PERBANDINGAN AKURASI BALANCE DAN IMBALANCE DATASET PADA KLASIFIKASI GAJI KARYAWAN MENGGUNAKAN METODE RANDOM FOREST

Kelana Chandra Helyandika[1], Muhammad Ariq Daffa[2], Rhafli Annuralim[3]

[1,2,3]Informatika, Fakultas Teknik, Universitas Jenderal Soedirman, Indonesia
NIM: [1]H1D020024, [2]H1D020064, [3]H1D020076
Email: [1]kelana.helyandika@mhs.unsoed.ac.id, [2]ariq.daffa@mhs.unsoed.ac.id,
[3]rhafli.annuralim@mhs.unsoed.ac.id

(Artikel dikirimkan tanggal : 04-06-2022)

Abstrak

Dalam dunia pekerjaan, karyawan mendapat upah atas hasil kerjanya yang diberikan oleh seorang majikan. Perhitungan gaji harus dilakukan secara detail agar menemukan nominal gaji yang sesuai dengan suatu pekerjaan atau jabatan. Tujuan penelitian ini bertujuan untuk menghitung perbandingan akurasi dataset klasifikasi gaji karyawan. Penelitian ini menggunakan sampel dataset yang kami dapatkan di website *Kaggle*. Metode yang digunakan dalam penelitian ini adalah metode *Random Forest*, yang akan membantu meningatkan akurasi pada perbandingan yang akan dilakukan. Hasil pengujian menunjukan akurasi perbandingan dan akurasi nominal gaji yang terdapat di dataset yang digunakan.
Kata kunci: *Dataset, Gaji, Kaggle, Random Forest.*

Abstract

In the world of work, employees get paid for their work given by an employer. Salary calculations must be done in detail in order to find a nominal salary that is in accordance with a job or position. The purpose of this study aims to calculate the comparison of the accuracy of the employee salary classification dataset. The study used a sample of datasets we obtained on the Kaggle website. The method used in this study is the Random Forest method, which will help increase the accuracy of the comparison to be made. The test results show the accuracy of comparison and nominal accuracy of salaries contained in the dataset used.
Keywords: *Dataset, Kaggle, Random Forest, Salary.*

1. PENDAHULUAN

Pemanfaatan perkembangan teknologi saat ini tidak lagi pada bidang tertentu saja, perkembangan yang begitu pesat berpengaruh pada kemudahan semua aktifitas yang sulit untuk dilakukan manusia dapat dikerjakan dengan mudan, efektif dan efisien. Dengan peranan teknologi bidang komputasi tentunya sangat memberikan peluang untuk menyelesaikan permasalahan yang kompleks.

Gaji adalah suatu kompensasi yang dibayarkan oleh organisasi kepada pegawai sebagai balas jasa atau kinerja yang telah diberikan oleh terhadap organisasi[1]. Dari sudut pandang pelaksanaan bisnis, gaji dapat dianggap sebagai biaya yang dibutuhkan untuk mendapatkan sumber daya manusia untuk menjalankan operasi, dan karenanya disebut dengan biaya personel atau biaya gaji. Gaji merupakan sebuah hal penting bagi seorang pekerja dan orang yang memberi pekerjaan, karena hal tersebut penentuan gaji adalah hal yang krusial, pemberi gaji biasanya akan menentukan besaran gaji yang akan diberikan berdasarkan berbagai kriteria yang dipertimbangkan. Karena hal tersebut jika penentuan dilakukan secara manual akan kesulitan untuk menentukan secara cepat dan efisien. Jika saat penentuan secara manual terjadi kesalahan dalam penentuan gaji, maka hal tersebut dapat mengakibatkan kerugian baik bagi pemberi maupun penerima gaji tersebut. Kemungkinan kesalahan tersebut dapat dikurangi kemungkinan terjadinya dengan menggunakan bantuan teknologi untuk membantu penentuan gaji baik gaji secara spesifik maupun penentuan batas gaji seperti seseorang akan menerima gaji di bawah atau di atas standar yang sudah ditentukan. Salah satu cara yang dapat digunakan adalah menggunakan bantuan sebuah metode kecerdasan buatan.

Artificial Intelligence (Kecerdasan Buatan) merupakan salah satu bagian ilmu computer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia. Pada awal diciptakannya, komputer hanya difungsikan sebagai alat hitung saja. Namun seiring dengan perkebangan jaman, maka peran computer semakin mendominasi kehidupan umat manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, lebih dari itu, computer diharapkan untuk dapat diberdayakan untuk mengerjakan segala sesuatu yang bisa dikerjakan oleh manusia[2]. Machine Learning memberi kita teknik terbaik untuk kecerdasan buatan seperti klasifikasi, regresi, pembelajaran terawasi dan pembelajaran tanpa pengawasan dan banyak lagi. Kita dapat menggunakan pengklasifikasi apapun seperti Decision Tree, Naïve Bayes, dan banyak lagi.

