

Penerapan *String Matching* Menggunakan Algoritma Boyer-Moore Pada Pengembangan Sistem Pencarian Buku Online

Yusuf Faqih¹, Yuri Rahmanto^{2,*}, Ahmad Ari Aldino¹, Budi Waluyo³

¹Fakultas Teknik dan Ilmu Komputer, Informatika, Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia

²Fakultas Teknik dan Ilmu Komputer, Teknologi Informasi, Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia

³Sistem Informasi, Sekolah Tinggi Teknologi Informasi NIIT, Jakarta Selatan, Indonesia

Email: ¹yfaqih5@gmail.com, ^{2,*}yurirahmanto@teknokrat.ac.id, ³ahmadarialdino@teknokrat.ac.id, ⁴mbudiku7@gmail.com

Email Penulis Korespondensi: yurirahmanto@teknokrat.ac.id

Abstrak—Algoritma Bayer Moore dapat di terapkan dalam pencarian buku online dengan tujuan mempermudah siswa sekolah dasar hingga siswa menengah atas untuk mencari buku yang ingin di pelajari atau mengetahui mata pelajaran yang sedang banyak di cari. Dalam pengembangan pencarian buku online dengan metode algoritma Bayer Moore dalam proses pengembangannya dibutuhkan beberapa tahapan penelitian yang dilakukan secara terencana, teratur, dan sistematis. Tahapan penelitian ini adalah pengembangan dari kerangka penelitian yang akan menentukan keberhasilan dari suatu sistem. Tahapan dari metode Research and Development (R&D) Pengumpulan data yang digunakan dalam penelitian ini merupakan kumpulan data diperoleh dari hasil crawling dengan tahapan menggunakan Bahasa pemograman php. Data Crawling tersebut nanti akan menjadi sebuah serachbox tentang sebuah buku yang akan bisa menggunakan link Shopee, Tokopedia, dan Gramedia. Prosedur sistematis yang digunakan untuk mengumpulkan data yaitu dengan metode penelitian kuantitatif. Pelaksanaan metode penelitian kuantitatif fokus pada penggunaan angka, tabel, data, *code*, *rules* dan untuk menampilkan hasil data yang diperoleh dan akan digunakan untuk bahan penganalisisan data terhadap Website pencarian buku menggunakan algoritma Boyer-Moore. Berdasarkan hasil penelitian dan pengujian hasil dari website pencarian buku dengan menggunakan metode Boyer-Moore di php yang diperoleh menunjukkan bahwa program tersebut dinyatakan layak untuk digunakan, karena aturan asosiatif yang dibentuk dan nilai pencarian yang dihasilkan sama dengan output program yang dikembangkan.

Kata Kunci: Algoritma Boyer-Moore; *Online Shop*; *Website* Pencarian Buku; *String Matching*

Abstract—The Bayer Moore algorithm can be applied in online book searches with the aim of making it easier for elementary school students to high school students to find books they want to study or find out which subjects are being sought. In developing an online book search using the Bayer Moore algorithm, several stages of research are needed in a planned, regular, and systematic way. This research stage is the development of a research framework that will determine the success of a system. The stages of the Research and Development (R&D) method. The data collection used in this study is a collection of data obtained from crawling results using the PHP programming language. The Crawling data will later become a searchbox about a book that will be able to use Shopee, Tokopedia, and Gramedia links. The systematic procedure used to collect data is quantitative research methods. The implementation of quantitative research methods focuses on the use of numbers, tables, data, codes, rules and to display the results of the data obtained and will be used for data analysis materials on book search websites using the Boyer-Moore algorithm. Based on the results of research and testing results from the book search website using the Boyer-Moore method in php, the results obtained indicate that the program is declared feasible to use, because the associative rules formed and the resulting search value are the same as the output of the developed program.

Keywords: Algorithm Boyer-Moore; Online Shop; Book Search Website; String Matching

1. PENDAHULUAN

Seiring perkembangan teknologi informasi, perpustakaan dituntut untuk memberikan informasi yang lebih dinamis, cepat, dan akurat. Hal ini dilakukan agar keeksistensian perpustakaan di era teknologi informasi ini dapat dipertahankan di tengah maraknya penyedia informasi lain yang lebih canggih sebagai kompetitor perpustakaan dengan pencarian buku online. Dalam hal ini tidak hanya berfokus pada segi pengembangan koleksi dari berbagai ilmu yang didapatkan dari buku Pendidikan sekolah, akan tetapi juga memperhatikan kemudahan akses bagi pengguna dan pencari referensi dalam penyelesaian tugas dan ulangan sekalipun. Salah satu alat yang dapat memudahkan pemustaka pada perpustakaan adalah katalog. Melalui katalog buku dan pencarian atau search buku online, pengguna dapat melakukan akses ke koleksi suatu buku Sehingga dapat menginformasikan keadaan sumber daya koleksi yang dimilikinya kepada pengguna, melalui katalognya [1].

