Edu Komputika 10 (2) (2023)

Edu Komputika Journal

http://journal.unnes.ac.id/sju/index.php/edukom

Klasifikasi Perputaran Karyawan Perusahaan Menggunakan Algoritma *Random Forest* dan *Random Over-sampling*

Dede Kurniadi✉, Fitri Nuraeni, Nadhif Faturrohman, dan Asri Mulyani

Program Studi Teknik Informatika, Jurusan Ilmu Komputer, Institut Teknologi Garut, Indonesia

Info Artikel

Sejarah Artikel:
Diterima: 24 Agustus 2023
Direvisi: 22 Juni 2024
Disetujui: 18 Juli 2024

Keywords:
Karyawan, Klasifikasi, Machine Learning, Perputaran Random Forest, Random Over-Sampling

Abstrak

Pergantian karyawan merupakan permasalahan yang berat dalam suatu perusahaan, karena pergantian karyawan dapat menyebabkan kinerja perusahaan menurun akibat kekurangan karyawan. Penelitian ini bertujuan untuk membangun model untuk mengklasifikasikan apakah karyawan akan meninggalkan perusahaan atau tidak untuk mencegah pergantian karyawan. Metode yang digunakan dalam penelitian ini adalah *Machine Learning Life Cycle* (MLLC). Model dibangun menggunakan algoritma Random Forest dan Random Over-sampling untuk mengatasi data yang tidak seimbang dengan rasio pembagian data untuk data pelatihan sebesar 90% dan data pengujian sebesar 10%. Selain itu untuk mengetahui kinerja model yang dibangun dilakukan evaluasi dengan menggunakan *Confusion Matrix* dan kurva AUC-ROC. Hasil penelitian ini menunjukkan bahwa model yang dibangun berdasarkan hasil evaluasi mempunyai kinerja yang sangat baik dan hampir sempurna, dengan nilai akurasi sebesar 99,8%, recall sebesar 100%, dan presisi sebesar 99,6%. Hanya terdapat 4 dari 2000 data pengujian yang tidak diklasifikasikan dengan benar, dengan nilai AUC yang dihasilkan sebesar 99,8%, sehingga model termasuk dalam kategori Excellent berdasarkan nilai AUC.

Abstract

Employee turnover is a severe problem in a company because employee turnover can cause company performance to decline due to a lack of employees. This research aims to build a model to classify whether or not employees will leave the company to prevent employee turnover. The method used in this research is the Machine Learning Life Cycle (MLLC). The model was built using the Random Forest algorithm and Random Over-sampling to overcome imbalanced data with a data sharing ratio for training data of 90% and testing data of 10%. Apart from that, to determine the performance of the model built, an evaluation was carried out using the Confusion Matrix and the AUC-ROC curve. This research shows that the model built based on the evaluation results has excellent performance and is almost perfect, with an accuracy value of 99.8%, recall of 100%, and precision of 99.6%. Only 4 data out of 2000 test data were not classified correctly, with the resulting AUC value being 99.8%. The model is included in the Excellent category based on the AUC value.

✉ Alamat korespondensi:
Institut Teknologi Garut
Jl. Mayor Syamsu No. 1, Kabupaten Garut, 44151
E-mail: dede.kurniadi@itg.ac.id

ISSN 2252-6811
E-ISSN 2599-297X

PENDAHULUAN

Berkembangnya teknologi secara tidak langsung dapat mempengaruhi sektor industri. Penggunaan teknologi dapat memberikan dampak positif pada perusahaan dengan membantu meningkatkan kinerja perusahaan. Namun, dengan menerapkan teknologi yang baru dapat menyebabkan masalah bagi suatu perusahaan yang sedang berjalan, seperti menerapkan perubahan teknologi pada suatu perusahaan yang menyebabkan harus dilakukannya pembenahan secara luas akan membutuhkan waktu yang lama, sehingga perusahaan harus berhenti dan kesulitan lainnya seperti biaya yang tidak sedikit (Brennan, 2020). Seiring berkembangnya teknologi, maka perusahaan akan mengalami perputaran karyawan yang mana hal ini dapat menyebabkan masalah yang cukup serius bagi perusahaan.

Perputaran karyawan dapat menyebabkan masalah yang cukup fatal bagi suatu perusahaan jika tidak diatasi dengan baik. Dengan terjadinya perputaran karyawan pada suatu perusahaan dapat memberikan efek yang negatif pada kinerja dan pendapat perusahaan (Al-suraihi, Samikon, Al-suraihi, & Ibrahim, 2021). Selain itu, akan timbul masalah ketika mengganti karyawan pada suatu perusahaan, seperti karyawan yang baru mungkin saja tidak melakukan pekerjaan dengan efisien jika dibandingkan dengan karyawan sebelumnya, karyawan yang baru mungkin saja tidak dilatih dengan benar untuk pekerjaanya sehingga membutuhkan waktu yang lama untuk menyesuaikan diri dengan tempat kerjanya serta adanya kemungkinan perbedaan budaya atau kebiasaan pada karyawan baru dan karyawan sebelumnya sehingga hal ini dapat mempengaruhi pada lambatnya performa kerja (Habib, Sheikh, & Nabi, 2018). Perputaran karyawan dapat terjadi dari beberapa faktor dan salah satu faktornya yaitu kepuasan kerja. Seperti pada penelitian yang dilakukan oleh (Ekhsan, 2019), kepuasan kerja memiliki pengaruh yang negatif sehingga semakin tinggi tingkat kepuasan kerja maka semakin kecil kemungkinan keluar.

Berdasarkan data dari (Zulfiyandi et al., 2021), bahwa ini banyak sekali karyawan di Indonesia yang dikeluarkan dari suatu perusahaan, khususnya pada rentang waktu Februari 2020 sampai dengan Februari 2021 terdapat sekitar 1,95 juta karyawan atau tenaga kerja yang di PHK sehingga hal ini dapat menyebabkan masalah pada perusahaan tersebut karena dengan banyaknya tenaga kerja atau karyawan yang PHK, perusahaan akan kewalahan dalam menjalankan bisnisnya karena berkurangnya karyawan.

