naive Bayes*.

ABSTRACT

This research aims to evaluate the performance of the *Naïve Bayes* algorithm boosted with *AdaBoost* in the classification of *E.coli* bacteria. *Naïve Bayes* is often considered a *weak classifier* due to its suboptimal accuracy on complex datasets, thus requiring *boosting* methods to enhance its performance. *AdaBoost* was chosen for its ability to strengthen *weak classifiers* by adjusting the weights of misclassified data, leading to more accurate classification results. The *E.coli* dataset used in this study consists of 7 biological features that describe signal and protein characteristics. The implementation process was carried out using Python, Jupyter Notebook, along with the *scikit-learn* and *imblearn* libraries. Performance evaluation was conducted using the *10-fold cross-validation* method and a *confusion matrix*. The results show that *Naïve Bayes* boosted with *AdaBoost* successfully improved classification accuracy from 76% to 94%, as well as increasing the average *cross-validation* score and *geometric mean*. Although *AdaBoost* does not always guarantee improved performance for every dataset, the collaboration of these two algorithms proved to provide more stable and accurate results for the *E.coli* dataset. This research concludes that the combination of *Naïve Bayes* and *AdaBoost* is effective in enhancing classification performance for complex biological datasets.

Keywords: *AdaBoost*, *E.coli*, Classification, *Naive Bayes*.

1. PENDAHULUAN

Penelitian dalam bidang klasifikasi data telah berkembang pesat, terutama dalam penggunaan metode-metode machine learning yang semakin kompleks dan adaptif. Salah satu algoritma yang banyak digunakan dalam klasifikasi adalah *Naïve Bayes*, yang memiliki keunggulan dalam kemampuannya menangani data dengan jumlah variabel yang besar serta proses komputasi yang cepat (Byna & Basit, 2020). Namun, meskipun efisien, *Naïve Bayes* sering dianggap sebagai "weak classifier" karena cenderung kurang akurat ketika dihadapkan pada data yang kompleks atau memiliki pola yang tidak terlalu teratur (Jahromi & Taheri, 2017). Untuk meningkatkan performa algoritma ini, metode boosting seperti *AdaBoost* sering kali diterapkan sebagai teknik penguatan untuk menghasilkan hasil klasifikasi yang lebih optimal (Mahendra et al., 2024).

Boosting merupakan teknik meta-learning yang bertujuan memperkuat weak classifiers agar dapat menjadi strong classifiers (Li & Li, 2010; Wang et al., 2022). *AdaBoost* (*Adaptive Boosting*) bekerja dengan memberikan bobot lebih pada data yang diklasifikasikan secara salah oleh model awal, sehingga model selanjutnya lebih fokus pada kesalahan tersebut. Dengan cara ini, kesalahan klasifikasi dapat diminimalkan secara bertahap dan meningkatkan performa keseluruhan model. Kolaborasi antara *Naïve Bayes* dan *AdaBoost* diharapkan dapat memberikan hasil klasifikasi yang lebih akurat, terutama dalam menangani dataset yang kompleks seperti dataset *E.coli*.

Bakteri *E.coli* adalah organisme yang sering digunakan dalam penelitian biologi dan genetika (Sprugnoli & Tonelli, 2019). Pengelompokan bakteri ini berdasarkan ciri-ciri biologisnya sangat penting, terutama dalam penelitian terkait pengenalan sinyal protein dan analisis struktur sel. Dataset *E.coli* yang digunakan dalam penelitian klasifikasi biasanya memiliki beberapa fitur yang menggambarkan karakteristik sinyal dan protein, seperti metode McGeoch's dan skor program prediksi wilayah membran. Dengan demikian, dataset *E.coli* ini menjadi tantangan yang baik untuk menguji kinerja algoritma klasifikasi dalam bidang bioinformatika.

Penelitian sebelumnya menunjukkan bahwa *Naïve Bayes* dapat memberikan hasil yang baik dalam berbagai aplikasi klasifikasi, namun masih memiliki keterbatasan dalam hal akurasi, terutama ketika dihadapkan dengan data yang mengandung banyak noise. (Putra et al., 2024) Untuk mengatasi masalah ini, penggunaan *AdaBoost* diharapkan

dapat meningkatkan kinerja *Naïve Bayes*, terutama dengan adanya teknik penyesuaian bobot yang membuat model lebih adaptif terhadap data yang sulit. Implementasi kolaborasi antara *Naïve Bayes* dan *AdaBoost* pada dataset *E.coli* diharapkan dapat memberikan wawasan baru tentang bagaimana kedua algoritma ini dapat bekerja bersama untuk meningkatkan akurasi klasifikasi (Utami & Wahono, 2015).