Katalog pencarian buku juga tidak luput dari objek revolusi, hal ini dibuktikan dengan munculnya inovasi pada media katalog dari masa ke masa Inovasi terhadap katalog perpustakaan ditujukan untuk memberi kemudahan kepada pengguna perpustakaan dalam menemukan bahan pustaka yang diinginkannya dari perpustakaan. Agar pencarian katalog buku dapat lebih cepat diperlukan suatu algoritme yang tepat agar dapat membantu dan mempermudah pencarian buku online. Algoritma Horspool merupakan penyederhanaan dari algoritma Boyer-Moore. Perbedaan antara keduanya adalah pada metode penggeseran patternnya. Jika Boyer-Moore menggunakan dua metode praproses bad character shift dan good shufix shift, akan tetapi Horspool hanya menggunakan satu metode praproses yaitu bad character shift. Kompleksitas rata-rata algoritma ini sama dengan Boyer-Moore $O(n)$, sedangkan untuk metode praproses nya adalah $O(m+\sigma)$ [2].

Ada banyak algoritma dalam pencarian salah satunya adalah algoritme Boyer-Moore adalah algoritme pencarian yang lebih efisien daripada algoritme binary dan Sequential Search. Hal ini dikarenakan algoritme ini tidak perlu menjelajah setiap elemen dari tabel. Kerugiannya adalah algoritme ini hanya bisa digunakan pada tabel yang elemennya sudah terurut baik menaik (ascending) maupun menurun (descending). Dengan adanya sistem informasi katalog buku dan skripsi di perpustakaan UMB diharapkan mampu membantu pemustaka dalam mencari referensi. Sistem informasi

Katalog buku dan skripsi di perpustakaan UMB ini semoga dapat memberikan informasi yang akurat dan efisien, dalam hal pencarian data buku dan skripsi yang diperlukan [3].

Algoritma Boyer-Moore banyak diterapkan terhadap proses pencarian. Fasilitas Find/Search pada berbagai aplikasi pengolah teks, web browser, dan aplikasi lainnya mungkin saja telah memanfaatkan algoritma Boyer-Moore dalam pencarian tersebut, karena algoritma Boyer-Moore paling banyak diimplementasikan dalam berbagai aplikasi untuk fasilitas pencarian teksnya walaupun sebenarnya pencarian yang dilakukan dapat berupa pencarian terhadap teks, dokumen, atau juga file yaitu dengan melakukan pencocokan string antara kata kunci dengan objek yang dicari atau terhadap kata dengan mengacu pada sebuah database leksikal. Algoritma Boyer-Moore akan menyimpan informasi pergeseran untuk melakukan pencarian string. Karakteristik utama dari algoritma Boyer-Moore adalah algoritma ini melakukan pencocokan string mulai dari kanan ke kiri [4]. Dengan karakteristik tersebut, ketidakcocokan saat terjadi perbandingan string akan membuat pergerakan pattern melompat lebih jauh untuk menghindari perbandingan karakter pada string yang diperkirakan gagal [5].

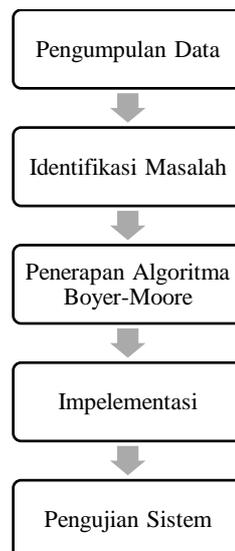
Pencarian tarian tradisional ini dirancang dengan menggunakan algoritma Turbo Boyer-Moore, dimana algoritma ini sebagai solusi dalam pencarian tarian tradisional, karena banyaknya macam-macam tarian di wilayah Indonesia sehingga sangat sulit untuk menampilkan penyajian tarian tradisional itu menurut dengan abjadnya. Tidak hanya itu, dengan terbatasnya ukuran layar smartphone yang dimiliki, maka sangat diperlukan string matching sebagai suatu proses dalam pencarian. Dimana dalam algoritma string matching penyajian tarian tradisional akan lebih sedikit karena adanya proses pengecekan berdasarkan *pattern* [6].

Penelitian ini bertujuan untuk mengembangkan sistem pencarian buku secara online dengan menerapkan *string matching* menggunakan algoritma Boyer-Moore. Sistem ini dibangun agar memudahkan seseorang yang mencari buku yang diinginkan sehingga langsung menampilkan hasil pencarian buku yang diambil dari internet berdasarkan kata kunci yang dimasukkan oleh pengguna. Algoritma Boyer-Moore melakukan pencocokan *string* mulai dari kanan ke kiri, sehingga *string* yang akan dicari akan relevan.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Untuk melakukan pencarian buku online dengan metode algoritma Bayer-Moore dalam proses pengembangannya dibutuhkan beberapa tahapan penelitian yang dilakukan secara terencana, teratur, dan sistematis. Tahapan penelitian ini adalah pengembangan dari kerangka penelitian yang akan menentukan keberhasilan dari suatu sistem [7]. Tahapan penelitian yang dilakukan mengacu pada metode *Research and Development* (R&D). Gambar 1 berikut ini adalah tahapan penelitian yang dilakukan.



