Optimasi Model CatBoost dengan Feature Selection dan Hyperparameter Tuning untuk Prediksi Nasabah Bank Potensial

Ahmad Muzakki Eko Fitra Firmandani[1], Ahmad Hudawi AS[2], Abu Tholib[3]
[1,2,3]Fakultas Teknik, Universitas Nurul Jadid Probolinggo, Indonesia
Email: [1]ahmadmuzakki906@gmail.com, [2]ahmad.hudawi@unuja.ac.id, [3]ebuenje@gmail.com

Abstrak - Persaingan ketat di industri perbankan menuntut kemampuan memprediksi nasabah potensial deposito secara akurat dan efisien. Penelitian ini bertujuan meningkatkan akurasi prediksi nasabah potensial deposito dengan mengurangi kompleksitas komputasi dan dimensionalitas data, terutama pada penanganan fitur kategorik. Metode yang diusulkan menggunakan algoritma CatBoost yang mampu menangani data kategorik secara efisien tanpa memerlukan one-hot encoding. Feature selection berbasis feature importance diaplikasikan untuk memilih fitur paling relevan, sementara hyperparameter tuning dengan Hyperopt digunakan untuk mengoptimalkan parameter model CatBoost. Eksperimen pada dataset Bank Marketing dengan 45.211 baris data dan 16 fitur menunjukkan kombinasi CatBoost, feature selection, dan hyperparameter tuning mampu mencapai akurasi 92,8%, sensitivitas 91,0%, dan spesifisitas 94,8% dalam memprediksi nasabah potensial deposito. Pendekatan ini terbukti efektif mengurangi kompleksitas komputasi sekaligus meningkatkan akurasi prediksi nasabah potensial deposito.

Kata Kunci – Prediksi, Fitur Kategorik, CatBoost, Feature Selection, Hyperparameter Tuning.

Abstract - The intense competition in the banking industry demands the ability to accurately and efficiently predict potential deposit customers. This research aims to improve the accuracy of predicting potential deposit customers by reducing computational complexity and data dimensionality, especially in handling categorical features. The proposed method utilizes the CatBoost algorithm, which can handle categorical data efficiently without the need for one-hot encoding. Feature selection based on feature importance is applied to select the most relevant features, while hyperparameter tuning with Hyperopt is used to optimize the parameters of the CatBoost model. Experiments on the Bank Marketing dataset with 45,211 rows of data and 16 features show that the combination of CatBoost, feature selection, and hyperparameter tuning can achieve an accuracy of 92.8%, a sensitivity of 91.0%, and a specificity of 94.8% in predicting potential deposit customers. This approach has proven effective in reducing computational complexity while increasing prediction accuracy of predicting potential deposit customers.

Keywords - Prediction, Categorical Features, CatBoost, Feature Selection, Hyperparameter Tuning.

I. PENDAHULUAN

Di era modern ini, industri perbankan dihadapkan dengan tantangan berupa peningkatan signifikan dalam intensitas persaingan menarik minat nasabah serta dinamika yang semakin kompleks [1]. Kondisi persaingan ketat yang terjadi, mendorong industri perbankan untuk secara proaktif mencari dan mengimplementasikan strategi baru guna mempertahankan dan meningkatkan pangsa pasarnya. Salah satu strategi yang dapat diterapkan adalah dengan meningkatkan kemampuan untuk memprediksi nasabah potensial secara akurat dan komprehensif [2]. Kemampuan prediksi ini menjadi kunci utama bagi bank dalam mengembangkan strategi pemasaran yang efektif, tepat sasaran, serta memudahkan mereka dalam mempromosikan berbagai produk atau instrumen keuangan secara lebih terarah dan efisien kepada kelompok masyarakat yang menjadi target pasar.

Salah satu produk unggulan yang kerap dipromosikan oleh bank adalah deposito. Berdasarkan Undang-Undang Nomor 10 Tahun 1998, deposito didefinisikan sebagai jenis simpanan yang penarikannya hanya dapat dilakukan pada jangka waktu tertentu berdasarkan perjanjian antara nasabah dengan bank[3]. Dengan memanfaatkan kemampuan memprediksi nasabah potensial deposito, bank dapat mengidentifikasi dan memetakan kelompok masyarakat yang memiliki kemampuan finansial serta minat berinvestasi dalam produk deposito dengan lebih tepat. Hal ini memungkinkan bank untuk mempromosikan produk deposito secara lebih efektif kepada target pasar yang relevan.

Pendekatan *machine learning* semakin banyak diadopsi untuk melakukan prediksi nasabah potensial deposito. Hal ini dikarenakan *machine learning* mampu mempelajari dan menganalisis data secara efektif dibandingkan dengan pendekatan konvensional, yang telah terbukti dalam berbagai konteks, seperti pada penelitian [4]–[8].

Dalam kaitannya dengan penelitian yang akan dibahas, beberapa penelitian sebelumnya telah menggunakan *machine learning* untuk memprediksi nasabah potensial deposito. penelitian [9], menggunakan pemilihan fitur berbasis korelasi dan klasifikasi *Multilayer Perceptron Neural Networks* (MLPNN), berhasil mengidentifikasi sejumlah atribut signifikan seperti *duration*, *previous*,

contact, cons.price.idx, month, cons.cof.idx, age, job, marital, dan housing, yang membantu memprediksi nasabah potensial deposito, menghasilkan akurasi prediksi sebesar 80,5%. Sementara penelitian [10] menguji beberapa model klasifikasi, yakni SGD *Classifier*, k-NN, dan *Random Forest*, dengan *Random Forest* mencapai akurasi tertinggi 87,5%. Penelitian lain yang dilakukan oleh [3] menerapkan algoritma klasifikasi *machine learning* seperti *Random Forest, XGBoost, SVC*, dan *Logistic Regression*. Hasilnya menunjukkan model *Random Forest* dan *XGBoost* menjadi model terbaik dalam memprediksi nasabah potensial deposito dengan akurasi mencapai 91,7%.

Namun, dalam penelitian-penelitian tersebut, penanganan data kategorik seringkali memerlukan proses yang dapat meningkatkan dimensionalitas data, terutama saat menghadapi dataset dengan fitur kategorik yang banyak [11]. Proses ini tidak hanya membebani komputasi, tetapi juga berpotensi menurunkan performa dan akurasi model klasifikasi.

