

Implementasi Knuth-Morris-Pratt (KMP) Untuk Pencarian Tempat Wisata

Sedihati Kayan Lumban Gaol

Teknik Komputer, Politeknik Unggulan Cipta Mandiri, Medan, Indonesia

Corresponding Email: kayan.marbun@gmail.com

Abstrak. Pencarian informasi tempat wisata secara cepat dan akurat merupakan kebutuhan penting dalam pengembangan sistem informasi pariwisata, terutama dalam era digital yang menuntut efisiensi dan kecepatan akses informasi. Salah satu pendekatan yang dapat digunakan untuk meningkatkan kinerja pencarian adalah dengan memanfaatkan algoritma pencocokan string. Algoritma Knuth-Morris-Pratt (KMP) merupakan salah satu algoritma yang dirancang untuk mencari pola dalam teks dengan efisiensi tinggi, khususnya dalam menangani proses pencocokan string secara cepat tanpa perlu memeriksa kembali karakter yang sudah dicocokkan sebelumnya. Penelitian ini bertujuan untuk mengimplementasikan algoritma KMP dalam sistem pencarian tempat wisata berbasis kata kunci guna meningkatkan kecepatan dan akurasi hasil pencarian. Dataset yang digunakan dalam penelitian ini mencakup informasi destinasi wisata di Indonesia, yang terdiri dari nama tempat, kategori wisata, lokasi, dan deskripsi singkat. Sistem melakukan pencocokan string berdasarkan input kata kunci dari pengguna, kemudian mencocokkannya dengan atribut pada dataset menggunakan algoritma KMP. Hasil pengujian menunjukkan bahwa algoritma ini mampu melakukan pencarian dengan waktu respons yang cepat dan memberikan hasil yang relevan dan akurat sesuai dengan kata kunci yang dimasukkan pengguna. Implementasi ini membuktikan bahwa algoritma KMP dapat diandalkan dalam pengolahan data teks, khususnya pada aplikasi sistem informasi wisata. Temuan ini diharapkan dapat menjadi landasan untuk pengembangan sistem pencarian yang lebih cerdas dan responsif di masa mendatang.

Kata Kunci: String Matching, Knuth-Morris-Pratt (KMP), Pencarian Tempat Wisata.

1. PENDAHULUAN

Pariwisata merupakan salah satu sektor penting dalam perekonomian Indonesia yang terus mengalami pertumbuhan [1]. Keanekaragaman destinasi wisata seperti wisata alam, budaya, sejarah, dan kuliner menjadikan Indonesia sebagai tujuan wisata yang menarik bagi wisatawan domestik maupun mancanegara. Seiring meningkatnya minat masyarakat untuk berwisata, kebutuhan akan informasi yang cepat, akurat, dan mudah diakses mengenai tempat-tempat wisata juga semakin tinggi. Hal ini mendorong pengembangan sistem informasi pariwisata yang mampu memberikan kemudahan dalam menemukan destinasi sesuai preferensi pengguna.

Salah satu fitur utama dalam sistem informasi pariwisata adalah kemampuan pencarian berdasarkan kata kunci. Pengguna biasanya mencari informasi destinasi berdasarkan nama tempat, jenis wisata, atau lokasi geografis. Oleh karena itu, dibutuhkan metode pencarian yang efisien untuk dapat menyajikan informasi dengan cepat tanpa mengurangi akurasi hasil pencarian. String Matching menjadi teknik yang banyak digunakan dalam menyelesaikan masalah ini, terutama ketika berhadapan dengan data teks dalam jumlah besar [2].

Algoritma Knuth-Morris-Pratt (KMP) merupakan salah satu algoritma String Matching yang dirancang untuk meningkatkan efisiensi dalam proses pencarian pola dalam teks [3]. Algoritma ini bekerja dengan memanfaatkan informasi dari pola yang sedang dicari untuk menghindari perbandingan ulang terhadap karakter yang telah dibandingkan sebelumnya [4]. Dengan strategi tersebut, proses pencarian dapat dilakukan lebih cepat dan sistematis, sehingga cocok untuk diterapkan pada sistem pencarian informasi berbasis teks, termasuk dalam konteks pariwisata.

