

**PREDIKSI *TURNOVER* KARYAWAN MENGGUNAKAN  
ALGORITMA *RANDOM FOREST***

**SKRIPSI**

Oleh :

**RIZKA MA'RIFATUL KHASANAH**  
**NIM. 200605110032**



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2024**

**PREDIKSI *TURNOVER* KARYAWAN MENGGUNAKAN  
ALGORITMA *RANDOM FOREST***

**SKRIPSI**

Diajukan kepada:  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Untuk memenuhi Salah Satu Persyaratan dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :  
**RIZKA MA'RIFATUL KHASANAH**  
**NIM. 200605110032**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2024**

**HALAMAN PERSETUJUAN**

**PREDIKSI *TURNOVER* KARYAWAN MENGGUNAKAN  
ALGORITMA *RANDOM FOREST***

**SKRIPSI**

**Oleh :**  
**RIZKA MA'RIFATUL KHASANAH**  
**NIM. 200605110032**

Telah Diperiksa dan Disetujui untuk Diuji:  
Tanggal: 23 September 2024

Pembimbing I,



Dr. M. Ainul Yaqin, M.Kom  
NIP. 19761013 200604 1 004


Pembimbing II,



Syahiduz Zaman, M.Kom  
NIP. 19700502 200501 1 005

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



  
Fauzan Kurniawan, M.MT, IPU  
NIP. 19771020 200912 1 001

## HALAMAN PENGESAHAN

### PREDIKSI *TURNOVER* KARYAWAN MENGGUNAKAN ALGORITMA *RANDOM FOREST*

#### SKRIPSI

Oleh :  
**RIZKA MA'RIFATUL KHASANAH**  
NIM. 200605110032

Telah Dipertahankan di Depan Dewan Penguji Skripsi  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer ( S.Kom )  
Tanggal: 02 Oktober 2024

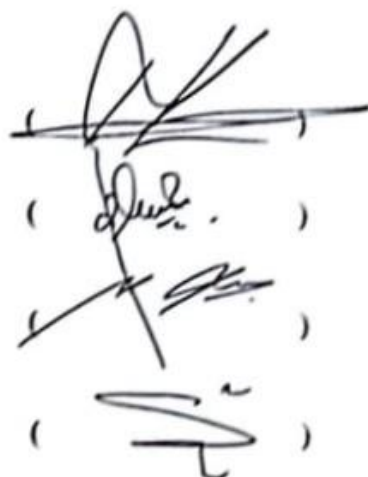
#### Susunan Dewan Penguji





Ketua Penguji : Supriyono, M. Kom  
NIP. 19841010 201903 1 012

Anggota Penguji I : Roro Inda Melani, M.T, M.Sc  
NIP. 19780925 200501 2 008

Anggota Penguji II : Dr. M. Ainul Yaqin, M.Kom  
NIP. 19761013 200604 1 004

Anggota Penguji III : Syahiduz Zaman, M.Kom  
NIP. 19700502 200501 1 005



(  )  
(  )  
(  )  
(  )

Mengetahui dan Mengesahkan,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Negeri Maulana Malik Ibrahim Malang



  
Abdul Kurniawan, M.MT, IPU  
NIP. 19771020 200912 1 001

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Rizka Ma'rifatul Khasanah  
NIM : 200605110032  
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika  
Judul Skripsi : Prediksi *Turnover* Karyawan menggunakan  
Algoritma *Random Forest*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 02 Oktober 2024  
Yang membuat pernyataan,



Rizka Ma'rifatul Khasanah  
NIM.200605110032

## MOTTO

*“Setiap kali ingin menyerah, coba ingat lagi sudah sejauh apa kita melangkah”*

*(Fiersa Besari)*

*“Do’amu pasti dikabulkan. Utuh, ditukar, atau jadi kekuatan”*

*... وَمَا كُنْتُ بِدُعَائِكَ رَبِّ شَقِيًّا*

*“...dan aku belum pernah kecewa dalam berdoa kepada-Mu, ya Tuhanku.”*

*(QS Maryam : 4)*

## HALAMAN PERSEMBAHAN

Dengan penuh rasa syukur kepada Allah SWT atas limpahan rahmat dan kemudahan-Nya, akhirnya skripsi ini dapat terselesaikan dengan baik.

Karya ini penulis persembahkan kepada:

Ibu tercinta, Rusmiati

Yang selalu mengalirkan kasih sayang, usaha terbaik, do'a-do'a tulus, dukungan, dan nasehat yang tiada henti.

Ayah tercinta, Arif Tarsito

Yang senantiasa memberikan kekuatan, usaha terbaik, serta dukungan moral maupun materi.

Adik tercinta, Fida Hanifatun Khasanah

Yang menjadi salah satu motivasi dan dorongan untuk terus maju hingga skripsi ini terselesaikan.

Segenap keluarga besar,

Yang selalu mengiringi perjalanan penulis dengan do'a dan dukungan.

*Last but not least*, teruntuk diri sendiri

Terima kasih untuk tidak menyerah, terima kasih sudah bertahan sejauh ini.

## KATA PENGANTAR

*Bismillahirrahmaanirrahiim*

*Alhamdulillah* segala puji dan Syukur senantiasa penulis panjatkan pada Allah subhanahu wa ta'ala atas berkat Rahmat, serta hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Prediksi *Turnover* Karyawan menggunakan Algoritma *Random Forest*”. Sholawat serta salam tetap tercurahkan kepada Nabi Muhammad SAW. Dan semoga kita semua mendapat syafaatnya di hari akhir kelak, Aamiin.

Penulis mengucapkan rasa terima kasih yang begitu besar kepada seluruh pihak yang memberikan dukungan dan motivasi kepada penulis sehingga dapat menyelesaikan skripsi ini. Ucapan terima kasih ini penulis disampaikan kepada:

1. Prof. Dr. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Harini, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, M.MT, IPU., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Dr. Ainul Yaqin, M. Kom., selaku pembimbing utama yang dengan penuh kesabaran dan ketulusan hati memberikan bimbingan, arahan, serta dorongan dalam setiap tahap penyusunan skripsi ini.

5. Syahiduz Zaman, M. Kom., selaku dosen wali sekaligus pembimbing kedua penulis yang selalu memberikan masukan selama perkuliahan hingga terselesaikannya skripsi ini.
6. Supriyono, M. Kom., selaku penguji utama dan Roro Inda Melani, M, T., M. Sc., selaku penguji kedua yang telah berkenan menguji serta memberikan masukan yang membangun sehingga skripsi ini dapat terselesaikan dengan baik.
7. Nia Faricha S, Si., selaku admin Program Studi Teknik Informatika yang selalu sabar memberikan informasi, membantu, dan memberikan arahan selama perkuliahan dan proses penulisan skripsi ini.
8. Segenap dosen, laboran, dan jajaran staff Program Studi Teknik Informatika yang telah memberikan ilmu, pengetahuan, dan dukungan selama penulis menjalani studi hingga selesainya skripsi ini.
9. Kedua orang tua tercinta, Ibu Rusmiati dan Ayah Arif Tarsito yang selalu menjadi sumber kekuatan bagi penulis. Terima kasih karena selalu mengusahakan yang terbaik. Semoga Allah senantiasa memberikan kesehatan dan lindungan, sehingga dapat selalu berada disetiap perjalanan dan pencapaian penulis.
10. Adik tersayang Fida Hanifatun Khasanah beserta seluruh keluarga besar yang tiada henti memberikan do'a dan dukungan sehingga penulis mampu menyelesaikan skripsi ini.

11. Sahabat seperjuangan "*enter new subject*" yang beranggotakan Vera, Bima, dan Zidan yang sudah setia menemani, memberikan semangat serta segala bantuan dari awal perkuliahan hingga saat ini.
12. Teman dekat penulis, Nurussakinah yang sudah kebersamai dan membantu penulis hingga saat ini. Tolong hidup lebih lama, masih banyak tempat yang belum kita kunjungi bersama.
13. Teman-teman penghuni "Asrama Pak Alex", Sasa, Vera, Indah, Lala, dan Widia yang telah setia menjadi *supporter* dan teman "menggila" sehingga penulis tidak pernah kesepian dan merasa sendirian. Mari tetap bertahan untuk waktu yang lama.
14. Teman-teman penulis, Zulfiyatun Muzayyanah, Rosa Maulida, Adelia Putri, Laudia Saronyx, Nadia Muizza, Mila Amarila, Kartika Wulandari, Faqiatun Khasanah, Widi Astuti, yang sudah berkenan menjadi teman baik, teman berkeluh kesah, dan selalu sedia saat penulis membutuhkan bantuan.
15. Seluruh warga Teknik Informatika khususnya angkatan 2020 "Integer" yang telah memberikan kehangatan, motivasi, dan dukungan kepada penulis.
16. Teman-teman Himatif Encoder, Divisi Religius '21 '22, dan teman-teman GenBI '23 yang telah menjadi tempat penulis untuk mengembangkan *hardskill* maupun *softskill* selama masa perkuliahan.

17. Bapak Machsun Zain, Pak Hafidz, Bu Aida, dan seluruh keluarga Kantor Kementerian Agama Kota Batu yang sudah berkenan berbagi ilmu dan pengalaman selama PKL.
18. Teman-teman KKM 174 Desa Pagedangan Kecamatan Turen tahun 2022, yang telah memberikan pengalaman dan kenangan yang tak terlupakan di masa perkuliahan.
19. Seluruh pihak yang telah terlibat, baik secara langsung maupun tidak langsung dari awal perkuliahan hingga akhir penulisan skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih jauh dari kata sempurna. Maka dari itu penulis menerima saran, kritik dan masukan yang bersifat membangun sehingga skripsi ini dapat lebih dikembangkan. Penulis berharap semoga skripsi ini dapat memberikan manfaat untuk kedepannya.

*Wassalamu'alaikum warahmatullahi wabarakatuh.*

Malang, 23 September 2024

Penulis

## DAFTAR ISI

<b>HALAMAN PENGAJUAN</b> .....	<b>ii</b>
<b>HALAMAN PERSETUJUAN</b> .....	<b>iii</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>iv</b>
<b>PERNYATAAN KEASLIAN TULISAN</b> .....	<b>v</b>
<b>MOTTO</b> .....	<b>vi</b>
<b>HALAMAN PERSEMBAHAN</b> .....	<b>vii</b>
<b>KATA PENGANTAR</b> .....	<b>viii</b>
<b>DAFTAR ISI</b> .....	<b>xii</b>
<b>DAFTAR GAMBAR</b> .....	<b>xiv</b>
<b>DAFTAR TABEL</b> .....	<b>xv</b>
<b>ABSTRAK</b> .....	<b>xvi</b>
<b>ABSTRACT</b> .....	<b>xvii</b>
<b>البحث مستخلص</b> .....	<b>xviii</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Batasan Masalah .....	5
1.4 Tujuan Penelitian .....	5
1.5 Manfaat Penelitian .....	5
<b>BAB II STUDI PUSTAKA</b> .....	<b>6</b>
2.1 Penelitian Terkait .....	6
2.2 <i>Turnover</i> .....	11
2.3 Prediksi .....	12
2.4 <i>Random Forest</i> .....	12
2.5 <i>Decision Tree</i> .....	14
2.6 <i>Pearson Correlation Coefficient</i> .....	15
2.7 <i>Confusion Matrix</i> .....	15
<b>BAB III DESAIN DAN IMPLEMENTASI</b> .....	<b>18</b>
3.1 Tahapan Penelitian .....	18
3.2 <i>Data Collecting</i> .....	19
3.3 Preprocessing .....	24
3.3.1 <i>Data Encoding</i> .....	24
3.3.2 <i>Data Scaling</i> .....	27
3.3.3 <i>Features Selection</i> .....	29
3.3.4 <i>Split Data</i> .....	31
3.4 Algoritma <i>Random Forest</i> .....	32
3.4.1 Penentuan <i>Hyperparameter Random Forest</i> .....	33
3.4.2 Pembentukan Model <i>Random Forest</i> .....	33
3.5 Evaluasi .....	33
3.6 Eksperimen .....	34
3.7 Skenario Uji Coba .....	39
<b>BAB IV HASIL DAN PEMBAHASAN</b> .....	<b>40</b>
4.1 Hasil Uji Coba .....	40

4.1.1 Uji Coba Skenario A.....	40
4.1.2 Uji Coba Skenario B .....	46
4.1.3 Uji Coba Skenario C .....	52
4.1.4 Uji Coba Skenario D.....	58
4.2 Pembahasan.....	67
4.1.1 Akurasi.....	68
4.1.2 Presisi.....	69
4.1.3 <i>Recall`</i> .....	71
4.1.4 <i>F1-score</i> .....	72
4.3 Integrasi Islam.....	75
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>79</b>
5.1 Kesimpulan .....	79
5.2 Saran .....	81
<b>DAFTAR PUSTAKA</b>	

## DAFTAR GAMBAR

Gambar 2.1 <i>Random Forest</i> .....	13
Gambar 3.1 Tahapan penelitian .....	18
Gambar 3.2 Diagram proses <i>Random Forest</i> .....	32
Gambar 3.3 Pembentukan <i>decision tree bootstrap</i> 1 .....	37
Gambar 3.4 Pembentukan <i>decision tree bootstrap</i> 2 dan 3 .....	37
Gambar 4.1 Hasil pengujian skenario A 90:10 .....	41
Gambar 4.2 <i>Confusion Matrix</i> kombinasi terbaik A 90:10.....	42
Gambar 4.3 Hasil pengujian skenario A 80:20 .....	43
Gambar 4.4 <i>Confusion Matrix</i> kombinasi terbaik A 80:20.....	44
Gambar 4.5 Hasil pengujian skenario A 70:30 .....	45
Gambar 4.6 <i>Confusion Matrix</i> kombinasi terbaik A 70:30.....	46
Gambar 4.7 Hasil pengujian skenario A 90:10 .....	47
Gambar 4.8 <i>Confusion Matrix</i> kombinasi terbaik B 90:10.....	48
Gambar 4.9 Hasil pengujian skenario B 80:20 .....	49
Gambar 4.10 <i>Confusion Matrix</i> kombinasi terbaik 80:20.....	50
Gambar 4.11 Hasil pengujian skenario B 70:30 .....	51
Gambar 4.12 <i>Confusion Matrix</i> kombinasi terbaik B 70:30.....	52
Gambar 4.13 Hasil pengujian skenario C 90:10 .....	53
Gambar 4.14 <i>Confusion Matrix</i> kombinasi terbaik C 90:10.....	54
Gambar 4.15 Hasil pengujian skenario C 80:20 .....	55
Gambar 4.16 <i>Confusion Matrix</i> kombinasi terbaik C 80:20.....	56
Gambar 4.17 Hasil pengujian skenario C 70:30 .....	57
Gambar 4.18 <i>Confusion Matrix</i> kombinasi terbaik C 70:30.....	58
Gambar 4.19 Hasil pengujian skenario D rasio 90:10 .....	59
Gambar 4.20 <i>Confusion Matrix</i> kombinasi terbaik D 90:10.....	60
Gambar 4.21 Hasil pengujian skenario D 80:20 .....	61
Gambar 4.22 <i>Confusion Matrix</i> kombinasi terbaik D 80:20.....	62
Gambar 4.23 Hasil pengujian skenario D 70:30 .....	63
Gambar 4.24 <i>Confusion Matrix</i> kombinasi terbaik D 70:30.....	64
Gambar 4.25 Grafik akurasi tiap skenario .....	69
Gambar 4.26 Grafik presisi tiap skenario .....	70
Gambar 4.27 Grafik <i>recall</i> tiap skenario.....	71
Gambar 4.28 Grafik <i>f1-score</i> tiap skenario.....	72

## DAFTAR TABEL

Tabel 2.1 Penelitian terkait .....	8
Tabel 2. 2 Confusion Matrix .....	16
Tabel 3.1 Deskripsi data.....	20
Tabel 3.2 Contoh dataset.....	22
Tabel 3.3 Contoh data sebelum <i>encoding</i> .....	25
Tabel 3.4 Contoh dataset setelah <i>encoding</i> .....	26
Tabel 3.5 Contoh dataset sebelum <i>scaling</i> .....	28
Tabel 3.6 Contoh dataset setelah <i>scaling</i> .....	28
Tabel 3. 7 Hasil Seleksi Fitur .....	30
Tabel 3. 8 Sampel dataset perhitungan manual.....	35
Tabel 3.9 Dataset setelah dilakukan <i>bootstrap</i> .....	36
Tabel 3.10 Skenario uji coba.....	39
Tabel 3.11 Nilai <i>hyperparameter</i> .....	39
Tabel 4.1 Hasil kombinasi hyperparameter terbaik tiap skenario.....	65
Tabel 4. 2 Hasil uji coba seluruh fitur tanpa PCC.....	74
Tabel 4. 3 Hasil uji coba seluruh fitur dengan PCC.....	75

## ABSTRAK

Khasanah, Rizka Ma'rifatul. 2024. **Prediksi *Turnover* Karyawan Menggunakan Algoritma *Random Forest***. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. M. Ainul Yaqin, M.Kom (II) Syahiduz Zaman, M. Kom.

**Kata Kunci:** *Prediksi, Turnover karyawan, Random Forest, Pearson Correlation Coefficient*

*Turnover* karyawan merupakan tantangan yang dapat memengaruhi stabilitas dan kinerja perusahaan. Penelitian ini bertujuan untuk memprediksi *turnover* karyawan menggunakan algoritma *Random Forest* dan seleksi fitur berbasis *Pearson Correlation Coefficient* (PCC). Dataset yang digunakan adalah data *Human Resources* (HR) dari *International Business Machines Corporation* (IBM), terdiri dari 1470 data dan 35 fitur. Pengujian dilakukan dengan berbagai skenario, termasuk penggunaan seluruh fitur serta 5, 10, dan 15 fitur teratas berdasarkan PCC, dengan rasio pembagian data 90:10, 80:20, dan 70:30, serta penyesuaian *hyperparameter*. Hasil penelitian menunjukkan bahwa model *Random Forest* mencapai akurasi tertinggi sebesar 88% pada skenario penggunaan seluruh fitur dengan rasio data 80:20, menggunakan kombinasi *hyperparameter* terbaik yaitu  $n\_estimators = 100$ ,  $max\_depth = 5$ , dan  $max\_features = 2$ . Penggunaan data dengan seleksi fitur menghasilkan penurunan akurasi hingga 2%. Meskipun PCC membantu perusahaan dalam mengidentifikasi faktor-faktor relevan yang memengaruhi *turnover*, uji coba menunjukkan bahwa tanpa atau dengan PCC hasilnya tetap konsisten ketika seluruh fitur digunakan karena model mampu memanfaatkan seluruh informasi yang terdapat dalam data. Penelitian ini diharapkan dapat membantu perusahaan dalam merancang strategi retensi untuk menekan angka *turnover* berbasis data prediktif.

## ABSTRACT

Khasanah, Rizka Ma'rifatul. 2024. **Prediksi Turnover Karyawan Menggunakan Algoritma *Random Forest***. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. M. Ainul Yaqin, M.Kom (II) Syahiduz Zaman, M. Kom.

Employee turnover is a challenge that can affect the stability and performance of the company. This study aims to predict employee turnover using the Random Forest algorithm and Pearson Correlation Coefficient (PCC)-based feature selection. The dataset used is Human Resources (HR) data from International Business Machines Corporation (IBM), consisting of 1470 data and 35 features. Testing was conducted with a variety of scenarios, including the use of the entire feature as well as the top 5, 10, and 15 features based on PCC, with data sharing ratios of 90:10, 80:20, and 70:30, as well as hyperparameter adjustments. The results show that the Random Forest model achieves the highest accuracy of 88% in the scenario of using all features with a data ratio of 80:20, using the best combination of hyperparameters namely  $n\_estimators = 100$ ,  $max\_depth = 5$ , and  $max\_features = 2$ . The use of data with feature selection results in a decrease in accuracy of up to 2%. Although PCC helps companies in identifying relevant factors that affect turnover, trials show that without or with PCC, the results remain consistent when all features are used because the model is able to utilize all the information contained in the data. This research is expected to help companies in designing retention strategies to reduce turnover numbers based on predictive data.

