



Implementasi Algoritma Approximate String Matching Pada Aplikasi Pengarsipan Dokumen Ekspor Barang

Yola Annisa Sihotang

Teknik Informatika, Sistem Informasi, Universitas Budidarma, Medan, Indonesia

Annisayola@gmail.com

Abstrak- Proposal Skripsi ini mengangkat judul “Implementasi Algoritma Approximate String Matching Pada Aplikasi Pengarsipan Dokumen Ekspor Barang (Studi Kasus :PT. Chitra Kalpika Mas Kab.Deli Serdang)”. Proposal ini di buat berdasarkan pengamatan yang dilakukan di PT. Chitra Kalpika Mas dengan melihat system pengarsipan dokumen ekspor yang masih dilakukan secara manual, penulis membuat rancangan sistem yang baru untuk memperbaiki sistem yang lama. Permasalahan yang dikaji dalam penelitian ini adalah bagaimana pencarian arsip dokumen ekspor serta kendala yang dihadapi oleh PT. Chitra Kalpika Mas. Dan kendala apa yang telah ditempuh untuk mengatasi kendala yang ada. Oleh karena itu, dibutuhkan sebuah program aplikasi yang dapat membantu pegawai untuk menyelesaikan tugasnya agar lebih mudah, salah satunya yaitu melakukan proses pencarian data arsip dokumen ekspor. Berdasarkan hasil penelitian maka dapat disimpulkan bahwa pencarian arsip dokumen ekspor pada PT. Chitra Kalpika Mas masih dilakukan secara manual. Oleh karena itu diperlukan rancangan aplikasi baru dengan menggunakan program computer sesuai dengan kebutuhan pegawai kantor sehingga penggunaan lebih mudah diperoleh dengan cepat dan akurat. Dengan adanya rancangan aplikasi arsip dokumen ekspor dapat mempermudah pengolahan laporan, penambahan, pengubahan, dan penghapusan data serta dapat memberikan informasi kapan saja di perlukan.

Kata kunci : Arsip Dokumen Ekspor, *Approximate String Matching*.

Abstract- This thesis proposal raises the title "Implementation of Approximate String Matching Algorithm in Export Document Filing Application (Case Study: PT. Chitra Kalpika Mas Kab.Deli Serdang)". This proposal is made based on observations made at PT. Chitra Kalpika Mas by looking at the filing system. Export documents are still done manually, the author makes a new system design to improve the old system. The problems studied in this study are how to search for export document archives and the obstacles faced by PT. Chitra Kalpika Mas. And what obstacles have been taken to overcome the existing obstacles. Therefore, we need an application program that can help employees complete their tasks to make it easier, one of which is to search for export document archive data. Based on the research results, it can be concluded that the search for export document archives at PT. Chitra Kalpika Mas is still done manually. Therefore, it is necessary to design a new application using a computer program in accordance with the needs of office employees so that use is easier to obtain quickly and accurately. With the design of an export document archive application, it can facilitate report processing, additions, changes, and deletions of data and can provide information whenever needed.

Keywords: Export Document Archives, *Approximate String Matching*,

1.PENDAHULUAN

Dalam era globalisasi dan perdagangan internasional yang semakin berkembang, pengelolaan dokumen ekspor barang menjadi semakin kompleks dan penting. Setiap transaksi ekspor melibatkan berbagai dokumen seperti invoice, packing list, bill of lading, sertifikat asal, dan banyak lagi. Efisiensi dalam pengarsipan dan pencarian dokumen tersebut sangat krusial untuk memastikan kelancaran proses ekspor dan kepatuhan terhadap regulasi internasional. Namun, dalam praktiknya, pengarsipan manual sering kali menghadapi berbagai tantangan, seperti kesalahan penulisan, inkonsistensi format dokumen, dan keragaman bahasa. Kesalahan ini dapat mengakibatkan kesulitan dalam pencarian dokumen yang akurat dan tepat waktu. Di sinilah teknologi informasi dan algoritma pencocokan string (string matching) berperan penting untuk meningkatkan efisiensi dan akurasi dalam pengarsipan dokumen.