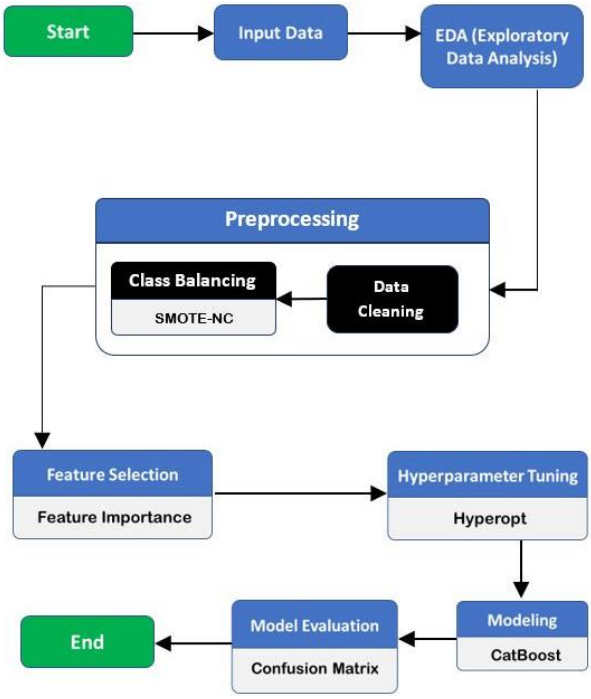
Untuk mengatasi masalah tersebut, penelitian ini mengusulkan penggunaan algoritma *CatBoost* untuk melakukan prediksi nasabah potensial deposito. *CatBoost* merupakan algoritma *machine learning* berbasis gradient boosting yang memiliki kemampuan unik untuk menangani data kategorik secara efisien tanpa perlu melakukan proses *one-hot encoding* yang umum dilakukan pada kebanyakan algoritma lainnya [12]. Dengan kemampuan tersebut, CatBoost diharapkan dapat mengurangi kompleksitas komputasi dan dimensionalitas data, terutama ketika menghadapi dataset dengan fitur kategorik yang banyak. Selain itu, pendekatan dalam penelitian ini juga menggabungkan teknik *feature selection* berbasis *feature importance* untuk memilih fitur yang paling relevan dengan prediksi nasbah potensial depsito, serta *hyperparameter tuning* dengan *Hyperopt* untuk mengoptimalkan parameter model *CatBoost* agar diperoleh performa terbaik.

Melalui pendekatan ini, penelitian diharapkan dapat menghasilkan model prediksi yang tidak hanya meningkatkan akurasi, tetapi juga mengurangi kompleksitas komputasi, sehingga dapat membantu industri perbankan dalam mengembangkan strategi pemasaran deposito yang lebih tepat sasaran dan efektif. Hasil dari penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam meningkatkan efektivitas kampanye pemasaran produk deposito di sektor perbankan, serta memperluas pemahaman tentang penerapan teknik *machine learning* canggih dalam konteks prediksi perilaku nasabah.

II. METODE PENELITIAN

Penelitian ini mengadopsi metode eksperimen untuk menerapkan algoritma *CatBoost* yang dikombinasikan dengan metode *Feature Selection* dan *Hyperparameter Tuning*. Eksperimen dilakukan dengan memanfaatkan *Google Colaboratory* sebagai alat komputasi, dilengkapi dengan berbagai pustaka (*library*) yang dibutuhkan selama proses analisis data dan pembangunan model *machine learning*.

Tahapan penelitian diawali dengan mengunggah dataset yang akan digunakan, kemudian dilanjutkan dengan menganalisis karakteristik data menggunakan teknik *Exploratory Data Analysis* (EDA). Selanjutnya, data akan melalui tahap *preprocessing* sebelum digunakan pada tahap pengujian model. Pada tahap ini, dilakukan proses *data cleaning* dan *class balancing*. Data yang telah diproses tersebut kemudian dilakukan *feature selection* untuk menyeleksi fitur yang relevan. Setelah itu, dilakukan *hyperparameter tuning* untuk mencari kombinasi parameter yang terbaik yang nantinya digunakan dalam model *CatBoost*. Selanjutnya, model *CatBoost* dibangun menggunakan fitur yang telah terseleksi beserta parameter yang telah ditentukan. Gambaran tahapan penelitian yang dilakukan dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian

A. *Input Data*

Dalam penelitian ini, dataset yang digunakan merupakan dataset sekunder dimana sumber datasetnya diperoleh dari website www.kaggle.com yang diunggah oleh Hariharanpavan pada tahun 2022 dengan judul "*Bank Marketing Dataset*". Data tersebut berasal dari kampanye pemasaran langsung melalui telepon yang dilakukan oleh sebuah institusi perbankan di Portugal antara bualn Mei 2008 hingga November 2010. Dataset ini terdiri dari 45.211 baris data dengan 16 atribut independen yang mencakup informasi demografis nasabah, riwayat interaksi dengan bank, serta detail kampanye pemasaran. Atribut-atribut tersebut meliputi *age, job, marital, education, default, balance, housing, loan, contact, day, month, duration, campaign, pdays, previous, dan poutcome*. Sebagai target variabel, digunakan label biner *y* yang menunjukkan apakah nasabah melakukan deposito (*yes*) atau tidak (*no*) sebagai hasil dari kampanye. Data ini dikumpulkan melalui kombinasi catatan transaksi bank dan hasil panggilan

telemarketing. Detail dari dataset tersebut dapat dilihat dalam tabel 1.

Tabel 1. Deskripsi Dataset

No	*Atribut*	*Keterangan*
1	*Age*	Usia (numerik)
2	*Job*	Jenis pekerjaan (kategorik: *"admin."*, *"unknown"*, *"unemployed"*, *"management"*, *"housemaid"*, *"entrepreneur"*, *"student"*, *"blue-collar"*, *"self-employed"*, *"retired"*, *"technician"*, *"services"*)
3	*Marital*	Status pernikahan (kategorik: *"married"*, *"divorced"*, *"single"*)
4	*Education*	Tingkat pendidikan (kategorikal: *"unknown"*, *"secondary"*, *"primary"*, *"tertiary"*)
5	*Default*	Apakah nasabah memiliki Riwayat kredit macet (kategorik: *"yes"*, *"no"*)
6	*Balance*	Saldo rekening dalam mata uang euro (numerik)
7	*Housing*	Apakah nasabah memiliki kredit pemilikan rumah (kategorik: *"yes"*, *"no"*)
8	*Loan*	Apakah nasabah memiliki pinjaman lain (kategorik: *"yes"*, *"no"*)
9	*Contact*	Media yang digunakan untuk menghubungi nasabah tersebut (kategorik: *"unknown"*, *"telephone"*, *"cellular"*)
10	*Day*	Hari terakhir saat nasabah tersebut dihubungi (numerik)
11	*Month*	bulan saat nasabah tersebut dihubungi (kategorik: *"jan"*, *"feb"*, *"mar"*, ..., *"nov"*, *"dec"*)
12	*Duration*	Durasi kontak dengan nasabah saat melakukan promosi(numerik)
13	*Campaign*	Jumlah promosi pemasaran yang diterima nasabah(numerik)
14	*Pdays*	Jumlah hari yang berlalu setelah orang tersebut dihubungi sebelum kampanye ini dilakukan (numerik, -1 menunjukkan tidak dihubungi sebelumnya)
15	*Previous*	Jumlah kali nasabah tersebut telah dihubungi sebelum kampanye ini (numerik)
16	*Putcome*	Hasil kampanye pemasaran terakhir (kategorik: *"unknown"*, *"other"*, *"failure"*, *"success"*)
17	*Y*	Apakah Nasabah berlangganan deposito atau tidak (kategorik: *"yes"*, *"no"*)