**Key words:** Prediction, Employee *Turnover*, *Random Forest*, Pearson Correlation Coefficient

## البحث مستخلص

حسنة، رزكا معرفة. 2024. التنبؤ بمعدل دوران الموظفين باستخدام خوارزمية الغابة العشوائية. أطروحة برنامج دراسة  
الهندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية، مالانج. المشرف  
أنا ( د.م.عين اليقين، م.كوم)الثاني (شهيدوز زمان، م.كوم)

الكلمات المفتاحية: التنبؤ، دوران الموظفين، الغابة العشوائية، معامل ارتباط بيرسون

دوران الموظفين هو التحدي الذي يمكن أن يؤثر على استقرار وأداء الشركة. تحدف هذه الدراسة إلى التنبؤ بدوران  
مجموعة البيانات (PCC) واختيار الميزة المستندة إلى معامل ارتباط بيرسون Random Forest الموظفين باستخدام خوارزمية  
International Business Machines Corporation (IBM) من شركة (HR) المستخدمة هي بيانات الموارد البشرية  
وتتكون من 1470 بيانات و 35 ميزة. تم إجراء الاختبار مع مجموعة متنوعة من السيناريوهات ، بما في ذلك استخدام الميزة ،  
مع نسب مشاركة البيانات 90:10 و 80:20 و 70:30 ، PCC بأكملها بالإضافة إلى أفضل 5 و 10 و 15 ميزة بناء على  
يحقق أعلى دقة بنسبة 88% في سيناريو Random Forest بالإضافة إلى تعديلات الملعلمات الفائقة. أظهرت النتائج أن نموذج  
و  $n\_estimators = 100$  استخدام جميع المعالم بنسبة بيانات 80:20 ، باستخدام أفضل مزيج من الملعلمات الفائقة وهي  
يؤدي استخدام البيانات مع تحديد الميزات إلى انخفاض في الدقة يصل إلى  $max\_features = 2$  و  $max\_depth = 5$   
تساعد الشركات في تحديد العوامل ذات الصلة التي تؤثر على معدل الدوران ، إلا أن التجارب تظهر PCC على الرغم من أن 2%  
تظل النتائج متسقة عند استخدام جميع الميزات لأن النموذج قادر على استخدام جميع المعلومات الواردة في ، PCC أنه بدون أو مع  
البيانات. من المتوقع أن يساعد هذا البحث الشركات في تصميم استراتيجيات الاحتفاظ لتقليل أرقام الدوران بناء على البيانات التنبؤية

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Dalam dunia bisnis yang semakin modern dan persaingan yang semakin ketat, manajemen sumber daya manusia merupakan salah satu faktor terpenting dalam mencapai kesuksesan bagi setiap perusahaan. Di tengah dinamika pasar kerja yang terus berubah, pengelolaan sumber daya yang efektif terhadap tenaga kerja tidak hanya berkontribusi pada produktivitas dan profitabilitas tetapi juga berperan penting dalam mempertahankan tenaga kerja yang berkualitas. Masalah tingkat *turnover* yang tinggi merupakan tantangan signifikan yang dihadapi oleh banyak perusahaan, dan memberikan dampak negatif pada stabilitas dan kinerja perusahaan (Budun *et al.*, 2021).

*Turnover* karyawan, diartikan sebagai perputaran karyawan yang meninggalkan suatu perusahaan tempatnya bekerja dan akan digantikan oleh karyawan baru, sehingga dapat memengaruhi produktivitas dan profitabilitas perusahaan (Krisna, 2022). Data terbaru menunjukkan bahwa tingkat *turnover* global mengalami lonjakan signifikan dari tahun 2020 hingga 2022, dengan penurunan di tahun 2023. Meskipun terjadi penurunan dari tahun-tahun sebelumnya (2020 – 2022), tingkat *turnover* secara global masih terhitung tinggi karena mencapai rata-rata sebesar 47% dan lebih dari 44 juta orang mengundurkan diri secara sukarela sepanjang tahun 2023 (Hansen, 2024). Angka-angka tersebut menunjukkan bahwa tantangan dalam mengelola *turnover* karyawan memerlukan

perhatian serius bagi perusahaan untuk menjaga stabilitas dan efektivitas perusahaan.

*Turnover* karyawan dipengaruhi oleh berbagai faktor seperti usia, masa kerja, beban kerja, kondisi lingkungan, kepuasan kerja, dan kompensasi yang diterima (Yuliani & Abdi, 2023). Tingkat *turnover* yang tinggi berpotensi menimbulkan kerugian signifikan bagi perusahaan, terutama jika karyawan yang meninggalkan perusahaan merupakan individu dengan kinerja unggul, keahlian khusus, dan pengalaman yang mendalam. Dampak *turnover* tidak hanya dirasakan oleh perusahaan, tetapi juga oleh karyawan yang tersisa. Fenomena ini dapat memicu efek domino, di mana karyawan lain mungkin terdorong untuk keluar, atau mereka harus menanggung beban kerja tambahan akibat kekosongan posisi. Dalam upaya mengisi posisi yang kosong, perusahaan juga harus mengalokasikan sumber daya tambahan untuk proses rekrutmen dan pelatihan karyawan baru, yang dapat meningkatkan biaya operasional perusahaan. Sehingga perusahaan membutuhkan biaya lebih untuk proses seleksi sekaligus pelatihan karyawan baru (D. Sari & Susanto, 2019).

Mengacu pada pentingnya peran karyawan dalam kelangsungan operasional dan kesuksesan perusahaan, strategi peningkatan retensi menjadi krusial bagi perusahaan untuk mempertahankan karyawan yang unggul. Upaya retensi yang efektif dapat dilakukan melalui berbagai inisiatif seperti program pengembangan kompetensi, penawaran kompensasi dan tunjangan yang kompetitif, serta pemberian apresiasi terhadap kontribusi karyawan. Selain itu, optimalisasi kinerja perusahaan secara keseluruhan juga menjadi faktor penting dalam menciptakan

lingkungan kerja yang mendukung dan memotivasi karyawan untuk tetap berkomitmen pada perusahaan (Iskandar & Rahadi, 2021).

Prediksi *turnover* karyawan dapat dilakukan dengan menggunakan teknik-teknik dalam data mining, yang memungkinkan ekstraksi informasi dan analisis pola dari data berukuran besar melalui pendekatan statistik, matematika, dan *machine learning* (Sekar Setyaningtyas *et al.*, 2022). Data mining menjadi alat yang sangat berguna untuk memprediksi tren, pola, atau perilaku di masa depan berdasarkan data historis yang tersedia, dan telah diterapkan di berbagai bidang seperti bisnis, ilmu pengetahuan, kesehatan, dan lain sebagainya.

Salah satu metode yang sering digunakan dalam prediksi adalah *Random Forest*, sebuah algoritma yang populer karena beberapa kelebihan yang dimilikinya. *Random Forest* mampu menangani *missing value*, *outlier*, serta dapat memproses data dalam jumlah besar dengan efisien. Selain itu, metode ini juga dilengkapi dengan mekanisme seleksi fitur yang dapat meningkatkan performa model prediksi secara keseluruhan (Amna, *et al.*, 2023). Penggunaan *Random Forest* dalam melakukan prediksi *turnover* karyawan memungkinkan perusahaan untuk mendapatkan hasil yang lebih akurat dan andal, sehingga strategi retensi yang dirumuskan dapat lebih tepat sasaran.

Dari perspektif hukum Islam, prediksi *turnover* karyawan perlu dilakukan dengan memperhatikan prinsip keadilan, keseimbangan, dan kesejahteraan. Ajaran Islam menekankan pentingnya perusahaan untuk bersikap adil dan menghormati hak-hak karyawan, ini sejalan dengan firman Allah pada surah An-Najm (53): 41

ثُمَّ يُجْزَاهُ الْجِزَاءَ الْأَوْفَى

“Kemudian akan diberi balasan kepadanya dengan balasan yang paling sempurna” (QS:An-Najm: 41)

Ayat dalam QS. An-Najm (53): 41 menegaskan bahwa setiap usaha manusia akan diberi balasan yang paling sempurna. Tafsir Jalalayn menjelaskan tentang janji Allah terhadap manusia jika kelak akan diberi balasan atas semua usahanya dengan balasan yang paling sempurna atau sebanding (*Surat An-Najm Ayat 41 - Qur'an Tafsir Perkata*, 2024). Prinsip ini dapat diterapkan dalam dunia kerja, di mana perusahaan harus memastikan bahwa setiap usaha karyawan dihargai secara adil dan memperhatikan hak-hak karyawan yang sesuai dengan usaha mereka.

Berdasarkan prinsip di atas, penulis mengusulkan skripsi dengan judul **"Prediksi *Turnover* Karyawan Menggunakan Algoritma *Random Forest*".** Melalui penelitian ini, diharapkan dapat memberikan kontribusi positif dalam memprediksi *turnover* karyawan serta memperkuat pemahaman tentang faktor-faktor yang memengaruhi *turnover* karyawan. Sehingga perusahaan dapat merancang strategi retensi yang tepat dan mampu mempertahankan karyawan dengan kinerja yang baik.

## 1.2 Rumusan Masalah

Berdasarkan uraian latar belakang di atas, masalah yang dapat dirumuskan adalah bagaimana penggunaan data historis dan algoritma *machine learning* untuk melakukan prediksi *turnover* secara akurat, sehingga membantu perusahaan untuk membuat keputusan dalam merancang strategi retensi yang tepat?

### 1.3 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah:

1. Data yang digunakan adalah data penelitian IBM (*International Business Machines Corporation*) tentang HR (*Human Resources*) yang didapat dari website Kaggle.com yang memiliki 35 atribut serta 1470 *record* data.
2. Fokus analisis prediksi hanya berdasar variabel-variabel yang tersedia dalam dataset IBM HR.

### 1.4 Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah memanfaatkan data historis dan mengimplementasikan algoritma *machine learning* untuk membuat model prediksi *turnover* karyawan, sehingga dapat membantu merancang strategi retensi karyawan yang tepat.

### 1.5 Manfaat Penelitian

Hasil dari penelitian ini diharapkan mampu memberi manfaat secara teoritis dan praktis sebagai berikut:

1. Secara teoritis, penelitian ini diharapkan dapat memperkaya literatur dalam bidang data mining dan *machine learning* khususnya penerapan algoritma *Random Forest* untuk melakukan prediksi.
2. Secara praktis, penelitian ini diharapkan dapat membantu perusahaan dalam mengidentifikasi faktor-faktor penyebab *turnover* sehingga dapat merancang strategi retensi yang tepat.

## BAB II

### STUDI PUSTAKA

#### 2.1 Penelitian Terkait

Penelitian dengan topik terkait dilakukan oleh (S. F. Sari & Lhaksana, 2022) berjudul “*Employee Attrition Prediction Using Feature Selection with Information Gain and Random Forest Classification*”. Penelitian tersebut mengimplementasikan metode *Random Forest* dengan membandingkan metode seleksi fitur *Information Gain*, *Select K Best*, dan *Recursive Feature Elimination* untuk mencari fitur mana yang menghasilkan performa terbaik. Berdasarkan skenario pengujian pada penelitian tersebut, *Information Gain* memperoleh akurasi tertinggi sebesar 89,2% ketika menggunakan 25 fitur, *Select K Best* memperoleh akurasi tertinggi sebesar 87,8% ketika menggunakan 20 fitur, dan *Recursive Feature Elimination* memperoleh akurasi tertinggi sebesar 88,8% ketika menggunakan 25 fitur.

Penelitian lain berjudul “Prediksi Pengunduran Diri Karyawan Perusahaan “Y” Menggunakan *Random Forest*” oleh (Manurung *et al.*, 2021) mengimplementasikan metode *Random Forest* dengan melakukan seleksi fitur menggunakan metode *Principal Component Analysis* (PCA) dengan menggunakan dataset yang memiliki 35 atribut dan 3310 *record* data. PCA berfungsi untuk mengelompokkan data berdasar pengamatan antar variabel yang berkorelasi serta membantu mengurangi dimensi untuk meringkas variabel numerik.. Hasil

penelitian menunjukkan bahwa metode *Random Forest* dapat memprediksi pengunduran diri karyawan dengan akurasi sebesar 87% dan error sebesar 13%.

Penelitian selanjutnya dilakukan oleh (Abiyyu & Lhaksana, 2021) dengan judul “Perbandingan Metode Seleksi Fitur untuk Mengoptimasi Model *Support Vector Machine* dalam Memprediksi *Turnover* Pegawai”. Seleksi fitur pada metode *Support Vector Machine* (SVM) dilakukan untuk mengurangi dimensi data dengan tujuan meningkatkan performa model algoritma *machine learning*. Seleksi fitur yang dilakukan berupa *filter methods*, *wrapper methods*, dan *embedded method*. Pengujian dilakukan dengan kategori dataset tanpa seleksi fitur dan pengujian dataset setelah dilakukan proses seleksi fitur. Berdasar hasil pengujian prediksi karyawan menggunakan metode SVM tanpa seleksi fitur, diperoleh nilai sebesar 0,56% untuk semua metode pengukuran yang meliputi akurasi, presisi, *recall*, dan *f1-score*. Selanjutnya pengujian yang dilakukan terhadap dataset setelah dilakukan seleksi fitur menghasilkan akurasi yang meningkat sekaligus menurun. Metode seleksi fitur yang menaikkan performa adalah wrapper method dengan nilai performa 0,60. Sedangkan dua metode lain yaitu *filter method* dan *embedded method* mengalami penurunan performa menjadi 0,55.

Penelitian serupa dilakukan oleh (Aryanto *et al.*, 2022) dengan judul “Prediksi Peringkat Mingguan Lagu Pada Spotify Amerika Serikat Menggunakan *Multiple Charts* Dataset Dengan Berbagai Metode”. Penelitian tersebut menggunakan metode *Multipler Linear Regression*, *Polynomial Regression*, *Gradient Boosting*, dan *Random Forest* untuk pembuatan modelnya dan

dibandingkan performanya menggunakan *Adjusted R-squad* dan *Mean Absolute Error* (MAE). Hasil penelitian menunjukkan bahwa metode yang paling baik untuk melakukan prediksi adalah *Random Forest* dengan *splitting rasio* terbaik adalah 8:2.

Selanjutnya (Syamsiah & Purwandani, 2021) melakukan penelitian berjudul “Penerapan *Ensemble Stacking* untuk Peramalan Laba Bersih Bank Syariah Indonesia (BSI)”. Penelitian tersebut menggunakan beberapa metode tunggal dan menerapkan metode *ensemble stacking* untuk menguji seberapa akurat metode-metode yang digunakan. Penelitian tersebut terdiri dari dua level, yang mekanisme kerjanya menggabungkan hasil prediksi beberapa metode di level bawah untuk selanjutnya digunakan oleh algoritma di level atas untuk mencapai hasil prediksi baru yang biasanya akan lebih akurat. Penelitian ini menggunakan tiga metode di level bawah, yaitu *Support Vector Machine*, *Random Forest* dan *Neural Networks*. Selanjutnya di lapisan atas menggunakan algoritma *Generalized Linier Model*. Peramalan yang menggunakan *ensemble stacking* terbukti lebih unggul dari hasil metode lainnya. Berdasarkan penelitian juga didapatkan bahwa meskipun menggunakan metode tunggal, *Random Forest* mampu menghasilkan akurasi yang lebih baik daripada metode tunggal lainnya.

Tabel 2.1 Penelitian terkait

No.	Nama Peneliti	Judul	Metode yang digunakan	Perbedaan Penelitian
1.	(Sari & Lhaksana, 2022)	Employee Attrition Prediction Using Feature Selection with Information Gain and <i>Random Forest</i> Classification	<i>Random Forest</i> dengan Seleksi Fitur	<ul style="list-style-type: none"> <li>- Preprocessing dengan <i>encoding</i>, <i>scaling</i>, dan <i>feature selection</i>.</li> <li>- <i>Feature selection</i> menggunakan metode <i>Pearson</i></li> </ul>

No.	Nama Peneliti	Judul	Metode yang digunakan	Perbedaan Penelitian
				<i>Correlation Coefficient.</i>
2.	(Manurung dkk., 2021)	Prediksi Pengunduran Diri Karyawan Perusahaan “Y” Menggunakan <i>Random Forest</i>	<i>Random Forest</i> dan seleksi fitur menggunakan metode <i>Principal Component Analysis</i> (PCA)	<ul style="list-style-type: none"> <li>- Data penelitian menggunakan dataset “<i>IBM HR Analytics</i>” dengan 35 atribut dan 1470 data.</li> <li>- Seleksi fitur dilakukan menggunakan metode <i>Pearson Correlation Coefficient.</i></li> </ul>
3.	(Abiyu & Lhaksana, 2023)	Perbandingan Metode Seleksi Fitur untuk Mengoptimasi Model <i>Support Vector Machine</i> dalam Memprediksi <i>Turnover</i> Pegawai	Seleksi Fitur <i>Support Vector Machine</i>	<ul style="list-style-type: none"> <li>- Menggunakan metode <i>Random Forest.</i></li> <li>- Seleksi fitur dilakukan menggunakan metode <i>Pearson Correlation Coefficient.</i></li> </ul>
4.	(Aryanto <i>et al.</i> , 2022)	Prediksi Peringkat Mingguan Lagu Pada Spotify Amerika Serikat Menggunakan <i>Multiple Charts</i> Dataset Dengan Berbagai Metode	Metode <i>multipler linear regression, polynomial regression, gradient boosting tree,</i> dan <i>random forest</i>	<ul style="list-style-type: none"> <li>- Data yang digunakan adalah dataset “<i>IBM HR Analytics</i>” tentang pengunduran diri karyawan.</li> <li>- Seleksi fitur dilakukan menggunakan metode <i>Pearson Correlation Coefficient.</i></li> </ul>
5.	(Syamsiah & Purwandani, 2021)	Penerapan <i>Ensemble Stacking</i> untuk Peramalan Laba Bersih Bank Syariah Indonesia (BSI)	<i>Support Vector Machine, Neural Networks, Random Forest,</i> dan <i>Ensemble Stacking</i>	<ul style="list-style-type: none"> <li>- Seleksi fitur dilakukan menggunakan metode <i>Pearson Correlation Coefficient.</i></li> <li>- Data penelitian menggunakan</li> </ul>

No.	Nama Peneliti	Judul	Metode yang digunakan	Perbedaan Penelitian
				dataset “ <i>IBM HR Analytics</i> ” tentang pengunduran diri karyawan.

Berdasarkan data pada tabel 2.1 di atas, dapat disimpulkan bahwa pertama, dalam hal seleksi fitur, penelitian ini menggunakan metode *Pearson Correlation Coefficient* (PCC). Seleksi fitur dilakukan untuk mengurangi dimensi data dan metode PCC dipilih karena kemampuannya mengidentifikasi hubungan linear antara variabel-variabel dalam data, sehingga memungkinkan untuk mengidentifikasi fitur-fitur yang memiliki korelasi tinggi dengan variabel target yaitu *attrition*. Dengan mempertahankan fitur-fitur yang paling relevan, diharapkan dapat meningkatkan kualitas model dan efisien waktu saat proses komputasi.

Kedua, penelitian ini menggunakan rasio *split* data untuk data latih dan data uji yang bervariasi dengan mengeksplorasi tiga rasio *split* data yang berbeda, yaitu 90:10, 80:20, dan 70:30. Tujuannya adalah untuk mengevaluasi bagaimana perbedaan *splitting* data dapat memengaruhi akurasi dari model prediksi yang dibuat. Dengan mengeksplorasi berbagai rasio *split* data, penelitian ini berusaha untuk mengidentifikasi rasio yang paling optimal dan memahami sensitivitas model terhadap perubahan dalam volume data pelatihan dan pengujian.

Ketiga, penggunaan algoritma *Random Forest* sebagai model prediksi diharapkan mampu memberikan keuntungan tambahan. *Random Forest* adalah salah satu metode yang kuat dalam melakukan prediksi karena kemampuannya dalam menangani data yang kompleks dan beragam, serta menangani data tidak

seimbang sehingga dapat diandalkan untuk melakukan prediksi *turnover* karyawan. Dalam penggunaannya sebagai model prediksi, terdapat beberapa *hyperparameter* yang dapat disesuaikan nilainya untuk meningkatkan kinerja dan fleksibilitas model. Beberapa *hyperparameter* yang akan digunakan dalam penelitian ini adalah *n\_estimators*, *max\_depth*, dan *max\_features*.

## 2.2 *Turnover*

*Turnover* dapat diartikan sebagai keluarnya karyawan meninggalkan tempat kerjanya. *Turnover* karyawan merujuk kepada fenomena di mana karyawan meninggalkan pekerjaan mereka, baik secara sukarela maupun tidak. Tingkat *turnover* karyawan merupakan indikator penting yang perlu diperhatikan oleh setiap perusahaan. Beberapa faktor yang menjadi penyebab tingginya tingkat *turnover* karyawan meliputi kurangnya motivasi di tempat kerja, beban kerja yang berlebihan, perbedaan antara gaji dan tingkat beban kerja yang tidak seimbang, sistem kerja yang tidak efisien, kurangnya keseimbangan antara kehidupan kerja dan pribadi, kepemimpinan yang tidak kompeten, serta kurangnya *feedback* dan pengakuan terhadap kinerja karyawan (Ardi Subakti *et al.*, 2023).

Dampak *turnover* karyawan yang tinggi dapat menyebabkan peningkatan biaya operasional, penurunan produktivitas dalam tim, risiko merusak citra perusahaan di mata pelanggan atau klien, serta menurunnya kepuasan kerja karyawan yang tersisa (Iskandar & Rahadi, 2021). Salah satu strategi yang dapat dilakukan oleh perusahaan dalam menekan angka *turnover* adalah melakukan retensi karyawan. Retensi mengacu kepada bagaimana perusahaan dapat mempertahankan dan memelihara sumber daya manusia yang dimiliki agar dapat

bekerja dengan baik sehingga memberi keuntungan bagi perusahaan (Pratiwi *et al.*, 2020). Melalui analisis prediksi *turnover*, maka perusahaan dapat mengetahui faktor apa saja yang potensial menyebabkan *turnover* sehingga perusahaan dapat melakukan identifikasi area mana saja yang memerlukan perbaikan dan peningkatan efisiensi sumber daya manusia.

### **2.3 Prediksi**

Prediksi merupakan proses sistematis yang dilakukan untuk memperkirakan tentang sesuatu yang paling mungkin akan terjadi di masa depan berdasar informasi yang ada pada masa lalu dan sekarang (Nawangsih & Fauziah, 2021). Prediksi bertujuan untuk mengurangi ketidakpastian dan membantu pengambilan keputusan dengan memberikan gambaran yang lebih baik tentang apa yang mungkin terjadi di masa depan.