Berbagai jenis algoritma pemilihan fitur tersedia untuk memilih fitur yang terbaik dan meminimalkan kumpulan data. Karena ini adalah masalah pengoptimalan maka banyak teknik yang digunakan untuk mengoptimalkan atau mengurangi dimensi dari dataset[3].

Data Mining adalah proses yang mempekerjakan satu atau lebih teknik pembelajaran computer (machine learning) untuk menganalisis dan mengekstraksi pengetahuan (knowledge) secara otomatis[4]. Data mining biasanya mengolah data dari database dengan ukuran yang besar. Dari data tersebut dilakukan pencarian pola atau trendsesuai dengan tujuan dari penerapan data mining tersebut. Hasil dari pengolahan data miningtersebut selanjutnya dapat digunakan untuk pengambilan keputusan maupun analisis yang dibutuhkan[5]. Salah satu contoh penerapan atau metode dalam sebuah kasus dari Data Mining adalah dengan menggunakan Metode Random Forest digunakan dalam penentuan klasifikasi gaji karyawan. Dalam menentukan proses penentuan klasifikasi gaji karyawan diperlukan sebuah dataset gaji. Random forest didasarkan pada teknik pohon keputusan sehingga mampu mengatasi masalah non-linier. Metode ini merupakan metode pohon gabungan. Untuk mengidentifikasi peubah penjelas yang relevan dengan peubah respons, random forest menghasilkan ukuran tingkat kepentingan (variable importance) peubah penjelas[6]. Random Forest merupakan modifikasi dari Decision Tree yang dimana dapat meningkatkan akurasi karena adanya pemilihan secara acak dalam membangkitkan simpul anak untuk setiap node (simpul diatasnya) dan diakumulasikan hasil klasifikasi dari setiap pohon (tree), kemudian dipilih hasil klasifikasi yang paling banyak muncul. Pada paper ini digunakan metode Random Forest dalam membandingkan sebuah dataset gaji karyawan yang sudah balance dan imbalance[7]. Karena memang pada kasus nyata akan sangat memungkinkan ditemukan sebuah dataset yang memiliki output target yang jumlahnya tidak seimbang, karena hal tersebut perlu diketahui apakah penggunaan dataset yang memiliki output target yang tidak seimbang tersebut dapat mempengaruhi performa dari model salah satunya dari model algoritma random forest.

Hasil yang diharapkan dari penelitian ini adalah akan mengetahui bagaimana pengaruh dan hasil perbandingan akurasi dan berbagai metrik evaluasi lainnya dari algoritma random forest dengan berbagai skema model yang digunakan dengan menggunakan dataset yang seimbang dan tidak seimbang pada bagian target pada training dataset yang akan dujikan pada testing datasetnya.

2. METODE

2.1 Dataset

Dataset yang kami gunakan dalam penelitian ini adalah Salary Prediction Classification yang didapat dari web Kaggle. Tujuan prediksi ini adalah untuk menentukan apakah seseorang akan mendapatkan gaji lebih dari 50 ribu dollar dalam setahun atau tidak.

Pada dataset ini terdapat dua jenis variabel. Adapun dua jenis variabel tersebut yaitu dua belas variabel prediksi dan satu variabel respon. Pada variabel prediksi terdapat variabel age, workclass, education, education-num, marital-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week, native-country. Dan variabel salary sebagai variabel respon dengan jenis respon berbentuk binary respon dimana terdiri dari dua buah kategori target yaitu “<=50K” dan “>50K”[3].

Pada dataset awal terdapat 32561 data dan setelah dilakukan proses pre-processing awal didapat hasil dataset yang siap digunakan berjumlah 31929 yang kedepannya akan dibagi menjadi dua skema data training (Balance dan Imbalance) dan data testing.

2.2 Preprocessing

Pre-processing data adalah proses pembersihan dan mempersiapkan teks untuk klasifikasi[8]**.** Preprocessing merupakan tahapan dalam proses data mining dimana dataset yang akan digunakan akan diproses sebelum nanti akhirnya akan siap digunakan oleh model yang dibuat.

Pada penelitian kali ini akan dibuat dua jenis model preprocessing pada dataset dimana yang pertama adalah pada semua kolom kategorikal akan dilakukan proses Ordinal Encoding. Ordinal Encoding merupakan sebuah cara dimana akan mengubah kolom kategorikal menjadi kolom numerik dengan mengubah tiap jenis kategori menjadi sebuah bilangan mulai dari 0,1,2,..,n.

