

Implementasi Algoritma XGBoost untuk Prediksi Kondisi Lingkungan pada Budidaya Jamur Tiram

Andika Syarif Hidayatullah^{*1}, Windarsyah², Ayu Ahadi Ningrum³

^{1,2,3}Program Studi Informatika, Fakultas Teknik, Universitas Muhammadiyah Banjarmasin

Email: ^{*1}andika_syarif_2255201110004@umbjm.ac.id, ²windarsyah@umbjm.ac.id,

³ayuahadi@umbjm.ac.id

(Naskah masuk: 2 Juni 2026, diterima untuk diterbitkan: 9 Juli 2026)

Abstrak: Budidaya jamur tiram (*Pleurotus ostreatus*) memerlukan pengendalian lingkungan yang presisi pada lima parameter kritis: suhu, kelembapan, intensitas cahaya, kadar CO₂, dan pH substrat. Penyimpangan dari rentang optimal pada salah satu parameter tersebut dapat menghambat pertumbuhan miselium hingga menyebabkan kegagalan panen. Sistem monitoring manual bersifat reaktif dan rentan terhadap keterlambatan respons, sehingga pendekatan prediktif menjadi kebutuhan yang mendesak. Penelitian ini mengimplementasikan klasifikasi langsung berbasis XGBoost untuk memprediksi kondisi lingkungan kumbung 6 jam ke depan menggunakan data sensor IoT yang direkam setiap 30 detik selama 17 hari dengan total 50.087 baris data. Pelabelan dilakukan pada nilai sensor aktual menggunakan aturan berjenjang yang membedakan sensor kritis (suhu, kelembapan, pH) dan sensor toleran (intensitas cahaya, CO₂), kemudian XGBClassifier dilatih secara langsung pada label tersebut. Ketidakseimbangan kelas ditangani menggunakan class weight seimbang dan threshold optimal dipilih berdasarkan MCC tertinggi. Hasil menunjukkan Balanced Accuracy 69,90% dan MCC 0,4296, dengan kelas cukup mencapai recall 0,92 dan kelas optimal mencapai precision 0,88. Sistem ini menjadi fondasi intervensi proaktif pada budidaya jamur cerdas berbasis IoT.

Kata Kunci – XGBoost; Machine Learning; IoT; Budidaya Jamur Tiram

Implementation of XGBoost Algorithm for Predicting Environmental Conditions in Oyster Mushroom Cultivation

Abstract: Oyster mushroom (*Pleurotus ostreatus*) cultivation requires precise environmental control across five parameters: temperature, humidity, light intensity, CO₂ levels, and substrate pH. Manual monitoring systems are reactive and prone to response delays, making a predictive approach an urgent necessity. This study implements a direct classification based on XGBoost to predict mushroom house environmental conditions 6 hours in advance, utilizing IoT sensor data recorded every 30 seconds over 17 days, totaling 50,087 data rows. Labeling was performed on actual sensor values using a tiered rule engine that distinguishes between critical sensors (temperature, humidity, pH) and tolerant sensors (light intensity, CO₂), after which the XGBClassifier was trained directly on these labels. Class imbalance was handled using balanced class weights, and the optimal threshold was selected based on the highest MCC. The results show a Balanced Accuracy of 69.90% and an MCC of 0.4296, with the 'sufficient' class achieving a recall of 0.92 and the 'optimal' class achieving a precision of 0.88. This system establishes a foundation for proactive intervention in IoT-based smart mushroom cultivation.

Keywords – XGBoost; Machine Learning; IoT; Oyster Mushroom Cultivation

1. PENDAHULUAN

Jamur tiram (*Pleurotus ostreatus*) merupakan salah satu komoditas pertanian bernilai ekonomi tinggi yang mengandung protein, mineral, vitamin, serta senyawa bioaktif yang bermanfaat bagi kesehatan manusia [1]. Permintaan pasar terhadap jamur tiram terus meningkat setiap tahunnya, didorong oleh kebutuhan hotel, restoran, dan komunitas vegetarian, namun pasokan dari para petani masih belum mampu memenuhi kebutuhan tersebut secara optimal [2]. Kondisi ini menjadikan peningkatan produktivitas budidaya jamur tiram sebagai kebutuhan yang mendesak. Salah satu hambatan utama dalam budidaya jamur tiram adalah ketidakstabilan kondisi

lingkungan di dalam kumbung, yang secara langsung memengaruhi keberhasilan pertumbuhan miselium maupun pembentukan tubuh buah.

Keberhasilan budidaya *Pleurotus ostreatus* sangat ditentukan oleh terpenuhinya kondisi lingkungan yang optimal secara simultan, meliputi suhu udara (20–30°C), kelembapan relatif (70–90%), intensitas cahaya (200–300 lux), kadar CO₂ di bawah 1000 PPM, serta pH substrat 5,5–7,0 [3], [4]. Penyimpangan dari rentang tersebut dapat menghambat pertumbuhan miselium hingga menyebabkan kegagalan panen [5]. Pengelolaan secara manual tidak efisien dan rentan terhadap keterlambatan respons [6].