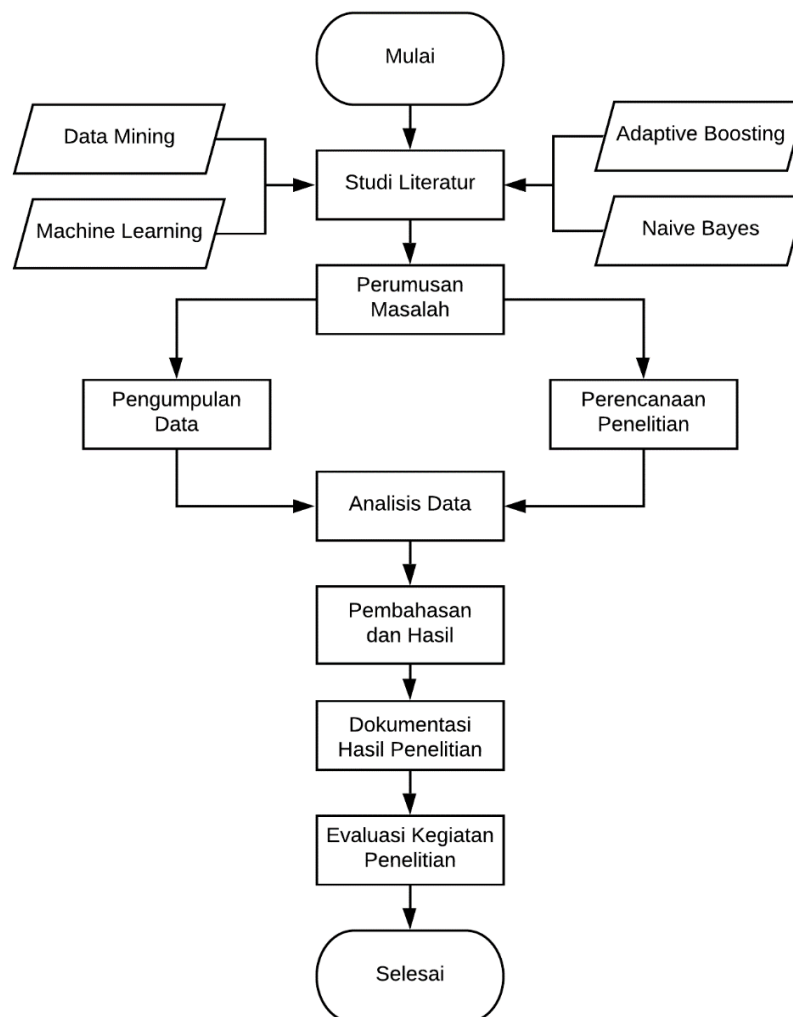
Oleh karena itu, penelitian ini akan memfokuskan pada kolaborasi *Naïve Bayes* dan *AdaBoost* dalam klasifikasi bakteri *E.coli*, dengan tujuan utama untuk melihat seberapa jauh peningkatan akurasi dan stabilitas model yang dihasilkan. Hasil penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan metode klasifikasi yang lebih efektif, khususnya dalam aplikasi bioinformatika, serta memberikan solusi untuk tantangan yang sering dihadapi dalam klasifikasi dataset biologis yang kompleks.

2. METODOLOGI

Implementasi algoritma *Naïve Bayes* yang dikombinasikan dengan metode boosting *Adaptive Boosting* pada dataset yang telah ditentukan. Proses implementasi dilakukan secara linear pada dataset menggunakan metode Min Max Normalization (Zhang et al., 2021). Model diuji menggunakan teknik K-Fold Cross Validation, sementara evaluasi akurasi dilakukan dengan metode Confusion Matrix yang diaplikasikan pada dataset. Pengujian ini dilakukan dengan memanfaatkan bahasa pemrograman Python dan lingkungan pengembangan terpadu (IDE) Jupyter Lab (Faridah et al., 2021).

Dataset *Ecoli* yang tersedia di UCI Machine Learning Repository yang berada pada alamat <https://archive.ics.uci.edu/dataset/39/Ecoli>. Dataset ini berfokus pada prediksi lokasi subseluler dari protein dalam organisme bakteri *E. coli*. Ini adalah dataset klasik yang sering digunakan dalam konteks pembelajaran mesin untuk klasifikasi multikelas. Tujuannya adalah untuk memprediksi ke mana protein-protein tersebut akan ditempatkan dalam sel berdasarkan beberapa atribut yang menjelaskan karakteristik fisik protein. Dataset ini berisi 8 atribut atau fitur yang diukur dari urutan protein. Atribut-atribut ini mencakup properti-properti seperti probabilitas alpha helix, pola distribusi asam amino, dan karakteristik lainnya yang terkait dengan struktur dan fungsi protein.

Secara keseluruhan, dataset *Ecoli* sangat relevan untuk penelitian dalam bioinformatika, pembelajaran mesin, dan ilmu data, khususnya dalam pengelompokan atau klasifikasi protein berdasarkan fitur-fitur fisik yang kompleks. Pada penelitian ini memiliki sebuah alur penelitian yang digunakan untuk menentukan tahap apa saja yang di lakukan oleh peneliti pada penelitian ini, sehingga bisa dilakukan hingga selesai. Alur penelitian ini ditunjukkan pada Gambar 1.