Gambar 1. Tahapan Penelitian

2.1.1 Pengumpulan Data

Teknik Pengumpulan data pada penelitian ini diantaranya sebagai berikut:

a) Eksperimen

Pengumpulan data penelitian dilakukan dengan melakukan percobaan terhadap suatu hal, yaitu pengambilan sampel objek (dataset) yang dilanjutkan dengan observasi, pelatihan, dan pencatatan hasil yang berkaitan dengan penelitian.

b) Studi Literatur

Studi literatur merupakan data bersumber dari literatur, jurnal, internet, buku, dan karya ilmiah lainnya dari penelitian sebelumnya mengenai objek yang diteliti dan parameter yang diteliti.

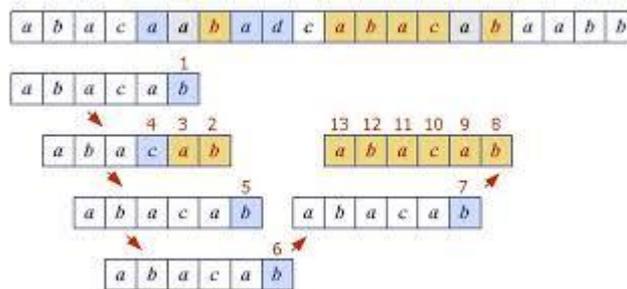
2.1.2 Identifikasi Masalah

Pada tahap ini pengembang akan melakukan pengumpulan kebutuhan dari sistem yang dibangun. Untuk menetapkan kebutuhan maka perlu melakukan identifikasi masalah yang akan diselesaikan [8]. Berdasarkan permasalahan yang telah didapatkan maka selanjutnya dilakukan analisa kebutuhan sistem. Analisa kebutuhan yang dilakukan berdasarkan analisa kebutuhan fungsional. Kebutuhan fungsional merupakan pernyataan mengenai fitur yang dibutuhkan, sehingga sistem dapat memberikan layanan yang sesuai dengan keinginan pengguna [9]. Masalah utama pada penelitian ini adalah bagaimana membangun sistem yang dapat mempermudah dalam pencarian buku sehingga pengguna dapat menemukan buku yang diinginkan dengan cepat dan sesuai.

2.1.3 Penerapan Algoritma Bayer Moore

String matching atau mencocokkan *string* merupakan pendekatan pemecahan masalah untuk mendapatkan pola berdasarkan susunan dari karakter *string* yang terdapat pada bagian dari isi teks atau *string* lainnya [10]. Sedangkan Boyer-Moore merupakan pendekatan yang mencocokkan *string* dengan cara melihat dari dari kanan kemudian menuju ke kiri dengan melakukan pelacakan karakter pada teks. *Good-suffix shift* dan *bad-character shift* merupakan dua fungsi *shift* yang digunakan Boyer-Moore dalam mengambil langkah dalam pencarian setelah terdapat ketidaksamaan antara karakter *pattern* dan karakter teks yang dilakukan pencocokan [5]. Algoritma Boyer-Moore diciptakan oleh R.M Boyer dan J.S Moore. Algoritma ini terkenal karena banyak diterapkan pada algoritma pencocokan untuk banyak string (*multi pattern*) [11]. Algoritma Boyer-Moore adalah salah satu dari pencarian string yang tepat dan terkenal algoritma pencocokan yang digunakan dalam pencocokan pola tunggal dan dianggap sangat cepat dalam kinerjanya [12]. Dengan menggunakan algoritma ini, secara rata-rata proses pencarian akan menjadi lebih cepat jika dibandingkan dengan algoritma lainnya. alasan melakukan pencocokan dari kanan (posisi terakhir *pattern* yang dicari) [13].

Algoritma Boyer-Moore memiliki loncatan karakter yang besar sehingga mempercepat pencarian *pattern* karena dengan hanya memeriksasedikit karakter, dapat langsung diketahui bahwa *string* yang dicari tidak ditemukan dan dapat digeser ke posisi berikutnya. Algoritma Boyer-Moore telah dibuktikan sebagai salah satu algoritma yang paling efisien dalam aplikasi pencarian string dengan menggunakan natural language (bukan *binary language*) [14]. Algoritma ini telah sering diimplementasikan untuk fungsi “Search” dan “Substitute” pada text editor. Padasarnya cara kerja algoritma ini mirip dengan algoritma Knuth-MorrisPrat (KMP) dimana kedua algoritma ini akan melakukan lompatan pengecekan dalam proses pencarian string [15]. Namun berbeda dengan algoritma KMP, algoritma Boyer-Moore ini melakukan perbandingan *pattern* mulai dari kanan ke kiri.



