

## Deteksi Kecurangan Transaksi Online Dan Transaksi Kartu Kredit Menggunakan Algoritma Support Vector Machine (Svm) Secara Real-Time

Putri Kurni Wati<sup>1</sup>, Widiya<sup>2</sup>, Suci Wulandari<sup>3</sup>, Mhd. Furqan<sup>4</sup>  
<sup>1,2,3,4</sup> Sains dan Teknologi, Ilmu Komputer, Universita Islam Negeri Sumatera Utara

---

### ABSTRACT

Recent technological advancements have triggered a significant increase in the use of credit cards, leading to a higher percentage of credit card fraud in both offline and online transactions. Although measures such as PIN codes, embedded chips, and additional security keys like tokens have enhanced credit card security, financial institutions are required to strengthen usage controls and implement real-time monitoring systems to promptly identify and mitigate suspicious activities. As cashless transactions become increasingly popular, credit card fraud has become one of the most common types of fraud, causing substantial losses for both financial institutions and individuals in real life. Fraud detection must be extremely fast and effective. With over one million transactions occurring daily, it is difficult to monitor each transaction individually. Therefore, effective fraud detection systems are used to distinguish between genuine and fraudulent transactions. In this journal, the Support Vector Machine (SVM) algorithm is utilized to build a model to address the issue of credit card fraud detection. This study evaluates the performance of the Support Vector Machine (SVM) algorithm in identifying suspicious financial transactions. The research aims to develop an accurate and efficient fraud detection model by leveraging the SVM algorithm. In this study, a dataset comprising both legitimate and fraudulent online and offline transactions will be used. The research process includes data preprocessing, model training using SVM, and performance evaluation based on accuracy, precision, recall, and F1-score metrics. It is expected that the developed model will be able to detect fraud with a high level of accuracy and can be effectively implemented in both online and offline transaction systems to enhance security and user trust.

---

#### *Keywords:*

Credit Card, Digital Wallet, Fraud Detection, Financial Transactions, SVM, Python, Library, Machine Learning.



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

---

#### *Corresponding Author:*

**Putri Kurni Wati**

Universita Islam Negeri Sumatera Utara

Email: [Kurniawatiputri61@gmail.com](mailto:Kurniawatiputri61@gmail.com)

---

## Introduction

E-commerce memainkan peran penting dalam aktivitas manusia sehari-hari. Penggunaan internet dan perangkat IOT telah meningkat pesat dan menghasilkan volume data yang sangat besar. Data meningkat karena pengumpulan data dari berbagai bentuk seperti data terstruktur dan semi-terstruktur, yang umumnya disebut sebagai Big Data. Volume data yang besar ini menjadi lebih penting bagi organisasi untuk analisis data, terutama bagi bank untuk melacak transaksi secara real-time. Keamanan kartu kredit telah menjadi ancaman besar di dunia modern karena luasnya penggunaan online dan transaksi data yang terus berkembang untuk dianalisis guna mendeteksi penipuan. Karena lebih mudah bagi pengguna untuk berbelanja, Karena penggunaan kartu kredit dan m-banking telah meningkat dan data telah tumbuh drastis, maka menjadi tantangan bagi bank untuk mendeteksi transaksi penipuan secara manual dan memblokir penggunaannya. (Haritija 2021).

Pengertian kecurangan mengacu pada Black's Law Dictionary yaitu "suatu tindakan dengan percobaan penipuan atau pelanggaran oleh satu orang atau lebih yang pada umumnya untuk mendapatkan keuntungan finansial" Salah satu kategori kecurangan adalah penipuan kartu kredit dan transaksi online. Tren kecurangan dalam transaksi terus meningkat yang mengakibatkan kerugian uang dalam jumlah besar setiap tahunnya. Diperkirakan kerugian meningkat setiap tahunnya dengan angka doubledigit pada tahun 2020. Hal ini dikarenakan kartu fisik tidak diperlukan dalam lingkungan transaksi online, dan informasi dari kartu sudah cukup untuk menyelesaikan pembayaran. Hal ini membuat terjadinya kecurangan menjadi lebih mudah dari sebelumnya. Salah satu mekanisme untuk meminimalkan risiko penipuan kartu kredit dan transaksi online adalah dengan menggunakan teknik deteksi transaksi yang sedang berlangsung untuk mengidentifikasi potensi penipuan. Beberapa teknik Machine Learning dan data mining telah digunakan dalam penelitian tentang deteksi penipuan kartu kredit (Indra 2020).

