

Penerapan Algoritma Knuth-Morris-Pratt pada Representasi Simbolik *Limit Order Book* untuk Deteksi Dini *Micro-Crash* di Pasar Kripto

Rava Khoman Tuah Saragih – 13524107

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jalan Ganesha 10, Bandung

E-mail: ravasaragih7@gmail.com, 13524107@std.stei.itb.ac.id

Abstract—Micro-crash merupakan anomali struktural di pasar kripto yang ditandai oleh penurunan harga secara drastis dalam rentang waktu yang sangat singkat. Penelitian ini mengusulkan sistem deteksi dini micro-crash yang mengintegrasikan algoritma Symbolic Aggregate approXimation (SAX) dengan algoritma pencocokan string Knuth-Morris-Pratt (KMP). Deret waktu Order Book Imbalance (OBI) yang diturunkan dari data Limit Order Book (LOB) dikonversi menjadi representasi simbolik menggunakan SAX, kemudian dipindai oleh KMP untuk mendeteksi pola-pola karakteristik yang mengindikasikan vakum likuiditas sebelum terjadinya micro-crash. Eksperimen dilakukan pada data simulasi BTC/USDT selama 7 hari dengan 8 kejadian crash yang disuntikkan. Hasil menunjukkan bahwa konfigurasi terbaik (segment size $w=10$, alphabet size $a=4$) mencapai precision 0,381, recall 1,000, dan F1-score 0,552, mengungguli metode baseline threshold OBI yang hanya mencapai F1-score 0,174. Dari sisi efisiensi komputasi, KMP membutuhkan 22% lebih sedikit perbandingan karakter dibandingkan pencarian brute force pada skala data 500.000 karakter, dengan rata-rata speedup 1,29x. Sistem ini menghasilkan peringatan rata-rata 11,5 menit sebelum terjadinya micro-crash.

Kata Kunci—pencocokan string, KMP, SAX, limit order book, micro-crash, deteksi anomali, kripto.

I. PENDAHULUAN

Pasar mata uang kripto dikenal dengan tingkat volatilitas yang sangat tinggi. Salah satu fenomena yang secara berulang menghantui pelaku pasar adalah *micro-crash* atau *flash crash*, yaitu penurunan harga aset secara drastis dalam jendela waktu yang sangat singkat, sering kali hanya dalam hitungan menit, yang kemudian diikuti oleh pemulihan harga [6]. Fenomena ini bukan disebabkan oleh perubahan fundamental nilai aset, melainkan oleh kegagalan mikrostruktur pasar, khususnya hilangnya likuiditas pada Limit Order Book (LOB) secara tiba-tiba [5].

Pendekatan konvensional untuk mendeteksi anomali pasar umumnya mengandalkan metode *threshold-based*, di mana peringatan dikirimkan ketika sebuah metrik skalar (misalnya Order Book Imbalance) melampaui nilai tertentu [4]. Pendekatan ini memiliki kelemahan mendasar: ia hanya memperhatikan besaran nilai pada satu titik waktu tanpa memperhitungkan bentuk dan urutan perubahan sepanjang waktu, sehingga cenderung menghasilkan banyak *false positives*.

Di sisi lain, pendekatan berbasis *deep learning* seperti arsitektur DeepLOB [11] mampu menangkap pola temporal yang kompleks, namun memiliki kelemahan dari segi interpretabilitas dan kebutuhan sumber daya komputasi yang besar.

Penelitian ini mengusulkan pendekatan alternatif yang menggabungkan dua teknik algoritmik klasik. Pertama, algoritma Symbolic Aggregate approXimation (SAX) [2] digunakan untuk mengonversi deret waktu OBI yang kontinu menjadi representasi string diskrit, sehingga profil perubahan likuiditas sebelum micro-crash dapat dikodekan sebagai pola teks. Kedua, algoritma Knuth-Morris-Pratt (KMP) [1] digunakan untuk memindai deret simbolik tersebut guna mendeteksi kemunculan pola-pola crash secara efisien dengan kompleksitas waktu linier $O(n + m)$.