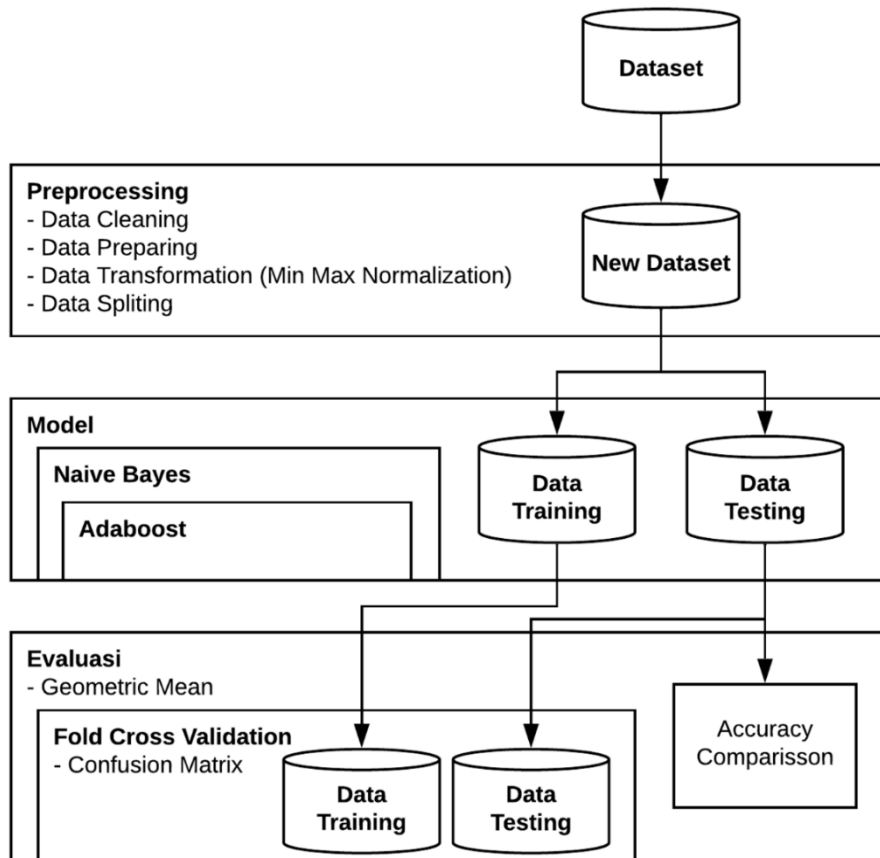
Gambar 1. Alur Penelitian

Alur penelitian yang ditampilkan pada gambar menunjukkan tahapan-tahapan sistematis yang diikuti dalam sebuah penelitian, khususnya terkait topik tentang algoritma *Naive Bayes* dan *Adaptive Boosting*, serta penggunaan teknik *Data Mining* dan *Machine Learning*. Berikut adalah penjelasan dari setiap langkah dalam alur penelitian tersebut:

1. **Mulai:** Ini adalah langkah awal dari penelitian, di mana ide atau gagasan awal ditentukan.
2. **Studi Literatur:** Peneliti memulai dengan melakukan studi literatur. Studi literatur mencakup pengumpulan informasi dari penelitian-penelitian sebelumnya terkait topik yang dibahas, seperti *Data Mining*, *Machine Learning*, *Naïve Bayes*, dan *Adaptive Boosting*.
3. **Perumusan Masalah:** Setelah memahami literatur, peneliti merumuskan masalah yang ingin diselesaikan. Langkah ini bertujuan untuk menentukan fokus penelitian dan apa yang ingin dicapai dari penelitian tersebut.
4. **Perencanaan Penelitian:** Setelah masalah dirumuskan, peneliti merencanakan bagaimana penelitian tersebut akan dilakukan, metode apa yang akan digunakan, dan dataset apa yang akan diolah.
5. **Pengumpulan Data:** Data yang diperlukan untuk penelitian dikumpulkan pada tahap ini. Sumber data bisa berasal dari dataset yang sudah ada atau data yang baru dikumpulkan untuk keperluan penelitian.
6. **Analisis Data:** Pada tahap ini, data yang telah dikumpulkan dianalisis menggunakan algoritma yang telah direncanakan, seperti *Naïve Bayes* yang dikombinasikan dengan *Adaptive Boosting*. Hasil dari analisis data ini akan menjadi dasar untuk evaluasi dan interpretasi hasil penelitian.
7. **Pembahasan dan Hasil:** Setelah analisis data dilakukan, hasil dari penelitian dibahas. Tahap ini mengulas apakah hasil yang diperoleh sesuai dengan hipotesis awal atau justru menunjukkan sesuatu yang berbeda. Selain itu, dibahas pula implikasi dari hasil penelitian tersebut.
8. **Dokumentasi Hasil Penelitian:** Pada tahap ini, hasil dan proses penelitian didokumentasikan. Ini penting untuk menjaga transparansi dan memungkinkan penelitian untuk ditinjau oleh orang lain.
9. **Evaluasi Kegiatan Penelitian:** Pada langkah ini, seluruh proses penelitian dievaluasi, termasuk metode yang digunakan, hasil yang diperoleh, serta kontribusi penelitian terhadap bidang yang sedang diteliti.