Pada jenis preprocessing yang kedua adalah perlakuan yang akan diberikan pada dataset adalah pada kolom kategorikal akan ada kolom yang akan dipreproses menggunakan Ordinal Encoding dan ada kolom yang akan diproses dengan menggunakan One Hot Encoding. One Hot Encoding adalah metode encoding yang akan merepresentasikan data bertipe kategori sebagai vektor biner yang bernilai integer, 0 dan 1, dimana semua elemen akan bernilai 0 kecuali satu elemen bernilai 1, yaitu elemen yang memiliki nilai kategori tersebut. Pembagian kolom yang menggunakan Ordinal encoding adalah kolom education, education-num, native country, dan occupation. Kemudian untuk kolom yang menggunakan One Hot Encoding adalah workclass, relationship, race, sex, dan marital-status. Terakhir pada proses kedua ini juga pada kolom numerik akan dilakukan standarisasi dengan rumus berikut

$$z = \frac{(x - u)}{s}$$

Keterangan:

z = nilai standarisasi

x = nilai data ke-n

u = rataan kolom

s = standar deviasi kolom

2.3 Undersampling

Random under sampling melakukan pemilihan data secara acak dari kelas mayoritas untuk dihapus dari kumpulan data training. Dengan menjalankan random under sampling, maka data training dari kelas mayoritas akan berkurang jumlahnya. Proses under sampling dapat diulang hingga diperoleh distribusi kelas yang diinginkan pada data training. Pendekatan ini dapat diterapkan pada kumpulan data dengan kelas yang tidak seimbang dimana kelas minoritas cukup untuk pembuatan model. Kekurangan dari undersampling adalah data dari kelas mayoritas yang dihapus adalah data acak sehingga ada kemungkinan data tersebut adalah data yang berguna atau bahkan sangat penting dalam pembangunan model klasifikasi yang baik. Pada under sampling, dimungkinkan dibuat jumlah data yang sama dari kelas mayoritas dan minoritas atau hanya mengurangi data mayoritas hingga jumlah tertentu[9]. Kelebihan undersampling adalah penghapusan beberapa data dapat secara signifikan mengurangi ukuran data sehingga dapat menurunkan biaya run-time terutama dalam kasus data yang besar[10].

2.4 Random Forest

Random forest merupakan sebuah model ensemble, yaitu model yang dibentuk dari banyak model Decision Tree, baik untuk regresi maupun untuk klasifikasi, dengan menerapkan metode bootstrap aggregating (bagging) dan random feature selection, serta merupakan metode yang dapat meningkatkan hasil akurasi, karena dalam membangkitkan simpul anak untuk setiap node dilakukan secara acak. Metode ini digunakan untuk membangun pohon keputusan yang terdiri dari root node, internal node, dan leaf node dengan mengambil atribut dan data secara acak sesuai ketentuan yang diberlakukan. Root node merupakan simpul yang terletak paling atas, atau biasa disebut sebagai akar dari pohon keputusan. Internal node adalah simpul percabangan, dimana node ini mempunyai output minimal dua dan hanya ada satu input. Sedangkan leaf node atau terminal node merupakan simpul terakhir yang hanya memiliki satu input dan tidak mempunyai output. Pohon keputusan dimulai dengan cara menghitung nilai entropy sebagai penentu tingkat ketidakmurnian atribut dan nilai information gain[3]. Algoritma Random Forest menunjukkan beberapa kelebihan diantaranya mampu menghasilkan error yang relative rendah, performa yang baik dalam klasifikasi, dapat mengatasi data pelatihan dalam jumlah besar secara efisien, serta metode yang efektif untuk mengestimasi missing data[11].

2.5 Randomized Grid Search

Randomized Grid Search merupakan versi lain dari Grid Search. Grid Search merupakan pencarian lengkap berdasarkan subset ruang *hyperparameter*. Grid search akan membagi jangkauan parameter yang akan dioptimalkan dimana dalam kasus ini akan mengoptimalkan akurasi dari model yang dibuat ke dalam grid dan akan menggunakan semua kemungkinan yang ada pada subset yang digunakan. *Grid Search* akan mengoptimalkan akurasi dari Random Forest menggunakan teknik cross validasi sebagai metrik kinerja. Tujuannya adalah untuk mengidentifikasi kombinasi *hyperparameter* terbaik yang menghasilkan nilai akurasi terbaik, yang membedakan Grid Search dengan Randomized Grid Search adalah menggunakan Randomized Grid Search tidak akan mencari semua kemungkinan yang ada, hanya mencari beberapa kombinasi dari subset dengan jumlah kombinasi yang sudah ditentukan dengan tujuan yang sama dengan Grid Search. Randomized Grid Search digunakan untuk mengatasi proses perhitungan yang lama dari metode Grid Search. Prinsip perhitungannya adalah dengan mengkalikan parameter model dengan bilang acak yang probabilitasnya sama pada setiap datum dengan interval antara 0 sampai 1 [12].

2.6 K-Fold Cross Validasi

Cross-validasi atau dapat disebut estimasi rotasi adalah sebuah teknik validasi model untuk menilai bagaimana hasil statistik analisis akan menggeneralisasi kumpulan data independen. Teknik ini utamanya digunakan untuk melakukan prediksi model dan memperkirakan seberapa akurat sebuah model prediktif ketika dijalankan dalam praktiknya. Salah satu teknik dari validasi silang adalah k-fold cross validation, yang mana memecah data menjadi k bagian set data dengan ukuran yang sama. Penggunaan k-fold cross validation untuk menghilangkan bias pada data. Pelatihan dan pengujian dilakukan sebanyak k kali[13].