Kontribusi penelitian ini adalah sebagai berikut: (1) perancangan pipeline deteksi dini micro-crash yang mengintegrasikan transformasi SAX pada data OBI dengan pencarian pola KMP, (2) evaluasi eksperimental terhadap berbagai konfigurasi parameter SAX, (3) perbandingan performa deteksi terhadap metode baseline threshold, dan (4) perbandingan efisiensi komputasi algoritma KMP terhadap pencarian brute force.

II. LANDASAN TEORI

A. Limit Order Book dan Order Book Imbalance

Limit Order Book (LOB) merupakan buku besar elektronik yang mencatat seluruh pesanan batas (*limit orders*) yang belum tereksekusi di sebuah bursa [3]. LOB terbagi menjadi dua sisi: sisi penawaran beli (*bid*), yang merepresentasikan harga tertinggi yang bersedia dibayarkan pembeli, dan sisi penawaran jual (*ask*), yang menunjukkan harga terendah yang bersedia diterima penjual [4].

Data *market depth* atau data Level 2 memberikan visibilitas terhadap volume pesanan pada beberapa tingkat harga teratas di kedua sisi LOB. Dari data ini, metrik Order Book Imbalance (OBI) dapat dihitung sebagai ukuran asimetri antara tekanan beli dan tekanan jual. Secara matematis, untuk N tingkat harga teratas, OBI didefinisikan sebagai

$$OBI_N = \frac{\sum_{i=1}^N V_i^{\text{bid}} - \sum_{i=1}^N V_i^{\text{ask}}}{\sum_{i=1}^N V_i^{\text{bid}} + \sum_{i=1}^N V_i^{\text{ask}}} \quad (1)$$

di mana V_i^{bid} dan V_i^{ask} merepresentasikan volume pesanan beli dan jual pada kedalaman ke- i . Nilai OBI selalu berada dalam rentang $[-1, 1]$. Nilai OBI positif mengindikasikan dominasi tekanan beli, sedangkan nilai OBI negatif mengindikasikan dominasi tekanan jual [4].

Fenomena *liquidity vacuum* terjadi ketika pembuat pasar algoritmik (*algorithmic market makers*) secara serentak membatalkan pesanan beli di LOB karena mendeteksi *toxic order flow* atau lonjakan volatilitas [6]. Pencabutan likuiditas ini menyebabkan OBI terjun ke nilai negatif ekstrem dan menghilangkan penyangga harga, sehingga pesanan jual sekecil apa pun dapat memicu penurunan harga yang dramatis [5].

B. Micro-Crash di Pasar Kripto

Micro-crash didefinisikan sebagai penurunan harga *peak-to-trough* minimum sebesar 3,5% hingga 5,0% yang terjadi dalam rentang waktu 5 hingga 30 menit, dengan kondisi pemulihan (*rebound*) minimum 60% dalam waktu 60 menit setelah titik terendah tercapai [12]. Penyebab utama micro-crash meliputi tiga mekanisme yang saling mengamplifikasi:

- 1) **Kaskade likuidasi:** penurunan harga memicu eksekusi margin otomatis pada posisi derivatif berleverage tinggi, yang selanjutnya menekan harga lebih dalam lagi secara siklis [13].
- 2) **Penarikan likuiditas simultan:** algoritma HFT berhenti beroperasi saat volatilitas meledak, sehingga likuiditas beli menghilang pada saat paling dibutuhkan [6].
- 3) **Kesalahan operasional:** pesanan jual dengan volume abnormal yang dikirimkan ke pasar tipis [7].

C. Symbolic Aggregate approXimation (SAX)

Algoritma SAX, yang diusulkan oleh Lin, Keogh, et al. [2], merupakan metode transformasi deret waktu kontinu menjadi representasi string diskrit. Proses transformasi SAX dilaksanakan melalui tiga tahap.