10. **Selesai:** Ini adalah tahap akhir penelitian, di mana seluruh langkah-langkah penelitian telah selesai dilakukan dan penelitian dinyatakan selesai.

Secara keseluruhan, alur ini menggambarkan tahapan yang sistematis dan terstruktur dalam melakukan penelitian, mulai dari perencanaan hingga penyelesaian dan evaluasi.



Gambar 2. Skenario Klasifikasi

Penelitian ini juga mencakup sebuah skenario klasifikasi untuk menyelesaikan masalah klasifikasi yang menggunakan metode boosting. Skenario tersebut dilakukan melalui empat tahapan utama, yaitu: pra-pemrosesan data, klasifikasi, ensemble classification, dan diakhiri dengan evaluasi. Tahapan-tahapan ini digambarkan lebih jelas melalui diagram alur klasifikasi pada Gambar 2. Gambar tersebut menggambarkan skenario klasifikasi yang menggunakan metode *Naïve Bayes* dan *Adaptive Boosting* (*AdaBoost*) untuk menyelesaikan masalah klasifikasi. Berikut penjelasan dari setiap tahap dalam alur skenario tersebut:

1. **Dataset:** Proses dimulai dengan pengumpulan dataset yang akan digunakan dalam penelitian.
2. **Preprocessing:** Dataset yang dikumpulkan kemudian diproses melalui beberapa tahapan pra-pemrosesan, yaitu:
 - a. **Data Cleaning:** Menghapus atau memperbaiki data yang tidak valid, duplikat, atau error.
 - b. **Data Preparing:** Menyiapkan data agar siap digunakan untuk proses klasifikasi, seperti pengorganisasian data yang diperlukan.
 - c. **Data Transformation (Min-Max Normalization):** Normalisasi data dilakukan untuk mengubah rentang nilai setiap fitur agar berada dalam rentang yang sama (biasanya antara 0 dan 1) menggunakan metode Min-Max Normalization.
 - d. **Data Splitting:** Dataset dipecah menjadi dua bagian: data training (pelatihan) dan data testing (pengujian).
3. **Model:** Pada tahap ini, model klasifikasi dibangun menggunakan dua algoritma:
 - a. **Naïve Bayes:** Algoritma klasifikasi dasar digunakan sebagai model dasar (weak classifier).
 - b. **AdaBoost:** Algoritma boosting diterapkan untuk memperkuat performa *Naïve Bayes*, mengubah *weak classifier* menjadi *strong classifier* dengan memberikan bobot lebih pada data yang sulit diklasifikasikan.
4. **Data Training dan Data Testing:** Dataset yang telah dibagi menjadi data training dan data testing digunakan pada tahap ini. Data training digunakan untuk melatih model, sementara data testing digunakan untuk menguji performa model.
5. **Evaluasi:** Pada tahap ini, model dievaluasi menggunakan metode:
 - a. **Fold Cross Validation:** Teknik evaluasi yang membagi dataset menjadi beberapa subset untuk menguji akurasi model secara lebih robust.
 - b. **Confusion Matrix:** Alat yang digunakan untuk mengevaluasi hasil klasifikasi dengan membandingkan prediksi dan data asli.

- c. **Geometric Mean:** Pengukuran performa model yang mempertimbangkan keseimbangan antara sensitivity dan specificity.
6. **Accuracy Comparison:** Setelah model diuji dan dievaluasi, dilakukan perbandingan akurasi antara model yang di-boosting dengan *AdaBoost* dan model dasar *Naïve Bayes*. Hasilnya menunjukkan apakah *AdaBoost* berhasil meningkatkan akurasi klasifikasi atau tidak.

Tabel 1. Dataset Bakteri *E.coli*

0	1	2	3	4	5	6	target
0.72	0.86	0.48	0.50	0.17	0.55	0.21	-1
0.63	0.75	0.48	0.50	0.64	0.73	0.66	-1
0.23	0.40	0.48	0.50	0.39	0.28	0.38	-1
0.89	0.55	0.48	0.50	0.51	0.72	0.76	-1
0.07	0.40	0.48	0.50	0.54	0.35	0.44	-1
0.00	0.51	0.48	0.50	0.35	0.67	0.44	-1
0.37	0.47	0.48	0.50	0.39	0.76	0.79	-1
0.44	0.56	0.48	0.50	0.50	0.46	0.54	-1
0.71	0.71	0.48	0.50	0.40	0.54	0.39	-1
0.00	0.38	0.48	0.50	0.42	0.48	0.55	-1

Secara keseluruhan, skenario ini menekankan pada peningkatan performa klasifikasi melalui penggunaan *AdaBoost* untuk memperbaiki kekurangan *Naïve Bayes* sebagai weak classifier, dengan melakukan evaluasi akurasi melalui beberapa teknik pengujian dan perbandingan.