Perkembangan teknologi Internet of Things (IoT) membuka peluang pemantauan otomatis dan real-time [7]. Chong *et al.* mengembangkan sistem pemantauan IoT untuk budidaya jamur rumahan [5]. Mulyanto *et al.* mengimplementasikan monitoring suhu dan kelembapan berbasis IoT dengan NodeMCU dan Cloud Data Server [2]. Ntihung *et al.* merancang sistem informasi monitoring kumbung berbasis IoT [8]. Arsella *et al.* membuktikan IoT efektif meningkatkan kualitas budidaya menggunakan sensor DHT22 dan Blynk [6]. Namun, semua sistem tersebut masih bersifat reaktif. Budihartono *et al.* menerapkan sistem monitoring suhu dan kelembapan tanah berbasis Arduino pada budidaya porang dengan keakuratan 99%, membuktikan efektivitas sistem monitoring berbasis sensor dalam mendukung keberhasilan budidaya tanaman [9]. Namun, semua sistem tersebut masih bersifat reaktif.

Pendekatan prediktif berbasis machine learning (ML) menjadi solusi yang menjanjikan. Omar *et al.* mengombinasikan IoT dan ML untuk otomasi pertanian jamur [7]. Moysiadis *et al.* menunjukkan ML efektif memonitor pertumbuhan jamur dari data sensor IoT [1]. Wang *et al.* mengembangkan pendekatan hibrid data-fisika untuk prediksi suhu lingkungan jamur [10]. Rahmah *et al.* menunjukkan kombinasi metode ML berbasis data deret waktu sensor lingkungan menghasilkan prediksi lebih reliabel [11]. Ge *et al.* mengaplikasikan XGBoost untuk memprediksi evapotranspirasi tanaman pada greenhouse menggunakan data sensor meteorologi, mempertegas relevansi algoritma tersebut untuk tugas prediksi parameter lingkungan berbasis sensor [12]. Devia dan Soewito membuktikan XGBoost mencapai akurasi 99% dengan waktu eksekusi lebih singkat dibandingkan Random Forest dan MLP pada data tabular berdimensi tinggi [13]. Apriyanthi *et al.* mengkonfirmasi superioritas XGBoost dibandingkan Random Forest dan Regresi Logistik pada metrik akurasi, presisi, recall, dan ROC-AUC, divalidasi dengan Uji McNemar [14].

Sebagai pembanding pada domain prediksi lingkungan yang lebih luas, Latifah *et al.* menerapkan Random Forest untuk memprediksi kategori kualitas air sungai dari data sensor pH, oksigen terlarut, padatan terlarut, dan kekeruhan, memperoleh akurasi 75% dengan variasi validasi silang yang cukup besar (62,5–87,5%) [15]. Temuan ini mengindikasikan bahwa algoritma *ensemble* berbasis pohon secara umum mampu menangani data sensor lingkungan, namun tingkat stabilitas dan akurasinya sangat bergantung pada kompleksitas pola data serta jenis algoritma *ensemble* yang digunakan.

Penelitian mengenai prediksi multivariat lima parameter lingkungan kumbung secara simultan menggunakan ML masih sangat terbatas; sebagian besar studi hanya memprediksi satu atau dua parameter. Jacob *et al.* menegaskan bahwa prediksi kondisi lingkungan berbasis AI masih perlu dieksplorasi lebih jauh [16]. Penelitian ini bertujuan mengimplementasikan algoritma XGBoost untuk memprediksi kondisi lingkungan kumbung jamur tiram (*Pleurotus ostreatus*) secara multivariat, yaitu suhu, kelembapan, intensitas cahaya, CO₂, dan pH substrat dengan horizon prediksi 6 jam ke depan, yang kemudian diklasifikasikan secara langsung menjadi kategori optimal dan cukup menggunakan pendekatan klasifikasi berbasis XGBClassifier dengan aturan pelabelan berjenjang yang membedakan tingkat kekritisitas tiap parameter.

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif eksperimental dengan rancangan pengembangan sistem prediksi berbasis machine learning untuk memperkirakan kondisi lingkungan kumbung jamur tiram (*Pleurotus ostreatus*). Objek penelitian adalah kumbung jamur yang dilengkapi perangkat sensor IoT. Data dikumpulkan secara otomatis melalui jaringan sensor,

kemudian diproses dan dianalisis menggunakan algoritma machine learning di lingkungan Google Colaboratory.