Tahap 1: Normalisasi-Z. Deret waktu $C = c_1, c_2, \dots, c_n$ dinormalisasi sehingga memiliki rata-rata $\mu = 0$ dan simpangan baku $\sigma = 1$:

$$c'_i = \frac{c_i - \mu}{\sigma} \quad (2)$$

Tahap 2: Piecewise Aggregate Approximation (PAA). Deret yang telah dinormalisasi direduksi dari panjang n menjadi w segmen ($w \ll n$). Setiap segmen direpresentasikan oleh nilai rata-rata aritmetiknya:

$$\bar{c}_i = \frac{w}{n} \sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w} \cdot i} c'_j \quad (3)$$

Tahap 3: Diskritisasi dengan breakpoints Gaussian. Nilai-nilai PAA dipetakan ke simbol alfabet diskrit berdasarkan titik potong (*breakpoints*) distribusi normal standar $\mathcal{N}(0, 1)$. Untuk ukuran alfabet a , deret dibagi menjadi

TABLE I
PERBANDINGAN KOMPLEKSITAS ALGORITMA PENCOCOKAN STRING

Algoritma	Kasus Terbaik	Kasus Terburuk
Brute Force	$O(n)$	$O(n \times m)$
Boyer-Moore	$O(n/m)$	$O(n \times m)$
Rabin-Karp	$O(n + m)$	$O(n \times m)$
KMP	$O(n)$	$O(n + m)$

a partisi ekuiprobabel dengan luas area $1/a$ di bawah kurva Gaussian [2]. Fungsi pemetaan didefinisikan sebagai

$$\text{SAX}(\bar{c}_i) = \text{huruf ke-}j \quad \text{jika} \quad \beta_{j-1} \leq \bar{c}_i < \beta_j \quad (4)$$

dengan $\beta_0 = -\infty$ dan $\beta_a = +\infty$. Sebagai contoh, untuk $a = 4$, breakpoints ditetapkan pada $[-0,67; 0; 0,67]$ dan alfabet yang digunakan adalah $\{a, b, c, d\}$.

Pemilihan parameter w (panjang kata) dan a (ukuran alfabet) sangat menentukan keseimbangan antara resolusi informasi dan ketahanan terhadap *noise*. Alfabet yang terlalu kecil menyamarkan varians mikrostruktural, sedangkan alfabet yang terlalu besar meningkatkan kerumitan string dan risiko kecocokan palsu [10].

D. Algoritma Knuth-Morris-Pratt (KMP)

Algoritma KMP [1] merupakan algoritma pencocokan string deterministik yang beroperasi dalam waktu linier. Terobosan utama KMP terletak pada fase prakomputasi yang membangun array *Longest Proper Prefix which is also Suffix* (LPS), dilambangkan π . Setiap elemen $\pi[i]$ menyimpan panjang maksimum prefiks sejati dari pola $P[0..i]$ yang juga merupakan sufiks dari sub-pola tersebut [1].

Array LPS memungkinkan KMP untuk menghindari *backtracking* pada teks target ketika terjadi ketidakcocokan karakter. Ketika karakter teks $T[i]$ tidak cocok dengan karakter pola $P[j]$, KMP merujuk nilai $\pi[j - 1]$ untuk menentukan posisi baru pada pola tanpa memundurkan posisi pada teks [9].

Kompleksitas waktu. Jika n adalah panjang teks dan m adalah panjang pola, KMP menjamin kompleksitas waktu $O(n + m)$. Fase prakomputasi LPS membutuhkan $O(m)$ dan fase pencarian membutuhkan $O(n)$, karena indeks teks tidak pernah berkurang [1].

Justifikasi pemilihan KMP. Untuk string SAX dengan ukuran alfabet kecil ($a \leq 5$), algoritma Boyer-Moore menjadi kurang efektif karena heuristik *bad character* jarang menghasilkan lompatan besar [8]. KMP lebih cocok karena performanya tidak bergantung pada ukuran alfabet dan indeks teks yang tidak pernah mundur menjadikannya ideal untuk pemrosesan data streaming [9].

III. METODOLOGI

A. Arsitektur Pipeline

Sistem deteksi dini micro-crash yang diusulkan terdiri dari lima komponen yang dieksekusi secara sekuensial, sebagaimana diilustrasikan pada Gambar 1.

