



Optimasi Klusterisasi pada Lama Tempo Pekerjaan Berbasis Gradient Boost Algorithm

Firza Septian^{1*}

¹ Universitas Serelo Lahat, Indonesia

* firzaseptian@unsela.ac.id

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 00 Oktober 00

Revisi Akhir: 00 November 00

Diterbitkan *Online*: 00 Desember 00

KATA KUNCI

Lama Tempo Pekerjaan, Gradient Boosting Algorithm, Analisis Kluster, Manajemen Sumber Daya Manusia.

KORESPONDENSI

Telepon: +6283801078224

E-mail: firzaseptian@unsela.ac.id

ABSTRACT

Penelitian ini mengeksplorasi analisis lama tempo pekerjaan dengan pendekatan algoritma metaheuristik, khususnya Gradient Boosting Algorithm (GBA). Penelitian ini fokus pada optimasi klusterisasi lama tempo pekerjaan melalui algoritma Gradient Boosting. Algoritma ini dikenal karena kemampuannya meningkatkan model secara bertahap dan mendeteksi pola kompleks dalam data. Data terkait lama tempo pekerjaan dikumpulkan dan diproses dengan Label Encoder untuk mempersiapkannya untuk analisis. Model GBA diterapkan dan menghasilkan tingkat akurasi sebesar 96%, dengan evaluasi kinerja yang mendalam menggunakan presisi, recall, f1-score, dan confusion matrix. Model cenderung sangat baik mengklasifikasikan kelas Full time dan Part time, sementara menunjukkan tantangan pada kelas Contract/Temp. Hasil evaluasi memberikan wawasan yang mendalam tentang kekuatan dan kelemahan model, membimbing penyesuaian untuk meningkatkan kinerja. Penelitian ini memberikan kontribusi pada pemahaman lama tempo pekerjaan dan penggunaan algoritma machine learning dalam konteks manajemen sumber daya manusia. Implikasi praktisnya dapat membantu organisasi membuat keputusan yang lebih tepat terkait sumber daya manusia dan perencanaan tenaga kerja.

1. PENDAHULUAN

Dalam era dinamis dan kompleks yang kita hadapi saat ini, lanskap bisnis dan dunia kerja mengalami perubahan yang cepat dan konstan [1]. Oleh karena itu, analisis data terkait lama tempo pekerjaan menjadi aspek krusial dalam pemahaman mendalam terhadap kestabilan dan evolusi tenaga kerja di tengah dinamika ini [2]. Pemahaman yang terperinci mengenai lama waktu yang dihabiskan oleh individu dalam suatu organisasi dapat memberikan wawasan yang berharga terkait keberlanjutan dan adaptabilitas organisasi di tengah tantangan pasar yang selalu berubah [3, 4].

Analisis lama tempo pekerjaan tidak hanya memberikan informasi terkait retensi karyawan, tetapi juga membuka pintu untuk memahami pola karir, perkembangan profesional, dan faktor-faktor yang mempengaruhi produktivitas individu. Hal ini menjadi semakin penting dalam konteks kebutuhan pengembangan karyawan, di mana pemahaman mendalam terhadap lama tempo pekerjaan dapat membantu perusahaan

merancang program pelatihan yang sesuai dan menciptakan lingkungan kerja yang mendukung pertumbuhan karir [5].

Selain itu, analisis lama tempo pekerjaan juga memberikan wawasan tentang efektivitas strategi manajemen sumber daya manusia. Dengan memahami sejauh mana karyawan tetap berkontribusi dalam lingkungan kerja, organisasi dapat mengevaluasi keberhasilan kebijakan manajemen sumber daya manusia yang telah diterapkan, serta menyesuaikannya dengan kebutuhan dan aspirasi karyawan.

Melalui fokus pada lama tempo pekerjaan, penelitian ini tidak hanya mencari pemahaman terkait karakteristik tenaga kerja, tetapi juga bertujuan memberikan pandangan holistik terhadap dinamika organisasi. Pemahaman ini diharapkan dapat membuka pintu untuk inovasi dalam pengelolaan sumber daya manusia, menciptakan organisasi yang responsif, adaptif, dan efisien dalam menghadapi perubahan kontinyu di era bisnis yang penuh tantangan ini.

