

Verifikasi Tanda Tangan Menggunakan Machine Learning

Muhammad Fauzi Putra

Program Studi Teknik Mekatronika, Jurusan Teknik Elektro, Politeknik Negeri Batam

muhammadfauziputra@gmail.com

Abstract

A signature is a tool for authentication or proof of a person's identity in authorizing a document. Each person has unique and diverse signature characteristics, however signature forgery cases are common. Signature forgery renders documents invalid or illegitimate, potentially causing harm to the affected parties. Therefore, an accurate system for signature verification is essential. In this study, the method used for signature verification is the Naïve Bayes method, implemented using the Orange application by comparing accuracy and precision levels based on the Confusion Matrix of the processed dataset. This research aims to perform signature verification using machine learning, with data collected manually. The signatures will be written on HVS paper and photographed using a smartphone camera. The photos will then be edited, cropped, and focused solely on the signature. The experiment involves 500 signatures converted into JPEG (Joint Photographic Experts Group) format. Subsequently, the accuracy will be calculated using the Naïve Bayes method through the Orange application.

Keywords: signature verification, dataset, application, authentication

Abstrak

Tanda tangan adalah alat autentifikasi atau bukti identitas seseorang dalam melakukan pengesahan sebuah dokumen. Setiap orang memiliki karakteristik tanda tangan yang khas dan beragam, namun tidak dapat mengelak dari banyaknya kasus pemalsuan tanda tangan. Pemalsuan tanda tangan membuat dokumen menjadi tidak sah atau tidak valid sehingga dapat merugikan pihak yang bersangkutan. Maka sistem yang akurat untuk memverifikasi tanda tangan sangat diperlukan. Pada penelitian ini metode yang digunakan untuk melakukan verifikasi tanda tangan adalah metode Naïve Bayes yang dilakukan melalui aplikasi Orange dengan membandingkan tingkat akurasi dan presisi berdasarkan Confusion Matrix dari dataset yang telah diolah. Penelitian ini bertujuan untuk melakukan Verifikasi Tanda Tangan Menggunakan Machine Learning yang data nya akan diambil secara manual seperti tanda tangan ditulis dikertas hvs dan akan di foto menggunakan kamera hp. Setelah itu foto akan di edit seperti di potong dan hanya tanda tangan saja yang terlihat digambar. Dan akan melakukan uji coba dengan tanda tangan sebanyak 500 yang sudah dijadikan JPEG (Joint Photographic Experts Group), setelah itu, akan dihitung tingkat akurasi dengan metode Naive Bayes menggunakan aplikasi.

Kata kunci: verifikasi tanda tangan, dataset, aplikasi, autentifikasi.

1. Pendahuluan

Tanda tangan adalah salah satu karakteristik yang dimiliki oleh setiap orang yang mana fungsinya adalah sebagai alat autentifikasi atau bukti identitas seseorang dalam melakukan pengesahan sebuah dokumen. Setiap orang memiliki karakteristik tanda tangan yang beragam, namun tetap saja cukup banyak ditemukan kasus pemalsuan tanda tangan. Pemalsuan tanda tangan tentu harus diwaspadai karena akan menjadi persoalan, seperti dokumen menjadi tidak sah atau tidak *valid* sehingga dapat merugikan pihak yang bersangkutan. Oleh karena munculnya permasalahan-permasalahan tersebut maka verifikasi tanda tangan sangat diperlukan[1].

Verifikasi tanda tangan (*signature verification*) meliputi dua bagian yang berbeda tetapi berkaitan erat satu sama lain. Yang pertama adalah identifikasi dari pemilik tanda tangan sedangkan yang kedua

adalah keputusan apakah tanda tangan tersebut asli atau dipalsukan[10].

Verifikasi tanda tangan dapat dilakukan secara manual, yaitu dengan membandingkan secara langsung dengan mata manusia namun cara ini masih banyak kekurangannya sehingga perlu adanya sistem yang lebih akurat untuk memverifikasi tanda tangan, verifikasi tanda tangan menggunakan komputer dapat menjadi solusi dari permasalahan tersebut[3].

Orange adalah *software open source* untuk pengolahan *Data Analytics/Data Mining*. Dibandingkan dengan *software Data Mining* lainnya, *Orange* unggul dalam hal visualisasi atau yang biasa kita sebut *visual programming*. *Orange* menyediakan banyak *Widget* yang kita letakkan pada *canvas/drawing board* kemudian kita hubungkan dengan *Widget-Widget* lainnya. Dengan media *canvas* ini maka akan memudahkan pemakai bermain dengan

data dan melakukan proses *data analytics* secara intuitif. Selain *Widget*, *orange* juga menyediakan beberapa *add-on/modul* untuk *problem* pada domain domain tertentu seperti *Text Mining / Text Analytics*, *Bioinformatics*, *Network Data/Social Network*, *Model Maps*, *PrototypesProcess*, dan lain lain[4].

