
Prediksi Fluktuasi Harga Bitcoin Dengan Menggunakan Random Forest Classifier

¹Chang Hung, ²Jacky Filbert Wijaya, ³Victor, ⁴Irpan Adiputra Pardosi, ⁵Frans Mikael Sinaga
^{1,2,3,4,5}Universitas Mikroskil, Jl. Thamrin No.124 Medan, (061) 4573767
^{1,2,3,4,5}Fakultas Informatika, Teknik Informatika, Universitas Mikroskil, Medan
e-mail: 191112652@students.mikroskil.ac.id, 19111606@students.mikroskil.ac.id,
19111584@students.mikroskil.ac.id, irpan@mikroskil.ac.id, frans.sinaga@mikroskil.ac.id

Dikirim: 24-08-2023 | Diterima: 11-09-2023 | Diterbitkan: 28-10-2023

Abstrak

Bitcoin merupakan salah satu cryptocurrency paling berharga di dunia dan diperdagangkan di lebih dari 40 bursa di seluruh dunia dan menerima lebih dari 30 mata uang berbeda dengan 250.000 transaksi per hari. Dalam perdagangannya, Bitcoin menunjukkan fluktuasi pada pasar yang diperdagangkan, dalam hal ini fluktuasinya dapat mencapai 10 kali lebih tinggi daripada fluktuasi nilai tukar mata uang asing. Karena fluktuasi harga bitcoin yang masif dan tinggi, prediksi fluktuasi harga sangat dibutuhkan, terutama karena harga bitcoin bergerak dengan sangat acak. Untuk melakukan prediksi fluktuasi harga, Random Forest classifier merupakan salah satu algoritma machine learning yang sering digunakan untuk prediksi, kesehatan, artificial intelligence, dll. K-means clustering juga dipergunakan untuk membantu algoritma random forest classifier dalam hal mengkluster data. Hasil dari penelitian ini yaitu melakukan prediksi terhadap naik atau turunnya harga bitcoin dengan akurasi sebanyak 71% yang didapatkan dari perbandingan hasil prediksi dan data asli dengan bantuan confusion matrix.

Kata kunci: Bitcoin, K-Means Clustering, Random Forest Classifier, Confusion Matrix

Abstract

Bitcoin is one of the most valuable cryptocurrencies in the world and is traded on more than 40 exchanges worldwide and accepts more than 30 different currencies with 250,000 transactions per day. In trading, Bitcoin shows fluctuations in the traded market, in which case the fluctuation can be up to 10 times higher than the fluctuation of foreign exchange rates. Due to the massive and high price fluctuations of bitcoin, price fluctuation prediction is needed, especially since the price of bitcoin moves very randomly. To perform price fluctuation prediction, random forest classifier which is one of the machine learning algorithms that are often used for prediction, health, artificial intelligence, etc. K-means clustering is also used to help the random forest classifier algorithm in terms of clustering data. The results of this study predict the rise or fall of bitcoin prices with an accuracy of 71% obtained from comparing the prediction results and the original data with the help of the confusion matrix.

Keywords: Bitcoin, K-Means Clustering, Random Forest Classifier, Confusion Matrix

1. PENDAHULUAN

Cryptocurrency adalah jenis aset digital yang menggunakan teknologi *blockchain* dalam transaksinya [1]. Teknologi *blockchain* dan *cryptocurrency* di bidang keuangan, bisnis, dan perdagangan berkembang sangat cepat dan menawarkan banyak keuntungan [2]. Salah satu *cryptocurrency* paling populer adalah *Bitcoin*, ditemukan oleh Satoshi Nakamoto pada tahun 2008 dalam penelitiannya *Bitcoin: A Peer-to-Peer Electronic Cash System* [3]. *Bitcoin* adalah *cryptocurrency*

paling berharga di dunia dan diperdagangkan di lebih dari 40 bursa di seluruh dunia, menerima lebih dari 30 mata uang berbeda dengan 250.000 transaksi per hari [4].

Pada awal 2009, *Bitcoin* bernilai \$0.08 untuk pertama kalinya. Mata uang berfluktuasi selama empat tahun hingga mencapai \$1,110 pada tahun 2013 [5]. Pada November 2021, harga *bitcoin* naik menjadi \$68,000 per koin dan total harga *bitcoin* melebihi \$1,2 triliun [6]. Telah ditetapkan bahwa pasar *Bitcoin* menunjukkan volatilitas yang berlebihan, dengan volatilitas hingga 10 kali lebih tinggi daripada volatilitas nilai tukar mata uang asing [7]. *Bitcoin* memiliki volatilitas harga *bitcoin* yang masif dan tinggi, prakiraan harga yang akurat sangat dibutuhkan, terutama karena harga *bitcoin* bergerak dengan cara yang tidak dapat diprediksi dan kacau. Teknik pembelajaran mesin dapat membantu menemukan solusi yang lebih baik untuk masalah ini [5]. Faktor-faktor yang mempengaruhi harga *Bitcoin* termasuk biaya transaksi, kesulitan penambangan, sirkulasi mata uang, sistem penghargaan [8] dan ketakutan investor dan pedagang terhadap krisis [9], kebijakan dan pernyataan politik, dan liputan peristiwa buruk yang muncul terhadap *Bitcoin* [10]. Pengambilan keputusan bisnis yang tepat membutuhkan strategi yang akurat dari para ekonom dan investor, namun beberapa investor mengalami kerugian ketika menerapkan strategi dari para ahli tersebut [11].

Pada penelitian yang dilakukan untuk memprediksi harga bitcoin dengan menggunakan model ARIMA, akurasi hasil prediksi harga penutupan bitcoin harian pada harga yang sangat fluktuatif adalah sebesar 49% [12]. Dalam sebuah penelitian oleh Edwin Sin dan Lipo Wang, prediksi harga *bitcoin* dilakukan dengan menggunakan model *Ensemble Neural Networks* untuk mengetahui perubahan harga *bitcoin* di hari berikutnya, prediksi harga dibuat untuk 50 hari ke depan, dan selanjutnya dilakukan pengujian terhadap perubahan harga *bitcoin*. Akurasi tes yang diperoleh dari *back-testing* adalah 64% dengan prediksi yang benar 32 hari dalam 50 hari [13].