B. EDA (Exploratory Data Analysis)

EDA *(exploratory data analysis)* merupakan sebuah proses untuk memahami karakteristik pola yang ada pada data melalui visualisasi data, sehingga dapat dilakukan penanganan yang tepat untuk mencapai hasil optimal[13]. Dengan melakukan pendekatan menggunakan EDA, kita dapat mengetahui *insight* dari data yang berupa pola distribusi data, keberadaan outlier, pada data[14]. EDA digunakan sebelum algoritma apa pun dipilih baik untuk untuk data yang bersifat struktural atau pun bersifat acak.

Tahap EDA yang dilakukan dalam penelitian ini yaitu visualisasi data berupa grafik menggunakan *Box Plot*, *Count Plot*. *Box Plot* digunakan untuk memvisualisasikan nilai minimum, kuartil pertama, median, kuartil ketiga, nilai maximum, dan mengidentifikasi adanya outlier pada sebuah variabel. *Count plot* digunakan untuk menampilkan distribusi frekuensi dari data kategorikal. Grafik ini menunjukkan jumlah kemunculan masing-masing data dalam suatu variabel kategorik

C. Preprocessing

Preprocessing memiliki peran krusial dalam meningkatkan kualitas, keandalan, dan konsistensi data, yang pada akhirnya meningkatkan kinerja pembelajaran mesin. Melalui penghapusan nilai outlier, pengisian nilai yang hilang, dan penghapusan sampel duplikat, *preprocessing* tidak hanya memperbaiki keakuratan model, tetapi juga memungkinkan algoritma pembelajaran mesin untuk lebih efektif membaca, menggunakan, dan menginterpretasi dataset[15].

Terdapat dua metode preprocessing yang digunakan dalam penelitian ini yaitu *data cleaning*, dan *class balancing*. Metode tersebut dilakukan berdasarkan hasil dari proses EDA yang telah dilakukan.

1) Data Cleaning

Data cleaning merupakan proses yang diperlukan untuk menangani nilai yang hilang, gangguan, dan inkonsistensi dalam data[16]. Proses ini bertujuan untuk meningkatkan kualitas dan integritas data yang akan digunakan dalam pelatihan model, sehingga diharapkan dapat menghasilkan

performa dan akurasi prediksi yang lebih baik. Pada tahap ini dilakukan pengecekan terhadap data untuk mendeteksi keberadaan nilai yang hilang, duplikasi data, dan juga penghapusan data ekstrem (outlier) berdasarkan informasi yang diperoleh dari analisis data dalam proses EDA.

2) Class balancing

Class Balancing dilakukan untuk menyeimbangkan distribusi kelas dalam dataset dimana hal ini memastikan bahwa setiap kelas dalam dataset memiliki representasi yang seimbang, sehingga mencegah model menjadi bias terhadap kelas mayoritas[17]. Dalam penelitian ini, teknik *oversampling* menggunakan SMOTE-NC digunakan untuk menyeimbangkan kelas.

SMOTE-NC adalah sebuah teknik *oversampling* yang memungkinkan kita untuk membuat sampel-sampel tambahan dari kelas minoritas dengan cara menciptakan data sintetis baru dari sampel-sampel yang sudah ada, baik yang bersifat numerik maupun kategorikal [18].teknik ini dipilih berdasarkan karakteristik dataset yang memiliki banyak data yang bertipe kategorik.

D. Feature Selection

Feature selection adalah proses yang digunakan untuk mencari fitur - fitur yang paling relevan dengan kelas target [19]. Hal ini dilakukan untuk mengurangi biaya komputasi, kebutuhan memori dari model prediksi, dan untuk meningkatkan generalisasi model prediksi dengan memastikan hanya fitur-fitur yang berguna yang digunakan [20].

Pada penelitian ini, *feature selection* dilakukan dengan memanfaatkan fungsi *"feature importance"* yang dimiliki oleh *CatBoost*. Fungsi ini mengevaluasi tingkat kepentingan masing-masing fitur terhadap target setelah melatih dataset menggunakan model *CatBoost*. Berikut konsep persamaan untuk menghitung nilai kepentingan fitur :

$$\sum_{tree,leafs_F} \begin{matrix} feature\_importance_F = \\ (v_1 - avr)^2 . c_1 \\ + (v_2 - avr)^2 . c_2 \end{matrix} \quad (1)$$

Dimana:

- $avr = \frac{v_1 . c_1 + v_2 . c_2}{c_1 + c_2}$
- c_1, c_2 mewakili total bobot objek di daun kiri dan kanan secara berturut-turut. Bobot ini sama dengan jumlah objek di setiap daun jika bobot tidak ditentukan untuk dataset
- v_1, v_2 mewakili nilai formula di daun kiri dan kanan secara berturut-turut

E. Hyperparameter Tuning

Hyperparameter Tuning merupakan teknik yang digunakan untuk menyesuaikan parameter-parameter yang digunakan dalam algoritma *Machine Learning/Deep Learning*, dengan tujuan meningkatkan kinerja model secara signifikan [21]. Penelitian ini menggunakan *Hyperopt* untuk melakukan *hyperparameter tuning*.