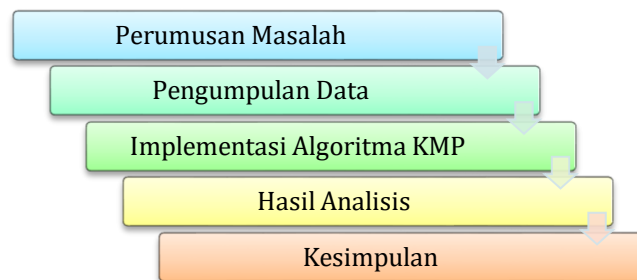
Penelitian ini bertujuan untuk mengimplementasikan algoritma KMP dalam sistem pencarian tempat wisata berbasis kata kunci. Fokus utama dari penelitian ini adalah bagaimana KMP dapat diintegrasikan ke dalam sistem informasi pariwisata untuk memproses masukan dari pengguna dan menghasilkan daftar tempat wisata yang sesuai dengan kata kunci yang diberikan. Selain itu, sistem yang dibangun juga dirancang untuk menampilkan hasil secara real-time guna meningkatkan pengalaman pengguna.

Melalui penelitian ini, diharapkan dapat diperoleh pemahaman yang lebih mendalam mengenai penerapan algoritma KMP dalam pengolahan data teks pada sistem pencarian pariwisata. Hasil yang diperoleh dari implementasi ini juga diharapkan dapat menjadi landasan bagi pengembangan sistem informasi wisata yang lebih cepat, efisien, dan responsif terhadap kebutuhan pengguna di masa depan.

2. METODOLOGI

2.1 Tahapan Penelitian

Tahapan penelitian yang dilakukan dalam studi ini dapat dijelaskan secara lebih rinci melalui ilustrasi Gambar 1 di bawah ini. Gambar tersebut menyajikan langkah-langkah sistematis yang diambil peneliti, mulai dari perumusan masalah, pengumpulan data, implementasi algoritma KMP, hasil analisis, hingga penarikan kesimpulan. Setiap tahap memiliki peran penting dalam menjamin validitas dan keakuratan hasil penelitian yang diperoleh



Gambar 1. Alur penelitian

Keterangan:

- Identifikasi Masalah**
Pada tahap ini, peneliti mengidentifikasi dan merumuskan permasalahan yang menjadi fokus utama penelitian.
- Pengumpulan Data**
Tahapan ini melibatkan proses pencarian dan pengumpulan data yang dibutuhkan untuk menguji efektivitas algoritma KMP. Data bisa berupa kumpulan teks (dokumen, artikel, atau dataset berbasis teks) dan pola (string yang ingin dicocokkan dalam teks).
- Implementasi Algoritma KMP**
Pada tahap ini, peneliti mengembangkan atau mengimplementasikan algoritma KMP untuk melakukan pencocokan pola pada teks.
- Hasil Analisis**
Setelah implementasi selesai, peneliti melakukan pengujian terhadap algoritma dan menganalisis hasil yang diperoleh.
- Kesimpulan**
Di tahap akhir ini, peneliti menyimpulkan temuan dari penelitian berdasarkan hasil analisis yang telah dilakukan. Kesimpulan mencakup keefektifan algoritma KMP dalam konteks pencocokan string, keunggulan serta batasan algoritma, dan kemungkinan pengembangan lebih lanjut.

2.2 String Matching

String Matching adalah proses pencarian kecocokan antara sebuah pola string dengan string lain yang lebih besar, dengan tujuan menemukan apakah pola tersebut muncul dalam teks dan di posisi mana [5]. Proses ini sering digunakan dalam berbagai aplikasi seperti pencarian teks di dokumen, mesin pencari, koreksi ejaan otomatis, analisis DNA, dan pendeteksian plagiarisme [6]. Ada dua jenis string matching utama, yaitu exact string matching yang mencari kecocokan pola yang persis sama menggunakan algoritma seperti Naive, Knuth-Morris-Pratt (KMP), Boyer-Moore, dan Rabin-Karp, serta approximate string matching yang mencari kemiripan meskipun terdapat perbedaan karakter, menggunakan metode seperti Levenshtein Distance atau Edit Distance [4]. Dengan demikian, string matching menjadi teknik fundamental dalam pengolahan teks dan data untuk menemukan pola secara efektif dan efisien [7].