Penelitian sebelumnya mengenai penetapan dan perputaran karyawan beberapa sudah dilakukan seperti penelitian yang sudah dilakukan oleh (Khera & Divya, 2019), (Setianto & Jatikusumo, 2020), (Kinoto et al., 2020), (Zhang, Xu, Cheng, Chao, & Zhao, 2018), (Ahmed, 2021) dan (Alexandrio, Susanti, & Aflaha, 2020). Namun, penelitian-penelitian tersebut belum ada yang menggunakan algoritma yang akan digunakan pada penelitian ini dan memiliki tujuan penelitian yang berbeda-beda seperti pada penelitian (Setianto & Jatikusumo, 2020) yang membahas mengenai perbandingan algoritma untuk mencari algoritma mana yang terbaik antara *Decision Tree* dan *Naïve Bayes*. Selain itu, pada penelitian tersebut terdapat masalah yang tidak diatasi yaitu dataset yang tidak seimbang atau *imbalanced* yang mana hal ini dapat menyebabkan model yang dibuat mungkin saja mendapatkan nilai akurasi yang tinggi, nilai *recall* dan *precision* yang kecil (Ashraf, Saleem, Ahmed, Aslam, & Muhammad, 2020) seperti pada penelitian (Khera & Divya, 2019) diketahui bahwa atribut yang dijadikan label atau kelas tidak seimbang, yaitu memiliki perbandingan 83,87% dan 16,13% dari 1650 data.

Penelitian yang dilakukan oleh (Khera & Divya, 2019) membahas mengenai pembuatan model *machine learning* untuk mengatasi perputaran karyawan di India dengan menggunakan algoritma *Support Vector Machine* (SVM) dan hasil evaluasi dari model tersebut yaitu memiliki nilai akurasi sebesar 0,85 atau 85%. Penelitian selanjutnya yaitu membahas mengenai perbandingan algoritma *Decision Tree* dan *Naive Bayes* yang dilakukan oleh (Setianto & Jatikusumo, 2020) yang sebelum dilakukan perbandingan, dataset akan dilakukan *data mapping* menggunakan algoritma *K-Means* dan hasilnya *Decision Tree* unggul dari *Naïve Bayes* dengan nilai akurasi sebesar 91,69% untuk *Decision Tree* dan 77,87% untuk *Naïve Bayes*. Penelitian selanjutnya yaitu penelitian yang dilakukan oleh (Kinoto et al., 2020) yang membahas mengenai pembuatan model *machine learning* dengan algoritma *Logistic Regression* dan hasilnya, model yang dibuat memiliki nilai akurasi sebesar 64,40%. Selanjutnya, penelitian yang dilakukan oleh (Zhang, Xu, Cheng, Chao, & Zhao, 2018) membahas mengenai penggunaan teknik *machine learning* dalam memilah karakteristik dari perputaran karyawan menggunakan dataset IBM HR Analytics. Dengan menggunakan algoritma *Logistic Regression*, model yang dibuat memiliki nilai akurasi sebesar 87,2% serta menganalisis nilai korelasi dari setiap fitur atau kolom pada dataset

yang digunakan. Dan penelitian selanjutnya yaitu penelitian yang membahas mengenai pembuatan model data mining menggunakan algoritma *Neural Network* yang dilakukan oleh (Ahmed, 2021) dan hasil evaluasi dari model yang dibuat yaitu model mendapatkan nilai akurasi sebesar 84% dan nilai AUC (ROC) sebesar 74%.

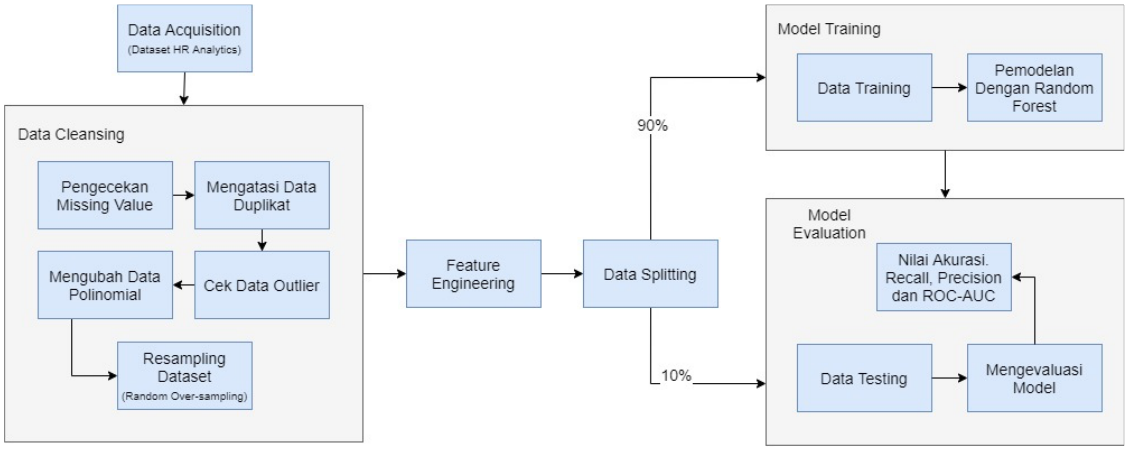
Dari uraian sebelumnya, penelitian ini bertujuan untuk membuat model yang dapat mengklasifikasi perputaran karyawan pada perusahaan menggunakan algoritma *Random Forest*. Evaluasi model yang akan digunakan yaitu *Confusion Matrix* yang terdiri dari akurasi, presisi, dan *recall* serta kurva ROC-AUC. Dataset yang digunakan didapatkan dari *website* penyedia dataset publik yaitu Kaggle dan proses pembuatan model akan menggunakan *tools* Google Colab yang menggunakan bahasa pemrograman Python.

METODE PENELITIAN

Klasifikasi adalah suatu proses yang mengelompokkan objek kedalam suatu kelas (*class*) (Putro, Vulandari, & Saptomo, 2020).

Model klasifikasi menyimpulkan dengan beberapa fungsi pemetaan secara valid yang didapatkan dari *data training* dan memprediksi kelas dengan bantuan dari fungsi pemetaan yang sudah dibuat untuk data yang baru. Berdasarkan tugas dari klasifikasinya, klasifikasi dibagi menjadi dua tipe, yaitu *binary classification* yang hanya menghasilkan dua kemungkinan seperti prediksi cuaca dan *multi-label classification* yang memiliki kemungkinan lebih dari dua kemungkinan seperti klasifikasi kinerja mahasiswa apakah sangat bagus, bagus, cukup, atau buruk (Sen, Hajra, & Ghosh, 2020).