Komponen pertama mengambil data *market depth* dari Binance API. Komponen kedua menghitung deret waktu OBI

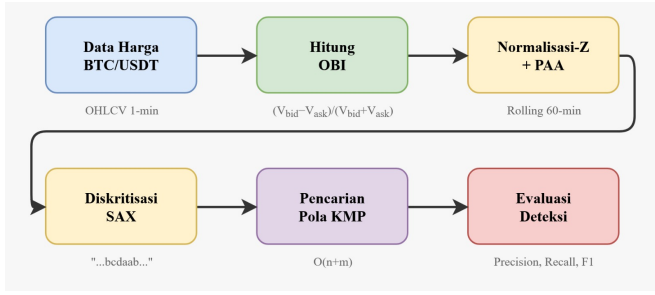


Fig. 1. Arsitektur pipeline deteksi dini micro-crash.

dari data tersebut menggunakan Persamaan (1). Komponen ketiga menerapkan normalisasi-Z berbasis *rolling window* dengan ukuran 60 menit untuk mengatasi non-stasioneritas sinyal OBI, dilanjutkan dengan reduksi PAA dan diskritisasi SAX sehingga dihasilkan string SAX kontinu. Komponen keempat menjalankan algoritma KMP untuk mencari kemunculan pola-pola crash pada string SAX tersebut. Komponen terakhir mengevaluasi kecocokan yang ditemukan dan mengirimkan peringatan.

B. Akuisisi dan Prapemrosesan Data

Data yang digunakan dalam penelitian ini adalah data simulasi yang mereplikasi karakteristik deret harga BTC/USDT pada interval 1 menit. Data dibangkitkan menggunakan model *Geometric Brownian Motion* dengan parameter drift $\mu = 0$ dan volatilitas harian $\sigma = 0,02$ (2%). Sebanyak 8 kejadian micro-crash disuntikkan ke dalam deret dengan magnitudo berkisar antara 3,5% hingga 8% dan durasi crash 5 hingga 12 menit.

Total data yang dibangkitkan mencakup 10.080 titik data (setara 7 hari), dengan periode simulasi 10 hingga 16 Agustus 2023. Volume beli dan jual disimulasikan secara terpisah, di mana rasio volume beli menurun drastis menjadi 10%–25% selama fase crash untuk mereplikasi penarikan likuiditas.

OBI proxy dihitung menggunakan rasio volume beli dan jual yang diagregasi dengan jendela bergulir 5 menit:

$$OBI_{\text{proxy}} = \frac{V_{\text{buy}}^{(5)} - V_{\text{sell}}^{(5)}}{V_{\text{buy}}^{(5)} + V_{\text{sell}}^{(5)}} \quad (5)$$

di mana $V_{\text{buy}}^{(5)}$ dan $V_{\text{sell}}^{(5)}$ adalah volume beli dan jual kumulatif dalam jendela 5 menit terakhir.

C. Konstruksi Pola Crash dan Transformasi SAX

Transformasi SAX pada deret OBI dilakukan dengan normalisasi-Z berbasis *rolling window* 60 menit, bukan normalisasi global. Pendekatan ini diperlukan karena OBI bukan sinyal stasioner; rata-rata dan variansnya berfluktuasi mengikuti rezim volatilitas pasar [10]. Normalisasi global berisiko menyembunyikan anomali lokal atau sebaliknya, merepresentasikan fluktuasi minor sebagai anomali signifikan.

Setelah normalisasi, deret OBI dibagi menjadi segmen-segmen berukuran w (segment size). Setiap segmen direduksi menjadi satu nilai PAA dan dipetakan ke satu karakter SAX.

Proses ini menghasilkan string SAX kontinu T yang menjadi teks masukan bagi algoritma KMP.

Pola-pola crash diperoleh secara data-driven: untuk setiap kejadian crash pada *ground truth*, substring SAX yang muncul tepat sebelum kejadian tersebut diekstraksi. Pola-pola yang memiliki rasio asosiasi tertinggi dengan kejadian crash dipilih sebagai pola pencarian. Sebagai contoh, pola “aadd” mengindikasikan transisi OBI dari kondisi normal-rendah ke kondisi negatif-ekstrem, merepresentasikan morfologi vakum likuiditas.

D. Pelabelan Ground Truth

Definisi operasional micro-crash yang digunakan untuk pelabelan kebenaran dasar (*ground truth*) adalah sebagai berikut: sebuah titik waktu t dilabeli sebagai pra-crash jika terjadi penurunan harga (*drawdown*) $\geq 3,5\%$ dalam jendela 10 menit ke depan, yaitu

$$\frac{P(t) - \min_{t \leq \tau \leq t+10} P(\tau)}{P(t)} \geq 0,035 \quad (6)$$

Evaluasi deteksi menggunakan matriks konfusi temporal dengan definisi sebagai berikut:

- **True Positive (TP)**: peringatan KMP yang diikuti oleh kejadian crash dalam jendela waktu yang ditetapkan.
- **False Positive (FP)**: peringatan KMP yang tidak diikuti oleh kejadian crash.
- **False Negative (FN)**: kejadian crash yang tidak didahului oleh peringatan KMP.

