

Optimalisasi Keputusan Perdagangan Berbasis Pola dengan Algoritma Boyer-Moore

Juan Alfred Widjaya - 13522073

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): alfredwidjaya64@gmail.com

Abstrak—Perdagangan pada instrumen investasi merupakan bentuk perdagangan yang populer karena potensi keuntungan dalam jangka pendek, tetapi volatilitas pasar menantang keputusan perdagangan. Untuk mengatasi tantangan ini, diperlukan strategi dan alat bantu yang tepat. Algoritma Boyer-Moore, algoritma pencocokan string yang cepat dan efisien, dapat diterapkan untuk mendeteksi pola dalam data harga saham historis. Dengan menerapkan toleransi pada algoritma Boyer-Moore, pola-pola yang hampir cocok dapat diidentifikasi untuk prediksi pergerakan harga yang lebih akurat. Makalah ini mengkaji penggunaan algoritma Boyer-Moore dengan toleransi dalam analisis pola perubahan harga untuk memberikan rekomendasi posisi perdagangan (beli/jual). Penelitian ini mencakup pengumpulan data harga, normalisasi data menjadi persentase perubahan harga, perhitungan toleransi, pembuatan dan pencocokan pola, serta evaluasi dan akumulasi poin untuk menghasilkan keputusan perdagangan. Hasil analisis menunjukkan bahwa penggunaan algoritma ini dapat membantu investor dalam mengoptimalkan keputusan perdagangan mereka.

Kata kunci—Boyer-Moore; toleransi; analisis teknikal; perdagangan

I. PENDAHULUAN

Perdagangan saham merupakan salah satu bentuk investasi yang populer di kalangan masyarakat karena potensi keuntungan yang dapat diperoleh dalam jangka waktu yang relatif singkat. Namun, volatilitas pasar saham yang tinggi membuat keputusan perdagangan menjadi tantangan tersendiri bagi para investor. Oleh karena itu, diperlukan strategi dan alat bantu yang dapat membantu investor dalam mengambil keputusan perdagangan yang lebih tepat dan optimal. Salah satu pendekatan yang dapat digunakan untuk mengoptimalkan keputusan perdagangan adalah dengan menggunakan algoritma pencocokan pola.

Algoritma pencocokan pola memungkinkan untuk mendeteksi pola-pola tertentu dalam data harga saham historis, yang kemudian dapat digunakan untuk memprediksi pergerakan harga di masa mendatang. Salah satu algoritma pencocokan pola yang efektif dan efisien adalah algoritma Boyer-Moore. Algoritma Boyer-Moore merupakan salah satu algoritma pencocokan string yang paling cepat dan efisien. Algoritma ini menggunakan teknik "bad character rule" dan "good suffix rule" untuk mempercepat proses pencocokan pola

dalam teks. Dalam konteks perdagangan saham, algoritma Boyer-Moore dapat digunakan untuk mendeteksi pola-pola tertentu dalam data harga saham yang dapat dijadikan acuan untuk mengambil keputusan perdagangan, seperti membeli atau menjual saham.

Makalah ini bertujuan untuk mengoptimalkan keputusan perdagangan saham dengan menggunakan algoritma Boyer-Moore dalam menganalisis pola perubahan harga. Makalah ini diharapkan dapat memberikan kontribusi dalam mengembangkan strategi perdagangan yang lebih baik dan meningkatkan keuntungan investasi bagi para investor. Melalui analisis pola yang lebih akurat, investor diharapkan dapat mengurangi risiko kerugian yang disebabkan oleh fluktuasi pasar yang tidak terduga.

Penerapan algoritma Boyer-Moore tidak hanya terbatas pada perdagangan saham saja, tetapi juga dapat diterapkan pada instrumen-instrumen perdagangan lainnya seperti forex, komoditas, dan mata uang kripto. Pola harga historis sering dijadikan acuan untuk memprediksi aksi harga di masa mendatang. Dengan menemukan pola yang sama dalam data historis dan menganalisis pergerakan harga setelah pola tersebut terjadi, investor dapat memperoleh konfirmasi tambahan (*confluence*) untuk keputusan membeli atau menjual. Hal ini meningkatkan keakuratan dan kepercayaan diri dalam pengambilan keputusan perdagangan.

Selain itu, penggunaan algoritma Boyer-Moore dalam analisis pola harga saham dapat memberikan manfaat tambahan dalam hal efisiensi waktu dan sumber daya. Dengan algoritma yang lebih cepat dan efisien, investor dapat mengurangi waktu yang diperlukan untuk analisis data dan lebih fokus pada pengambilan keputusan yang strategis. Kecepatan dan efisiensi algoritma Boyer-Moore memungkinkan investor untuk merespons perubahan pasar dengan lebih cepat, yang pada gilirannya dapat meningkatkan peluang untuk meraih keuntungan.