Pendekatan metode yang digunakan pada penelitian ini yaitu *Machine Learning Life Cycle* (MLLC) yang terdiri dari tahap *data acquisition*, *data cleansing*, *feature engineering*, *data splitting*, *model training* dan *model evaluation* (Ibnu Daqiqil Id, 2021). Alur penelitian ditunjukan pada Gambar 1.



Gambar 1. Diagram alur penelitian

Semua Tahapan pada penelitian ini dilakukan menggunakan bahasa pemrograman Python.

A. *Data Acquisition*

Pada tahapan ini, dilakukan pengumpulan data yang didapatkan dari berbagai sumber seperti sistem yang sedang berjalan atau dari perilaku pengguna.

B. *Data Cleansing*

Pada tahapan ini, dataset yang didapatkan akan diidentifikasi dan diperbaiki, jika terdapat kesalahan atau ketidaksesuaian, serta melakukan beberapa penyesuaian untuk meningkatkan kualitas data. Pada tahapan ini akan dilakukan pengecekan *missing value*, mengecek dan mengatasi data duplikat, mengubah data untuk atribut *polinomial* dari yang berbentuk teks menjadi numerik, pengecekan data *outlier* dan melakukan *resampling* pada dataset karena kelas pada dataset jumlahnya tidak seimbang sehingga dilakukan *resampling* dan teknik *resampling* yang digunakan yaitu *Random Over-sampling*. *Random Over-sampling* mengatasi data yang tidak seimbang dengan cara menambahkan data *minority target* secara berulang dan acak (Mohammed, Rawashdeh, & Abdullah, 2020). Hal tersebut ditegaskan pula pada penelitian yang dilakukan oleh (Magnolia, Nurhopipah, &

Kusuma, 2023) dan bahwa dengan mengatasi *imbalanced data,* dapat meningkatkan performa.

C. *Feature Engineering*

Pada tahapan ini, dataset akan disesuaikan dengan melakukan proses ekstraksi atribut pada dataset menjadi atribut yang lainnya atau mereduksi atribut tersebut.

D. *Data Splitting*

Tahapan ini akan melakukan proses pembagian dataset menjadi dua bagian, yaitu *data training* yang akan digunakan untuk membangun model dan *data testing* untuk menguji model yang sudah dibuat.

E. *Model Training*

Pada *model training*, dilakukan proses pembuatan pola atau pengetahuan dari *data training* yang sudah disiapkan sebelumnya dengan menggunakan algoritma tertentu dalam memecahkan masalah. Algoritma yang digunakan dalam membangun model adalah *Random Forest*.

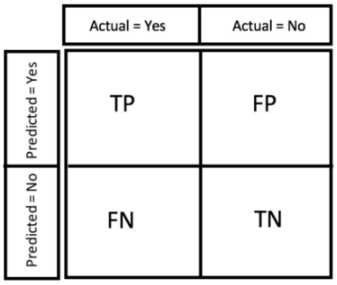
Random Forest adalah salah satu algoritma dari banyaknya algoritma yang dapat digunakan untuk kasus klasifikasi dan juga regresi dari agregasi beberapa pohon keputusan. *Random Forest* merupakan pengembangan lebih lanjut dari *Classification and Regression Tree* (CART) dengan menambahkan *Bootstrap Aggregating* (Bagging) dan *Random Feature Selection*. *Random Forest* dikenal dengan algoritma klasifikasi yang mampu mendapatkan akurasi yang bagus tanpa harus melakukan pencarian secara menyeluruh pada parameter, efektif untuk dataset yang terdapat *missing value*, menghasilkan keputusan dari yang kompleks menjadi lebih sederhana serta membangun lebih dari satu pohon keputusan sehingga hal ini dapat meningkatkan hasil prediksi (Widyastuti, 2020).

F. *Model Evaluation*

Setelah mendapatkan model, maka dilakukan evaluasi untuk mengukur kinerja dari model yang sudah dibuat pada tahapan sebelumnya. Metode evaluasi yang digunakan pada tahapan ini yaitu *Confusion Matrix* dengan metrik akurasi, *recall* dan *precision* serta kurva AUC-ROC untuk mengetahui, seberapa baik model dalam membedakan antar kelas.

1. *Confusion Matrix*

Confusion Matrix adalah metode atau *metric* dengan bentuk matrik berukuran *NxN*, *N* merupakan jumlah dari kelas yang akan diprediksi. *Confusion Matrix* meringkas semua hasil prediksi yang dilakukan dengan membandingkan hasil yang seharusnya dan hasil prediksi. Berikut merupakan gambaran dari *Confusion Matrix*, ditunjukkan pada gambar 2 (Ibnu Daqiqil Id, 2021).



Gambar 2. *Confusion matrix*

Pada Gambar 2, TP (*True Positive*) merupakan data yang diprediksi *Yes* atau benar dan kelasnya *Yes* atau benar juga, FP (*False Positive*) merupakan data yang diprediksi *Yes* atau benar, namun seharusnya terklasifikasi *No* atau tidak, FN (False Negative) merupakan data yang diprediksi *No* atau tidak*,* namun seharusnya *Yes* atau benar, kemudian yang terakhir yaitu TN (*True Negative*) merupakan data yang diprediksi *No* atau tidak dan seharusnya *No* atau tidak.

Dari nilai-nilai Confusion Matrix tersebut, dapat digunakan untuk mencari nilai akurasi, *recall* dan *precision*. Nilai akurasi, *recall* dan *precision* dapat dicari menggunakan persamaan berikut:

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

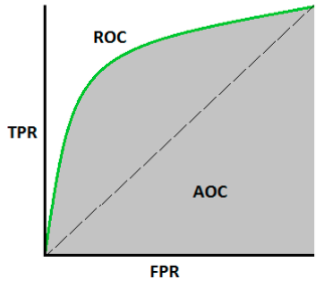
$$Precision = \frac{TP}{TP+FP} \quad (3)$$

Keterangan:

TP = *True Positive* FP = *False Positive*
TN = *True Negative* FN = *False Negative*

2. Kurva AUC-ROC

Kurva AUC-ROC adalah pengukuran performa untuk kasus klasifikasi untuk berbagai macam th*reshold* dengan ROC sebagai kurva probabilitas dan AUC yang mewakili derajat atau ukuran secara terpisah yang dapat memberi tahu seberapa bagus model dalam membedakan antara kelas satu dengan yang lainnya. Semakin tinggi nilai AUC maka semakin bagus model dalam memprediksi dengan sesuai (Ibnu Daqiqil Id, 2021).