Untuk mengatasi masalah penelitian ini, disediakan algoritma *random forest Classifier*, di mana langkah pertama adalah memberi label pada kumpulan data yang digunakan dan kemudian memunculkan indikator seperti *simple-moving average (SMA)*, *exponential moving average (EMA)*, *relative-strength index (RSI)*, *stochastic oscillator (STI)*, dan *price rate of change (PROC)* dalam analisis teknis [14], selanjutnya *preprocessing* menggunakan algoritma normalisasi *min-max* untuk membuat rentang baru dari data yang sudah ada yang dapat menyesuaikan data dalam batas yang telah ditentukan, dengan harapan hasil dari normalisasi nilainya lebih kecil dan lebih dekat untuk membuat variasi perkiraan lebih akurat [15], Kemudian dilakukan clustering menggunakan algoritma *K-means clustering*, ini adalah metode untuk menemukan struktur *cluster* dalam kumpulan data yang dicirikan oleh kesamaan terbesar dalam cluster yang sama dan perbedaan terbesar antara *cluster* yang berbeda yang bertujuan agar hasil yang didapatkan bersifat konvergen [16], Setelah menerima data dari hasil *clustering*, dapat dilakukan *training* data dan data testing, dengan mengambil beberapa variabel dan melakukan perhitungan berdasarkan variabel tersebut [17]. Dari hasil tersebut, dapat dilakukan prediksi dengan menggunakan algoritma *random forest*. Setelah itu, *Confusion* matriks digunakan dalam fase pengujian akurasi dan hasil prediksi [18].

Penelitian ini dilakukan untuk dapat memprediksi flutuasi harga bitcoin pada rentang tertentu berdasarkan data yang telah ada, dengan menggunakan metode *Random Forest Classifier*.

2. TINJAUAN PUSTAKA

2.1 Bitcoin

Bitcoin adalah salah satu koin paling terkenal, *Bitcoin* dianggap sebagai *cryptocurrency* asli. *Bitcoin* dibuat pada tahun 2009 sebagai perangkat lunak sumber terbuka. Penulis buku putih yang mendirikan mata uang digital ini menggunakan nama samaran Satoshi Nakamoto. *Bitcoin* adalah protokol komunikasi Internet yang memfasilitasi penggunaan mata uang *virtual*, termasuk pembayaran elektronik. Aturan *Bitcoin* dirancang oleh para insinyur tanpa masukan nyata dari pengacara atau regulator. Alih-alih menyimpan transaksi pada satu *server* atau sekelompok *server* [20].

2.2 Indikator Teknikal

2.2.1 SMA (*Simple Moving Average*)

Simple Moving Average adalah metode yang populer dalam *time series technical analysis*.

Defenisi SMA adalah nilai rata-rata selama waktu yang ditentukan, misalnya; SMA5 berarti nilai rata-rata asset dengan periode 5 hari.

$$SMA = \frac{A_1 + A_2 + \dots + A_n}{n} \quad (1)$$

Di mana :

n = total periode

A_n = nilai pada periode n [21]

2.2.2 EMA (*Exponential Moving Average*)

EMA adalah metode dalam *Moving Average* yang menggunakan rumus perhitungan yang memberikan bobot pada harga saat ini relatif terhadap harga awal dari perhitungan EMA pada rentang waktu tertentu. Semakin pendek rentang waktu yang digunakan, maka semakin baik pula penerapan EMA untuk memberikan nilai rata-rata terbaru.

$$EMA = K * (C - P) + P \quad (2)$$

$$K = \frac{2}{N + 1} \quad (3)$$

Di mana :

C = harga close hari ini

K = smoothing EMA dimana $2/(N+1)$

N = Jumlah periode dalam perhitungan EMA

P = Nilai EMA sebelumnya [21]

2.2.3 RSI (*Relative Strength Index*)

RSI adalah metode yang digunakan untuk menghitung daya tarik kenaikan dan penurunan harga. Nilainya berkisar dari 0 – 100. Dengan RSI, kita dapat mengetahui apakah suatu harga overbought atau oversold. Jika RSI bernilai sangat tinggi (diatas 70) berarti overbought sehingga ada potensi terjadi penurunan harga. Sebaliknya, jika RSI bernilai sangat rendah (dibawah 30) berarti oversold sehingga ada potensi terjadi kenaikan harga.

$$RSI = \left(\frac{X}{Y} \right) * 100 \quad (4)$$

Di mana :

X = harga close sekarang

Y = Harga close terakhir dalam periode yang telah ditentukan [21]

2.2.4 STI (*Stochastic Oscillator*)

Stochastic Oscillator adalah indikator yang berfokus pada momentum harga yang berguna untuk mengidentifikasi tingkat overbought dan oversold dari suatu saham dan memprediksi *trend reversal*. Para trader dan investor menggunakan indikator ini Bersama dengan *Moving Average* untuk membantu meningkatkan akurasi *trading*.

$$STI = \frac{(\text{Today's close} - \text{Lowest low in } k \text{ period})}{(\text{Highest high in } k \text{ period} - \text{Lowest low in } k \text{ period})} * 100 \quad (5)$$

Di mana :

Today close = Nilai *close* terkini.

Lowest Low = Nilai *Low* terendah pada periode pengamatan.

Highest high = Nilai *high* tertinggi pada periode pengamatan [21]

2.2.5 PROC (*Price Rate of Change*)

Price Rate of Change adalah indikator yang menunjukkan perbedaan antara harga saat ini dengan harga yang lampau dalam periode yang ditentukan. Hasil dari PROC ditampilkan dalam bentuk persentase.

$$PROC = \frac{\text{Today's close} - \text{close } x \text{ periods ago}}{\text{close } x \text{ periods ago}} \quad (6)$$

Di mana:

Today's close = Nilai *Close* terkini.

Close x periods ago = Nilai *Close* pada periode sebelumnya [21]

2.3 Min-Max Normalization

Normalisasi adalah teknik penskalaan atau teknik pemetaan atau langkah pra-pemrosesan. Dimana kita menemukan pilihan baru dari pilihan yang ada, ini bisa sangat membantu untuk tujuan prediksi atau peramalan. Seperti yang kita ketahui, banyak sekali cara untuk membuat ramalan atau prediksi, namun semuanya bisa sangat bervariasi. Jadi, untuk mempertahankan variabilitas prakiraan dan proyeksi yang tinggi, diperlukan teknik normalisasi untuk mengaproksimasikannya. Normalisasi min-max adalah teknik sederhana dimana teknik tersebut secara khusus dapat menyesuaikan data ke batas yang diberikan dengan batas yang diberikan .

$$A' = \left(\frac{A - \text{min value of } A}{\text{max value of } A - \text{min value of } A} \right) * (D - C) + C \quad (7)$$

Di mana:

A = Nilai data.