Dengan demikian, pembaca dapat memahami pentingnya penggunaan algoritma Boyer-Moore dalam pengambilan keputusan perdagangan saham serta manfaat yang dapat diperoleh dari makalah ini. Makalah ini tidak hanya berfokus pada peningkatan keuntungan investasi tetapi juga pada pengurangan risiko dan peningkatan efisiensi dalam proses perdagangan. Diharapkan makalah ini dapat menjadi referensi

bagi investor dan peneliti lainnya dalam mengembangkan strategi perdagangan yang lebih baik di masa depan.

II. STUDI LITERATUR

A. Algoritma Boyer-Moore

Algoritma Boyer-Moore adalah salah satu algoritma pencocokan string yang paling efisien dan banyak digunakan dalam berbagai aplikasi, termasuk pengolahan teks, analisis genom, dan pencocokan pola dalam data harga saham [1]. Algoritma ini pertama kali diperkenalkan oleh Robert S. Boyer dan J. Strother Moore pada tahun 1977. Algoritma ini dirancang untuk mencari pola (*pattern*) dalam teks (*text*) dengan cara yang lebih cepat dan efisien dibandingkan dengan metode pencocokan string tradisional lainnya seperti algoritma brute force.

Algoritma Boyer-Moore memanfaatkan dua aturan utama untuk mempercepat proses pencocokan string:

1. Bad character rule (aturan karakter buruk):

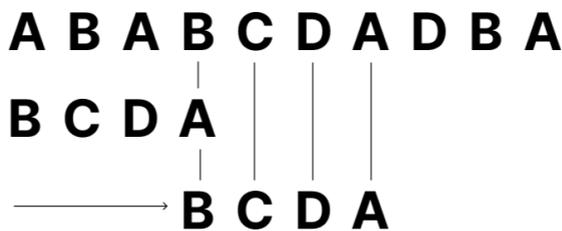
Aturan ini bekerja dengan menggeser pola ke kanan saat ditemukan karakter yang tidak cocok. Perpindahan didasarkan pada posisi terakhir dari karakter tersebut dalam pola. Jika karakter yang tidak cocok tidak ada dalam pola, pola akan digeser sepenuhnya di luar posisi karakter tersebut. Jika ada, pola digeser sehingga karakter tersebut berbaris dengan kemunculan terakhirnya dalam pola.

Contoh:

Teks: "ABABCDADBA"

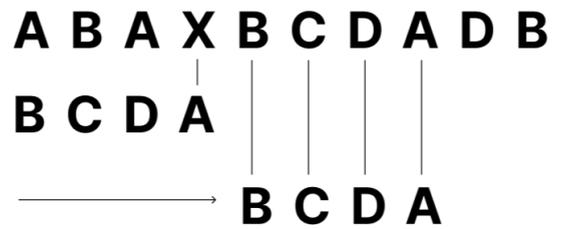
Pola: "BCDA"

Karakter yang tidak cocok: 'A' di posisi ke-4 pada pola dengan 'B' di posisi ke-4 pada text. Posisi terakhir 'B' dalam pattern adalah 1. Pola digeser sepenuhnya hingga karakter yang tidak cocok menjadi cocok pada pola. Jika tidak ditemukan karakter yang cocok maka pola digeser sepenuhnya. Berikut ialah ilustrasinya jika karakter salah dan pola digeser hingga karakter yang salah tersebut cocok dengan karakter pola,



Gambar 1. Contoh pergeseran pola kasus bad character pertama (Sumber: Arsip Pribadi)

Berikut adalah ilustrasi jika karakter salah dan pola digeser sepenuhnya karena tidak ada karakter yang salah pada sisa pola (karakter yang tidak cocok yakni 'X'),



Gambar 2. Contoh pergeseran pola kasus bad character kedua (Sumber: Arsip Pribadi)

2. Good suffix rule (aturan sufiks baik):

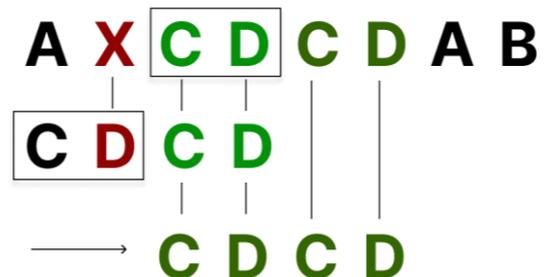
Aturan ini bekerja ketika sebagian pola telah cocok dengan teks. Ketika ditemukan ketidaksesuaian, pola digeser sehingga sufiks yang sudah cocok (*good suffix*) berbaris dengan kemunculan lain dari sufiks tersebut dalam pola. Jika tidak ada kemunculan lain, pola digeser sepenuhnya di luar posisi karakter tersebut.