2.3 Knuth-Morris-Pratt (KMP)

Algoritma Knuth-Morris-Pratt (KMP) adalah algoritma pencarian string yang digunakan untuk menemukan apakah suatu pola (pattern) terdapat di dalam suatu teks (text), serta di posisi mana saja pola tersebut muncul [8]. Tujuan KMP yaitu mencari pola dalam teks dengan efisiensi waktu lebih baik dibanding pencarian brute-force

[9]. KMP tidak mengulang pencocokan karakter yang sudah dibandingkan sebelumnya, berkat struktur prefix table (juga disebut LPS - Longest Prefix Suffix) [10].

3. HASIL DAN DISKUSI

3.1 Hasil

Contoh studi kasus pencarian tempat wisata yang diselesaikan menggunakan algoritma Knuth-Morris-Pratt (KMP) dalam penelitian ini menggunakan kata Danau Toba sebagai Text dan kata Toba sebagai pattern. Adapun Hasil implementasi Knuth-Morris-Pratt (KMP) untuk pencarian tempat wisata dalam penelitian dapat terlihat pada uraian di bawah ini:

a. Pecocokan String 1

Pada proses pencocokan string yang pertama ini dilakukan pencocokan pattern **T** dengan string **D** pada Text yang sejajar dengan index 0, namun ditemukan ketikacocokan, sehingga dilakukan pergeseran ke arah kanan satu langkah untuk proses pencocokan string selanjutnya.

Tabel 1. Pecocokan String 1

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern	T	o	b	a						

b. Pecocokan String 2

Pada proses pencocokan string yang kedua ini dilakukan pencocokan pattern **T** dengan string **a** pada Text yang sejajar dengan index 1, namun ditemukan ketikacocokan, sehingga dilakukan pergeseran ke arah kanan satu langkah untuk proses pencocokan string selanjutnya.

Tabel 2. Pecocokan String 2

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern		T	o	b	a					

c. Pecocokan String 3

Pada proses pencocokan string yang ketiga ini dilakukan pencocokan pattern **T** dengan string **n** pada Text yang sejajar dengan index 2, namun ditemukan ketikacocokan, sehingga dilakukan pergeseran ke arah kanan satu langkah untuk proses pencocokan string selanjutnya.

Tabel 3. Pecocokan String 3

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern			T	o	b	a				

d. Pecocokan String 4

Pada proses pencocokan string yang keempat ini dilakukan pencocokan pattern **T** dengan string **a** pada Text yang sejajar dengan index 3, namun ditemukan ketikacocokan, sehingga dilakukan pergeseran ke arah kanan satu langkah untuk proses pencocokan string selanjutnya.

Tabel 4. Pecocokan String 4

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern				T	o	b	a			

e. Pecocokan String 5

Pada proses pencocokan string yang kelima ini dilakukan pencocokan pattern **T** dengan string **u** pada Text yang sejajar dengan index 4, namun ditemukan ketikacocokan, sehingga dilakukan pergeseran ke arah kanan satu langkah untuk proses pencocokan string selanjutnya.

Tabel 5. Pecocokan String 5

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern					T	o	b	a		

f. Pecocokan String 6

Pada proses pencocokan string yang kelima ini dilakukan pencocokan pattern **T** dengan string **Spasi** pada Text yang sejajar dengan index 5, namun ditemukan ketikacocokan, sehingga dilakukan pergeseran ke arah kanan satu langkah untuk proses pencocokan string selanjutnya.