Salah satu masalah utama dalam pengarsipan dokumen adalah pencarian dokumen yang tidak selalu akurat karena perbedaan format atau kesalahan penulisan. Algoritma approximate string matching (pencocokan string mendekati) menawarkan solusi untuk masalah ini dengan memungkinkan pencarian yang toleran terhadap kesalahan (fuzzy search). Algoritma ini dapat mengidentifikasi dan mencocokkan dokumen yang memiliki kemiripan meskipun terdapat perbedaan kecil dalam teks. Informasi sangat penting untuk mendukung kelangsungan perkembangan dan kemajuan suatu perusahaan. Teknologi dan informasi merupakan dua hal yang tidak dapat dipisahkan satu sama lain. Perkembangan teknologi yang pesat dalam perkembangan perangkat keras dan perangkat lunak serta teknologi komunikasi menjadikan sistem informasi menjadi alternatif bagi suatu perusahaan untuk menunjang pengolahan data yang baik. Apabila pengolahan data tidak teratur dan tidak terkoordinasi dengan baik akan mengakibatkan sulitnya mengetahui data dan informasi secara tepat, cepat dan akurat. Dalam hal ini PT. Chitra Kalpika Mas merupakan salah satu perusahaan bisnis yang bergerak di bidang pengiriman barang ke luar negeri (ekspor). PT. Chitra Kalpika Mas memproduksi dan mengirim



barang ke berbagai negara tiap minggunya. Setiap pengiriman ekspor disertai dengan beberapa dokumen ekspor seperti laporan ekspor, *invoice*, *packing list*, *certificate of origin*, *bill of lading* atau *air way bill*, pemberitahuan ekspor barang (PEB), nota pelayanan ekspor (NPE), dan dokumen pelengkap lainnya sesuai dengan ketentuan permintaan pembeli. Semakin banyaknya dokumen yang dikelola sehingga menyebabkan beberapa kesalahan, seperti lupa mengirim dokumen, lupa memberitahu sales jika dokumen sudah selesai serta kesulitannya mendapatkan informasi ekspor dan juga arsip dokumen ekspor. Untuk mengatasi permasalahan diatas penulis membangun aplikasi pengarsipan dokumen ekspor PT. Chitra Kalpika Mas dengan mengimplementasikan algoritma string matching *Approximate* pada pengarsipan dokumen ekspor. Algoritma *Approximate* string matching ialah pencocokan string dengan dasar kemiripan dari segi penulisannya (jumlah karakter, susunan karakter dalam arsip dokumen).

2.METODOLOGI PENELITIAN

2.1 String Matching

Menurut Riyanto Sarno, dkk *String Matching* adalah proses pencarian semua kemunculan query yang selanjutnya disebut *pattern* ke dalam string yang lebih panjang [teks]. *Pattern* di lambangkan dengan $x=x[0..m-1]$ dan panjangnya adalah m . Teks dilambangkan dengan $y=y[0..n-1]$ dan panjangnya adalah n . Kedua string terdiri dari sekumpulan karakter yang disebut alfabet yang dilambangkan dengan Σ dan mempunyai ukuran σ . *String matching* dibagi menjadi dua, yakni *exact matching* dan *heuristic* atau *statistical matching*[3]. *String matching* menurut Ernawati *matching* adalah sebuah algoritme yang digunakan dalam pencocokkan suatu pola kata tertentu terhadap suatu kalimat atau teks panjang. Pencocokan string merupakan bagian penting dari sebuah proses pencarian string [*string searching*] dalam sebuah dokumen. Hasil dari pencarian sebuah string dalam dokumen tergantung dari teknik atau cara pencocokan string yang digunakan[4].

2.2 Algoritma Approximate String Matching

Algoritma *Approximate String Matching* adalah pencocokan *string* dengan dasar kemiripan dari segi penulisannya [jumlah karakter, susunan karakter dalam dokumen]. Tingkat kemiripan ditentukan dengan jauh tidaknya beda penulisan dua buah *string* yang dibandingkan tersebut. Metode ini dapat digunakan untuk pencarian kata tidak baku karena dapat mengidentifikasi *string* yang sama dan yang memiliki kemiripan penulisan. Beberapa algoritma yang dapat digunakan untuk pencarian *string* dalam *approximate string matching* yaitu: *Hamming Distance*, *Levenshtein Distance*, *Damerau Levenshtein Distance*, *Jaro Winkler Distance* [9] [1]. Pada penelitian ini menerapkan salah satu turunan dari *Algoritma Approximate String Matching* yaitu *Algoritma Levenshtein Distance*. *Levenshtein Distance* digunakan untuk mengukur nilai kesamaan atau kemiripan antara dua buah kata [*string*].

