# Prediksi *Employee Attrition* Menggunakan Metode *Decision Tree* dan *XGBoost* dengan Seleksi Fitur *Chi-Square*

Arla Sifhana Putri,
*Fakultas Informatika,*
Universitas Telkom, Bandung
arlasifhana@students.telkomuniversity.ac.id

Kemas Muslim Lhaksmana
*Fakultas Informatika,*
Universitas Telkom, Bandung
kemasmuslim@telkomuniversity.ac.id

**Abstrak**

*Employee attrition* adalah peristiwa di mana suatu perusahaan kehilangan karyawan karena berbagai alasan. *Employee attrition* dapat berdampak negatif terhadap produktivitas dan stabilitas perusahaan, sehingga perusahaan perlu mengambil langkah pencegahan yang tepat terhadap terjadinya *hal tersebut*. Dalam penelitian ini, metode klasifikasi yang digunakan adalah *Decision Tree* dan *XGBoost*, dengan menerapkan seleksi fitur *Chi-square.* Metode *Decision Tree* dipilih karena kemudahan interpretasi dan implementasinya, sementara *XGBoost* dipilih karena memiliki kinerja prediksi yang sangat baik. Seleksi fitur *Chi-square* digunakan untuk mengidentifikasi fitur-fitur yang memiliki hubungan signifikan dengan fitur target. Evaluasi performa antara kedua metode dilakukan menggunakan metrik seperti *accuracy*, *precision*, *recall*, dan *f1-score*. Hasil penelitian menunjukkan bahwa metode *Decision Tree* mencapai akurasi tertinggi sebesar 93.58% dengan memanfaatkan 20 fitur dengan nilai *Chi-square* tertinggi. Sementara itu, metode *XGBoost* berhasil mencapai akurasi terbaik sebesar 98.65% dengan memanfaatkan 25 fitur dengan nilai *Chi-square* tertinggi. Penggunaan seleksi fitur *Chi-square* secara signifikan meningkatkan performa model prediksi. Hal ini menunjukkan bahwa model dengan metode *XGBoost* lebih unggul dalam memprediksi kemungkinan terjadinya *employee attrition* dibandingkan dengan metode *Decision Tree*.

**Kata kunci: employee attrition, prediksi, decision tree, xgboost, chi-square**

**Abstract**

Employee attrition is an event in which a company loses employees for various reasons. It can negatively impact the company's productivity and stability, so companies need to take appropriate preventive measures to avoid it. In this study, the classification methods used are Decision Tree and XGBoost, with Chi-square feature selection applied. The Decision Tree method was chosen for its ease of interpretation and implementation, while XGBoost was selected for its excellent predictive performance. Chi-square feature selection was employed to identify features that are significantly correlated with the target feature. The performance of both methods was evaluated using metrics such as accuracy, precision, recall, and f1-score. The results showed that the Decision Tree method achieved the highest accuracy of 93.58% by utilizing 20 features with the highest Chi-square values. Meanwhile, the XGBoost method achieved the best accuracy of 98.65% by utilizing 25 features with the highest Chi-square values. The use of Chi-square feature selection significantly improved the performance of the predictive model. This indicates that the XGBoost method is superior in predicting the likelihood of employee attrition compared to the Decision Tree method.

**Keywords: *employee attrition*, prediction, *decision tree*, *xgboost*, chi-square**

## 1. PENDAHULUAN

### Latar Belakang

Persaingan antar perusahaan sangat bergantung pada produktivitas karyawan [1]. Kamus Besar Bahasa Indonesia (KBBI) mendefinisikan karyawan sebagai orang yang bekerja pada suatu lembaga (kantor, perusahaan, dan sebagainya) dengan mendapatkan gaji (upah). Karyawan direkrut dengan tujuan untuk mewujudkan visi dan misi perusahaan. Setiap perusahaan pasti menginginkan karyawan yang berkualitas tinggi, sehingga manajemen sumber daya manusia (SDM) yang efektif sangat diperlukan. Salah satu tujuan manajemen SDM adalah mempertahankan karyawan yang kompeten dalam pekerjaannya. Namun, dalam pengelolaan SDM terdapat tantangan berupa *employee attrition* [2]. *Employee attrition* adalah karyawan produktif yang memutuskan untuk meninggalkan perusahaan karena berbagai alasan, seperti tekanan kerja, lingkungan yang tidak sesuai, atau gaji yang tidak memuaskan [3].

Tingkat *employee attrition* yang tinggi menimbulkan masalah signifikan bagi perusahaan karena tingginya biaya pemutusan hubungan kerja, lowongan kerja, rekrutmen, pelatihan, dan penggantian karyawan [4]. Oleh karena itu, perusahaan harus dapat mengurangi tingkat *attrition* dan mengambil tindakan pencegahan yang

diperlukan untuk meminimalkan kerugian yang mungkin dialami.

Penelitian ini memprediksi *employee attrition* menggunakan seleksi fitur *Chi-square* untuk memilih fitur-fitur yang memiliki hubungan signifikan dengan fitur target. Metode klasifikasi yang digunakan adalah *Decision Tree* dan *Extreme Gradient Boosting (XGBoost)*. *Decision Tree* sangat populer digunakan untuk membangun model klasifikasi dibandingkan dengan metode lainnya karena kemudahannya dalam interpretasi dan implementasi [5]. *Decision Tree* dapat menangani data numerik dan kategorikal, memerlukan sedikit pemrosesan data, dan memiliki kinerja baik dengan dataset yang kompleks dalam waktu yang relatif singkat [6]. Sementara itu, *XGBoost* efektif dalam menangani hubungan yang kompleks dalam data, memiliki ketahanan terhadap *overfitting*, dan memberikan kinerja prediksi yang sangat baik [7].