Tabel 6. Pecocokan String 6

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern						T	o	b	a	

g. Pecocokan String 7

Pada proses pencocokan string yang kelima ini dilakukan pencocokan pattern **T** dengan string **T** pada Text yang sejajar dengan index 6, pada tahap ini ditemukan kecocokan, sehingga dilakukan tidak pergeseran ke arah kanan satu langkah, namun dilakukan pencocokan string pada string selanjutnya, yaitu pattern **o** dengan string **o**.

Tabel 7. Pecocokan String 7

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern							T	o	b	a

h. Pecocokan String 8

Pada proses pencocokan string antara pattern **o** dengan string **o** pada Text yang sejajar dengan index 7 ditemukan kecocokan, sehingga dilakukan tidak pergeseran ke arah kanan satu langkah, namun dilakukan pencocokan string pada string selanjutnya, yaitu pattern **b** dengan string **b**.

Tabel 8. Pecocokan String 8

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern							T	o	b	a

i. Pecocokan String 9

Pada proses pencocokan string antara pattern **b** dengan string **b** pada Text yang sejajar dengan index 8 ditemukan kecocokan, sehingga dilakukan tidak pergeseran ke arah kanan satu langkah, namun dilakukan pencocokan string pada string selanjutnya, yaitu pattern **a** dengan string **a**.

Tabel 9. Pecocokan String 9

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern							T	o	b	a

j. Pecocokan String 10

Pada proses pencocokan string antara pattern **a** dengan string **a** pada Text yang sejajar dengan index 9 ditemukan kecocokan, sehingga pencocokan string string dihentikan dikarenakan seluruh string Pattern telah ditemukan pada Text.

Tabel 10. Pecocokan String 10

Index	0	1	2	3	4	5	6	7	8	9
Text	D	a	n	a	u		T	o	b	a
Pattern							T	o	b	a

3.2 Diskusi

Berdasarkan hasil implementasi algoritma Knuth-Morris-Pratt (KMP) terhadap pencarian kata “Toba” dalam string “Danau Toba”, dapat diketahui bahwa algoritma ini menunjukkan efisiensi dalam melakukan pencocokan string dengan cara menghindari perbandingan ulang karakter yang telah diperiksa. Proses pencocokan dilakukan sebanyak 10 langkah (iterasi), di mana hanya pada langkah ke-7 hingga ke-10 ditemukan kecocokan secara berurutan antara string pattern dan text.

Pada awal proses (langkah 1 hingga 6), ditemukan ketidakcocokan karakter antara huruf pertama pattern (‘T’) dengan karakter pada teks (‘D’, ‘a’, ‘n’, ‘a’, ‘u’, dan spasi). Setiap kali terjadi ketidakcocokan, algoritma KMP melakukan pergeseran pola ke kanan tanpa harus memeriksa kembali karakter sebelumnya, hal ini berbeda dengan metode pencarian string konvensional (seperti brute force) yang cenderung memeriksa ulang seluruh karakter jika terjadi mismatch.

Efisiensi KMP terlihat saat memasuki langkah ke-7, di mana ditemukan kecocokan karakter pertama (‘T’ pada pattern dengan ‘T’ pada teks di indeks ke-6). Setelah itu, pencocokan berlanjut ke karakter berikutnya secara berurutan (o, b, a) hingga seluruh pattern cocok sepenuhnya dengan bagian dari text pada indeks 6–9. Hal ini menunjukkan bahwa pattern “Toba” berhasil ditemukan pada posisi indeks ke-6 dalam string “Danau Toba”.

4. KESIMPULAN

Dari hasil penelitian ini dapat disimpulkan bahwa algoritma KMP bekerja dengan efektif terutama dalam kasus di mana terdapat banyak mismatch pada awal pencarian, karena algoritma ini mengandalkan prefix function untuk menghindari perbandingan ulang yang tidak perlu. Meskipun dalam studi kasus ini panjang teks dan pattern relatif pendek, pendekatan KMP tetap menunjukkan manfaatnya dalam hal efisiensi pencarian.