Contoh:

Teks: "AXCDCDAB"

Pola: "CDCD"

Good suffix: 'CD' di posisi ke-3 dan ke-4 pada pola cocok dengan 'CD' di posisi ke-3 dan ke-4 pada teks. Pola digeser sehingga terdapat 'CD' atau substring dari 'CD' hingga cocok dengan seluruh atau sebagian dari *good suffix*. Berikut adalah ilustrasi jika terdapat potongan dari pattern yang cocok dengan *good suffix*-nya,



Gambar 3. Contoh pergeseran pola kasus good suffix (Sumber: Arsip Pribadi)

Berikut adalah langkah - langkah algoritma Boyer-Moore:

1. Preprocessing:

Bad Character Heuristic: Membuat tabel yang menyimpan posisi terakhir dari setiap karakter dalam pola.

Good Suffix Heuristic: Membuat tabel yang menyimpan panjang perpindahan berdasarkan kecocokan sufiks.

2. Searching:

- a. Memulai pencocokan dari karakter terakhir pola ke karakter pertama.

- b. Menggunakan tabel "bad character" dan "good suffix" untuk menentukan perpindahan pola.
- c. Mengulangi proses hingga seluruh teks diperiksa atau pola ditemukan.

Algoritma Boyer-Moore memiliki kelebihan dibandingkan algoritma lainnya, berikut adalah kelebihannya,

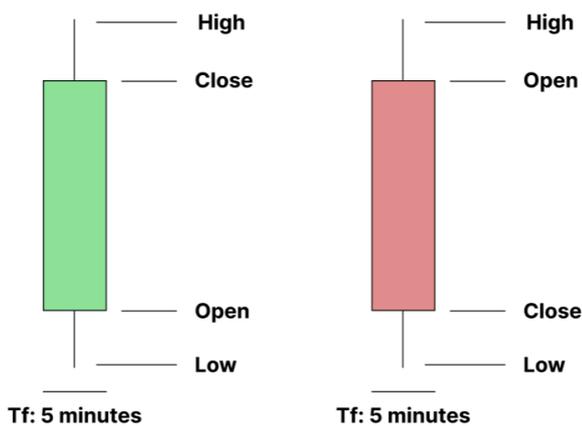
- a. Algoritma ini memungkinkan lompatan karakter yang signifikan, sehingga mengurangi jumlah perbandingan yang diperlukan untuk menemukan pola dalam teks.
- b. Algoritma ini sangat cepat dan efisien, terutama untuk teks yang panjang dan pola yang relatif pendek.
- c. Algoritma ini digunakan dalam berbagai aplikasi, termasuk editor teks, mesin pencari, dan analisis data genom [2].

B. Harga dan Struktur

Data harga suatu instrumen investasi biasanya direpresentasikan dalam empat komponen utama:

- a. *Open* (harga pembukaan): Harga pada saat perdagangan dimulai dalam periode waktu tertentu.
- b. *Close* (harga penutupan): Harga pada saat perdagangan berakhir dalam periode waktu tertentu.
- c. *High* (harga tertinggi): Harga tertinggi yang dicapai selama periode waktu tertentu.
- d. *Low* (harga terendah): Harga terendah yang dicapai selama periode waktu tertentu.

Selain komponen harga, timeframe juga merupakan aspek penting dalam analisis teknis. *Timeframe* mengacu pada interval waktu yang digunakan untuk mengamati pergerakan harga. *Timeframe* yang umum digunakan dalam analisis teknis meliputi 1 menit, 5 menit, 15 menit, 1 jam, 4 jam, harian, mingguan, dan bulanan. Data suatu harga dapat diilustrasikan dalam grafik *candlestick* [3]. Berikut merupakan contohnya,



Gambar 4. Grafik *candlestick timeframe* 5 menit (Sumber: Arsip Pribadi)

C. Analisis Teknikal

Analisis teknikal adalah disiplin perdagangan yang digunakan untuk mengevaluasi investasi dan mengidentifikasi peluang perdagangan dengan menganalisis tren statistik yang diperoleh dari aktivitas perdagangan, seperti pergerakan harga dan volume [4]. Prinsip utama analisis teknikal adalah bahwa harga mencerminkan semua informasi yang tersedia di pasar dan bahwa pola harga cenderung berulang karena psikologi pasar yang konsisten.

Charles Dow, salah satu pelopor analisis teknikal, merilis serangkaian editorial yang membahas teori analisis teknikal. Tulisannya mencakup dua asumsi dasar yang membentuk kerangka kerja untuk perdagangan analisis teknikal:

1. Pasar efisien: Nilai-nilai pasar mewakili faktor-faktor yang mempengaruhi harga sekuritas.
2. Pola dan tren: Pergerakan harga pasar yang acak tampaknya bergerak dalam pola dan tren yang dapat diidentifikasi dan cenderung berulang dari waktu ke waktu [5].