C = Batas bawah.

D = Batas atas [15]

2.4 Silhouette Method

Metode *Silhouette* pertama kali diusulkan oleh *Peter J. Rousseeuw*. Metode ini menggabungkan dua faktor yaitu kohesi dan resolusi. Kohesi adalah kesamaan antara objek dan cluster. Nilai *Silhouette*, berada dalam kisaran -1 sampai 1 . Nilai *Silhouette* mendekati 1 , menunjukkan bahwa terdapat hubungan yang erat antara objek dengan *cluster*. Jika cluster data dalam model dihasilkan dengan nilai *Silhouette* yang relatif tinggi, maka model tersebut cocok dan dapat diterima.

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (8)$$

Di mana :

$a(i)$ = Nilai rata-rata jarak antara data dengan data pada cluster yang sama.

$b(i)$ = Nilai rata-rata jarak antara titik data dengan data pada cluster tetangga.

$\max\{a(i), b(i)\}$ = Nilai paling besar antara $a(i)$ dan $b(i)$ [22].

2.5 K-Means Clustering

Algoritma *K-Means Clustering* merupakan algoritma pengelompokan iteratif yang melakukan partisi set data ke dalam sejumlah K cluster yang sudah ditetapkan di awal. Algoritma *K-Means* sederhana untuk diimplementasikan dan dijalankan, relatif cepat, mudah beradaptasi, dan umum penggunaannya dalam praktek. Secara historis, *K-Means* menjadi salah satu algoritma yang paling penting dalam bidang *data mining* [23].

Algoritma *k-Means Clustering* terdiri dari tahap beberapa tahapan, yaitu:

1. Initialization (Inisiasi):

Tahapan dimana terjadinya beberapa proses sebagai berikut:

- a. Tentukan jumlah kluster (K) yang diinginkan.
- b. Pilih secara acak K titik sebagai pusat awal kluster (centroid).

2. Assignment (Pengelompokan):

Tahapan dimana terjadinya beberapa proses sebagai berikut:

- a. Hitung jarak *Euclidean* antara setiap data dan centroid.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (9)$$

- b. Selanjutnya, atribusikan setiap titik data dengan cluster berdasarkan jarak terdekat

$$d_i = \min(X_i, \mu_j) \quad (10)$$

3. Perbarui centroid

Tahapan dimana menghitung kembali posisi *centroid* baru untuk setiap kluster dengan mengambil rata-rata dari semua data yang termasuk dalam kluster tersebut.

$$\mu_j = \frac{1}{N_j} \sum_{i=1}^{N_j} X_i \quad (11)$$

Dimana:

N_j : Jumlah titik dalam *cluster*

Ulangi hingga cluster center berhenti berubah.

4. Ulangi langkah 2 dan 3 hingga kondisi berhenti terpenuhi

Kondisi berhenti bisa berupa jumlah iterasi yang telah ditentukan, atau jika tidak ada perubahan dalam penempatan data antara iterasi.

5. Hasil akhir

a. Setelah konvergensi (*centroid* tidak berubah) atau berhenti, kluster terbentuk.

b. Setiap data termasuk dalam satu kluster tertentu berdasarkan *centroid* terdekat [23].

2.6. Random Forest Classifier

Random Forest Classifier terdiri dari beberapa *decision tree* yang memiliki sifat *non-metric classifier* karena itu, tidak perlu diberikan *learning parameter* dan untuk pengambilan model prediksi, Random Forest membuat sebuah prediksi untuk setiap kelas berdasarkan *individual treenya* dan kelas dengan vote terbanyak akan dipilih sebagai model dasar prediksi. Setiap terjadinya split pada *tree* maka akan ada *random sample* dari predictor yang akan dipilih sebagai *split candidate*.

Berikut algoritma dari *random forest* klasifikasi.

1. Untuk $b = 1$ sampai B :

a. Buat sampel bootstrap Z^* dengan ukuran N dari *data training*.

b. Buat/Tumbuhkan pohon *random forest* T_b dengan data bootstrap, dengan mengulangi langkah-langkah berikut secara rekursif sampai ukuran node minimum n_{\min} terpenuhi:

i. pilih m variabel secara acak dari p , dimana p adalah jumlah variabel pada dataset

ii. pilih variable split / *best split* diantara m variabel tersebut.

iii. Pisahkan *node* menjadi dua *node* anak.

2. Hasil dari *ensemble tree* adalah $\{T_b\}_1^B$.

3. Untuk membuat prediksi terhadap data baru x :

$$C_{rf}^B(x) = \text{majority vote } \{C_b(x)\}_1^B. \quad (12)$$

Dimana: $C_b(x)$ adalah prediksi kelas dari pohon random forest [24].

2.7 Confusion Matrix

Confusion matrix merupakan salah satu metode yang digunakan untuk mengevaluasi performa model dalam masalah klasifikasi. Jumlah contoh yang diklasifikasikan dengan benar adalah jumlah diagonal dalam matriks, di luar dari diagonal contoh diklasifikasikan dengan salah. *Confusion Matrix* dapat digunakan untuk menghitung *accuracy*, *precision*, *recall*, *f1-score* [21].

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad (13)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (14)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (15)$$

$$F1 - \text{Score} = \frac{2TP}{2TP+FP+FN} \quad (16)$$

Dimana:

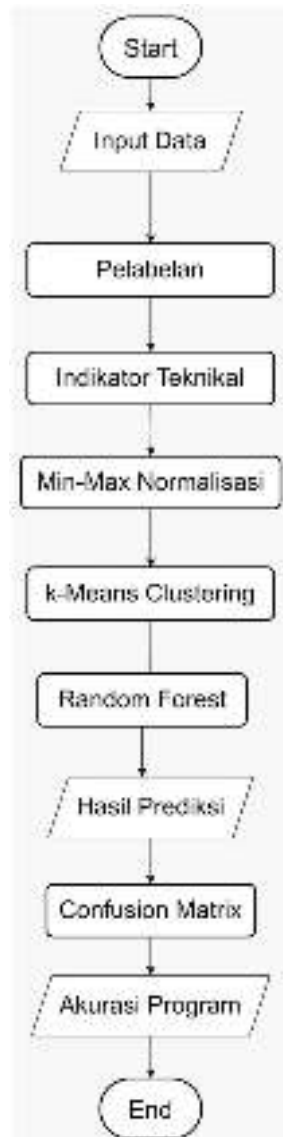
TP = *True positive*

FP = *False positive*

TN = *True negative*

FN = *False negative*

3. METODE PENELITIAN



Gambar 1. Gambaran Flowchart Penelitian

3.1 Input Data

Pada tahap ini akan diambil dataset *bitcoin* harga buka (*Open*), harga tertinggi (*high*), harga terendah (*low*), harga tutup (*close*), *volume*, dan tanggal.