Metrik evaluasi yang digunakan adalah *precision*, *recall*, dan skor F1:

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN} \quad (7)$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (8)$$

IV. EKSPERIMEN DAN HASIL

A. Setup Eksperimen

Eksperimen diimplementasikan menggunakan Python dengan pustaka NumPy, Pandas, SciPy, dan Matplotlib. Seluruh eksperimen dijalankan dengan *seed* yang sama untuk memastikan reproduisibilitas. Enam konfigurasi parameter SAX diuji, dengan variasi segment size $w \in \{3, 5, 10\}$ dan alphabet size $a \in \{3, 4, 5\}$. Dataset terdiri dari 10.080 titik data 1 menit (7 hari) dengan 8 kejadian crash yang disuntikkan. Dari total data, 61 titik waktu (0,61%) dilabeli sebagai pra-crash.

B. Hasil Deteksi pada Berbagai Parameter SAX

Tabel II menyajikan perbandingan performa deteksi untuk setiap konfigurasi parameter.

Konfigurasi terbaik diperoleh pada $w=10$ dan $a=4$ dengan skor F1 sebesar 0,552. Konfigurasi ini menghasilkan 21 peringatan, di mana 8 di antaranya merupakan true positive (seluruh kejadian crash terdeteksi) dan 13 merupakan false positive.

TABLE II
PERBANDINGAN PERFORMA DETEKSI PADA BERBAGAI KONFIGURASI
PARAMETER SAX

Konfig.	TP	FP	FN	Prec.	Rec.	F1
$w=3, a=3$	8	186	0	0,041	1,000	0,079
$w=3, a=4$	8	72	0	0,100	1,000	0,182
$w=5, a=3$	8	116	0	0,065	1,000	0,121
$w=5, a=4$	8	33	0	0,195	1,000	0,327
$w=5, a=5$	7	14	1	0,333	0,875	0,483
$w=10, a=4$	8	13	0	0,381	1,000	0,552

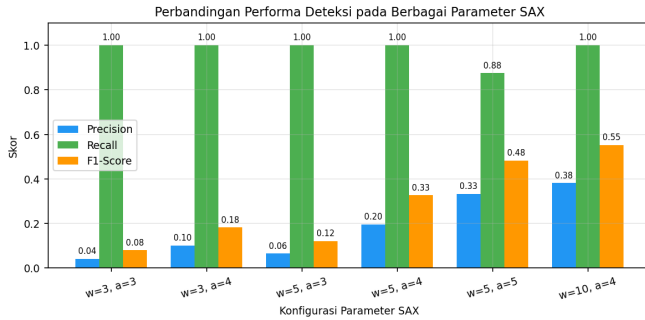


Fig. 2. Perbandingan precision, recall, dan F1-score pada berbagai konfigurasi parameter SAX.

Terdapat *trade-off* yang jelas antara precision dan recall. Konfigurasi dengan segment size kecil ($w=3$) menghasilkan recall sempurna namun dengan precision yang sangat rendah karena string SAX yang pendek meningkatkan probabilitas kecocokan acak. Sebaliknya, segment size yang lebih besar ($w=10$) menghasilkan string SAX yang lebih panjang dan diskriminatif, sehingga mengurangi jumlah peringatan palsu secara signifikan.

Peningkatan ukuran alfabet dari $a=3$ ke $a=4$ secara konsisten memperbaiki precision tanpa menurunkan recall, karena resolusi diskritisasi yang lebih tinggi memungkinkan pola crash dibedakan secara lebih tajam dari pola non-crash.

C. Perbandingan Performa Komputasi: KMP vs Brute Force

Tabel III menyajikan perbandingan waktu eksekusi dan jumlah perbandingan karakter antara algoritma KMP dan pencarian brute force pada berbagai ukuran teks.