Analisis teknikal berjalan dengan tiga asumsi, yakni:

1. Pasar mendiskon segalanya: Analisis teknikal percaya bahwa semua hal mulai dari fundamental perusahaan hingga faktor pasar yang luas dan psikologi pasar sudah tercermin dalam harga saham. Pandangan ini sesuai dengan Hipotesis Pasar Efisien (EMH) yang mengasumsikan kesimpulan serupa tentang harga. Satu-satunya hal yang tersisa adalah analisis pergerakan harga, yang dipandang sebagai produk dari penawaran dan permintaan untuk saham tertentu di pasar.
2. Harga bergerak dalam tren: Analisis teknikal mengharapkan bahwa harga, bahkan dalam pergerakan pasar yang acak, akan menunjukkan tren terlepas dari kerangka waktu yang diamati. Dengan kata lain, harga saham lebih mungkin melanjutkan tren masa lalu daripada bergerak secara acak. Sebagian besar strategi perdagangan teknikal didasarkan pada asumsi ini.
3. Sejarah cenderung berulang: Analisis teknikal percaya bahwa sejarah cenderung berulang. Sifat berulang dari pergerakan harga sering dikaitkan dengan psikologi pasar, yang cenderung sangat dapat diprediksi berdasarkan emosi seperti ketakutan atau kegembiraan. Analisis teknikal menggunakan pola grafik untuk menganalisis emosi ini dan pergerakan pasar berikutnya untuk memahami tren. Meskipun banyak bentuk analisis teknikal telah digunakan selama lebih dari 100 tahun, mereka masih dianggap relevan karena menggambarkan pola dalam pergerakan harga yang sering kali berulang [6].

Beberapa alat dan teknik yang umum digunakan dalam analisis teknikal meliputi:

- a. Grafik harga: Contoh - contoh grafik harga yakni seperti grafik garis atau *line chart* yang menampilkan harga penutupan dari setiap periode waktu dalam bentuk

garis. Grafik ini sederhana dan memberikan gambaran umum tentang tren harga. Ada juga grafik batang atau bar chart yang menampilkan harga pembukaan, tertinggi, terendah, dan penutupan (OHLC) dalam bentuk batang vertikal. Grafik ini memberikan informasi lebih detail dibandingkan grafik garis. Terakhir, ada grafik *candlestick* yang menampilkan harga pembukaan, tertinggi, terendah, dan penutupan dalam bentuk batang yang menyerupai lilin. *Candlestick* sering digunakan karena visualisasinya yang mudah dibaca dan interpretasinya yang kaya akan informasi.

b. Indikator teknis:

Contoh - contoh indikator teknis yakni *moving averages* yang digunakan untuk menghaluskan data harga dan mengidentifikasi arah tren. Contohnya *Simple Moving Average* (SMA) dan *Exponential Moving Average* (EMA). Ada juga *Relative Strength Index* (RSI) mengukur kecepatan dan perubahan pergerakan harga untuk mengidentifikasi kondisi *overbought* (jenuh beli) atau *oversold* (jenuh jual). Selain itu masih banyak lagi indikator teknis yang memberikan *insights* tambahan.

c. Pola harga:

Contoh - contoh dari pola harga yakni pola kepala dan bahu (*head and shoulders*) yang merupakan pola pembalikan yang menunjukkan bahwa tren yang ada kemungkinan akan berbalik arah. Ada juga pola segitiga (*triangles*) yang merupakan pola kelanjutan yang menunjukkan periode konsolidasi sebelum tren melanjutkan arah yang ada. Contoh, segitiga simetris, *ascending*, dan *descending*. Dan masih banyak pola-pola lainnya.

Dengan menggunakan alat dan teknik ini, analisis teknikal berusaha untuk mengidentifikasi tren pasar dan pola harga yang dapat memberikan sinyal untuk mengambil keputusan perdagangan. Kombinasi dari berbagai alat dan teknik ini memungkinkan para trader untuk membuat keputusan yang lebih terinformasi dan meningkatkan peluang keberhasilan dalam perdagangan.

