
IMPLEMENTASI STACKING ENSEMBLE UNTUK PREDIKSI PRODUKTIVITAS TENAGA KERJA PADA INDUSTRI GARMEN

Hiya Nalatissifa*

Universitas Bina Sarana Informatika
Email: hiya.hys@bsi.ac.id
*Penulis Korespondensi

Abstrak

Produktivitas tenaga kerja merupakan indikator penting dalam menentukan efisiensi dan keberlanjutan operasional di industri garmen. Penelitian ini bertujuan untuk membangun model klasifikasi produktivitas pekerja dengan pendekatan *Stacking Ensemble*, yang menggabungkan tiga algoritma pembelajaran mesin unggulan seperti *XGBoost*, *Random Forest*, dan *LightGBM*. Dataset yang digunakan diperoleh dari Kaggle dengan 1.197 data observasi, yang mencakup berbagai atribut seperti departemen, waktu kerja, lembur, insentif, dan produktivitas aktual. Proses prapemrosesan mencakup *encoding* data kategorikal, rekayasa fitur, dan penanganan ketidakseimbangan kelas menggunakan teknik SMOTE. Model *Stacking Ensemble* dilatih menggunakan teknik validasi silang dan dievaluasi menggunakan metrik klasifikasi seperti akurasi, *presisi*, *recall*, dan *F1-score*. Hasil evaluasi menunjukkan bahwa model yang diusulkan mampu mencapai akurasi 99,58%, *presisi* 100%, dan *recall* 99,29%, yang secara signifikan lebih tinggi dibandingkan penelitian sebelumnya, seperti *Random Forest* (94,36%) dan *Classification C50* (79,10%). Penelitian ini menegaskan keunggulan metode ensemble dalam meningkatkan performa klasifikasi, serta potensinya sebagai alat bantu dalam pengambilan keputusan manajerial terkait peningkatan produktivitas tenaga kerja di sektor tekstil dan garmen berbasis data.

Kata kunci: *Stacking Ensemble, Produktivitas Pekerja, Prediksi*

Abstract

Labor productivity is a key indicator in determining the efficiency and sustainability of operations in the garment industry. This study aims to develop a worker productivity classification model using the Stacking Ensemble approach, which combines three leading machine learning algorithms: XGBoost, Random Forest, and LightGBM. The dataset used was obtained from Kaggle, comprising 1,197 observations with various attributes such as department, working hours, overtime, incentives, and actual productivity. The preprocessing phase involved categorical data encoding, feature engineering, and handling class imbalance using the SMOTE technique. The Stacking Ensemble model was trained using cross-validation and evaluated with classification metrics such as accuracy, precision, recall, and F1-score. Evaluation results indicate that the proposed model achieved an accuracy of 99.58%, precision of 100%, and recall of 99.29%, which are significantly higher compared to previous studies, such as Random Forest (94.36%) and Classification C50 (79.10%). This study highlights the superiority of ensemble methods in enhancing classification performance and their potential as decision-support tools for data-driven productivity improvement in the textile and garment sectors.

Keywords: *Stacking Ensemble, Worker Productivity, Prediction*

1. PENDAHULUAN

Industri garmen merupakan salah satu sektor manufaktur padat karya yang memiliki kontribusi signifikan terhadap perekonomian nasional, terutama di negara berkembang (Suryandari et al., 2025). Industri ini tidak hanya menyediakan lapangan kerja dalam skala besar, tetapi juga menjadi bagian dari rantai pasok global yang menuntut efisiensi tinggi dalam proses produksinya (Paritama, 2023). Dengan meningkatnya persaingan global dan dinamika pasar yang cepat, perusahaan garmen dituntut untuk terus

meningkatkan efisiensi operasionalnya, salah satunya melalui optimalisasi produktivitas tenaga kerja (Rizqi, 2025).