3.2 Pelabelan Data

Pada tahapan pelabelan data maka akan dilakukan pelabelan data berdasarkan parameter dari harga “*Close*” pada *Dataset* yang ada. Label “1” menyatakan harga *Bitcoin* pada hari tersebut “Naik” dan Label “0” menyatakan harga *Bitcoin* pada hari tersebut “Turun. Penentuan label dilakukan dengan membandingkan nilai *close* saat ini dengan nilai *close* sebelumnya. Jika nilai *close* saat ini lebih besar daripada nilai *close* sebelumnya maka label adalah 1 yang berarti naik. tetapi jika nilai *close* saat ini lebih kecil daripada nilai *close* sebelumnya maka label adalah 0 yang berarti turun.

3.3 Indikator Teknikal

Pada proses ini, akan dilakukan perhitungan teknikal indikator dengan menggunakan dataset yang telah diinput. Indikator teknikal yang akan dihitung yaitu *simple-moving average* (SMA) 10, 20,

30, *exponential moving average* (EMA) 10, 20, 30, *relative-strength index* (RSI) 14, 19, *stochastic oscillator* 14, dan *price rate of change* (PROC).

3.4 Normalisasi Data

Pada proses ini, dataset bitcoin beserta hasil teknikal indikator akan dinormalisasi menggunakan metode *Min-Max*. Normalisasi bertujuan untuk mengecilkan rentang nilai dari dataset. Rentang nilai yang digunakan adalah 0 sampai dengan 1.

3.5 Mengcluster Data

Pada proses ini, data yang telah dinormalisasi akan dicluster menggunakan algoritma *K-Means*. Sebelum data di cluster menggunakan algoritma *K-Means*, akan dicari terlebih dahulu nilai K yang paling optimal menggunakan *Silhouette Method*. Setelah nilai K yang optimal didapatkan, data akan dicluster dengan nilai K yang paling optimal menggunakan algoritma *K-Means*.

3.6 Proses Training Model

Pada proses ini, model akan dilatih menggunakan algoritma *Random Forest Classifier*.

Langkah-langkah dalam proses pelatihan model *Random Forest Classifier* adalah sebagai berikut:

1. Persiapan data *training*: Mempersiapkan data training yang terdiri dari fitur-fitur dan label atau kelas yang sesuai untuk setiap sampel. Fitur-fitur menggambarkan atribut dari setiap sampel, sedangkan label atau kelas adalah nilai yang ingin Anda prediksi.
2. Pembentukan pohon-pohon keputusan: Model *random forest* terdiri dari banyak pohon keputusan yang independen. Setiap pohon keputusan dibangun dengan mengambil subset acak dari data *training*. Pohon-pohon ini dapat dibangun menggunakan algoritma *decision tree*.
3. Proses pembuatan pohon: Pada setiap langkah dalam pembangunan pohon, pemilihan fitur terbaik dilakukan berdasarkan kriteria yang sesuai (misalnya, Gini impurity atau Entropy untuk klasifikasi). Pohon terus dibagi menjadi cabang-cabang yang lebih kecil berdasarkan aturan ini hingga mencapai kondisi berhenti (misalnya, kedalaman maksimum atau ukuran minimum node).
4. Pemilihan mayoritas suara: Pada setiap node daun (leaf) dalam pohon keputusan, kelas mayoritas dalam subset data training yang mencapai node tersebut dipilih sebagai prediksi.
5. Agregasi pohon-pohon: Setelah semua pohon keputusan selesai dibangun, prediksi dari setiap pohon digunakan untuk membuat "voting" atau mayoritas suara. Kelas yang mendapatkan mayoritas suara menjadi prediksi akhir.

3.7 Proses Testing

Langkah-langkah dalam proses pengujian model *Random Forest Classifier* adalah sebagai berikut:

1. Persiapan data *testing*: Mempersiapkan data *testing* yang hanya terdiri dari fitur-fitur. Data *testing* ini belum pernah dilihat oleh model sebelumnya dan digunakan untuk menguji performa model pada data yang belum pernah dilihat.
2. Prediksi dengan setiap pohon: Setiap pohon keputusan dalam *ensemble random forest* digunakan untuk memprediksi kelas pada setiap sampel data *testing*. Setiap pohon memberikan prediksi sendiri-sendiri berdasarkan fitur-fitur dari sampel tersebut.
3. Agregasi hasil prediksi pohon-pohon: Hasil prediksi dari setiap pohon dalam *ensemble* digabungkan menggunakan mayoritas suara. Kelas yang mendapatkan mayoritas suara dianggap sebagai prediksi akhir dari model.

3.8 Evaluasi performa model

Pada proses ini, hasil prediksi model terhadap *data testing* akan dibandingkan dengan label asli dari data, sehingga akan menghasilkan nilai TP (*True Positive*), TN (*True Negative*), FP (*False Positive*), FN (*False Negative*). Nilai-nilai tersebut nantinya akan digunakan untuk menghitung *accuracy*, *precision*, *recall* dan *f1-score*.

4. HASIL DAN PEMBAHASAN

4.1. Tampilan aplikasi

Berikut adalah tampilan dari aplikasi yang telah dibuat

a. Halaman input data

Halaman input data merupakan halaman yang digunakan untuk memasukan data, data yang telah diunggah kemudian akan ditampilkan isinya ke dalam sebuah tabel.