2.1. Tempat dan Waktu Penelitian

Pengumpulan data dilaksanakan pada budidaya kumbung jamur tiram. Perekaman data sensor dilakukan selama periode 23 April–10 Mei 2026 secara kontinu dengan interval perekaman setiap 30 detik, menghasilkan total 50.087 baris data. Dataset yang direkam memuat enam atribut, yaitu timestamp (waktu pencatatan), suhu udara (°C), kelembapan udara relatif (%), intensitas cahaya (lux), kadar CO₂ (PPM), dan pH substrat (baglog). Kelima parameter lingkungan tersebut dipilih karena secara langsung memengaruhi keberhasilan pertumbuhan miselium dan pembentukan tubuh buah *Pleurotus ostreatus*, sehingga pemantauan dan prediksinya secara simultan menjadi kunci dalam pengelolaan kumbung yang optimal.

2.2. Perangkat Pengumpulan Data

Sistem IoT yang digunakan terdiri dari beberapa sensor yang dipasang di dalam kumbung untuk merekam lima variabel lingkungan secara bersamaan, yaitu:

- Suhu udara (°C), diukur menggunakan sensor DHT11.
- Kelembapan udara relatif (%), diukur menggunakan sensor DHT11.
- Intensitas cahaya (lux), diukur menggunakan sensor cahaya LDR.
- Kadar CO₂ di udara (PPM), diukur menggunakan sensor gas MQ135.
- Nilai pH substrat (baglog), diukur menggunakan sensor pH.

Data dari seluruh sensor dikirimkan secara otomatis ke sistem penyimpanan dan dicatat dalam format CSV. Setiap baris data memuat timestamp dan nilai kelima sensor secara bersamaan.

2.3. Konfigurasi Domain dan Batas Fisik Sensor

Sebelum dilakukan pemrosesan, ditetapkan satu sumber kebenaran (single source of truth) untuk seluruh pipeline berupa batas fisik wajar tiap sensor, yang digunakan sebagai acuan deteksi outlier. Batas fisik tersebut disajikan pada Tabel 1.

Tabel 1. Batas fisik tiap sensor

Sensor	Batas Bawah	Batas Atas
Suhu	20,0	32,0
Kelembapan	60,0	100,0
Lux	3,5	500,0
pH	5,5	7,5
PPM	500,0	1500,0

2.4. Pembagian dan Penanganan Outlier

Data mentah dibagi secara kronologis dengan rasio 80:20 untuk data latih dan data uji sebelum dilakukan proses apa pun pada statistik data, guna mencegah kebocoran informasi (data leakage) dari data uji ke data latih. Penanganan outlier dilakukan menggunakan metode batas fisik pada Tabel 1: nilai yang berada di luar rentang batas fisik diidentifikasi sebagai outlier dan diganti dengan nilai median, yang dihitung khusus dari data latih dan diterapkan secara konsisten pada kedua subset (latih dan uji) untuk menjaga konsistensi statistik antar subset.

2.5. Pelabelan Kondisi Kumbung

Pelabelan kondisi kumbung dilakukan terhadap nilai sensor aktual, menggunakan aturan berjenjang yang membedakan tingkat kekritisannya masing-masing parameter terhadap pertumbuhan *Pleurotus ostreatus*. Suhu, kelembapan, dan pH dikategorikan sebagai sensor kritis, sedangkan

intensitas cahaya dan kadar CO₂ dikategorikan sebagai sensor toleran. Batas label optimal dan cukup tiap sensor disajikan pada Tabel 2.

Tabel 2. Batas label per sensor

Sensor	Rentang Optimal	Rentang Cukup
Suhu	22,0 – 28,0	20,0 – 32,0
Kelembapan	80,0 – 90,0	70,0 – 95,0
<i>Lux</i>	50,0 – 300,0	10,0 – 400,0
pH	6,0 – 7,0	5,5 – 7,5
PPM	800 – 1000	500 – 1500

Aturan pelabelan diterapkan secara berjenjang sebagai berikut:

- Apabila salah satu sensor kritis (suhu, kelembapan, atau pH) berada di luar batas rentang cukup, kondisi kumbang langsung dilabeli cukup, tanpa memperhitungkan sensor lain.
- Apabila dua atau lebih sensor toleran (lux dan PPM) berada di luar batas rentang cukup, kondisi dilabeli cukup.
- Apabila tepat satu sensor toleran berada di luar batas rentang cukup, kondisi tetap dilabeli cukup
- Untuk kondisi pada zona abu-abu yang tidak terjaring oleh aturan A–C, digunakan weighted scoring dengan bobot relatif: suhu dan kelembapan masing-masing bernilai 3, pH bernilai 2, serta lux dan PPM masing-masing bernilai 1. Kondisi dilabeli optimal apabila skor gabungan ternormalisasi berada pada atau di bawah ambang 27% dari skor maksimum, dan dilabeli cukup apabila di atas ambang tersebut.

2.6. Feature Engineering

Fitur yang digunakan dalam pelatihan model terdiri dari beberapa kelompok berikut.