Gambar 3. Kurva AUC-ROC

Pada gambar 3, sumbu y merupakan nilai *True Positive Rate* (TPR) dan sumbu x merupakan nilai *False Positive Rate* (FPR). Ketika kurva ROC semakin ke sudut kiri atas, maka model semakin bagus dalam membedakan antar kelas.

HASIL DAN PEMBAHASAN

A. Hasil

1. *Data Acquisition*

Pada tahapan ini, data yang dibutuhkan dikumpulkan dan diidentifikasi atau dipelajari untuk mengetahui, apa saja yang harus dilakukan dalam mempersiapkan dataset sehingga dapat digunakan.

Dataset didapatkan dari salah satu website penyedia dataset publik, yaitu Kaggle dengan nama HR Data for Analytics. Dataset ini memiliki 14999 data dengan 10 atribut, 9 atribut dan 1 kelas. Berikut merupakan penjelasan setiap atribut:

a. *satisfaction_level*, menjelaskan mengenai tingkat kepuasan karyawan dengan rentang nilai 0,00 sampai 1,00.
b. *last_evaluation*, menjelaskan mengenai hasil evaluasi terakhir dari karyawan dengan rentang nilai yang sama dengan *satisfaction_level*, yaitu 0,00 sampai 1,00.
c. *number_project*, menjelaskan sudah berapa banyak proyek yang dikerjakan oleh karyawan dengan nilai numerik, seperti 1, 2, 3.
d. *average_mounthly_hours*, menjelaskan rata-rata jam kerja dalam satu bulan dengan nilai yang sama dengan *number_project*, yaitu numerik.
e. *time_spend_company*, menjelaskan sudah berapa lama karyawan tersebut pada perusahaan tersebut dalam hitungan tahun dengan nilai numerik.
f. *Work_accident*, menjelaskan apakah karyawan pernah mengalami kecelakaan ketika bekerja atau tidak dengan nilai *binomial*, yaitu 0 dan 1.
g. *promotion_last_5years*, menjelaskan apakah karyawan mendapatkan promosi pada 5 tahun terakhir atau tidak dengan nilai yang sama dengan *Work_accident*, yaitu 0 dan 1.
h. *Department*, menjelaskan dari departemen mana karyawan tersebut dengan nilai *polinomial*.
i. *salary*, menjelaskan berapa gaji karyawan tersebut dengan nilai yang sama dengan *Departement* yaitu *polinomial*.
j. *left,* menjelaskan apakah karyawan tersebut keluar dari perusahaan atau tidak. Atribut ini akan digunakan sebagai kelas pada penelitian ini dengan nilai 0 dan 1.

Dataset yang digunakan memiliki ketidak seimbangan antara jumlah kelas yang satu dengan yang lain. Hal ini dapat menyebabkan masalah seperti menurut Yanmin Sun dkk. pada penelitian (Mohammed, Rawashdeh, & Abdullah, 2020) ketika kelas pada suatu dataset cenderung hanya pada satu kelas, yaitu kelas terbanyak atau *majority class* akan mengabaikan kelas yang sedikit atau *minority class* sehingga menyebabkan performa model yang buruk. Sampel dari dataset ditampilkan pada Tabel 1.

Tabel 1. Data Sampel dari Dataset

No.	SL	LE	NP	…	*salary*
1	0,38	0,53	2	…	*low*
2	0,80	0,86	5	…	*medium*
3	0,11	0,88	7	…	*medium*
4	0,72	0,87	5	…	*low*
5	0,37	0,52	2	…	*low*
…	…	…	…	…	…
14999	0,37	0,52	2	…	*Low*

Keterangan:
SL = *satisfaction_level*
LE = *last_evaluation*
NP = *number_project*

2. *Data Cleansing*

Pada tahapan ini, dataset yang sudah didapatkan akan dilakukan proses penyesuaian atau pembersihan agar dataset tersebut dapat digunakan. Hal ini dilakukan agar tidak mengalami hasil klasifikasi yang tidak seusai, seperti output yang tidak sesuai, model yang tidak akurat, dan lain sebagainya (Purohit, 2021). Berikut merupakan proses yang dilakukan pada tahapan ini.

a. Pengecekan *Missing Value*

Pada proses ini dataset akan dicek untuk melihat apakah terdapat nilai yang hilang atau *missing value* pada dataset atau tidak. Berikut merupakan hasil dari proses ini yang disajikan pada tabel 2.

Tabel 2. Hasil Cek *Missing Value*

No	Nama Atribut	Jumlah *Missing Value*
1	*satisfaction_level*	0
2	*last_evaluation*	0
3	*number_project*	0
4	*average_montly_hours*	0
5	*time_spend_company*	0
6	*work_accident*	0
7	*left*	0
8	*promotion_last_5years*	0
9	*Department*	0
10	*salary*	0

Pada Tabel 2 dapat disimpulkan bahwa pada dataset tersebut tidak ada *missing value* sehingga tidak dilakukan penanganan *missing value*.

b. Mengatasi Data Duplikat

Pada proses ini, dataset di cek terlebih dahulu, apakah ada data yang duplikat atau tidak. Setelah dilakukan pengecekan, terdapat data duplikat yaitu sebanyak 3008 data sehingga untuk mengatasinya, data duplikat tersebut dihapus yang mengakibatkan jumlah data pada dataset berkurang dari yang sebelumnya terdapat 14999 data menjadi 11991 data.

c. Mengubah Data *Polinomial*

Pada proses ini, data polinomial akan diubah menjadi numerik karena model yang dibuat hanya menerima data dalam bentuk numerik. Data tersebut yaitu data pada atribut *Department* dan *salary*. Berikut merupakan hasil sebelum dan sesudah data diubah.

Tabel 3. Perbandingan Atribut *Salary*

No.	Nilai Sebelum	Nilai Setelah
1	low	1
2	medium	2
3	high	3

Pada Tabel 3 dapat dilihat bahwa data yang ada pada atribut *salary* sudah berubah, dari sebelumnya berbentuk teks diubah menjadi numerik.