Penelitian ini bertujuan untuk mengoptimalkan proses klusterisasi lama tempo pekerjaan menggunakan pendekatan algoritma Gradient Boosting Algorithm (GBC). Algoritma ini dikenal karena kemampuannya dalam meningkatkan model

secara bertahap dan mendeteksi pola kompleks dalam data. Dengan keahlian ini, diharapkan algoritma Gradient Boosting mampu memberikan kontribusi signifikan terhadap analisis lama tempo pekerjaan dengan memperhatikan kompleksitas dan variasi data.

Pentingnya penelitian ini terletak pada potensinya untuk memberikan landasan baru dalam pengelolaan sumber daya manusia dengan menyediakan pemahaman yang lebih baik terhadap pola lama tempo pekerjaan. Dengan memanfaatkan kekuatan algoritma metaheuristik, diharapkan hasil penelitian ini dapat memberikan kontribusi signifikan terhadap pengembangan metodologi analisis pekerjaan yang lebih canggih dan efektif. Melalui pendekatan ini, penelitian ini berusaha mengatasi tantangan dalam penentuan kluster lama tempo pekerjaan yang dapat membantu organisasi membuat keputusan yang lebih tepat dan terinformasi dalam mengelola sumber daya manusia mereka.

2. TINJAUAN PUSTAKA

2.1. Lama Tempo Pekerjaan dan Retensi Karyawan

Lama tempo pekerjaan atau tenure di dalam sebuah organisasi telah lama menjadi fokus utama dalam penelitian sumber daya manusia [6]. Pemahaman mendalam terkait lama waktu yang dihabiskan oleh karyawan dalam suatu perusahaan menjadi krusial, karena hal ini tidak hanya mempengaruhi kestabilan organisasi, tetapi juga memiliki dampak signifikan terhadap produktivitasnya. Seiring dengan itu, menjaga retensi karyawan menjadi suatu aspek yang sangat penting dalam strategi manajemen sumber daya manusia [7].

Disoroti bahwa ketika karyawan tetap bekerja dalam organisasi untuk jangka waktu yang cukup lama, hal ini dapat membentuk fondasi kestabilan organisasi secara keseluruhan. Selain itu, memiliki karyawan yang bertahan dalam waktu yang lama juga dapat mengurangi biaya penggantian karyawan yang pada umumnya lebih tinggi dibandingkan dengan biaya mempertahankan karyawan yang sudah terlatih.

Selanjutnya, lama tempo pekerjaan yang memadai tidak hanya sekadar menciptakan kestabilan organisasi, tetapi juga memberikan peluang untuk peningkatan keterampilan dan pengetahuan karyawan seiring berjalannya waktu. Dalam jangka panjang, hal ini dapat memberikan kontribusi positif terhadap efektivitas dan efisiensi tenaga kerja, menghasilkan tim yang lebih terampil dan berpengetahuan.

Pemahaman mendalam terkait lama tempo pekerjaan memberikan dasar yang kuat untuk pengembangan strategi manajemen sumber daya manusia yang berorientasi pada retensi karyawan [8]. Selain itu, pemahaman ini juga menciptakan landasan untuk pengembangan program pelatihan dan pengembangan karyawan yang lebih terarah, dengan mempertimbangkan keunggulan yang diperoleh melalui lama waktu pengabdian karyawan di dalam organisasi.

2.2. Algoritma Ensemble Learning

Dalam penelitian terkini, penerapan algoritma ensemble learning, seperti Gradient Boosting Algorithm [7], telah muncul sebagai pendekatan yang inovatif untuk analisis kluster di berbagai bidang, termasuk studi lama tempo pekerjaan. Algoritma ini,

yang dikenal dengan kemampuannya untuk meningkatkan model secara bertahap dan mendeteksi pola kompleks dalam data, menawarkan pendekatan yang efisien untuk menangani kompleksitas dan variasi data.

Kemampuan Gradient Boosting Algorithm dalam memecahkan masalah optimasi kompleks dapat diuji [10]. Mereka menyoroti bahwa algoritma ini tidak hanya memiliki kinerja yang handal namun juga mampu memberikan solusi yang optimal dalam konteks permasalahan optimasi yang beragam. Dengan kemampuannya untuk menghadapi tantangan kompleksitas data, algoritma metaheuristik seperti Gradient Boosting Algorithm menjadi salah satu alat yang menarik dan relevan dalam konteks analisis kluster, terutama ketika menghadapi sejumlah besar variabel pekerjaan yang beraneka ragam.