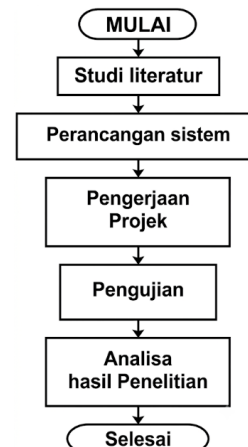
Naive Bayes merupakan *metode* pengklasifikasian paling populer digunakan dengan tingkat keakuratan yang baik. Banyak penelitian tentang pengklasifikasian yang telah dilakukan dengan menggunakan algoritma ini. Berbeda dengan *metode* pengklasifikasian dengan *logistic regression ordinal* maupun nominal, pada algoritma *Naive Bayes* pengklasifikasian tidak membutuhkan adanya pemodelan maupun uji statistik. *Naive Bayes* merupakan *metode* pengklasifikasian berdasarkan probabilitas sederhana dan dirancang agar dapat dipergunakan dengan asumsi antar variabel penjelas saling bebas (*independen*). Pada algoritma ini pembelajaran lebih ditekankan pada pengestimasian probabilitas. Keuntungan algoritma *Naive Bayes* adalah tingkat nilai *error* yang didapat lebih rendah ketika dataset berjumlah besar, selain itu akurasi *Naive Bayes* dan kecepatannya lebih tinggi pada saat diaplikasikan ke dalam dataset yang jumlahnya lebih besar[5]

Penelitian ini bertujuan untuk melakukan “Verifikasi Tanda Tangan Menggunakan *Machine Learning*” yang data nya akan diambil secara manual seperti tanda tangan ditulis dikertas hvs dan akan di foto menggunakan kamera hp. Setelah itu foto akan di *edit* seperti di potong dan hanya tanda tangan saja yang terlihat digambar. Dan akan melakukan uji coba dengan tanda tangan sebanyak 500 yang sudah dijadikan *JPEG (Joint Photographic Experts Group)*, setelah itu, akan dihitung tingkat akurasi dengan *metode Naive Bayes* menggunakan aplikasi *Orange*.

Berdasarkan permasalahan diatas, ruang lingkup penelitian ini membahas hal – hal sebagai berikut: Verifikasi tanda tangan dilakukan secara *offline*, Masukkan berupa *file* citra tanda tangan *Joint Photographic ExpertsGroup (JPEG)* yang sudah di *convert* menjadi *CSV*, Jenis data yang digunakan adalah data sekunder berupa citra tanda tangan yang di ambil langsung oleh partisipan, berupa citra tanda tangan asli dan palsu yang didapat sebanyak 500 tanda tangan dari 2 orang, Pengujian ini dilakukan dengan hanya membandingkan hasil *Test and Score* antara algoritma *Naive Bayes* dan *Adaboost*.

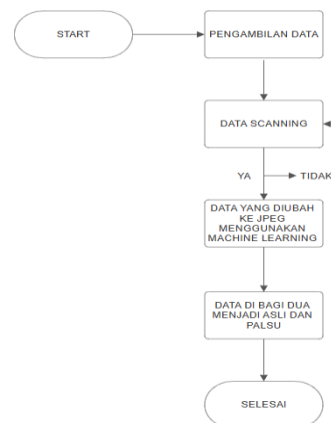
2. Metode Penelitian

Dalam merancang sebuah sistem, diperlukan suatu metode pelaksanaan agar rancangan penelitian yang dibuat terstruktur dan jelas. Tahapan-tahapan dalam pelaksanaan ini tentunya harus efektif dan efisien. sehingga tidak membuang banyak biaya dan waktu secara percuma. Berikut merupakan diagram alur metode pelaksanaan pembuatan sistem.



Gambar 1. Diagram Alir Perancangan Penelitian

Dalam merancang sebuah sistem, diperlukan suatu metode pelaksanaan agar rancangan penelitian yang dibuat terstruktur dan jelas. Tahapan-tahapan dalam pelaksanaan ini tentunya harus efektif dan efisien. sehingga tidak membuang banyak biaya dan waktu secara percuma. Berikut merupakan diagram alir metode pelaksanaan pembuatan sistem.

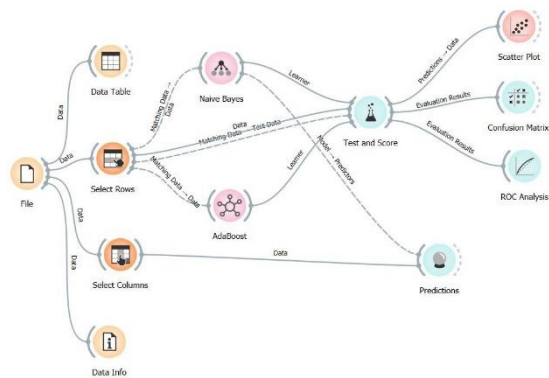


Gambar 2. Diagram Alir Sistem

Penelitian ini diawali dengan pengumpulan data tanda tangan secara manual, yang dilakukan dengan cara dituliskan didalam kertas yang kemudian data discanner. Dan data dijadikan *jpeg file* agar bisa nantinya di *import* ke aplikasi *Orange* tersebut. Kemudian data dibagi 2 atau di pisahkan menjadi data asli dan data peniru.

Dalam perancangan yang ke 2 ini adalah saat data sudah dibagi 2 yaitu data asli dan data peniru yang

akan dijadikan 1 *folder*, yang bahkan di alihkan ke aplikasi *Orange* untuk dilakukan verifikasi tersebut