III. METODE

A. Menyiapkan Data

Untuk menganalisis pola, diperlukan data perubahan harga yang sudah dinormalisasi menjadi persentase dalam *timeframe* tertentu. Normalisasi ini penting karena dalam pencocokan pola, yang diperhatikan adalah perubahan harga dalam bentuk persentase, bukan nilai absolutnya. Hal ini memungkinkan pola yang sama dapat dikenali meskipun terjadi pada rentang harga yang berbeda. *Timeframe* per elemen mengacu pada interval waktu di mana harga buka (*open*) dan harga tutup (*close*) diukur. Contohnya, *timeframe* 5 menit berarti harga buka dan tutup diukur setiap 5 menit. *Timeframe* per elemen ini sangat fleksibel dan dapat disesuaikan sesuai kebutuhan analisis, mulai dari 1 menit hingga harian atau bahkan mingguan. Rentang waktu data mengacu pada keseluruhan periode waktu yang dicakup oleh data harga yang dikumpulkan. Misalnya, data harga dari 1 Januari 2020 hingga

31 Desember 2020 dengan *timeframe* 5 menit per elemen berarti setiap elemen mewakili perubahan harga dalam 5 menit, tetapi rentang waktu data mencakup seluruh tahun 2020.

Langkah-langkah untuk menyiapkan data perubahan harga menjadi persentase adalah sebagai berikut:

1. Mengumpulkan data harga:

Data harga dapat diperoleh secara real-time menggunakan API dari berbagai *exchange* atau *broker*, atau menggunakan data historis yang disediakan oleh *platform trading*. Data harga biasanya terdiri dari harga buka (*open*), harga tutup (*close*), harga tertinggi (*high*), dan harga terendah (*low*) dalam berbagai *timeframe* per elemen (misalnya 1 menit, 5 menit, 1 jam, harian, dll). Rentang waktu data dapat dipilih sesuai kebutuhan analisis. Semakin panjang rentang waktu yang digunakan, semakin besar kemungkinan untuk menangkap pola yang signifikan dan mendapatkan keputusan perdagangan yang lebih baik. Namun, penting untuk menyeimbangkan antara panjang rentang waktu dan kecepatan respons dalam perdagangan.

2. Menghitung perubahan harga dalam persentase:

Untuk setiap periode waktu (*timeframe* per elemen) yang dipilih, hitung perubahan harga dalam persentase menggunakan formula berikut:

$$\text{Perubahan} = \left(\frac{\text{Harga Tutup} - \text{Harga Buka}}{\text{Harga Buka}} \right) \times 100\%$$

Setiap elemen dalam data yang dihasilkan harus berbentuk persentase yang merepresentasikan perubahan harga relatif terhadap harga buka.

Normalisasi ini memastikan bahwa perbandingan pola hanya didasarkan pada perubahan persentase, sehingga pola yang terjadi pada rentang harga yang berbeda dapat dikenali. Normalisasi juga memungkinkan algoritma pencocokan pola untuk bekerja dengan lebih efisien.

B. Menghitung Toleransi

Dalam analisis teknikal, toleransi diperlukan karena setiap pola pergerakan harga tidak selalu bisa dihitung secara eksak. Pola pergerakan harga terjadi terus menerus, namun tidak selalu sama dalam setiap perhitungan. Oleh karena itu, diperlukan toleransi dalam pencarian pola untuk mengakomodasi variasi kecil dalam data harga. Toleransi ini dapat dihitung menggunakan standar deviasi dari perubahan harga. Berikut adalah langkah-langkah untuk menghitung toleransi:

1. Menghitung perubahan harga absolut:

Langkah pertama adalah menghitung perubahan harga absolut dari data harga yang sudah dinormalisasi menjadi persentase. Ini berarti menghitung perbedaan antara setiap harga penutupan dengan harga buka dalam bentuk persentase.

2. Menghitung standar deviasi dari perubahan harga:

Dari perubahan harga yang telah dihitung, kita kemudian menghitung nilai standar deviasi. Standar deviasi adalah ukuran statistik yang menunjukkan seberapa besar variasi atau sebaran data harga dari rata-rata.

3. Mengalikan standar deviasi dengan faktor skala:

Untuk membuat standar deviasi menjadi toleransi yang dapat digunakan dalam pencarian pola, kita mengalikan standar deviasi dengan faktor skala. Faktor ini bisa diatur sesuai keinginan untuk menentukan toleransi yang baik. Salah satu faktor yang dapat digunakan adalah 0.3, tetapi ini dapat disesuaikan berdasarkan kebutuhan analisis. Penting untuk dicatat bahwa faktor toleransi yang digunakan dapat mempengaruhi hasil analisis dan perlu dilakukan *testing* lebih lanjut untuk menemukan toleransi terbaik yang dapat menghindari sinyal palsu (*false signals*). Dengan melakukan *backtesting* menggunakan berbagai faktor, kita dapat menentukan faktor yang memberikan hasil paling akurat dan andal.

Dengan menggunakan langkah-langkah ini, kita dapat menghitung toleransi yang diperlukan untuk mengakomodasi variasi kecil dalam data harga.