Gambar 2. Cara Kerja Algoritma Bayer-Moore

Berdasarkan Gambar 1, bahwa ide dari Algoritma Boyer-Moore menggunakan pengetahuan tentang pencarian teks untuk meningkatkan kecepatan pencarian secara signifikan. Algoritma Boyer-Moore menggunakan suatu langkah sebelum proses untuk membuat *occurrence-function* dan *shifting-function* yang digunakan untuk melakukan bad character heuristics dan *good suffix heuristics* masing-masing [16]. Boyer-Moore menggunakan dua heuristik untuk memutuskan seberapa jauh melompat: *badcharacter heuristic*, juga sering disebut *occurrence heuristic*, dan heuristik yang *good-character*, juga disebut *match heuristic*. Informasi untuk heuristik setiap dipertahankan dalam sebuah array yang dibangun pada awal operasi yang cocok.

2.1.4 Implementasi Sistem

Proses selanjutnya yaitu *coding*, dimana tahapan ini akan mengimplementasikan tahapan sebelumnya menjadi sebuah sistem dengan menggunakan bahasa pemrograman tertentu [17][18]. Pada penelitian ini sistem dibangun berbasis web, sehingga pada pengkodeannya dengan bahasa pemrograman PHP menggunakan *text editor* Notepad++ dan untuk penyimpanan datanya menggunakan MySQL.

2.1.5 Pengujian Sistem

Setelah semua proses dilakukan, tahap selanjutnya adalah melakukan pengujian. Tahap ini memiliki tujuan agar dapat dipastikan bahwa sistem yang dikembangkan mampu bekerja dengan baik dan bebas dari kesalahan [19]. Penelitian ini

menggunakan teknik uji yaitu *black-box testing*. Metode ini akan melakukan uji sistem berdasarkan dari fungsionalitas sistem, sehingga perangkat lunak yang diuji fungsi-fungsinya telah berjalan dengan baik [20][17].

3. HASIL DAN PEMBAHASAN

Pengumpulan data yang digunakan dalam penelitian ini merupakan kumpulan data diperoleh dari hasil *crawling* dengan tahapan menggunakan Bahasa pemrograman php. Data *crawling* tersebut nanti akan menjadi sebuah *serachbox* tentang sebuah buku yang akan bisa menggunakan link shopee,tokoped,gramedia. Prosedur sistematis yang digunakan untuk mengumpulkan data yaitu dengan metode penelitian kuantitatif. Pelaksanaan metode penelitian kuantitatif fokus pada penggunaan angka, tabel, data, code, rules dan untuk menampilkan hasil data yang diperoleh dan akan digunakan untuk bahan penganalisisan data terhadap website pencarian buku menggunakan algoritma Boyer-Moore.

Untuk mengembangkan sistem pencarian buku, maka dibutuhkan perangkat yang sesuai dengan *requirement* yang telah ditetapkan. Kebutuhan berupa perangkat keras maupun perangkat lunak perlu ditentukan, agar sistem yang dikembangkan dapat berjalan dengan baik. Berikut kebutuhan yang diperlukan dalam penelitian ini :

- a. Hardware :
 1. Komputer/laptop dengan Processor Intel® core™ i7-6600 CPU @ 260 GHz(4 CPUs), -2.8GHz
 2. Ram 8 gb
 3. Gpu Nvidia Intel® HD Graphics 520
- b. Software:
 1. Oprasi Sistem Windows 10.
 2. Php 3.7.
 3. Xampp.
 4. Visual Studio Code
 5. Cmd
 6. Shopee,Tokopedia,Gramedia

Untuk menerapkan *string matching* dengan menggunakan algoritma Boyer-Moore maka diawali dengan mempelajari cara kerja *sistring matching*. *String matching* merupakan proses pencarian sebuah *string* yang terdiri dari *pattern* terhadap karakter pada teks yang dicari. Dalam pencocokan *string* terdapat dua pendekatan yaitu *exact string matching* dan *inexact string matching*. Dalam penelitian ini pendekatan yang digunakan *exact string matching*, karena pendekatan ini digunakan untuk pencocokan *string* secara tepat dengan melibatkan struktur karakter teks yang dicocokkan memiliki jumlah urutan dan struktur *tring* yang sama. Salah satu pendekatan *exact string matching* yang efisien dapat melakukan perbandingan *pattern* adalah Boyer-Moore. Ide dasar dari algoritma Boyer-Moore adalah dengan melakukan pencocokan dan memindai pola *string* dari kanan ke kiri mulai dari karakter paling kanan dari *string* yang akan dicari. Cara kerja Algoritma Boyer-Moore dengan melakukan *matching* atau pencocokan antara *pattern* dan *text* dengan memperhatikan urutan yang dimulai dari karakter paling kanan terlebih dahulu kemudian ke kiri, dimana pencocokan dilakukan dengan dasar nilai dari *occurence heuristic* (OH) dan *match heuristic* (MH) [12][3]. Sebagai contoh penerpan algoritma Boyer-Moore dalam menentukan tingkat kemiripan judul Skripsi/TA dengan menggunakan kata “PANTUN” sebagai *string pattern* dan kata “SASTRA PANTUN” sebagai *string text*, berikut tahapan penyelesaiannya pada Taeb1 1.