Gambar 3. Diagram Alir Sistem di Aplikasi Orang

Proses dimulai dengan memasukkan dataset melalui langkah *file*, di mana data diunggah dari *file* dengan format *.CSV*. Setelah itu, data dianalisis secara awal menggunakan modul *Data Info*, yang memberikan informasi deskriptif tentang dataset, seperti jumlah baris, kolom, dan tipe data. Langkah ini membantu memahami struktur dasar dataset sebelum melanjutkan ke proses berikutnya. Dataset kemudian divisualisasikan dalam bentuk tabel menggunakan *Data Table*. Ini memungkinkan pengguna untuk melihat seluruh data. Selanjutnya, data disaring menggunakan *Select Rows*, yang digunakan pemilihan baris tertentu berdasarkan kriteria tertentu, seperti nilai tertentu atau rentang nilai yang relevan untuk analisis. Setelah baris dipilih, data kolom data melalui modul *Select Columns*. Pada tahap ini, kolom yang tidak relevan atau tidak dibutuhkan dihapus, sehingga hanya fitur penting yang tersisa untuk proses selanjutnya. Data yang telah diproses kemudian digunakan untuk melatih model prediktif pertama menggunakan algoritma *Naive Bayes*, yang cocok untuk klasifikasi dengan asumsi independensi antar fitur. Selain itu, model ditingkatkan lebih lanjut melalui teknik *boosting* menggunakan *AddBoost*. Teknik ini bertujuan untuk meningkatkan akurasi prediksi dengan menggabungkan beberapa model lemah menjadi model yang lebih kuat. Langkah selanjutnya adalah *Test and Score*, di mana dataset dibagi menjadi data pelatihan dan data pengujian untuk mengevaluasi performa model. Hasil evaluasi dinilai menggunakan berbagai metrik, seperti akurasi, *F1-score*, dan *area under the curve (AUC)*. Untuk menganalisis hasil lebih dalam, modul *Confusion Matrix* digunakan untuk melihat bagaimana model mengklasifikasikan setiap kelas, termasuk kesalahan prediksi. Selain itu, modul *Scatter Plot* digunakan untuk memvisualisasikan hubungan antara variabel dalam dataset atau hasil prediksi model. Terakhir, modul *ROC Analysis* digunakan untuk mengevaluasi performa model klasifikasi melalui grafik *Receiver Operating Characteristic (ROC)*. Grafik ini membantu menganalisis *trade-off* antara sensitivitas (*true positive rate*) dan spesifisitas (*false positive*

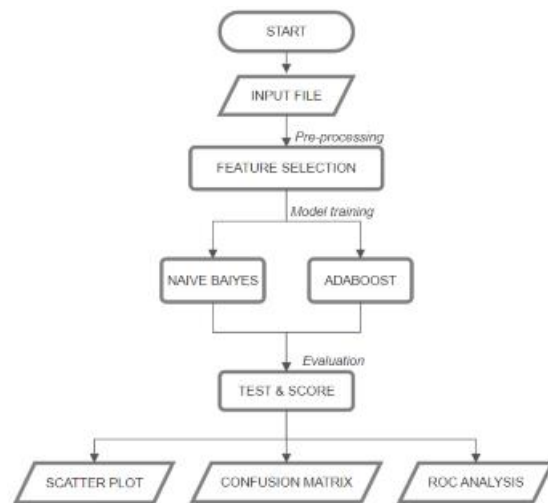
rate) pada berbagai ambang batas prediksi. Dengan alur ini, proses analisis data secara menyeluruh dilakukan, mulai dari tahap awal pengolahan data hingga evaluasi dan visualisasi hasil model.

Alat dan bahan yang dipakai sebagai berikut:

Tabel 1. Estimasi Biaya

No.	Alat/bahan	Harga Satuan (Rp.)	Jumlah	Total (Rp.)	Keterangan ¹
1	Kertas hvs	-	50 pcs	-	Milik sendiri
2	Pena	-	2 pcs	-	Milik sendiri
3	Penggaris	-	1 pcs	-	Milik sendiri
4	Kamera HP	-	1 pcs	-	Milik Sendiri

Target pengujian dalam penelitian ini adalah untuk melakukan verifikasi dan validasi terhadap keaslian tanda tangan asli dan tanda tangan palsu menggunakan aplikasi *orange*. Pengujian dilakukan dengan mencocokkan dataset uji dengan dataset yang telah dilatih menggunakan *orange* dengan memperhatikan 2 variabel input sebagai parameter set seperti *gray scale* dan *edge intensity* pada setiap gambar yang telah dikonversi menjadi *file CSV*.



Gambar 4. Flowchart Sistem

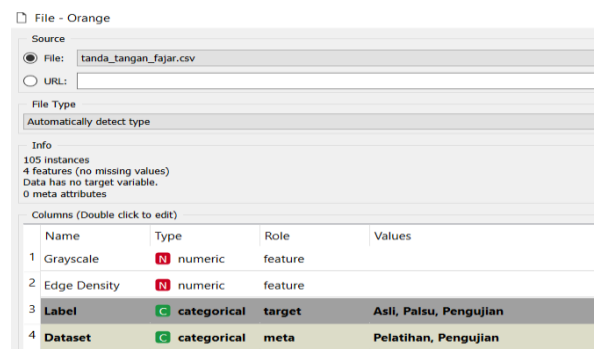
Pada penelitian ini proses pelatihan data dilakukan dengan memasukkan *file* data ke dalam sistem, data ini kemudian melalui serangkaian pemeriksaan awal melalui *Data Info*, yang memberikan gambaran umum tentang karakteristik data. Selanjutnya, data memasuki tahap pemilihan yang lebih spesifik. Melalui *Select Columns* dan *Select Rows*, penulis memilih fitur-fitur yang relevan dan memfilter data yang akan digunakan. Setelah proses pelatihan data selesai, proses dilanjutkan dengan pelatihan model. Di sini, penulis menggunakan dua pendekatan berbeda, yaitu model *Naive Bayes*, yang bekerja dengan menghitung probabilitas setiap kejadian. Kemudian *AdaBoost*, yang bekerja dengan saling memperkuat dan memperbaiki prediksi satu sama lain. Tahap evaluasi di mana kita menguji seberapa baik model kita bekerja.