3. METODOLOGI

3.1. Pengumpulan Data

Penelitian ini memulai proses dengan pengumpulan data dari sumber-sumber yang relevan terkait lama tempo pekerjaan. Data melibatkan variabel-variabel seperti *job_title*, *company*, *descriptions*, *location*, *category* (Engineering or Information & Communication Technology), *subcategory* (sub work), dan *type* (Full time or Contract). Data ini dapat diperoleh melalui sumber internal perusahaan atau platform pengumpulan data eksternal yang dapat memberikan gambaran komprehensif terkait variabel-variabel tersebut.

3.2. Preprocessing Data

Setelah pengumpulan data, langkah selanjutnya adalah melakukan preprocessing data untuk memastikan kualitas dan keterpaduan data. Proses ini melibatkan pembersihan data dari potensi duplikasi, missing values, dan outliers yang dapat memengaruhi hasil analisis. Data juga diterjemahkan ke dalam bentuk yang dapat dipahami oleh algoritma klusterisasi, termasuk konversi variabel kategorikal menjadi bentuk numerik jika diperlukan.

3.3. Gradient Boosting Algorithm (GBA)

Gradient Boosting Algorithm (GBA) adalah algoritma machine learning yang termasuk dalam kategori ensemble learning, di mana sejumlah model lemah (weak learners) disusun secara bertahap untuk membentuk model yang lebih kuat. GBA bekerja dengan cara mengurangi kesalahan prediksi model sebelumnya dan memfokuskan pada pola yang tidak terduga dalam data. Secara khusus, algoritma ini menggunakan pendekatan pemberian bobot pada setiap model, memberikan penekanan lebih pada data yang sulit dijelaskan oleh model sebelumnya. Dengan melakukan iterasi sejumlah kali, GBA menghasilkan model yang mampu mengatasi kompleksitas dan non-linearitas dalam data, menjadikannya pilihan yang efektif untuk analisis kluster dalam konteks lama tempo pekerjaan yang melibatkan variabel pekerjaan yang beragam. Keunggulan GBA terletak pada kemampuannya untuk mengatasi tantangan data yang kompleks dan memberikan solusi yang akurat dan efisien dalam analisis kluster.

3.4. Analisis dan Interpretasi Hasil

Hasil klusterisasi akan dianalisis secara mendalam untuk mengidentifikasi pola lama tempo pekerjaan yang muncul. Setiap

kluster akan diinterpretasikan dalam konteks variabel-variabel yang digunakan, dan temuan signifikan akan dieksplorasi lebih lanjut. Interpretasi ini menjadi dasar untuk mendapatkan wawasan baru terkait tren pekerjaan, kebutuhan karyawan, dan dinamika organisasi.

4. HASIL DAN PEMBAHASAN

4.1. Pre-processing data

Import Library

Dalam implementasi kode yang diberikan, terdapat penggunaan beberapa library kunci yang berperan penting dalam analisis dan pengembangan model klasifikasi. Pertama, kita mengimport pandas untuk memanipulasi dan mengelola data dalam format DataFrame. Selanjutnya, `sklearn.model_selection` digunakan untuk membagi dataset menjadi set pelatihan dan pengujian dengan `train_test_split`, memungkinkan kita untuk melakukan evaluasi yang lebih obyektif terhadap kinerja model. Penggunaan `sklearn.ensemble` dengan `GradientBoostingClassifier` memberikan kita kemampuan untuk mengimplementasikan model Boosting, yang dikenal karena kemampuannya meningkatkan performa model melalui kombinasi dari beberapa model lemah.

Selanjutnya, penggunaan `sklearn.metrics`, termasuk `accuracy_score`, `classification_report`, dan `confusion_matrix`, memberikan kita alat untuk mengevaluasi kinerja model. `Accuracy_score` memberikan informasi tentang akurasi prediksi model, sedangkan `classification_report` dan `confusion_matrix` memberikan pemahaman yang lebih mendalam terkait performa model pada setiap kelas target.