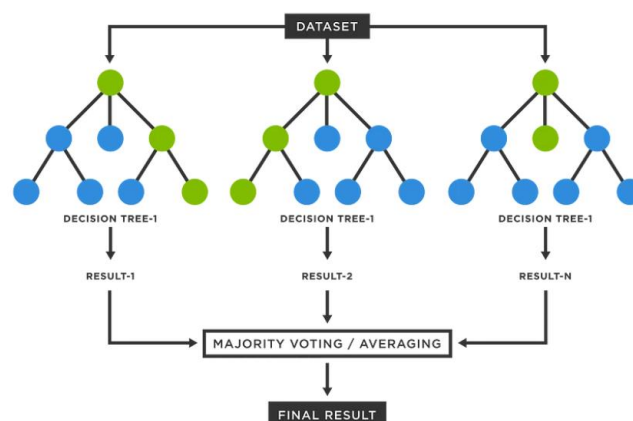
Pada data mining sendiri, prediksi merujuk pada proses analisis yang bertujuan memperkirakan nilai atau kategori variabel yang belum diketahui berdasarkan pola yang ditemukan pada data historis. Metode prediksi yang paling umum digunakan adalah regresi dan klasifikasi. Regresi digunakan untuk memprediksi nilai kontinu, sedangkan klasifikasi berfungsi untuk menentukan kategori dari data baru berdasarkan model yang telah dilatih pada data historis.

### **2.4 *Random Forest***

*Random Forest* didasarkan pada prinsip *decision tree* dan dapat digunakan untuk menyelesaikan masalah klasifikasi maupun prediksi. *Random Forest* terdiri dari kumpulan beberapa *decision tree*, di mana masing-masing pohon

menghasilkan keputusan, dan hasil akhir ditentukan berdasarkan pada *majority vote* (mayoritas keputusan tersebut). *Random Forest* membangun sejumlah besar *decision tree* secara acak dari subset data latih yang juga dipilih secara acak. Setiap pohon dalam *Random Forest* melakukan prediksi berdasarkan fitur-fitur yang ada dalam data, dan hasil prediksi tersebut bersifat independen. Prediksi dari semua pohon kemudian dikombinasikan, di mana hasil akhir diperoleh dengan mengambil modus pada masalah klasifikasi atau menghitung nilai rata-rata pada masalah regresi (Suliztia, 2020).

Teknik utama yang digunakan dalam *Random Forest* adalah *bagging*, yang berfungsi untuk membangun *ensemble decision tree* dan dapat mengurangi risiko *overfitting*, terutama saat data pelatihan berukuran kecil. *Random Forest* menawarkan berbagai keunggulan, termasuk kemampuannya untuk mencegah *overfitting*, efisiensi dalam penyimpanan data, toleransi terhadap ketidakseimbangan data, kemampuan mengatasi *missing* data, serta kemampuannya untuk meningkatkan akurasi model dan menghasilkan error yang rendah (Junus *et al.*, 2023).



Gambar 2.1 *Random Forest* (Medium.com)

Pada gambar 2.1, *Random Forest* membentuk banyak *tree* (pohon) yang menghasilkan nilai prediksi beragam. Untuk mendapatkan nilai akhir yaitu melihat mayoritas nilai yang muncul. Adapun cara kerja *Random Forest* sebagai berikut :

1. Menentukan jumlah pohon yang akan dibuat.
2. Melakukan *bagging*, yaitu membangun pohon dari setiap data sampel sebanyak jumlah yang ditentukan.
3. Membuat pohon dari kumpulan pohon-pohon. Lakukan langkah 1 dan 2 sehingga menghasilkan pohon keputusan sebanyak-banyaknya.
4. Melihat hasil prediksi dengan menggunakan *majority vote* dari hasil pohon keputusan yang merupakan hasil akhir dari *Random Forest*.

## **2.5 Decision Tree**

*Decision tree* atau juga disebut pohon keputusan adalah salah satu teknik yang dalam data mining yang paling populer untuk melakukan pengambilan keputusan. Seperti namanya, model pada *decision tree* menggunakan struktur hierarki atau struktur pohon yang konsepnya adalah mengubah data menjadi aturan keputusan serta pohon keputusan. Data dibagi menjadi himpunan bagian yang lebih kecil dan mengembangkan pohon keputusan secara bertahap. Setelah dibagi dan dikembangkan, hasil akhirnya yaitu pohon yang memiliki node keputusan dan node daun (*leaf*). Contoh node keputusan adalah cuaca yang memiliki cabang hujan, mendung, dan cerah (Amna, *et al.*, 2023).

## 2.6 *Pearson Correlation Coefficient*

*Pearson Correlation Coefficient* (PCC) adalah metode statistik yang digunakan untuk mengevaluasi hubungan linier antara dua variabel. PCC dapat digunakan dalam berbagai bidang, termasuk penelitian medis, pendidikan, dan teknologi (Iacobello *et al.*, 2021). Metode PCC mengukur sejauh mana dua variabel berkorelasi linier satu sama lain. Koefisien ini dapat memberikan pemahaman apakah suatu variabel memiliki pengaruh signifikan terhadap variabel lain dalam konteks prediksi *turnover* karyawan. Rumus dari PCC adalah sebagai berikut:

$$r = \frac{\sum(X_i - X)(Y_i - Y)}{\sqrt{\sum(X_i - X)^2 \sum(Y_i - Y)^2}} \quad (2.1)$$

Keterangan :

r	= koefisien korelasi pearson,
X <sub>i</sub>	= nilai dalam variabel X,
Y <sub>i</sub>	= nilai dalam variabel Y,
X	= rata-rata dari variabel X,
Y	= rata-rata dari variabel Y.

Rumus di atas mengukur sejauh mana dua variabel bergerak bersama-sama dalam hubungan linier. Nilai r berkisar antara -1 dan 1. Nilai r yang lebih mendekati 1 atau -1 menunjukkan korelasi yang lebih kuat, sedangkan nilai yang lebih mendekati 0 menunjukkan korelasi yang lebih lemah atau tidak ada korelasi (Romadloni & Hilman F Pardede, 2019).

## 2.7 *Confusion Matrix*

*Confusion Matrix* adalah salah satu teknik evaluasi yang digunakan untuk mengevaluasi kinerja model yang menyajikan tabel perbandingan antara nilai aktual dan nilai prediksi (Normawati & Prayogi, 2021). *Confusion Matrix*

membantu memahami seberapa baik model dalam melakukan prediksi. Tabel *Confusion Matrix* adalah sebagai berikut.

Tabel 2. 2 *Confusion Matrix*

ACTUAL	PREDICTED	
	<i>Positive</i>	<i>Negative</i>
<i>Positive</i>	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>
<i>Negative</i>	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>

Terdapat empat kategori dalam *Confusion Matrix*, yaitu:

1. *True Positive (TP)*: Jumlah kasus positif yang diprediksi benar sebagai kelas positif.
2. *True Negative (TN)*: Jumlah kasus negatif yang diprediksi benar sebagai kelas negatif.
3. *False Positive (FP)*: Jumlah kasus negatif yang diprediksi salah sebagai kelas positif.
4. *False Negative (FN)*: Jumlah kasus positif yang diprediksi salah sebagai kelas negatif.

Dari tabel 2.2 di atas, dapat dilakukan perhitungan untuk mengukur *accuracy*, *precision*, *recall*, dan *F-1 Score* (Derisma, 2020).

1. *Accuracy* (Akurasi)

Akurasi adalah prediksi benar pada kedua kelas (positif dan negatif) dari total prediksi. Akurasi menggambarkan seberapa baik model dalam melakukan prediksi. Rumus akurasi adalah sebagai berikut:

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.2)$$

2. *Precision (Presisi)*

Presisi adalah jumlah prediksi positif yang diprediksi dengan benar, dari keseluruhan data yang benar-benar positif.

$$Presisi = \frac{TP}{TP + FP} \quad (2.3)$$

3. *Recall (Sensitifitas)*

*Recall* atau sensitifitas adalah keberhasilan model dalam menangkap informasi. *Recall* mengukur seberapa banyak kasus positif yang berhasil diprediksi oleh model.

$$Recall = \frac{TP}{TP + FN} \quad (2.4)$$

4. *F1-score*

*F1-score* adalah perbandingan rata-rata presisi dan *recall*. *F1-score* dapat dikatakan baik jika presisi dan *recall* seimbang.

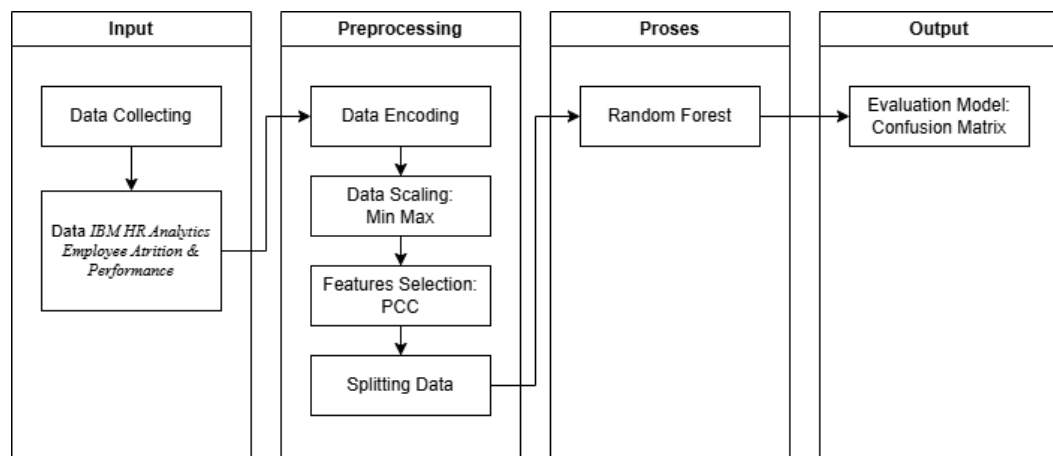
$$F1 - Score = 2 * \frac{Recall \times Presisi}{Recall + Presisi} \quad (2.5)$$

## BAB III

### DESAIN DAN IMPLEMENTASI

#### 3.1 Tahapan Penelitian

Bab ini membahas tentang langkah-langkah penelitian yang akan dilakukan untuk mendapat hasil yang diinginkan. Tahapan penelitian ditunjukkan oleh gambar berikut:



Gambar 3.1 Tahapan Penelitian

Gambar 3.1 menjelaskan tahapan penelitian yang dimulai dengan data *collecting* atau pengumpulan data serta fitur yang akan digunakan. Tahap berikutnya adalah *preprocessing* data, yang mencakup proses *encoding* dan *scaling* untuk mempersiapkan data sebelum analisis lebih lanjut. Setelah data dilakukan *encoding* dan *scaling*, tahap selanjutnya adalah *features selection* atau seleksi fitur, di mana fitur-fitur yang paling relevan dipilih untuk meningkatkan kinerja model. Data kemudian dibagi menjadi data *training* (data latih) dan data *testing* (data uji) untuk tujuan pelatihan dan pengujian model. Selanjutnya implementasi algoritma

*Random Forest* untuk membuat model prediksi. Langkah terakhir adalah *evaluation*, model akan dievaluasi untuk melihat seberapa baik dalam melakukan prediksi.

### 3.2 Data Collecting

Data pada penelitian ini merupakan data sekunder dari dataset “*IBM HR Analytics Employee Attrition & Performance*” yang diambil dari <https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset>.

Dataset ini disusun oleh ilmuwan IBM (*International Business Machines Corporation*) yang berbasis di Amerika Serikat dan mencakup informasi tentang karyawan termasuk data demografi, data kepegawaian, dan data kinerja. Fokus utama dari dataset ini adalah untuk menganalisis kinerja karyawan sebagai bagian dari strategi peningkatan kinerja dan kepuasan karyawan, serta untuk mengidentifikasi faktor-faktor yang memengaruhi kinerja karyawan.

Dataset IBM ini terdiri dari 35 atribut dengan total 1470 *record* data. Kolom-kolom dalam dataset meliputi informasi demografi seperti usia, jenis kelamin, pendidikan, dan status perkawinan. Informasi keahlian meliputi jabatan, masa kerja, dan keahlian. Serta informasi kepegawaian meliputi komitmen karir, kepuasan kerja, dan kinerja yang diperoleh.

Penggunaan dataset IBM ini bertujuan untuk melakukan prediksi *turnover* karyawan serta mengidentifikasi faktor-faktor yang signifikan terhadap kinerja karyawan. Analisis ini diharapkan dapat menghasilkan pola dan hubungan yang relevan, yang nantinya dapat menjadi acuan bagi perusahaan dalam merumuskan strategi retensi yang tepat.

Penjelasan dari fitur-fitur pada dataset dapat dilihat pada tabel 3.1.

Tabel 3.1 Fitur-fitur pada dataset

No	Fitur	Keterangan
1	<i>Age</i>	Merupakan usia karyawan.
2	<i>Atrition</i>	Menunjukkan apakah karyawan tersebut mengundurkan diri atau tidak.
3	<i>Business Travel</i>	Tingkat frekuensi perjalanan bisnis yang dilakukan oleh karyawan.
4	<i>Daily Rate</i>	Tarif harian atau gaji harian karyawan.
5	<i>Departement</i>	Menunjukkan departemen di mana karyawan bekerja,
6	<i>Distance from Home</i>	Jarak tempuh dari rumah karyawan ke tempat kerja.
7	<i>Education</i>	Tingkat pendidikan karyawan.
8	<i>Education Field</i>	Bidang studi atau keahlian pendidikan karyawan.
9	<i>Employee Count</i>	Jumlah karyawan dalam satu unit atau kelompok tertentu.
10	<i>Employee Number</i>	Nomor identifikasi untuk setiap karyawan.
11	<i>Environment Satisfaction</i>	Tingkat kepuasan karyawan terhadap lingkungan kerja.
12	<i>Gender</i>	Jenis kelamin karyawan.
13	<i>Hourly Rate</i>	Tarif atau gaji yang diterima karyawan per jam.
14	<i>Job Involvement</i>	Tingkat keterlibatan karyawan dalam pekerjaannya.
15	<i>Job Level</i>	Tingkat jabatan karyawan.
16	<i>Job Role</i>	Peran atau posisi yang diemban oleh karyawan.
17	<i>Job Satisfaction</i>	Tingkat kepuasan karyawan terhadap pekerjaannya.
18	<i>Marial Status</i>	Status pernikahan karyawan.
19	<i>Monthly Income</i>	Jumlah pendapatan yang diterima karyawan perbulan.
20	<i>Monthly Rate</i>	Tarif atau gaji bulanan karyawan.
21	<i>Num Companies Worked</i>	Jumlah perusahaan tempat karyawan tersebut pernah bekerja sebelumnya.
22	<i>Over18</i>	Menunjukkan apakah karyawan tersebut berusia di atas 18 tahun.
23	<i>Over Time</i>	Apakah karyawan tersebut melakukan lembur atau tidak.
24	<i>Percent Salary Hike</i>	Persentase kenaikan gaji terakhir yang diterima karyawan.
25	<i>Performance Rating</i>	Nilai atau rating kinerja karyawan.
26	<i>Relationship Satisfaction</i>	Tingkat kepuasan karyawan terhadap hubungan interpersonal di lingkungan kerja.
27	<i>Standard Hours</i>	Jumlah jam kerja standar per minggu.
28	<i>Stock Option Level</i>	Tingkat kepemilikan opsi saham oleh karyawan.
29	<i>Total Working Years</i>	Total tahun pengalaman kerja karyawan.
30	<i>Training Times Last Year</i>	Jumlah pelatihan yang diikuti oleh karyawan pada tahun sebelumnya.
31	<i>Work Life Balance</i>	Tingkat keseimbangan antara pekerjaan dan kehidupan pribadi karyawan.
32	<i>Years At Company</i>	Jumlah tahun yang sudah karyawan habiskan di perusahaan saat ini.
33	<i>Years In Current Role</i>	Jumlah tahun yang sudah karyawan bekerja dalam peran atau posisi pekerjaan saat ini.

No	Fitur	Keterangan
34	<i>Years Since Last Promotion</i>	Jumlah tahun sejak karyawan terakhir kali mendapatkan promosi.
35	<i>Years With Curr Manager</i>	Jumlah tahun yang sudah karyawan bekerja di bawah manajer saat ini.

Tabel 3.2 Contoh dataset

<i>Age</i>	<i>Attrition</i>	<i>Business Travel</i>	<i>DailyRate</i>	<i>Department</i>	<i>Distance FromHome</i>	<i>Education</i>	<i>Education Field</i>	<i>Employee Count</i>	<i>Employee Number</i>	<i>Environment Satisfaction</i>	<i>Gender</i>	<i>HourlyRate</i>	<i>JobInvolvement</i>	<i>JobLevel</i>	<i>JobRole</i>	<i>JobSatisfaction</i>	<i>Marital Status</i>
41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1	2	Female	94	3	2	Sales Executive	4	Single
49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2	3	Male	61	2	2	Research Scientist	2	Married
37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4	4	Male	92	2	1	Laboratory Technician	3	Single
33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5	4	Female	56	3	1	Research Scientist	3	Married
27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7	1	Male	40	3	1	Laboratory Technician	2	Married

Tabel 3.2 Contoh dataset (lanjutan)

<i>Monthly Income</i>	<i>MonthlyRate</i>	<i>NumCompaniesWorked</i>	<i>Over18</i>	<i>Overtime</i>	<i>PercentSalaryHike</i>	<i>PerformanceRating</i>	<i>RelationshipSatisfaction</i>	<i>StandardHours</i>	<i>StockOptionLevel</i>	<i>Totalworking years</i>	<i>TrainingTimes LastYear</i>	<i>WorkLifeBalance</i>	<i>YearsAt Company</i>	<i>YearsInCurrentRole</i>	<i>YearsSinceLast Promotion</i>	<i>YearsWithCurr Manager</i>
5993	19479	8	Y	Yes	11	3	1	80	0	8	0	1	6	4	0	5
5130	24907	1	Y	No	23	4	4	80	1	10	3	3	10	7	1	7
2090	2396	6	Y	Yes	15	3	2	80	0	7	3	3	0	0	0	0
2909	23159	1	Y	Yes	11	3	3	80	0	8	3	3	8	7	3	0
3468	16632	9	Y	No	12	3	4	80	1	6	3	3	2	2	2	2

### 3.3 *Preprocessing*

*Preprocessing* merupakan tahap penting dalam pengolahan data sebelum penerapan algoritma *machine learning*. Proses ini bertujuan untuk mempersiapkan data sehingga lebih mudah dipahami dan dapat diproses dengan efektif oleh model serta memaksimalkan hasil analisis yang diperoleh. Dalam penelitian ini, tahapan *preprocessing* difokuskan pada dua langkah utama yaitu *encoding* dan *scaling*.

#### 3.3.1 *Data Encoding*

Data *encoding* adalah salah satu tahap yang penting dilakukan pada *preprocessing* sebelum data kemudian diproses menggunakan algoritma *machine learning*. Proses *encoding* bertujuan untuk mengubah data kategorikal menjadi format yang lebih mudah dipahami oleh algoritma *machine learning*. Pada penelitian ini, dilakukan proses *encoding* menggunakan metode label *encoding*, yaitu mengubah data kategorik menjadi numerik.

Proses dari label *encoding* melibatkan identifikasi kolom kategorikal yang akan diubah. Jika datanya ordinal, urutan kategori ditetapkan sebelum penggantian, tetapi jika datanya nominal, angka akan diberikan secara acak. Selanjutnya, setiap data kategorik diubah menjadi nilai numerik. Misal pada kolom “*Atrition*” dengan kategori [“No”, “Yes”], label *encoding* akan mengubah datanya menjadi 0 dan 1. Untuk contoh data sebelum dan sesudah *encoding* dapat dilihat pada tabel 3.3 dan 3.4 di bawah:

Tabel 3.3 Contoh data sebelum *encoding*

Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	Gender	HourlyRate	JobInvolvement	JobLevel	JobRole	JobSatisfaction	MaritalStatus
41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1	2	Female	94	3	2	Sales Executive	4	Single
49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2	3	Male	61	2	2	Research Scientist	2	Married
37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4	4	Male	92	2	1	Laboratory Technician	3	Single
33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5	4	Female	56	3	1	Research Scientist	3	Married
27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7	1	Male	40	3	1	Laboratory Technician	2	Married

Tabel 3.4 Contoh dataset setelah *encoding*

<i>Age</i>	<i>Attrition</i>	<i>Business Travel</i>	<i>DailyRate</i>	<i>Department</i>	<i>DistanceFromHome</i>	<i>Education</i>	<i>EducationField</i>	<i>EmployeeCount</i>	<i>EmployeeNumber</i>	<i>EnvironmentSatisfaction</i>	<i>Gender</i>	<i>HourlyRate</i>	<i>JobInvolvement</i>	<i>JobLevel</i>	<i>JobRole</i>	<i>JobSatisfaction</i>	<i>MaritalStatus</i>
41	1	2	1102	2	1	2	1	1	1	2	0	94	3	2	7	4	2
49	0	1	279	1	8	1	1	1	2	3	1	61	2	2	6	2	1
37	1	2	1373	1	2	2	4	1	4	4	1	92	2	1	2	3	2
33	0	1	1392	1	3	4	1	1	5	4	0	56	3	1	6	3	1
27	0	2	591	1	2	1	3	1	7	1	1	40	3	1	2	2	1

### 3.3.2 Data Scaling

Data *scaling* adalah tahap *preprocessing* yang bertujuan untuk mengubah rentang nilai fitur tanpa mengubah proporsi atau hubungan antar data aslinya. Hal ini dilakukan agar algoritma pemodelan dapat bekerja secara optimal. Ketika data memiliki rentang yang sangat berbeda, model dikhawatirkan cenderung lebih memprioritaskan fitur dengan rentang yang lebih besar sehingga dapat mengurangi performa model.