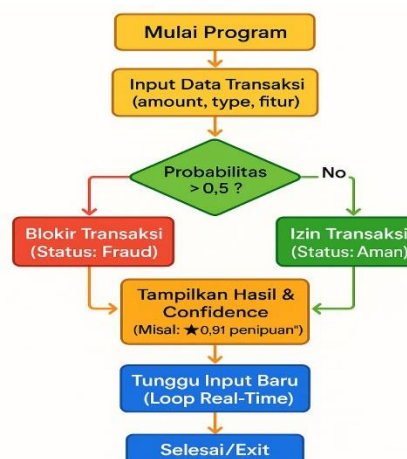
Perkembangan teknologi telah mengubah cara transaksi keuangan, dengan transaksi online dan penggunaan kartu kredit menjadi dominan. Namun, penipu sering menggunakan informasi kartu kredit, seperti nomor kartu dan kode keamanan untuk melakukan transaksi penipuan secara online. Hal tersebut membuka peluang bagi pelaku kejahatan untuk memanfaatkan situasi. Dampaknya tidak hanya pada kerugian finansial, tetapi potensi pencurian identitas yang serius. Jadi penting bagi masyarakat untuk meningkatkan kewaspadaan dan kebijaksanaan dalam melakukan transaksi online (Berliana, 2024). Perkembangan teknologi digital telah mempengaruhi banyak aspek kehidupan, termasuk dalam bidang perdagangan dan transaksi keuangan. Saat ini, banyak orang lebih memilih transaksi online karena memberikan kemudahan, kecepatan, dan fleksibilitas. Namun, disisi lain ada resiko besar yang berhubungan dengan keamanan data dan kepercayaan konsumen, terutama yang berkaitan dengan penipuan dalam transaksi online. Menemukan penipuan dalam transaksi online menjadi tantangan yang signifikan karena pola penipuan sering kali berubah dan sulit untuk di perediksi (Handry, 2024).

Kartu kredit adalah kartu yang diterbitkan kepada konsumen (pemegang kartu), yang dapat menggunakannya untuk melakukan pembelian hingga batas tertentu atau menarik uang tunai dari lokasi mana pun. Dengan menggunakan kartu kredit, bank menyediakan berbagai layanan kepada konsumennya. Misalnya, dengan memundahkannya ke tagihan berikutnya, hal ini memungkinkan pelanggan untuk membayar di kemudian hari. Penipuan adalah perilaku ilegal atau kriminal yang bertujuan untuk mendapatkan keuntungan finansial atau pribadi (Odeyale, 2024). Transaksi menggunakan kartu kredit maupun m-banking merupakan salah satu metode pembayaran yang terbilang cukup praktis karena pelanggan dapat melakukan transaksi walaupun tidak membawa uang tunai contohnya (e-wallet), Gopay, OVO, DANA, ShopeePay, atau LinkAja, serta aplikasi mobile, Livin, by Mandiri, atau BRImo. Seiring dengan berkembangnya teknologi, aksi penipuan kartu kredit dapat terdeteksi dengan pembangunan model dari data histori transaksi melalui bantuan Machine Learning. Machine

Learning menggunakan data transaksi kartu kredit dan transaksi online memiliki tantangan tersendiri, permasalahan utamanya adalah biasanya data transaksi memiliki dimensi sangat besar dan kelas yang tidak seimbang yang mana kelas transaksi asli lebih banyak dibandingkan kelas transaksi penipuan. Data yang tidak seimbang membutuhkan perhatian khusus karena berpengaruh besar terhadap nilai akurasi. Selain itu dalam proses transaksi kartu kredit durasi waktu antara pelanggan melakukan pembayaran dan pembayaran sampai ke rekening tujuan biasanya sangat singkat. Untuk mencegah kerugian yang besar, pendeteksian penipuan kartu kredit haruslah berjalan dengan cepat, maka diperlukan proses pendeteksian dengan waktu komputasi yang singkat (Lailan, 2024).