- Fitur temporal: jam, menit, hari dalam minggu, hari dalam bulan, serta transformasi siklus harian menggunakan fungsi sin dan cos terhadap jam (jam_sin, jam_cos) untuk menangkap pola periodik 24 jam.
- Fitur lag (Lag feature): nilai historis tiap sensor pada 1, 3, 6, 12, dan 24 jam sebelumnya, seluruhnya mengacu ke masa lalu untuk mencegah future leakage.
- Statistik bergulir (rolling statistics): rata-rata dan deviasi standar bergulir tiap sensor pada jendela waktu 1, 3, dan 6 jam.
- Fitur diferensial (tren): selisih nilai sensor terhadap 1 jam dan 6 jam sebelumnya, untuk menangkap arah perubahan (trend) tiap parameter.
- Target label: label kondisi kumbang (optimal/cukup) pada 6 jam ke depan, yang dihasilkan dari proses pelabelan pada Bagian 2.5 yang diterapkan pada nilai sensor shifted sejauh horizon prediksi.

2.7. Arsitektur Metode Pelatihan

Penelitian ini mengadopsi pendekatan klasifikasi langsung (*direct classification*), pendekatan klasifikasi langsung dipilih untuk menghindari *cascading error*, yaitu akumulasi kesalahan prediksi regresi yang ikut terbawa ke tahap pelabelan akhir.

Model klasifikasi dibangun menggunakan XGBClassifier dengan konfigurasi hyperparameter sebagai berikut: `n_estimators=500`, `max_depth=6`, `learning_rate=0,05`, `subsample=0,8`, `colsample_bytree=0,8`, `objective='binary:logistic'`, dan `early_stopping_rounds=30`. Pemilihan XGBoost sebagai algoritma utama didasarkan pada ketangguhannya yang telah dibuktikan secara konsisten dalam menangani data deret waktu sensor lingkungan, sebagaimana ditunjukkan oleh Rajkumar dan Umagandhi yang mencapai $R^2=0,92$ dalam memprediksi kelembapan pada proses pelayuan teh menggunakan sistem IoT berbasis XGBoost [17]. Data latih dibagi lebih lanjut menjadi data pelatihan (90%) dan validation set (10% terakhir secara kronologis) yang digunakan untuk early stopping selama proses pelatihan.

Penanganan ketidakseimbangan kelas dilakukan menggunakan class weight yang dihitung secara seimbang (balanced) dari distribusi kelas pada data latih, dipilih sebagai alternatif teknik

SMOTE karena lebih sesuai untuk data deret waktu yang struktur temporalnya tidak boleh diacak. Threshold klasifikasi optimal kemudian dicari secara otomatis pada rentang 0,30 hingga 0,80 dengan langkah 0,01, berdasarkan nilai Matthews Correlation Coefficient (MCC) tertinggi pada data uji, alih-alih menggunakan nilai default 0,5.

Objek penelitian berupa kumbung jamur tiram memiliki karakteristik iklim mikro yang dapat mengalami fluktuasi akibat gangguan eksternal, seperti pembukaan pintu, penyiraman manual, dan perubahan suhu lingkungan sekitar. Sugiartawan dan Desnanjaya melaporkan bahwa variabilitas iklim mikro semacam ini merupakan tantangan utama dalam pemodelan prediktif kondisi kumbung [18]. Pertimbangan ini turut mendasari pemilihan horizon prediksi 6 jam yang relatif lebih pendek, guna mengurangi akumulasi ketidakpastian akibat fluktuasi iklim mikro jangka panjang.

2.8. Evaluasi Model

Distribusi kelas hasil pelabelan ini relatif lebih seimbang (rasio cukup terhadap optimal sekitar 44:56), evaluasi kinerja klasifikasi tetap menitikberatkan pada metrik yang tidak bias terhadap kelas mayoritas. Kinerja model dievaluasi menggunakan metrik-metrik berikut.

- a. Akurasi (*Accuracy*): Mengukur proporsi prediksi label yang benar terhadap keseluruhan sampel uji.
- b. *Precision*: Mengukur proporsi prediksi positif yang benar-benar positif, mencerminkan kemampuan model meminimalkan kesalahan prediksi positif palsu (*false positive*).
- c. *Recall*: Mengukur proporsi sampel positif aktual yang berhasil diidentifikasi oleh model, mencerminkan kemampuan model meminimalkan kondisi positif yang terlewat (*false negative*).
- d. *F1-Score*: Rata-rata harmonik dari *Precision* dan *Recall*, berguna pada kondisi data yang tidak seimbang.
- e. *Balanced Accuracy*: Rata-rata *recall* dari masing-masing kelas, sehingga tidak bias terhadap kelas mayoritas, dan menjadi metrik akurasi utama dalam penelitian ini.
- f. *Matthews Correlation Coefficient* (MCC): Mengukur korelasi antara label prediksi dan label aktual dengan memperhitungkan keempat nilai *confusion matrix* (TP, TN, FP, FN) secara seimbang. Nilai MCC berkisar dari -1 (prediksi selalu salah) hingga +1 (prediksi selalu benar), dengan 0 menunjukkan performa setara tebakan acak.