Hyperopt merupakan sebuah *library Python* yang dikembangkan untuk mengoptimalkan hyperparameter dalam algoritma pembelajaran mesin [22]. *Treestructured Parzen Estimator* (TPE) merupakan metode yang digunakan oleh Hyperopt untuk proses *hyperparameter tuning*, dimana probabilitas hyperparameter didefinisikan menggunakan dua densitas berikut:

$$p(x|y) \begin{cases} l(x) \ \ if \ y < y^* \\ g(x) \ \ if \ y \geq y^* \end{cases} \quad (2)$$

Dimana :

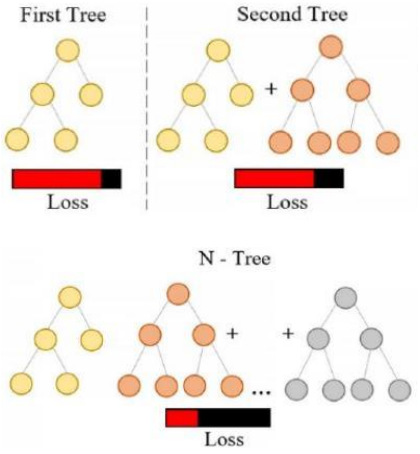
- $l(x)$ merupakan densitas yang terbentuk dengan menggunakan pengamatan $\{x^{(i)}\}$ di mana nilai kerugian yang sesuai $f(x^{(i)})$ lebih rendah dari y^*
- $g(x)$ merupakan densitas yang terbentuk dengan menggunakan pengamatan yang tersisa

Algoritma TPE memilih y^* sebagai kuantil γ dari nilai y yang diamati, sehingga $p(y < y^*) = \gamma$. Konfigurasi *hyperparameter* yang menghasilkan nilai tertinggi yang lebih rendah dari $l(x)$ atau $g(x)$akan di evaluasi pada fungsi tujuan.

F. Modeling

Modeling adalah tahapan dimana algoritma diterapkan untuk mencari, mengidentifikasi, dan membuat pola yang akan diterapkan pada data penelitian[23]. Dalam penelitian ini, algoritma yang digunakan yaitu *CatBoost*.

CatBoost adalah sebuah algoritma hasil pengembangan dari GBDT *(Gradient Boosting Decision Tree)* oleh perusahaan teknologi Rusia bernama Yandex pada tahun 2017[24]. Tujuan utama dari algoritma ini adalah untuk menggabungkan pembelajaran yang lemah dengan cara meminimalkan fungsi kerugian agar dapat mencapai model yang optimal [25]. Proses atau mekanisme dari algoritma *CatBoost* tersebut dijabarkan melalui Gambar 2.



Gambar 2. Mekanisme *CatBoost*

CatBoost memiliki kemampuan khusus dimana secara otomatis menangani data kategorik dengan menggunakan metode statistik. Dengan mengoptimalkan parameter masukan, *CatBoost* mencegah *overfitting* data tanpa memerlukan pengolahan khusus terhadap karakteristik kategori[26]. Selain itu, *CatBoost* melakukan permutasi acak pada data kategorik daripada penggantian biner, dengan menghitung rata-rata setiap label[27].

G. Model Evaluation

Model evaluation adalah proses melakukan sebuah evaluasi untuk mengukur kinerja dari model yang dijalankan. Pada penelitian ini, *confusion matrix* digunakan untuk mengevaluasi kinerja dari model *CatBoost*.

Confusion matrix merupakan tabel yang menunjukkan jumlah data uji yang diklasifikasikan dengan benar dan salah yang biasanya digunakan untuk mengevaluasi seberapa efektif suatu sistem[28]. *Confusion matrix* memiliki empat istilah yang digunakan dalam penggunaannya[29], yaitu:

1. *True Positive* (TP): Jumlah data yang teridentifikasi sebagai ’Pelanggan Potensial’ dan hasil prediksinya ’Pelanggan Potensial’.
2. *True Negative* (TN): Jumlah data yang teridentifikasi sebagai ’Pelanggan Tidak Potensial’ dan hasil prediksinya ’Pelanggan Tidak Potensial’.
3. *False Positive* (FP): Jumlah data yang teridentifikasi sebagai ’Pelanggan Tidak Potensial’ tetapi hasil prediksinya ’Pelanggan Potensial’.
4. *False Negative* (FN): Jumlah data data yang teridentifikasi sebagai ’Pelanggan Potensial’ tetapi hasil prediksinya ’Pelanggan Tidak Potensial’.

Dengan menggunakan empat istilah tersebut, *confusion matrix* akan melakukan penghitungan *Accuracy, Sensitivity*, dan, *Specificity* [30], dimana :

1. *Accuracy* : Seberapa banyak data yang diprediksi dengan benar.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

2. *Sensitivity* : seberapa baik model dalam memprediksi nilai TP.

$$Sensitivity = \frac{TP}{TP+FN} \quad (4)$$

3. *Specificity* : seberapa baik model dalam memprediksi nilai TN

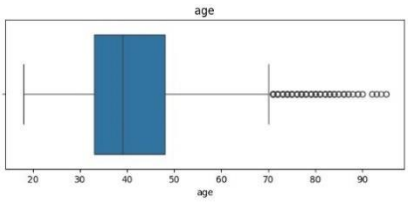
$$Specificiy = \frac{TN}{TN+FP} (5)$$

III. HASIL DAN PEMBAHASAN

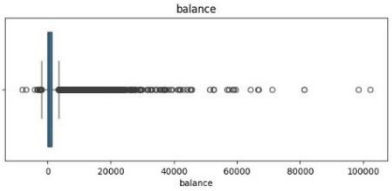
A. EDA (Exploratory Data Analysis)

EDA dilakukan dengan menampilkan visualisasi dari dataset berupa visualisasi *Box Plot* dan *Count Plot* untuk menemukan outlier yang terdapat dalam dataset dan mengetahui distribusi data yang ada pada dataset terutama pada kelas target.