2.7 Perhitungan Akurasi

Akurasi diperlukan untuk evaluasi dan mengukur keakuratan dari hasil klasifikasi, semakain besar nilai akurasi maka semakin baik tingkat klasifikasinya[14]. Kita dapat menghitung nilai akurasi dengan rumus:

$$Accuracy = \frac{Jumlah\ Dokumen\ yang\ terklasifikasi}{Jumlah\ Dokumen\ Keseluruhan} x100\%$$

2.8 Perhitungan Precision, Recall, dan F1-Score

Precision Mengukur tingkat kepastian (exactness) atau jumlah data testing yang diklasifikasikan dengan benar oleh model klasifikasi yang dibangun[15].

Rumus Precision(pre):

$$pre = \frac{TP}{FP + TP}$$

Sedangkan pada Recall mengukur sensitivitas atau rasio dari data untuk setiap label yang diklasifikasikan dengan benar terhadap data yang salah diklasifikasikan ke label lainnya[15].

Rumus Recall(rec):

$$rec = \frac{TP}{FN + TP}$$

Selanjutnya hasil perhitungan Precision dan Recall yang dibobotkan dapat diperhitungkan perbandingannya untuk mendapat F1-Score[15]. F1-Score dapat dihitung dengan rumus :

$$f_{score} = \frac{2xRecallxPrecision}{Recall\ +\ Precision}$$

3. HASIL DAN PEMBAHASAN

Pada bab ini akan membahas hasil dari penelitian yang dilakukan tentang perbandingan akurasi random forest yang nantinya akan menggunakan balanced dan imbalanced dataset. Untuk pembagian dataset awal dari sebuah dataset utuh dibagi menjadi dua yaitu training dataset dan testing dataset dibagi dengan rasio 80:20 dengan 20% akan menjadi testing dataset.

Kemudian training dataset awal pada data targetnya memiliki rasio yang tidak seimbang sehingga dataset training awal ini yang digunakan sebagai dataset training imbalanced. Kemudian untuk mendapatkan balanced dataset akan digunakan imbalanced dataset sebagai dataset yang dimana targetnya akan dibuat seimbang dengan menggunakan teknik undersampling dimana hasil yang didapatkan adalah target dari dataset tersebut akan menjadi seimbang sehingga menjadi balanced training dataset dan kemudian untuk testing hanya menggunakan satu testing dataset dengan pembagian seperti pada tabel 1.

Tabel 1. Jumlah Pembagian Dataset

Dataset	<=50K	>50K	Total
Imbalanced Training	19752	5791	25543
Balanced Training	5791	5791	11582
Testing Dataset	4938	1448	6386

3.1. Training Model

Setelah pembagian dataset dilakukan, kemudian adalah proses training pada model Random Forest. Pada proses sebelum dataset training dilakukan fitting terhadap model maka dilakukan preprocessing terhadap model dengan dua jenis model preprocessing yang sudah dijelaskan pada bagian Preprocessing dan yang pertama ini dilakukan fitting terhadap imbalanced training dataset. Kemudian setelah dilakukan pada masing - masing proses preprocessing dan dilakukan fitting maka akan tercipta dua model berbeda dan dari masing - masing model preprocessing yang berbeda yaitu model satu menggunakan hanya ordinal encoding pada data trainingnya dan model satunya akan menggunakan ordinal encoding, one hot encoding, dan standarisasi data pada data trainingnya.

Dari dua model yang sudah tercipta sebelumnya yang dilakukan fitting terhadap imbalanced train dataset, kemudian masing - masing model dicari parameter terbaiknya menggunakan Randomized Grid Search dengan pencarian acak terhadap 10 kali kombinasi acak hyperparameter dari Random Forest. Setelah ditemukan kombinasi acak terbaik yang menghasilkan nilai akurasi cross validasi terbaik maka dilakukan fitting kembali pada masing - masing model tadi, dimana berarti pada akhirnya akan menghasilkan total empat buah model Random Forest dengan penjelasan seperti pada tabel berikut

Tabel 2. Macam Model Random Forest

Model	Keterangan
NormalRF	Random Forest menggunakan ordinal encoder pada data training.
NormalRF GS	Random Forest menggunakan ordinal encoder pada data training dan dilakukan penyesuaian pada *hyperparameter*nya menggunakan Randomized Grid Search.
ScalingRF	Random Forest menggunakan ordinal encoder, one hot encoder, dan standarisasi data pada data training.