**Topik dan Batasannya**

Rumusan masalah dalam penelitian ini adalah penerapan metode *Decision Tree* dan *XGBoost* untuk memprediksi *employee attrition*, serta menganalisis pengaruh penerapan seleksi fitur *Chi-square* terhadap performa prediksi. Seleksi fitur ini bertujuan untuk meningkatkan akurasi model dengan memilih fitur-fitur yang paling relevan. Untuk menguji pengaruh seleksi fitur terhadap performa model, digunakan lima skenario model. Skenario pertama adalah model *baseline*, yaitu model yang dibangun tanpa menggunakan seleksi fitur. Skenario berikutnya adalah model dengan seleksi fitur berdasarkan 10, 15, 20, dan 25 fitur terbaik yang dipilih menggunakan seleksi fitur *Chi-square*. Setiap skenario bertujuan untuk melihat sejauh mana jumlah fitur yang dipilih dapat mempengaruhi hasil prediksi. Selain itu, penelitian ini juga membandingkan dan mengevaluasi performa masing-masing model dari kedua metode dalam hal akurasi prediksi *employee attrition*.

Adapun batasan masalah dalam penelitian ini adalah penggunaan dataset *Employee Attrition for Healthcare* yang diperoleh dari repositori Kaggle. Dataset tersebut terdiri dari 35 fitur, yang mencakup 34 fitur input dan 1 fitur target, dengan total 1.676 sampel. Data disajikan dalam format CSV dan menggunakan Bahasa Inggris.

**Tujuan**

Tujuan dari penelitian ini adalah untuk mengimplementasikan metode *Decision Tree* dan *XGBoost* dalam memprediksi *employee attrition*. Penelitian ini juga bertujuan untuk menganalisis pengaruh penerapan seleksi fitur *Chi-square* terhadap hasil prediksi yang dihasilkan oleh kedua metode tersebut. Selain itu, penelitian ini bertujuan untuk membandingkan dan mengevaluasi performa kedua metode dalam memprediksi *employee attrition*.

**Organisasi Tulisan**

Bagian selanjutnya dalam penelitian ini adalah bagian 2 yang membahas studi terkait dengan mengulas penelitian sebelumnya yang relevan dengan topik ini. Bagian 3 menjelaskan sistem yang dibangun. Bagian 4 memaparkan evaluasi terhadap hasil pengujian model. Terakhir, bagian 5 menyimpulkan hasil penelitian serta memberikan saran untuk penelitian lebih lanjut.

## 2. KAJIAN TEORI

**Penelitian Terkait**

Penelitian "Employee Turnover Analysis Using Comparison of Decision Tree and Naive Bayes Prediction Algorithms on K-Means Clustering Algorithms at PT. AT" [8] menganalisis tingkat *employee attrition* untuk memprediksi *turnover* tahun 2020. Dataset terdiri dari 7 fitur data karyawan PT. AT tahun 2015-2019. Metode yang digunakan adalah *Decision Tree* dan *Naïve Bayes*. Hasil penelitian menunjukkan bahwa metode *Decision Tree* mencapai akurasi 91.69% dengan presisi 96.82% dan recall 92.64%. Sementara itu, *Naïve Bayes* hanya mencapai akurasi 77.88% dengan presisi 83.83% dan recall 89.6%.

Penelitian "Attrition Analysis using XGBoost and Support Vector Machine Algorithm" [9] membandingkan *XGBoost* dan *Support Vector Machine (SVM)* dalam analisis *attrition*. Dataset yang digunakan adalah *IBM HR Analytics Employee Attrition & Performance* yang terdiri dari 35 fitur dengan 1.470 sampel. Hasilnya penelitian menunjukkan bahwa *XGBoost* unggul dengan akurasi 86%, presisi 89%, recall 94%, dan f1-score 92%. Sementara itu, *SVM* hanya memperoleh akurasi 84%, presisi 91%, recall 90%, dan f1-score 90%.

Penelitian "Envisaging Employee Churn Using MCDM and Machine Learning" [10] menganalisis penyebab *employee churn* dan menyusun strategi retensi. Dataset yang digunakan adalah *Employee Attrition* yang terdiri dari 8 fitur dengan 4.507 data karyawan. Metode yang digunakan meliputi *CatBoost*, *Support Vector Machine*, *Decision Tree*, *Random Forest*, dan *XGBoost*. Hasilnya penelitian menunjukkan bahwa *Decision Tree* mencapai akurasi 97.1%, presisi 98.5%, dan recall 97.9%. Sementara itu, *XGBoost* hanya mencapai akurasi 97.3%, presisi 96.1%, dan recall 97.2%.

Penelitian "Employee Attrition Prediction In Industry Using Machine Learning Techniques" [11] membantu organisasi mengambil keputusan proaktif melalui prediksi *employee attrition*. Dataset yang digunakan adalah *IBM HR Analytics Employee Attrition & Performance* yang terdiri dari 35 fitur. Metode yang digunakan meliputi *Artificial Neural Network*, *Support Vector Machine*, *Gradient Boosting*, *Bagging*, *Random Forest*, dan *Decision Tree*. Teknik seleksi fitur yang diterapkan adalah *Correlation-based Feature Selection*, *Information*

*Gain*, *Gain Ratio*, *Chi-square*, dan *Fisher Exact Test*. Hasil penelitian menunjukkan bahwa metode *Decision Tree* tanpa seleksi fitur memperoleh akurasi 42.86% dan presisi 45.76%. Sementara itu, dengan seleksi fitur *Chi-square*, akurasi meningkat menjadi 81.22% dan presisi 78.18%.

Penelitian "Predictive Modeling of Employee Churn Analysis for IoT-Enabled Software Industry" [12] menganalisis faktor-faktor yang mempengaruhi *employee churn*. Dataset yang digunakan terdiri dari 10 fitur dengan 14.999 data karyawan. Metode yang digunakan meliputi *Support Vector Machine*, *Decision Tree*, *Neural Network*, *Linear Regression*, dan *Decision Forest*. Teknik seleksi fitur yang diterapkan adalah *Chi-square*, *Spearman Correlation*, *Fisher Score*, dan *R Coefficient Correlation*. Hasil penelitian menunjukkan bahwa *Decision Tree* dengan seleksi fitur *Chi-square* mencapai akurasi 98.6%, presisi 97.7%, dan recall 96.4%.