Tabel 1. Occurence heuristic (OH) dan match heuristic (MH)

Pattern	P	A	N	T	U	N
OH	5	4	3	2	1	0
MH	6	6	6	6	6	1

Pada tahap pertama, untuk pencocokan *string* karakter “N” pada *pattern* tidak cocok dengan karakter “A” pada *text*, sehingga akan dilakukan pergeseran nilai berdasarkan tabel *Occurence Heuristic* (OH) dan *Match Heuristic* (MH). Pada tabel OH dan MH karakter “A” tidak ada, maka jumlah pergeseran dilakukan sebanyak 6 langkah seperti pada Tabel 2.

Tabel 2. Pencocokan antara text dan pattern

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Text	S	A	S	T	R	A		P	A	N	T	U	N
Pattern	P	A	N	T	U	N							

Tahap kedua, karakter “N” setelah mengalami pergeseran ternyata tidak cocok dengan karakter “U” pada *text*, maka selanjutnya adalah melakukan pergeseran berdasarkan nilai OH. Pada tabel OH untuk karakter “N” memiliki nilai 1, untuk itu selanjutnya akan dilakukan pergeseran sebanyak 1 langkah seperti pada Tabel 3.

Tabel 3. Pencocokan antara text dan pattern dengan 6 langkah pergeseran

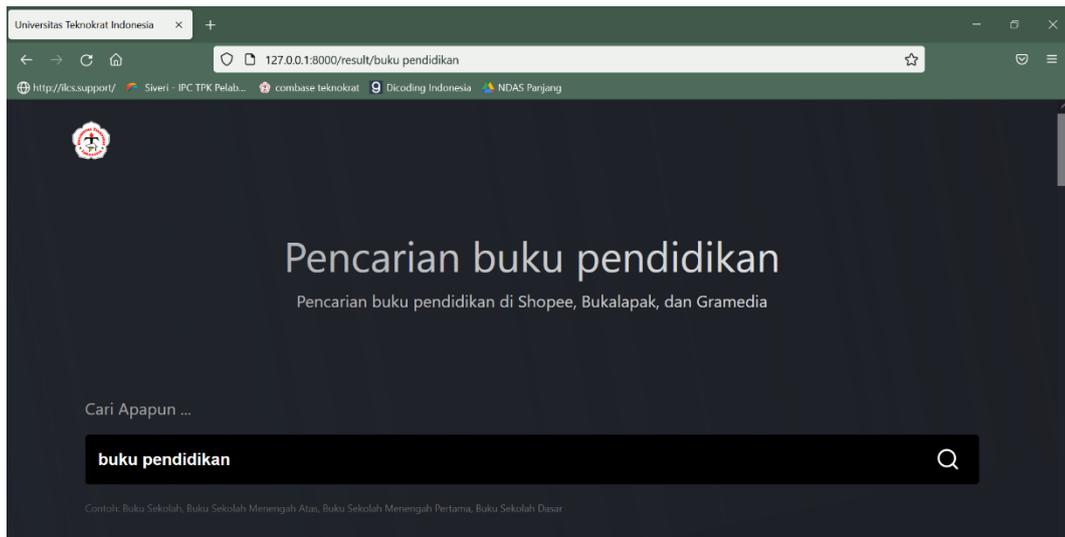
Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Text	S	A	S	T	R	A		P	A	N	T	U	N
Pattern							P	A	N	T	U	N	

Sampai dengan tahap ini *string* pada *text* paling kanan yaitu karakter “N” telah sama atau cocok dengan *string* pada *pattern* yaitu karakter “N”, begitu pula dengan *string* pada *text* selanjutnya yaitu “P”, “A”, “N”, “T”, “U” telah sama dengan *pattern*. Untuk itu pergeseran berakhir pada tahap ketiga karena telah menemukan kesamaan, seperti pada Tabel 4.