Salah satu teknik *scaling* yang umum digunakan adalah *min-max scaling*. *Min-max scaling* mengubah nilai data menjadi rentang tertentu, biasanya antara 0 hingga 1 atau -1 hingga 1. Dengan *min-max scaling*, nilai minimal dari data asli akan diubah menjadi batas bawah yang diinginkan (misalnya, 0), sedangkan nilai maksimal akan diubah menjadi batas atas yang diinginkan (misalnya, 1).

Rumus yang digunakan dalam *min-max scaling* adalah sebagai berikut:

$$z = \frac{x - \min(x)}{[\max(x) - \min(x)]} \quad (3.1)$$

Keterangan:

$z$  = hasil *scaling*,  
 $x$  = nilai  $x$  (asli),  
 $\min(x)$  = nilai minimal untuk variabel  $x$ ,  
 $\max(x)$  = nilai maksimal untuk variabel  $x$ .

Setelah proses *min-max scaling*, semua nilai akan berada dalam rentang yang sama, sehingga algoritma dapat mengolah setiap fitur secara seimbang, meningkatkan stabilitas dan kecepatan model dalam mencapai hasil yang optimal pada tahap pelatihan.

Tabel 3.5 Contoh dataset sebelum *scaling*

Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	Gender	HourlyRate	JobInvolvement	JobLevel	JobRole	JobSatisfaction	MaritalStatus
41	1	2	1102	2	1	2	1	1	1	2	0	94	3	2	7	4	2
49	0	1	279	1	8	1	1	1	2	3	1	61	2	2	6	2	1
37	1	2	1373	1	2	2	4	1	4	4	1	92	2	1	2	3	2
33	0	1	1392	1	3	4	1	1	5	4	0	56	3	1	6	3	1
27	0	2	591	1	2	1	3	1	7	1	1	40	3	1	2	2	1

Tabel 3.6 Contoh dataset setelah *scaling*

Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	Gender	HourlyRate	JobInvolvement	JobLevel	JobRole	JobSatisfaction	MaritalStatus
0.5476190476	1	1	0.7158196135	1	0	0.25	0.2	0	0	0.3333333333	0	0.9142857143	0.6666666667	0.25	0.875	1	1
0.7380952381	0	0.5	0.1267000716	0.5	0.25	0	0.2	0	0.0004837929366	0.6666666667	1	0.4428571429	0.3333333333	0.25	0.75	0.3333333333	0.5
0.4523809524	1	1	0.9098067287	0.5	0.03571428571	0.25	0.8	0	0.00145137881	1	1	0.8857142857	0.3333333333	0	0.25	0.6666666667	1
0.3571428571	0	0.5	0.9234073014	0.5	0.07142857143	0.75	0.2	0	0.001935171746	1	0	0.3714285714	0.6666666667	0	0.75	0.6666666667	0.5
0.2142857143	0	1	0.350035791	0.5	0.03571428571	0	0.6	0	0.00290275762	0	1	0.1428571429	0.6666666667	0	0.25	0.3333333333	0.5

### 3.3.3 *Features Selection*

*Features Selection* atau seleksi fitur dilakukan untuk memilih fitur mana saja yang relevan dengan variabel target. Pada penelitian ini, metode seleksi fitur yang dipilih adalah PCC, yang digunakan untuk mengukur korelasi antara setiap fitur dengan variabel target, yaitu *atrition*. Fitur-fitur dengan korelasi tinggi, baik positif maupun negatif, dipilih karena dianggap paling berpengaruh terhadap resiko *turnover*. Setelah proses perhitungan korelasi, fitur-fitur diurutkan berdasarkan besarnya korelasi dengan variabel target.

Selanjutnya untuk keperluan analisis terhadap pelatihan dan melihat kinerja model, dilakukan beberapa skenario sebagai berikut:

1. Skenario pertama, menggunakan seluruh fitur yang ada dengan tujuan untuk membandingkan apakah performa model menjadi lebih baik ketika menggunakan seluruh informasi dalam dataset atau malah sebaliknya.
2. Skenario kedua, menggunakan 5 fitur teratas atau dengan korelasi tertinggi terhadap variabel target. Tujuannya agar model dapat menjadi lebih sederhana dan hanya fokus terhadap fitur-fitur yang paling relevan.
3. Skenario ketiga, menggunakan 10 fitur teratas. Tujuannya memberikan lebih banyak informasi kepada model sehingga dapat meningkatkan performa prediksinya. Selain itu, jumlah fitur yang lebih banyak diharapkan membantu model dalam menangkap lebih banyak variasi data.
4. Skenario keempat, menggunakan 15 fitur teratas. Tujuannya agar model dapat memanfaatkan lebih banyak fitur yang signifikan, namun tetap menjaga relevansi dengan variabel target.

Untuk hasil seleksi fitur yang telah dilakukan menggunakan metode PCC adalah sebagai berikut:

Tabel 3. 7 Hasil Seleksi Fitur

<b>Hasil Seleksi Fitur</b>			
	<b>No.</b>	<b>Fitur</b>	<b>Correlation Coefficient</b>
		<i>Attrition</i>	1.000000
5 fitur teratas	1.	<i>Overtime</i>	0.246118
	2.	<i>Totalworkingyears</i>	0.171063
	3.	<i>JobLevel</i>	0.169105
	4.	<i>MaritalStatus</i>	0.162070
	5.	<i>YearsInCurrentRole</i>	0.160545
10 fitur teratas	6.	<i>MonthlyIncome</i>	0.159840
	7.	<i>Age</i>	0.159205
	8.	<i>YearsWithCurrManager</i>	0.156199
	9.	<i>StockOptionLevel</i>	0.137145
	10.	<i>YearsAtCompany</i>	0.134392
15 fitur teratas	11.	<i>JobInvolvement</i>	0.130016
	12.	<i>JobSatisfaction</i>	0.103481
	13.	<i>EnvironmentSatisfaction</i>	0.103369
	14.	<i>DistanceFromHome</i>	0.077924
	15.	<i>JobRole</i>	0.067151
20 fitur teratas	16.	<i>Department</i>	0.063991
	17.	<i>WorkLifeBalance</i>	0.063939
	18.	<i>TrainingTimesLastYear</i>	0.059478
	19.	<i>DailyRate</i>	0.056652
	20.	<i>RelationshipSatisfaction</i>	0.045872
25 fitur teratas	21.	<i>NumCompaniesWorked</i>	0.043494
	22.	<i>YearsSinceLastPromotion</i>	0.033019
	23.	<i>Education</i>	0.031373
	24.	<i>Gender</i>	0.029453
	25.	<i>EducationField</i>	0.026846

Hasil Seleksi Fitur			
	No.	Fitur	Correlation Coefficient
30 fitur teratas	26.	<i>MonthlyRate</i>	0.015170
	27.	<i>PercentSalaryHike</i>	0.013478
	28.	<i>EmployeeNumber</i>	0.010577
	29.	<i>HourlyRate</i>	0.006846
	30.	<i>PerformanceRating</i>	0.002889
Seluruh fitur	31.	<i>BusinessTravel</i>	0.000074
	32.	<i>EmployeeCount</i>	0.000049
	33.	<i>Over18</i>	0.000011
	34.	<i>StandardHours</i>	0.000009

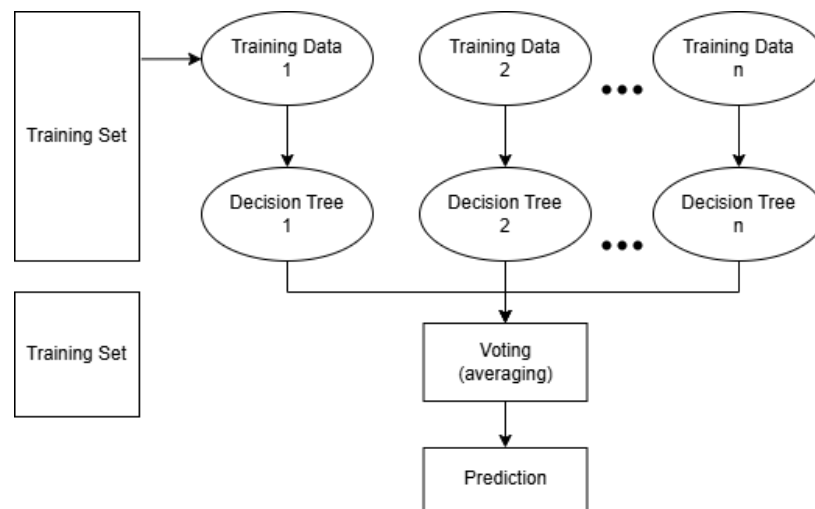
### 3.3.4 Split Data

*Split* data berguna untuk membagi data menjadi dua bagian yaitu data *training* (data latih) dan data *testing* (data uji). Hal ini sangat penting dilakukan karena dalam pemodelan data *machine learning*, model harus dilatih dan diuji dengan data yang berbeda. Data *training* digunakan untuk membangun model, sementara data *testing* digunakan untuk mengevaluasi kinerja model.

Pada penelitian ini, terdapat tiga rasio pembagian data untuk data *training* dan data *testing*. Rasio pertama yaitu 90:10, 90% untuk data *training* dan 10% untuk data *testing*. Rasio kedua yaitu 80:20, 80% untuk data *training* dan 20% untuk data *testing*. Rasio ketiga adalah 70:30, 70% untuk data *training* dan 10% untuk data *testing*. Tujuan dari penggunaan rasio yang bervariasi adalah untuk menganalisis pengaruh rasio *split* data terhadap kinerja model prediksi.

### 3.4 Algoritma *Random Forest*

Tahapan selanjutnya adalah implementasi algoritma *Random Forest* untuk model prediksi. Berikut adalah diagram proses dari algoritma *Random Forest*:



Gambar 3.2 Diagram proses *Random Forest*

Gambar 3.2 menunjukkan tahapan dari implementasi algoritma *Random Forest* dalam membuat model prediksi. Pertama, *training* set akan dibagi menjadi beberapa subset acak menggunakan teknik *bootstrap sampling*. Setiap subset, yang disebut *training data 1*, *training data 2*, hingga *training data n* adalah sampel dari *training* set asli yang dibuat dengan dipilih secara acak melalui pengulangan. Setelah subset terbentuk, masing-masing subset tersebut digunakan untuk membangun *decision tree*. Setiap *decision tree* dilatih menggunakan subset data yang berbeda dan memilih fitur secara acak di setiap *node* yang akan menghasilkan pohon-pohon keputusan yang sedikit berbeda satu sama lain. Setelah semua *decision tree* terbentuk, selanjutnya adalah proses *voting* untuk menggabungkan hasil dari setiap pohon dan menemukan suara terbanyak untuk menjadi hasil akhir.

### 3.4.1 Penentuan *Hyperparameter Random Forest*

Penentuan *hyperparameter Random Forest* merupakan proses untuk mencari nilai optimal dari beberapa *hyperparameter* yang diharapkan dapat memaksimalkan performa model *Random Forest*. *Hyperparameter* adalah parameter yang nilainya ditetapkan sebelum proses pelatihan model dan tidak diubah selama pelatihan berlangsung. Beberapa *hyperparameter* yang diterapkan pada penelitian ini adalah:

1. *n\_estimatorss* : Jumlah *tree* yang akan dibangun dalam *Random Forest*.
2. *max\_depth* : Maksimum kedalaman setiap *tree*.
3. *max\_features* : Jumlah fitur yang akan digunakan dalam setiap *tree*.

### 3.4.2 Pembentukan Model *Random Forest*

Pembentukan model *Random Forest* untuk prediksi *turnover* karyawan akan dilakukan menggunakan bahasa pemrograman *python* dengan menggunakan *tools google colab*. Pembentukan model ini melibatkan seluruh proses mulai dari persiapan data yang meliputi *preprocessing* dan seleksi fitur, kemudian data dibagi menjadi data *training* dan data *testing* untuk kemudian dilakukan pembentukan model, setelah model terbentuk dilakukan pelatihan pada model, tahap terakhir adalah evaluasi model.

## 3.5 Evaluasi

Evaluasi model prediksi adalah proses untuk menilai kinerja model yang digunakan untuk melakukan prediksi. Terdapat beberapa metode untuk melakukan evaluasi model prediksi, salah satunya adalah menggunakan *Confusion Matrix*.

*Confusion Matrix* akan menggambarkan seberapa sering model memprediksi label yang benar dan salah. Setiap baris dari tabel tersebut mewakili sebuah label aktual, sedangkan setiap kolom mewakili label yang diprediksi oleh model. Empat hasil potensial yang dapat diperoleh dari *Confusion Matrix* adalah:

1. *True Positive* (TP): Merupakan data positif yang diprediksi benar.
2. *True Negative* (TN): Merupakan data negatif yang diprediksi benar.
3. *False Positive* (FP): Merupakan data negatif yang diprediksi positif.
4. *False Negative* (FN): Merupakan data positif yang diprediksi negatif.

Dengan menggunakan *Confusion Matrix*, kita dapat menghitung berbagai metrik evaluasi seperti akurasi, presisi, *recall*, dan *f1-score*. Akurasi menghitung rata-rata perbedaan antara nilai prediksi dengan nilai aktual, presisi menghitung rasio jumlah prediksi yang benar terhadap jumlah prediksi yang dilakukan, dan *recall* menghitung rasio jumlah prediksi yang benar terhadap jumlah data yang seharusnya diprediksi, dan *f1-score* menggabungkan presisi dan *recall* untuk memberikan gambaran yang lebih menyeluruh tentang performa model, terutama saat terdapat ketidakseimbangan kelas.

### **3.6 Eksperimen**

Pada tahap eksperimen ini, akan dilakukan perhitungan manual menggunakan 10 baris data dan 2 fitur terpilih berdasarkan hasil seleksi fitur yang telah dilakukan sebelumnya.

Berdasarkan data pada tabel 3.7, fitur *Overtime* (kerja lembur) menunjukkan peringkat tertinggi dalam hal korelasi dengan variabel *Attrition*. Selanjutnya, fitur *Totalworkingyears* (lama bekerja di perusahaan) juga termasuk di antara fitur-fitur

teratas yang berkorelasi. Analisis ini mengindikasikan adanya hubungan positif yang lemah antara *Overtime* dan *Attrition*, di mana karyawan yang terlibat dalam kerja lembur memiliki kecenderungan yang lebih besar untuk mengalami *Attrition*. Selain itu, *Totalworkingyears* juga menunjukkan korelasi positif dengan *Attrition*, kekuatan hubungan ini relatif lebih lemah dibandingkan dengan *Overtime*. Analisis ini memberikan penemuan bahwa faktor kerja lembur lebih signifikan dalam memengaruhi kemungkinan terjadinya *attrition* dibandingkan dengan lama bekerja di perusahaan.

Tabel 3. 8 Sampel dataset perhitungan manual

<i>Attrition</i>	<i>Overtime</i>	<i>Totalworkingyears</i>
Yes	Yes	8
No	No	10
Yes	Yes	7
No	Yes	8
No	No	6
Yes	No	8
No	Yes	12
No	No	1
Yes	No	10
No	No	17

Tabel 3.8 menunjukkan data karyawan berdasar variabel *Atrition*, *Overtime*, dan *Totalworkingyears* diambil 10 baris teratas untuk selanjutnya akan dilakukan perhitungan manual menggunakan metode *Random Forest*. Untuk langkah-langkah perhitungannya sebagai berikut:

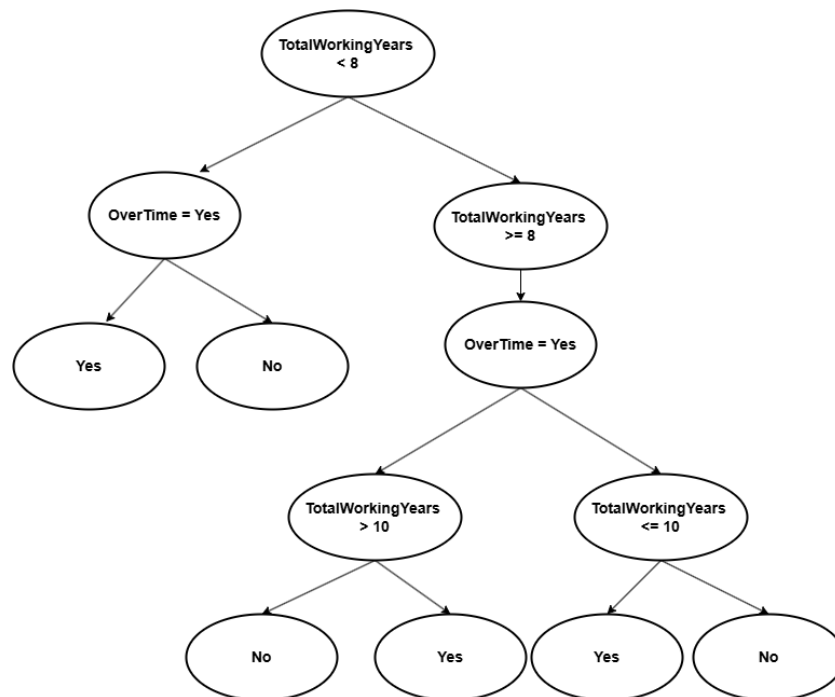
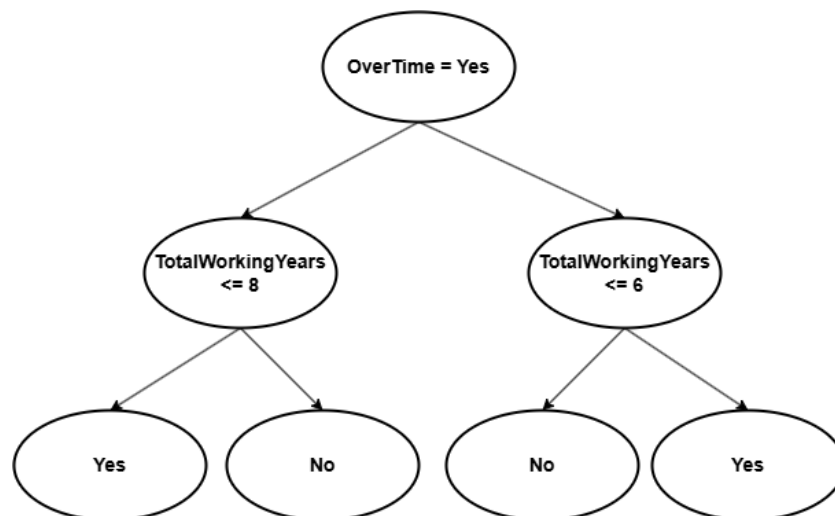
1. Pembuatan sampel *bootstrap* (data baru) dari data di atas agar tidak menggunakan data yang sama di setiap *tree*.

Tabel 3.9 Dataset setelah dilakukan *bootstrap*

Sampel <i>Bootstrap</i> 1			Sampel <i>Bootstrap</i> 2			Sampel <i>Bootstrap</i> 3		
<i>Attrition</i>	<i>Overtime</i>	<i>Total Working</i>	<i>Attrition</i>	<i>Overtime</i>	<i>Total Working</i>	<i>Attrition</i>	<i>Overtime</i>	<i>Total Working</i>
Yes	Yes	8	No	No	10	No	Yes	12
Yes	Yes	7	Yes	Yes	7	Yes	Yes	7
Yes	Yes	8	No	No	10	Yes	Yes	8
No	No	6	No	No	6	No	No	17
Yes	No	10	No	No	1	No	No	1
Yes	No	10	No	Yes	12	Yes	Yes	7
Yes	No	8	Yes	No	10	Yes	Yes	8
No	Yes	12	No	No	10	No	No	6
No	No	1	No	Yes	8	Yes	Yes	8
No	No	17	No	No	6	Yes	No	10

2. Pelatihan *decision tree* menggunakan masing-masing sampel *bootstrap* yang telah dibuat sebelumnya.

Pada tahap ini, akan dibuat pelatihan melalui pembentukan beberapa *decision tree* untuk memprediksi kelas target (untuk kelas target pada prediksi ini adalah “*Attrition*”) yang berarti karyawan akan melakukan pengunduran diri atau tidak. Atribut yang dianalisis dalam pelatihan *decision tree* ini adalah “*Overtime*” dan “*TotalWorkingYears*”. *Decision tree* yang dibentuk digambarkan pada gambar 3.3 dan 3.4 di bawah:

Gambar 3.3 Pembentukan *decision tree bootstrap 1*Gambar 3.4 Pembentukan *decision tree bootstrap 2 dan 3*

Pelatihan pada *decision tree* ini dilakukan menggunakan tiap baris data secara berurutan mulai dari sampel *bootstrap* pertama hingga ketiga, dan seterusnya hingga mendapatkan jumlah *decision tree* yang diinginkan.

3. Prediksi menggunakan data baru untuk setiap *decision tree* yang telah dibentuk.

Contoh data baru pertama :

*Overtime* : No

*Totalworkingyears* : 5

Hasil Prediksi

*Tree 1* = No

*Tree 2* = No

*Tree 3* = No

Berdasarkan ketiga *decision tree* yang telah dibuat sebelumnya, semua *tree* memprediksi “No” yang berarti hasil prediksi akhir untuk data di atas adalah “No”.