Scaling RF GS	Random Forest menggunakan ordinal encoder, one hot encoder, dan standarisasi data pada data training dan dilakukan penyesuaian pada *hyperparameter*nya menggunakan Randomized Grid Search

Maka tercipta total empat buah model Random Forest yang sudah dilakukan fitting dengan imbalanced training dataset. Kemudian dari empat model tersebut dilakukan refitting kembali, namun kali ini dari empat model tersebut dilakukan fitting pada balanced training dataset. Karena hal itu, berarti total model yang ada adalah delapan model yaitu masing - masing 4 model untuk masing - masing training dataset yang digunakan

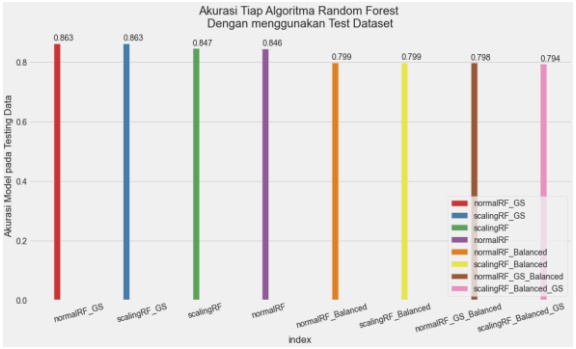
3.2. Testing Model

Kemudian proses selanjutnya yang dilakukan adalah melakukan testing terhadap keseluruhan delapan buah model yang dibuat dengan menggunakan satu jenis testing dataset yang sama untuk semua model yang ditest.

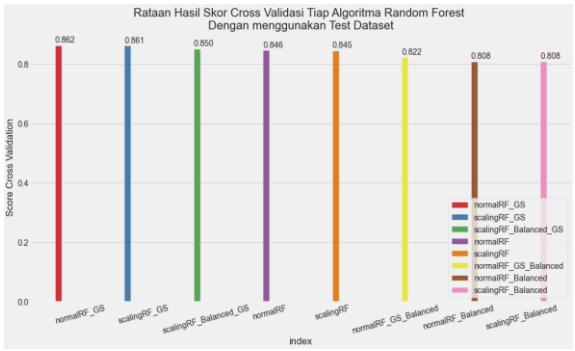
Dari hasil testing yang dilakukan dengan menggunakan beberapa metrik perhitungan untuk menghitung kualitas model yang dihasilkan untuk yang pertama adalah melihat akurasi dan skor dari cross validasi yang dilakukan, hasil secara rinci dapat terlihat pada tabel 3 dan secara visual dapat terlihat pada gambar 1 dan gambar 2.

Tabel 3. Macam Model Random Forest

Model	Jenis Dataset	Akurasi	Skor Cross Validasi
NormalRF	Imbalanced	84.6%	84.6%
	Balanced	79.9%	80.8%
NormalRF GS	Imbalanced	86.3%	86.2%
	Balanced	79.8%	82.2%
ScalingRF	Imbalanced	84.7%	84.5%
	Balanced	79.9%	80.8%
Scaling RF GS	Imbalanced	86.3%	86.1%
	Balanced	79.4%	85%



Gambar 1. Akurasi Model Pada Testing Data



Gambar 2. Skor Cross Validasi Model Pada Testing Data

Skor cross validasi yang dihasilkan dari testing dataset dari semua model terlihat hasil yang cukup baik didominasi oleh model yang ditraining oleh Imbalanced Dataset, dimana model Normal RF GS dan Scaling RF GS dengan Imbalanced Dataset merupakan model yang menghasilkan skor terbaik yaitu keduanya sekitar 86.3% dan Normal RF dan juga Scaling RF dengan Balanced Dataset merupakan model dengan nilai skor cross validasi terendah yaitu keduanya sekitar 80.8%

Selain melihat akurasi serta hasil skor cross validasi, berikut adalah hasil testing dari tiap - tiap model per 10% dataset yang digunakan terlihat pada Tabel 4, Tabel 5, dan juga pada Gambar 3

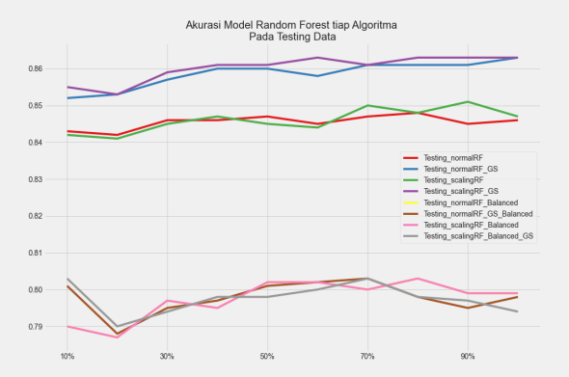
Tabel 4. Akurasi Model Pada Tiap 10% Dataset Pada Imbalanced Dataset

Dataset	Normal RF	Normal RF GS	Scaling RF	Scaling RF GS
10%	84.3%	85.2%	84.2%	85.5%
20%	84.2%	85.3%	84.1%	85.3%
30%	84.6%	85.7%	84.5%	85.9%
40%	84.6%	86%	84.7%	86.1%
50%	84.7%	86%	84.5%	86.1%
60%	84.5%	85.8%	84.4%	86.3%
70%	84.7%	86.1%	85%	86.1%
80%	84.8%	86.1%	84.8%	86.3%
90%	84.5%	86.1%	85.1%	86.3%
100%	84.6.%	86.3%	84.7%	86.3%