Dataset *E. coli* yang tersedia di [UCI Machine Learning Repository](#) adalah dataset yang berfokus pada klasifikasi sub-seluler protein *E. coli*. Data ini sering digunakan untuk masalah klasifikasi dalam penelitian machine learning dan bioinformatika dimana ditunjukkan pada Tabel 1. Berikut adalah rincian dataset tersebut:

Dataset ini digunakan untuk memprediksi lokasi sub-seluler protein *E. coli* berdasarkan urutan fitur yang diperoleh dari analisis sinyal peptida. Klasifikasi lokasi sub-seluler penting dalam *biologi molekuler* untuk memahami fungsi protein dalam sel.

Dataset *E. coli* memiliki 8 fitur atau atribut yang menjelaskan karakteristik dari protein *E. coli*. Berikut adalah deskripsi fitur-fiturnya:

1. **McGeoch's method for signal sequence recognition (mcg)**: Skor pengenalan sinyal berdasarkan metode McGeoch untuk mengidentifikasi keberadaan urutan sinyal peptida.
2. **von Heijne's method for signal sequence recognition (gvh)**: Skor dari metode *von Heijne* yang digunakan untuk mengenali urutan sinyal.
3. **Score of Signal Peptidase II consensus sequence recognition (lip)**: Skor pengenalan urutan konsensus untuk *Signal Peptidase II von Heijne*.
4. **Charge of the N-terminus (chg)**: Informasi mengenai muatan pada *N-terminus lipoprotein* yang diprediksi.
5. **Amino acid content (aac)**: Skor dari kandungan asam amino yang diukur dari membran luar dan *protein periplasmik*.
6. **Program ALOM membrane spanning region prediction (alm1)**: Skor dari program ALOM yang memprediksi keberadaan wilayah rentang membran.
7. **ALOM prediction after excluding putative cleavable signal regions (alm2)**: Skor prediksi ALOM setelah mengeluarkan daerah sinyal yang bisa dibelah dari urutan protein.
8. **Class (class)**: Variabel target yang menyatakan lokasi sub-seluler dari protein. Kelas ini terdiri dari beberapa kategori lokasi sub-seluler seperti cp, im, pp, om, imU, imL, omL.

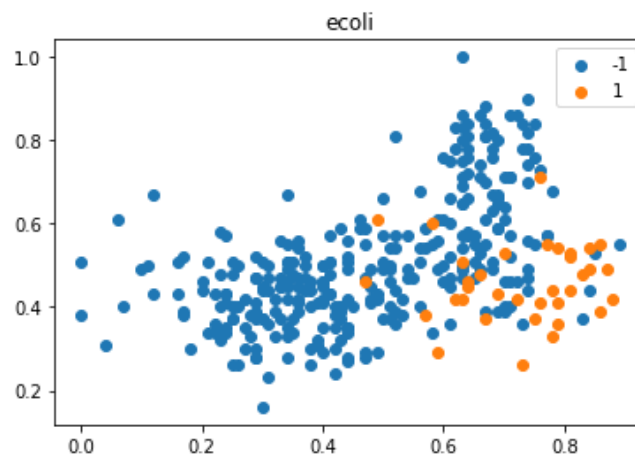
Informasi Lain:

- a. **Jumlah Sampel**: 336 entri.
- b. **Jumlah Fitur**: 7 fitur numerik yang merupakan hasil dari berbagai metode analisis, dan 1 fitur target (kelas).
- c. **Tugas**: Klasifikasi protein berdasarkan lokasi sub-selulernya.
- d. **Jumlah Kelas**: 8 kelas sub-seluler protein, yaitu cp (cytoplasm), im (inner membrane without signal sequence), pp (periplasm), om (outer membrane), imU (inner membrane with uncleavable signal sequence), imL (inner

membrane lipoprotein), omL (outer membrane lipoprotein), dan om (outer membrane lipoprotein).

3. HASIL DAN PEMBAHASAN

Bagian ini akan memaparkan hasil analisis dari implementasi algoritma *Naïve Bayes* yang di-boosting menggunakan metode *AdaBoost* pada dataset *E. coli*. Hasil-hasil tersebut mencakup performa model yang dievaluasi menggunakan metrik-metrik utama seperti akurasi, rata-rata *cross-validation*, dan geometric mean. Selain itu, perbandingan antara hasil klasifikasi yang dilakukan oleh *Naïve Bayes* tanpa boosting dan yang di-boosting dengan *AdaBoost* juga akan dijelaskan secara rinci. Pembahasan akan mengeksplorasi interpretasi dari hasil-hasil ini, termasuk peningkatan yang dihasilkan oleh penggunaan *AdaBoost* dalam memperbaiki kelemahan *Naïve Bayes* sebagai weak classifier, serta evaluasi dampak dari preprocessing dan teknik validasi yang diterapkan pada dataset. Analisis ini akan memberikan wawasan mengenai efektivitas kolaborasi kedua algoritma tersebut dalam klasifikasi data biologis yang kompleks.