Contoh data baru kedua :

*Overtime* : Yes

*Totalworkingyears* : 5

Hasil Prediksi

*Tree 1* = Yes

*Tree 2* = No

*Tree 3* = No

Berdasarkan ketiga *decision tree* yang telah dibuat sebelumnya, pada *tree* pertama menghasilkan prediksi “Yes” sedangkan pada *tree* ke 2 dan ke 3 menghasilkan prediksi “No”. Jika dilihat dari *majority vote*, maka prediksi akhir untuk data kedua adalah “No”. Semua *tree* memprediksi “No” yang berarti hasil prediksi akhir untuk data di atas adalah “No”.

### 3.7 Skenario Uji Coba

Skenario uji coba dilakukan untuk mengavaluasi model yang telah dibuat dan bertujuan untuk mengetahui kombinasi parameter mana yang menghasilkan akurasi terbaik. Pembagian skenario uji coba terdapat pada tabel 3.10 sebagai berikut:

Tabel 3.10 Skenario uji coba

Skenario		Sub Skenario	
A	Seluruh fitur	Ratio <i>Split</i> Data 90 : 10 80 : 20 70 : 30	<i>Tuning Hyperparameter</i>
B	5 Fitur		
C	10 Fitur		
D	15 fitur		

Berdasar skenario di atas, pengujian akan dilakukan dengan melibatkan eskplorasi kinerja model dengan variasi jumlah fitur dan penyesuaian nilai *hyperparameter*. Detail nilai dari tiap-tiap *hyperparameter* yang akan digunakan dalam skenario pengujian adalah sebagai berikut:

Tabel 3.11 Nilai *hyperparameter*

<i>Hyperparameter</i>	Nilai
<i>n_estimators</i>	[10,50,100],
<i>max_depth</i>	[1,5,10],
<i>max_features</i>	[2,5,10]

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1 Hasil Uji Coba**

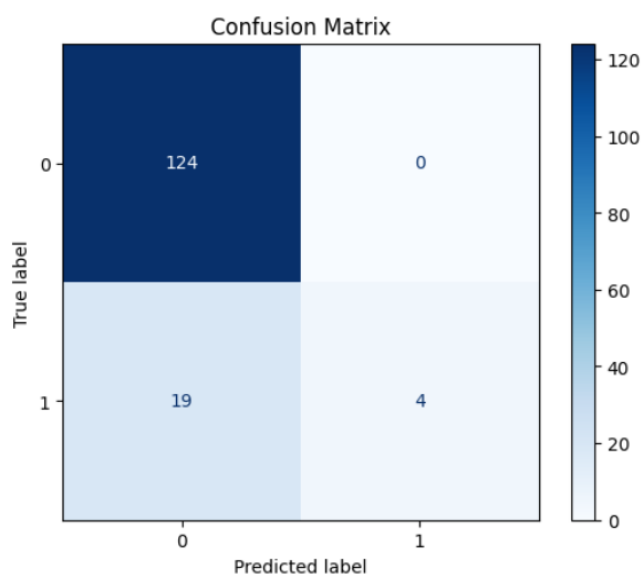
Bab ini menyajikan hasil pengujian dan analisis yang telah dilakukan berdasarkan skenario uji coba yang dijelaskan pada tabel 3.10 dan 3.11. Skenario uji coba tersebut dirancang untuk mengevaluasi performa model dengan berbagai kombinasi, termasuk pengaruh seleksi fitur menggunakan PCC dan variasi rasio *split* data antara *training* dan *testing*. Pengujian dalam penelitian ini mencakup empat skenario utama, yaitu pengujian dengan dan tanpa seleksi fitur. Setiap skenario utama ini kemudian dibagi lagi menjadi tiga sub skenario, yang masing-masing menggambarkan variasi dalam rasio *split* data antara *training* dan *testing*, yaitu rasio 90:10, 80:20, dan 70:30. Setiap skenario juga melakukan uji coba tuning *hyperparameter* yang bertujuan untuk mencari kombinasi terbaik yang dapat meningkatkan akurasi dan performa model.

##### **4.1.1 Uji Coba Skenario A**

Uji coba skenario A dilakukan dengan menggunakan seluruh fitur pada *dataset* yang kemudian dibagi menjadi tiga rasio pelatihan, yaitu 90:10, 80:20, dan 70:30. Pada masing-masing sub skenario tersebut, dilakukan *tuning hyperparameter* untuk menemukan kombinasi terbaik.

### A. Rasio 90:10

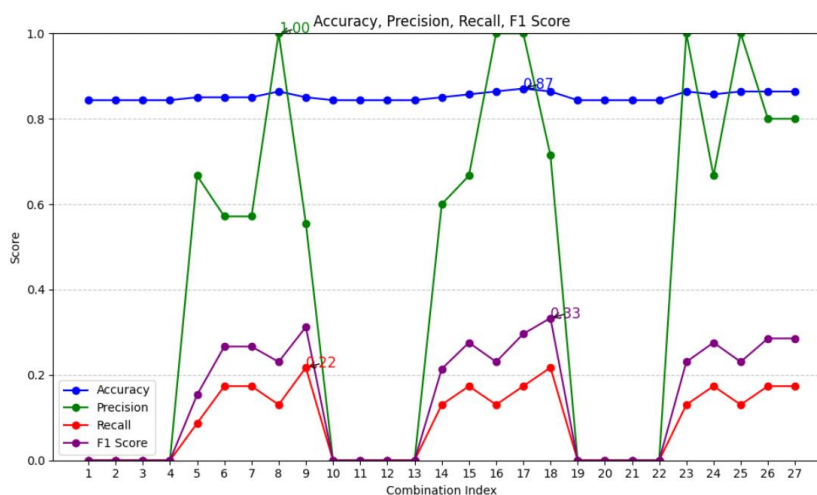
Pengujian menggunakan rasio *split* data 90% untuk data *training* dan 10% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut adalah *confusion matrix* untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.1 *Confusion Matrix* kombinasi terbaik skenario A 90:10

Gambar 4.1 merupakan hasil *confusion matrix* dari model dalam memprediksi kelas 0 (negatif) dan 1 (positif) dengan menggunakan kombinasi *hyperparameter* terbaik. Nilai-nilai dalam matriks menunjukkan jumlah observasi yang diprediksi dengan benar atau salah. Model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 124 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model. *False Positive* (FP) sebanyak 0, yaitu tidak adanya data karyawan yang tidak mengalami *turnover* namun diprediksi sebagai *turnover*. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 19 data,

yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 4 data, yang berarti karyawan yang mengalami *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:



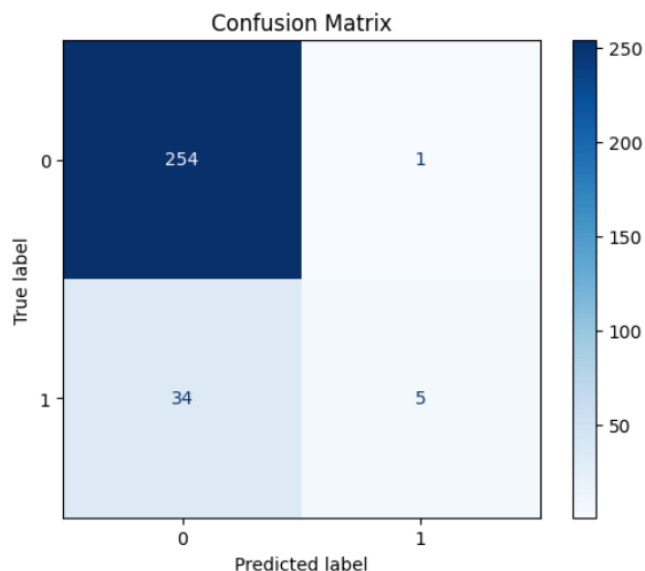
Gambar 4.2 Hasil seluruh pengujian skenario A 90:10

Gambar 4.2 menunjukkan akurasi model cenderung stabil di kisaran 84% hingga 87%, namun presisi fluktuatif dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* bervariasi, dengan nilai rendah pada beberapa kombinasi, dan *f1-score* menunjukkan pola serupa dengan *recall*. Hal ini mungkin disebabkan oleh ketidakseimbangan antara kelas 0 dan 1.

## B. Rasio 80:20

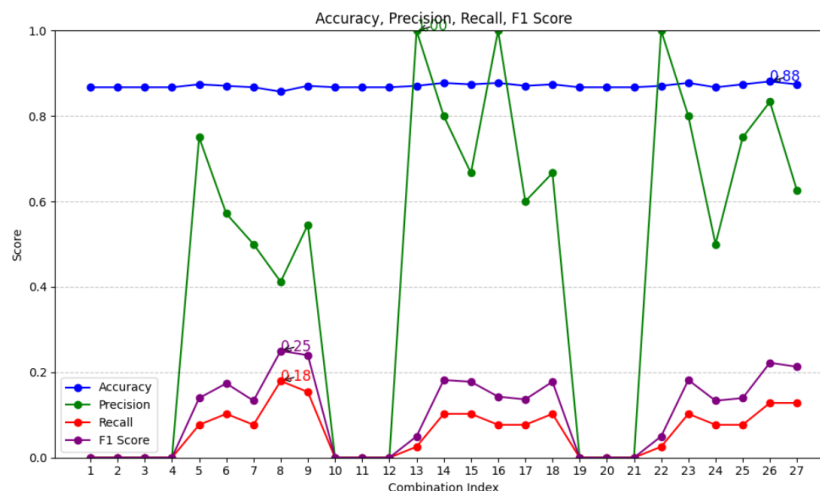
Pengujian ini menggunakan rasio *split* data 80% untuk data *training* dan 20% untuk data *testing* yang selanjutnya dilakukan tuning *hyperparameter*. Berikut

adalah *confusion matrix* untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.3 *Confusion Matrix* kombinasi terbaik skenario A 80:20

Gambar 4.3 merupakan hasil *confusion matrix* dari model dalam memprediksi kelas 0 dan 1. Nilai-nilai dalam matriks menunjukkan jumlah observasi yang diprediksi dengan benar atau salah. Model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 254 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model. *False Positive* (FP) sebanyak 1 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 34 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover*. *True Positive* (TP) hanya menghasilkan sebanyak 5 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:

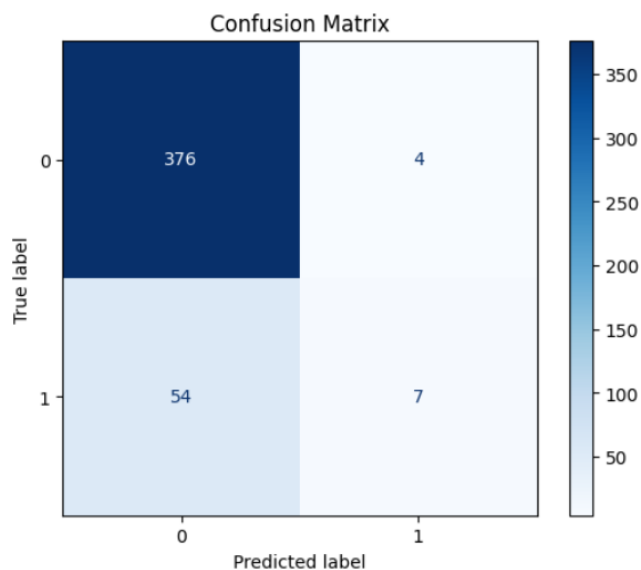


Gambar 4.4 Hasil seluruh pengujian skenario A 80:20

Gambar 4.4 menunjukkan akurasi model cenderung stabil di kisaran 85% hingga 88%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter*. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

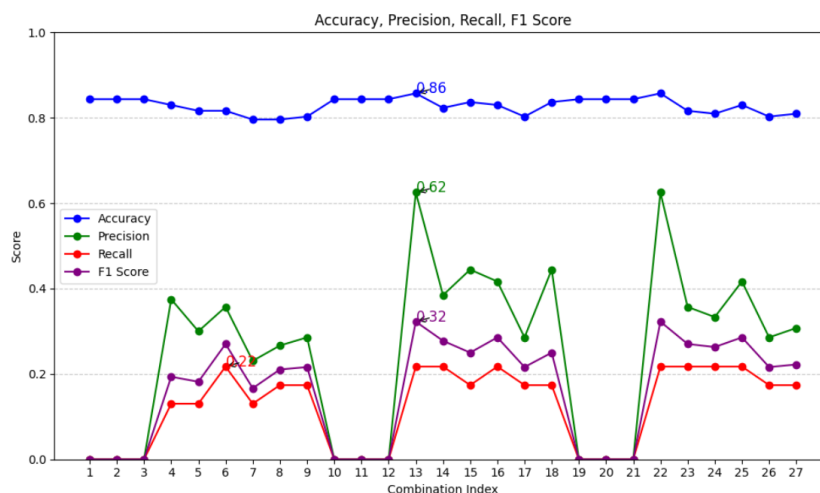
### C. Rasio 70:30

Pengujian ini menggunakan rasio *split* data 70% untuk data *training* dan 30% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut adalah *confusion matrix* untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.5 *Confusion matrix* kombinasi terbaik skenario A 70:30

Gambar 4.5 merupakan hasil *confusion matrix* dari model dalam memprediksi kelas 0 dan 1. Nilai-nilai dalam matriks menunjukkan jumlah observasi yang diprediksi dengan benar atau salah. Model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 376 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model. *False Positive* (FP) sebanyak 4 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 54 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover*. *True Positive* (TP) juga hanya menghasilkan sebanyak 7 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:



Gambar 4.6 Hasil seluruh pengujian skenario A 70:30

Gambar 4.6 menunjukkan akurasi model cenderung stabil di kisaran 84% hingga 86%, akan tetapi presisi tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

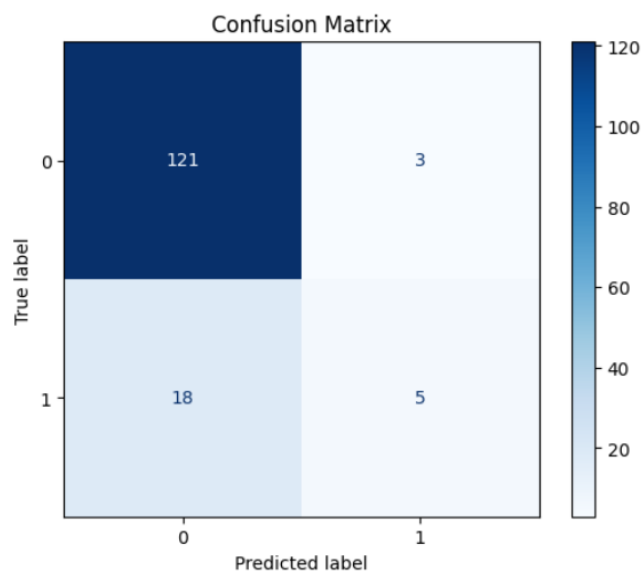
#### 4.1.2 Uji Coba Skenario B

Uji coba skenario B dilakukan dengan menggunakan 5 fitur teratas berdasarkan proses seleksi fitur yang telah dilakukan, kemudian dibagi menjadi tiga rasio pelatihan, yaitu 90:10, 80:20, dan 70:30. Pada masing-masing sub skenario tersebut, dilakukan *tuning hyperparameter* untuk menemukan kombinasi terbaik.

##### A. Rasio 90:10

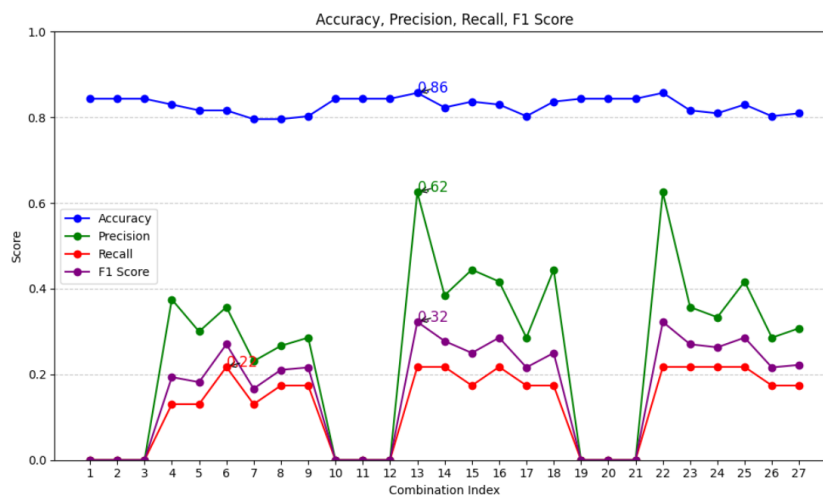
Pengujian ini menggunakan rasio *split* data 90% untuk data *training* dan 10% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut

adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.7 *Confusion matrix* kombinasi terbaik skenario A 90:10

Gambar 4.7 merupakan hasil *confusion matrix* dari model dalam memprediksi kelas 0 dan 1. Model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 121 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model dan *False Positive* (FP) sebanyak 3 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 18 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 5 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:

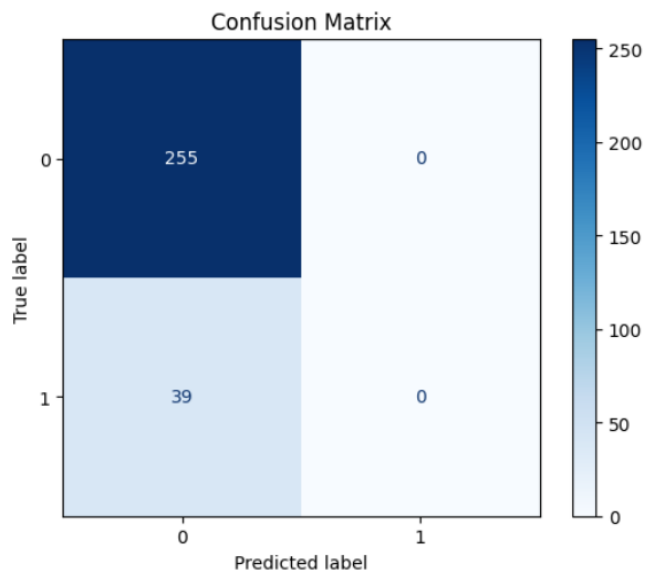


Gambar 4.8 Hasil seluruh pengujian skenario B 90:10

Gambar 4.8 menunjukkan akurasi model cenderung stabil di kisaran 84% hingga 86%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

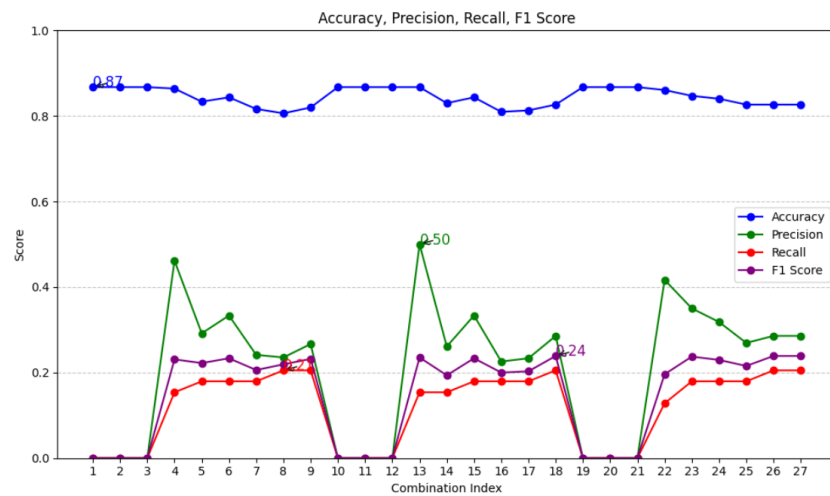
## B. Rasio 80:20

Pengujian ini menggunakan rasio *split* data 80% untuk data *training* dan 20% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.9 *Confusion matrix* kombinasi terbaik skenario B 80:20

Gambar 4.9 menunjukkan bahwa model terlihat sangat baik dalam memprediksi kelas 0, namun kurang optimal dalam mendeteksi kelas 1. Terdapat 255 sampel dari kelas 0 yang diklasifikasikan dengan benar sebagai kelas 0 (*True Negative*), sementara tidak ada sampel dari kelas 0 yang salah diklasifikasikan sebagai kelas 1 (*False Positive*). Di sisi lain, model tidak mampu mengidentifikasi kelas 1 dengan benar sama sekali, yang ditunjukkan dengan tidak adanya nilai *True Positive* untuk kelas tersebut. Sebanyak 39 sampel dari kelas 1 diklasifikasikan sebagai kelas 0 (*False Negative*), yang menunjukkan bahwa model cenderung bias terhadap kelas 0. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:

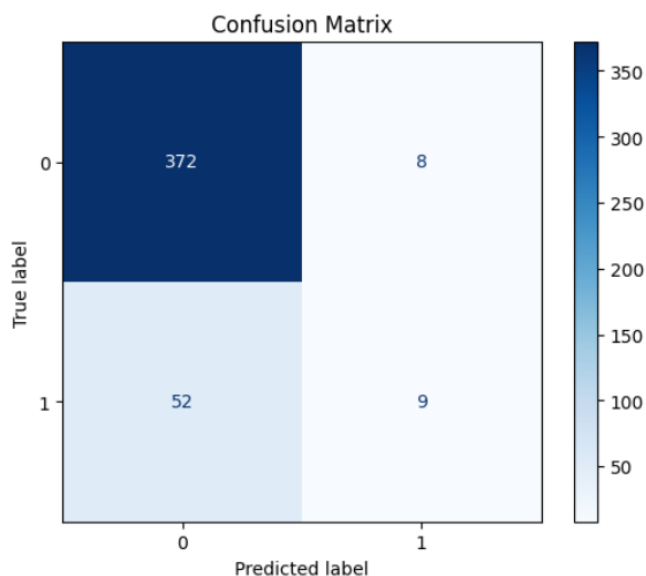


Gambar 4.10 Hasil seluruh pengujian skenario B 80:20

Gambar 4.10 menghasilkan akurasi model cenderung stabil di kisaran 85% hingga 87%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