Test and Score bertindak seperti ujian komprehensif, mengukur kemampuan model dari berbagai aspek.

Confusion Matrix digunakan untuk melihat di mana model pengujian berhasil dan di mana *error* terjadi. *ROC Analysis* memberikan gambaran visual tentang performa model. *Scatter Plot* digunakan untuk memvisualisasikan hasil dalam bentuk yang lebih mudah dipahami. Sistem ini tidak hanya memberikan hasil, tetapi juga memastikan bahwa setiap langkah dapat dipahami dan dievaluasi dengan baik, membuat proses analisis data menjadi lebih transparan dan dapat diandalkan.

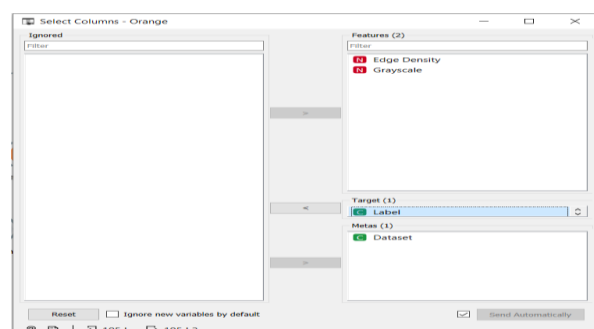
Pengujian ini dilakukan dengan membandingkan nilai parameter kedua algoritma melalui dataset yang telah diolah dan dibagi untuk membandingkan setiap parameter set yang ada.

Dalam melakukan pengujian dilakukan beberapa proses seperti impor dan pembagian data, data yang digunakan dalam penelitian ini adalah data tanda tangan yang diidentifikasi sebagai tanda tangan asli dan tanda tangan palsu, kemudian dataset akan dibagi menggunakan *Widget* yang disediakan oleh aplikasi *orange*. Dalam membagi data pada *orange*, Penentuan parameter dapat digunakan dengan menggunakan mengatur *columns* dan *rows* yang disediakan terpisah oleh *Orange*. Hal ini bertujuan agar *orange* dapat mengidentifikasi data yang telah di *input*.



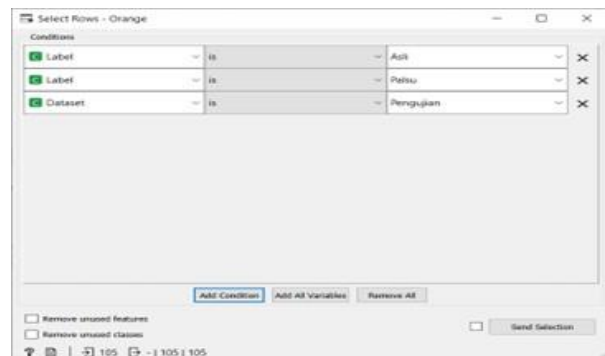
Gambar 5. Widget File

Pada aplikasi *Orange*, *Widget File* digunakan untuk membaca dan memuat data dari berbagai sumber *file*, seperti *file CSV*, *Excel*, atau *file* teks lainnya. Fungsi utamanya adalah untuk mengimpor dataset yang akan digunakan dalam analisis lebih lanjut di dalam aplikasi. Pada penelitian ini *file* yang di impor adalah format *CSV*.



Gambar 6. Widget Columns

Widget Column di *Orange* digunakan untuk memanipulasi dan mengelola kolom-kolom dalam dataset yang telah dimuat. Fungsi utama dari *Widget* ini adalah untuk melakukan berbagai operasi pada kolom data, seperti memilih, menghapus, atau mengubah kolom tertentu sesuai dengan kebutuhan analisis data. Pada penelitian ini, *Edge Density* dan *Grayscale* dikategorikan sebagai *Features*, dan *Label* sebagai *Target* untuk melakukan analisis dalam pengujian yang dilakukan.



Gambar 7. Widget Rows

Widget Rows di *Orange* digunakan untuk memanipulasi baris (atau data instance) dalam dataset. Fungsi utama dari *Widget* ini adalah untuk memungkinkan pengguna memilih atau menghapus baris tertentu berdasarkan kriteria yang diinginkan. Ini membantu untuk menyaring data, misalnya, hanya dengan memilih subset data yang relevan untuk analisis atau eksperimen model.

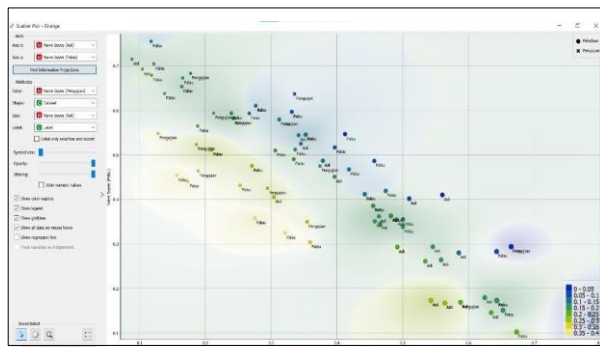
3. Hasil dan Pembahasan

Hasil dan pembahasan berisi data yang harus diketahui dalam penelitian ini. Pengambilan data memiliki tujuan untuk membuktikan bahwa penelitian ini sesuai dengan tujuan yang diharapkan. Berikut merupakan data hasil pengujian yang telah dilakukan dengan membandingkan kedua algoritma dan melakukan evaluasi terhadap hasil pelatihan kedua algoritma menggunakan *Widget Test and score* yang digunakan untuk evaluasi hasil akurasi dan presisi kedua algoritma, *Roc Analysis & Confusion Matrix* yang digunakan untuk analisis hasil pengujian algoritma serta *Scatter Plot* yang digunakan untuk visualisasi.