Formula perhitungan metrik utama adalah sebagai berikut.

$$1. \text{ Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$2. \text{ Precision} = \frac{TP}{TP+FP} \quad (2)$$

$$3. \text{ Recall} = \frac{TP}{TP+FN} \quad (3)$$

$$4. \text{ F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

keempat metrik di atas, mengingat distribusi kelas pada penelitian ini tidak sepenuhnya seimbang, digunakan pula dua metrik tambahan yang lebih tahan terhadap bias kelas mayoritas:

$$1. \text{ Balanced Accuracy} = \frac{1}{2} \times (\text{Recall kelas cukup} + \text{Recall kelas optimal}) \quad (5)$$

$$2. \text{ MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (6)$$

Validasi tambahan dilakukan melalui Rolling-Origin Cross Validation dengan 5 fold menggunakan TimeSeriesSplit, untuk memastikan model mampu melakukan generalisasi pada berbagai periode waktu yang berbeda tanpa melanggar urutan kronologis data. Pada setiap fold, model dilatih ulang dengan class weight dan threshold optimal dicari kembali berdasarkan MCC tertinggi pada subset validasi fold tersebut.

3. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil klasifikasi kondisi lingkungan kumbung jamur tiram (*Pleurotus ostreatus*) menggunakan algoritma XGBClassifier dengan horizon prediksi 6 jam ke depan. Kinerja

model dievaluasi secara komprehensif menggunakan metrik *Accuracy*, *Balanced Accuracy*, *Matthews Correlation Coefficient* (MCC), *Precision*, *Recall*, dan *F1-Score* pada dataset pengujian yang tidak terlihat. Untuk memberikan pemahaman yang lebih mendalam mengenai interpretasi hasil, metrik utama dihitung berdasarkan nilai dari *Confusion Matrix*, di mana:

- a. *True Positive* (TP): Jumlah kondisi optimal yang diprediksi benar sebagai optimal.
- b. *True Negative* (TN): Jumlah kondisi cukup yang diprediksi benar sebagai cukup.
- c. *False Positive* (FP): Jumlah kondisi cukup yang salah diprediksi sebagai optimal.
- d. *False Negative* (FN): Jumlah kondisi optimal yang salah diprediksi sebagai cukup (kondisi optimal terlewat).

3.1. Hasil Pra-pemrosesan Data

Dataset yang digunakan dalam penelitian ini direkam selama periode 23 April–10 Mei 2026 dengan interval perekaman 30 detik, menghasilkan total 50.087 baris data mentah. Data dibagi secara kronologis dengan rasio 80:20 sebelum dilakukan penanganan outlier maupun pelabelan, guna mencegah kebocoran informasi statistik dari data uji ke data latih. Median untuk penanganan outlier dihitung khusus dari data latih dan diterapkan secara konsisten pada kedua subset.

Pelabelan kondisi kumbang menggunakan aturan berjenjang yang membedakan sensor kritis (suhu, kelembapan, pH) dan sensor toleran (lux, PPM) menghasilkan distribusi kelas yang relatif lebih seimbang. Pada data uji, diperoleh 2.815 sampel berlabel cukup dan 3.603 sampel berlabel optimal (total 6.418 sampel), atau dengan rasio sekitar 44:56.

3.2. Hasil Pelatihan Model Klasifikasi

Model XGBClassifier dilatih menggunakan sample weight berbasis kelas untuk menangani sisa ketidakseimbangan kelas, dengan validation set terpisah sebesar 10% dari data latih untuk keperluan early stopping. Pencarian threshold optimal berdasarkan MCC pada rentang 0,30–0,80 menghasilkan nilai threshold terbaik sebesar 0,54, sedikit di atas nilai default 0,5, yang mengindikasikan model sedikit lebih konservatif dalam memprediksi kelas optimal.

3.3. Evaluasi Kinerja Model

Hasil evaluasi kinerja model pada data uji disajikan pada Tabel 3.

Tabel 3. Hasil evaluasi model

Metrik	Nilai
<i>Threshold</i> terbaik	0,54
<i>Accuracy</i>	67,19%
<i>Balanced Accuracy</i>	69,90%
MCC	0,4296
<i>F1 macro</i>	0,6657
<i>F1 weighted</i>	0,6602

Rincian *classification report* per kelas disajikan pada tabel 4.