Penggunaan `LabelEncoder` dari `sklearn.preprocessing` berguna dalam menangani variabel kategorikal dengan mengonversinya menjadi bentuk numerik. Hal ini penting untuk memastikan model dapat memahami dan memproses data dengan efisien. Selain itu, visualisasi hasil dapat ditingkatkan melalui penggunaan `matplotlib` dan `seaborn`, seperti yang ditunjukkan dengan import `plt` dan `sns`.

Secara keseluruhan, penggunaan library-library ini menciptakan kerangka kerja yang solid untuk analisis klasifikasi,

memungkinkan eksplorasi data, pembagian dataset, pengembangan model, dan evaluasi performa dengan lebih efektif.

Label Encoder

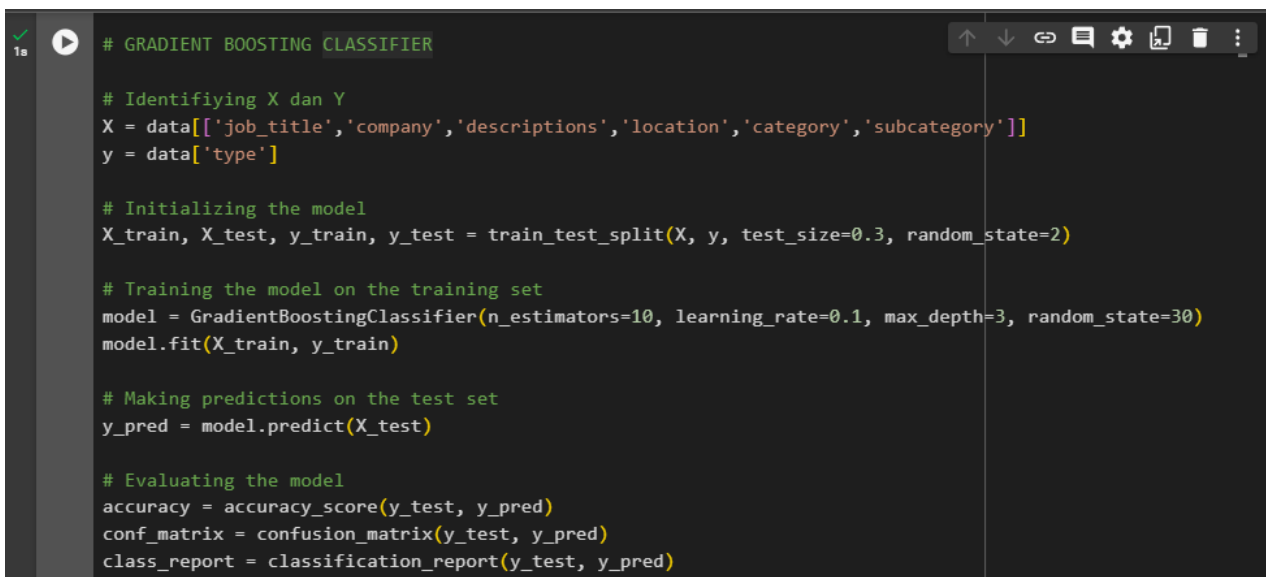
Implementasi `Label Encoder` pada kode tersebut menunjukkan pendekatan yang efisien dalam menangani variabel kategorikal dalam dataset. Penggunaan `LabelEncoder` dari `scikit-learn` memungkinkan transformasi variabel kategorikal menjadi bentuk numerik, yang merupakan langkah kritis dalam persiapan data untuk model machine learning.

Pada kode tersebut, sebuah loop digunakan untuk iterasi melalui setiap kolom dalam DataFrame 'data'. Jika tipe data kolom tersebut adalah 'object', yang menandakan bahwa itu adalah variabel kategorikal, maka `Label Encoder` akan diterapkan. Proses ini akan mengonversi nilai-nilai kategori menjadi bilangan bulat secara otomatis, menggantikan kategori-kategori tersebut dengan representasi numerik. Dengan melakukan ini, model machine learning dapat lebih efektif memahami dan memproses informasi, karena algoritma cenderung berkinerja lebih baik dengan data yang bersifat numerik.

Pendekatan ini menjadi penting terutama ketika menggunakan model-machine learning yang memerlukan input numerik, seperti `Gradient Boosting Classifier` dalam contoh sebelumnya. `Label Encoder` dapat meningkatkan interpretabilitas data dan membantu mencegah bias yang mungkin muncul ketika menggunakan variabel kategorikal tanpa transformasi. Oleh karena itu, implementasi `Label Encoder` pada kode tersebut dapat dianggap sebagai langkah penting dalam pra-pemrosesan data sebelum melibatkan model machine learning.