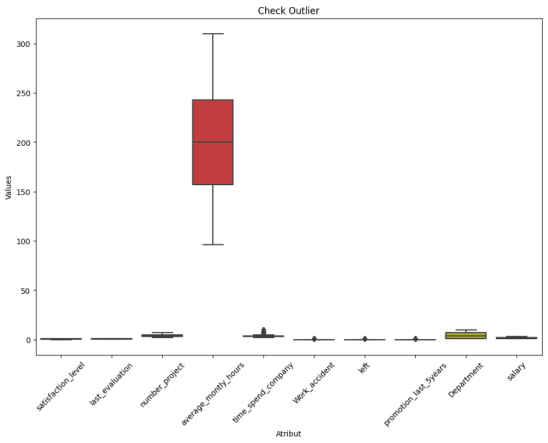
Tabel 4. Perbandingan Atribut *Department*

No.	Nilai Sebelum	Nilai Setelah
1	*sales*	1
2	*accounting*	2
3	*hr*	3
4	*technical*	4
5	*support*	5
6	*management*	6
7	*IT*	7
8	*product_mng*	8
9	*marketing*	9
10	*RandD*	10

Sama seperti pada Tabel 3, data yang ada pada tabel 4 berubah dari yang sebelumnya berbentuk teks menjadi numerik.

d. Pengecekan Outlier

Proses yang selanjutnya yaitu pengecekan *outlier* yang ditampilkan dalam *boxplot* untuk melihat sebaran nilai dari setiap atribut. Berikut merupakan *boxplot* dari dataset yang digunakan.

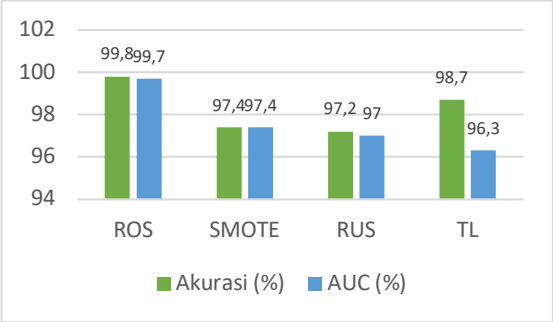


Gambar 4. *Boxplot* dari dataset

Dari Gambar 4 dapat dilihat bahwa sebaran nilai pada dataset tidak terlalu jauh sehingga pada proses ini tidak dilakukan penanganan *outlier*.

e. *Resampling* Dataset

Pada proses terakhir dalam tahapan ini yaitu melakukan *resampling* pada dataset untuk mengatasi tidak seimbangnya jumlah kelas. Teknik resampling yang digunakan pada penelitian ini yaitu *Random Over-sampling*. Namun, dilakukan perbandingan dengan teknik *resampling* yang lain untuk mengetahui apakah ada teknik *resampling* lain yang lebih baik dari *Random Over-sampling* (ROS) pada penelitian ini. Teknik *resampling* tersebut yaitu *Synthetic Minority Over-sampling Technique* (SMOTE), *Random Under-sampling* (RUS) dan *Tomek Links* (TL). Berikut merupakan hasil perbandingannya.

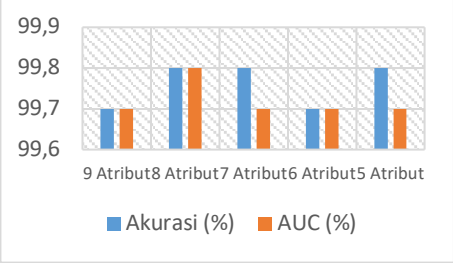


Gambar 5. Perbandingan teknik *resampling*

Dari Gambar 5 dapat disimpulkan bahwa ROS atau *Random Over-sampling* merupakan teknik resampling yang paling baik pada model ini dibandingkan dengan ketiga teknik *resampling* tersebut. Hasil dari proses ini, jumlah antara kelas 0 dan kelas 1 sudah sama yaitu 10000 data dari yang sebelumnya berjumlah 10000 untuk kelas 0 dan 1991 untuk kelas 1.

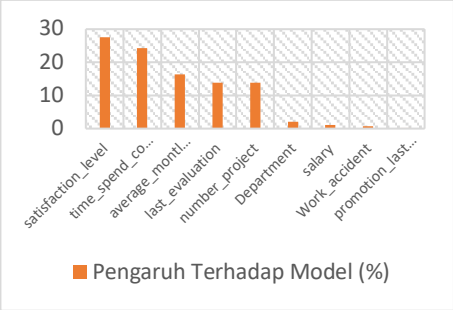
3. *Feature Engineering*

Tahapan selanjutnya yaitu *Feature Engineering* untuk memilih *feature* atau atribut apa saja yang digunakan pada penelitian ini. Hal ini dilakukan agar model lebih sederhana dan mengurangi waktu komputasi yang kompleks (Ma et al., 2022). Teknik pemilihan atribut yang digunakan pada penelitian ini yaitu *Recursive Feature Elimination* (RFE). Karena pada teknik RFE ini dapat ditentukan berapa atribut yang mau digunakan, maka untuk menentukannya dilakukan percobaan. Berikut merupakan hasil percobaannya.



Gambar 6. Perbandingan jumlah atribut

Dari gambar 6 dapat disimpulkan bahwa jumlah atribut yang cocok untuk digunakan pada model ini yaitu 8 atribut yang memiliki nilai akurasi dan AUC paling tinggi. Setelah mendapatkan jumlah atribut yang akan digunakan, selanjutnya menjari atribut apa yang akan dihapus atau tidak digunakan. Untuk mencari hal tersebut, dibuat peringkat pada setiap atribut untuk melihat pengaruh dari atribut tersebut. Berikut hasil dari peringkat atribut berdasarkan pengaruhnya.

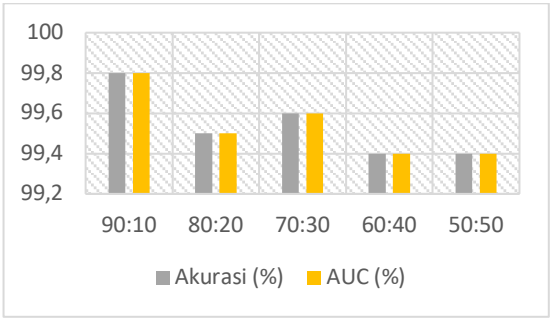


Gambar 7. Peringkat atribut Berdasarkan pengaruh

Dari gambar 7 dapat dilihat bahwa atribut *promotion_last_5years* memiliki peringkat yang paling rendah, sehingga atribut *promotion_last_5years* tidak akan digunakan atau dihapus.