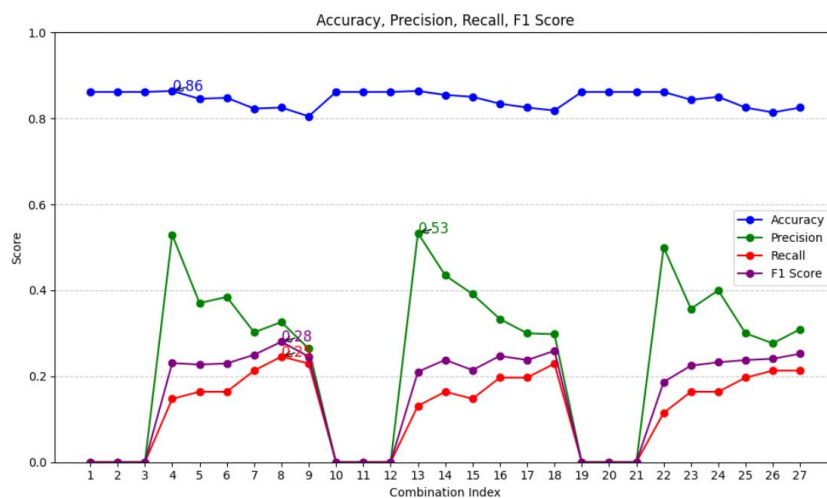
### C. Rasio 70:30

Pengujian ini menggunakan rasio *split* data 70% untuk data *training* dan 30% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.11 *Confusion matrix* kombinasi terbaik skenario B 70:30

Gambar 4.11 menghasilkan model yang terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 372 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model dan *False Positive* (FP) sebanyak 8 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 52 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 9 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:



Gambar 4.12 Hasil seluruh pengujian skenario B 70:30

Gambar 4.12 menunjukkan akurasi model cenderung stabil di kisaran 84% hingga 86%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

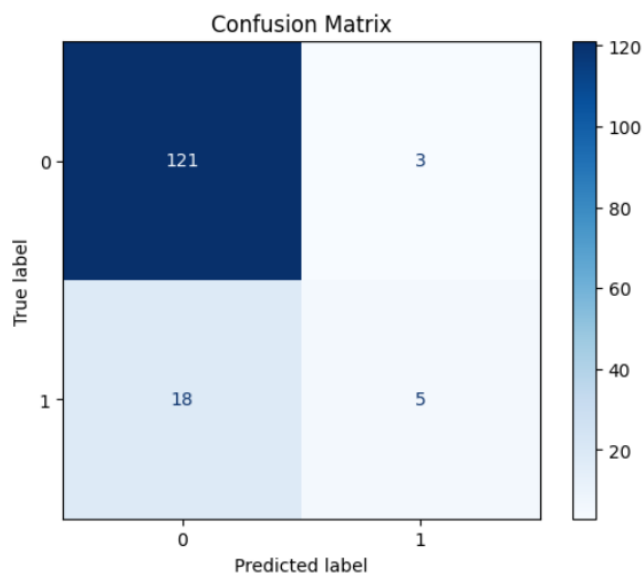
### 4.1.3 Uji Coba Skenario C

Uji coba skenario C dilakukan dengan menggunakan 10 fitur teratas berdasarkan proses seleksi fitur yang telah dilakukan, kemudian dibagi menjadi tiga rasio pelatihan, yaitu 90:10, 80:20, dan 70:30. Pada masing-masing sub skenario tersebut, dilakukan *tuning hyperparameter* untuk menemukan kombinasi terbaik.

#### A. Rasio 90:10

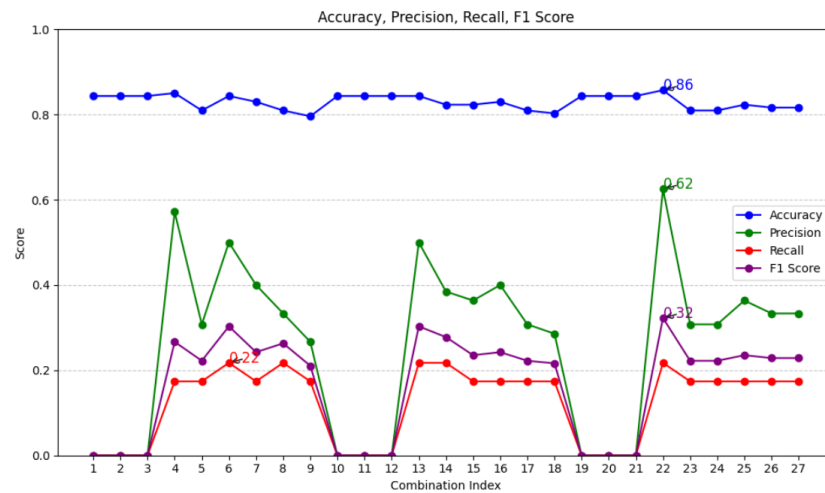
Pengujian ini menggunakan rasio *split* data 90% untuk data *training* dan 10% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut

adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.13 *Confusion matrix* kombinasi terbaik skenario C 90:10

Gambar 4.13 menunjukkan bahwa model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 121 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model dan *False Positive* (FP) sebanyak 3 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 18 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 5 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:

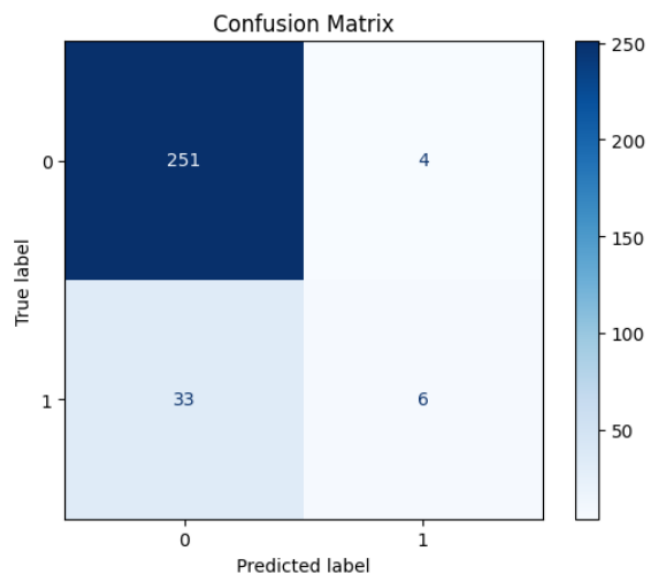


Gambar 4.14 Hasil seluruh pengujian skenario C 90:10

Gambar 4.14 menunjukkan akurasi model cenderung stabil di kisaran 84% hingga 86%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

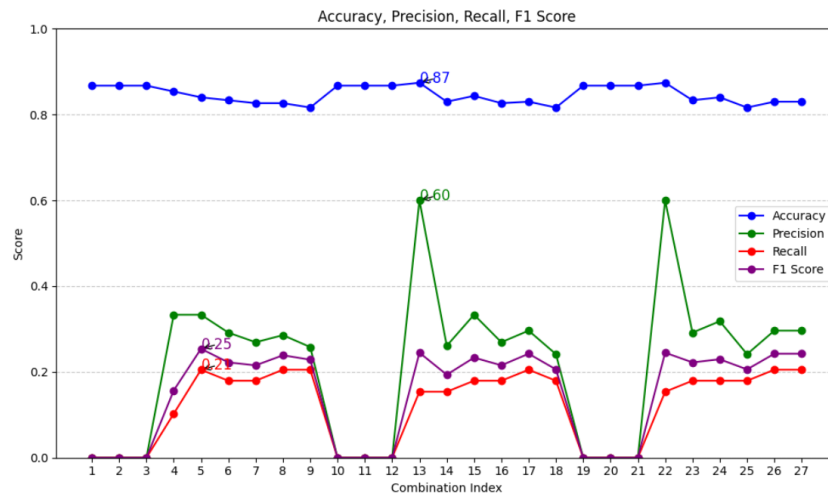
## B. Rasio 80:20

Pengujian ini menggunakan rasio *split* data 80% untuk data *training* dan 20% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.15 Hasil pengujian skenario C 80:20

Gambar 4.15 menunjukkan bahwa model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 251 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model dan *False Positive* (FP) sebanyak 4 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 33 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 6 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:

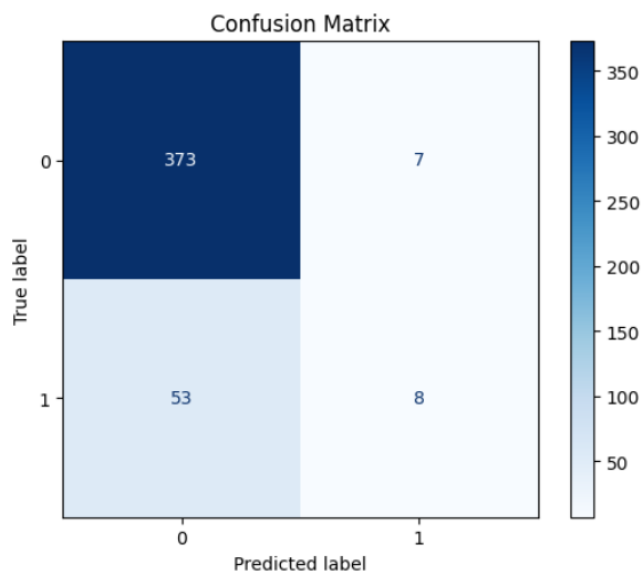


Gambar 4.16 Hasil seluruh pengujian skenario C 80:20

Gambar 4.16 menunjukkan akurasi model cenderung stabil di kisaran 85% hingga 86%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

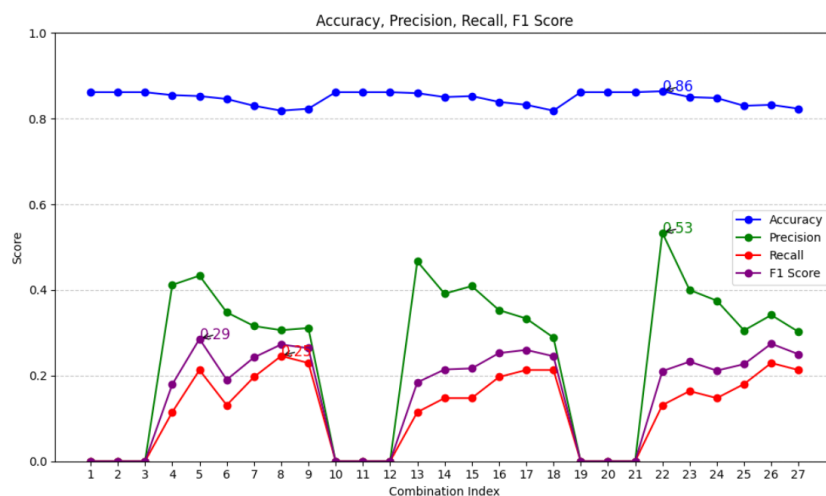
### C. Rasio 70:30

Pengujian ini menggunakan rasio *split* data 70% untuk data *training* dan 30% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.17 *Confusion matrix* kombinasi terbaik skenario C 70:30

Gambar 4.17 menunjukkan bahwa model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 373 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model dan *False Positive* (FP) sebanyak 7 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 53 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 8 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:



Gambar 4.18 Hasil seluruh pengujian skenario C 70:30

Gambar 4.18 menunjukkan akurasi model cenderung stabil di kisaran 84% hingga 86%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

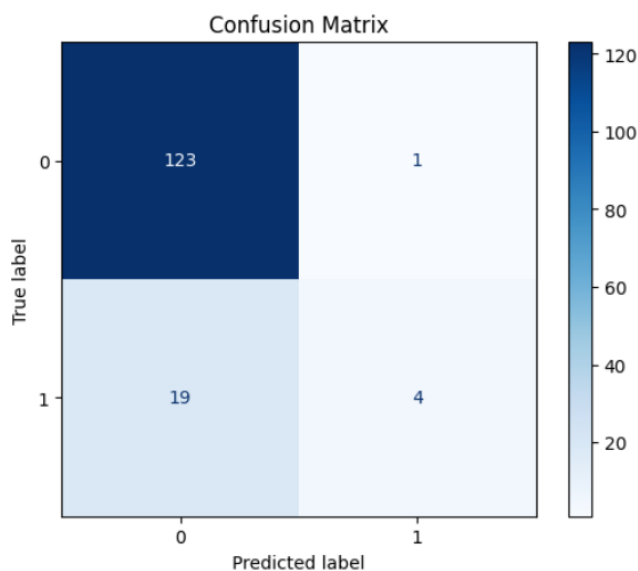
#### 4.1.4 Uji Coba Skenario D

Uji coba skenario D dilakukan dengan menggunakan 15 fitur teratas berdasarkan proses seleksi fitur yang telah dilakukan, kemudian dibagi menjadi tiga rasio pelatihan, yaitu 90:10, 80:20, dan 70:30. Pada masing-masing sub skenario tersebut, dilakukan *tuning hyperparameter* untuk menemukan kombinasi terbaik.

##### A. Rasio 90:10

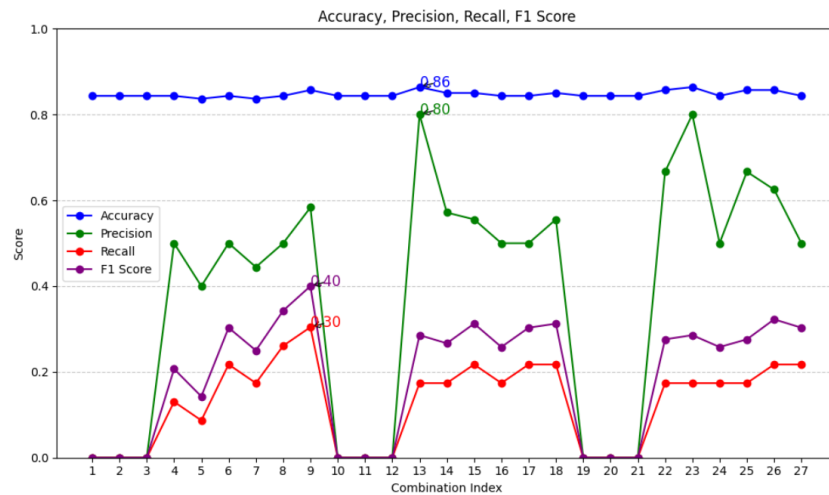
Pengujian ini menggunakan rasio *split* data 90% untuk data *training* dan 10% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut

adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.19 *Confusion matrix* kombinasi terbaik skenario D 90:10

Gambar 4.19 menunjukkan bahwa terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 123 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model dan *False Positive* (FP) sebanyak 1 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 19 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 4 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:

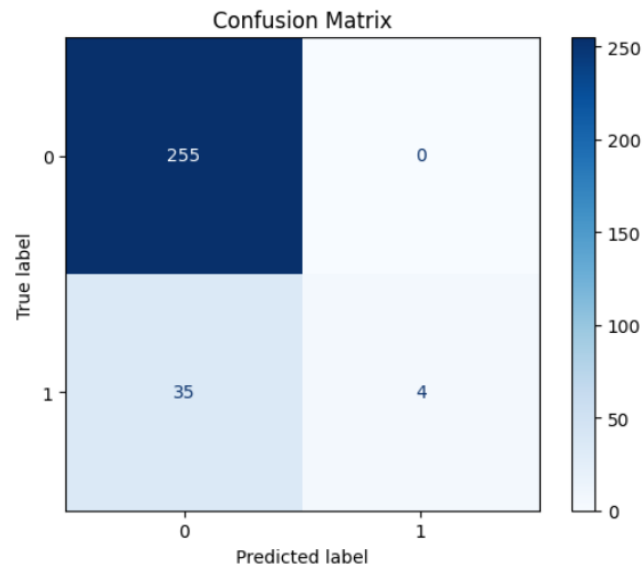


Gambar 4.20 Hasil seluruh pengujian skenario D 90:10

Gambar 4.20 menunjukkan akurasi model cenderung stabil di kisaran 84% hingga 86%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

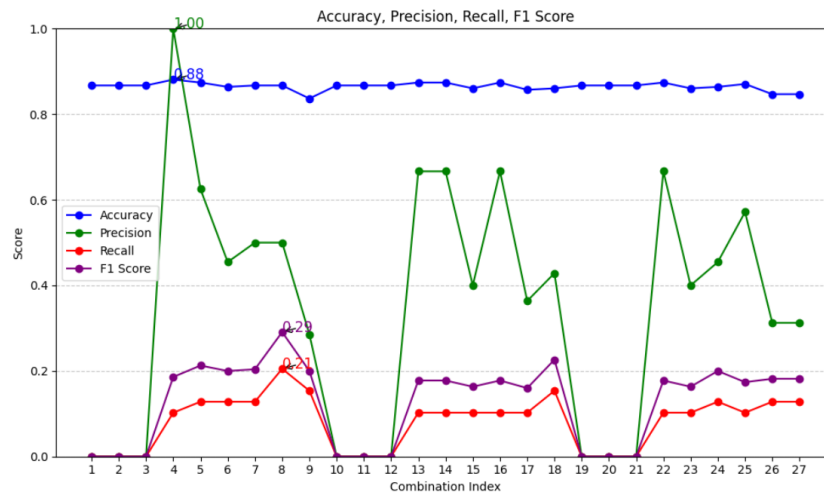
## B. Rasio 80:20

Pengujian ini menggunakan rasio *split* data 80% untuk data *training* dan 20% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.21 *Confusion matrix* kombinasi terbaik skenario D 80:20

Gambar 4.21 menunjukkan bahwa model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 255 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model dan tidak ada data *False Positive* (FP). Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 35 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 4 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:

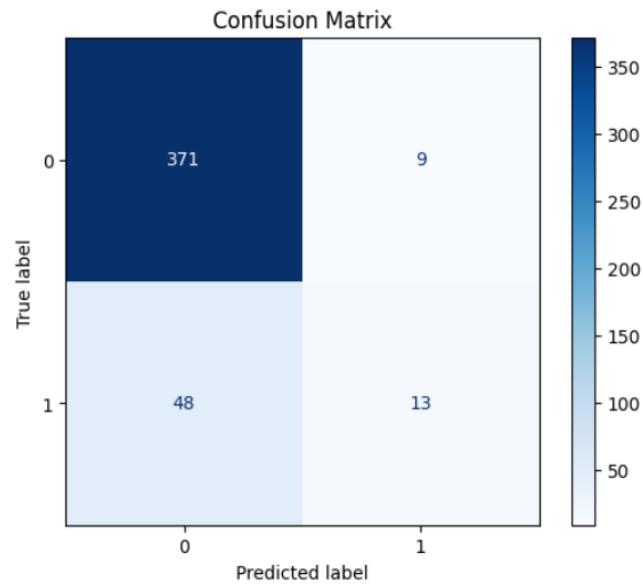


Gambar 4.22 Hasil seluruh pengujian skenario D 80:20

Gambar 4.22 menunjukkan akurasi model cenderung stabil di kisaran 85% hingga 88%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

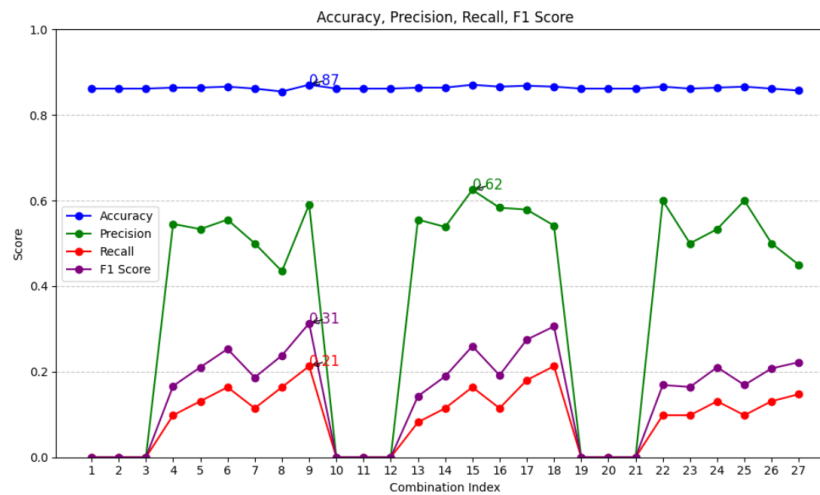
### C. Rasio 70:30

Pengujian ini menggunakan rasio *split* data 70% untuk data *training* dan 30% untuk data *testing* yang selanjutnya dilakukan *tuning hyperparameter*. Berikut adalah *confusion matrix* untuk kombinasi untuk kombinasi *hyperparameter* terbaik pada pengujian ini:



Gambar 4.23 *Confusion matrix* kombinasi terbaik skenario D 70:30

Gambar 4.23 menunjukkan bahwa model terlihat sangat baik dalam memprediksi kelas 0, dengan *True Negative* (TN) sebanyak 371 data yang berarti karyawan yang tidak mengalami *turnover* berhasil diprediksi benar oleh model dan *False Positive* (FP) sebanyak 9 data, yang berarti karyawan diprediksi *turnover* oleh model padahal sebenarnya tidak. Namun, model kurang baik dalam memprediksi kelas 1, dengan *False Negative* (FN) sebanyak 48 data, yang berarti karyawan yang sebenarnya mengalami *turnover* diprediksi sebagai tidak *turnover* dan *True Positive* (TP) hanya menghasilkan sebanyak 13 data, yang berarti karyawan yang *turnover* berhasil diprediksi dengan benar oleh model. Berikut grafik akurasi, presisi, *recall*, dan *f1-score* dari seluruh pengujian pada skenario ini:



Gambar 4.24 Hasil seluruh pengujian skenario D 70:30

Gambar 4.24 menunjukkan akurasi model cenderung stabil di kisaran 85% hingga 87%, akan tetapi presisi cenderung tidak stabil dan mengalami fluktuasi dengan beberapa kombinasi *hyperparameter* yang tinggi. *Recall* sangat bervariasi, bahkan mencapai nilai rendah pada beberapa kombinasi. Dan *f1-score* juga mengalami pola yang serupa dengan *recall*.