Gambar 3.Distribusi kelas -1 dan 1 pada Dataset Bakteri *E.coli*

Berdasarkan Gambar 3 yang ditampilkan, distribusi kelas pada dataset *E. coli* terlihat terbagi menjadi dua kelas yang direpresentasikan dengan warna berbeda:

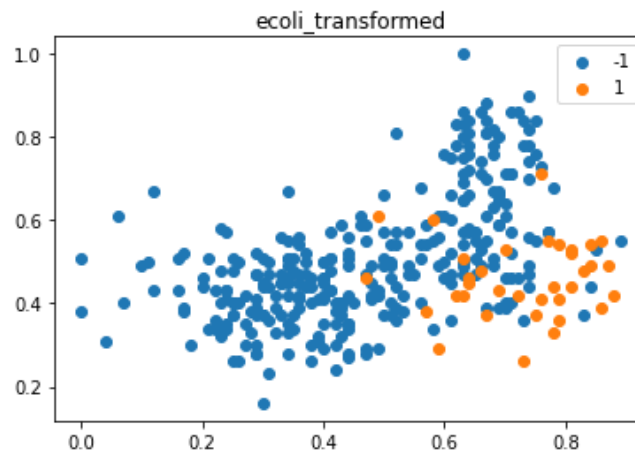
- a. **Kelas -1:** Diwakili oleh titik-titik berwarna biru.
- b. **Kelas 1:** Diwakili oleh titik-titik berwarna oranye.

Distribusi titik menunjukkan bahwa kelas -1 (biru) mendominasi sebagian besar area pada plot, menunjukkan bahwa data pada kelas ini lebih banyak daripada kelas 1

(oranye). Sementara itu, kelas 1 terlihat terkonsentrasi di area tertentu, meskipun jumlahnya lebih sedikit dibandingkan dengan kelas -1.

Secara visual, kita bisa melihat bahwa ada pola distribusi dari kedua kelas tersebut yang mengikuti tren tertentu pada dua dimensi dari dataset yang divisualisasikan. Kedua kelas ini tampak saling tumpang tindih di beberapa bagian, tetapi secara umum masih ada perbedaan yang cukup jelas antara area dominasi masing-masing kelas.

Distribusi ini mengindikasikan bahwa model klasifikasi harus mampu menangani ketidakseimbangan kelas yang ada, di mana data kelas -1 lebih dominan dibandingkan kelas 1, yang merupakan karakteristik umum dalam masalah klasifikasi dengan dataset tidak seimbang (imbalanced dataset).



Gambar 4. Hasil Normalisasi min Max pada Dataset Bakteri *E.coli*

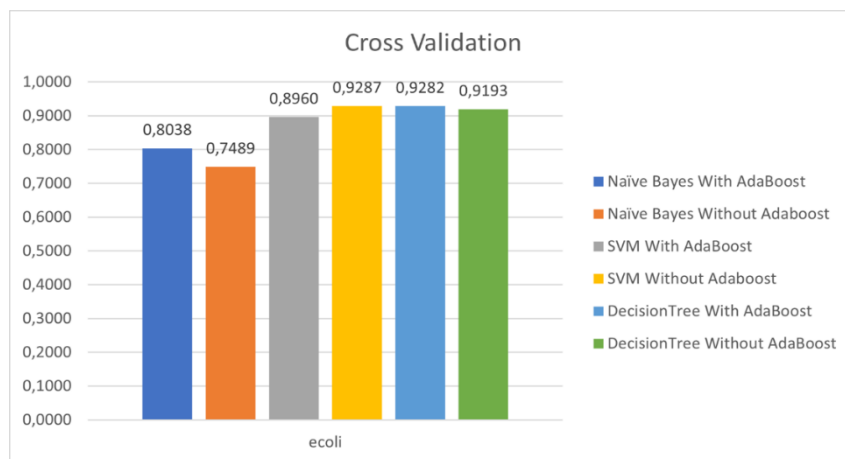
Pada Gambar 4 menunjukkan distribusi dataset bakteri *E. coli* setelah melalui proses normalisasi menggunakan metode Min-Max. Normalisasi ini berfungsi untuk mengubah nilai setiap fitur dalam dataset agar berada dalam rentang nilai 0 hingga 1, yang terlihat jelas pada sumbu x dan y grafik.

Normalisasi Min-Max: Proses ini memastikan bahwa setiap nilai fitur pada dataset *E. coli* berada dalam rentang 0 hingga 1, yang membantu mengatasi perbedaan skala di antara fitur-fitur yang mungkin memiliki nilai awal yang bervariasi. Ini penting agar setiap fitur memberikan kontribusi yang setara saat digunakan dalam model machine learning.

Kelas -1 (berwarna biru) dan Kelas 1 (berwarna oranye) masih memiliki distribusi yang sama seperti sebelumnya, tetapi nilai-nilai fitur dari kedua kelas tersebut telah

dinormalisasi. Kelas -1 tetap mendominasi dengan distribusi yang lebih luas, sementara kelas 1 terkonsentrasi di area tertentu.