KMP secara konsisten lebih cepat dibandingkan brute force dengan rata-rata speedup 1,29x. Pada skala 500.000 karakter, KMP menyelesaikan pencarian dalam 74 ms dibandingkan brute force yang membutuhkan 96 ms. Jumlah perbandingan karakter KMP tepat $O(n)$ karena indeks teks tidak pernah mundur, sedangkan brute force melakukan sekitar 1,33 kali lebih banyak perbandingan karena adanya pemindaian ulang pada ketidakcocokan parsial.

Speedup yang moderat (1,29x dan bukan ordo-orde besaran) disebabkan oleh ukuran alfabet SAX yang kecil ($a=4$). Dengan alfabet yang sempit, pencarian brute force juga cenderung menemukan ketidakcocokan pada karakter awal, sehingga tidak banyak pemindaian ulang yang terjadi. Keunggulan KMP

TABLE III
PERBANDINGAN WAKTU EKSEKUSI KMP VS BRUTE FORCE

n	KMP (ms)	BF (ms)	Komp. BF	Speedup
1.000	0,153	0,176	1.344	1,15x
5.000	0,727	0,951	6.685	1,31x
10.000	1,477	1,882	13.387	1,27x
50.000	7,528	9,468	66.621	1,26x
100.000	14,626	18,943	133.021	1,30x
500.000	74,020	95,765	664.199	1,29x

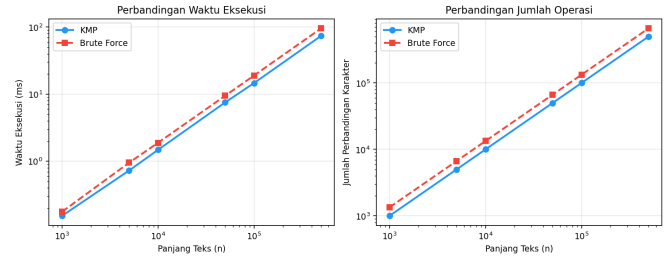


Fig. 3. Perbandingan waktu eksekusi (kiri) dan jumlah perbandingan karakter (kanan) antara KMP dan brute force pada skala logaritmik.

akan semakin terlihat pada pola yang memiliki banyak prefiks berulang atau pada alfabet yang lebih besar.

D. Perbandingan dengan Baseline Threshold OBI

Untuk membuktikan nilai tambah dari pendekatan berbasis pola, performa SAX+KMP dibandingkan dengan metode baseline yang mengirimkan peringatan semata-mata ketika nilai OBI mentah turun di bawah ambang batas tertentu, tanpa mempertimbangkan urutan temporal.

SAX+KMP secara signifikan mengungguli seluruh konfigurasi baseline threshold. Baseline threshold terbaik ($OBI \leq -0,3$) hanya mencapai F1-score 0,174, sedangkan SAX+KMP mencapai 0,552, sebuah peningkatan sebesar 3,17 kali lipat. Hasil ini mengkonfirmasi bahwa pendekatan berbasis pola temporal mampu menangkap sinyal crash yang tidak terdeteksi oleh pengamatan skalar sesaat, sekaligus mengurangi peringatan palsu.

E. Analisis Detection Lag

Pada konfigurasi terbaik ($w=10, a=4$), sistem menghasilkan peringatan rata-rata 11,5 menit sebelum terjadinya titik terendah micro-crash. Rentang waktu ini cukup memadai bagi sistem perdagangan algoritmik untuk mengeksekusi tindakan mitigasi, seperti menarik pesanan yang terekspos atau memperluas rentang kuotasi, sebagaimana disarankan dalam literatur [12].

V. PEMBAHASAN

Hasil eksperimen menunjukkan bahwa integrasi SAX dan KMP menghasilkan sistem deteksi yang secara signifikan lebih baik dibandingkan metode threshold konvensional. Beberapa poin diskusi yang relevan adalah sebagai berikut.