**Employee Attrition**

Karyawan adalah aset yang sangat berharga bagi suatu perusahaan, bahkan hanya dengan kehadirannya saja [5]. Ketika karyawan meninggalkan perusahaan, baik secara sukarela maupun karena tidak sukarela, hal ini disebut sebagai *employee attrition* [13]. *Employee attrition* menjadi perhatian utama dalam perusahaan karena dampak negatifnya yang luas, mulai dari penurunan semangat kerja dan produktivitas di tempat kerja, hingga gangguan terhadap kelangsungan proyek dan strategi pertumbuhan jangka panjang [14]. Mempertahankan karyawan merupakan tantangan besar bagi perekrut dan pengusaha, karena kepergian karyawan tidak hanya berarti hilangnya keterampilan, pengalaman, dan tenaga kerja, tetapi juga berpotensi menyebabkan kerugian dalam peluang bisnis [15].

**Decision Tree**

*Decision Tree* adalah salah satu metode *supervised learning* yang didefinisikan sebagai partisi rekursif *top-down*. Dalam setiap iterasi, metode ini memilih suatu keputusan hingga menghasilkan pohon terstruktur hierarkis, di mana setiap cabang dari akar hingga daun dapat mewakili aturan *IF-THEN* [16]. Metode *Decision Tree* memecah dataset menjadi sejumlah *subset* yang lebih kecil dan disaat yang bersamaan pohon keputusan dikembangkan secara bertahap [17]. Pemecahan *Decision Tree* dibagi menjadi 3 yaitu *root node*, *decision node*, dan *leaf node*.

**XGBoost**

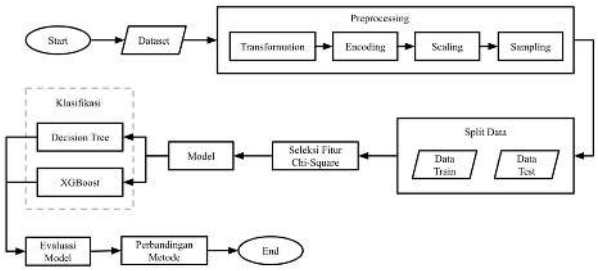
*XGBoost* adalah algoritma pohon yang ditingkatkan (*boosted tree algorithm*) yang mengikuti prinsip *Gradient Boosting*, di mana beberapa set pembelajar (*tree*) lemah digabungkan menjadi sebuah model kuat [14]. Model ini dikembangkan menggunakan metode *boosting*, di mana setiap model baru dibangun berdasarkan model sebelumnya dengan tujuan mengurangi kesalahan prediksi dari model sebelumnya. XGBoost merupakan varian dari *Gradient Boosting Machine* yang lebih efisien dan terukur, serta mampu menyelesaikan berbagai permasalahan seperti regresi, pemeringkatan, dan klasifikasi [18].

**Chi-Square**

Seleksi fitur adalah proses pemilihan fitur-fitur dalam dataset yang paling relevan dengan tujuan pemodelan prediktif yang sedang dijalankan. Seleksi fitur digunakan karena dapat membantu dalam memahami data, mengurangi kebutuhan komputasi, mengurangi dimensi data, dan meningkatkan kinerja prediktor [19]. Pada penelitian ini, akan digunakan metode seleksi fitur *Chi-square*. *Chi-square* atau chi-kuadrat adalah uji yang dilakukan untuk memberikan bukti adanya hubungan atau tidak ada hubungan antara fitur-fitur kategorikal [20]. *Chi-square* akan memilih fitur-fitur yang memiliki hubungan kuat dengan fitur target.

## 3. METODE

Penelitian ini mengembangkan sistem klasifikasi melalui beberapa tahapan, yaitu eksplorasi dataset, *preprocessing* (termasuk *data transformation*, *encoding*, *scaling*, dan *sampling*), pembagian data (menjadi *data train* dan *data test*), serta seleksi fitur *Chi-square*. Pemodelan dilakukan dengan metode klasifikasi *Decision Tree* dan *XGBoost*, yang dievaluasi menggunakan metrik seperti *accuracy*, *precision*, *recall*, dan *f1-score*. Terakhir, hasil kedua metode dibandingkan untuk menilai performa masing-masing model. Flowchart perancangan sistem dapat dilihat pada Gambar 1.




Gambar 1. Flowchart Perancangan Sistem

**Dataset**

Data yang digunakan dalam penelitian ini adalah dataset *Employee Attrition for Healthcare* yang diperoleh dari repositori Kaggle. Dataset tersebut terdiri dari 35 fitur, yang mencakup 34 fitur input dan 1 fitur target, dengan total 1.676 sampel. Fitur-fitur dalam dataset disajikan dalam Tabel 1.

TABEL 1
Fitur Dataset

| Nama Fitur | Tipe | Nama Fitur | Tipe |
|---|---|---|---|
| EmployeeID | Numerik | MonthlyIncome | Numerik |
| Age | Numerik | MonthlyRate | Numerik |
| Attrition | Kategorikal | NumCompaniesWorked | Numerik |
| BusinessTravel | Kategorikal | Over18 | Kategorikal |
| DailyRate | Numerik | OverTime | Kategorikal |
| Department | Kategorikal | PercenSalaryHike | Numerik |
| DistanceFromHome | Numerik | PerformanceRating | Kategorikal |
| Education | Kategorikal | RelationshpSatisfaction | Kategorikal |
| EducationField | Kategorikal | StandardHours | Numerik |
| EmployeeCount | Numerik | Shift | Kategorikal |
| EnvirontmentSatisfaction | Kategorikal | TotalWorkingYears | Numerik |
| Gender | Kategorikal | TrainingTimesLastYear | Numerik |
| HourlyRate | Numerik | WorkLifeBalance | Kategorikal |
| JobInvolvement | Kategorikal | YearsAtCompany | Numerik |
| JobLevel | Kategorikal | YearsInCurrentRole | Numerik |
| JobRole | Kategorikal | YearsSinceLastPromotion | Numerik |
| JobSatisfaction | Kategorikal | YearsWithCurrManager | Numerik |
| MaritalStatus | Kategorikal | | |