Tumpang tindih antar kelas: Seperti pada grafik sebelumnya, ada beberapa area di mana kelas -1 dan kelas 1 saling tumpang tindih, meskipun sebagian besar data untuk setiap kelas terkonsentrasi di area yang berbeda. Secara keseluruhan, proses normalisasi ini tidak mengubah pola distribusi data, tetapi mengubah skala nilainya sehingga lebih sesuai untuk digunakan dalam algoritma machine learning, khususnya yang sensitif terhadap perbedaan skala fitur.



Gambar 5. Perbandingan Hasil Evaluasi dengan Cross Validation

Grafik pada Gambar 5 menunjukkan hasil evaluasi beberapa metode klasifikasi yang diukur menggunakan teknik *Cross Validation* pada dataset *E. coli*. Ada enam metode yang dibandingkan, yaitu *Naïve Bayes*, *SVM*, dan *Decision Tree*, masing-masing diuji dengan dan tanpa penggunaan *AdaBoost*. Berikut adalah penjelasan lebih lanjut mengenai hasilnya:

1. ***Naïve Bayes Without AdaBoost (Oranye)***: Algoritma *Naïve Bayes* tanpa boosting memiliki akurasi yang terendah di antara semua metode dengan nilai **0,7489**. Hal ini menunjukkan bahwa tanpa boosting, *Naïve Bayes* tidak mampu menangani dataset dengan baik.
2. ***Naïve Bayes With AdaBoost (Biru)***: Setelah di-boosting dengan *AdaBoost*, akurasi *Naïve Bayes* meningkat menjadi **0,8038**. Peningkatan ini menunjukkan bahwa

AdaBoost berhasil memperbaiki kinerja *Naïve Bayes*, meskipun hasilnya masih lebih rendah dibandingkan metode lainnya.

3. **SVM Without *AdaBoost* (Kuning)**: Algoritma SVM tanpa *AdaBoost* menghasilkan akurasi yang cukup tinggi, yaitu **0,9287**, menunjukkan bahwa SVM secara intrinsik kuat untuk klasifikasi dataset ini tanpa memerlukan boosting.
4. **SVM With *AdaBoost* (Abu-abu)**: Setelah SVM di-boosting dengan *AdaBoost*, hasil akurasinya sedikit menurun menjadi **0,8960**. Ini mengindikasikan bahwa *AdaBoost* tidak selalu memberikan peningkatan kinerja pada algoritma yang sudah cukup kuat seperti SVM.
5. **Decision Tree Without *AdaBoost* (Hijau)**: Algoritma Decision Tree tanpa *AdaBoost* menghasilkan akurasi sebesar **0,9193**, menunjukkan performa yang baik tanpa boosting.
6. **Decision Tree With *AdaBoost* (Biru muda)**: Setelah di-boosting, Decision Tree mencapai akurasi tertinggi sebesar **0,9282**, mendekati nilai SVM tanpa *AdaBoost*. Ini menunjukkan bahwa *AdaBoost* memberikan sedikit peningkatan pada Decision Tree.

Penggunaan *AdaBoost* secara umum meningkatkan kinerja *Naïve Bayes* dan Decision Tree, tetapi tidak selalu meningkatkan kinerja SVM. Algoritma SVM tanpa boosting menghasilkan akurasi terbaik secara keseluruhan, diikuti oleh Decision Tree yang di-boosting dengan *AdaBoost*. *Naïve Bayes* mendapatkan peningkatan yang signifikan dengan *AdaBoost*, tetapi masih kalah dibandingkan dengan SVM dan Decision Tree. Hasil ini menunjukkan bahwa pemilihan algoritma klasifikasi dan penerapan boosting sangat bergantung pada sifat dataset dan algoritma dasar yang digunakan.

Tabel 2. Nilai Akurasi data Train dan Test

No	Skenario Perbandingan	<i>Ecoli</i> - Accuracy	
		Train	Test
1	<i>Naïve Bayes With AdaBoost</i>	0,9403	0,9412
2	<i>Naïve Bayes Without AdaBoost</i>	0,7463	0,7647
3	SVM With <i>AdaBoost</i>	0,8881	0,9265
4	SVM Without <i>AdaBoost</i>	0,9291	0,9265
5	DecisionTree With <i>AdaBoost</i>	1,0000	0,8824
6	DecisionTree Without <i>AdaBoost</i>	1,0000	0,8971

Pada Tabel 2 tersebut menunjukkan hasil evaluasi perbandingan akurasi dari beberapa model machine learning pada dataset *Ecoli*, baik untuk data latih (train) maupun data uji (test). Model yang diuji meliputi *Naïve Bayes*, Support Vector Machine (SVM), dan Decision Tree, dengan dan tanpa penggunaan teknik ensemble *AdaBoost*.