## METHODS

Penelitian ini dilakukan dengan tujuan menciptakan model yang mampu mendeteksi penipuan dengan tepat dan efisien menggunakan algoritma SVM. Dalam studi ini, data yang akan di jadikan dataset terdiri dari transaksi online dan transaksi kartu kredit yang mencakup transaksi yang valid serta merupakan penipuan. Proses penelitian mencakup tahap-tahap, desain penelitian, dataset, pra-pemrosesan data (preprocessing), pelatihan model, prediksi dan evaluasi model, implementasi real-time, tools dan implementasi. Dengan berdasarkan model SVM dan model FastAPI diharapkan, model yang dibuat bisa mengenali penipuan dengan tingkat ketepatan yang tinggi dan dapat di terapkan secara efisien dalam sistem transaksi daring untuk memperkuat keamanan dan kepercayaan pengguna.



Gambar 1. flowchart sistem deteksi penipuan real-time

### 1. Desain Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan metode eksperimen dan studi kasus pada transaksi online dan transaksi kartu kredit. Tujuannya adalah membangun model deteksi penipuan (fraud detection) yang mampu memblokir transaksi secara real-time menggunakan algoritma Support Vector Machine (SVM).

### 2. Dataset

Dataset yang digunakan terdiri dari data transaksi online dan transaksi kartu kredit yang diklasifikasikan sebagai transaksi sah (legit) dan penipuan (fraud). Setiap data transaksi memiliki fitur seperti:

- a. Jumlah transaksi (amount)
- b. Jenis transaksi (transaction\_type, seperti online atau offline)

- c. Fitur perilaku pelanggan (misalnya waktu, lokasi, device ID)
  - d. Label 1 untuk fraud dan 0 untuk transaksi sah
3. Pra-pemrosesan Data (Preprocessing)
- Sebelum digunakan untuk pelatihan model, data terlebih dahulu diproses melalui tahapan:
- a. Pembersihan data (data cleaning): menghapus nilai kosong atau tidak logis
  - b. Enkoding data kategorikal: seperti `transaction_type` menjadi nilai numerik
  - c. Normalisasi fitur numerik: menggunakan `MinMaxScaler` agar fitur memiliki skala seragam
  - d. Split data: menjadi data latih dan data uji (misalnya 80%:20%)
4. Pelatihan Model
- Model klasifikasi yang diterapkan adalah Support Vector Machine (SVM) dengan probabilitas keluaran yang di aktifkan. Proses pelatihan dilakukan pada data yang telah dinormalisasi. Model ini dilatih untuk membedakan pola antara transaksi sah dan penipuan berdasarkan fitur-fitur transaksi.
5. Prediksi dan Evaluasi Model
- Setelah pelatihan, model diuji dengan data uji untuk mengevaluasi performa berdasarkan metrik:
- a. Akurasi: Persentase klasifikasi benar terhadap seluruh data
  - b. Presisi: Kemampuan model dalam mendeteksi fraud yang benar
  - c. Recall (Sensitivity): Kemampuan model menemukan semua kasus fraud
  - d. F1-Score: Harmoni antara presisi dan recall
6. Implementasi Real-Time
- Model yang telah dilatih diintegrasikan ke dalam sistem real-time dengan alur sebagai berikut:
- a. Sistem menerima input data transaksi baru
  - b. Data transaksi dinormalisasi secara instan menggunakan parameter yang sama dari pelatihan
  - c. Model memprediksi probabilitas transaksi tersebut merupakan penipuan
  - d. Jika probabilitas  $> 0.5$ , maka transaksi diblokir (tidak diproses)
  - e. Jika  $\leq 0.5$ , maka transaksi dilanjutkan
  - f. Alur ini divisualisasikan dalam flowchart sistem real-time, yang menggambarkan proses dari input hingga keputusan akhir.
7. Tools dan Implementasi
- a. Bahasa pemrograman: Python
  - b. Library: `scikit-learn`, `pandas`, `numpy`
  - c. Model: `SVC (probability=True)`

## RESULTS AND DISCUSSION

Platform real-time dijalankan melalui CLI (Command Line) dan dapat dikembangkan lebih lanjut menjadi API atau dashboard.