Tabel 4. Pencocokan antara text dan pattern dengan 1 langkah pegeseran

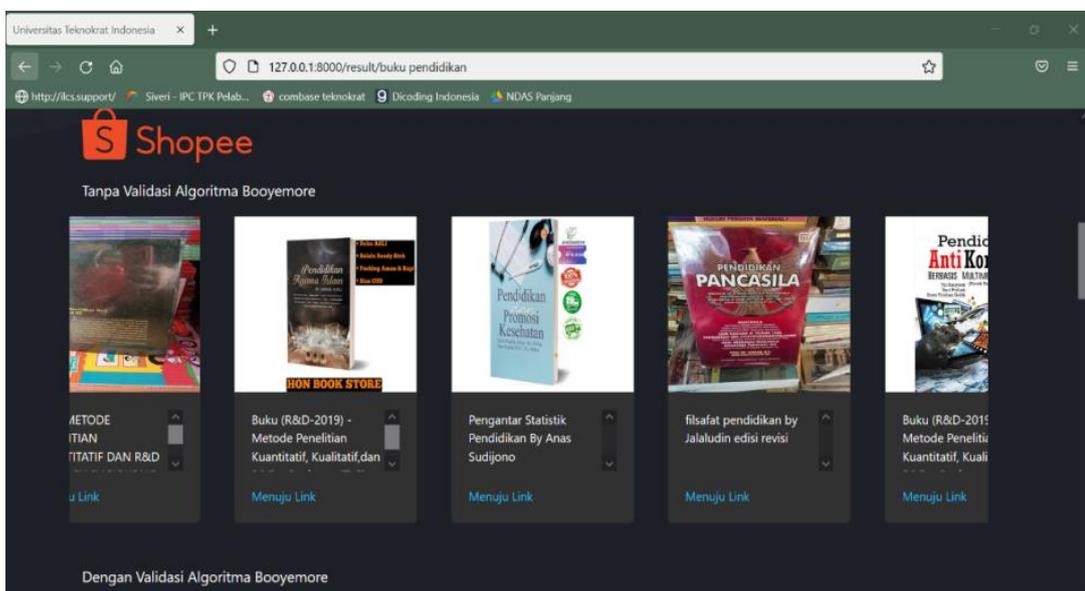
<i>Index</i>	0	1	2	3	4	5	6	7	8	9	10	11	12
<i>Text</i>	S	A	S	T	R	A		P	A	N	T	U	N
<i>Pattern</i>							P	A	N	T	U	N	

Proses selanjutnya yaitu *coding*, dimana tahapan ini akan mengimplementasikan tahapan sebelumnya menjadi sebuah sistem dengan menggunakan bahasa pemrograman tertentu. Pada penelitian ini sistem dibangun berbasis web, sehingga pada pengkodeannya dengan bahasa pemrograman PHP menggunakan *text editor* Notepad++ dan untuk penyimpanan datanya menggunakan MySQL. Langkah awal penggunaan sistem yaitu dengan memasukkan kata kunci buku Pendidikan lalu algoritma boyer more akan bekerja otomatis untuk menentukan kata dan Menyusun kata tercepat. Tampilan awal website pencarian buku dapat dilihat pada gambar 3.



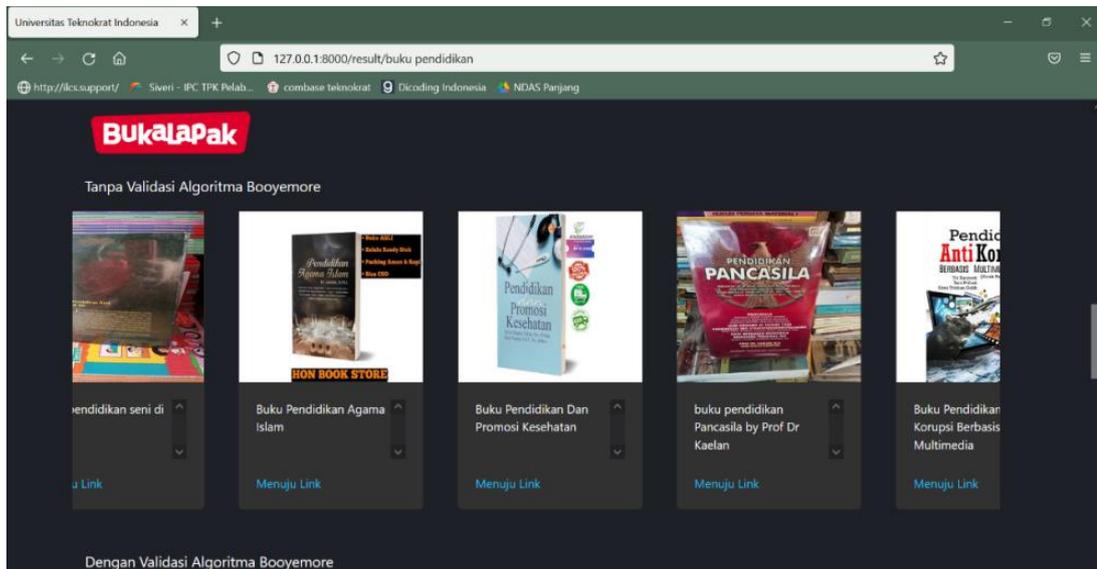
Gambar 3. Tampilan awal Website Pencarian buku

Kemudian, ketika sudah muncul akan ada validasi dari algoritma boyermore tersebut maka jika muncul pasti tervalidasi dengan valid disini buku sudah bisa melink dari shopee dan bisa langsung mengklik atau tab pada sub menu “menuju link”. Tampilan setelah validasi dapat dilihat pada Gambar 4.