3.HASIL DAN PEMBAHASAN

3.1 Analisa

Analisa dilakukan untuk mengkaji atau membahas dari sistem yang sedang berjalan atau sistem lama untuk mempermudah mendefinisikan permasalahan – permasalahan yang timbul di dalam sistem yang sedang berjalan di PT. Chitra Kalpika Mas. Pengertian analisa sistem adalah proses pemecahan sistem menjadi beberapa subsistem yang lingkungannya lebih kecil dengan maksud agar lebih mudah dalam mengidentifikasi permasalahan, hambatan dan kesempatan yang ada dalam sistem untuk mengetahui kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan. Analisa Sistem Arsip Dokumen Ekspor yang sedang berjalan di PT. Chitra Kalpika Mas masih berjalan secara manual, belum efisien dan akurat, yaitu dengan cara mencatat barang keluar yang telah di kirim pada laporan rangkap 2 dan disimpan dalam sebuah map, yang apabila data tersebut dibutuhkan akan mencari dan membongkarnya kembali, sehingga membutuhkan waktu yang kurang efisien dan agak lama

3.2 Implementasi Algoritma Approximate String

Algoritma *Approximate String* merupakan pengembangan dari algoritma pencarian string sebelumnya, yaitu algoritma *Brute Force*. Algoritma *Brute Force* merupakan algoritma dasar yang paling sederhana dalam menyelesaikan persoalan pencocokan string yang melakukan pencarian pada setiap posisi di dalam teks antara 0 dan $n-m$, dimana n adalah panjang teks/banyaknya nama file yang tersimpan di komputer dan m adalah panjang karakter dari suatu pattern (kata yang akan dicari). Tahap pencocokan *string* menggunakan algoritma *Approximate String* adaah sebagai berikut :

1. Tentukan nilai S dan *pattern string* S

A	C	A	N	A	C	A	R	A	K	A	B	K
---	---	---	---	---	---	---	---	---	---	---	---	---

Pattern

C	A	R	A	K	A
---	---	---	---	---	---

2. Bandingkan *pattern* [1] dan *string* [1]

A	C	A	N	A	C	A	R	A	K	A	B	K
---	---	---	---	---	---	---	---	---	---	---	---	---



C	A	R	A	K	A
---	---	---	---	---	---

Pattern [1] tidak cocok dengan *string* [1]. Maka *pattern* akan bergeser satu posisi ke kanan.

3. Bandingkan *pattern* [1] dan *string* [2]

A	C	A	N	A	C	A	R	A	K	A	B	K
---	---	---	---	---	---	---	---	---	---	---	---	---



C	A	R	A	K	A
---	---	---	---	---	---

4. Bandingkan *pattern* [2] dan *string* [3]

A	C	A	N	A	C	A	R	A	K	A	B	K
---	---	---	---	---	---	---	---	---	---	---	---	---



C	A	R	A	K	A
---	---	---	---	---	---

5. Bandingkan *pattern* [3] dan *string* [4]

A	C	A	N	A	C	A	R	A	K	A	B	K
---	---	---	---	---	---	---	---	---	---	---	---	---



C	A	R	A	K	A
---	---	---	---	---	---

Pattern [3] tidak cocok dengan *String* [4]. Karena tidak ada kecocokan, maka algoritma *Approximate String* akan menggunakan informasi yang telah disimpan pada langkah dua dan tiga untuk menentukan pergeseran selanjutnya. Sehingga pencocokan yang dilakukan bukanlah antara *pattern* dengan *string* [3] seperti jika menggunakan algoritma *Brute Force*, namun antara *pattern* [1] dengan *string* [4].

6. Bandingkan *pattern* [3] dan *string* [4]

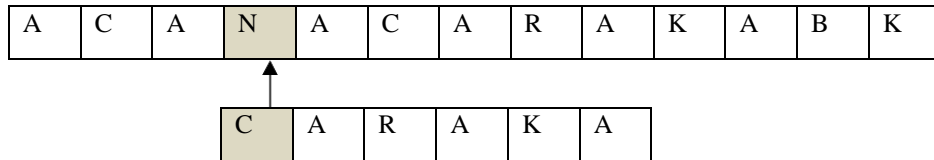
A	C	A	N	A	C	A	R	A	K	A	B	K
---	---	---	---	---	---	---	---	---	---	---	---	---