### 1. Input Transaksi

Setiap kali terjadi transaksi kartu kredit, data utama (misalnya jumlah transaksi dalam field `Amount` dan waktu transaksi) dikirim ke sistem deteksi melalui endpoint API (metode HTTP POST). FastAPI menerima data JSON ini dan mem-parsing-nya ke dalam model Pydantic yang telah didefinisikan (mis. Kelas `Transaction` dengan atribut `Amount`, `Timestamp`, dll). Data mentah tersebut kemudian diubah menjadi struktur yang sesuai (misalnya objek Data Frame `pandas`) agar dapat diproses oleh model Machine Learning.

### 2. Preprocessing Data

Sebelum diprediksi, fitur transaksi dinormalisasi agar sesuai dengan asumsi model SVM. Sebagai contoh, data JSON yang berisi angka nominal diubah menjadi `DataFrame` lalu

fitur-fitur numerik diekstrak. Standarisasi fitur sangat penting untuk SVM karena algoritma ini tidak skala-invarian . Oleh karena itu, setiap atribut numerik (seperti jumlah transaksi) dinormalisasi menggunakan StandardScaler dari scikitlearn sehingga nilai mean menjadi 0 dan varians menjadi 1. Langkah ini memastikan fitur-fitur yang memiliki rentang nilai berbeda menjadi sebanding, sehingga model SVM dapat bekerja optimal (misalnya jika ada fitur waktu, fitur tersebut juga dapat diubah ke skala yang sesuai dan diskalakan).

### 3. Prediksi dengan Model SVM

Model SVM telah dilatih sebelumnya (offline) menggunakan data transaksi berlabel fraud/tidak fraud. Model dan objek scaler yang dihasilkan (contoh disimpan dengan joblib ) kemudian dimuat ke memori saat aplikasi berjalan . Data transaksi yang telah distandarisasi diteruskan ke model SVM untuk klasifikasi. Dengan konfigurasi SVM probability=True , metode model.predict() menghasilkan label kelas (misalnya 0 = sah, 1 = fraud), sedangkan model.predict\_proba() menghasilkan probabilitas prediksi fraud . Hasil ini sesuai dengan tujuan sistem yaitu memprediksi kemungkinan fraud dari setiap transaksi.

### 4. Integrasi Model dengan FastAPI

Aplikasi FastAPI menyediakan endpoint (misalnya /predict/ ) untuk penanganan request real-time. Contoh implementasi: setelah FastAPI diinisialisasi, model SVM dan scaler di-load di awal (contoh kode: model = joblib.load(...) , scaler = joblib.load(...) ) . Endpoint tersebut menerima data transaksi dalam bentuk JSON, mem-parsing-nya menjadi objek Transaction , lalu mengubahnya menjadi DataFrame pandas . Data ini kemudian diskalakan (misalnya scaled\_data = scaler.transform(data) ) dan diberikan ke model untuk prediksi ( model.predict , model.predict\_proba ).

### 5. Penentuan Status Transaksi

Hasil prediksi model berupa probabilitas fraud kemudian digunakan untuk menentukan status transaksi. Umumnya dipakai ambang batas (threshold) antara 0 dan 1: jika , transaksi diklasifikasikan fraud, jika tidak dianggap sah . Dalam konteks ini, jika probabilitas fraud melebihi threshold (misalnya 0.5), maka sistem menetapkan status “Diblokir” karena dianggap penipuan; jika di bawah threshold, transaksi dianggap aman dan diproses normal. Pendekatan threshold-based ini umum dalam fraud detection, seperti digambarkan pada gambar di atas . Dengan cara ini, sistem dapat secara otomatis mengambil tindakan (blokir atau lanjutkan) berdasarkan skor model.