Gambar 4. Tampilan Setelah Validasi

Selanjutnya, pengguna akan ditampilkan link menuju hasil pencarian, sebagai contoh adalah olshop bukalapak sama seperti yang diatas jika muncul pasti sudah tervalidasi dengan algoritma boyermore dan disini buku sudah bisa melink dari bukalapak dan langsung bisa klik atau tab pada sub menu “menuju link”. Tampilan bukalapak dapat dilihat pada Gambar 5.



Gambar 5. Hasil Pencarian

Setelah sistem berhasil dibangun, selanjutnya adalah melakukan pengujian sistem agar sistem bebas dari kesalahan. Uji yang digunakan adalah dengan *black-box testing*, yang melakukan uji berdasarkan fungsional sistem. Untuk hasil uji *black-box testing* dapat dilihat pada tabel 5 berikut ini.

Tabel 5. Hasil Uji *Black-Box Testing*

No	Fitur Uji	Hasil yang Diharapkan	Kesimpulan
1	Menu Utama	Menu utama dapat tampil.	Valid
2	Menu Pencarian	Pengguna dapat mengakses menu pencarian dan dapat melakukan pencarian dengan memasukkan kata kunci.	Valid
3	Menampilkan Hasil Pencarian	Sistem dapat menampilkan hasil pencarian	Valid
4	Menampilkan Link Hasil Pencarian	Sistem menampilkan link hasil pencarian sehingga user dapat menuju link hasil pencarian	Valid
5	Melihat Rekomendasi Hasil Pencarian	Sistem menampilkan urutan hasil pencarian	Valid

Berdasarkan Tabel 5 memperlihatkan bahwa sistem telah berjalan sebagaimana mestinya, karena berdasarkan *black-box testing* semua fitur uji memiliki nilai valid.

4. KESIMPULAN

Berdasarkan hasil penelitian dan pengujian hasil dari website pencarian buku dengan menggunakan metode Boyer-Moore di php yang diperoleh menunjukkan bahwa program tersebut dinyatakan layak untuk digunakan, karena aturan asosiatif yang dibentuk dan nilai pencarian yang dihasilkan sama dengan output program yang dikembangkan. Informasi keterkaitan antar item yang dihasilkan merupakan aturan asosiatif yang memiliki index di shopee, tokepedia, dan gramedia. Hasil pengolahan Website pencarian buku dengan menggunakan algoritma Boyer-Moore dapat digunakan untuk mencari buku untuk dalam mendapatkan link pembelian market place pada platform shopee, tokepedia, dan Gramedia yang ada sehingga dapat dijadikan sebagai penunjang informasi dalam pengambilan keputusan strategi penjualan baik untuk tujuan meningkatkan pengguna atau pelajar bisa mengetahui buku yang dicari dengan adanya website tersebut dapat memudahkan pengguna/pelajar dalam hal mencari buku dan membeli buku. Waktu pemrosesan pencarian data buku dengan algoritma Boyer-Moore sangat bergantung pada nilai metode pencarian itu sendiri.

REFERENCES

- [1] D. Kartini, R. Y. Tanjung, and I. Lasmana, “Penerapan Algoritma Boyer Moore Horspool Pada Pencarian Katalog Buku,” in