1. ***Naïve Bayes dengan AdaBoost*** menghasilkan akurasi sebesar 0.9403 pada data latih dan 0.9412 pada data uji. Perbedaan kecil ini menunjukkan bahwa model memiliki generalisasi yang baik dan konsisten antara data latih dan uji.
2. ***Naïve Bayes tanpa AdaBoost*** menunjukkan akurasi yang lebih rendah, dengan 0.7463 atau 74% pada data latih dan 0.7647 atau 76% pada data uji. Peningkatan signifikan terlihat ketika *AdaBoost* diterapkan, yang menandakan bahwa *Naïve Bayes* lebih optimal dengan teknik ensemble ini.
3. ***SVM dengan AdaBoost*** memiliki performa yang baik dengan akurasi 0.8881 pada data latih dan 0.9265 pada data uji, yang menunjukkan bahwa penggunaan *AdaBoost* dapat meningkatkan kinerja SVM, terutama pada data uji.

4. KESIMPULAN

Dataset pada penelitian ini memiliki 7 fitur yang menjelaskan karakteristik biologis terkait urutan sinyal. Algoritma *Naïve Bayes* di implementasikan menggunakan *Python*, *Jupyter Notebook*, serta *library scikit-learn* dan *imblearn*. Evaluasi dilakukan dengan metode 10-fold *cross-validation* dan *confusion matrix*. Hasil penelitian menunjukkan bahwa *Naïve Bayes* yang di-boosting dengan *AdaBoost* meningkatkan akurasi klasifikasi dari 76% menjadi 94%, serta meningkatkan rata-rata *cross-validation* dan *geometric mean*. Meskipun demikian, *AdaBoost* tidak selalu memberikan peningkatan performa, karena hasilnya dipengaruhi oleh karakteristik dataset dan kehadiran noise. Secara keseluruhan, *Naïve Bayes* dengan *AdaBoost* mencapai performa yang baik dengan rata-rata *cross-validation* 87%, akurasi 94%, dan *geometric mean* 96% pada dataset *Ecoli*.

5. DAFTAR PUSTAKA

- Byna, A., & Basit, M. (2020). Penerapan Metode *AdaBoost* Untuk Mengoptimasi Prediksi Penyakit Stroke Dengan Algoritma *Naïve Bayes*. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, 9(3), 407–411. <https://doi.org/10.32736/sisfokom.v9i3.1023>

- Faridah, N., Dewi, C., & Soebroto, A. A. (2021). Hybrid Of *AdaBoost* Algorithm And *Naïve Bayes* Classifier On Selection Of Contraception Methods. *Journal of Enviromental Engineering and Sustainable Technology*, 8(2), 63–69. <https://doi.org/10.21776/ub.jeest.2021.008.02.6>
- Jahromi, A. H., & Taheri, M. (2017). A non-parametric mixture of Gaussian naive Bayes classifiers based on local independent features. *2017 Artificial Intelligence and Signal Processing Conference (AISP)*, 209–212. <https://doi.org/10.1109/AISP.2017.8324083>
- Li, W., & Li, Q. (2010). Using Naive Bayes with *AdaBoost* to Enhance Network Anomaly Intrusion Detection. *2010 Third International Conference on Intelligent Networks and Intelligent Systems*, 486–489. <https://doi.org/10.1109/ICINIS.2010.133>
- Mahendra, I. G. A. P., Wirawan, I. M. A., & Gunadi, I. G. A. (2024). Enhancement performance of the *Naïve Bayes* method using *AdaBoost* for classification of diabetes mellitus dataset type II. *International Journal of Advances in Applied Sciences*, 13(3), 733. <https://doi.org/10.11591/ijaas.v13.i3.pp733-742>
- Putra, I. M. A. A. D., Sunarya, I. M. G., & Gunadi, I. G. A. (2024). Perbandingan Algoritma Naive Bayes Berbasis Feature Selection Gain Ratio dengan Naive Bayes Kovenisional dalam Prediksi Komplikasi Hipertensi. *JTIM: Jurnal Teknologi Informasi Dan Multimedia*, 6(1), 37–49. <https://doi.org/10.35746/jtim.v6i1.488>
- Sprugnoli, R., & Tonelli, S. (2019). Novel Event Detection and Classification for Historical Texts. *Computational Linguistics*, 45(2), 229–265. https://doi.org/10.1162/coli_a_00347
- Utami, L. D., & Wahono, R. S. (2015). Integrasi Metode Information Gain untuk Seleksi Fitur dan *AdaBoost* untuk Mengurangi Bias pada Analisis Sentimen Review Restoran Menggunakan Algoritma Naive Bayes. *Journal of Intelligent Systems*, 1(2), 120–126. <http://journal.ilmukomputer.org/index.php?journal=jis&page=article&op=download&path%5B%5D=50&path%5B%5D=48>
- Wang, S., Ren, J., Lian, X., Bai, R., & Jiang, X. (2022). Boosting the Discriminant Power of Naive Bayes. *2022 26th International Conference on Pattern Recognition (ICPR)*, 4906–4912. <https://doi.org/10.1109/ICPR56361.2022.9956358>
- Zhang, J., Huang, W., Ji, D., & Ren, Y. (2021). Globally normalized neural model for joint entity and event extraction. *Information Processing and Management*, 58(5), 102636. <https://doi.org/10.1016/j.ipm.2021.102636>