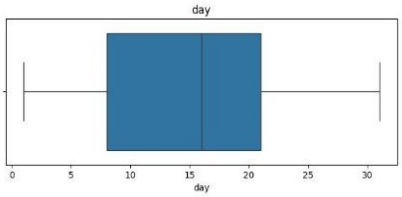
1) Box Plot



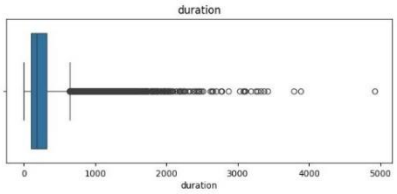
(a) Fitur age



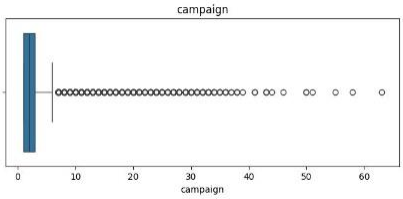
(b) Fitur balance



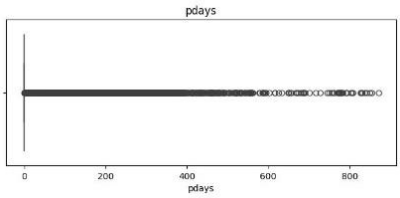
(c) Fitur day



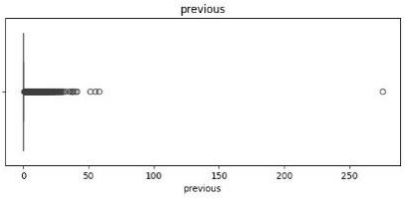
(d) Fitur duration



(e) Fitur campaign



(f) Fitur pdays

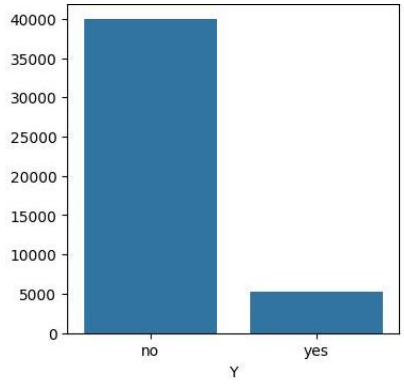


(g) Fitur previous

Gambar 3. Visualisasi *Box Plot* Fitur Numerik

Dalam visualisasi data menggunakan *Box Plot* yang di tampilkan oleh gambar 3, ditemukan beberapa hal menarik. Yaitu, di kolom *"age",* ada beberapa orang yang berusia sekitar 100 tahun. Ini menunjukkan adanya orang-orang yang berusia cukup tua dalam dataset tersebut, yang bisa penting untuk memahami perilaku dan keputusan keuangannya. Selain itu, di dalam fitur *"duration"*, ada nilai negatif yang aneh dalam satuan detik. Nilai tersebut harus diperiksa lebih lanjut untuk memastikan bahwa ini tidak mungkin terjadi dan memperbaiki data tersebut. Di dalam fitur *"previous"*, ada nilai yang sangat tinggi, mendekati 300. Karena nilai yang sangat tinggi bisa menjadi tidak realistis dalam konteks jumlah kontak sebelumnya dengan nasabah. Pada umumnya, jumlah kontak sebelumnya tidak seharusnya terlalu tinggi seperti itu.

2) Count Plot



Gambar 4. Distribusi Kelas Fitur "Y"

Dari hasil visualisasi *Count Plot* yang telah ditampilkan oleh gambar 4, ditemukan bahwa terdapat ketidakseimbangan dalam distribusi kelas variabel target. Lebih banyak nasabah yang tidak melakukan deposito dibandingkan dengan yang melakukan deposito. Oleh karena itu, langkah selanjutnya adalah menerapkan metode *oversampling* untuk menyeimbangkan distribusi tersebut.

B. Preprocessing

1) Data Cleaning

Fokus utama pada tahap data cleaning ini adalah menangani keberadaan outlier atau nilai-nilai ekstrem yang berpotensi mengganggu proses pembelajaran model. Berdasarkan analisis yang dilakukan, ditemukan adanya outlier pada variabel durasi kontak (*duration*) dan jumlah kontak sebelumnya (*previous*). Tabel 2 menunjukkan jumlah data yang dihapus karena mengandung nilai outlier pada kedua fitur tersebut.

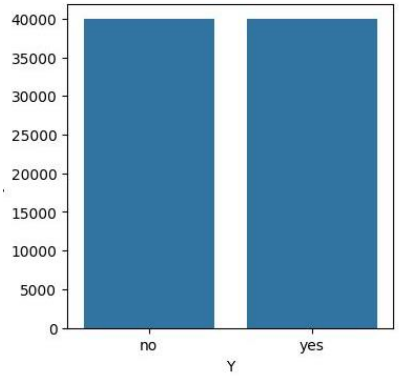
Tabel 2. Data yang Dihapus pada Tahap Data Cleaning

No	*Alasan*	*Jumlah Data*
1	Nilai negatif pada fitur *"duration"*	3
2	Nilai *"previous"* lebih dari 100	1

Sebanyak 3 baris data dihapus karena mengandung nilai negatif pada fitur *"duration"*, dan 1 baris data dihapus karena memiliki nilai lebih dari 100 pada fitur *"previous"*. Nilai-nilai tersebut dianggap sebagai outlier yang tidak valid dalam konteks pemasaran bank. Setelah penghapusan outlier, dataset yang tersisa berjumlah 45.207 baris dari total 45.211 baris data awal. Keberadaan outlier ini dapat disebabkan oleh kesalahan penginputan data, kondisi khusus, atau data anomali. Dalam kasus ini, tidak ditemukan masalah terkait *missing value* atau data duplikat, sehingga *data cleaning* hanya berfokus pada penanganan outlier pada fitur *"duration"* dan *"previous"*.

2) Class Balancing

Sebelum dilakukan class balancing, distribusi data antara kelas *"yes"* (nasabah yang melakukan deposito) dan *"no"* sangat tidak seimbang seperti yang ditunjukkan pada Gambar 4. Jumlah sampel pada kelas *"no"* jauh lebih banyak dibandingkan dengan kelas *"yes"*. Untuk menyeimbangkan kelas-kelas tersebut, digunakan teknik *oversampling* dengan SMOTE-NC. Berikut hasil dari penyeimbangan kelasnya :



Gambar 5. Kelas fitur "Y" Hasil Oversampling

Setelah diterapkan teknik oversampling dengan SMOTE-NC, jumlah sampel pada kelas minoritas (kelas *"yes"*) meningkat secara signifikan sehingga rasio antara kedua kelas menjadi seimbang seperti yang ditunjukkan pada bagian kanan Gambar 5. Dengan melakukan *oversampling* pada kelas minoritas, diharapkan model dapat mempelajari dan memprediksi kedua kelas dengan performa yang baik dan seimbang

C. Feature Selection

Selama proses *feature selection* menggunakan fungsi *"feature importance"*, didapatkan sejumlah 10 fitur dengan nilai kepentingan tertinggi terhadap target yang dianggap paling relevan dan berpengaruh dalam memprediksi apakah seorang nasabah akan melakukan deposito atau tidak. Berikut detail dari fitur - fitur tersebut:

Tabel 3. 10 Fitur Hasil *Feature Selection*

No	*Fitur*	*Nilai Kepentingan*
1	Duration	23.034733423174092
2	Month	13.002639478005829
3	Contact	9.552350988843171
4	Day	8.613479074516862
5	Poutcome	8.010717868596675
6	Balance	6.790500599865537
7	Housing	6.014267236653641
8	Pdays	4.891514503880521
9	Age	4.166553246346829
10	Job	3.9649717785453573

Oleh karena itu model selanjutnya dibangun menggunakan 10 fitur berdasarkan tabel 3 tersebut sehingga pelatihan model menjadi lebih efisien tanpa terpengaruh oleh fitur-fitur yang kurang penting.