Model	Train time [s]	Test time [s]	AUC	CA	F1	Precision	Recall	LogLoss	Specificity
Naive Bayes	0.045	0.007	0.641	0.562	0.528	0.498	0.562	0.987	0.688
AdaBoost	0.081	0.011	0.636	0.562	0.557	0.554	0.562	15.131	0.716

Gambar 8. Test and Score

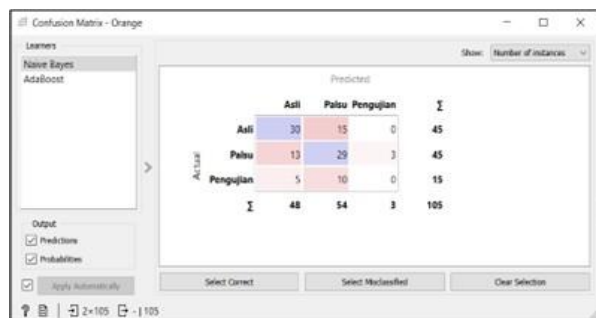
Gambar diatas menunjukkan nilai *Test and Score* yang dilakukan menggunakan *Cross-Validation* dengan 10 *folds*. Pada pelatihan ini data yang digunakan untuk latihan sebanyak 70% dengan pengacakan sebanyak 100 kali (*Random Sampling*). Pada gambar ditunjukkan bahwa nilai *AUC* antara dua algoritma dalam membedakan kelas memiliki *gap* yang sangat kecil, *Naive Bayes* memiliki nilai *AUC* 0.641 & *Adaboost* 0.636. Namun *Adaboost* memiliki nilai *F1*, *Precision*, dan *Specificity* lebih tinggi dibandingkan dengan *Naive Bayes*. Secara keseluruhan *Adaboost* unggul dalam memisahkan kelas yang di set. Namun, *Naive Bayes* secara signifikan lebih unggul pada probabilitas, hal ini tergantung pada *output* yang diharapkan. Penelitian ini berfokus dalam prediksi tanda tangan asli dan palsu, pada hasil penelitian ini *Naive Bayes* lebih dapat diandalkan dalam melakukan prediksi.



Gambar 9. Scatter Plot

Pada gambar di atas, terlihat tampilan *Scatter Plot* yang dihasilkan menggunakan perangkat lunak *Orange* untuk memvisualisasikan hubungan antara dua variabel atau atribut dalam dataset. *Plot* ini memberikan gambaran tentang bagaimana model *Naive Bayes* memprediksi dua atribut yang berbeda, yaitu "Asli" dan "Palsu". Pada sumbu X, kita melihat nilai prediksi model *Naive Bayes* untuk atribut "Asli," sementara pada sumbu Y, terletak nilai prediksi untuk atribut "Palsu". Ini digunakan untuk membandingkan kedua prediksi secara langsung, mengidentifikasi bagaimana model memisahkan kedua kategori tersebut. Data dalam *plot* ini diwakili dengan titik-titik yang memiliki atribut visualisasi tertentu. Warna titik mewakili kategori yang diprediksi oleh model *Naive Bayes* untuk atribut "Pengujian," memberikan gambaran yang jelas tentang distribusi kategori berdasarkan hasil klasifikasi. Setiap titik dilabeli dengan nama kategori, seperti "Asli," "Palsu," "Pengujian," atau "Reluajuan," yang membantu untuk memahami kelas data yang ada. Label ini memberikan konteks lebih lanjut tentang data yang sedang dianalisis, serta memudahkan pengenalan pola dalam visualisasi. Dalam hal pengaturan visualisasi, transparansi atau

opacity disesuaikan agar semua titik data tetap terlihat dengan jelas. Teknik *jittering* digunakan untuk sedikit menggeser posisi titik, yang membantu menghindari ketidakseimbangan antar titik yang berada di posisi yang sangat dekat satu sama lain. Selain itu, pengaturan untuk menampilkan garis kisi atau *gridlines* diaktifkan untuk mempermudah interpretasi nilai-nilai pada sumbu X dan Y. Secara keseluruhan, penyebaran titik-titik pada *plot* ini menunjukkan bagaimana data tersebar berdasarkan prediksi yang dihasilkan oleh model *Naive Bayes*. Kategori-kategori seperti "Asli" dan "Palsu" diidentifikasi berdasarkan klasifikasi model, dan pola distribusi keakuratan model dalam memisahkan kelas. Untuk setiap titik terpisah yang terlihat jelas menunjukkan bahwa model memiliki kemampuan yang baik dalam membedakan kedua kategori tersebut. Sebaliknya, jika titik titik terlalu berdekatan menandakan bahwa model mungkin kurang akurat dalam memisahkan data. Dengan demikian, *Scatter Plot* ini menjadi alat yang sangat berguna untuk menganalisis kinerja model *Naive Bayes* dalam mengklasifikasikan data. Melalui *plot* ini, kita dapat dengan mudah mengidentifikasi prediksi yang salah atau kelas yang salah label, dan menggunakannya sebagai dasar untuk mengevaluasi dan memperbaiki model lebih lanjut.