Pertama, keunggulan pendekatan berbasis pola terletak pada kemampuannya menangkap *bentuk* perubahan OBI sepanjang

TABLE IV
PERBANDINGAN SAX+KMP VS BASELINE THRESHOLD OBI

Metode	Prec.	Rec.	F1
SAX+KMP ($w=10, a=4$)	0,381	1,000	0,552
Threshold OBI $\leq -0,3$	0,133	0,250	0,174
Threshold OBI $\leq -0,4$	0,000	0,000	0,000
Threshold OBI $\leq -0,5$	0,000	0,000	0,000
Threshold OBI $\leq -0,6$	0,000	0,000	0,000

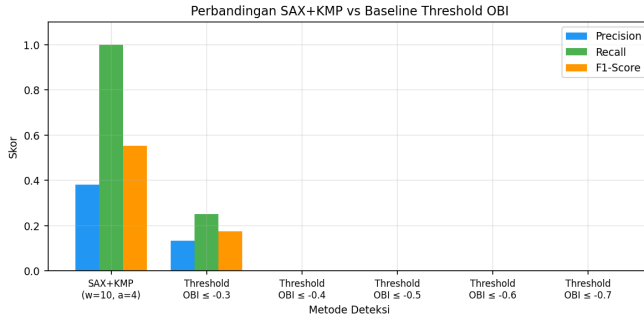


Fig. 4. Perbandingan performa deteksi SAX+KMP terhadap metode baseline threshold OBI.

waktu, bukan hanya nilainya pada satu titik. Pola crash merepresentasikan transisi dari kondisi OBI positif atau netral ke negatif ekstrem, yang secara eksplisit mengkodekan proses penarikan likuiditas. Metode threshold hanya melihat nilai skalar dan tidak mampu membedakan penurunan OBI yang terjadi secara gradual (non-crash) dari penurunan yang terjadi secara mendadak (pra-crash).

Kedua, precision sistem (0,381) menunjukkan bahwa masih terdapat 62% peringatan palsu. Hal ini disebabkan oleh fakta bahwa pola-pola SAX yang sama dapat muncul pada konteks non-crash, terutama selama periode volatilitas tinggi. Pengembangan lebih lanjut dapat mempertimbangkan penggunaan pola yang lebih panjang atau integrasi fitur tambahan seperti volume atau spread.

Ketiga, penelitian ini menggunakan proxy OBI yang diturunkan dari data volume beli/jual pada level candlestick, bukan dari data Level 2 LOB yang sesungguhnya. Keterbatasan ini perlu diakui karena data Level 2 memberikan informasi yang lebih granular mengenai distribusi likuiditas pada berbagai tingkat harga. Pengembangan sistem dengan data LOB real melalui WebSocket API berpotensi meningkatkan akurasi deteksi secara signifikan.

Terakhir, eksperimen dilakukan pada data simulasi, bukan data pasar riil. Meskipun data simulasi dirancang untuk mereplikasi karakteristik statistik pasar kripto, validasi pada data historis yang sesungguhnya diperlukan untuk mengkonfirmasi generalisasi hasil.

VI. KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan sistem deteksi dini micro-crash di pasar kripto yang mengintegrasikan algoritma SAX untuk transformasi simbolik deret waktu OBI dan algoritma KMP untuk pencarian



Fig. 5. Harga BTC/USDT (atas) dan Order Book Imbalance proxy (bawah) selama periode simulasi 7 hari. Area berwarna merah menandai zona micro-crash.

pola crash secara efisien. Hasil eksperimen menunjukkan tiga temuan utama.

Pertama, kombinasi SAX+KMP dengan konfigurasi segment size $w=10$ dan alphabet size $a=4$ mencapai skor F1 sebesar 0,552, yang mengungguli metode baseline threshold OBI dengan peningkatan 3,17 kali lipat. Seluruh 8 kejadian crash berhasil terdeteksi (recall 1,000).

Kedua, algoritma KMP secara konsisten lebih efisien dibandingkan pencarian brute force dengan rata-rata speedup 1,29x dan jumlah perbandingan karakter yang lebih sedikit, memvalidasi keunggulan kompleksitas waktu $O(n + m)$ dibandingkan $O(n \times m)$.

Ketiga, sistem menghasilkan peringatan rata-rata 11,5 menit sebelum titik terendah micro-crash, memberikan jendela waktu yang cukup untuk tindakan mitigasi risiko.

Pengembangan selanjutnya dapat mencakup: (1) penggunaan data LOB Level 2 yang sesungguhnya melalui Binance WebSocket API, (2) penerapan algoritma Aho-Corasick untuk pencarian multi-pola secara simultan, dan (3) validasi pada data historis pasar kripto yang riil.