### 6. Format Respons API

API FastAPI mengembalikan respons dalam format JSON yang berisi hasil prediksi dan metadata. Field yang dikirim biasanya mencakup prediksi label (misalnya "prediction": 1 untuk fraud atau 0 untuk legitimate), probabilitas kejadian fraud ( "probability": 0.87 jika 87% yakin fraud), dan status transaksi sesuai kebijakan (misalnya "status": "Diblokir" atau "Diproses").

Sistem ini menerima input data transaksi berupa Java Script object notation (JSON) melalui endpoint HTTP POST di FastAPI (/deteksi) data ini kemudian:

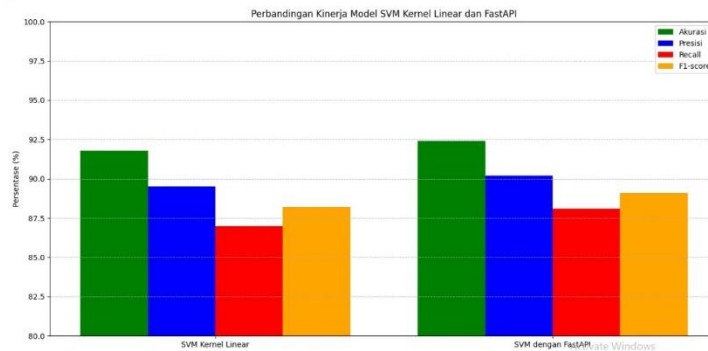
1. Diparsing oleh FastAPI ke dalam objek transaction lalu diubah ke format data frame dan distandarisasi menggunakan standard scaler.
2. Diprediksi menggunakan model SVM yang sudah dilatih (SVC (probability= True)) lalu Hasil prediksi dan probabilitas dikembalikan dalam response JSON.

Penelitian ini menggunakan dataset transaksi online dan transaksi kartu kredit yang terdiri dari data transaksi yang valid dan data penipuan. Dataset terbagi menjadi dua bagian : 60% dialokasikan untuk pelatihan model dan 40% digunakan untuk pengujian. Penerapan SVM dilakukan dengan menggunakan berbagai parameter dan kernel untuk mencapai hasil yang terbaik. Setelah menyelesaikan proses pelatihan dan pengujian, model SVM dievaluasi berdasarkan beberapa metric utama seperti akurasi, presisi, recal, dan F1-Score. Hasil evaluasi kinerja model SVM dengan menggunakan model FastAPI di bandingkan dengan kernel linear

untuk menentukan kinerja terbaik, dengan hasil akurasi, presisi, recall, dan F1-Score. Berikut adalah hasil perbandingan kinerja model SVM dengan model FastAPI :

**Tabel 1.** Perbandingan kinerja model SVM dengan model FastAPI

Model	Akurasi (%)	Presisi (%)	Recall (%)	F1-score (%)
SVM Kernel Linear	91.8	89.5	87.0	88.2
SVM dengan FastAPI	92.4	90.2	88.1	89.1



Berikut adalah hasil dari pembuktian transaksi yang dicurigai dan transaksi normal, beserta penjelasan di bawah ini:

```

Response body
{
  "prediksi": 1,
  "probabilitas": 0.8217232695522696,
  "status": "⚠ Diblokir - Transaksi dicurigai penipuan"
}

Response headers
content-length: 106
content-type: application/json
date: Sun, 18 May 2025 07:30:17 GMT
server: uvicorn

```

*Gambar 2. Hasil uji coba transaksi dicurigai penipuan*

Contoh Transaksi terindikasi penipuan :

1. prediksi: 1 → Model memprediksi transaksi ini adalah fraud.
2. probabilitas: 0.82 → Model yakin 82% bahwa ini adalah transaksi penipuan.
3. status: "Diblokir" → Karena probabilitas > 0.5, sistem memblokir transaksi

```

Response body
{
  "prediksi": 0,
  "probabilitas": 0.9862405784854752,
  "status": "✅ Diterima - Transaksi normal"
}

Response headers
content-length: 91
content-type: application/json
date: Sun, 18 May 2025 07:24:13 GMT
server: uvicorn

```

*Gambar 3. Hasil uji coba transaksi normal/tidak dicurigai*

Contoh Transaksi normal :

1. prediksi: 0 → Model menyimpulkan bahwa transaksi bukan penipuan.
2. probabilitas: 0.98 → Model yakin 98% bahwa ini adalah transaksi sah.
3. status: "Diterima" → Karena skor < 0.5 untuk fraud, transaksi diteruskan.