Setelah dilakukan serangkaian uji coba dengan melibatkan seluruh skenario penggunaan fitur, rasio *split* data, dan *tuning hyperparameter*, diperoleh hasil perbandingan performa model berdasarkan kombinasi *hyperparameter* terbaik tiap skenario. Hasil tersebut dirangkum dalam tabel 4.1 di bawah, yang juga mencakup akurasi, presisi, *recall*, dan *f1-score* untuk memberi gambaran secara keseluruhan tentang efektivitas model dalam melakukan prediksi terhadap data *turnover* karyawan.

Tabel 4.1 Hasil kombinasi *hyperparameter* terbaik tiap skenario

Skenario	Rasio Split Data	Best Index	Best Hyperparameters	Best Confusion Matrix	Accuracy	Precision	Recall	F1-score
A (Seluruh Fitur)	90:10	16	(50, 10, 5)	[[124 0] [ 19 4]]	0,87	0,93	0,58	0,61
	80:20	25	(100, 10, 5)	[[254 1] [ 34 5]]	0,88	0,85	0,56	0,57
	70:30	5	(10, 5, 10)	[[376 4] [ 54 7]]	0,86	0,75	0,55	0,56
B (5 Fitur)	90:10	12	(50, 5, 2)	[[121 3] [ 18 5]]	0,85	0,74	0,59	0,62
	80:20	0	(10, 1, 2)	[[255 0] [ 39 0]]	0,86	0,43	0,5	0,46
	70:30	3	(10, 5, 2)	[[372 8] [ 52 9]]	0,86	0,70	0,56	0,57
C (10 Fitur)	90:10	21	(100, 5, 2)	[[121 3] [ 18 5]]	0,85	0,74	0,59	0,62
	80:20	12	(50, 5, 2)	[[251 4] [ 33 6]]	0,87	0,74	0,56	0,58
	70:30	21	(100, 5, 2)	[[373 7] [ 53 8]]	0,86	0,70	0,55	0,56
D (15 Fitur)	90:10	12	(50, 5, 2)	[[123 1] [ 19 4]]	0,86	0,83	0,58	0,60
	80:20	3	(10, 5, 2)	[[255 0] [ 35 4]]	0,88	0,93	0,55	0,56
	70:30	8	(10, 10, 10)	[[371 9] [ 48 13]]	0,87	0,73	0,59	0,62

Berdasarkan tabel 4.1 dengan berbagai skenario jumlah fitur dan rasio *split* data, terlihat beberapa pola menarik yang dapat dijadikan dasar analisis performa model. Pada skenario A, yang menggunakan seluruh fitur, model cenderung memiliki akurasi dan presisi tinggi pada rasio *split* 90:10 dan 80:20, masing-masing dengan akurasi 0,87 dan 0,88. Namun, *recall* dan *F1-score* berada di bawah 0,6, menunjukkan model lebih efektif dalam mendeteksi kelas mayoritas namun kurang optimal dalam mendeteksi kelas minoritas. Rasio 70:30 menghasilkan akurasi sedikit lebih rendah (0,86) dengan presisi dan *recall* yang masih relatif rendah, menunjukkan bahwa lebih banyak data uji mungkin mengurangi kemampuan model dalam mengenali pola pada kelas minoritas.

Pada skenario B, yang menggunakan hanya 5 fitur, akurasi konsisten di angka 0,85 hingga 0,86 pada semua rasio data, namun precision dan *recall* cenderung bervariasi. Pada *split* 90:10 dan 70:30, *F1-score* masing-masing mencapai 0,62 dan 0,57. Namun, pada *split* 80:20, model menunjukkan performa kurang baik dalam *recall* dan *F1-score*, masing-masing 0,5 dan 0,46, yang menunjukkan bahwa jumlah fitur yang lebih sedikit mungkin tidak cukup untuk mempertahankan performa model secara konsisten.

Skenario C dengan 10 fitur menunjukkan hasil yang stabil pada ketiga rasio *split*, dengan akurasi berkisar di angka 0,85 hingga 0,87 dan *F1-score* mendekati atau sedikit di bawah 0,6. Kombinasi *hyperparameter* yang optimal pada skenario ini tampaknya menghasilkan keseimbangan yang lebih baik dibandingkan skenario A dan B, khususnya pada rasio 80:20 dengan akurasi 0,87 dan *F1-score* 0,58. Hasil ini menunjukkan bahwa pemilihan fitur yang tepat dapat meningkatkan kinerja model tanpa menggunakan seluruh fitur yang tersedia.

Pada skenario D dengan 15 fitur, akurasi terbaik diperoleh pada rasio 80:20 dengan nilai 0,88, presisi tinggi sebesar 0,93, namun *recall* tetap rendah pada 0,55. Hal ini serupa dengan pola pada skenario lainnya di mana presisi tinggi namun *recall* masih rendah, mengindikasikan ketergantungan pada kelas mayoritas. Rasio *split* 70:30 memberikan *F1-score* tertinggi (0,62) di antara semua skenario 15 fitur, sehingga pilihan 15 fitur dengan kombinasi rasio data dan *hyperparameter* yang tepat dapat menjadi alternatif yang untuk mempertahankan kinerja model yang seimbang.

## 4.2 Pembahasan

Berdasarkan hasil dari masing-masing skenario pengujian dengan kombinasi *hyperparameter* yang telah ditentukan, pola performa model yang dihasilkan dapat dipahami adalah penggunaan jumlah fitur yang berbeda serta perbedaan rasio *split* data memberikan pengaruh terhadap performa model prediksi. Pada skenario A, yang menggunakan seluruh fitur, model menunjukkan akurasi yang cukup tinggi, terutama pada rasio *split* 80:20, tetapi kurang optimal dalam mendeteksi kelas minoritas, terlihat dari nilai *recall* dan *f1-score* yang rendah. Hal ini dapat disebabkan karena data yang tidak seimbang antara kelas mayoritas dan minoritas.

Pada skenario B, dengan 5 fitur teratas akurasi tetap konsisten, namun ketidakstabilan presisi dan *recall* pada rasio 80:20 menunjukkan bahwa fitur yang lebih sedikit tidak dapat mempertahankan keseimbangan performa model. Pengurangan fitur menyebabkan penurunan kinerja model dalam mendeteksi variasi antar kelas, meskipun model lebih sederhana dan cepat. Skenario C, dengan 10 fitur, menunjukkan performa yang stabil di semua rasio *split* data, dibanding hanya menggunakan 5 fitur. Ini menunjukkan bahwa pemilihan 10 fitur dapat memberikan keseimbangan kinerja yang lebih optimal, karena mencakup informasi yang cukup tanpa menambah kompleksitas model secara berlebihan.

Pada skenario D, penggunaan 15 fitur menunjukkan peningkatan akurasi, terutama pada rasio *split* 80:20, dengan presisi yang lebih tinggi. Namun, model masih kurang sensitif terhadap kelas minoritas, yang tercermin dari nilai *recall* yang tetap rendah. Saat menggunakan rasio *split* 70:30, model mencapai keseimbangan

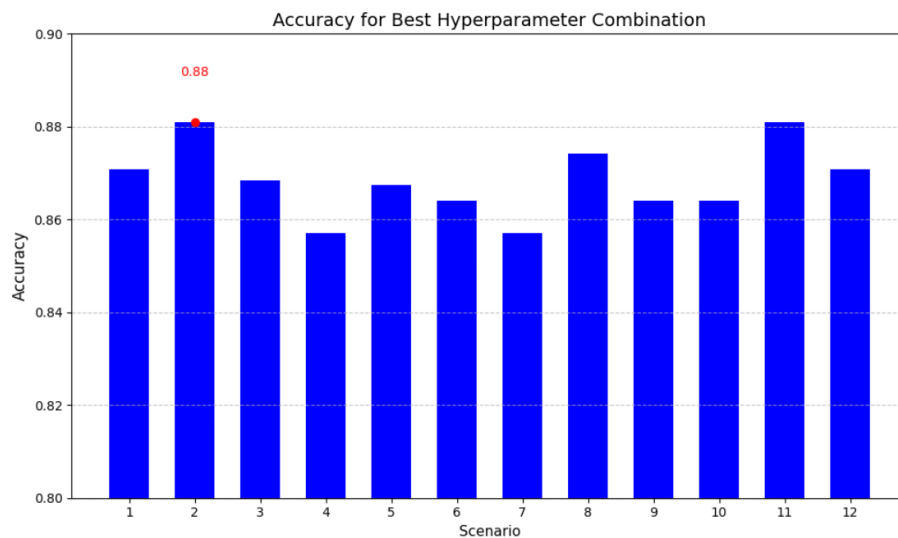
yang lebih baik antara presisi dan *recall*, meunjukkan kemampuannya untuk mendeteksi *turnover* lebih efektif ketika lebih banyak data tersedia untuk pengujian. Keseimbangan dalam *training* dan *testing* ini menjadikan model lebih mampu mengidentifikasi faktor-faktor penting dari kedua kelas, meskipun dengan tantangan dalam mendeteksi kelas minoritas secara optimal.

Hasil pengujian menunjukkan bahwa penggunaan seluruh fitur memberikan kinerja model yang terbaik. Hal ini ditunjukkan dengan adanya penurunan akurasi saat model hanya menggunakan 5, 10, atau 15 fitur. Saat menggunakan seluruh fitur, model mampu mencapai akurasi tertinggi sebesar 88% sedangkan saat menggunakan 5 fitur, model hanya mencapai akurasi tertinggi sebesar 86%. Selanjutnya, pada saat menggunakan 10 fitur, model mencapai akurasi tertinggi sebesar 87%. Dan saat menggunakan 15 fitur, model mencapai akurasi tertinggi sebesar 88% namun dengan presisi, *recall*, dan *f1-score* yang kurang seimbang pada pengujian dengan 15 fitur.

#### **4.1.1 Akurasi**

Akurasi model cenderung tetap konstan dengan rata-rata sekitar 86% untuk setiap kombinasi yang diuji. Namun, peningkatan akurasi signifikan dicapai ketika parameter *max\_depth* dan *max\_features* disesuaikan ke nilai yang lebih tinggi. Misalnya, ketika *max\_depth* diatur ke 5 atau 10 dan *max\_features* diubah menjadi 5 atau 10, model menunjukkan peningkatan performa. Hal ini menunjukkan bahwa model prediksi menjadi lebih akurat dengan peningkatan kompleksitas dan jumlah fitur yang dipertimbangkan. Penyesuaian *hyperparameter* ini memiliki peran penting dalam mengoptimalkan kinerja model, memungkinkan prediksi yang lebih

akurat. Dengan demikian, *tuning hyperparameter* yang tepat dapat meningkatkan efisiensi model dalam memprediksi *turnover* karyawan. Berikut adalah grafik perbandingan akurasi dari tiap skenario:



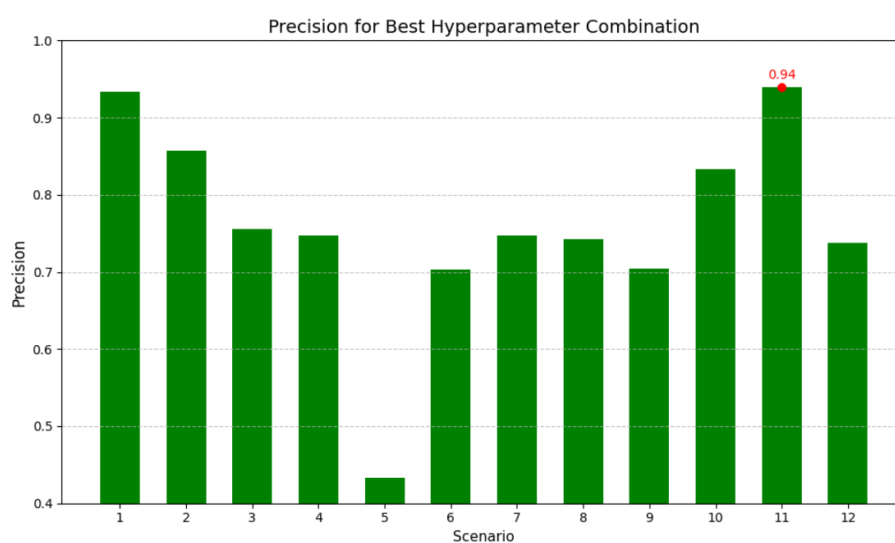
Gambar 4.25 Grafik akurasi terbaik tiap skenario

Grafik di atas menunjukkan akurasi model pada 12 kombinasi *hyperparameter* yang berbeda. Akurasi tertinggi terjadi pada indeks 2, dengan penurunan sedikit pada indeks 3 dan penurunan yang signifikan ke titik terendah pada indeks 4. Akurasi berfluktuasi secara signifikan di sepanjang kombinasi, menunjukkan bahwa skenario ke-2 dapat sangat memengaruhi performa model, dengan beberapa kombinasi menghasilkan akurasi yang relatif rendah.

#### 4.1.2 Presisi

Presisi meningkat secara signifikan saat *max\_depth* dan *max\_features* ditingkatkan. Hal ini terjadi karena nilai *max\_depth* yang rendah menyebabkan model menjadi terlalu sederhana, sehingga tidak mampu menangkap pola yang kompleks dalam data. Sebaliknya, ketika *max\_depth* dan *max\_features*

ditingkatkan, model menjadi lebih kompleks dan mampu menangkap lebih banyak informasi dari fitur-fitur yang ada. Hal ini memungkinkan model untuk membuat prediksi yang lebih akurat pada kelas positif, sehingga meningkatkan presisi secara keseluruhan. Penyesuaian *hyperparameter* ini sangat penting untuk mengoptimalkan performa model dan memastikan prediksi yang lebih andal dan akurat. Berikut adalah grafik perbandingan presisi dari tiap skenario:

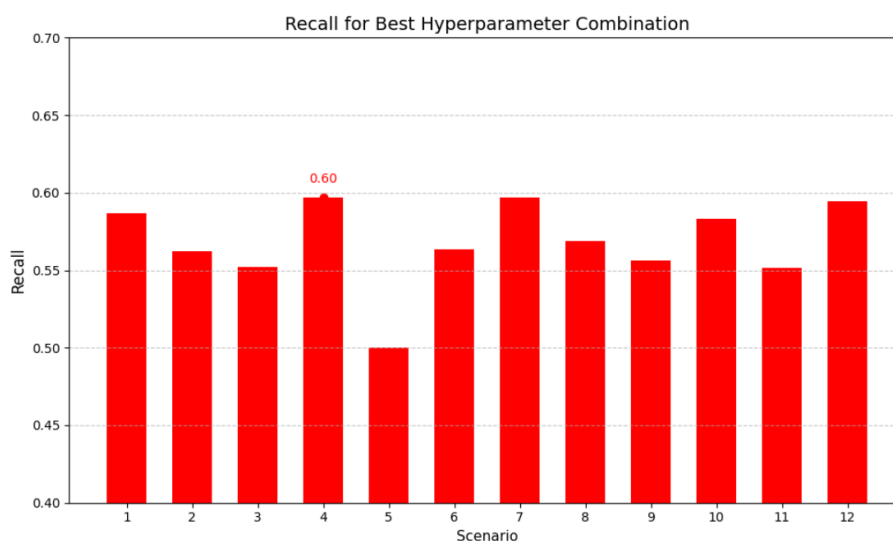


Gambar 4.26 Grafik presisi terbaik tiap skenario

Pada grafik di atas, presisi diukur pada 12 kombinasi *hyperparameter*. Presisi pada indeks pertama cukup tinggi akan tetapi mengalami penurunan pada indeks berikutnya, dan menghasilkan presisi tertinggi berada pada indeks 11. Pola ini menunjukkan bahwa meskipun beberapa kombinasi menghasilkan presisi yang tinggi, kombinasi lainnya menyebabkan penurunan, menunjukkan sensitivitas presisi terhadap perubahan *hyperparameter*.

### 4.1.3 Recall

*Recall* mampu meningkat ketika *max\_depth* ditingkatkan dan *max\_features* disesuaikan. Hal ini terjadi karena dengan *max\_depth* yang rendah, model terlalu sederhana dan tidak mampu mengenali pola-pola penting dalam data yang diperlukan untuk mendeteksi kasus positif. Sebaliknya, dengan meningkatkan *max\_depth* dan menyesuaikan *max\_features*, model menjadi lebih kompleks dan dapat menangkap lebih banyak informasi dari data. Ini memungkinkan model untuk mendeteksi lebih banyak kasus positif, sehingga meningkatkan *recall* secara keseluruhan. Penyesuaian ini menunjukkan pentingnya konfigurasi *hyperparameter* yang tepat untuk meningkatkan kemampuan model dalam mendeteksi dan mengklasifikasikan data dengan benar. berikut adalah grafik perbandingan *recall* dari tiap skenario:



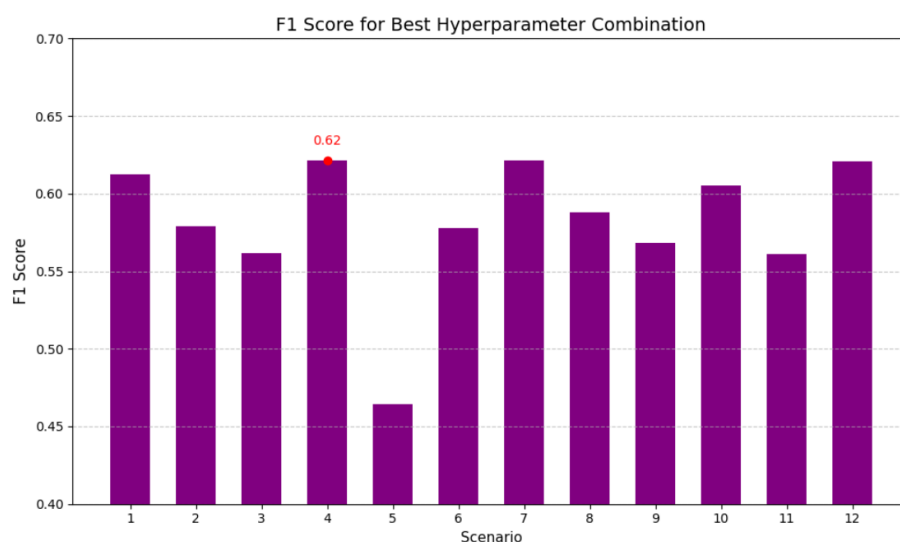
Gambar 4.27 Grafik *recall* terbaik tiap skenario

Grafik di atas menggambarkan *recall* pada 12 kombinasi *hyperparameter*. *Recall* bervariasi secara signifikan, menekankan bahwa pemilihan *hyperparameter*

memainkan peran penting dalam mempertahankan kemampuan model untuk mengidentifikasi semua kasus yang relevan.

#### 4.1.4 *F1-score*

*F1-score* mampu meningkat secara signifikan pada kombinasi dengan *max\_depth* dan *max\_features* yang ditingkatkan menjadi 5 atau 10. Hal ini terjadi karena nilai *max\_depth* yang rendah menyebabkan model tidak mampu menangkap pola yang kompleks dalam data, sehingga gagal dalam memprediksi kasus positif dengan benar. Dengan meningkatkan *max\_depth* dan menyesuaikan *max\_features*, model menjadi lebih kompleks dan mampu memanfaatkan lebih banyak informasi dari data. Perubahan meningkatkan presisi maupun *recall*, yang selanjutnya meningkatkan *F1-score*. Penyesuaian *hyperparameter* sangat penting untuk mengoptimalkan performa model, sehingga prediksi yang dilakukan menjadi lebih akurat.



Gambar 4.28 Grafik *f1-score* terbaik tiap skenario

Setelah melakukan analisis pada berbagai kombinasi *hyperparameter*, ditemukan bahwa kombinasi (100, 5, 5) dengan seluruh fitur dan rasio *split* data sebesar 80:20 memberikan kinerja terbaik karena menghasilkan akurasi tertinggi sebesar 88%, dan memiliki keseimbangan yang lebih baik pada presisi, *recall*, dan *f1-score* jika dibandingkan dengan kombinasi lain. Penggunaan berbagai rasio *split* data memiliki dampak yang berbeda-beda terhadap performa model. Rasio 90:10 memberikan hasil yang lebih rendah terhadap akurasi dan evaluasi model karena jumlah data *testing* yang terbatas, meskipun data *training* cukup besar. Rasio 80:20 memberikan keseimbangan optimal antara data *training* dan data *testing*, memungkinkan evaluasi yang lebih akurat dengan variasi data uji yang lebih representatif, menghasilkan akurasi tertinggi di sebagian besar skenario. Rasio 70:30, meskipun memberikan hasil yang masih baik, memiliki sedikit kekurangan karena data latih yang lebih kecil, yang memengaruhi kemampuan model untuk belajar secara maksimal.