**Preprocessing**

*Preprocessing* adalah serangkaian langkah yang dilakukan sebelum proses klasifikasi untuk mempersiapkan data mentah menjadi format yang lebih sesuai untuk analisis. Langkah-langkah preprocessing yang dilakukan pada penelitian ini adalah sebagai berikut:

a. Data Transformation

*Data transformation* adalah proses menghapus fitur-fitur dalam dataset yang tidak memiliki nilai kontinu atau dianggap tidak relevan untuk model prediksi. Fitur-fitur yang dihilangkan meliputi *EmployeeID* yang bernilai unik pada setiap karyawan, *EmployeeCount* yang selalu bernilai 1, *Over18* yang hanya memiliki nilai Y, dan *StandardHours* yang secara konsisten bernilai 80.

b. Data Encoding

*Data encoding* adalah proses mengubah tipe data suatu fitur dari satu tipe ke tipe lainnya. Dalam penelitian ini, encoding yang diterapkan adalah *label encoding*, yang digunakan untuk mengubah data kategorikal menjadi numerik untuk mempermudah pemodelan.

c. Data Scaling

*Data scaling* adalah proses mengubah nilai fitur numerik ke dalam rentang skala yang seragam. Pada dataset, terdapat fitur dengan rentang yang berbeda-beda, seperti *Age* (18-60), *MonthlyIncome* ($1.009-$19.999), *TotalWorkingYears* (0-40), serta fitur numerik lainnya yang memiliki rentang skala yang berbeda. Untuk menyamakan skala, data dinormalisasi menggunakan *Min Max Scaler*, yang memetakan nilai fitur ke rentang 0 hingga 1. Rumus *Min Max Scaler* dapat dituliskan sebagai berikut.

$$\chi_{scaled} = \frac{\chi - \chi_{min}}{\chi_{max} - \chi_{min}} \quad (1)$$

$\chi_{scaled}$ adalah nilai yang telah dinormalisasi, $\chi$ adalah nilai awal, sementara $\chi_{min}$ dan $\chi_{max}$ adalah nilai minimum dan maksimum fitur.

**Split Data**

Dataset dipisah menjadi dua subset, yaitu *data train* dan *data test*, dengan proporsi 90% untuk d*ata train* dan 10% untuk *data test*. Pembagian data ini dilakukan secara acak. *Data train* digunakan untuk melatih model dalam menghasilkan prediksi yang akurat, sementara *data test* digunakan untuk menguji performa model.

**Data Sampling**

*Data sampling* adalah proses mengatasi ketidakseimbangan antara kelas mayoritas dan minoritas dalam dataset. Proses ini dilakukan dengan menggunakan metode *oversampling*, dimana jumlah sampel pada kelas minoritas ditingkatkan agar seimbang dengan kelas mayoritas. Metode *oversampling* yang digunakan adalah *SMOTE* (*Synthetic Minority Over-Sampling Technique*), yang menghasilkan sampel sintetis untuk kelas minoritas.

**Seleksi Fitur**

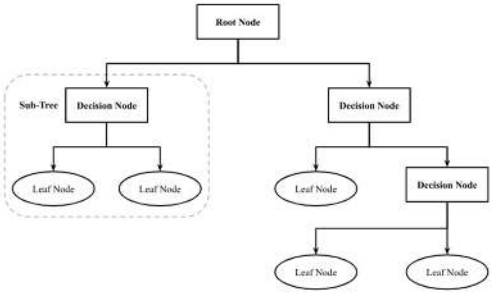
Pada tahap ini, setiap fitur diuji untuk menentukan seberapa signifikan hubungannya dengan fitur target. Fitur-fitur dengan nilai *Chi-square* yang tinggi akan dipilih sebagai fitur terbaik, sementara fitur-fitur yang dianggap memiliki hubungan yang kurang signifikan dengan fitur target dapat diabaikan atau dihapus dari dataset. Rumus *Chi-Square* dapat dituliskan seperti berikut.

$$\chi^2 = \sum_i^r \sum_j^c \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \quad (2)$$

$\chi^2$ adalah statistik *Chi-Square*, $r$ adalah jumlah baris, $c$ adalah jumlah fitur, $O_{ij}$ adalah frekuensi nilai pengamatan pada baris i fitur j, dan $E_{ij}$ adalah frekuensi nilai harapan pada baris i fitur j.

**Klasifikasi**

Tahap klasifikasi menggunakan metode *Decision Tree* dan *XGBoost* memanfaatkan model fitur *Chi-Square*. Model ini mengaplikasikan seleksi fitur *Chi-Square* untuk meningkatkan hasil akurasi dibandingkan dengan tanpa menggunakan seleksi fitur. Arsitektur metode *Decision Tree* ditunjukkan pada Gambar 2, sementara arsitektur metode *XGBoost* ditunjukkan pada Gambar 3.

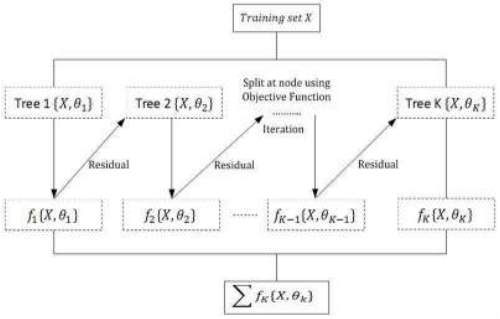



GAMBAR 2
Arsitektur Decision Tree

Pada Gambar 2. terdapat tiga jenis node, yaitu *root node*, *decision node*, dan *leaf node*. *Root node* adalah node awal yang mewakili seluruh kumpulan data atau keputusan yang akan dianalisis. *Decision node* adalah titik bercabang yang menunjukkan bahwa keputusan harus diambil berdasarkan suatu kriteria. Setiap *decision node* memiliki dua atau lebih cabang. *Leaf node* adalah titik akhir dari *decision tree* yang menunjukkan hasil atau keputusan akhir setelah melalui berbagai cabang. Selain node, *decision tre*e juga mempunyai *branches* (ranting) yang merupakan garis yang menghubungkan node satu dengan yang lainnya, menunjukkan jalur atau pilihan dari satu keputusan ke keputusan lainnya. Rumus *Decision Tree* dituliskan seperti berikut.

$$Entropy(S) = \sum_{i=1}^{n} - pi \times log_2 pi \quad (3)$$

$S$ adalah himpunan kasus, $n$ Jumlah kasus pada partisi $S$, dan $pi$ adalah proporsi $i$ terhadap $S$.