C. Mencari Pola dan Analisis Keputusan

Untuk mencari pola, perlu dilakukan modifikasi terhadap algoritma Boyer-Moore dengan menggunakan toleransi. Suatu "karakter" pada pola (*pattern*) yang dalam kasus ini adalah persentase perubahan harga dianggap cocok dengan karakter pada "teks" (data persentase perubahan harga) jika nilai absolut dari selisih mereka masih lebih kecil atau sama dengan nilai toleransi. Dengan demikian, algoritma ini tidak mencocokkan karakter tetapi mencocokkan nilai dan memeriksa apakah nilai tersebut masih dalam batas toleransi dari suatu nilai lain. Meskipun begitu, jalur kerja pencarian pola masih mengikuti algoritma Boyer-Moore. Algoritma Boyer-Moore yang digunakan di sini hanya menggunakan heuristik *bad character*.

Preprocessing dimulai dengan membuat tabel *bad character* atau tabel kemunculan terakhir (*last occurrence*). Tabel ini berbeda dari tabel kemunculan biasanya karena mengakomodasi jangkauan atau rentang nilai. Menggunakan jangkauan pada tabel *bad character* memungkinkan algoritma Boyer-Moore yang dimodifikasi untuk mengakomodasi variasi kecil dalam data. Dengan cara ini, algoritma dapat lebih fleksibel dan efektif dalam mencocokkan pola yang tidak persis sama tetapi masih relevan. Berikut adalah langkah *preprocessing*-nya:

1. Membuat tabel *bad character* dengan jangkauan:

Jangkauan didapat dari nilai minimal (nilai perubahan dikurangi toleransi) hingga nilai maksimal (nilai perubahan ditambah toleransi).

2. Menghindari irisan antara dua jangkauan:

Ketika dua atau lebih jangkauan beririsan, perlu dilakukan perubahan jangkauan tersebut hingga tidak beririsan.

Contoh: jika jangkauan A memiliki titik tengah 5 dan nilai maksimal 7, sedangkan jangkauan B memiliki titik tengah 6 dan nilai minimal 4, maka kedua jangkauan tersebut beririsan. Untuk menghindari irisan, update nilai maksimal dari jangkauan A dan nilai minimal dari jangkauan B dengan mencari titik tengah dari kedua titik tengah A dan B. Maka jangkauan A akan memiliki nilai maksimal hingga titik tengah antara titik tengah A dan B, sedangkan jangkauan B akan memiliki nilai minimal dari titik tengah antara titik tengah A dan B.

Berikut adalah ilustrasi pembentukan tabel *bad character* jika karakter yang valid adalah 1, 3, 5, dan 4 dengan menggunakan jangkauan dan menghindari irisan antara dua jangkauan,

Pattern : 1 3 5 4 3 1 3

Tolerance : 1

Value	1		3		5		4	
Range (min, max)	0	2	2	4	4	6	3	5
Last Occurance	6		7		3		4	



Value	1		3		5		4	
Range (min, max)	0	2	2	3.5	4.5	6	3.5	4.5
Last Occurance	6		7		3		4	

Gambar 5. Tabel kemunculan terakhir dengan jangkauan (Sumber: Arsip Pribadi)

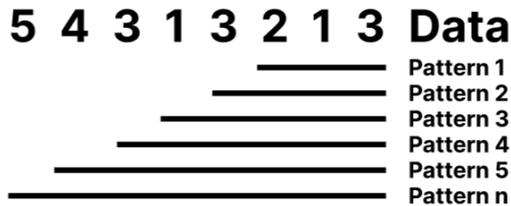
Setelah pembentukan tabel *last occurrence*, algoritma Boyer-Moore dengan *bad character rule* digunakan seperti pada umumnya. Pencarian *bad character* (nilai perubahan yang tidak cocok dengan pola) dalam tabel dilakukan berdasarkan jangkauan, bukan lagi melihat nilai perubahan (*value*) secara langsung dalam tabel.

Berikut adalah langkah - langkah selanjutnya,

1. Pembuatan pola:

Pola (*pattern*) yang dicari diambil dari data perubahan harga dengan menggunakan tiap sublist minimal tiga elemen dari belakang. *Pattern* diambil dari belakang karena *pattern* yang ingin dicari merupakan *pattern* terkini (asumsi data terkini ialah data paling belakang).

Contoh: Jika data perubahan harga adalah [1, 2, 3, 4, 5], maka pola yang dicari adalah [3, 4, 5], kemudian [2, 3, 4, 5], dan seterusnya.