4. *Data Splitting*

Tahap selanjutnya *data splitting*, data yang sudah melalui tahapan *data cleansing* dan *feature engineering* dibagi menjadi dua jenis data, *data training* yang digunakan pada pembuatan model dengan melatih model dan *data testing* yang digunakan dalam mengevaluasi model yang sudah dibuat atau dilatih. Untuk menentukan rasio pembagian data, maka dilakukan percobaan untuk membandingkan beberapa rasio, yaitu rasio 90:10, 80:20, 70:30, 60:40 dan 50:50. Berikut merupakan hasil dari percobaannya.

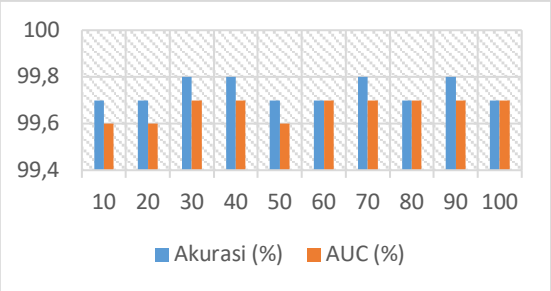


Gambar 7. Perbandingan jumlah pembagian data

Pada Gambar 7 ditunjukkan bahwa rasio pembagian yang paling bagus yaitu 90:10 dengan memiliki nilai akurasi dan AUC yang paling tinggi dibandingkan dengan rasio yang lainnya sehingga rasio yang digunakan yaitu 90:10.

5. *Model Training*

Tahap ini merupakan tahap yang paling utama, karena pada tahap ini model dilatih menggunakan *data training* untuk membuat model sehingga model dapat digunakan dalam mengklasifikasi perputaran karyawan. Model dibuat menggunakan algoritma *Random Forest* dengan parameter jumlah pohon keputusan yang dibuat dan fungsi *split* yang digunakan untuk membuat pohon keputusan. Dalam menentukan jumlah pohon keputusan, dilakukan beberapa percobaan. Percobaan yang pertama yaitu mencari jumlah pohon yang paling baik dari 10 sampai 100. Berikut hasil dari percobaan pertama.



Gambar 8. Percobaan mencari jumlah pohon keputusan

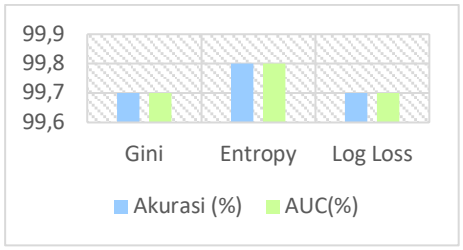
Dari Gambar 8 dapat dilihat bahwa terdapat beberapa jumlah pohon keputusan yang memiliki nilai akurasi dan AUC yang sama, seperti 10 pohon keputusan, 20 pohon keputusan dan 50 pohon keputusan. Oleh karena itu, dilakukan percobaan kedua dengan melatih model sebanyak 10 kali. Jumlah pohon keputusan yang akan digunakan pada percobaan kedua yaitu 50, 90 dan 100 pohon keputusan. Hal ini didasarkan karena jumlah pohon keputusan tersebut mewakili jumlah pohon keputusan yang memiliki nilai akurasi dan AUC yang sama. Berikut merupakan hasil dari percobaan kedua.

Tabel 5. Percobaan 10 Kali Jumlah Pohon Keputusan

No.	50 Pohon (Akurasi)	90 Pohon (Akurasi)	100 Pohon (Akurasi)
1	99,8%	99,7%	99,7%
2	99,7%	99,7%	99,7%
3	99,7%	99,7%	99,7%
4	99,8%	99,8%	99,8%
5	99,7%	99,7%	99,7%
6	99,8%	99,7%	99,7%
7	99,7%	99,7%	99,7%
8	99,8%	99,8%	99,8%
9	99,7%	99,7%	99,7%
10	99,7%	99,7%	99,7%

Hasil percobaan kedua dapat dilihat pada tabel 5 bahwa 50 pohon keputusan merupakan jumlah pohon yang paling baik dengan memiliki nilai akurasi tertinggi sebanyak empat kali sedangkan 90 dan 100 pohon keputusan hanya muncul dua kali dari 10 kali percobaan sehingga untuk parameter jumlah pohon keputusan yang digunakan yaitu 50 pohon keputusan.

Setelah menentukan jumlah pohon keputusan, selanjutnya menentukan parameter fungsi *split* untuk membuat pohon keputusan. Sama seperti sebelumnya, dilakukan percobaan untuk mencari fungsi yang paling baik untuk digunakan. Berikut merupakan hasil percobaan yang dilakukan.



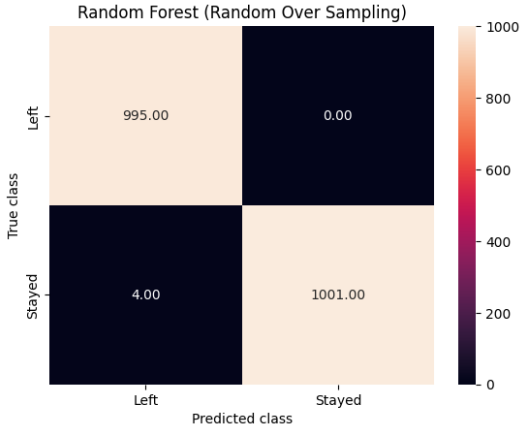
Gambar 9. Percobaan parameter fungsi *split*

Dari Gambar 9 dapat disimpulkan bahwa fungsi *entropy* merupakan fungsi yang paling baik dibandingkan dengan fungsi yang lainnya sehingga parameter fungsi *split* yang digunakan yaitu *entropy*.

Model dibangun menggunakan algoritma *Random Forest* dengan parameter 50 pohon keputusan dan fungsi *split entropy*. Hasil dari model yaitu kumpulan pohon keputusan dengan jumlah 50 pohon keputusan. Jika ingin melihat lebih jelas, dapat diakses pada tautan berikut: http://bit.ly/50PohonKeputusan.

6. *Model Evaluation*

Model yang sudah dilatih, selanjutnya akan dievaluasi untuk melihat seberapa bagus model yang sudah dibuat. Evaluasi yang pertama dilakukan yaitu *Confusion Matrix* dan dari hasil tersebut, dapat dicari nilai akurasi, *recall* dan *precision* dari model. Berikut merupakan *Confusion Matrix* dari model yang dibuat.