4.2. Processing data

Dalam implementasi `Gradient Boosting Classifier` pada kode Gambar 1 tersebut, langkah-langkah analisis dan pengembangan model klasifikasi dilakukan dengan hati-hati. Pertama-tama, variabel independen (X) dan variabel dependen (y) diidentifikasi dengan memilih kolom-kolom yang relevan dari dataset, termasuk 'job_title', 'company', 'descriptions', 'location', 'category', dan 'subcategory' sebagai fitur (X), serta 'type' sebagai target (y).



```
# GRADIENT BOOSTING CLASSIFIER

# Identifying X dan Y
X = data[['job_title', 'company', 'descriptions', 'location', 'category', 'subcategory']]
y = data['type']

# Initializing the model
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=2)

# Training the model on the training set
model = GradientBoostingClassifier(n_estimators=10, learning_rate=0.1, max_depth=3, random_state=30)
model.fit(X_train, y_train)

# Making predictions on the test set
y_pred = model.predict(X_test)

# Evaluating the model
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
class_report = classification_report(y_test, y_pred)
```

Gambar 1. Processing Data dengan GBC



Selanjutnya, dataset dibagi menjadi set pelatihan (training set) dan set pengujian (test set) menggunakan fungsi `train_test_split` dari `scikit-learn`. Dengan menggunakan 70% data sebagai training set dan 30% sebagai test set, model dapat dilatih dengan baik dan diuji secara obyektif.

Model Gradient Boosting Classifier diinisialisasi dengan parameter-parameter tertentu, seperti jumlah pohon keputusan (`n_estimators`), tingkat pembelajaran (`learning_rate`), kedalaman maksimum pohon (`max_depth`), dan seed acak (`random_state`). Model kemudian dilatih menggunakan data pelatihan (`X_train`, `y_train`) dengan menggunakan fungsi `fit()`.

Setelah model dilatih, prediksi dibuat pada data pengujian (`X_test`) dengan menggunakan fungsi `predict()`. Evaluasi kinerja model dilakukan dengan menghitung akurasi (`accuracy`), matriks kebingungan (`confusion matrix`), dan laporan klasifikasi (`classification report`) menggunakan fungsi-fungsi dari `scikit-learn`.

Hasil evaluasi tersebut memberikan pemahaman mendalam tentang sejauh mana model mampu mengklasifikasikan data pengujian. Akurasi memberikan indikasi persentase prediksi yang benar, sementara matriks kebingungan memberikan gambaran lebih rinci tentang jumlah prediksi yang benar dan salah pada setiap kelas. Laporan klasifikasi memberikan metrik evaluasi seperti presisi, recall, dan f1-score untuk setiap kelas target.

Dengan memahami dan menganalisis hasil evaluasi tersebut, peneliti atau praktisi dapat membuat penyesuaian dan perbaikan pada model untuk meningkatkan kinerjanya atau membuat keputusan yang lebih tepat dalam konteks aplikasi tertentu.

4.3. Hasil Pengolahan Data

Pengolahan data menggunakan aturan Label Encoder:

Label 0 is Contract/Temp

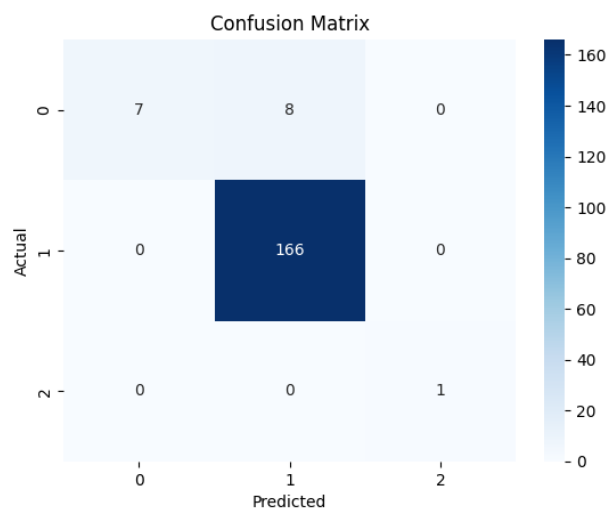
Label 1 is Full time

Label 2 is Part time

Model yang dihasilkan oleh algoritma memiliki tingkat akurasi sebesar 96%, yang mengindikasikan kemampuan model untuk membuat prediksi yang benar terhadap data pengujian. Dengan mengacu pada tabel presisi, recall, f1-score, dan support, kita dapat menggali lebih dalam tentang performa model untuk setiap label kelas. Untuk kelas 0, model memiliki presisi sebesar 1.00, menunjukkan bahwa semua prediksi yang diberikan oleh model untuk kelas ini adalah benar, meskipun recall hanya sebesar 0.47, yang berarti model tidak dapat mengidentifikasi secara efektif semua instance positif sebenarnya. Sebaliknya, untuk kelas 1, model memiliki presisi sebesar 0.95 dan recall sebesar 1.00, menunjukkan keseimbangan yang baik antara akurasi positif dan negatif. Untuk kelas 2, model mencapai hasil yang sempurna dengan presisi, recall, dan f1-score sebesar 1.00, namun harus dicatat bahwa support hanya 1, yang berarti hanya ada satu instance dari kelas ini dalam data pengujian. Dengan demikian, evaluasi ini memberikan gambaran menyeluruh tentang keandalan model dalam mengklasifikasikan data untuk masing-masing kelas.

<https://doi.org/10.25077/>

Confusion matrix memberikan pandangan rinci tentang performa model klasifikasi pada setiap kelas seperti pada Gambar 2. Dalam konteks ini, confusion matrix terdiri dari tiga kelas: Contract/Temp (Label 0), Full time (Label 1), dan Part time (Label 2).



Gambar 2. Confusion Matrix

Dari matriks tersebut, dapat dilihat bahwa model melakukan 7 prediksi benar untuk kelas Contract/Temp, 166 prediksi benar untuk kelas Full time, dan 1 prediksi benar untuk kelas Part time. Namun, model juga menghasilkan 8 prediksi yang salah untuk kelas Contract/Temp dan tidak membuat kesalahan pada kelas Full time dan Part time. Kesalahan ini terkonsentrasi pada kelas Contract/Temp, menunjukkan bahwa model memiliki tantangan dalam mengklasifikasikan dengan akurat data yang termasuk dalam kelas tersebut. Sementara itu, performa yang sangat baik pada kelas Full time dan Part time dapat diartikan sebagai kemampuan model untuk mengidentifikasi dan membedakan kedua kelas tersebut dengan sangat baik. Sehingga, analisis confusion matrix ini memberikan wawasan yang lebih mendalam terkait kekuatan dan kelemahan model dalam mengklasifikasikan setiap kelas.

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dalam penelitian ini, dilakukan analisis dan pengembangan model klasifikasi lama tempo pekerjaan dengan menggunakan Gradient Boosting Classifier. Proses ini melibatkan beberapa langkah, termasuk pre-processing data, label encoding, dan pemrosesan data menggunakan algoritma machine learning. Berikut adalah kesimpulan yang dapat diambil dari hasil penelitian ini:

- Penggunaan beberapa library kunci seperti `pandas`, `scikit-learn`, `matplotlib`, dan `seaborn` memberikan kerangka kerja yang solid untuk analisis dan

pengembangan model klasifikasi. Library tersebut memfasilitasi manipulasi data, pembagian dataset, pengembangan model, dan evaluasi performa.

- Implementasi Label Encoder pada data kategorikal membuktikan keefisienan dalam mengonversi variabel kategorikal menjadi bentuk numerik. Langkah ini kritis untuk memastikan model machine learning dapat memproses data dengan efisien dan menghindari bias pada variabel kategorikal.
- Model yang dihasilkan dari algoritma Gradient Boosting Classifier mencapai tingkat akurasi sebesar 96%. Evaluasi performa melibatkan analisis presisi, recall, f1-score, dan confusion matrix untuk setiap kelas target.
- Confusion matrix memberikan wawasan mendalam tentang kekuatan dan kelemahan model dalam mengklasifikasikan setiap kelas. Fokus kesalahan terutama terkonsentrasi pada kelas Contract/Temp, sedangkan kelas lainnya memiliki performa yang sangat baik.
- Secara keseluruhan, penelitian ini memberikan pemahaman yang mendalam tentang lama tempo pekerjaan dan implementasi model klasifikasi untuk mengklasifikasikan jenis pekerjaan. Hasil evaluasi dapat digunakan sebagai dasar untuk membuat penyesuaian model atau keputusan yang lebih tepat dalam konteks aplikasi tertentu. Penelitian ini memberikan kontribusi pada pemahaman tentang bagaimana algoritma machine learning, khususnya Gradient Boosting Classifier, dapat digunakan dalam menganalisis dan mengklasifikasikan lama tempo pekerjaan. Dengan pemahaman ini, organisasi dapat mengambil langkah-langkah yang lebih informasional dan tepat dalam manajemen sumber daya manusia dan perencanaan tenaga kerja.