Gambar 6. Ilustrasi pengambilan pola dari data (Sumber: Arsip Pribadi)

2. Pencocokan pola dengan algoritma Boyer-Moore:

Untuk setiap pola yang terbentuk, algoritma Boyer-Moore yang dimodifikasi digunakan untuk mencari keberadaan pola tersebut dalam data perubahan harga. Pola dianggap cocok jika nilai dalam data berada dalam jangkauan toleransi dari nilai dalam pola. Untuk setiap pola yang terbentuk, cari semua kemunculannya, bukan hanya satu kemunculan. Karena pola diambil dari data, maka pasti ada minimal satu pola pada data. Oleh karena itu, pencocokan pola pada data dari mana pola tersebut diambil tidak dihitung sebagai pola cocok atau tidak perlu dilakukan pencocokan.

3. Evaluasi pola:

Setelah pola ditemukan dalam data, nilai perubahan setelah pola tersebut diperiksa. Jika nilai perubahan setelah pola bernilai positif, maka diberikan poin +1. Jika negatif, diberikan poin -1.

4. Akumulasi poin:

Poin kumulatif dari tiap pola dihitung berdasarkan berapa kali pola tersebut ditemukan dalam data dan apakah perubahan setelah pola tersebut bernilai positif atau negatif.

Contoh: Jika pola A ditemukan sebanyak 10 kali, di mana 6 kali setelah pola tersebut nilai perubahannya positif, maka poin kumulatif dari pola tersebut adalah +1.

5. Keputusan berdasarkan poin kumulatif:

Semua poin kumulatif dari tiap pola yang memungkinkan diakumulasikan. Jika total poin kumulatif semua pola bernilai positif, maka keputusan adalah untuk membeli (*BUY*) karena berdasarkan pola, harga cenderung naik setelah pola

tersebut muncul. Jika total poin kumulatif semua pola bernilai negatif, maka keputusan adalah untuk menjual (*SELL*) karena berdasarkan pola, harga cenderung turun setelah pola tersebut muncul. Jika total poin kumulatif semua pola bernilai nol, tidak ada rekomendasi untuk melakukan aksi apapun (*HOLD*) karena tidak ada indikasi yang kuat mengenai arah perubahan harga.

Dengan demikian, penggunaan algoritma Boyer-Moore yang dimodifikasi dengan toleransi memungkinkan untuk analisis pola yang lebih fleksibel dan akurat dalam data perubahan harga, membantu pengambilan keputusan perdagangan yang lebih baik berdasarkan pola historis dalam data.

D. Pengujian dan Hasil

Untuk pengujian, data yang digunakan didapat dari Binance, *exchange crypto* terbesar di dunia (<https://data.binance.vision/?prefix=data/spot/daily/klines/BTCUSDT/5m/>). Data yang digunakan mencakup rentang waktu sehari pada tanggal 10 Juni 2024 dengan *timeframe* 5 menit pada *trading pair* BTC/USDT. Contoh pengujian ini diproses menggunakan script Python. Data dan script Python yang digunakan dapat ditemukan pada *repository Github* (<https://github.com/juanaw6/price-pattern-matching>).

Berikut ialah langkah pemrosesannya,

1. Pengambilan data:

Data harga diambil dari Binance dalam rentang waktu sehari pada tanggal 10 Juni 2024 dengan interval 5 menit.

2. Normalisasi data:

Data harga normalisasi menjadi persentase perubahan harga berdasarkan harga buka (*open*) dan harga tutup (*close*) dari setiap interval 5 menit.

3. Penghitungan toleransi:

Toleransi dihitung menggunakan standar deviasi dari persentase perubahan harga dengan faktor 0.3. Pada contoh ini, toleransi yang diperoleh adalah 0.02236.

4. Pembuatan dan pencocokan pola:

Pola dengan panjang minimal tiga elemen dibuat dari data perubahan harga. Pencocokan pola dilakukan menggunakan algoritma Boyer-Moore yang dimodifikasi untuk mempertimbangkan toleransi.

5. Evaluasi dan akumulasi poin:

Ditemukan bahwa hanya terdapat pola dengan panjang tiga yang ditemukan dalam data. Pola ini ditemukan empat kali. Poin kumulatif dihitung berdasarkan apakah perubahan harga setelah pola tersebut positif atau negatif. Ditemukan bahwa poin kumulatifnya adalah 2.

6. Keputusan

Karena poin kumulatifnya positif, keputusan yang dihasilkan dari analisis ini adalah untuk membeli

(BUY). Keputusan ini didasarkan pada fakta bahwa pola yang ditemukan cenderung diikuti oleh kenaikan harga.