	Date	Open	High	Low
1	2014-09-17	460.964014	468.174011	452.421997
2	2014-09-18	456.859985	456.859985	413.104004
3	2014-09-19	474.102997	477.834991	384.532013
4	2014-09-20	504.673004	473.705990	380.887996
5	2014-09-21	408.084991	412.425995	393.101000
6	2014-09-22	399.100006	406.915985	397.130005

Gambar 2. Tampilan halaman input

b. Halaman teknis indikator

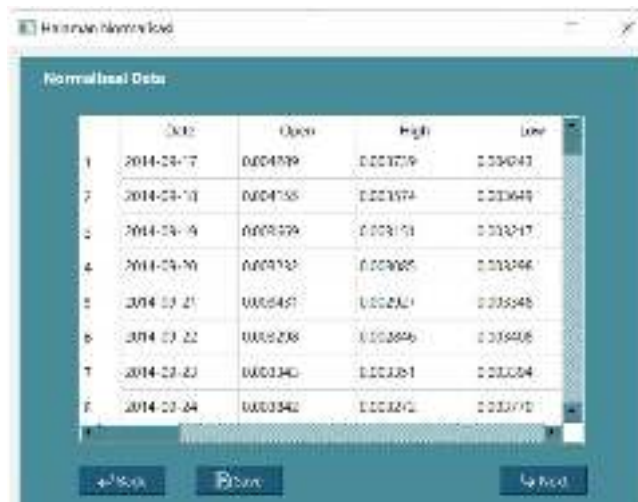
Halaman ketiga merupakan halaman yang digunakan untuk menjalankan perhitungan teknis indikator dan kemudian memunculkan hasil perhitungan pada tabel yang tersedia pada halaman tersebut.

	Date	Open	High	Low	Close
1	2014-09-17	460.964014	468.174011	452.421997	457.314015
2	2014-09-18	456.859985	456.859985	413.104004	424.442002
3	2014-09-19	474.102997	477.834991	384.532013	384.757990
4	2014-09-20	504.673004	473.705990	380.887996	408.803997
5	2014-09-21	408.084991	412.425995	393.101000	388.821014
6	2014-09-22	399.100006	406.915985	397.130005	400.157003
7	2014-09-23	402.092010	411.572007	390.183991	425.703985
8	2014-09-24	415.751007	415.112000	421.111008	423.254997

Gambar 3. Tampilan halaman teknis indikator

c. Halaman normalisasi

Halaman normalisasi merupakan halaman yang digunakan untuk menjalankan perhitungan normalisasi *min-max* dan kemudian memunculkan hasil perhitungan pada tabel yang tersedia pada halaman tersebut.

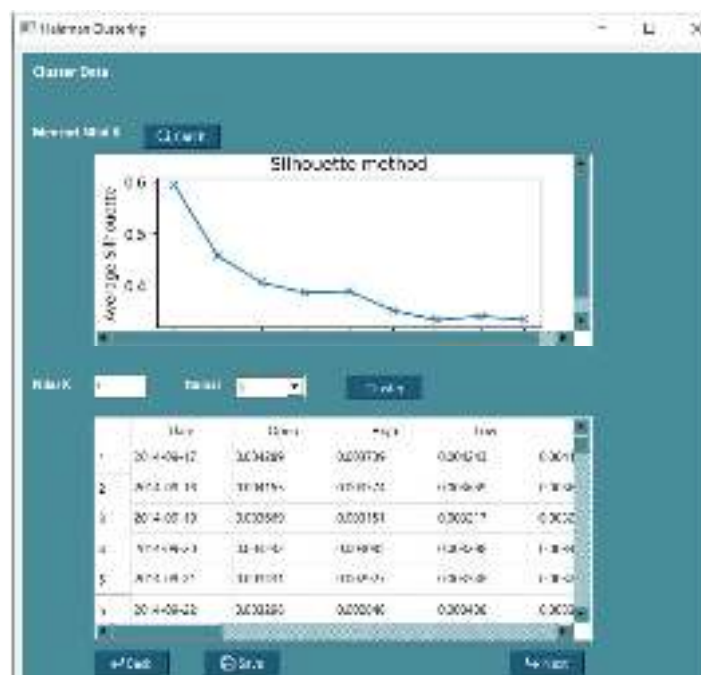


	Date	Open	High	Low
1	2014-09-17	0.004879	0.007079	0.004043
7	2014-09-18	0.004155	0.005574	0.003640
2	2014-09-19	0.005329	0.008151	0.005247
4	2014-09-20	0.004152	0.005025	0.003346
5	2014-09-21	0.005451	0.008201	0.005348
6	2014-09-22	0.005208	0.008296	0.004906
7	2014-09-23	0.003340	0.004351	0.003054
8	2014-09-24	0.003342	0.004352	0.003054

Gambar 4. Tampilan halaman normalisasi

d. Halaman *clustering*

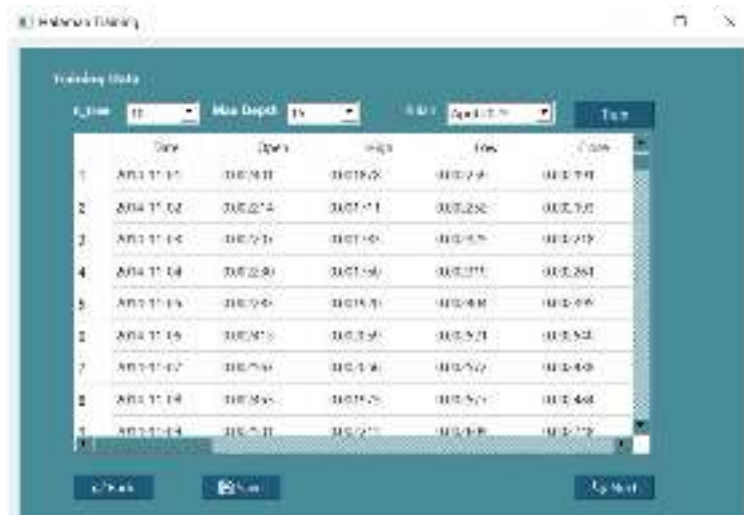
Halaman *clustering* merupakan halaman yang digunakan untuk mencari nilai k, kemudian memunculkan grafik dan hasil pencarian dari nilai k, setelah itu melakukan perhitungan kluster sesuai dengan iterasi yang ditentukan dan kemudian memunculkan hasil perhitungan pada tabel yang tersedia pada halaman tersebut.



Gambar 5. Tampilan halaman clustering

e. Halaman training model

Halaman training model merupakan halaman yang digunakan untuk melakukan *training* data dengan jumlah pohon dan kedalaman yang di tentukan oleh pengguna kemudian memunculkan *training* data pada tabel yang tersedia pada halaman tersebut.