Gambar 10. Naive Bayes Confusion Matrix

Gambar tersebut menunjukkan *Confusion Matrix* untuk evaluasi performa model *Naive Bayes* dalam membedakan tanda tangan asli dan palsu. Bagian-bagian *Confusion Matrix* yaitu, Baris *Actual* yang menunjukkan kelas sebenarnya dari data (Asli, Palsu, Pengujian), kolom *Predicted* yang menunjukkan prediksi model untuk setiap kelas (Asli, Palsu, Pengujian), angka di dalam tabel yang menunjukkan jumlah *instance* pada setiap kategori sesuai dengan prediksi model dan kelas aktualnya

Tabel 2. Confusion Matrix Naive Bayes

Kelas Aktual	Prediksi Asli	Prediksi Palsu	Prediksi Pengujian	Interpretasi
Asli	30 (<i>True Positive, TP</i>)	15 (<i>False Negative, FN</i>)	0	Model berhasil mengklasifikasikan 30 sampel dengan benar, tetapi masih ada 15 kesalahan FN.
Palsu	13 (<i>False Positive, FP</i>)	29 (<i>True Negative, TN</i>)	3 (Kesalahan Prediksi)	Model salah mengklasifikasikan 13 sampel palsu sebagai asli, namun berhasil mengidentifikasi 29 sampel palsu dengan benar.
Pengujian	5 (Kesalahan Prediksi)	10 (Kesalahan Prediksi)	0	Model masih mengalami kesulitan dalam membedakan kelas Pengujian dengan kelas lain.

Σ : Total *instance* di masing-masing baris adalah 45 untuk Asli, 45 untuk Palsu, dan 15 untuk Pengujian, dengan total keseluruhan 105 *instance*.

		Predicted			Σ
		Asli	Palsu	Pengujian	
Actual	Asli	26	15	4	45
	Palsu	12	29	4	45
	Pengujian	7	4	4	15
Σ		45	48	12	105

Gambar 11. Adaboost Confusion Matrix

Gambar tersebut menunjukkan *Confusion Matrix* untuk evaluasi performa model *AdaBoost* dalam membedakan tanda tangan asli dan palsu. Bagian-Bagian *Confusion Matrix* yaitu, Baris *Actual* yang menunjukkan kelas sebenarnya dari data (Asli, Palsu, Pengujian), kolom *Predicted*: Menunjukkan prediksi model untuk setiap kelas (Asli, Palsu, Pengujian), angka di dalam tabel: Jumlah *instance* pada setiap kategori sesuai dengan prediksi model dan kelas aktualnya.

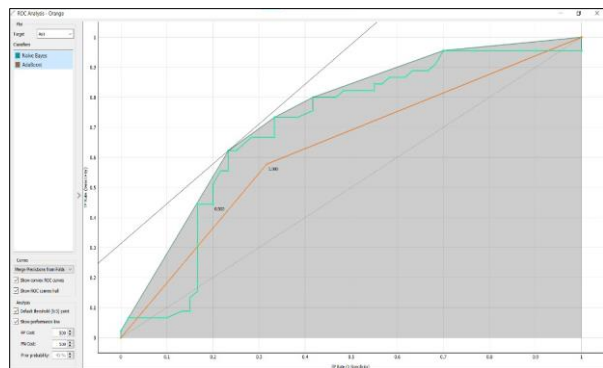
Tabel 3. Confusion Matrix Adaboost

Kelas Aktual	Prediksi Asli	Prediksi Palsu	Prediksi Pengujian	Interpretasi
Asli	26 (<i>True Positive, TP</i>)	15 (<i>False Negative, FN</i>)	4 (Kesalahan Prediksi)	Model berhasil mengklasifikasikan 26 sampel dengan benar, tetapi 15 sampel salah diklasifikasikan sebagai palsu.
Palsu	12 (<i>False Positive, FP</i>)	29 (<i>True Negative, TN</i>)	4 (Kesalahan Prediksi)	Model salah mengklasifikasikan 12 sampel palsu sebagai asli, namun berhasil mengidentifikasi 29 sampel palsu dengan benar.
Pengujian	4 (Kesalahan Prediksi)	4 (Kesalahan Prediksi)	4 (Benar)	Model masih mengalami kesalahan dalam membedakan kelas Pengujian dengan kelas lain.

Σ : Total *instance* di masing-masing baris adalah 45 untuk Asli, 45 untuk Palsu, dan 15 untuk Pengujian, dengan total keseluruhan 105 *instance*.

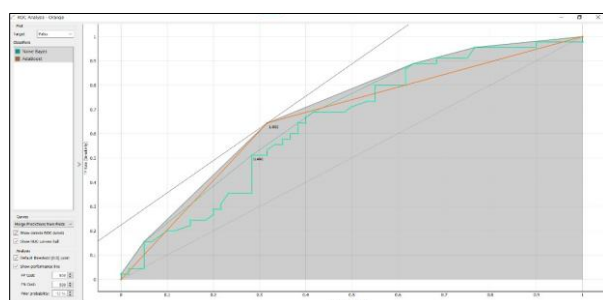
Interpretasi Kinerja Model, Benar Klasifikasi (*True Positives dan True Negatives*) yaitu, asli yang benar diprediksi: 26, palsu yang benar diprediksi: 29, pengujian yang benar diprediksi: 4. Kesalahan Klasifikasi (*False Positives dan False Negatives*)

yaitu, asli diprediksi salah sebagai Palsu: 15, palsu diprediksi salah sebagai Asli: 12.