Tabel 5. Akurasi Model Pada Tiap 10% Dataset Pada Balanced Dataset

Dataset	Normal RF	Normal RF GS	Scaling RF	Scaling RF GS
10%	79.0%	80.1%	79%	80.3%
20%	78.7%	78.8%	78.7%	79%
30%	79.7%	79.5%	79.7%	79.4%
40%	79.5%	79.7%	79.5%	79.8%
50%	80.2%	80.1%	80.2%	79.8%
60%	80.2%	80.2%	80.2%	80%

70%	80%	80.3%	80%	80.3%
80%	80.3%	79.8%	80.3%	79.8%
90%	79.9%	79.5%	79.9%	79.7%
100%	79.9%	79.8%	79.9%	79.4%



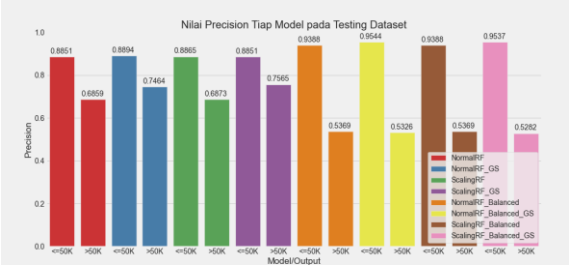
Gambar 3. Akurasi Model Tiap 10% Dataset

Gambar 3 menunjukkan bahwa secara rataan dan hasil akhir menunjukkan bahwa model terbagi menjadi tiga cluster dimana menariknya model yang menggunakan Balanced Dataset sebagai data trainingnya semuanya mempunyai hasil akurasi akhir pada 100% testing dataset tidak lebih dari 8% sedangkan model yang hasilnya cukup baik semuanya dihasilkan oleh model yang menggunakan Imbalanced Dataset sebagai data trainingnya. Kemudian model dengan garis ungu (Scaling RF GS) dan garis biru (Normal RF GS) merupakan model yang secara rataan punya akurasi terbaik dari model lainnya dengan akurasi sekitar 86.3% untuk kedua model tersebut.

Metrik selanjutnya yang akan menjadi pembanding bagi model yang telah dibuat akan dibandingkan hasil *precision*, *recall*, dan *f1-score* dari semua model yang dibuat. Untuk hasil dari *precision* tiap output pada tiap model dapat dilihat pada Tabel 6 dan visualisasinya ada pada Gambar 4

Tabel 6. Skor *Precision* Model Random Forest

Model	Jenis Dataset	<=50K	>50K
NormalRF	Imbalanced	0.8851	0.6859
	Balanced	0.9388	0.5369
NormalRF GS	Imbalanced	0.8894	0.7464
	Balanced	0.9544	0.5326
ScalingRF	Imbalanced	0.8865	0.6873
	Balanced	0.9388	0.5369
Scaling RF GS	Imbalanced	0.8851	0.7565
	Balanced	0.9537	0.5282



Gambar 4. Skor *Precision* Model Random Forest

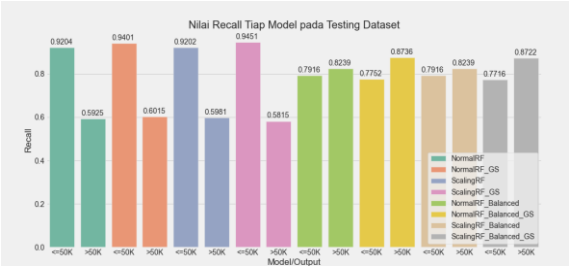
Hasil dari precision untuk semua model terlihat bahwa model yang ditraining dengan Imbalanced Dataset memiliki kecenderungan untuk memiliki hasil yang lebih rata untuk kedua output yang dihasilkan, walaupun pada output <=50K punya kecenderungan lebih rendah daripada model yang ditraining menggunakan Balanced dataset yang rata rata nilai precisionnya pada output <=50K lebih dari 90% namun punya nilai precision pada >50K yang rendah hanya sekitar 0.53 pada seluruh model yang ditraining dengan Balanced Dataset.

Secara rataan dari precision model Scaling RF GS dengan Imbalanced dataset merupakan model menghasilkan nilai precision terbaik untuk kedua buah dengan skor 0.8851 untuk output <=50K dan 0.7565 untuk output >50K.

Perbandingan selanjutnya dapat melihat skor *Recall* pada tiap model yang datanya dapat dilihat pada Tabel 7 dan visualisasinya pada Gambar 5 di bawah ini.