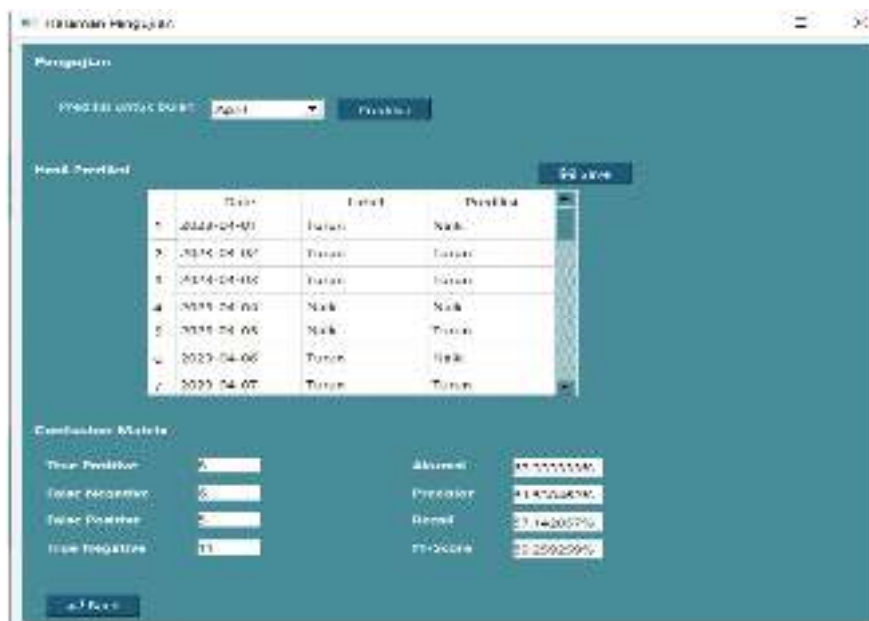


	Date	Day	Depth	Iteration	Prediction
1	2014-09-17	09/17/14	05	00150	00150
2	2014-09-18	09/18/14	05	00151	00151
3	2014-09-19	09/19/14	05	00152	00152
4	2014-09-20	09/20/14	05	00153	00153
5	2014-09-21	09/21/14	05	00154	00154
6	2014-09-22	09/22/14	05	00155	00155
7	2014-09-23	09/23/14	05	00156	00156
8	2014-09-24	09/24/14	05	00157	00157
9	2014-09-25	09/25/14	05	00158	00158

Gambar 6. Tampilan halaman training

f. Halaman pengujian model

Halaman pengujian model merupakan halaman yang menampilkan prediksi dan performa dari model random forest terhadap data testing.



	Date	Day	Depth	Iteration	Prediksi
1	2023-04-03	04/03/23	05	00150	00150
2	2023-04-04	04/04/23	05	00151	00151
3	2023-04-05	04/05/23	05	00152	00152
4	2023-04-06	04/06/23	05	00153	00153
5	2023-04-07	04/07/23	05	00154	00154
6	2023-04-08	04/08/23	05	00155	00155
7	2023-04-09	04/09/23	05	00156	00156
8	2023-04-10	04/10/23	05	00157	00157
9	2023-04-11	04/11/23	05	00158	00158

Confusion Matrix	
True Positive	5
False Positive	5
True Negative	13
Accuracy	87,500000%
Precision	50,000000%
Recall	27,777778%
F1-Score	33,333333%

Gambar 7. Tampilan halaman pengujian model

4.2. Pengujian sistem

Pada tahap ini akan dilakukan pengujian pada aplikasi yang telah kami buat untuk mendukung penelitian ini, kemudian dengan menggunakan dataset bitcoin yang berasal dari website yahoofinance dari tanggal 17 september 2014 sampai 31 april 2023. Metode confusion matrix digunakan pada pengujian untuk mencari nilai akurasi, precision, recall, dan f1-score. Parameter yang uji berupa iterasi, tree, depth, dan bulan yang akan diprediksi. Nilai iterasi yang diuji adalah 5, 15, dan 25. Nilai tree yang diuji adalah 5, 10, 15, 20, dan 25. Nilai depth yang diuji adalah 5, 15, 25, 35, dan 45. Bulan yang diprediksi ialah bulan mei 2023. Berikut dibawah ini hasil pengujian.

Tabel 1. Hasil pengujian Pada iterasi 5

iterasi	tree	depth	prediksi	Akurasi	precision	Recall	f1-score
5	5	5	mei	67,741935%	59,090909%	92,857143%	72,222222%

5	5	15	mei	61,290323%	57,142857%	57,142857%	57,142857%
5	5	25	mei	54,838710%	50,000000%	100,000000%	66,666667%
5	5	35	mei	64,516129%	57,894737%	78,571429%	66,666667%
5	5	45	mei	67,741935%	59,090909%	92,857143%	72,222222%
5	10	5	mei	58,064516%	52,380952%	78,571429%	62,857143%
5	10	15	mei	42,161290%	40,000000%	42,857143%	41,379310%
5	10	25	mei	48,387097%	45,000000%	64,285714%	52,941176%
5	10	35	mei	51,612903%	46,666667%	50,000000%	48,275862%
5	10	45	mei	67,741935%	64,285714%	64,285714%	64,285714%
5	15	5	mei	70,967742%	60,869565%	100,000000%	75,675676%
5	15	15	mei	67,741935%	59,090909%	92,857143%	72,222222%
5	15	25	mei	61,290323%	55,000000%	78,571429%	64,705882%
5	15	35	mei	70,967742%	60,869565%	100,000000%	75,675676%
5	15	45	mei	58,064516%	52,380952%	78,571429%	62,857143%
5	20	5	mei	61,290323%	54,166667%	92,857143%	68,421053%
5	20	15	mei	61,290323%	56,250000%	64,285714%	60,000000%
5	20	25	mei	61,290323%	55,555556%	71,428571%	62,500000%
5	20	35	mei	67,741935%	61,111111%	78,571429%	68,750000%
5	20	45	mei	67,741935%	59,090909%	92,857143%	72,222222%
5	25	45	mei	67,741935%	59,090909%	92,857143%	72,222222%

Pada Tabel 1 menunjukkan perbandingan hasil akurasi, precision, recall dan f1-score pada prediksi bulan mei 2023 dengan menggunakan parameter iterasi untuk *k-means* sama dengan 5, jumlah tree untuk algoritma random forest sama dengan 5-25, jumlah depth untuk algoritma random forest sama dengan 5-45.