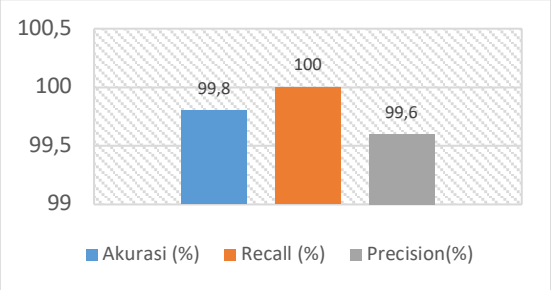


Gambar 10. *Confusion matrix* dari model

Dari gambar 10 dapat disimpulkan bahwa model memiliki performa yang sangat bagus dalam mengklasifikasi. Hanya terdapat 4 data saja yang tidak terklasifikasi dengan baik dari 2000 data sedangkan sisanya yaitu 1996 data lainnya terklasifikasi dengan baik.

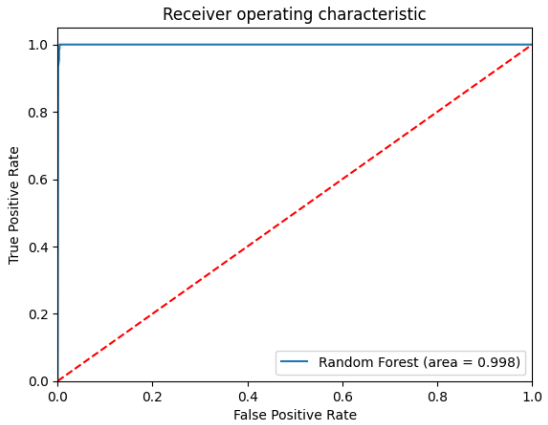
Setelah mendapatkan nilai TP, FP, TP dan TN, maka dapat dihitung akurasi, *recall* dan *precision* dari model dengan persamaan (1), (2)

dan (3). Berikut merupakan nilai akurasi, *recall* dan *precision* dari model.



Gambar 11. Nilai akurasi, *recall* dan *precision*

Pada Gambar 11 dapat disimpulkan bahwa model yang dibuat memiliki kinerja yang sangat bagus dengan nilai akurasi dan *precision* yang hampir mendekati sempurna. Selain itu, untuk melihat seberapa bagus model yang dibuat dalam membedakan kelas, maka dilakukan evaluasi menggunakan kurva ROC-AUC. Berikut merupakan hasil kurva ROC-AUC dari model yang dibuat.



Gambar 12. Kurva ROC dan Nilai AUC

Gambar 12 menunjukkan bahwa model yang dihasilkan dapat membedakan kelas dengan sangat baik. Garis biru pada kurva ROC yang mendekati sudut kiri atas menandakan kurva ROC memiliki performa yang bagus dalam membedakan kelas. Selain itu, nilai AUC yang dihasilkan model ini sangat bagus dan hampir sempurna dengan hasil AUC sebesar 99,8% sehingga model ini termasuk kategori *excellent*.

B. Pembahasan

Dari hasil yang diperoleh, model yang dibuat dapat mengklasifikasi perputaran karyawan dengan sangat baik. Model memiliki performa yang sangat bagus, sehingga hasil klasifikasi dari model memiliki kemungkinan terjadinya kesalahan sangat kecil. Jika dilihat pada gambar 10, dari 2000 data yang digunakan untuk menguji model hanya 4 data saja yang tidak terklasifikasi dengan benar.

Selain itu, jika dibandingkan dengan penelitian sebelumnya, model yang dibuat pada penelitian ini merupakan yang paling baik dari penelitian-penelitian sebelumnya. Jika dilihat dari nilai akurasinya, untuk kasus perputaran karyawan, model dengan algoritma yang berbasis pohon keputusan memiliki nilai akurasi yang besar, seperti pada penelitian yang dilakukan oleh (Setianto & Jatikusumo, 2020) yang membandingkan dua algoritma yaitu *Decision Tree* dan *Naïve Bayes*, *Decision Tree* lebih unggul jika dibandingkan dengan *Naïve Bayes* dengan nilai akurasi dari *Decision Tree* sebesar 91.69% dan *Naïve Bayes* 78.89%. Tidak hanya itu, jika dibandingkan dengan algoritma lain yang digunakan pada penelitian lainnya, *Decision Tree* ini memiliki nilai akurasi yang paling tinggi dan model pada penelitian ini juga yang menggunakan algoritma *Random Forest* yang berbasis pohon keputusan juga memiliki nilai akurasi yang tinggi. Untuk lebih jelasnya dapat dilihat pada tabel berikut.

Tabel 6. Perbandingan Algoritma Penelitian Sebelumnya

No.	Penelitian	Algoritma	Akurasi
1	(Khera & Divya, 2019)	*SVM*	85%
2	(Setianto & Jatikusumo, 2020)	*Decision Tree*	91,69%
3	(Setianto & Jatikusumo, 2020)	*Naïve Bayes*	78,89%
4	(Kinoto et al., 2020)	*Logistic Regression*	64,4%
5	(Zhang, Xu, Cheng, Chao, & Zhao, 2018)	*Logistic Regression*	87,2%
6	(Ahmed, 2021)	*Neural Network*	84%
7	Penelitian ini	*Random Forest*	99.8%

Pada Tabel 6 dapat disimpulkan bahwa algoritma yang berbasis pohon keputusan memiliki nilai akurasi yang paling baik untuk kasus perputaran karyawan meskipun pada penelitian-penelitian tersebut menggunakan dataset yang berbeda-beda. Pada tabel, terdapat dua algoritma yang menggunakan pohon keputusan yaitu *Decision Tree* dan *Random Forest* dan dari kedua algoritma tersebut, *Random Forest* merupakan algoritma terbaik dari *Decision Tree* maupun algoritma yang lainnya.