Tabel 4. Classification report

Kelas	Precision	Recall	F1 score	Support
Cukup	0.58	0.92	0.71	2.185
Optimal	0.88	0.48	0.62	3.603
Macro avg	0.73	0.70	0.67	6.418
Weighted avg	0.75	0.67	0.66	6.418

Model mencapai *Balanced Accuracy* sebesar 69,90% dan MCC sebesar 0,4296. Nilai MCC tersebut tergolong dalam kategori korelasi moderat, mengindikasikan bahwa model memiliki kemampuan prediksi yang secara substansial lebih baik dibandingkan tebakan acak (MCC=0), meskipun belum mencapai performa optimal (MCC=1). Penggunaan *Balanced Accuracy* dan MCC

sebagai metrik utama, alih-alih akurasi biasa (accuracy), didasarkan pada pertimbangan bahwa akurasi konvensional cenderung menyesatkan pada data dengan distribusi kelas yang tidak seimbang. Hal ini terbukti pada pendekatan awal penelitian ini, di mana akurasi yang tampak tinggi (90,47%) ternyata menyembunyikan kegagalan model dalam mendeteksi kelas minoritas (recall hanya 0,15).

Pada pendekatan klasifikasi langsung ini, kelas cukup menunjukkan recall yang sangat tinggi (0,92) namun precision yang lebih rendah (0,58), mengindikasikan model cenderung berhati-hati dan sering menandai kondisi sebagai cukup meskipun sebagian sebenarnya optimal. Sebaliknya, kelas optimal menunjukkan precision yang tinggi (0,88) namun recall yang lebih rendah (0,48), berarti ketika model memprediksi kondisi optimal, prediksi tersebut sangat dapat diandalkan, namun model masih melewatkan banyak kondisi optimal yang sebenarnya terjadi. Pola trade-off antara precision dan recall ini konsisten dengan pemilihan threshold berdasarkan MCC yang mengoptimalkan keseimbangan kedua kelas secara simultan, alih-alih memaksimalkan satu metrik secara sepihak.

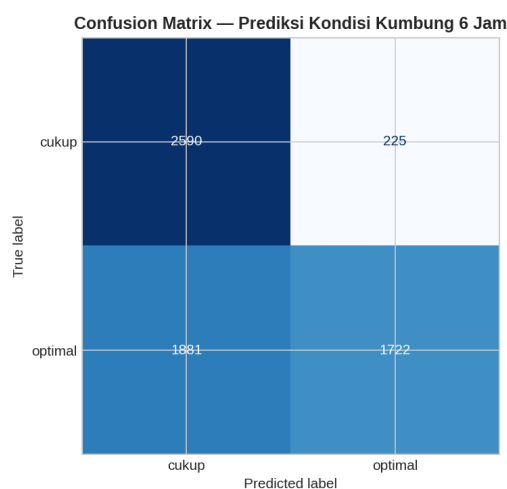
Performa ini sejalan dengan karakteristik umum algoritma berbasis gradient boosting pada data sensor lingkungan yang bersifat dinamis, di mana tingkat akurasi sangat dipengaruhi oleh kompleksitas pola data, horizon prediksi, dan definisi label yang digunakan. Nilai MCC sebesar 0,4296 pada penelitian ini menunjukkan bahwa model mampu menangkap pola yang bermakna pada data sensor lingkungan kumbung jamur tiram, meskipun tingkat ketelitiannya masih dapat ditingkatkan lebih lanjut.

3.4. Hasil Rolling Origin Cross Validation

Validasi tambahan melalui Rolling-Origin Cross Validation dengan 5 fold menggunakan TimeSeriesSplit dilakukan untuk menguji konsistensi performa model pada berbagai periode waktu yang berbeda dalam dataset. Pada setiap fold, model dilatih ulang menggunakan class weight dan threshold optimal dicari kembali berdasarkan MCC tertinggi pada subset validasi fold tersebut. Pendekatan validasi ini memastikan performa model tidak hanya bergantung pada satu pembagian data latih-uji tertentu, melainkan teruji konsistensinya pada periode waktu yang berbeda-beda, sehingga memberikan keyakinan yang lebih kuat terhadap kemampuan generalisasi model dibandingkan evaluasi tunggal pada satu split data saja.

3.5. Analisis Confusion Matrix

Untuk memberikan pemahaman yang lebih mendalam terhadap distribusi prediksi model, hasil klasifikasi divisualisasikan dalam bentuk Confusion Matrix sebagaimana disajikan pada Gambar 1.



Gambar 1. Confusion matrix prediksi 6 jam

Berdasarkan Gambar 1, dari total 6.418 sampel data uji, model menghasilkan rincian prediksi sebagai berikut:

- True Negative* (TN) = 2.590: Jumlah kondisi cukup yang diprediksi dengan benar sebagai cukup. Ini merupakan nilai terbesar dalam *confusion matrix*, menunjukkan bahwa model sangat andal dalam mengenali kondisi cukup.
- False Positive* (FP) = 225: Jumlah kondisi cukup yang salah diprediksi sebagai optimal. Nilai ini relatif kecil, mengindikasikan rendahnya tingkat *false alarm* terhadap kondisi optimal.
- False Negative* (FN) = 1.881: Jumlah kondisi optimal yang salah diprediksi sebagai cukup. Ini merupakan kelemahan utama model, di mana hampir separuh kondisi optimal yang sebenarnya terjadi tidak berhasil dideteksi.
- True Positive* (TP) = 1.722: Jumlah kondisi optimal yang diprediksi dengan benar sebagai optimal.