GAMBAR 3
Arsitektur XGBoost

Pada Gambar 3. proses dimulai dengan input data, yang kemudian diproses melalui beberapa pohon keputusan secara berurutan. Setiap pohon bertugas mengoreksi kesalahan dari pohon sebelumnya dengan memfokuskan pada residu atau kesalahan prediksi yang belum diperbaiki. Proses ini dikenal sebagai *boosting*, di mana model-model lemah digabungkan untuk membentuk model yang lebih kuat. Setelah melalui serangkaian pohon keputusan, hasil prediksi dari setiap pohon digabungkan untuk menghasilkan prediksi akhir. Proses ini melibatkan penambahan bobot pada setiap pohon, yang ditentukan berdasarkan kinerjanya dalam mengurangi kesalahan. Rumus *XGBoost* dapat dituliskan seperti berikut.

$$\mathcal{L}(y_i, y\hat{}_i) = -[y_i \log(y\hat{}_i) + (1 - y_i) \log(1 - y\hat{}_i)] \quad (4)$$

$y_i$ adalah label aktual (0 atau 1) untuk sampel ke-$i$ dan $y_i$ adalah probabilitas yang diprediksi bahwa sampel ke-$i$ termasuk kelas positif.

**Evaluasi**

Pada tahap ini, model yang telah diimplementasi dievaluasi untuk mengetahui tingkat kinerja yang dimilikinya. Hasil klasifikasi dinyatakan efektif jika nilai performa menunjukkan nilai performa yang tinggi. Untuk menghitung nilai performa model digunakan *confusion matrix*. *Confusion matrix* merupakan sebuah matriks yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan membandingkan hasil prediksi model *machine learning* dengan nilai sebenarnya [10]. Dalam *confusion matrix*, terdapat empat kemungkinan hasil, yaitu *true positive* (TP), *true negative* (TN), *false positive* (FP), dan *false negative* (FN). Tabel *confusion matrix* disajikan dalam Tabel 2.

TABEL 2
Confusion Matrix

| | Aktual Positif | Aktual Negatif |
|---|---|---|
| Prediksi Positif | TP | FP |
| Prediksi Negatif | FN | TN |

Hasil dari *confusion matrix* dapat dianalisis

menggunakan fungsi *classification report*, yang mencakup metrik seperti *accuracy*, *precision*, *recall*, dan *f1-score*. Metriks-metriks evaluasi yang digunakan dalam penelitian ini adalah sebagai berikut:

a. Accuracy

*Accuracy* adalah tingkat ketepatan model dalam mengklasifikasikan data dengan benar [9]. *Accuracy* dapat diterapkan menggunakan rumus berikut.

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (5)$$

b. Precision

*Precision* adalah tingkat kesesuaian antara data yang diharapkan dengan hasil prediksi yang dihasilkan oleh model [9]. *Precision* dapat diterapkan menggunakan rumus berikut.

$$precision = \frac{TP}{TP + FP} \quad (6)$$

c. Recall

*Recall* adalah rasio yang mengukur nilai aktual positif dibandingkan dengan nilai aktual positif secara keseluruhan [9]. *Recall* dapat diterapkan menggunakan rumus berikut.

$$recall = \frac{TP}{TP + FN} \quad (7)$$

d. F1-Score

*F1-score* adalah matriks yang mengukur nilai rata-rata harmonik antara *precision* dan *recall* [2]. *F1-score* dapat diterapkan menggunakan rumus berikut.

$$f1 - score = \frac{2 \times (precison \times recall)}{precision + recall} \quad (8)$$

## 4. HASIL DAN PEMBAHASAN

**Hasil Preprocessing**

Pada tahap *preprocessing*, dilakukan penghapusan terhadap empat fitur dalam dataset, yaitu fitur *EmployeeID*, *EmployeeCount*, *Over18*, dan *StandardHours*, sehingga hanya tersisa 31 fitur yang digunakan untuk analisis lebih lanjut. Selain itu, untuk mengatasi ketidakseimbangan kelas, dilakukan *oversampling* data menggunakan metode *SMOTE*. Sebelum penanganan, jumlah sampel untuk kelas *attrition* adalah 199 sampel dengan nilai 1 (*Yes*) dan 1.477 sampel dengan nilai 0 (*No*). Setelah penanganan, jumlah sampel untuk kelas 1 (*Yes*) meningkat menjadi 1.477, sementara jumlah sampel untuk kelas 0 (*No*) tetap 1.477.

**Hasil Seleksi Fitur**

Seleksi fitur menggunakan metode *Chi-square* dilakukan untuk mengukur hubungan antara setiap fitur dalam dataset dengan target. Hasil pengujian kemudian diurutkan berdasarkan skor tertinggi, sehingga fitur dengan skor *Chi-square* paling tinggi akan berada di urutan teratas. Fitur-fitur yang dipilih untuk pengujian model terdiri dari 10, 15, 20, dan 25 fitur terbaik yang memiliki skor *Chi-square* tertinggi. Hasil pemeringkatan fitur berdasarkan skor *Chi-square* disajikan dalam Tabel 3.