SIMPULAN

Dari hasil penelitian yang sudah dilakukan, dapat disimpulkan bahwa dataset yang digunakan pada penelitian ini harus dilakukan beberapa penyesuaian agar model yang dihasilkan memiliki performa yang sangat baik. Penyesuaian yang dilakukan yaitu diantaranya mengatasi data duplikat, mengubah data *polinomial* dari yang sebelumnya berbentuk teks menjadi numerik, melakukan resampling menggunakan *Random Over-sampling* serta seleksi fitur dari 9 atribut, hanya menggunakan 8 atribut dan atribut yang tidak digunakan yaitu *promotion_last_5years*. Kemudian, dataset dibagi menjadi dua jenis data, yaitu *data training* yang berjumlah 90% dari dataset dan *data testing* dengan jumlah 10% dari dataset. Model dibangun menggunakan algoritma *Random Forest* dengan parameter 50 pohon keputusan dan fungsi *split entropy*. Hasil evaluasi dari model yang sudah dibangun yaitu model memiliki performa yang sangat baik dan hampir mendekati sempurna, dengan nilai akurasi sebesar 99,8%, *recall* 100% dan *precision* 99,6% serta evaluasi *Confusion Matrix* yang sangat baik, dengan hanya terdapat 4 data saja yang diklasifikasi salah atau tidak tepat dari 2000 data. Model yang dibuat juga memiliki nilai AUC yang sangat baik, yaitu 99,8% sehingga model masuk dalam kategori *excellent*.

DAFTAR PUSTAKA

Ahmed, T. M. (2021). A Novel Classification Model for Employees Turnover Using Neural Network to Enhance Job Satisfaction in Organizations. *Journal of Information and Organizational Sciences*, *45*(2), 361–374. https://doi.org/10.31341/jios.45.2.1

Al-suraihi, W. A., Samikon, S. A., Al-suraihi, A. A., & Ibrahim, I. (2021). Employee Turnover : Causes , Importance and Retention Strategies. *European Journal of Business and Management Research*, *6*(June). https://doi.org/10.24018/ejbmr.2021.6.3.893

Alexandrio, B., Susanti, A. I., & Aflaha, D. S. I. (2020). Sistem Pendukung Keputusan Kepemilikan Karyawan Tetap Di PT Surya Air Menggunakan Metode SAW. *Edu Komputika Journal*, *7*(2), 61–69. https://doi.org/10.15294/edukomputika.v7i2.42385

Ashraf, S., Saleem, S., Ahmed, T., Aslam, Z., & Muhammad, D. (2020). Conversion of Adverse Data Corpus to Shrewd Output Using Sampling Metrics. *Visual Computing for Industry, Biomedicine, and Art*, *3*(1). https://doi.org/10.1186/s42492-020-00055-9

Brennan, D. (2020). Process technology evolution and adoption. In *Process Industry Economics*. https://doi.org/10.1016/b978-0-12-819466-9.00007-9

Ekhsan, M. (2019). The Influence Job Satisfaction And Organizational Commitment On Employee Turnover Intention. *Journal of Business, Management, and Accounting P*, *1*(1), 48–55.

Habib, A., Sheikh, H., & Nabi, N. (2018). Employee Turnover & It ' s Impact on Apparel Industry in Bangladesh : A Case Study of Mondol Group. *Human Resource Management Research*, *8*(3), 63–68. https://doi.org/10.5923/j.hrmr.20180803.03

Ibnu Daqiqil Id. (2021). *Machine Learning : Teori, Studi Kasus dan Implementasi Menggunakan Python* (1st ed., Issue July). UR PRESS. https://doi.org/10.5281/zenodo.5113507

Khera, S. N., & Divya. (2019). Predictive Modelling of Employee Turnover in Indian IT Industry Using Machine Learning Techniques. *Vision*, *23*(1), 12–21. https://doi.org/10.1177/0972262918821221

Kinoto, J., Damanik, J. L., Tri, E., Situmorang, S., Siregar, J., & Harahap, M. (2020). *Prediksi Employee Churn Dengan Uplift Modeling Menggunakan Algoritma Logistic Regression*. *3*, 503–508. https://doi.org/https://doi.org/10.34012/jutikomp.v3i2.1645

Ma, M., Tian, X., Chen, F., Ma, X., Guo, W., & Lv, X. (2022). The Application of Feature Engineering in Establishing A Rapid and Robust Model for Identifying Patients with Glioma. *Lasers in Medical Science*, *37*(2), 1007–1015. https://doi.org/10.1007/s10103-021-03346-6

Magnolia, C., Nurhopipah, A., & Kusuma, B. A. (2023). Penanganan Imbalanced Dataset untuk Klasifikasi Komentar Program

Kampus Merdeka Pada Aplikasi Twitter. *Edu Komputika Journal*, *9*(2), 105–113. https://doi.org/10.15294/edukomputika.v9i2.61854

Mohammed, R., Rawashdeh, J., & Abdullah, M. (2020). Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results. *2020 11th International Conference on Information and Communication Systems, ICICS 2020*, 243–248. https://doi.org/10.1109/ICICS49469.2020.239556

Purohit, K. (2021). Separation of Data Cleansing Concept from EDA. *International Journal of Data Science and Analysis*, *7*(3), 89. https://doi.org/10.11648/j.ijdsa.20210703.16

Putro, H. F., Vulandari, R. T., & Saptomo, W. L. Y. (2020). Penerapan Metode Naive Bayes Untuk Klasifikasi Pelanggan. *Jurnal Teknologi Informasi Dan Komunikasi (TIKomSiN)*, *8*(2). https://doi.org/10.30646/tikomsin.v8i2.500

Sen, P. C., Hajra, M., & Ghosh, M. (2020). Supervised Classification Algorithms in Machine Learning: A Survey and Review. *Advances in Intelligent Systems and Computing*, *937*, 99–111. https://doi.org/10.1007/978-981-13-7403-6_11

Setianto, S. K., & Jatikusumo, D. (2020). Employee Turnover Analysis Using Comparison of Decision Tree and Naive Bayes Prediction Algorithms on K-Means Clustering Algorithms at PT. AT. *Jurnal Mantik*, *4*, 1573–1581. https://doi.org/https://doi.org/10.35335/mantik.Vol4.2020.893.pp1573-1581

Widyastuti, R. W. (2020). *Prediksi Harga Televisi Dengan Menggunakan Penerapan Metode Random Forest Dan Framework Flask*. Universitas Islam Indonesia.

Zhang, H., Xu, L., Cheng, X., Chao, K., & Zhao, X. (2018). Analysis and Prediction of Employee Turnover Characteristics based on Machine Learning. *ISCIT 2018 - 18th International Symposium on Communication and Information Technology*, *Iscit*, 433–437. https://doi.org/10.1109/ISCIT.2018.8587962

Zulfiyandi, Franciscus Anton Wirawan, Tanjung, N. P. S., Yolanda, R., Zaini, M., Andrian, D., Syafitri, K., Amaldi, G., & Sidantha, I. N. B. (2021). *Ketenagakerjaan Dalam Data Edisi 4 Tahun 2021*.