D. Hyperparameter Tuning

Proses *hyperparameter tuning* menggunakan *library Hyperopt* dilakukan dengan menentukan terlebih dahulu ruang pencarian (*search space*) untuk setiap hyperparameter yang akan di-tuning. *Search space* ini membatasi rentang nilai yang akan dicari oleh algoritma tuning dalam mengoptimalkan performa model. Gambar 6 menunjukan *search space* yang digunakan dalam proses *hyperparameter tuning* pada gambar 6 :

```
# Mendefinisikan ruang pencarian hyperparameter
space = {
    'max_depth': hp.quniform('max_depth', 6,10, q=1),
    'l2_leaf_reg': hp.uniform('l2_leaf_reg', 0.1, 10),
    'learning_rate': hp.uniform('learning_rate', 0.01, 0.5),
    'subsample': hp.uniform('subsample', 0.5, 1.0)
}
```

Gambar 6. Ruang Pencarian *Hyperparameter*

Berdasarkan gambar diatas, *search space* yang digunakan adalah sebagai berikut:

Parameter max_depth (kedalaman maksimum dari setiap pohon yang dibangun), dengan rentang nilai yang dieksplorasi antara 6 hingga 10. Parameter l2_leaf_reg (tingkat regularisasi L2 pada leaf dalam setiap pohon), dengan nilai yang diuji berkisar antara 0,1 hingga 10. Learning rate (tingkat pembelajaran model), divariasikan dari 0,01 hingga 0,5. Terakhir, parameter subsample (persentase dataset yang digunakan untuk membangun setiap pohon), dengan rentang nilai dari 0,5 hingga 1,0.

Setelah menetapkan *search space* tersebut, proses tuning dilakukan dengan mencari kombinasi nilai *hyperparameter* terbaik yang dapat mengoptimalkan metrik evaluasi yang digunakan, yaitu *logloss*. Setelah melalui 20 iterasi tuning, diperoleh kombinasi nilai *hyperparameter* terbaik pada tabel 4 berikut:

Tabel 4. *Hyperparameter* Terbaik *CatBoost*

No	*Hyperparameter*	*Nilai Terbaik*
1	max_depth	9.0
2	12_leaf_reg	2.1509964758438764
3	learning_rate	0.15324979289799676
4	subsample	0.6258962666318186

Dengan menggunakan kombinasi *hyperparameter* terbaik ini, diharapkan model *CatBoost* yang dibangun dapat memberikan performa klasifikasi yang optimal dalam memprediksi keputusan nasabah untuk melakukan deposito atau tidak.

E. Modeling

Setelah melalui proses *preprocessing*, *feature selection*, dan *hyperparameter tuning*, langkah selanjutnya adalah membangun model klasifikasi menggunakan algoritma *CatBoost*. Sebelum membangun model, dilakukan identifikasi fitur kategorik yang ada pada 10 fitur terpilih sebelumnya pada tabel 4. Berikut adalah fitur-fitur bertipe data kategorik yang teridentifikasi dalam tabel 5:

Tabel 5. Fitur Kategorik yang Teridentifikasi

No	*Fitur*	*Tipe Data*
1	Month	Object
2	Contact	Object
3	Poutcome	Object
4	Housing	Object
5	Job	Object

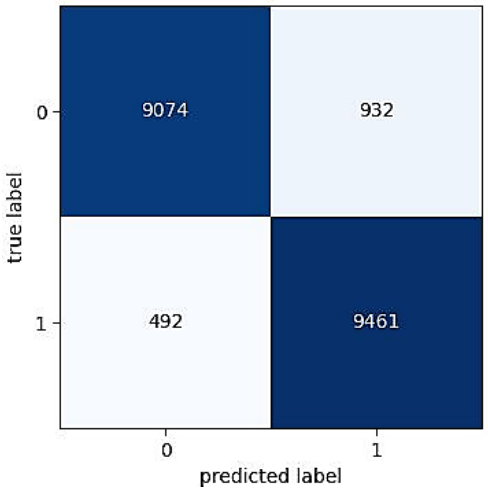
Fitur kategorik yang diidentifikasi kemudian digunakan sebagai nilai parameter "*cat_features*" dalam model *CatBoost*. Parameter ini membantu memberitahukan model *CatBoost* mengenai fitur mana yang bertipe kategorik, sehingga model dapat mengeksekusi langsung fitur kategorik tersebut tanpa perlu melakukan *one-hot-encoding*.

Selanjutnya adalah membagi dataset menjadi data latih (*train*) dan data uji (*test*) dengan rasio 75:25. Data latih yang terdiri dari 59.877 baris digunakan untuk melatih model, sementara 19.959 baris data uji disisihkan untuk

mengevaluasi performa model. Barulah, model *CatBoost* dibangun dengan menggunakan kombinasi hyperparameter terbaik yang diperoleh dari proses *hyperparameter tuning* sebelumnya.

F. Model Evaluation

Untuk mengevaluasi performa model *CatBoost* yang dikombinasikan dengan *feature selection* dan *hyperparameter tuning*, digunakan *confusion matrix*. *Confusion matrix* memberikan gambaran tentang data yang diklasifikasikan dengan benar dan salah oleh model. Berikut adalah *confusion matrix* untuk model *CatBoost* dalam memprediksi kemungkinan nasabah berlangganan deposito, seperti yang ditunjukkan pada gambar 7:



Gambar 7. *Confusion Matrix* Hasil *Modeling*

Dari *confusion matrix* di atas, dapat dilihat bahwa *True Positive* (TP) berjumlah 9461, Menunjukkan bahwa model berhasil memprediksi dengan benar 9461 nasabah yang benar-benar akan berlangganan deposito. Angka ini menunjukkan keberhasilan kampanye pemasaran yang tepat sasaran, memungkinkan bank untuk merencanakan strategi retensi yang lebih baik untuk nasabah yang diprediksi berlangganan.

False Positive (FP) berjumlah 492, menunjukkan terdapat 492 nasabah yang diprediksi akan berlangganan deposito namun ternyata tidak. Walaupun ini menunjukkan adanya biaya pemasaran yang sia-sia, angkanya relatif kecil dibandingkan dengan TP, yang menunjukkan bahwa model ini cukup efisien dalam mengurangi biaya pemasaran yang tidak efektif. Bank dapat menggunakan informasi ini untuk lebih memfokuskan kampanye pemasaran mereka dan mengurangi jumlah nasabah yang tidak tertarik.