Tabel 2. Hasil pengujian Pada iterasi 15

iterasi	Tree	depth	prediksi	akurasi	precision	recall	f1-score
15	5	5	mei	67,741935%	59,090909%	92,857143%	72,222222%
15	5	15	mei	61,290323%	57,142857%	57,142857%	57,142857%
15	5	25	mei	54,838710%	50,000000%	100,000000%	66,666667%
15	5	35	mei	64,516129%	57,894737%	78,571429%	66,666667%
15	5	45	mei	67,741935%	59,090909%	92,857143%	72,222222%
15	10	5	mei	58,064516%	52,380952%	78,571429%	62,857143%
15	10	15	mei	42,161290%	40,000000%	42,857143%	41,379310%
15	10	25	mei	48,387097%	45,000000%	64,285714%	52,941176%
15	10	35	mei	51,612903%	46,666667%	50,000000%	48,275862%
15	10	45	mei	67,741935%	64,285714%	64,285714%	64,285714%
15	15	5	mei	70,967742%	60,869565%	100,000000%	75,675676%
15	15	15	mei	67,741935%	59,090909%	92,857143%	72,222222%
15	15	25	mei	61,290323%	55,000000%	78,571429%	64,705882%
15	15	35	mei	70,967742%	60,869565%	100,000000%	75,675676%
15	15	45	mei	58,064516%	52,380952%	78,571429%	62,857143%
15	20	5	mei	61,290323%	54,166667%	92,857143%	68,421053%

15	20	15	mei	61,290323%	56,250000%	64,285714%	60,000000%
15	20	25	mei	61,290323%	55,555556%	71,428571%	62,500000%
15	20	35	mei	67,741935%	61,111111%	78,571429%	68,750000%
15	20	45	mei	67,741935%	59,090909%	92,857143%	72,222222%
15	25	45	mei	67,741935%	59,090909%	92,857143%	72,222222%

Pada tabel 2 menunjukkan perbandingan hasil akurasi, precision, recall dan f1-score pada prediksi bulan mei 2023 dengan menggunakan parameter iterasi untuk *k-means* sama dengan 15, jumlah tree untuk algoritma random forest sama dengan 5-25, jumlah depth untuk algoritma random forest sama dengan 5-45.

Tabel 3. Hasil pengujian Pada iterasi 25

iterasi	tree	depth	prediksi	akurasi	precision	recall	f1-score
25	5	5	mei	67,741935%	59,090909%	92,857143%	72,222222%
25	5	15	mei	61,290323%	57,142857%	57,142857%	57,142857%
25	5	25	mei	54,838710%	50,000000%	100,000000%	66,666667%
25	5	35	mei	64,516129%	57,894737%	78,571429%	66,666667%
25	5	45	mei	67,741935%	59,090909%	92,857143%	72,222222%
25	10	5	mei	58,064516%	52,380952%	78,571429%	62,857143%
25	10	15	mei	42,161290%	40,000000%	42,857143%	41,379310%
25	10	25	mei	48,387097%	45,000000%	64,285714%	52,941176%
25	10	35	mei	51,612903%	46,666667%	50,000000%	48,275862%
25	10	45	mei	67,741935%	64,285714%	64,285714%	64,285714%
25	15	5	mei	70,967742%	60,869565%	100,000000%	75,675676%
25	15	15	mei	67,741935%	59,090909%	92,857143%	72,222222%
25	15	25	mei	61,290323%	55,000000%	78,571429%	64,705882%
25	15	35	mei	70,967742%	60,869565%	100,000000%	75,675676%
25	15	45	mei	58,064516%	52,380952%	78,571429%	62,857143%
25	20	5	mei	61,290323%	54,166667%	92,857143%	68,421053%
25	20	15	mei	61,290323%	56,250000%	64,285714%	60,000000%
25	20	25	mei	61,290323%	55,555556%	71,428571%	62,500000%
25	20	35	mei	67,741935%	61,111111%	78,571429%	68,750000%
25	20	45	mei	67,741935%	59,090909%	92,857143%	72,222222%
25	25	45	mei	67,741935%	59,090909%	92,857143%	72,222222%

Pada Tabel 3 menunjukkan perbandingan hasil akurasi, precision, recall dan f1-score pada prediksi bulan mei 2023 dengan menggunakan parameter iterasi untuk *k-means* sama dengan 25, jumlah tree untuk algoritma random forest sama dengan 5-25, jumlah depth untuk algoritma random forest sama dengan 5-45.

Tabel 4. Hasil Pengujian Teknikal Indikator

indikator	akurasi	precision	recall	f1-score
sma 10	45,161290%	45,161290%	100,000000%	62,222222%
sma 20	-	-	-	-
sma 30	45,161290%	45,161290%	100,000000%	62,222222%
ema 10	45,161290%	45,161290%	100,000000%	62,222222%
ema 20	45,161290%	45,161290%	100,000000%	62,222222%

ema 30	45,161290%	45,161290%	100,000000%	62,222222%
rsi 14	61,290323%	100,000000%	14,285714%	25,000000%
rsi 19	51,612903%	47,368421%	64,285714%	54,545455%
sti 14	61,290323%	100,000000%	14,285714%	25,000000%
proc	51,612903%	46,153846%	42,857143%	44,444444%
Sma 10, 20, 30, ema 10, 20, 30, rsi 14, 19, sti 14, proc	67,741935%	59,090909%	92,857143%	72,222222%

Pada Tabel 4 menunjukkan perbandingan hasil akurasi, precision, recall dan f1-score pada prediksi bulan mei 2023 dengan menggunakan teknikal indikator SMA10, SMA20, SMA30, EMA10, EMA20, EMA30, RSI14, RSI19, STI14, PROC. iterasi untuk *k-means* sama dengan 5, jumlah tree yang digunakan untuk algoritma random forest sama dengan 5, jumlah depth yang digunakan untuk algoritma random forest sama dengan 5.

Pada pengujian parameter iterasi yang menggunakan nilai 5, 15, dan 25. karena dengan nilai tersebut perubahan nilai hasil pengujian bisa lebih terlihat dibandingkan menggunakan nilai 1,2,3,4,5. Iterasi pada algoritma *k-means* menentukan kapan perulangan pada *k-means* akan berhenti. Dari hasil penelitian kami pengujian iterasi setelah iterasi 25, kluster telah konvergen atau tidak berubah sama sekali. Karena algoritma *k-means* sendiri dimana jika kluster telah konvergen, dilanjutkan ke iterasi berikutnya hasil kluster akan tetap sama. Sama halnya dengan pengujian parameter *tree* dan *depth* pada *random forest*, pada pengujian juga menggunakan parameter kelipatan 5 sehingga perubahan nilai hasil pengujian bisa terlihat dibandingkan dengan menggunakan nilai 1,2,3,4,5. Dari hasil pengujian kami kombinasi tree dan depth diluar 45 dan 25 tidak mengalami perubahan hasil pengujian yang signifikan.