TABEL 3
Skor Chi-Square

| Rank | Nama Fitur | Skor | Rank | Nama Fitur | Skor |
|---|---|---|---|---|---|
| 1 | OverTime | 136.592 | 16 | YearsSinceLastPromotion | 3.933 |
| 2 | JobLevel | 43.313 | 17 | WorkLifeBalance | 2.447 |
| 3 | Shift | 38.122 | 18 | NumCompaniesWorked | 1.153 |
| 4 | MartialStatus | 27.801 | 19 | Education | 0.914 |
| 5 | MonthlyIncome | 13.410 | 20 | DailyRate | 0.803 |
| 6 | TotalWorkingYears | 12.432 | 21 | MonthlyRate | 0.588 |
| 7 | YearsInCurrentRole | 12.408 | 22 | TrainingTimesLastYear | 0.496 |
| 8 | YearsWithCurrManager | 12.207 | 23 | BusinessTravel | 0.366 |
| 9 | Age | 10.146 | 24 | HourlyRate | 0.362 |
| 10 | YearsAtCompany | 8.980 | 25 | RelationshipSatisfaction | 0.299 |
| 11 | JobInvolment | 8.641 | 26 | Gender | 0.289 |
| 12 | EnvironmentSatisfaction | 7.623 | 27 | EducationField | 0.232 |
| 13 | DistanceFromHome | 5.397 | 28 | JobRole | 0.023 |
| 14 | JobSatisfaction | 4.997 | 29 | PerformanceRating | 0.00 |

| | | | | | 7 |
|---|---|---|---|---|---|
| 15 | Department | 4.123 | 30 | PercentSalaryHike | 0.003 |

Proses ini memanfaatkan fungsi *Select K Best* yang diterapkan dengan skor *Chi-square* untuk mengidentifikasi fitur-fitur yang memiliki pengaruh paling signifikan terhadap target. Fitur-fitur yang terpilih kemudian digunakan dalam pengujian model, yang memungkinkan evaluasi kinerja model dengan menggunakan sejumlah fitur terbaik berdasarkan skor *Chi-square* tertinggi. Daftar fitur yang terpilih disajikan dalam Tabel 4.

TABEL 4
Daftar Fitur Terpilih

| Jumlah Fitur | Fitur Terpilih |
|---|---|
| 10 Fitur | 'OverTime', 'JobLevel', 'Shift', 'MaritalStatus', 'MonthlyIncome', 'TotalWorkingYears', 'YearsInCurrentRole', 'YearsWithCurrManager', 'Age', 'YearsAtCompany' |
| 15 Fitur | 'OverTime', 'JobLevel', 'Shift', 'MaritalStatus', 'MonthlyIncome', 'TotalWorkingYears', 'YearsInCurrentRole', 'YearsWithCurrManager', 'Age', 'YearsAtCompany', 'JobInvolvement', 'EnvironmentSatisfaction', 'DistanceFromHome', 'JobSatisfaction', 'Department' |
| 20 Fitur | 'OverTime', 'JobLevel', 'Shift', 'MaritalStatus', 'MonthlyIncome', 'TotalWorkingYears', 'YearsInCurrentRole', 'YearsWithCurrManager', 'Age', 'YearsAtCompany', 'JobInvolvement', 'EnvironmentSatisfaction', 'DistanceFromHome', 'JobSatisfaction', 'Department', 'YearsSinceLastPromotion', 'WorkLifeBalance', 'NumCompaniesWorked', 'Education', 'DailyRate' |
| 25 Fitur | 'OverTime', 'JobLevel', 'Shift', 'MaritalStatus', 'MonthlyIncome', 'TotalWorkingYears', 'YearsInCurrentRole', 'YearsWithCurrManager', 'Age', 'YearsAtCompany', 'JobInvolvement', 'EnvironmentSatisfaction', 'DistanceFromHome', 'JobSatisfaction', 'Department', 'YearsSinceLastPromotion', 'WorkLifeBalance', 'NumCompaniesWorked', 'Education', 'DailyRate', 'MonthlyRate', 'TrainingTimesLastYear', 'BusinessTravel', 'HourlyRate', 'RelationshipSatisfaction' |

**Hasil Pengujian**

Setelah proses seleksi fitur selesai, langkah berikutnya adalah membangun model klasifikasi menggunakan metode *Decision Tree* dan *XGBoost*. Terdapat lima skenario model yang diuji. Skenario pertama adalah model *baseline*, yaitu model yang dibangun tanpa menggunakan seleksi fitur. Skenario berikutnya adalah model dengan seleksi fitur berdasarkan 10, 15, 20, dan 25 fitur terbaik yang dipilih menggunakan seleksi fitur *Chi-square*.

**Metode Decision Tree**

Hasil performa dari model *Decision Tree* menunjukkan peningkatan yang signifikan ketika fitur-fitur yang paling relevan dipilih menggunakan *Chi-square*. Hasil performa dari kelima skenario yang diuji pada metode *Decision Tree* disajikan dalam Tabel 5.

TABEL 5
Hasil Pengujian *Decision Tree*

| Model | Jumlah Fitur | Akurasi | Presisi | Recall | F1-Score |
|---|---|---|---|---|---|
| Tanpa seleksi fitur | 31 | 89.53% | 87.74 % | 91.89 % | 89.77 % |
| Chi-Square | 10 | 90.20% | 88.89 % | 91.89 % | 90.37 % |
| | 15 | 91.89% | 90.79 % | 93.24 % | 92.00 % |
| | 20 | **93.58 %** | **92.72 %** | **94.59 %** | **93.65 %** |
| | 25 | 92.57% | 90.91 % | 94.59 % | 92.72 % |

Hasil pengujian menggunakan model *Decision Tree* menunjukkan variasi performa yang signifikan tergantung pada jumlah fitur yang dipilih melalui seleksi fitur *Chi-square*. Pada model tanpa seleksi fitur, dengan menggunakan 31 fitur, didapatkan akurasi sebesar 89.53%, presisi 87.74%, recall 91.89%, dan f1-score 89.77%. Ketika menggunakan seleksi fitur *Chi-square* dengan 10 fitur, akurasi sedikit meningkat menjadi 90.20%, dengan presisi 88.89%, recall 91.89%, dan f1-score 90.37%. Peningkatan lebih lanjut terlihat pada jumlah fitur yang lebih banyak. Dengan 15 fitur, akurasi mencapai 91.89%, presisi 90.79%, recall 93.24%, dan f1-score 92.00%. Pada 20 fitur, akurasi tertinggi tercatat yaitu 93.58%, diikuti dengan presisi 92.72%, recall 94.59%, dan f1-score 93.65%. Namun, setelah menambahkan jumlah fitur menjadi 25, meskipun akurasi sedikit menurun menjadi 92.57%, presisi sedikit meningkat menjadi 90.91%, recall tetap pada 94.59%, dan f1-score turun menjadi 92.72%.