Selain itu, *False Negative* (FN) berjumlah 932, yang mencerminkan ada potensi pendapatan yang terlewatkan karena nasabah ini mungkin tidak mendapatkan penawaran yang seharusnya dapat menarik minat mereka untuk berlangganan lebih awal. Dengan mengevaluasi dan memahami kasus-kasus FN ini, bank dapat meningkatkan strategi pemasaran untuk lebih efektif menjangkau dan mengkonversi nasabah potensial ini, untuk meningkatkan pendapatan secara keseluruhan.

Terakhir, *True Negative* (TN) berjumlah 9074, yaitu nasabah sebanyak 9074 diprediksi tidak akan berlangganan dan memang tidak berlangganan. Hal ini menunjukkan keberhasilan model dalam mengidentifikasi nasabah yang tidak tertarik pada penawaran deposito, menghindari pengeluaran yang tidak perlu dalam pemasaran kepada mereka. Bank dapat menggunakan informasi ini untuk mengalokasikan sumber daya pemasaran lebih efisien, fokus pada segmen nasabah yang lebih mungkin untuk merespons penawaran lain atau mempertahankan produk yang sudah dimiliki.

Berdasarkan hasil *confusion matrix* tersebut maka diperoleh nilai *Accuracy* 92,8%, *Sensitivity* 91,0%, dan *Specificity* nya 94,8%. Hasil evaluasi ini menunjukkan bahwa penggunaan *feature selection* dan *hyperparameter tuning* dapat mengurangi kompleksitas dataset serta mampu menghasilkan nilai akurasi yang cukup tinggi dari model *CatBoost*.

Hasil penelitian ini sangat berguna bagi bank dalam mengelola produk deposito. Dengan akurasi model 92,8%, bank bisa memasarkan deposito lebih cerdas, menargetkan nasabah yang berpotensi tertarik, menghemat biaya, dan memberikan layanan lebih personal. Staf bank dapat segera mengetahui minat nasabah terhadap deposito, sehingga pelayanan menjadi lebih efisien.

Model ini juga membantu bank merancang produk deposito yang lebih menarik dan sesuai kebutuhan nasabah, serta mempertahankan nasabah dengan memberikan penawaran khusus bagi mereka yang berpotensi menarik depositonya. Ini penting untuk menjaga hubungan jangka panjang dengan nasabah.

Selain itu, model ini mendukung pengambilan keputusan bisnis yang lebih baik, seperti memperkirakan dana masuk dari deposito dan merencanakan penggunaan dana tersebut dengan lebih efisien. Namun, model ini hanyalah alat bantu dan perlu digunakan dengan bijak, serta diperbarui secara rutin agar tetap akurat. Dengan pemanfaatan yang tepat, bank dapat meningkatkan layanan, kepuasan nasabah, dan keuntungan.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat disimpulkan:

Metode *CatBoost* telah terbukti memiliki kelebihan dalam mengolah data kategorik dan dataset besar secara efisien, tanpa memerlukan proses *one-hot encoding*. Kunci keberhasilan pendekatan ini adalah kemampuannya untuk mengelola berbagai jenis data dengan cepat dan efektif, menjadikannya cocok untuk analisis data berskala besar. Selain itu, Penggunaan teknik *feature selection* dan *hyperparameter tuning* membantu mengurangi dimensi data dengan memilih fitur-fitur penting, mengurangi kompleksitas model, serta menghasilkan akurasi prediksi tinggi, yaitu 92,8% untuk akurasi, 91,0% untuk sensitivitas, dan 94,8% untuk spesifisitasnya. Keberhasilan teknik ini berkontribusi cukup besar dalam mengidentifikasi nasabah

potensial deposito dengan lebih akurat dan efisien. Ini menunjukkan potensi besar penggunaan *CatBoost* dalam aplikasi bisnis lain yang memerlukan analisis cepat dan tepat pada data berskala besar dengan kombinasi tipe data yang beragam.

B. Saran

Adapun beberapa saran untuk berbagai peluang menarik yang dapat dikembangkan dan dieksplorasi lebih lanjut adalah sebagai berikut:

Pertama, penelitian tentang interpretabilitas model CatBoost perlu dilakukan. Meskipun CatBoost menunjukkan performa yang baik, pemahaman yang lebih baik tentang bagaimana model ini membuat keputusan dapat meningkatkan kepercayaan dan adopsinya dalam aplikasi bisnis yang kritis. Kedua, eksplorasi integrasi CatBoost dengan teknik deep learning dapat menjadi arah penelitian yang menjanjikan. Ini bisa mencakup pengembangan model hybrid yang menggabungkan kekuatan CatBoost dalam menangani data kategorik dengan kemampuan deep learning dalam mengolah data tidak terstruktur seperti teks atau gambar. Terakhir, penelitian dapat diarahkan pada pengembangan metode feature selection dan hyperparameter tuning yang lebih canggih dan otomatis untuk CatBoost. Ini dapat mencakup penggunaan teknik optimasi berbasis evolusi atau pendekatan meta-learning untuk menemukan konfigurasi optimal secara lebih efisien.

DAFTAR PUSTAKA

[1] V. Fejza Ademi, A. Avdullahi, Q. Tmava, and E. Durguti, "Analysis of the Banking Sector Competition in Kosovo," *Stud. Univ. „Vasile Goldis" Arad – Econ. Ser.*, vol. 32, pp. 2022–2054, 2022.

[2] K. M. Sagar, "Customer Deposit Forecasting - Optimizing Deposit Predictions through CRM and Machine Learning Integration," *INTERANTIONAL J. Sci. Res. Eng. Manag.*, vol. 07, pp. 1–11, 2023.

[3] M. R. A. Riyyasy, W. N. Aghniya, and H. Tantyoko, "Penerapan Algoritma Machine Learning Untuk Memprediksi Term Deposit Nasabah Perbankan," *LEDGER J. Inform. Inf. Technol.*, vol. 2, no. 3, pp. 145–156, 2023.

[4] A. Hudawi, N. Octavia, A. Elfandiono, A. B. Setiawan, A. A. Ghafur, and A. E. Susanto, "Klasifikasi Pemahaman Santri dalam Pembelajaran Kitab Kuning Menggunakan Algoritma c4. 5. Pohon keputusan (Decision Tree) di Pondok Pesantren Nurul Jadid," *TRILOGI J. Ilmu Teknol. Kesehatan, dan Hum.*, vol. 2, no. 3, pp. 266–269, 2021.