Dari nilai-nilai tersebut, dapat dihitung ulang metrik evaluasi utama sebagai berikut. Precision kelas optimal = $1722 / (1722 + 225) = 0,88$, yang berarti dari seluruh prediksi optimal yang dihasilkan model, 88% terbukti benar. Recall kelas optimal = $1722 / (1722 + 1881) = 0,48$, yang berarti model hanya berhasil mendeteksi 48% dari seluruh kondisi optimal yang sebenarnya terjadi. Recall kelas cukup = $2590 / (2590 + 225) = 0,92$, mengkonfirmasi kemampuan model yang sangat baik dalam mendeteksi kondisi cukup.

Pola pada *confusion matrix* ini mengkonfirmasi trade-off yang telah diidentifikasi sebelumnya: model cenderung lebih konservatif dalam memprediksi kondisi optimal, yang merupakan konsekuensi langsung dari pemilihan threshold 0,54 berdasarkan MCC tertinggi. Nilai FN yang relatif tinggi (1.881) mengindikasikan bahwa dalam konteks implementasi sistem kendali otomatis kumbung, masih diperlukan penyesuaian threshold yang lebih sensitif terhadap kelas optimal apabila tujuan utama sistem adalah memaksimalkan deteksi kondisi optimal secara proaktif.

4. KESIMPULAN

Penelitian ini telah berhasil mengimplementasikan algoritma XGBoost dalam pendekatan klasifikasi langsung untuk memprediksi kondisi lingkungan kumbung budidaya jamur tiram (*Pleurotus ostreatus*) secara multivariat dengan horizon prediksi 6 jam ke depan. Berdasarkan hasil penelitian yang telah dilakukan, dapat ditarik simpulan sebagai berikut.

Pertama, aturan pelabelan berjenjang yang membedakan sensor kritis (suhu, kelembapan, pH) dan sensor toleran (intensitas cahaya, CO₂) menghasilkan distribusi kelas yang lebih seimbang (rasio cukup terhadap optimal sekitar 44:56), sehingga metrik evaluasi yang dihasilkan menjadi lebih representatif terhadap kemampuan model yang sebenarnya.

Kedua, model XGBClassifier yang dilatih menggunakan class weight seimbang dan threshold optimal berbasis Matthews Correlation Coefficient (MCC) mencapai Balanced Accuracy sebesar 69,90% dan MCC sebesar 0,4296, yang tergolong dalam kategori korelasi moderat. Kelas cukup menunjukkan recall tinggi (0,92) namun precision lebih rendah (0,58), sementara kelas optimal menunjukkan precision tinggi (0,88) namun recall lebih rendah (0,48), mencerminkan trade-off yang dihasilkan dari pemilihan threshold yang menyeimbangkan kedua kelas secara simultan.

Ketiga, validasi tambahan melalui Rolling-Origin Cross Validation 5-fold memberikan keyakinan terhadap konsistensi performa model pada berbagai periode waktu yang berbeda, memastikan hasil evaluasi tidak hanya bergantung pada satu pembagian data latih-uji tertentu.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada dosen pembimbing Program Studi S1 Informatika Universitas Muhammadiyah Banjarmasin atas bimbingan, arahan dan masukan selama proses penelitian dan penulisan. Ucapan terima kasih juga disampaikan kepada UD Jamur Borneo selaku

pengelola kumbung budidaya jamur tiram yang telah memberikan izin dan akses untuk pengambilan data sensor IoT selama penelitian.