Gambar 12. ROC Analysis Naive Bayes Adaboost

Gambar tersebut menunjukkan hasil analisis *Receiver Operating Characteristic (ROC)* menggunakan perangkat lunak *Orange* untuk membandingkan kinerja dua metode klasifikasi *Naive Bayes* dan *AdaBoost* yaitu, Sumbu X menunjukkan tingkat positif palsu (*False Positive Rate, 1 - Spesifisitas*) dan Sumbu Y menunjukkan tingkat positif benar (*True Positive Rate, Sensitivitas*). Kemudian 2 kurva ROC, Kurva biru menunjukkan performa metode *Naive Bayes* & Kurva merah menunjukkan performa metode *AdaBoost*. Garis diagonal (warna abu-abu) merepresentasikan kinerja acak atau *baseline* ($AUC = 0.5$). Dapat dilihat dari gambar bahwa kurva *Adaboost* berada di atas *Naive Bayes* untuk sebagian besar titik, hal ini menunjukkan bahwa *Adaboost* memiliki performa yang cenderung lebih baik dibandingkan *Naive Bayes*. Selanjutnya, terlihat pada gambar bahwa nilai AUC untuk *Naive Bayes* (kurva biru) adalah 0,500 yang menunjukkan kinerja mendekati acak, sedangkan AUC untuk *Adaboost* (kurva merah) adalah 1,000 yang menunjukkan kinerja yang lebih baik dalam memisahkan kelas.



Gambar 13. ROC Analysis Adaboost and Naive Bayes

Gambar tersebut menunjukkan hasil analisis *Receiver Operating Characteristic (ROC)* menggunakan perangkat lunak *Orange* untuk membandingkan kinerja dua metode klasifikasi *Naive Bayes* dan *AdaBoost* yaitu, Sumbu X menunjukkan tingkat positif palsu (*False Positive Rate, 1 - Spesifisitas*) dan Sumbu Y menunjukkan tingkat positif benar (*True Positive Rate, Sensitivitas*)

Kemudian 2 kurva ROC, Kurva biru menunjukkan performa metode *Naive Bayes* & Kurva merah menunjukkan performa metode *AdaBoost*. Garis diagonal (warna abu-abu) merepresentasikan kinerja acak atau *baseline* ($AUC = 0.5$). Dapat dilihat dari gambar bahwa kurva *Adaboost* berada di atas *Naive Bayes* untuk di sebagian besar titik, hal ini menunjukkan bahwa *Adaboost* memiliki performa yang cenderung lebih baik dibandingkan *Naive Bayes*. Selanjutnya, terlihat pada gambar bahwa nilai AUC untuk *Naive Bayes* (kurva biru) adalah 0,490 menunjukkan kinerja mendekati acak, sedangkan AUC untuk *Adaboost* (kurva merah) adalah 1,000 yang menunjukkan kinerja yang lebih baik dalam memisahkan kelas.

4. Kesimpulan

Berdasarkan hasil percobaan yang dilakukan dalam penelitian ini, dapat disimpulkan bahwa evaluasi model klasifikasi menggunakan perangkat lunak Orange memberikan wawasan mendalam mengenai kinerja dua model yang diuji, yaitu *Naive Bayes* dan *Adaboost*. Dengan menggunakan metode *cross-validation* 10 folds dan set pelatihan yang mencakup 70% data, penelitian ini berhasil menggali kemampuan kedua model dalam menangani tantangan klasifikasi data dengan lebih komprehensif. Keunggulan *Naive Bayes* terlihat jelas dalam beberapa aspek, terutama pada AUC (Area Under Curve), yang menunjukkan kemampuan model ini untuk membedakan kelas dengan sangat baik. *Naive Bayes* tidak hanya unggul dalam hal AUC, tetapi juga lebih unggul dalam memberikan prediksi probabilitas yang lebih akurat, menjadikannya pilihan yang sangat baik ketika kualitas prediksi probabilitas menjadi prioritas utama dalam mendeteksi tanda tangan asli dan palsu. Namun, meskipun *Adaboost* menunjukkan kinerja yang solid dalam hal F1- score, precision, dan specificity, yang menandakan keunggulannya dalam mengidentifikasi kelas positif dengan tepat, *Naive Bayes* tetap menunjukkan kelebihan dalam hal kemampuannya memberikan gambaran probabilistik yang lebih baik, menjadikannya model yang lebih dapat diandalkan dalam situasi yang membutuhkan prediksi probabilitas yang lebih presisi. Secara keseluruhan, meski kedua model ini memiliki kekuatan dan kelemahan masing-masing, *Naive Bayes* menunjukkan keunggulan yang lebih menonjol dalam hal akurasi klasifikasi dan prediksi probabilitas. Hal ini menjadikannya pilihan yang sangat tepat, terutama dalam konteks analisis data yang berfokus pada klasifikasi tanda tangan asli dan tanda tangan palsu. *Naive Bayes* adalah pilihan yang sangat potensial bagi mereka yang membutuhkan model klasifikasi yang handal, efisien, dan lebih dapat diandalkan dalam memberikan prediksi yang tepat dan probabilitas yang jelas. Berdasarkan hasil penelitian, algoritma *Naive Bayes* yang diaplikasikan pada *Orange* terbukti

efektif untuk klasifikasi tanda tangan, namun ada beberapa saran untuk meningkatkan hasilnya yaitu, Pengoptimalan *Cross-Validation*, alam melakukan penelitian selanjutnya diharapkan untuk menyesuaikan jumlah *fold* dalam *cross-validation* berdasarkan ukuran dataset untuk hasil yang lebih akurat. Feature Engineering, Penelitian dapat dilakukan dengan menambah variabel uji seperti menambahkan data terkait kecepatan atau tekanan tanda tangan, yang dapat meningkatkan akurasi klasifikasi. Penggabungan dua metode *learning*, Cobalah metode *ensemble learning* (seperti *Random Forest* atau *Gradient Boosting*) untuk menggabungkan model dan meningkatkan akurasi serta kestabilan prediksi. Perbanyak Dataset, Untuk hasil yang lebih baik, disarankan untuk memperbanyak dataset dengan variasi yang lebih luas, seperti variasi tulisan tangan atau jenis kertas.