Produktivitas tenaga kerja menjadi indikator penting dalam mengukur efisiensi produksi di industri garmen (Iqbal Muhammad Luthfi & Wijaya, 2024). Namun, produktivitas pekerja seringkali dipengaruhi oleh berbagai faktor kompleks seperti beban kerja, insentif, waktu lembur, dan kinerja tim. Oleh karena itu, pendekatan tradisional dalam mengukur dan mengevaluasi produktivitas seringkali tidak cukup untuk menangkap dinamika tersebut secara akurat (Zahir & Muhlis, 2022). Munculnya data historis dalam jumlah besar membuka peluang untuk menerapkan metode analitik canggih berbasis kecerdasan buatan, salah satunya adalah *machine learning* (Adolph, 2024).

Dalam ranah *machine learning*, salah satu pendekatan yang efektif dalam analisis data adalah metode *ensemble learning*, yang menggabungkan keunggulan beberapa algoritma untuk meningkatkan akurasi dan stabilitas model (Indahyanti et al., 2022). Dalam penelitian ini, metode *Stacking Ensemble* yang mengintegrasikan *XGBoost*, *Random Forest*, dan *LightGBM* digunakan untuk memprediksi produktivitas tenaga kerja berdasarkan berbagai fitur seperti departemen, waktu kerja, dan target produksi. Penggunaan pendekatan ini diharapkan dapat mengurangi kesalahan klasifikasi dan memberikan prediksi yang lebih presisi dibandingkan model Tunggal (Haya & Ramme, 2024).

Penelitian ini memperluas dan memperbaiki pendekatan dari studi sebelumnya yang dilakukan oleh (Amrulloh et al., 2024) dengan memanfaatkan algoritma *Random Forest* dengan *tuning parameter* untuk klasifikasi produktivitas pekerja garmen. Penelitian tersebut mencapai akurasi sebesar 94,36%, dan membandingkan performa dengan model *Decision Tree* dan SVM. Meskipun hasilnya cukup tinggi, model yang digunakan bersifat tunggal (*single model*), sehingga masih terdapat peluang peningkatan performa dengan menggabungkan berbagai model kuat dalam bentuk *ensemble*. Dalam penelitian ini, pendekatan *Stacking Ensemble* berhasil mendorong akurasi prediksi lebih lanjut hingga mencapai 99,58%, menunjukkan peningkatan signifikan dalam klasifikasi produktivitas. Selain itu, (Manurung & Ai, 2025) melakukan penelitian dengan dataset yang sama menggunakan *Regression Tree* dan *Classification C50* untuk prediksi produktivitas tenaga kerja, pendekatan yang digunakan dalam studi ini menawarkan keunggulan dalam hal akurasi dan kemampuan generalisasi. Penelitian tersebut menghasilkan akurasi sebesar 89,76% untuk regresi dan 79,10% untuk klasifikasi. Meskipun metode yang digunakan mudah diinterpretasikan, tingkat akurasinya belum seoptimal pendekatan *ensemble modern* yang digunakan dalam penelitian ini.

Oleh karena itu, dengan pendekatan dan akurasi yang lebih tinggi, penelitian ini diharapkan dapat menjadi referensi baru dalam pemanfaatan teknologi *machine learning* untuk mendukung efisiensi produksi di sektor tekstil dan garmen. Selain itu, hasil model ini juga dapat diterapkan dalam sistem pendukung keputusan manajerial guna mengidentifikasi pola kinerja serta merumuskan strategi peningkatan produktivitas yang lebih akurat dan objektif.

2. METODE PENELITIAN

Metode penelitian yang diterapkan dalam memprediksi produktivitas tenaga kerja di industri garmen dengan menggunakan metode *Stacking Ensemble* dijelaskan sebagai berikut:

2.1. Pengumpulan Data

Penelitian ini menggunakan *dataset* dari *Kaggle* dengan judul *garments_worker_productivity* yang terdiri dari 1.197 data baris dan 15 atribut. Terdapat label target yang dibentuk menjadi dua kelas: "produktif" dan "tidak produktif" berdasarkan nilai ambang 0.5 pada kolom *actual_productivity*. Berikut penjelasan lengkap dari *dataset* yang digunakan pada penelitian:

<i>date</i>	: Tanggal dalam MM-DD-YYYY
<i>day</i>	: Hari dalam Seminggu
<i>quarter</i>	: Sebagian dari bulan. Satu bulan dibagi menjadi empat quarter
<i>department</i>	: Departemen terkait dengan instansi
<i>team_no</i>	: Nomor tim terkait dengan instansi
<i>no_of_workers</i>	: Jumlah pekerja di setiap tim
<i>no_of_style_change</i>	: Jumlah perubahan gaya produk tertentu
<i>targeted_productivity</i>	: Produktivitas yang ditargetkan yang ditetapkan oleh Otoritas untuk setiap tim untuk setiap hari.
<i>smv</i>	: Nilai Menit Standar, yaitu waktu yang dialokasikan untuk suatu tugas
<i>wip</i>	: Pekerjaan yang sedang berlangsung. Termasuk jumlah item yang belum selesai untuk produk
<i>over_time</i>	: Mewakili jumlah lembur oleh setiap tim dalam menit

incentive : Mewakili jumlah insentif *finansial* (dalam BDT) yang memungkinkan atau memotivasi tindakan tertentu.
idle_time : Jumlah waktu Ketika produksi terganggu karena beberapa alasan
idle_men : Jumlah pekerja yang menganggur karena gangguan produksi
actual_productivity : *Persentase* aktual produktivitas yang dihasilkan oleh pekerja (0-1)

2.2. Pra-pemrosesan Data

Langkah-langkah preprocessing yang dilakukan yaitu:

1. Label Encoding

Tiga atribut kategorikal, yaitu *department*, *day*, dan *quarter*, dikodekan ke dalam bentuk numerik menggunakan *Label Encoding (One-Hot Encoding)*. Proses ini penting karena sebagian besar algoritma klasifikasi tidak dapat mengolah data dalam bentuk string atau kategori secara langsung (Mumtazah & Sancoko, 2024).

```
df['department'] = LabelEncoder().fit_transform(df['department'])  
df['day'] = LabelEncoder().fit_transform(df['day'])  
df['quarter'] = LabelEncoder().fit_transform(df['quarter'])
```

Gambar 1. Encode label kategori

2. Fitur Creation

Mengubah skema prediksi regresi menjadi klasifikasi biner. Label target *productivity_label* dibuat berdasarkan nilai *actual_productivity* dengan ketentuan: jika produktivitas aktual lebih dari 0.75 maka diklasifikasikan sebagai produktif (1), dan sebaliknya sebagai tidak produktif (0). Pendekatan ini mengikuti batasan umum dalam industri garmen mengenai tingkat produktivitas yang optimal.

```
df['productivity_label'] = df['actual_productivity'].apply(lambda x: 1 if x > 0.75 else 0)
```

Gambar 2. Feature creation

3. Feature Engineering

Untuk mengoptimalkan representasi data dan meningkatkan kemampuan prediktif model dilakukan pembuatan fitur baru sebagai berikut:

incentive_per_worker : Rasio insentif terhadap jumlah pekerja. Ditambah 1 untuk menghindari pembagian nol.
idle_ratio : Rasio waktu menganggur terhadap SMV (*Standard Minute Value*), mencerminkan efisiensi waktu.
overtime_per_worker : Rasio waktu lembur terhadap jumlah pekerja.
efficiency_gap : Selisih antara target dan realisasi produktivitas, menggambarkan "gap" kinerja.

```
df['incentive_per_worker'] = df['incentive'] / (df['no_of_workers'] + 1)  
df['idle_ratio'] = df['idle_time'] / (df['smv'] + 1)  
df['overtime_per_worker'] = df['over_time'] / (df['no_of_workers'] + 1)  
df['efficiency_gap'] = df['targeted_productivity'] - df['actual_productivity']
```