**Metode XGBoost**

Hasil performa dari model *XGBoost* menunjukkan kinerja yang lebih baik dibandingkan dengan *Decision Tree* pada semua skenario. Hasil performa dari kelima skenario yang diuji pada metode *XGBoost* disajikan dalam Tabel 6.

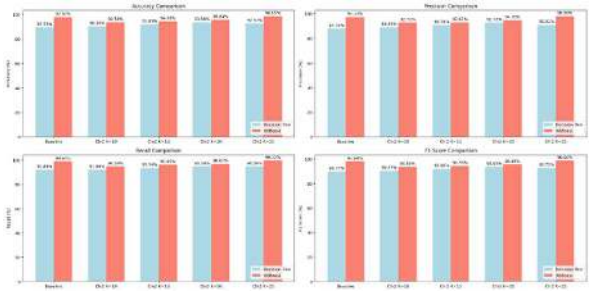
TABEL 6
Hasil Pengujian *XGBoost*

| Model | Jumlah Fitur | Akurasi | Presisi | Recall | F1-Score |
|---|---|---|---|---|---|
| Tanpa seleksi fitur | 31 | 97.97% | 97.33 % | 98.65 % | 97.99 % |
| Chi-Square | 10 | 93.58% | 92.72 % | 94.59 % | 93.65 % |
| | 15 | 94.26% | 92.81 % | 95.95 % | 94.35 % |
| | 20 | 95.61% | 94.70 % | 96.62 % | 95.65 % |
| | 25 | **98.65 %** | **98.00 %** | **99.32 %** | **98.66 %** |

Hasil pengujian menggunakan model XGBoost menunjukkan kinerja yang sangat baik di berbagai pengaturan jumlah fitur. Pada model tanpa seleksi fitur, dengan menggunakan 31 fitur, didapatkan akurasi yang sangat tinggi yaitu 97.97%, dengan presisi 97.33%, recall 98.65%, dan f1-score 97.99%. Namun, penggunaan seleksi fitur *Chi-square* dengan jumlah fitur yang lebih sedikit menunjukkan variasi performa yang lebih beragam. Pada seleksi fitur *Chi-square* dengan 10 fitur, akurasi menurun menjadi 93.58%, dengan presisi 92.72%, recall 94.59%, dan f1-score 93.65%. Peningkatan performa terlihat pada 15 fitur, dengan akurasi mencapai 94.26%, presisi 92.81%, recall 95.95%, dan f1-score 94.35%. Selanjutnya, dengan 20 fitur, performa model semakin membaik, mencatatkan akurasi 95.61%, presisi 94.70%, recall 96.62%, dan f1-score 95.65%. Meskipun terjadi penurunan akurasi pada seleksi fitur yang lebih sedikit, hasil yang sangat baik tercatat pada penggunaan 25 fitur, di mana akurasi meningkat signifikan menjadi 98.65%, presisi 98.00%, recall 99.32%, dan f1-score 98.66%.

**Analisis Hasil Pengujian**

Model *Decision Tree* dan *XGBoost* menunjukkan perbedaan performa yang signifikan, baik pada skenario baseline maupun setelah penerapan seleksi fitur *Chi-square*. Perbandingan performa antara metode *Decision Tree* dan *XGBoost* dalam mengembangkan model ditunjukkan pada Gambar 4.



GAMBAR 4
Perbandingan Performa Metode Decision Tree dan XGBoost

Dari hasil pengujian model *Decision Tree* dan *XGBoost*, terlihat perbedaan kinerja antara kedua model dalam hal seleksi fitur dan metrik evaluasi. Pada model *Decision Tree*, penerapan seleksi fitur *Chi-square* menunjukkan peningkatan yang signifikan dalam akurasi, presisi, recall, dan f1-score dibandingkan dengan model tanpa seleksi fitur. Peningkatan performa terbesar terjadi pada 20 fitur, yang memberikan akurasi tertinggi dan hasil f1-score terbaik. Namun, peningkatan lebih lanjut pada 25 fitur justru menunjukkan penurunan kecil dalam akurasi, meskipun recall tetap tinggi.

Di sisi lain, *XGBoost* menunjukkan performa luar biasa dengan akurasi yang sangat tinggi pada model tanpa seleksi fitur. Seleksi fitur *Chi-square* pada *XGBoost* dengan jumlah fitur yang lebih sedikit (seperti 10 atau 15 fitur) menyebabkan penurunan performa yang cukup besar, namun performa meningkat kembali pada 20 fitur dan mencapai puncaknya pada 25 fitur dengan akurasi hampir sempurna dan f1-score yang sangat tinggi.

Secara keseluruhan, model dari kedua metode menunjukkan bahwa seleksi fitur dapat mengurangi *overfitting* dan meningkatkan efisiensi model, namun *XGBoost* lebih unggul dalam mempertahankan akurasi tinggi meskipun menggunakan seleksi fitur.

## 5. KESIMPULAN

Berdasarkan evaluasi, dapat disimpulkan bahwa baik model *Decision Tree* maupun *XGBoost* menunjukkan efektivitas yang baik dalam memprediksi *employee attrition*. Penerapan seleksi fitur *Chi-square* terbukti meningkatkan performa kedua model, dengan *XGBoost* menunjukkan kinerja yang lebih konsisten dan lebih tinggi, terutama pada jumlah fitur yang optimal. Pada model *Decision Tree*, hasil terbaik dicapai saat menggunakan seleksi fitur *Chi-square* 20 fitur, dengan akurasi 93.58%, presisi 92.72%, recall 94.59%, dan f1-score 93.65%. Sementara itu, model *XGBoost* mencapai performa tertinggi saat menggunakan seleksi fitur *Chi-square* 25 fitur, dengan akurasi 98.65%, presisi 98.00%, recall 99.32%, dan F1-score 98.66%. Secara keseluruhan, *XGBoost* lebih unggul dalam hal prediksi *employee attrition*, namun kedua metode tetap memberikan hasil yang memadai dengan penerapan seleksi fitur *Chi-square*. Penelitian selanjutnya disarankan untuk mengeksplorasi

seleksi fitur atau metode klasifikasi lainnya untuk menemukan pendekatan yang dapat memberikan hasil yang lebih baik dibandingkan dengan metode yang digunakan dalam penelitian ini.