Dengan implementasi langkah-langkah di atas, Diharapkan penelitian selanjutnya dapat mengoptimalkan penggunaan aplikasi untuk klasifikasi tanda tangan, menghasilkan model yang lebih akurat, efisien, dan handal dalam mendeteksi tanda tangan asli maupun palsu menggunakan aplikasi *Orange*.

Daftar Rujukan

- [1] Roni Salambue, "Pengenalan Pola Tanda Tangan dengan Metode Moment Invariant dan Euclidean Distance", *Prosiding Semirata FMIPA*, Vol.1, No.1, 2013.
- [2] Jollyta, Deny, William Ramdhan, and Muhammad Zarlis. *Konsep Data Mining Dan Penerapan*. Deepublish, 2020.
- [3] Andani, Medeline Widia. "Verifikasi tanda tangan menggunakan ekstraksi fitur lbp dan klasifikasi lvq." *Publikasi Tugas Akhir S-1 PSTI FT- UNRAM* (2020).
- [4] Satria, Fiqih. "IMPLEMENTASI ALGORITMA NAÏVE BAYES UNTUK MEMPREDIKSI PENGARUH MEDIA SOSIAL TERHADAP SEMANGAT BELAJAR MAHASISWA DI MASA PANDEMI COVID 19." *Jurnal Informasi dan Komputer* 10.1 (2022): 50-56.
- [5] Sartika, Dewi, and Dana Indra Sensuse. "Perbandingan algoritma klasifikasi Naive Bayes, Nearest Neighbour, dan Decision Tree pada studi kasus pengambilan keputusan pemilihan pola pakaian." *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)* 3.2 (2017): 151-161.
- [6] Saputro, Andri Bimo. *Penerapan Machine Learning untuk mengidentifikasi faktor-faktor yang mempengaruhi Kemampuan Komunikasi Matematis pada Materi Program Linear*. BS thesis. Jakarta: FITK UIN Syarif Hidayatullah Jakarta.
- [7] Rosyid, Harits Ar, Syaad Patmanthara, and

- Imanda Rahmadwito Cahyudi. *Game development*. Ahlimedia Book, 2021.
- [8] Ghani, Muhamad Abdul, and Hamdun Sulaiman. "Deteksi Spam Email dengan Metode Naive Bayes dan Particle Swarm Optimization (PSO)." *Infotek: Jurnal Informatika dan Teknologi* 6.1 (2023): 11-20.
- [9] Tahyudin, Imam, Indika Manggala Putra, and Alif Yahya Syafa'at. *Data Mining Dan Data Warehouse Menggunakan Aplikasi KNIME*. Vol. 1. Zahira Media Publisher, 2021.
- [10] Y. D. Qur'ani and S. Rosmalinda, "Jarangan Syaraf Tiruan Learning Vector Quantization untuk Aplikasi Pengenalan Tanda Tangan," in Seminar Nasional Aplikasi Teknologi Informasi, 2010, pp. 1–5.
- [11] H. Annur, "Klasifikasi Masyarakat Miskin Menggunakan Metode Naive Bayes," *Ilk. J. Ilm.*, vol. 10, no. 2, pp. 160–165, 2018.
- [12] B. Krithiga, P. Sabari, I. Jayasri, and I. Anjali, "Early detection of coronary heart disease by using naive bayes algorithm," *J. Phys. Conf. Ser.*, vol. 1717, no. 1, 2021.
- [13] H. Hafizan and A. N. Putri, "Penerapan Metode Klasifikasi Decision Tree Pada Status Gizi Balita Di Kabupaten Simalungun," *KESATRIA J. Penerapan Sist. Inf. (Komputer Manajemen)*, vol. 1, no. 2, pp. 68–72, 2020.
- [14] NEGARA, Yudha Dwi Putra, et al. PENERAPAN METODE DECISION TREE UNTUK KLASIFIKASI DATA PRODUK SKINCARE UNTUK IBU HAMIL MENGGUNAKAN APLIKASI ORANGE. *Insand Comtech: Information Science and Computer Technology Journal*, 2023,
- [15] Ardella Dean, Awalia. "KLASIFIKASI IMBALANCED DATA UNTUK POST-TRANSLATIONAL MODIFICATION (PTM) PADA SEQUENCES PROTEIN LISIN MENGGUNAKAN ADAPTIVE SYNTHETIC (ADASYN) SAMPLING DENGAN METODE LIGHT GRADIENT BOOSTING MACHINE (LightGBM)." (2023).
- [16] Yunardi, Ir Riky Tri, NASA ZATA DINA S. Kom, and M. Kom. *Data Mining dan Machine Learning dengan Orange3 Tutorial dan Aplikasinya*. Airlangga University Press, 2022.
- [17] Jaya, Hendra. "Desain dan Implementasi Sistem Robotika Berbasis Mikrokontroler." (2016).