Tabel 7. Skor *Recall* Model Random Forest

Model	Jenis Dataset	<=50K	>50K
NormalRF	Imbalanced	0.9204	0.5925
	Balanced	0.7916	0.8239
NormalRF GS	Imbalanced	0.9401	0.6015
	Balanced	0.7752	0.8736
ScalingRF	Imbalanced	0.9202	0.5981
	Balanced	0.7916	0.8239
Scaling RF GS	Imbalanced	0.9451	0.5815
	Balanced	0.7716	0.8722



Gambar 5. Skor *Recall* Model Random Forest

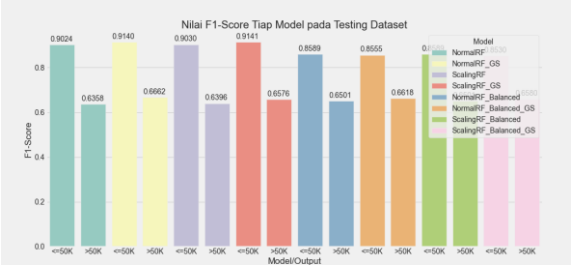
Perbandingan pada hasil skor *Recall* terlihat seperti kebalikan dari skor *Precision* dimana model yang ditraining dengan Balanced Dataset menghasilkan skor yang cukup seimbang untuk output *Recall* pada kedua output yang dihasilkan,

sedangkan model yang di training dengan Imbalanced Dataset memiliki selisih yang cukup jauh antar outputnya.

Karena hal di atas, untuk melihat untuk melihat hasilnya secara general dari hasil *Precision* dan *Recall* dapat dilihat pada *F1-Score*Inya yang datanya ada pada Tabel 8 dan visualisasi ada pada Gambar 6

Tabel 8. *F1-Score* Model Random Forest

Model	Jenis Dataset	<=50K	>50K
NormalRF	Imbalanced	0.9024	0.6358
	Balanced	0.8589	0.6501
NormalRF GS	Imbalanced	0.914	0.6662
	Balanced	0.8555	0.6618
ScalingRF	Imbalanced	0.903	0.6396
	Balanced	0.8589	0.6501
Scaling RF GS	Imbalanced	0.9141	0.6576
	Balanced	0.8530	0.6580



Gambar 6. *F1-Score* Model Random Forest

Melihat hasil *F1-Score* pada semua model mungkin terlihat pada output ">50K" memiliki nilai yang relatif sama pada semua model yang ada, namun jika dilihat pada output "<=50K" model yang menggunakan Imbalanced Dataset sebagai data trainingnya rata - rata memiliki *F1-Score* lebih baik sekitar 6% lebih tinggi dari pada model yang menggunakan Balanced Dataset sebagai data trainingnya pada skema model yang sama.

Dari data pada Tabel 8 maka model terbaik yang didapatkan berdasarkan *F1-Score* adalah model Normal RF GS yang menggunakan Imbalanced Dataset sebagai data trainingnya yang menghasilkan *F1-Score* sebesar 0.914 pada output "<=50K" dan 0.662 pada output ">50K".

4. KESIMPULAN

Dari percobaan yang dilakukan dengan 4 buah skema model Random Forest pada 2 buah training dataset yang menghasilkan 8 model dimana pada hasil akurasi pada testing dataset, model yang menggunakan Imbalanced Dataset menghasilkan rata - rata akurasi lebih baik yaitu sekitar 84.6 - 86.3 persen dari model yang menggunakan Balanced Dataset sebagai data trainingnya yang memiliki rata - rata akurasi 79.9. Begitu juga pada hasil *F1-Score* yang merupakan kombinasi dari *Recall* dan *Precision*, model dengan Imbalanced Dataset sebagai data training cenderung menghasilkan skor yang lebih baik. Begitu juga pada skor cross validasi dimana pada skema yang sama, model akan menghasilkan skor cross validasi pada lebih baik pada testing datanya jika model tersebut ditraining menggunakan Imbalanced Dataset

Melihat pada hasil akurasi pada testing dataset dan *F1-Score* yang merupakan gabungan dari *Precision* dan *Recall*. Hasil percobaan yang dilakukan menemukan bahwa model Normal RF GS yang ditraining menggunakan Imbalanced Dataset merupakan model terbaik yang menghasilkan hasil dengan testing dataset dengan akurasi model 86.3% dan menghasilkan rataan skor cross validasi sebesar 86.2%, kemudian menghasilkan *F1-Score* pada output "<=50K" sebesar 0.914 dan 0.6662 pada output ">50K".

Dapat disimpulkan pada percobaan algoritma Random Forest ini data Imbalanced menghasilkan hasil yang lebih baik daripada data Balanced, hal tersebut menurut kami dikarenakan karena jumlah Imbalanced yang lebih banyak secara totalnya dimana berbeda 14 ribu data dari total data training untuk Balanced Dataset. Berarti jumlah data yang digunakan pada Random Forest juga dapat mempengaruhi hasil dari model yang ada karena adanya perbedaan jumlah yang cukup signifikan antara Imbalanced dan Balanced Dataset. Kemudian dari hasil perbaikan yang dilakukan pada model algoritma menggunakan Randomized Grid Search terlihat memang dengan menggunakan cara tersebut dapat menaikan hasil dari model yang digunakan. Untuk penggunaan variasi cara untuk preprocessing data sebelum dilakukan training ke model tidak mempengaruhi hasil pada output model dengan melihat hasil yang sangat mirip antara model yang hanya menggunakan Ordinal Encoding pada data preprocessingnya dan model yang menggunakan Ordinal Encoding, One Hot Encoding, dan Standarisasi data numerikal pada proses preprocessingnya.