- Seminar Nasional Sains & Teknologi Informasi (SENSASI)*, 2019, pp. 341–344.
- [2] M. Aulia, “Penerapan Algoritma Boyer Moore Untuk Pencarian Data Member Pada PT. Boenk Cosmetic Manufacture Berbasis Desktop,” *J. Sist. Komput. dan Inform.*, vol. 1, no. 3, p. 235, 2020.
- [3] I. J. Sinaga, R. C. G. I. Kembaren, D. M. Br Bangun, and N. Marbun, “Penerapan Algoritma Boyer Moore Pada Aplikasi Filofofer Berbasis Website,” *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 3, no. 1, pp. 436–438, 2019.
- [4] B. Harpad and S. Salmon, “Penerapan Algoritma Boyer-Moore Dalam Perpustakaan Virtual SMK Muhammadiyah 1 Samarinda Sebagai Pendukung Pembelajaran Daring,” *J. Ilm. Matrik*, vol. 23, no. 2, pp. 182–188, 2021.
- [5] R. I. Borman and A. Pratama, “Penerapan String Matching Dengan Algoritma Boyer Moore Pada Aplikasi Font Italic Untuk Deteksi Kata Asing,” *J. Teknoinfo*, vol. 10, no. 2, pp. 1–5, 2016.
- [6] S. Sulastri, “Implementasi Algoritma Turbo Boyer Moore Pada Aplikasi Tarian Tradisional Berbasis Android,” *Bull. Comput. Sci. Res.*, vol. 1, no. 1, pp. 18–23, 2020.
- [7] N. Y. Arifin *et al.*, *Analisa Perancangan Sistem Informasi*. Batam: Cendikia Mulia Mandiri, 2021.
- [8] M. Melinda, R. I. Borman, and E. R. Susanto, “Rancang Bangun Sistem Informasi Publik Berbasis Web (Studi Kasus : Desa Durian Kecamatan Padang Cermin Kabupaten Pesawaran),” *J. Tekno Kompak*, vol. 11, no. 1, p. 1, 2018.
- [9] R. Napianto, Y. Rahmanto, R. I. Borman, and O. Lestari, “Software Development Sistem Pakar Penyakit Kanker Pada Rongga Mulut Berbasis Web,” in *Dalam Seminar Nasional Pengaplikasian Telematika (Sinaptika)*, 2019.
- [10] I. Mulyawati, R. T. Subagio, and D. Martha, “Implementasi Metode String Matching Untuk Aplikasi Pengarsipan Dokumen (Studi Kasus : SMPN 3 Sumber Kab. Cirebon),” *J. Digit*, vol. 7, no. 1, pp. 50–61, 2017.
- [11] M. H. Rifqo and A. Andilala, “Implementasi Algoritma Boyer-Moore pada Aplikasi Kamus Istilah Komputer Berbasis Android,” *JSI J. Sist. Inf.*, vol. 12, no. 2, pp. 2149–2160, 2020.
- [12] A. Fau, Mesran, and G. L. Ginting, “Analisa Perbandingan Boyer Moore Dan Knuth Morris Pratt Dalam Pencarian Judul Buku Menerapkan Metode Perbandingan Eksponensial (Studi Kasus : Perpustakaan STMIK Budi Darma),” *J. Times (Technology Informatics Comput. Syst.)*, vol. 6, no. 1, pp. 12–22, 2017.
- [13] I. Ahmad, R. I. Borman, G. G. Caksana, and J. Fakhrurozi, “Implementasi String Matching Dengan Algoritma Boyer-Moore Untuk Menentukan Tingkat Kemiripan Pada Pengajuan Judul Skripsi/Ta Mahasiswa (Studi Kasus : Universitas XYZ),” *SINTECH (Science Inf. Technol. J.)*, vol. 4, no. 1, pp. 53–58, 2021.
- [14] D. Surianto, D. Triyanto, and U. Ristian, “Penerapan Algoritma Boyer Moore dan Metode N-Gram Pada Aplikasi penyunting Naskah Teks Bahasa Indonesia Berbasis Web,” *Coding J. Komput. dan Apl.*, vol. 08, no. 03, pp. 50–60, 2020.
- [15] F. T. Waruwu and R. Mandala, “Perbandingan Algoritma Knuth Morris Pratt dan Boyer Moore Dalam Pencocokan String Pada Aplikasi Kamus Bahasa Nias,” *J. Ilm. INFOTEK*, vol. 1, no. 1, pp. 36–43, 2016.
- [16] E. Rahmanita, “Pencarian String Menggunakan Algoritma Boyer Moore Pada Dokumen,” *J. Ilm. NERO*, vol. 1, no. 1, pp. 15–26, 2014.
- [17] I. Ahmad, R. I. Borman, J. Fakhrurozi, and G. G. Caksana, “Software Development Dengan Extreme Programming (XP) Pada Aplikasi Deteksi Kemiripan Judul Skripsi Berbasis Android,” *J. Invotek Polbeng - Seri Inform.*, vol. 5, no. 2, pp. 297–307, 2020.
- [18] I. Ahmad, A. T. Prastowo, E. Suwarni, and R. I. Borman, “Pengembangan Aplikasi Online Delivery Sebagai Upaya Untuk Membantu Peningkatan Pendapatan,” *JMM (Jurnal Masy. Mandiri)*, vol. 5, no. 6, pp. 4–12, 2021.
- [19] R. Napianto, Y. Rahmanto, R. I. Borman, O. Lestari, and N. Nugroho, “Dhempster-Shafer Implementation in Overcoming Uncertainty in the Inference Engine for Diagnosing Oral Cavity Cancer,” *CSRID (Computer Sci. Res. Its Dev. Journal)*, vol. 13, no. 1, pp. 45–53, 2018.
- [20] W. N. Cholifah, Y. Yulianingsih, and S. M. Sagita, “Pengujian Black Box Testing Pada Aplikasi Action & Strategy Berbasis Android Dengan Teknologi Phonegap,” *J. String*, vol. 3, no. 2, pp. 206–210, 2018.