Pemilihan jumlah fitur memiliki pengaruh besar terhadap performa model. Penggunaan seluruh fitur memberikan akurasi dan presisi tinggi, terutama pada rasio 80:20, karena model mampu memanfaatkan seluruh informasi yang tersedia dalam data. Pengurangan jumlah fitur menjadi 5 dan 10 menyebabkan penurunan akurasi dan presisi karena hilangnya fitur-fitur penting yang relevan. Penggunaan 15 fitur memberikan keseimbangan yang lebih baik dibandingkan 5 atau 10 fitur, meskipun performa untuk keseimbangan presisi, *recall*, dan *f1-score* masih di bawah saat menggunakan seluruh fitur.

Penyesuaian nilai *hyperparameter* juga memiliki pengaruh penting terhadap performa model. Penggunaan  $n\_estimators = 100$  memberikan stabilitas dan akurasi yang lebih tinggi dibandingkan 50, terutama pada rasio 80:20. Kedalaman pohon optimal ditemukan pada  $max\_depth = 5$ , yang menjaga keseimbangan antara kompleksitas dan generalisasi, sementara kedalaman yang lebih dalam berisiko menyebabkan *overfitting*. Penggunaan 2 fitur maksimum ( $max\_features = 2$ ) memberikan hasil terbaik, sedangkan  $max\_features = 5$  cenderung membuat model lebih kompleks dan mengurangi *precision*.

Telah dilakukan uji coba pada seluruh fitur menggunakan data asli dan data setelah diurutkan fiturnya menggunakan PCC. Uji coba dilakukan sebanyak 5 kali dan model *Random Forest* menunjukkan hasil evaluasi yang konsisten pada skenario dengan menggunakan seluruh fitur baik sebelum maupun setelah dilakukan seleksi fitur berdasarkan nilai PCC. Hal ini mengindikasikan bahwa urutan fitur tidak memengaruhi performa model jika menggunakan seluruh informasi dalam data. Berikut adalah tabel hasil evaluasi model pada seluruh fitur tanpa dan dengan menggunakan PCC:

Tabel 4. 2 Hasil uji coba seluruh fitur tanpa PCC

Skenario	Rasio Split Data	<i>Best Index</i>	<i>Best Hyperparameters</i>	<i>Best Confusion Matrix</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
Seluruh fitur tanpa PCC	90:10	16	(50, 10, 5)	$\begin{bmatrix} 124 & 0 \\ 19 & 4 \end{bmatrix}$	0,87	0,93	0,58	0,61
	80:20	25	(100, 10, 5)	$\begin{bmatrix} 254 & 1 \\ 34 & 5 \end{bmatrix}$	0,88	0,85	0,56	0,57
	70:30	5	(10, 5, 10)	$\begin{bmatrix} 376 & 4 \\ 54 & 7 \end{bmatrix}$	0,86	0,75	0,55	0,56

Tabel 4. 3 Hasil uji coba seluruh fitur dengan PCC

Skenario	Rasio Split Data	<i>Best Index</i>	<i>Best Hyperparameters</i>	<i>Best Confusion Matrix</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
Seluruh fitur dengan PCC	90:10	16	(50, 10, 5)	$\begin{bmatrix} 124 & 0 \\ 19 & 4 \end{bmatrix}$	0,87	0,93	0,58	0,61
	80:20	25	(100, 10, 5)	$\begin{bmatrix} 254 & 1 \\ 34 & 5 \end{bmatrix}$	0,88	0,85	0,56	0,57
	70:30	5	(10, 5, 10)	$\begin{bmatrix} 376 & 4 \\ 54 & 7 \end{bmatrix}$	0,86	0,75	0,55	0,56

Tabel 4.2 adalah hasil uji coba seluruh fitur tanpa PCC dan tabel 4.3 adalah hasil uji coba seluruh fitur dengan PCC. Terlihat pada kolom *Best Index*, *Best Hyperparameters*, dan *Best Confusion Matrix* pada kedua tabel tersebut menunjukkan hasil yang sama. Sehingga akurasi, presisi, recall, dan f-1 score yang dihasilkan juga sama. Hal ini menunjukkan bahwa dengan ataupun tanpa PCC pada tahap preprocessing, hasil prediksi tetap konsisten ketika menggunakan seluruh fitur. Model *Random Forest* mampu memanfaatkan informasi dari seluruh data secara optimal tanpa memperhatikan urutan fitur tertentu.

### 4.3 Integrasi Islam

Integrasi prinsip-prinsip Islam sangat relevan dalam upaya meningkatkan strategi retensi yang tepat dalam mengurangi resiko *turnover*. Penerapan nilai-nilai seperti kejujuran, keadilan, dan tanggungjawab dapat memberikan landasan moral yang kuat bagi perusahaan dalam mengelola hubungan dengan karyawan, sebagaimana dijelaskan dalam QS. An-Nisa ayat 58:

إِنَّ اللَّهَ يَأْمُرُكُمْ أَنْ تُؤَدُّوا الْأَمَانَاتِ إِلَىٰ أَهْلِهَا وَإِذَا حَكَمْتُمْ بَيْنَ النَّاسِ أَنْ تَحْكُمُوا بِالْعَدْلِ ۗ إِنَّ اللَّهَ نِعِمَّا يَعِظُكُمْ بِهِ ۗ إِنَّ اللَّهَ كَانَ سَمِيعًا بَصِيرًا

"*Sesungguhnya Allah menyuruh kamu menyampaikan amanat kepada yang berhak menerimanya, dan (menyuruh kamu) apabila menetapkan hukum di antara manusia supaya kamu menetapkan dengan adil. Sesungguhnya Allah memberi pengajaran yang sebaik-baiknya kepadamu. Sesungguhnya Allah adalah Maha Mendengar lagi Maha Melihat*". (QS An-Nisa:58)

Berdasar Tafsir Ibnu Katsir, ayat di atas memiliki kaitan dengan manajemen karyawan agar perusahaan mampu bertanggungjawab dan berlaku adil terhadap karyawan (*Tafsir Surah An-Nisa Ayat 58*, n.d.). Integrasi nilai ini dengan penelitian berjudul "Prediksi *Turnover* Karyawan Menggunakan Algoritma *Random Forest*" sangat relevan, terutama dalam hal tanggungjawab perusahaan untuk senantiasa memelihara lingkungan kerja yang adil dan transparan. Melalui penelitian ini, perusahaan dapat memprediksi kemungkinan karyawan yang akan meninggalkan perusahaan dan mengidentifikasi faktor-faktor penyebab *turnover*.

Tafsir Ibnu Katsir menyoroti bahwa amanah dalam manajemen karyawan mencakup tanggung jawab untuk memberikan hak-hak karyawan, termasuk lingkungan kerja yang sehat, upah yang adil, dan perlakuan yang setara. Dengan menggunakan metode *Random Forest* untuk memprediksi *turnover*, perusahaan dapat proaktif dalam mengidentifikasi masalah internal yang mungkin tidak terlihat secara langsung tetapi berdampak signifikan terhadap kepuasan dan retensi karyawan.

Misalnya, melalui analisis data, perusahaan dapat menemukan bahwa beban kerja yang berlebihan, kurangnya peluang pengembangan karir, atau masalah komunikasi antar tim adalah faktor utama yang menyebabkan *turnover*.

Mengetahui hal ini, perusahaan dapat mengambil langkah-langkah konkret untuk memperbaiki kondisi tersebut, seperti menyesuaikan beban kerja, menyediakan pelatihan dan pengembangan karir, serta memperbaiki sistem komunikasi internal.

Selanjutnya, dalam penggalan QS. Al-Baqarah ayat 286 juga relevan dalam konteks ini. Ayatnya berbunyi:.

لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا

"Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya..."  
(QS Al-Baqarah:256)

Tafsir Ibnu Katsir menjelaskan bahwa QS. Al-Baqarah ayat 286 menunjukkan betapa Allah Maha Bijaksana dan Maha Pengasih, yang tidak membebani manusia melebihi kemampuan mereka (*Tafsir Surah Al-Baqarah Ayat 286*, 2023). Amanah yang dapat diambil adalah dalam manajemen karyawan, perusahaan harus memperhatikan batas kemampuan karyawan dan memastikan bahwa beban kerja mereka sesuai dengan kapasitas masing-masing individu. Hal ini penting untuk membangun lingkungan kerja yang kondusif dan meminimalkan risiko *turnover* karyawan.

Penelitian yang menggunakan metode *Random Forest* untuk memprediksi *turnover* dapat membantu perusahaan dalam mengidentifikasi karyawan yang berisiko tinggi dan faktor-faktor yang menyebabkan mereka meninggalkan perusahaan. Dengan memahami penyebab *turnover*, seperti stres kerja, ketidakpuasan terhadap kompensasi, atau kurangnya kesempatan pengembangan karir, perusahaan dapat mengambil langkah-langkah yang tepat untuk mengatasi masalah ini. Langkah-langkah tersebut termasuk menyediakan program

kesejahteraan karyawan, menyesuaikan beban kerja, atau meningkatkan komunikasi antara manajemen dan karyawan.

Prinsip keadilan juga ditekankan dalam ayat ini dan diperkuat oleh Tafsir Ibnu Katsir. Dalam membuat keputusan berdasarkan hasil prediksi *turnover*, perusahaan harus memastikan bahwa kebijakan yang diterapkan adil dan tidak diskriminatif. Misalnya, jika analisis data menunjukkan bahwa kelompok karyawan tertentu, seperti karyawan junior atau karyawan perempuan, lebih cenderung mengalami *turnover*, perusahaan harus mengembangkan kebijakan yang mendukung kelompok ini tanpa merugikan kelompok lainnya. Ini bisa mencakup program mentoring, kebijakan kerja fleksibel, atau kesempatan pengembangan karir yang lebih baik.

Dengan menerapkan prinsip-prinsip kejujuran, keadilan, dan tanggung jawab yang diajarkan dalam ayat-ayat Al-Qur'an, perusahaan dapat menciptakan lingkungan kerja yang lebih adil, harmonis, dan produktif. Prinsip-prinsip ini tidak hanya membantu dalam mengurangi *turnover* karyawan tetapi juga meningkatkan kesejahteraan karyawan secara keseluruhan. Dengan demikian, penelitian prediksi *turnover* karyawan menggunakan metode *Random Forest* yang diintegrasikan dengan nilai-nilai Islam dapat menjadi pedoman dalam membangun manajemen karyawan yang beretika dan efektif.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Dari hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa penggunaan data historis karyawan dan algoritma *Random Forest* mampu menghasilkan prediksi *turnover* karyawan yang cukup baik. Model yang dibangun berhasil mencapai akurasi tertinggi sebesar 88% pada skenario A menggunakan seluruh fitur dengan rasio 80:20. Kombinasi *hyperparameter* terbaik yaitu 100 untuk *n\_estimators*, 5 untuk *max\_depth*, dan 2 untuk *max\_features*.

Penggunaan data historis dan proses seleksi fitur menggunakan metode *Pearson Correlation Coefficient* (PCC) membuat model dapat mengenali pola dan faktor-faktor yang paling relevan dengan prediksi *turnover*. Dengan demikian, perusahaan dapat memperoleh wawasan yang lebih mendalam terkait karyawan yang berisiko untuk meninggalkan perusahaan. Penggunaan seluruh fitur memberikan hasil yang lebih baik, karena model mampu menggali dan memanfaatkan seluruh informasi yang tersedia dalam data. Jika dibandingkan dengan hanya menggunakan 5 fitur, model mengalami penurunan akurasi sebesar 2% pada rasio 80:20, yang menunjukkan bahwa pemilihan fitur yang terbatas membuat model menjadi kehilangan informasi dalam data sehingga mengalami penurunan performa. Demikian juga saat menggunakan 10 fitur, model mengalami penurunan akurasi sebesar 1% jika dibandingkan dengan menggunakan seluruh fitur. Saat menggunakan 15 fitur, model juga menunjukkan adanya penurunan

akurasi sebesar 1% dibanding menggunakan seluruh fitur. Jadi meskipun model dengan 10 dan 15 fitur masih menghasilkan performa yang baik, tetapi adanya penurunan akurasi menunjukkan bahwa adanya pembatasan fitur membatasi kemampuan model dalam memanfaatkan informasi dalam data, sehingga membuat performanya menurun.

Selain itu, telah dilakukan uji coba pada seluruh fitur menggunakan data asli dan data setelah diurutkan fiturnya berdasarkan PCC sebanyak 5 kali. Model *Random Forest* menunjukkan hasil evaluasi yang konsisten baik sebelum maupun setelah diurutkan fiturnya menggunakan PCC. Hal ini mengindikasikan bahwa urutan fitur tidak memengaruhi performa model jika menggunakan seluruh informasi dalam data. Hasilnya, penggunaan PCC tidak memberikan pengaruh yang signifikan pada performa model jika seluruh fitur dalam dataset digunakan.

Dengan hasil prediksi ini, perusahaan dapat mengambil langkah-langkah konkret untuk menekan angka *turnover*, misalnya dengan mengidentifikasi faktor-faktor utama yang menyebabkan karyawan berisiko tinggi untuk meninggalkan perusahaan. Strategi yang bisa diterapkan seperti mengurangi beban kerja atau memberikan intensif jika melakukan kerja lembur. Selain itu, perusahaan dapat memperkuat loyalitas karyawan lama dengan menawarkan peluang pengembangan karier yang lebih jelas, seperti memberikan pelatihan, promosi, atau rotasi jabatan untuk menghadapi kebosanan. Jika karyawan berprestasi, dapat diberi kesempatan untuk memegang posisi yang lebih senior seperti menjadi manajer atau kepala departemen. Hal ini memberikan tantangan baru dan peluang untuk pertumbuhan yang lebih lanjut dalam karier mereka. Kombinasi antara data historis, algoritma

*machine learning*, dan pendekatan seleksi fitur dapat menjadi alat yang efektif dalam merancang strategi retensi yang tepat guna menekan angka *turnover*.

## 5.2 Saran

Berdasarkan proses dan hasil penelitian ini, peneliti menyadari bahwa penelitian ini masih banyak kekurangan. Maka dari itu, diharapkan pada penelitian selanjutnya agar dapat melakukan peningkatan dan perbaikan agar hasil yang didapatkan menjadi lebih baik. Beberapa saran untuk penelitian selanjutnya yaitu:

1. Menggunakan dataset yang seimbang antara kelas positif dan negatif (*balanced*) sehingga model dapat memprediksi masing-masing kelas dengan lebih baik
2. Mengeksplorasi metode seleksi fitur yang lain, yang mampu menangkap hubungan linier maupun non-linier antara fitur-fitur dan target.
3. Mengeksplorasi lebih lanjut pengaruh *hyperparameter* lain yang belum dieksplorasi secara mendalam pada penelitian ini, seperti *min\_sample\_split*, *min\_sample\_leaf*, dan lain-lain.

## DAFTAR PUSTAKA

- Abiyyu, A. S., & Lhaksana, K. M. (2021). Perbandingan Metode Seleksi Fitur untuk Mengoptimasi Model Support Vector Machine dalam Memprediksi Turnover Pegawai. *e-Proceeding of Engineering*, 10, 1921.
- Amna, S. W., Sudipa, I. G. I., Putra, T. A. E., Wahidin, A. S., Wardhani, A. K., Heryana, N., Indriyani, T., & Santoso, L. W. (2023). *Data Mining*. PT GLOBAL EKSEKUTIF TEKNOLOGI.
- Ardi Subakti, Adam Irnandito Syahrizal, & Eva Dwi Kurniawan. (2023). Analisis Penyebab Turnover Intention Karyawan Pada Perusahaan Dalam Novel Resign Karya Almira Bastari. *Jurnal Manajemen Kreatif dan Inovasi*, 2(1), 251–260. <https://doi.org/10.59581/jmki-widyakarya.v2i1.2187>
- Aryanto, C., Palit, H. N., & Gunawan, A. (2022). *Prediksi Peringkat Mingguan Lagu Pada Spotify Amerika Serikat Menggunakan Multiple Charts Dataset Dengan Berbagai Metode*. <https://publication.petra.ac.id/index.php/teknik-informatika/article/view/12796>
- Budun, M., Amberi, M., & Rahmawati, E. (2021). Turnover pada PT. Jasapower Indoneisa. *Jurnal Bisnis dan Pembangunan*, 10(2), 38. <https://doi.org/10.20527/jbp.v10i2.10958>
- Derisma, D. (2020). Perbandingan Kinerja Algoritma untuk Prediksi Penyakit Jantung dengan Teknik Data Mining. *Journal of Applied Informatics and Computing*, 4(1), 84–88. <https://doi.org/10.30871/jaic.v4i1.2152>
- Hansen, J. (2024). *2023 Average Employee Turnover Rates by Industry*. <https://www.award.co/blog/employee-turnover-rates>
- Iacobello, G., Ridolfi, L., & Scarsoglio, S. (2021). A review on turbulent and vortical flow analyses via complex networks. *Physica A: Statistical Mechanics and Its Applications*, 563, 125476. <https://doi.org/10.1016/j.physa.2020.125476>
- Iskandar, Y. C., & Rahadi, D. R. (2021). Strategi Organisasi Penanganan Turnover Melalui Pemberdayaan Karyawan. *Solusi*, 19(1), 102. <https://doi.org/10.26623/slsi.v19i1.3003>
- Junus, C. Z. V., Tarno, T., & Kartikasari, P. (2023). Klasifikasi Menggunakan Metode Support Vector Machine Dan Random Forest Untuk Deteksi Awal Risiko Diabetes Melitus. *Jurnal Gaussian*, 11(3), 386–396. <https://doi.org/10.14710/j.gauss.11.3.386-396>
- Krisna, M. (2022, January 4). Turnover Adalah Masalah Mengerikan Perusahaan, Berikut Pengendaliannya! *Weefer*. <https://www.weefer.co.id/2022/01/pengertian-turnover-dan-pengendaliannya/>

- Manurung, D. D. E., Sandi, F., Akbardipura, F., Ashfahan, H., & Prasvita, D. S. (2021). Prediksi Pengunduran Diri Karyawan Perusahaan “Y” Menggunakan Random Forest. *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*.
- Nawangsih, I., & Fauziah, S. (2021). Prediksi Pengangkatan Karyawan Dengan Metode Algoritma C5.0 (Studi Kasus Pt. Mataram Cakra Buana Agung). *Pelita Teknologi*, 16(2), 24–33. <https://doi.org/10.37366/pelitatekno.v16i2.672>
- Normawati, D., & Prayogi, S. A. (2021). *Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter*. 5. <https://tunasbangsa.ac.id/ejurnal/index.php/jsakti>
- Pratiwi, W. N., Komariah, K., & Jhoansyah, D. (2020). Turnover Intention Berdasarkan Retensi Karyawan dan Insentif. *BUDGETING: Journal of Business, Management and Accounting*, 2(1), 313–324. <https://doi.org/10.31539/budgeting.v2i1.1760>
- Romadloni, N. T. & Hilman F Pardede. (2019). Seleksi Fitur Berbasis Pearson Correlation Untuk Optimasi Opinion Mining Review Pelanggan. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 3(3), 505–510. <https://doi.org/10.29207/resti.v3i3.1189>
- Sari, D., & Susanto, S. (2019). Mengungkap Tingginya Turnover Intention PT. WBS Semarang. *Solusi*, 17(2). <https://doi.org/10.26623/.v17i2.1462>
- Sari, S. F., & Lhaksmana, K. M. (2022). Employee Attrition Prediction Using Feature Selection with Information Gain and Random Forest Classification. *Journal of Computer System and Informatics (JoSYC)*, 3(4), 410–419. <https://doi.org/10.47065/josyc.v3i4.2099>
- Sekar Setyaningtyas, Indarmawan Nugroho, B., & Arif, Z. (2022). Tinjauan Pustaka Sistematis Pada Data Mining: Studi Kasus Algoritma K-Means Clustering. *Jurnal Teknoif Teknik Informatika Institut Teknologi Padang*, 10(2), 52–61. <https://doi.org/10.21063/jtif.2022.V10.2.52-61>
- Suliztia, M. L. (2020). *Penerapan Analisis Random Forest Pada Prototype Sistem Prediksi Harga Kamera Bekas Menggunakan Flask*.
- Surat An-Najm Ayat 41—Qur'an Tafsir Perkata*. (2024). <https://quranhadits.com/quran/53-an-najm/an-najm-ayat-41/>
- Syamsiah, N. O., & Purwandani, I. (2021). Penerapan Ensemble Stacking untuk Peramalan Laba Bersih Bank Syariah Indonesia (BSI). *Building of Informatics, Technology and Science (BITS)*, 3(3), 295–301. <https://doi.org/10.47065/bits.v3i3.1017>
- Tafsir Surah Al-Baqarah Ayat 286*. (2023). <https://tafsirweb.com/1052-surat-al-baqarah-ayat-286.html>

*Tafsir Surah An-Nisa Ayat 58.* (n.d.). Retrieved September 23, 2024, from <https://tafsirweb.com/1590-surat-an-nisa-ayat-58.html>

Yuliani, R., & Abdi, M. (2023). *Faktor-Faktor yang Memengaruhi Turnover Intention Intention Karyawan pada Fahira Hotel Bukittinggi.* XVII. <https://jurnal.umsb.ac.id/index.php/menarailmu/article/viewFile/4582/pdf>