Gambar 3. Source code Feature Engineering

4. Seleksi Atribut dan Penanganan Nilai Kosong

Seluruh fitur yang tidak berkontribusi langsung terhadap proses klasifikasi seperti *date*, *actual_productivity*, dan label *target productivity_label* dihapus dari data latih. Selanjutnya, dilakukan imputasi terhadap nilai-nilai kosong menggunakan strategi *mean imputation*, yaitu mengganti nilai kosong dengan rata-rata kolom tersebut.

```
X = df.drop(columns=['actual_productivity', 'productivity_label', 'date'])  
X = SimpleImputer(strategy='mean').fit_transform(X)  
y = df['productivity_label']
```

Gambar 4. Seleksi dan Imputasi Fitur

5. Pembagian Data

Dataset dibagi menjadi data latih dan data uji dengan proporsi 80:20 menggunakan fungsi `train_test_split`. Pemisahan ini dilakukan secara acak dengan `random_state=42` untuk memastikan reproduktibilitas hasil.

```
# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Gambar 5. *Split Data Training dan Testing*

6. SMOTE

Distribusi data menunjukkan adanya ketidakseimbangan antara kelas produktif dan tidak produktif. Oleh karena itu, digunakan metode SMOTE (*Synthetic Minority Oversampling Technique*) untuk menyeimbangkan jumlah data pada kelas minoritas dengan membuat sampel sintesis berdasarkan kemiripan antar data (Sulistiyono et al., 2021). Proses oversampling ini hanya dilakukan pada data latih untuk menghindari kebocoran informasi ke data uji.

2.3. Pemodelan

Pada penelitian ini, digunakan pendekatan *Stacking Ensemble Classifier* sebagai metode utama untuk melakukan klasifikasi terhadap produktivitas tenaga kerja. Stacking merupakan teknik *ensemble learning* yang menggabungkan beberapa model pembelajar dasar (*base learners*) dan mengintegrasikan hasil prediksi mereka menggunakan model meta-klasifier (*meta-learner*).

1. Base Learners

Tiga algoritma *machine learning* digunakan sebagai base learners, yaitu:

- XGBoost Classifier (XGBClassifier)**: Merupakan algoritma *boosting* berbasis pohon keputusan yang efisien dan akurat (Wibowo & Isnain, 2025). Parameter `use_label_encoder=False` dan `eval_metric='logloss'` digunakan untuk menyesuaikan versi terbaru dari pustaka XGBoost.
- Random Forest Classifier**: Algoritma *bagging* berbasis kumpulan pohon keputusan, digunakan dengan 100 estimator untuk mengurangi varians dan *overfitting* (Erlin et al., 2022).
- LightGBM Classifier (LGBMClassifier)**: Algoritma *boosting* berbasis histogram yang dirancang untuk performa cepat dan efisiensi memori yang tinggi, cocok untuk dataset besar dengan fitur numerik (Pramudyantoro et al., 2024).

Ketiga model ini dipilih karena performanya yang kuat dalam menangani klasifikasi pada data tabular dan kemampuannya dalam menangkap berbagai pola *non-linear* secara efektif.

```
base_learners = [
    ('xgb', XGBClassifier(use_label_encoder=False, eval_metric='logloss', random_state=42)),
    ('rf', RandomForestClassifier(n_estimators=100, random_state=42)),
    ('lgbm', LGBMClassifier(random_state=42))
]
```

Gambar 6. Model Pembelajar Dasar (*base learners*)

2. Meta-learner

Model akhir atau meta-klasifier menggunakan model *Logistic Regression*. Model ini mengambil output prediksi dari base learners sebagai input dan menghasilkan keputusan akhir. Pemilihan *Logistic Regression* sebagai meta-learner dilakukan karena sifatnya yang sederhana namun efektif dalam menggabungkan hasil prediksi dari model-model sebelumnya secara linier.

3. Konfigurasi Stacking

Proses stacking dikonfigurasi menggunakan validasi silang sebanyak 5 lipatan (*5-fold cross-validation*) untuk meningkatkan generalisasi model dan mengurangi risiko *overfitting*.