5.2. Saran

Berikut adalah beberapa saran yang dapat diterapkan untuk meningkatkan atau mengembangkan penelitian ini:

- Pertimbangkan untuk menambahkan fitur tambahan yang mungkin mempengaruhi lama tempo pekerjaan, seperti faktor-faktor lingkungan kerja, kepuasan pekerja, atau tren industri. Ini dapat meningkatkan keakuratan model dan memberikan wawasan lebih mendalam.
- Lakukan penyesuaian parameter pada model Gradient Boosting Classifier atau pertimbangkan untuk menggunakan model lain yang mungkin lebih cocok dengan karakteristik data. Hal ini dapat dilakukan melalui teknik penalaan atau pencarian grid untuk menemukan parameter optimal.
- Pertimbangkan untuk mengumpulkan lebih banyak data, terutama pada kelas yang memiliki support rendah. Data yang lebih banyak dapat membantu meningkatkan representasi dan keakuratan model.
- Cobalah menggunakan berbagai algoritma machine learning untuk membandingkan kinerja dan mengeksplorasi pendekatan lain dalam analisis lama tempo pekerjaan.

DAFTAR PUSTAKA

- [1] K. Bozkus, "Organizational Culture Change and Technology: Navigating the Digital Transformation," *Intech Open*, pp. 1-22, 2023.
- [2] A. A. Davidescu, S.-A. Apostu, A. Paul dan I. Casuneanu, "Work Flexibility, Job Satisfaction, and Job Performance among Romanian Employees— Implications for Sustainable Human Resource Management," *sustainability*, vol. 12, pp. 1-53, 2020.
- [3] U. Dhanda dan V. K. Shrotryia, "Corporate sustainability: the new organizational reality," *Corporate sustainability*, 2020.
- [4] A. S. Atichasari, A. Ratnasari, U. Kulsum, H. S. Kahp, S. S. Wulandari dan A. Marfu, "Examining non-performing loans on corporate financial sustainability: Evidence from Indonesia," *Sustainable Futures*, vol. 6, 2023.
- [5] E. R. Peeters, M. C. Canifels dan M. Verbruggen, "Dust yourself off and try again: the positive process of career changes or shocks and career resilience," *Emerald*, vol. 27, no. 3, pp. 372-390, 2022.
- [6] P. Garengo, A. Sardi dan S. S. Nudurupati, "Human resource management (HRM) in the performance measurement and management (PMM) domain: a bibliometric review," *International Journal of Productivity and Performance Management*, vol. 71, no. 7, pp. 3056-3077, 2022.
- [7] N. sultana dan S. Goswami, "Impact of Human Resource Management Practice, on Employee Retention Strategy: An Empirical Analysis," *IJCRT*, vol. 8, no. 7, pp. 4492-4499, 2020.
- [8] Z. Hassan, "Employee retention through effective human resource management practices in Maldives: Mediation effects of compensation and rewards system," *Journal of Entrepreneurship, Management and Innovation*, vol. 18, no. 2, pp. 137-173, 2022.
- [9] S. M. Ganie, P. K. D. Pramanik, M. B. Malik, S. Mallik dan H. Qin, "An ensemble learning approach for diabetes prediction using boosting techniques," *Frontiers in Genetics*, pp. 1-15, 2023.
- [10] Mustikasari, A. R. Ramli dan A. K. Gibran, "Analisis Kinerja Algoritma Machine Learning Untuk Deteksi Penyakit Daun Teh Dengan Particle Swarm Optimization," *JESSI*, vol. 4, no. 2, pp. 97-106, 2023.