C	A	R	A	K	A
---	---	---	---	---	---

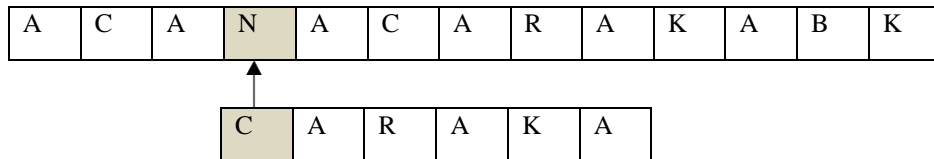
Pattern [3] tidak cocok dengan *String* [4]. Karena tidak ada kecocokan, maka algoritma *Approximate String* akan menggunakan informasi yang telah disimpan pada langkah dua dan tiga untuk menentukan pergeseran selanjutnya. Sehingga pencocokan yang dilakukan bukanlah antara *pattern*, dengan *string* [3] seperti jika menggunakan algoritma *Brute Force*, namun antara *pattern* [1] dengan *string* [4] sehingga dapat mempercepat proses pencocokan *string*.

7. Bandingkan *pattern* [1] dan *string* [4]



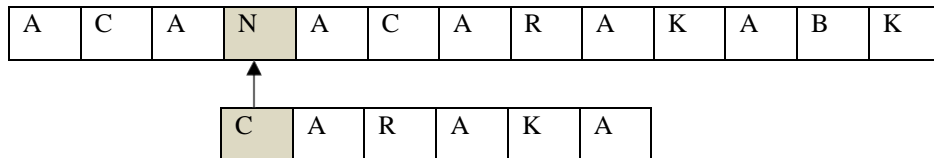
Pattern [1] tidak cocok dengan *string* [4]. Maka *pattern* akan bergeser satu posisi ke kanan.

8. Bandingkan *pattern* [1] dan *String* [5]



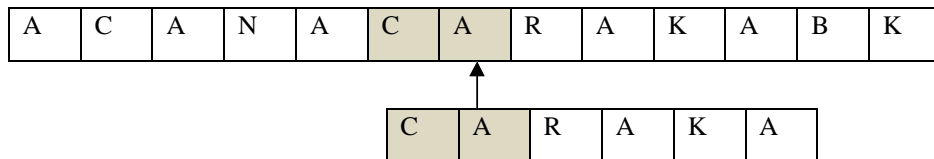
Pattern [1] tidak cocok dengan *string* [5]. Maka *pattern* akan bergeser satu posisi ke kanan.

9. Bandingkan *pattern* [1] dan *string* [5]

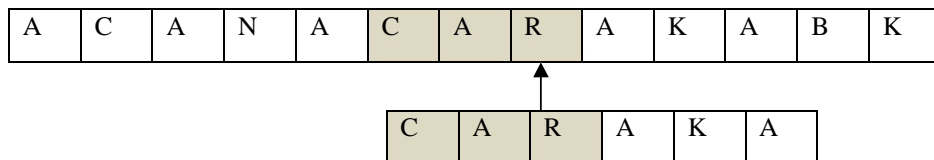


Pattern [1] cocok dengan *string* [6]. Karena ada kecocokan maka algoritma *Approximate String* akan menyimpan informasi ini, dan *pattern* tidak akan melakukan pergeseran dan melanjutkan pencocokan *pattern* [2] dengan *string* [7].

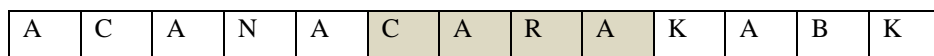
10. Bandingkan *pattern* [2] dan *string* [7]