PRANALA VIDEO YOUTUBE

<https://youtu.be/084Bq6MjyFo>

LAMPIRAN

GitHub: <https://github.com/ravasrgh/stima-micro-crash>

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada:

- 1) Tuhan Yang Maha Esa yang telah menjadi sumber segala kasih dan karunia, sumber pengetahuan, dan sumber kekuatan selama proses penulisan makalah ini. Dialah Allah Bapa, Putra dan Roh Kudus yang menyertai penulis dalam berbagai hal,
- 2) Kedua Orangtua tercinta yang telah memberikan kasih sayang, motivasi, nasehat, memenuhi kebutuhan penulis serta dukungan. Kalau bukan karena mereka pasti penulis tidak akan bisa sampai di titik ini,

- 3) Pasangan saya tercinta, Kerin, yang telah kebersamaan dan mendukung tiada henti penulis sejak pertemuan pertama, serta senantiasa memberikan semangat, motivasi, dan perhatian selama perkuliahan dan hingga penyusunan makalah ini,
- 4) Kakas Generative AI untuk bantuan perumusan ide serta penyempurnaan bahasa untuk memastikan kejelasan dan keterbacaan naskah,
- 5) Institut Teknologi Bandung, almamater saya, yang telah menjadi tempat saya menimba ilmu pengetahuan selama hampir 2 tahun,
- 6) Program Studi Teknik Informatika, dan
- 7) Dosen mata kuliah IF2211 Strategi Algoritma, Bapak Dr. Ir. Rinaldi Munir, M.T.

REFERENCES

- [1] D. E. Knuth, J. H. Morris, dan V. R. Pratt, "Fast pattern matching in strings," *SIAM Journal on Computing*, vol. 6, no. 2, hal. 323–350, 1977.
- [2] J. Lin, E. Keogh, L. Wei, dan S. Lonardi, "Experiencing SAX: a novel symbolic representation of time series," *Data Mining and Knowledge Discovery*, vol. 15, no. 2, hal. 107–144, 2007.
- [3] R. Cont, S. Stoikov, dan R. Talreja, "A stochastic model for order book dynamics," *Operations Research*, vol. 58, no. 3, hal. 549–563, 2010.
- [4] R. Cont, "Empirical properties of asset returns: stylized facts and statistical issues," *Quantitative Finance*, vol. 1, no. 2, hal. 223–236, 2001.
- [5] T. Foucault, O. Kadan, dan E. Kandel, "Limit order book as a market for liquidity," *Review of Financial Studies*, vol. 18, no. 4, hal. 1171–1217, 2005.
- [6] D. Easley, M. M. López de Prado, dan M. O'Hara, "The microstructure of the 'Flash Crash': flow toxicity, liquidity crashes, and the probability of informed trading," *The Journal of Portfolio Management*, vol. 37, no. 2, hal. 118–128, 2011.
- [7] A. S. Kyle, "Continuous auctions and insider trading," *Econometrica*, vol. 53, no. 6, hal. 1315–1336, 1985.
- [8] R. S. Boyer dan J. S. Moore, "A fast string searching algorithm," *Communications of the ACM*, vol. 20, no. 10, hal. 762–772, 1977.
- [9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, dan C. Stein, *Introduction to Algorithms*, edisi ke-3. Cambridge: MIT Press, 2009.
- [10] S. Park dan Y. Jung, "SAX-ARM: Deviant event pattern discovery from multivariate time series using symbolic aggregate approximation and association rule mining," *Expert Systems with Applications*, vol. 141, hal. 112950, 2020.
- [11] Z. Zhang, S. Zohren, dan S. Roberts, "DeepLOB: Deep Convolutional Neural Networks for Limit Order Books," *IEEE Transactions on Signal Processing*, vol. 67, no. 11, pp. 3001–3012, 2019.
- [12] A. Georgantas dan M. Doumpos, "Explainable ensemble learning for predicting stock market crises: calibration, threshold optimization, and robustness analysis," *Information*, vol. 17, no. 2, hal. 114, 2026.
- [13] N. Decker, "Crypto-assets and risks to financial stability," Financial Stability Board Working Paper, 2021.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 19 Juni 2026