```
stacked_model = StackingClassifier(
    estimators=base_learners,
    final_estimator=LogisticRegression(),
    cv=5
)
```

Gambar 7. *Stacking Konfigurasi*

3.4. Evaluasi Model

Setelah model *Stacking Ensemble* selesai dibangun, langkah selanjutnya adalah melakukan proses pelatihan dan evaluasi untuk menilai performa model dalam mengklasifikasikan tingkat produktivitas tenaga kerja. Model dilatih menggunakan data latih hasil oversampling dengan SMOTE (x_{train_sm} dan y_{train_sm}) untuk memastikan distribusi kelas yang seimbang. Proses pelatihan dilakukan dengan memanggil fungsi `.fit()` pada objek *StackingClassifier*. Setelah pelatihan selesai, model digunakan untuk melakukan prediksi pada data uji (X_{test}), yaitu bagian data yang tidak digunakan dalam proses pelatihan untuk memastikan evaluasi yang objektif. Evaluasi model dilakukan dengan menggunakan *Confusion Matrix*, Akurasi, Presisi, *Recall*, dan *F1-Score*.

```
# Training dan evaluasi
stacked_model.fit(X_train_sm, y_train_sm)
y_pred_stack = stacked_model.predict(X_test)

# Output
print("Model: Stacking Ensemble (XGBoost + RF + LightGBM)")
print("Akurasi:", accuracy_score(y_test, y_pred_stack))
print("Presisi:", precision_score(y_test, y_pred_stack))
print("Recall:", recall_score(y_test, y_pred_stack))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred_stack))
print("Classification Report:\n", classification_report(y_test, y_pred_stack))
```

Gambar 8. *Training dan Evaluasi Model*

3. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil dari implementasi model *Stacking Ensemble* untuk klasifikasi produktivitas tenaga kerja pada industri garmen, serta pembahasan terhadap temuan tersebut. Evaluasi dilakukan berdasarkan berbagai metrik performa klasifikasi guna menilai efektivitas model yang dibangun. Selanjutnya, hasil yang diperoleh dibandingkan dengan penelitian sebelumnya untuk melihat peningkatan kinerja model dan kontribusi inovatif dari pendekatan yang digunakan.

3.1. Hasil

Model yang digunakan dalam penelitian ini adalah *Stacking Ensemble Classifier* yang menggabungkan tiga algoritma pembelajaran mesin sebagai *base learners*, yaitu *XGBoost*, *Random Forest*, dan *LightGBM*, dengan *Logistic Regression* sebagai *meta-learner*. Evaluasi performa model dilakukan menggunakan data uji yang telah dipisahkan sebelumnya, dan hasilnya ditunjukkan pada metrik evaluasi berikut:

1. Metrik Evaluasi Model

Hasil evaluasi model menggunakan *classification report*, diperoleh bahwa model *Stacking Ensemble* menunjukkan performa klasifikasi yang sangat tinggi. Untuk kelas 0 (tidak produktif), model mencatat nilai *precision* sebesar 0.99, *recall* sebesar 1.00, dan *f1-score* sebesar 0.99 dari total 99 data. Sementara itu, pada kelas 1 (produktif), model memperoleh *precision* sempurna sebesar 1.00, *recall* sebesar 0.99, dan *f1-score* sebesar 1.00 dari 141 data. Secara keseluruhan, model mencapai akurasi sebesar 100% pada data uji, yang menunjukkan bahwa hampir seluruh prediksi dilakukan dengan benar. Selain itu, nilai *macro average* untuk *precision*, *recall*, dan *f1-score* masing-masing adalah 0.99, 1.00, dan 1.00, sedangkan *weighted average* untuk ketiga metrik tersebut adalah 1.00 secara konsisten.