## CONCLUSION

Pelelitian ini berhasil mengembangkan dan mengimplementasikan sistem deteksi penipuan transaksi online dan kartu kredit secara real-time dengan memanfaatkan algoritma Support Vector Machine (SVM) yang diintegrasikan menggunakan FastAPI. Sistem mampu melakukan klasifikasi terhadap transaksi sah dan penipuan secara cepat dan efisien dengan pendekatan berbasis skor probabilitas. Model yang dikembangkan telah dievaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil evaluasi menunjukkan bahwa integrasi model SVM dengan FastAPI memberikan performa yang lebih baik dibandingkan dengan model SVM standar menggunakan kernel linier. Hal ini terlihat dari nilai akurasi sebesar 92.4%, presisi 90.2%, recall 88.1%, dan F1-score 89.1%. Sistem ini juga mampu memberikan keputusan secara otomatis terhadap status transaksi berdasarkan ambang batas probabilitas, serta mengembalikan hasil deteksi dalam format JSON melalui endpoint API. Dengan pendekatan ini, sistem deteksi penipuan yang dibangun dinilai cukup efektif untuk meningkatkan keamanan dan kepercayaan pengguna dalam bertransaksi, baik secara online maupun melalui kartu kredit.

## REFERENCES

- [1] Bhayyel, A., Singh, G. K., Dhamnaskar, S., Patil, S., & Phulari, S. V. (2021). Deteksi Penipuan Kartu Kredit Menggunakan Isolasi Hutan. *Jurnal Internasional Kelajuan Terkini dalam Topik Multidisiplin*, 2(6), 118–120. <https://www.ijramt.com>.
- [2] Bodelpudi, H. (2021). Credit Card Fraud Detection Using Unsupervised Machine Learning Algorithms. *International Journal of Computer Trends and Technology (IJCTT)*, 69(8), 1–3. <https://doi.org/10.14445/22312803/IJCTT-V69I8P101>.
- [3] Musiliudelehn, O. K., Moruff, O. A., Taofelekat, S.-I. T., & Kayodel, S. M. (2024). Model Deteksi Penipuan Kartu Kredit Support Vector Machine Berdasarkan Datasets Ketidaxselimbangan Tinggi. *Jurnal Komputer untuk Masyarakat*, 5(2), 85–94. <https://doi.org/10.17509/jcs.v5i2.70802>.
- [4] Tarissa, B. V., & Delwayanto, T. (2024). Penerapan Machine Learning dan Deep Learning pada Peningkatan Deteksi Credit Card Fraud: A Systematic Literature Review. *Diponegoro Journal of Accounting*, 13(3), 1–15. <http://ejournal-s1.undip.ac.id/index.php/accounting>.
- [5] Waspada, I., Bahtiar, N., Wirawan, P. W., & Awan, B. D. A. (2020). Analisis Kinerja Algoritma Hutan Isolasi pada Deteksi Penipuan Transaksi Kartu Kredit. *Khazanah Informatika*, 6(2), 165–171. <http://journals.ums.ac.id/index.php/khif>.
- [6] Xia, J. (2022). Deteksi Penipuan Kartu Kredit Berdasarkan Support Vector Machine. *Prosiding IPIIS 2022: Sorotan dalam Sains, Teknik dan Teknologi*, 23, 93–97.
- [7] Ningsih, P. T. S., Gusvarizon, M., & Helrmawan, R. (2022, 30 September). Analisis sistem deteksi penipuan transaksi kartu kredit menggunakan algoritma Pembelajaran Mesin. *Jurnal Teknologi Informasi dan Komputer*, 8(2), 386–401. <https://doi.org/10.37012/jtik.v8i2.1306>.