5. KESIMPULAN

Setelah membuat aplikasi prediksi harga bitcoin dengan Random Forest, maka kesimpulan yang dapat ditarik oleh penulis adalah sebagai berikut.

1. Dapat disimpulkan performa tertinggi diperoleh dengan menggunakan semua teknikal analisis, kemudian pada *k-means* clustering iterasi yang menghasilkan performa tertinggi terjadi pada iterasi 5, 15, dan 25, dan pada random forest classifier performa tertinggi terjadi pada saat jumlah pohon berjumlah 15 dan kedalaman pohon 35 dengan akurasi prediksi 71%, presisi 61%, recall 100%, dan f1-score 76% untuk prediksi harga bitcoin pada bulan mei 2023.
2. Dengan adanya jurnal ini, maka dapat melengkapi literatur untuk mengatasi permasalahan tingginya volatilitas harga bitcoin yang menyebabkan susah diprediksi.

6. SARAN

Beberapa saran yang dapat penulis berikan untuk mengembangkan sistem dalam memprediksi harga bitcoin ini lebih lanjut adalah sebagai berikut.

1. Dapat melakukan prediksi terhadap koin lain yang ada pada Cryptocurrency.
2. Dapat mengembangkan algoritma dengan menggabungkan beberapa algoritma lainnya ke dalam random forest untuk menjadikan metode ensemble heterogen.
3. Dapat meningkatkan akurasi dalam melakukan prediksi.

UCAPAN TERIMA KASIH

Terima kasih kami ucapkan kepada dosen pembimbing kami dan Universitas Mikroskil atas dukungan dan sarannya sehingga kami dapat membuat jurnal ini.

DAFTAR PUSTAKA

- [1] P. Zimmerman, "Blockchain structure and cryptocurrency prices," 2020.
- [2] D. P. Sakas, N. T. Giannakopoulos, N. Kanellos, and S. P. Migkos, "Innovative cryptocurrency trade websites' marketing strategy refinement, via digital behavior," *IEEE Access*, vol. 10, pp. 63163–63176, 2022.
- [3] U. W. Chohan, "A history of bitcoin," *Available at SSRN 3047875*, 2022.
- [4] S. McNally, J. Roche, and S. Caton, "Predicting the price of bitcoin using machine learning," in *2018 26th euromicro international conference on parallel, distributed and network-based processing (PDP)*, IEEE, 2018, pp. 339–343.
- [5] S. Erfanian, Y. Zhou, A. Razzaq, A. Abbas, A. A. Safeer, and T. Li, "Predicting Bitcoin (BTC) Price in the Context of Economic Theories: A Machine Learning Approach," *Entropy*, vol. 24, no. 10, p. 1487, 2022.
- [6] J. Chen, "Analysis of Bitcoin Price Prediction Using Machine Learning," *Journal of Risk and Financial Management*, vol. 16, no. 1, p. 51, 2023.
- [7] D. G. Baur and T. Dimpfl, "The volatility of Bitcoin and its role as a medium of exchange and a store of value," *Empir Econ*, vol. 61, no. 5, pp. 2663–2683, 2021.
- [8] Y. Sovbetov, "Factors influencing cryptocurrency prices: Evidence from bitcoin, ethereum, dash, bitcoin, and monero," *Journal of Economics and Financial Analysis*, vol. 2, no. 2, pp. 1–27, 2018.
- [9] D. Anastasiou, A. Ballis, and K. Drakos, "Cryptocurrencies' price crash risk and crisis sentiment," *Financ Res Lett*, vol. 42, p. 101928, 2021.
- [10] F. Kjørland, M. Meland, A. Oust, and V. Øyen, "How can bitcoin price fluctuations be explained?," 2018.
- [11] O. Sattarov *et al.*, "Recommending cryptocurrency trading points with deep reinforcement learning approach," *Applied Sciences*, vol. 10, no. 4, p. 1506, 2020.
- [12] M. Poongodi, V. Vijayakumar, and N. Chilamkurti, "Bitcoin price prediction using ARIMA model," *International Journal of Internet Technology and Secured Transactions*, vol. 10, no. 4, pp. 396–406, 2020.
- [13] E. Sin and L. Wang, "Bitcoin price prediction using ensembles of neural networks," in *2017 13th International conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD)*, IEEE, 2017, pp. 666–671.
- [14] I. K. Nti, A. F. Adekoya, and B. A. Weyori, "A systematic review of fundamental and technical analysis of stock market predictions," *Artif Intell Rev*, vol. 53, no. 4, pp. 3007–3057, 2020.
- [15] S. Patro and K. K. Sahu, "Normalization: A preprocessing stage," *arXiv preprint arXiv:1503.06462*, 2015.
- [16] K. P. Sinaga and M.-S. Yang, "Unsupervised K-Means Clustering Algorithm," *IEEE Access*, vol. 8, pp. 80716–80727, 2020, doi: 10.1109/ACCESS.2020.2988796.
- [17] M. Mathur, S. Mhadalekar, S. Mhatre, and V. Mane, "Algorithmic trading bot," in *ITM Web of Conferences*, EDP Sciences, 2021, p. 03041.
- [18] S. Visa, B. Ramsay, A. L. Ralescu, and E. Van Der Knaap, "Confusion matrix-based feature selection.," *Maics*, vol. 710, no. 1, pp. 120–127, 2011.
- [19] L. Khaidem, S. Saha, and S. R. Dey, "Predicting the direction of stock market prices using random forest," *arXiv preprint arXiv:1605.00003*, 2016.
- [20] J. Dewani, M. Ghodawat, S. Vadhera, M. Patel, A. Mishra, and N. Kotha, "A research study on awareness regarding crypto currency among investors," *International Journal on Integrated Education*, vol. 3, no. 3, pp. 114–125, 2020.
- [21] F. M. Sinaga, M. Jonas, and A. Halim, "Stock trend prediction using SV-kNNC and SOM," in *2019 Fourth International Conference on Informatics and Computing (ICIC)*, IEEE, 2019, pp. 1–5.
- [22] C. Yuan and H. Yang, "Research on K-value selection method of K-means clustering algorithm," *J (Basel)*, vol. 2, no. 2, pp. 226–235, 2019.
- [23] S. Marsland, *Machine learning: an algorithmic perspective*. CRC press, 2015.
- [24] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer, 2009.