DAFTAR PUSTAKA

- [1] V. Moysiadis, G. Kokkonis, S. Bibi, I. Moscholios, N. Maropoulos, and P. Sarigiannidis, "Monitoring Mushroom Growth with Machine Learning," *Agriculture*, vol. 13, no. 1, pp. 223, Jan. 2023, doi: 10.3390/agriculture13010223.
- [2] Y. Mulyanto, F. Iditriani, E. S. Susanto, and S. Sulastri, "Implementasi Sistem Monitoring Berbasis Internet Of Things (IoT) pada Rumah Budidaya Jamur Tiram," *Digital Transformation Technology*, vol. 3, no. 2, pp. 871–878, Jan. 2024, doi: 10.47709/digitech.v3i2.3404.
- [3] R. A. Rahman, D. N. Ramadan, S. Hadiyoso, S. S. Maidin, and I. D. Irawati, "A Smart Kumbung For Monitoring and Controlling Environment in Oyster Mushroom Cultivation Based on Internet of Things Framework," *Journal of Applied Engineering and Technological Science (JAETS)*, vol. 5, no. 1, pp. 245–257, Dec. 2023, doi: 10.37385/jaets.v5i1.2248.
- [4] A. Anggrawan, C. Satria, and M. Zulfikri, "Building an IoT-Based Oyster Mushroom Cultivation and Control System and Its Practical Learning Effects on Students," *TEM Journal*, pp. 1853–1867, Aug. 2023, doi: 10.18421/TEM123-69.
- [5] J. L. Chong, K. W. Chew, A. P. Peter, H. Y. Ting, and P. L. Show, "Internet of Things (IoT)-Based Environmental Monitoring and Control System for Home-Based Mushroom Cultivation," *Biosensors (Basel)*, vol. 13, no. 1, pp. 98, Jan. 2023, doi: 10.3390/bios13010098.
- [6] S. Arsella, M. Fadhlil, and L. Lindawati, "Optimasi Pertumbuhan Jamur Tiram Melalui Monitoring Suhu dan Kelembaban Menggunakan Teknologi IoT," *Jurnal RESISTOR (Rekayasa Sistem Komputer)*, vol. 6, no. 1, pp. 34–42, Apr. 2023, doi: 10.31598/jurnalresistor.v6i1.1405.
- [7] Shafie Omar et al., "IoT Enabled Mushroom Farm Automation with Machine Learning," *Advanced and Sustainable Technologies (ASET)*, vol. 3, no. 1, pp. 29–37, Jun. 2024, doi: 10.58915/aset.v3i1.786.
- [8] M. E. N. Ntuhung, P. Sugiartawan, and A. Willdalia, "Sistem Informasi Monitoring Kumbung Jamur Tiram Berbasis Internet of Things," *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, vol. 14, no. 1, pp. 105, Apr. 2024, doi: 10.22146/ijeis.95513.
- [9] E. Budihartono and A. Rakhman, "Monitoring Suhu dan Kelembaban Tanah pada Budidaya Porang Berbasis Arduino," *Smart Comp: Jurnalnya Orang Pintar Komputer*, vol. 11, no. 1, pp. 9–13, Jan. 2022, doi: 10.30591/smartcomp.v11i1.3225.
- [10] M. Wang et al., "Temperature Prediction of Mushrooms Based on a Data–Physics Hybrid Approach," *Agriculture*, vol. 14, no. 1, pp. 145, Jan. 2024, doi: 10.3390/agriculture14010145.
- [11] I. L. Rahmah, A. Nugroho, and R. Y. M. T. Siregar, "Prediksi Curah Hujan Menggunakan Random Forest dan VAR di Kediri Raya," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 7, no. 4, pp. 539–545, Oct. 2025, doi: 10.47233/jteksis.v7i4.2141.
- [12] J. Ge et al., "Prediction of Greenhouse Tomato Crop Evapotranspiration Using XGBoost Machine Learning Model," *Plants*, vol. 11, no. 15, pp. 1923, Jul. 2022, doi: 10.3390/plants11151923.
- [13] A. Devia and B. Soewito, "Analisis Perbandingan Metode Seleksi Fitur untuk Mendeteksi Anomali pada Dataset CIC-IDS-2018," *Jurnal Teknologi Dan Sistem Informasi Bisnis-JTEKSIS*, vol. 5, no. 4, p. 572, 2023, doi: 10.47233/jteksis.v5i4.1069.
- [14] N. P. E. Apriyanti, C. M. Dhewanty, P. D. W. Ayu, I. M. R. A. Nugroho, and I. W. R. Wijaya, "Analisis Performa Komparatif Algoritma Machine Learning untuk Deteksi Fraud dalam Transaksi Blockchain," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 8, no. 1, pp. 8–15, Jan. 2026, doi: 10.47233/jteksis.v8i1.2285.
- [15] J. Latifah, F. D. Marleny, A. A. Ningrum, and M. Munsiy, "SISTEM PREDIKSI KUALITAS AIR SUNGAI PADA PULAU CURIK MENGGUNAKAN ALGORITMA RANDOM FOREST," *Technologia: Jurnal Ilmiah*, vol. 16, no. 3, pp. 503, Jul. 2025, doi: 10.31602/tji.v16i3.19329.

- [16] M. S. Jacob, A. Xu, K. Qian, Z. Qi, X. Li, and B. Zhang, "Artificial Intelligence in Edible Mushroom Cultivation, Breeding, and Classification: A Comprehensive Review," *Journal of Fungi*, vol. 11, no. 11, pp. 758, Oct. 2025, doi: 10.3390/jof11110758.
- [17] B. Rajkumar and R. Umagandhi, "IoT-Enabled Real-Time Monitoring and Machine Learning-Based Prediction System for Optimizing Tea Withering Process," *Communications on Applied Nonlinear Analysis*, vol. 31, no. 8s, pp. 56–86, Sep. 2024, doi: 10.52783/cana.v31.1457.
- [18] P. Sugiartawan and I. G. N. Desnanjaya, "Smart Farming Untuk Pengaturan Suhu Ruangan Pada Budidaya Jamur Tiram Berbasis Backpropagation," *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, vol. 12, no. 2, pp. 191, Oct. 2022, doi: 10.22146/ijeis.78546.