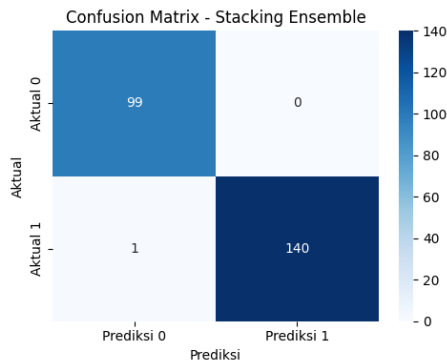
Classification Report:\n		precision	recall	f1-score	support
0	0.99	1.00	0.99	0.99	99
1	1.00	0.99	1.00	1.00	141
accuracy			1.00	1.00	240
macro avg	0.99	1.00	1.00	1.00	240
weighted avg	1.00	1.00	1.00	1.00	240

Gambar 9. *Classification Report Stacking Ensemble*

Hasil ini menunjukkan bahwa model mampu melakukan klasifikasi dengan sangat baik, tidak hanya dalam mengenali kelas mayoritas, tetapi juga dalam mendeteksi kelas minoritas secara akurat. Nilai *precision* dan *recall* yang tinggi di kedua kelas mengindikasikan bahwa model tidak hanya akurat, tetapi juga seimbang dalam mendeteksi berbagai kategori produktivitas tenaga kerja.

2. Confusion Matrix

Visualisasi confusion matrix ditampilkan pada Gambar 10. Matriks ini menunjukkan bahwa dari 240 data uji sebanyak **99 data kelas 0** diprediksi dengan benar, sebanyak **140 data kelas 1** diprediksi dengan benar dan hanya terdapat **1 kesalahan klasifikasi**, yaitu satu data kelas 1 yang diprediksi sebagai kelas 0.



Gambar 10. Confusion Matrix Model Stacking Ensemble

3.2. Pembahasan

Hasil evaluasi menunjukkan bahwa model *Stacking Ensemble* yang dibangun mampu memberikan performa klasifikasi yang sangat baik. Hasil ini jauh lebih tinggi dibandingkan penelitian sebelumnya oleh (Amrulloh et al., 2024) yang menggunakan *Random Forest* dan mencapai akurasi 94,36%, serta penelitian oleh (Manurung & Ai, 2025) yang menggunakan *Classification C50* dengan akurasi 79,10%.

Tabel 1. Perbandingan Hasil Penelitian

Metode yang Digunakan	Akurasi
<i>Random Forest + Hyperparameter Tuning</i>	94,36%
<i>Classification C50</i>	79,10%
<i>Stacking Ensemble (XGBoost, RF, LightGBM)</i>	99,58%

Hasil perbandingan tersebut menunjukkan bahwa metode ensemble yang menggabungkan beberapa algoritma canggih mampu meningkatkan akurasi secara signifikan dalam klasifikasi produktivitas tenaga kerja.

4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan pendekatan *Stacking Ensemble* yang terdiri dari *XGBoost*, *Random Forest*, dan *LightGBM* untuk memprediksi produktivitas pekerja di industri garmen. Dengan akurasi mencapai **99,58%**, pendekatan ini terbukti jauh lebih unggul dibandingkan pendekatan sebelumnya yang hanya menggunakan satu model. Model ini berpotensi besar untuk digunakan dalam sistem pengambilan keputusan perusahaan dalam hal pengelolaan kinerja pekerja dan alokasi tenaga kerja. Untuk pengembangan lebih lanjut, disarankan mengeksplorasi teknik meta-learning lainnya, melakukan validasi silang lebih luas, dan menguji model terhadap data real-time atau produksi dari industri garmen yang berbeda sebagai studi komparatif.

REFERENSI

- Adolph, R. (2024). Perkembangan Kecerdasan Buatan (Ai) Dan Dampaknya Pada Dunia Teknologi. *Jitu: Jurnal Informatika Utama*, 2(2), 1–23.
- Amrulloh, F. H., Farhani, G., Aji, P., S, R. G., Sagi, V., Anindyajati, A., Nabila, L. R., & Tantyoko, H. (2024). Klasifikasi Produktivitas Pekerja Garmen Menggunakan Algoritma Random Forest. *Jurnal Buffer Informatika*, 10(April).