7. Hasil pemrosesan

```
Tolerance: 0.022362923600278584
-----
Pattern [-0.028548831950077683, 0.013177931568272395, 0.02976151518190029
found at index [50, 57, 127, 159], score: 2
Score: 2
-----
Decision: BUY
```

Gambar 7. Hasil pemrosesan data dari contoh kasus (Sumber: Arsip Pribadi)

Toleransi: 0.02236

Panjang pola yang ditemukan: 3

Jumlah pola yang ditemukan: 4

Poin kumulatif: 2

IV. KESIMPULAN

Makalah ini berhasil menunjukkan bahwa algoritma Boyer-Moore yang dimodifikasi dengan toleransi dapat digunakan untuk mengoptimalkan keputusan perdagangan saham. Algoritma ini memungkinkan untuk mendeteksi pola-pola harga dalam data historis dengan lebih fleksibel dan akurat, mengingat pergerakan harga yang tidak selalu eksak. Implementasi algoritma ini pada data harga historis dari Binance menunjukkan bahwa pendekatan ini efektif dalam memberikan rekomendasi posisi trading (beli/jual) berdasarkan pola-pola harga yang ditemukan.

Dengan menggunakan algoritma Boyer-Moore yang dimodifikasi, investor dapat mengidentifikasi pola-pola yang signifikan dan memprediksi pergerakan harga di masa mendatang. Hal ini memberikan keuntungan dalam mengurangi risiko dan meningkatkan peluang keuntungan dalam perdagangan saham. Proses normalisasi data menjadi persentase perubahan harga serta penggunaan toleransi berdasarkan standar deviasi memastikan bahwa pola yang ditemukan relevan meskipun terjadi pada rentang harga yang berbeda.

V. SARAN

Beberapa saran dapat diberikan untuk pengembangan lebih lanjut dan aplikasi praktis dalam perdagangan saham dan instrumen keuangan lainnya. Pertama, disarankan untuk melakukan *backtesting* secara lebih luas dengan menggunakan data historis yang lebih panjang dan mencakup berbagai kondisi pasar, termasuk periode volatilitas tinggi dan rendah, untuk menguji keandalan dan robustitas algoritma yang digunakan. Kedua, penerapan algoritma Boyer-Moore yang dimodifikasi ini dapat diperluas ke instrumen keuangan lainnya seperti forex, komoditas, dan mata uang kripto untuk mengevaluasi efektivitasnya di berbagai pasar. Selain itu, penting untuk terus mengoptimalkan parameter toleransi agar dapat menghindari sinyal palsu dan meningkatkan akurasi keputusan perdagangan. Penggunaan *machine learning* untuk menyesuaikan parameter toleransi secara dinamis berdasarkan

kondisi pasar terkini juga bisa menjadi area penelitian yang menarik. Terakhir, integrasi algoritma ini dalam *platform trading* otomatis dapat memberikan keuntungan tambahan dengan memungkinkan eksekusi perdagangan yang lebih cepat dan tepat waktu. Dengan demikian, penelitian lebih lanjut dan penerapan praktis dari temuan ini diharapkan dapat memberikan kontribusi signifikan dalam pengambilan keputusan perdagangan yang lebih baik dan menguntungkan.

UCAPAN TERIMA KASIH

Penulis ingin mengungkapkan rasa syukur kepada Tuhan Yang Maha Esa atas berkat-Nya yang telah memungkinkan penyelesaian makalah berjudul "Optimalisasi Keputusan Perdagangan Berbasis Pola dengan Algoritma Boyer-Moore" tanpa kendala yang berarti. Penulis juga ingin mengucapkan terima kasih kepada Pak Dr. Ir. Rinaldi, M.T., yang telah menjadi dosen pengajar dalam mata kuliah IF2211 Strategi Algoritma dan memberikan bimbingan selama satu semester. Penulis juga merasa berterima kasih kepada semua pihak yang telah memberikan dukungan, meskipun tidak dapat disebutkan satu per satu, yang turut membantu dalam menyelesaikan pembuatan makalah ini.

REFERENSI

- [1] Boyer, R. S., & Moore, J. S. (1977). "A Fast String Searching Algorithm." *Communications of the ACM*, 20(10), 762-772. doi:10.1145/359842.359859
- [2] Hume, Andrew; Sunday, Daniel (November 1991). "Fast String Searching". *Software: Practice and Experience*. 21 (11): 1221-1248. doi:10.1002/spe.4380211105. S2CID 5902579
- [3] StockCharts.com. Pengantar grafik candlestick. Diakses pada 11 Juni 2024, dari https://school.stockcharts.com/doku.php?id=chart_analysis:introduction_to_candlesticks
- [4] Investopedia. Analisis teknikal. Diakses pada 11 Juni 2024, dari <https://www.investopedia.com/terms/t/technicalanalysis.asp>
- [5] CFA Institute Research Foundation. "Technical Analysis: Modern Perspectives," Halaman 1
- [6] CMT Association. "Technical Analysis: Three Premises."

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, 12 Juni 2024



Juan Alfred Widjaya
13522073