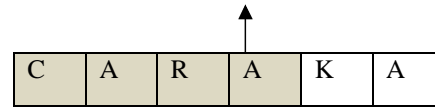


11. Bandingkan *pattern* [3] dan *string* [8]

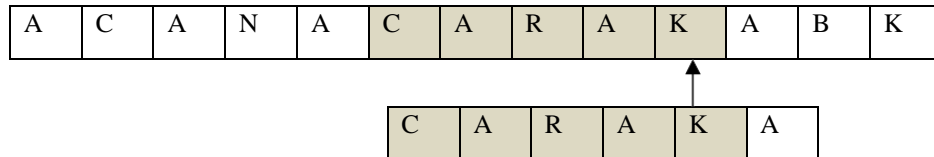


12. Bandingkan *pattern* [4] dan *string* [9]

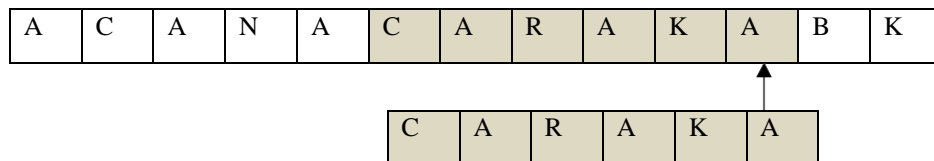




13. Bandingkan *pattern* [5] dan *string* [10]



14. Bandingkan *pattern* [6] dan *string* [11]



4.KESIMPULAN

Berdasarkan hasil analisa yang telah penulis uraikan pada bab sebelumnya ,maka dapat disimpulkan bahwa kendala-kendala yang dihadapi dalam mengarsipkan dokumen ialah Aplikasi penyimpanan arsip yang ada pada bagian dokumen ekspor belum baik karena *system* kearsipannya masih manual, Kurangnya peralatan dan perlengkapan yang digunakan seperti tidak adanya tanda batas. Hal ini berdampak pada saat penemuan arsip dan input data yang membutuhkan waktu relatif lama, Aplikasi arsip dokumen ini dirancang dengan menggunakan *Visual Basic 2008* dengan tampilan yang lebih efisien dan mudah untuk digunakan dan sebagai saran Pemakai atau *user* harus memiliki penguasaan dan kemampuan dalam bidang *computer* baik hardware maupun *software* secara baik yang nantinya akan dibutuhkan dalam arsip dokumen, Agar penerapan aplikasi mengarsip dokumen ekspor dapat berjalan dengan baik, pemeliharaan dan penyempurnaan *system* harus terus dilakukan sesuai dengan perkembangan dan kebijaksanaan perusahaan agar *system* dapat berfungsi sesuai dengan apa yang diinginkan serta Sebaiknya gunakan password atau kata sandi dalam computer tersebut, dan hanya administrasi kantor tertentu saja yang boleh tahu tentang kata sandi itu, untuk menjaga keamanan data dari orang-orang yang berniat tidak baik.

REFERENCES

- [1] L. Wardani, "Perancangan Aplikasi Pencarian Makna Motif Batik Menggunakan Metode Galil Seiferas," *JURIKOM (Jurnal Riset Komputer)*, vol. 7, no. 2, pp. 264-268, 2020.
- [2] R. Sarno, Y. Anistiyasari and F. Rahimi, *Semantic Search*, Yogyakarta: Andi, 2012.
- [3] Ernawati, A. Johar and S. Setiawan, "IMPLEMENTASI METODE STRING MATCHING UNTUK PENCARIAN BERITA UTAMA PADA PORTAL BERITA BERBASIS ANDROID (STUDI KASUS: HARIAN RAKYAT BENGKULU)," *Jurnal Pseudocode*, vol. VI, no. 1, pp. 77-82, 2019.
- [4] W. Astuti, "ANALISIS STRING MATCHING PADA JUDUL SKRIPSI DENGAN ALGORITMA KNUTH-MORRIS PRATT (KMP)," *ILKOM Jurnal Ilmiah*, vol. 9, no. 2, pp. 167-172, 2017.
- [5] P. Sulistyorini, "Pemodelan Visual dengan Menggunakan UML dan Rational Rose," *Jurnal Teknologi Informasi DINAMIK*, vol. XIV, no. 1, pp. 23-29, 2009.
- [6] R. A. Sukamto and M. Shalahuddin, *Rekayasa Perangkat Lunak (Arsitektur Dan Berorientasi Objek)*, Bandung: Bandung : Informatika, 2014.
- [7] Hendrayudi, *VB 2008 untuk Berbagai Keperluan Pemrograman*, Jakarta: PT. Elex Media



Buletin Ilmiah Informatika Teknologi

Vol 2, No 3, Mei, Hal. 109 - 114

ISSN 2962-0945 (media online)

DOI 10.58369/biit.v2i3.57

<https://ejurnal.amikstiekomsu.ac.id/index.php/BIIT>

Komputindo, 2009.

- [8] P. A. d. S. (PAS), Membuat Aplikasi Client Server dengan Visual Basic 2008, Yogyakarta: Penerbit : Andi, 2010.
- [9] Wikipedia, "Wikipedia Ensiklopedia Bebas," 11 Juni 2020. [Online]. Available: <https://id.wikipedia.org/wiki/Legenda>. [Accessed 05 Juli 2020].