- Erlin, E., Desnelita, Y., Nasution, N., Suryati, L., & Zoromi, F. (2022). Dampak Smote Terhadap Kinerja Random Forest Classifier Berdasarkan Data Tidak Seimbang. *Matrik: Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, 21(3), 677–690. <https://doi.org/10.30812/Matrik.V21i3.1726>
- Haya, A. N., & Ramme, M. Y. (2024). Penerapan Algoritma Stacking Ensemble Machine Learning Berbasis Pohon Untuk Prediksi Penyakit Diabetes. 2024(Senada), 954–961.
- Indahyanti, U., Azizah, N. L., & Setiawan, H. (2022). Pendekatan Ensemble Learning Untuk Meningkatkan Akurasi Prediksi Kinerja Akademik Mahasiswa. *Jurnal Sains Dan Informatika*, 8(2), 160–169. <https://doi.org/10.34128/Jsi.V8i2.459>
- Iqlal Muhammad Luthfi, & Wijaya, F. (2024). Pengaruh Disiplin Kerja Dan Motivasi Kerja Terhadap Produktivitas Kerja Karyawan Divisi Washing Pada Pt Multi Garmen Jaya Bandung. *Jemsi (Jurnal Ekonomi, Manajemen, Dan Akuntansi)*, 10(1), 704–714. <https://doi.org/10.35870/Jemsi.V10i1.2074>
- Manurung, V. A., & Ai, T. J. (2025). Analisis Produktivitas Tenaga Kerja Dengan Menggunakan Regression Tree Dan Classification C50. *Jurnal Rekayasa Industri (Jri)*, 7(1), 1–11.
- Mumtazah, B. B., & Sancoko, S. D. (2024). Adult Clothing Size Recommendation Using K-Nearest Neighbor And Support Vector Machine Algorithm Rekomendasi Ukuran Baju Dewasa Menggunakan Algoritma K-Nearest Neighbor Dan Support Vector Machine. *Malcom: Indonesian Journal Of Machine Learning And Computer Science*, 4(October), 1635–1645.
- Paritama, J. R. (2023). Industri Garmen Dengan Metode Supply Chain Operation Reference Dan Analytical Hierarchy Process (Scor-Ahp). *J@Ti Undip: Jurnal Teknik Industri*, April 2023, 1–5.
- Pramudyantoro, A., Utami, E., & Ariatmanto, D. (2024). Penggabungan K-Nearest Neighbors dan Lightgbm Untuk prediksi Diabetes Pada Dataset Pima Indians: Menggunakan pendekatan Exploratory Data Analysis. *Jipi (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 9(3), 1133–1144. <https://jurnal.stkipppgritulungagung.ac.id/index.php/jipi/article/view/4966/2114>
- Rizqi, I. (2025). Strategi Efektif Untuk Meningkatkan Efisiensi Operasional Perusahaan Journal Of Business Economics And Management. *Journal Of Business Economics And Management*, 01(03), 328–334.
- Sulistiyono, M., Pristyanto, Y., Adi, S., & Gumelar, G. (2021). Implementasi Algoritma Synthetic Minority Over-Sampling Technique Untuk Menangani Ketidakseimbangan Kelas Pada Dataset Klasifikasi. *Sistemasi*, 10(2), 445. <https://doi.org/10.32520/Stmsi.V10i2.1303>
- Suryandari, R. Y., Kasikoen, K. M., Martini, E., & Tamsil, F. (2025). Dinamika Sosial, Ekonomi, Dan Lingkungan Akibat Pembangunan Industri Padat Karya Di Kabupaten Jepara. *Media Bina Ilmiah*, 19(1978), 5409–5420.
- Wibowo, A., & Isnain, A. R. (2025). Implementation Of Machine Learning Algorithm For Environmental Sound Classification Implementasi Algoritma Machine Learning Untuk Klasifikasi Suara Lingkungan. *Malcom: Indonesian Journal Of Machine Learning And Computer Science*, 5(April), 616–625.
- Zahir, L. A., & Muhlis, M. (2022). Jaringan Saraf Tiruan (Neural Network) Untuk Estimasi Produktivitas Pekerja Konstruksi (Artificial Neural Networks For Estimating Construction Worker Productivity). *Jurnal Teknik Sipil Universitas Tulungagung*, 02(02), 142–153.