Selain itu, penerapan algoritma ini dalam konteks pencarian tempat wisata dapat memberikan keuntungan dalam sistem pencarian berbasis teks pada aplikasi pariwisata digital. Misalnya, ketika pengguna mengetik sebagian dari nama objek wisata, sistem dapat dengan cepat menemukan lokasi yang dimaksud meskipun terdapat banyak data tempat wisata dalam database.

Dengan demikian, implementasi algoritma KMP tidak hanya relevan dalam konteks akademik, tetapi juga memiliki aplikasi nyata dalam pengembangan sistem informasi pencarian yang cepat dan efisien, khususnya pada sistem rekomendasi atau pencarian tempat wisata berbasis teks. Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan bahwa algoritma Raita mampu memberikan pencarian yang cepat dan efisien dengan jumlah perbandingan karakter yang minimal. Ini menjadi sangat relevan dalam konteks sistem informasi manajemen penjualan sparepart mobil, di mana data produk dapat sangat banyak dan membutuhkan metode pencarian yang efisien.

Referensi

- [1] H. Fadilla, “Pengembangan Sektor Pariwisata untuk Meningkatkan Pendapatan Daerah di Indonesia,” *Benefit J. Bussiness, Econ. Financ.*, vol. 2, no. 1, pp. 36–43, 2024, doi: 10.37985/benefit.v2i1.375.
- [2] A. Azhar, N. Marbun, S. Aripin, and E. Buulolo, “Implementasi Algoritma Horspool Pada Aplikasi Istilah Fashion,” *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 3, no. 1, pp. 549–551, 2019, doi: 10.30865/komik.v3i1.1641.
- [3] C. B. Setiawan, “Penerapan dan Perbandingan Algoritma String Matching pada Aplikasi UUD 1945 dan UU di Indonesia,” *Jsika*, vol. 4, pp. 1–7, 2019.
- [4] Andreansyah and N. Ratama, “Implementasi Algoritma Knuth Morris Pratt Dalam Pencarian Kamus Bahasa Isyarat,” *JORAPI J. Res. Publ. Innov.*, vol. 1, no. 1, pp. 120–125, 2023.
- [5] Y. Napitupulu, “Perancangan Aplikasi Kode Etik Profesi Dengan Menerapkan Algoritma Raita,” *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 2, no. 2, pp. 80–83, 2021, [Online]. Available: <http://djournals.com/klik/article/view/222>
- [6] S. Maesyaroh, G. P. Suwanto, F. I. Komputer, U. Kuningan, P. Kecamatan, and K. Kabupaten, “pencocokan teks; mahasiswa; algoritma horspool; algoritma raita;,” vol. 9, pp. 1–9, 2023.
- [7] F. Yani Dalimunthe, “Perancangan Aplikasi Katalog Museum Dengan Menerapkan Algoritma Zhutakaoka (Studi Kasus : Museum Negeri Provinsi Sumatra Utara,” *Bull. Inf. Technol.*, vol. 3, no. 1, pp. 20–26, Mar. 2022, doi: 10.47065/bit.v3i1.184.
- [8] A. Qur’ania, Triastinurmiatiningsih, and E. Candra, “Kamus Digital Tanaman Obat Menggunakan Algoritme Knuth Morris Pratt Berbasis Mobile,” *KOMPUTASI J. Ilm. Ilmu Komput. dan Mat.*, vol. 19, no. 1, pp. 354–361, 2022.

- [9] N. Novianti, R. C. G. I. Kembaren, D. M. Br Bangun, and N. Marbun, "Implementasi Algoritma Knuth Morris Pratt Pada Aplikasi Sinopsis Film Bioskop Berbasis Web," *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 3, no. 1, pp. 398–401, 2019, doi: 10.30865/komik.v3i1.1619.
- [10] R. Angelina, P. Hutabarat, J. S. Hutapea, D. Marlina, and M. Lubis, "Penerapan Algoritma String Matching Dalam Pencocokan Data String," *J. Tek. Inform.*, vol. 15, no. 2, 2023.