DAFTAR PUSTAKA

[1] Y. Yuliana, A. Arwin, and J. D. Pratiwi, "Dampak Gaji dan Gaya KepemimpinanTerhadap Kepuasan Kerja Karyawan (Studi Kasus Pada PT Cipta Mandiri Agung Jaya)," *BISMA Cendekia*, vol. 1, no. 1, pp. 1–6, 2020.

[2] H. Jaya *et al.*, *Kecerdasan Buatan*, vol. 53, no. 9. 2018.

[3] V. W. Siburian and I. E. Mulyana, "Prediksi Harga Ponsel Menggunakan Metode Random Forest," *J. Sist. Komput.*, vol. 4, no. 1, 2018.

[4] M. G. Sadewo, A. P. Windarto, and D. Hartama, "Penerapan Datamining Pada Populasi Daging Ayam Ras Pedaging Di Indonesia Berdasarkan Provinsi Menggunakan K-Means Clustering," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 2, no. 1, pp. 60–67, 2017, doi: 10.30743/infotekjar.v2i1.164.

[5] S. Widaningsih, "Perbandingan Metode Data Mining Untuk Prediksi Nilai Dan Waktu Kelulusan Mahasiswa Prodi Teknik Informatika Dengan Algoritma C4,5, Naïve Bayes, Knn Dan Svm," *J. Tekno Insentif*, vol. 13, no. 1, pp. 16–25, 2019, doi: 10.36787/jti.v13i1.78.

[6] N. K. Dewi, S. Y. Mulyadi, and U. D. Syafitri, "Penerapan Metode Random Forest Dalam Driver Analysis," *Forum Stat. Dan Komputasi*, vol. 16, no. 1, pp. 35–43, 2012, [Online]. Available: http://journal.ipb.ac.id/index.php/statistika/article/view/5443

[7] T. Purwa, "Perbandingan Metode Regresi Logistik dan Random Forest untuk Klasifikasi Data Imbalanced (Studi Kasus: Klasifikasi Rumah Tangga Miskin di Kabupaten Karangasem, Bali Tahun 2017)," *J. Mat. Stat. dan Komputasi*, vol. 16, no. 1, p. 58, 2019, doi: 10.20956/jmsk.v16i1.6494.

[8] E. Indrayuni, "Klasifikasi Text Mining Review Produk Kosmetik Untuk Teks Bahasa Indonesia Menggunakan Algoritma Naive Bayes," *J. Khatulistiwa Inform.*, vol. 7, no. 1, pp. 29–36, 2019, doi: 10.31294/jki.v7i1.1.

[9] W. I. Sabilla and C. B. Vista, "Implementasi SMOTE dan Under Sampling pada Imbalanced Dataset untuk Prediksi Kebangkrutan Perusahaan," *J. Komput. Terap.*, vol. 7, no. 2, pp. 329–339, 2021, [Online]. Available: https://jurnal.pcr.ac.id/index.php/jkt/

[10] P. S. Sarjana, D. Statistika, F. Matematika, and D. A. N. S. Data, "Penerapan Combine Undersampling Pada Klasifikasi Data Imbalanced Biner (Studi Kasus : Desa Tertinggal Di Jawa Timur Tahun 2014)," 2018.

[11] F. Y. Pamuji and V. P. Ramadhan, "Komparasi Algoritma Random Forest dan Decision Tree untuk Memprediksi Keberhasilan Immunotheraphy," *J. Teknol. dan Manaj. Inform.*, vol. 7, no. 1, pp. 46–50, 2021, doi: 10.26905/jtmi.v7i1.5982.

[12] A. Arimuko, A. S. W. Wibawa, and A. Firmansyah, "Analisis Perbandingan Penentuan Hiposentrum Menggunakan Metode Grid Search, Geiger, dan Random Search: Studi Kasus pada Letusan Gunung Sinabung 2017," *Diffraction*, vol. 1, no. 2, pp. 22–28, 2019, doi: 10.37058/diffraction.v1i2.1290.

[13] F. Tempola, M. Muhammad, and A. Khairan, "Perbandingan Klasifikasi Antara KNN dan Naive Bayes pada Penentuan Status Gunung Berapi dengan K-Fold Cross Validation," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 5, p. 577, 2018, doi: 10.25126/jtiik.201855983.

[14] A. Y. Permana and M. M. Effendi, "Optimasi Stemming Porter KBBI dan Cross Validation Naïve Bayes untuk Klasifikasi Topik Soal UN Bahasa Indonesia," *J. Ilm. Komputasi*, vol. 17, no. 4, 2018, doi: 10.32409/jikstik.17.4.2492.

[15] A. Z. Farmadiansyah, A. F. Hidayatullah, and F. Rahma, "Deteksi Surel Spam dan Non Spam Bahasa Indonesia," *Automata*, 2021, [Online]. Available: https://journal.uii.ac.id/AUTOMATA/article/view/19514