[5] W. J. Shudiq, A. H. As, and M. F. Rahman, "Penentuan Metode Terbaik Dalam Menentukan Jenis Pohon Pisang Menurut Tekstur Daun (Metode K-NN dan SVM)," *J. Teknol. dan Manaj. Inform.*, vol. 6, no. 2, pp. 128–136, 2020.

[6] A. Tholib, N. K. Agusmawati, and F. Khoiriyah, "Prediksi Harga Emas Menggunakan Metode Lstm Dan Gru," *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3, 2023.

[7] F. N. Fajri, A. Tholib, and W. Yuliana, "Application of Machine Learning Algorithm for Determining Elective Courses in Informatics Study Program," *J. Tek. Inform. dan Sist. Inf.*, vol. 8, no. 3, pp. 485–496, 2022.

[8] V. V. Putri, A. Tholib, and C. Novia, "DETEKSI KAGGLE BOT ACCOUNT MENGGUNAKAN DEEP NEURAL NETWORKS," *NJCA (Nusantara J. Comput. Its Appl.*, vol. 8, no. 1, pp. 13–21, 2023.

[9] A. N. Puteri, A. Arizal, and A. D. Achmad, "Feature Selection Correlation-Based pada Prediksi Nasabah Bank Telemarketing untuk Deposito," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 20, no. 2, pp. 335–342, 2021.

[10] A. M. Zaki, N. Khodadadi, W. H. Lim, and S. K. Towfek, "Predictive Analytics and Machine Learning in Direct Marketing for Anticipating Bank Term Deposit Subscriptions," *Am. J. Bus. Oper. Res.*, vol. 11, no. 1, pp. 79–88, 2024.

[11] P. Cerda and G. Varoquaux, "Encoding High-Cardinality String Categorical Variables," *IEEE Trans. Knowl. Data Eng.*, vol. PP, p. 1, 2020.

[12] D. Wang and H. Qian, "CatBoost-Based Automatic Classification Study of River Network," *ISPRS International Journal of Geo-Information*, vol. 12, no. 10, 2023.

[13] I. Cholissodin and A. A. Soebroto, "AI , Machine Learning & Deep Learning (Teori & Implementasi)," 2021.

[14] K. Sahoo, A. K. Samal, J. Pramanik, and S. K. Pani, "Exploratory data analysis using Python," *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 12, pp. 4727–4735, 2019.

[15] S.-A. Alexandropoulos, S. Kotsiantis, and M. Vrahatis, "Data preprocessing in predictive data mining," *Knowl. Eng. Rev.*, vol. 34, p. e1, 2019.

[16] N. P. A. Widiari, I. Suarjaya, and D. P. Githa, "Teknik Data Cleaning Menggunakan Snowflake untuk Studi Kasus Objek Pariwisata di Bali," *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, vol. 8, no. 2, p. 137, 2020.

[17] F. A. Rafrastara, C. Supriyanto, A. Amiral, S. R. Amalia, M. D. Al Fahreza, and F. Ahmed, "Performance Comparison of k-Nearest Neighbor Algorithm with Various k Values and Distance Metrics for Malware Detection," *J. MEDIA Inform. BUDIDARMA*, vol. 8, no. 1, pp. 450–458, 2024.

[18] M. Mukherjee and M. Khushi, "SMOTE-ENC: A novel SMOTE-based method to generate synthetic data for nominal and continuous features," *Appl. Syst. Innov.*, vol. 4, no. 1, p. 18, 2021.

[19] R. E. Nogales and M. E. Benalcázar, "Analysis and Evaluation of Feature Selection and Feature Extraction Methods," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, p. 153, 2023.

[20] J. Bach and K. Böhm, "Alternative feature selection with user control," *Int. J. Data Sci. Anal.*, pp. 1–23, 2024.

[21] A. E. Minarno, M. H. C. Mandiri, and M. R. Alfarizy, "Klasifikasi COVID-19 menggunakan Filter Gabor dan CNN dengan Hyperparameter

Tuning," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 9, no. 3, p. 493, 2021.

[22] J. Bergstra, D. Yamins, and D. D. Cox, "Hyperopt: A Python Library for Optimizing the Hyperparameters of Machine Learning Algorithms.," *SciPy*, vol. 13, p. 20, 2013.

[23] M. A. Hasanah, S. Soim, and A. S. Handayani, "Implementasi CRISP-DM Model Menggunakan Metode Decision Tree dengan Algoritma CART untuk Prediksi Curah Hujan Berpotensi Banjir," *J. Appl. Informatics Comput.*, vol. 5, no. 2, pp. 103–108, 2021.

[24] W. Xiao, C. Wang, J. Liu, M. Gao, and J. Wu, "Optimizing Faulting Prediction for Rigid Pavements Using a Hybrid SHAP-TPE-CatBoost Model," *Appl. Sci.*, vol. 13, p. 12862, Nov. 2023.

[25] J. T. Hancock and T. M. Khoshgoftaar, "CatBoost for big data: an interdisciplinary review," *J. big data*, vol. 7, no. 1, p. 94, 2020.

[26] M. Saber *et al.*, "Examining LightGBM and CatBoost models for wadi flash flood susceptibility prediction," *Geocarto Int.*, vol. 37, no. 25, pp. 7462–7487, 2022.

[27] P. S. Kumar, A. Kumari, S. Mohapatra, B. Naik, J. Nayak, and M. Mishra, "CatBoost ensemble approach for diabetes risk prediction at early stages," in *2021 1st Odisha International Conference on Electrical Power Engineering, Communication and Computing Technology (ODICON)*, pp. 1–6 ,2021.

[28] A. A. Ibrahim, R. L. Ridwan, M. M. Muhammed, R. O. Abdulaziz, and G. A. Saheed, "Comparison of the CatBoost classifier with other machine learning methods," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 11, 2020.

[29] M. D. Purbolaksono, D. T. B. Pratama, and F. Hamzah, "Perbandingan Gini Index dan Chi Square pada Sentimen Analsis Ulasan Film menggunakan Support Vector Machine Classifier," *JEPIN (Jurnal Edukasi dan Penelit. Inform.*, vol. 9, no. 3, pp. 528–534, 2023.

[30] A. Jiménez-Cordero and S. Maldonado, "Automatic feature scaling and selection for support vector machine classification with functional data," *Appl. Intell.*, vol. 51, no. 1, pp. 161–184, 2021.