REFERENSI

[1] S. M. Arqawi *et al.*, "Predicting Employee Attrition and Performance Using Deep Learning," *J Theor Appl Inf Technol*, vol. 100, no. 21, pp. 6526–6536, 2022.

[2] I. Jayanto and Benisius, "Analisis Perbandingan Algoritma Decision Tree untuk Prediksi Karyawan dengan Potensi Atrisi di PT. XYZ," *Jurnal Informatika Komputer, Bisnis dan Manajemen*, vol. 22, no. 1, pp. 49–59, 2024, doi: 10.61805/fahma.v22i1.112.

[3] S. Al-Darraji, D. G. Honi, F. Fallucchi, A. I. Abdulsada, R. Giuliano, and H. A. Abdulmalik, "Employee attrition prediction using deep neural networks," *Computers*, vol. 10, no. 11, pp. 1–11, 2021, doi: 10.3390/computers10110141.

[4] M. Atef, D. S. Elzanfaly, and S. Ouf, "Early Prediction of Employee Turnover Using Machine Learning Algorithms 135 Original Scientific Paper," *International journal of electrical and computer engineering systems* , pp. 135–144, 2022.

[5] A. Chourey, S. Phulre, and S. Mishra, "Employee attrition prediction using various machine learning techniques," *The International Journal of Analytical and Experimental Modal Analysis*, vol. XI, no. 2718, pp. 2718–2724, 2019.

[6] A. Qutub, A. Al-Mehmadi, M. Al-Hssan, R. Aljohani, and H. S. Alghamdi, "Prediction of Employee Attrition Using Machine Learning and Ensemble Methods," *Int J Mach Learn Comput*, vol. 11, no. 2, pp. 110–114, 2021, doi: 10.18178/ijmlc.2021.11.2.1022.

[7] N. Ben Yahia, J. Hlel, and R. Colomo-Palacios, "From Big Data to Deep Data to Support People Analytics for Employee Attrition Prediction," *IEEE Access*, vol. 9, pp. 60447–60458, 2021, doi: 10.1109/ACCESS.2021.3074559.

[8] S. K. Setianto and D. Jatikusumo, "Employee Turnover Analysis Using Comparison of Decision Tree and Naive Bayes Prediction Algorithms on K-Means Clustering Algorithms at PT. AT," *Jurnal Mantik*, vol. 4, no. 3, pp. 1573–1581, 2020, [Online]. Available: https://iocscience.org/ejournal/index.php/mantik

[9] B. Prihanto, C. O. Sereati, M. A. Kartawidjaja, and M. Siregar, "Atrition Analysis using XG Boost and Support Vector Machine Algorithm," *Int J Innov Sci Res Technol*, vol. 8, no. 6, pp. 2096–2112, 2023.

[10] M. Chaudhary, L. Gaur, N. Z. Jhanjhi, M. Masud, and S. Aljahdali, "Envisaging Employee Churn Using MCDM and Machine Learning," *Intelligent Automation and Soft Computing*, vol. 33, no. 2, pp. 1009–1024, 2022, doi: 10.32604/iasc.2022.023417.

[11] M. Subhashini and R. Gopinath, "Employee Attrition Prediction in Industry Using Machine Learning Techniques," *International Journal of Advanced Research in Engineering and Technology*, vol. 11, no. 12, pp. 3329–3341, 2020, doi: 10.34218/IJARET.11.12.2020.313.

[12] K. Naz, I. F. Siddiqui, J. Koo, M. A. Khan, and N. M. F. Qureshi, "Predictive Modeling of Employee Churn Analysis for IoT-Enabled Software Industry," *Applied Sciences (Switzerland)*, vol. 12, no. 20, 2022, doi: 10.3390/app122010495.

[13] A. Raza, K. Munir, M. Almutairi, F. Younas, and M. M. S. Fareed, "Predicting Employee Attrition Using Machine Learning Approaches," *Applied Sciences (Switzerland)*, vol. 12, no. 13, 2022, doi: 10.3390/app12136424.

[14] R. Punnoose and P. Ajit, "Prediction of Employee Turnover in Organizations using Machine Learning Algorithms," *International Journal of Advanced Research in Artificial Intelligence*, vol. 5, no. 9, pp. 22–26, 2016, doi: 10.14569/ijarai.2016.050904.

[15] M. Nandal, V. Grover, D. Sahu, and M. Dogra, "Employee Attrition: Analysis of Data Driven Models," *EAI Endorsed Transactions on Internet of Things*, vol. 10, pp. 1–10, 2024, doi: 10.4108/eetiot.4762.

[16] C. Jin, F. Li, S. Ma, and Y. Wang, "Sampling scheme-based classification rule mining method using decision tree in big data environment," *Knowl Based Syst*, vol. 244, p. 108522, 2022, doi: https://doi.org/10.1016/j.knosys.2022.108522.

[17] K. Bhuva and K. Srivastava, "Comparative Study of The Machine Learning Techniques for Predicting The Employee Attrition," *Ijrar*, vol. 5, no. 3, pp. 568–577, 2018, [Online]. Available: www.ijrar.org

[18] G. A. Mursianto, I. M. Falih, M. Irfan, T. Sakinah, and D. S. Prasvita, "Perbandingan Metode Klasifikasi Random Forest dan XGBoost Serta Implementasi Teknik SMOTE pada Kasus Prediksi Hujan," *Jurnal Senamika*, vol. 2, no. 2, pp. 41–50, 2021.

[19] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers and Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[20] S. Kilic, "Chi-square Test," *MEDSURG Nursing*, vol. 28, no. 2, p. 127, 2019, doi: 10.5455/jmood.20160803110534.