

# Pemberian Saran Untuk Mencari Kosakata Menggunakan Algoritma Brute Force Dan Steganografi

Indra Gunawan<sup>1)</sup>, Sumarno<sup>2)</sup>, Heru Satria Tambunan<sup>3)</sup>, Abdi Rahim Damanik<sup>4)</sup>

<sup>1,2</sup>Program Studi Teknik Informatika, <sup>3,4</sup>Program Studi Sistem Informasi

<sup>1,2,3,4</sup>STIKOM Tunas Bangsa Pematangsiantar

<sup>1</sup>indra@amiktunasbangsa.ac.id, <sup>2</sup>sumarno@amiktunasbangsa.ac.id, <sup>3</sup>heru@amiktunasbangsa.ac.id,

<sup>4</sup>abdirahimdamanik@stikomtb.ac.id

## Abstrak

Melakukan pencarian data didalam sistem komputer dengan menggunakan alat pencarian data dapat memudahkan user untuk mencari data-data dan informasi. Agar dapat lebih mempermudah user dalam melakukan pencarian data-data dan informasi, maka sangatlah perlu memodifikasi dan menerapkan beberapa algoritma atau metode untuk memberikan fitur tambahan dengan memberikan saran kosa kata dengan melakukan pencocokan string yang dapat menjadi alternatif untuk pencarian data-data dan informasi. Algoritma brute force memiliki cara kerja melakukan penelusuran string dengan melakukan kombinasi pencarian secara acak dari tiap-tiap string sampai menemukan string yang cocok dan yang akan dipergunakan untuk melakukan pencarian data, sedangkan algoritma steganografi memiliki cara kerja melakukan penyisipan objek, baik itu berupa karakter maupun gambar. Memadukan kedua algoritma tersebut serta mengkombinasikan masing-masing kelebihan kinerjanya, sehingga dapat memberikan saran dalam proses pencarian data-data dan informasi melalui pencarian data. Hasilnya adalah algoritma brute force akan melakukan pencarian data-data dengan melakukan pergeseran dari tiap-tiap string, sehingga dalam melakukan pencarian data akan membutuhkan waktu yang singkat sedangkan algoritma steganografi melakukan penyimpanan data-data yang telah dilakukan pencocokan dalam proses pencarian data-data yang bertujuan untuk melakukan pencarian data-data berikutnya.

**Kata kunci:** Algoritma, Brute Force, Steganografi, Kosa Kata

## Abstract

Searching data in the computer system by using the data search tool can make it easier for users to search for data and information. In order to make it easier for users to search for data and information, it is necessary to modify and implement several algorithms or methods to provide additional features by providing vocabulary suggestions by matching strings that can be an alternative to the search for data and information. Brute force algorithm has a way of performing string searches by randomly searching from each string to find a matching string and which will be used to perform a data search, while the steganography algorithm has a way of working to insert objects, be it in the form of characters or images. Combining the two algorithms and combining each of its performance advantages, so as to provide suggestions in the process of searching for data and information through data search. The result is brute force algorithm will search the data by shifting from each string, so in doing the data search will take a short time while the steganography algorithm performs the storage of data that has been done matching in the process of searching for data that aims to search the next data.

**Keywords:** Algorithm, Brute Force, Steganography, Vocabulary

## 1. PENDAHULUAN

**B**rute force merupakan algoritma yang dapat digunakan untuk memecahkan suatu masalah dengan sangat sederhana, langsung dan dengan cara yang jelas/lempang. Penyelesaian

masalah kode *cracking* dengan menggunakan algoritma *brute force* akan menempatkan dan mencari semua kemungkinan kode dengan memasukkan karakter dan panjang kode tertentu dan tentunya dengan banyaknya kombinasi kode tertentu yang digunakan[1]. Untuk melakukan pencarian data, *brute force* akan melakukan penelusuran kesetiap *string* serta melakukan percobaan-percobaan pencarian dengan menempatkan dan menggeser setiap posisi string dari satu posisi ke posisi lainnya, sehingga dapat menemukan kombinasi data yang sesuai dengan pencarian.

Dalam hal ini setelah proses percobaan pencarian data sesuai, maka dilakukan kombinasi pencocokan kata-kata yang sesuai dengan kata kunci yang akan dicari oleh *user* dalam melakukan pencarian data-data dan informasi. Sebagai arsip setelah proses pencarian data, maka hasil string pencarian data tersebut akan tersimpan kedalam memori. Dalam proses penyimpanan tersebut, algoritma steganografi akan melakukan tugasnya untuk memproses disposisi penyimpanan dari tiap-tiap string data. Algoritma *steganografi* mempunyai metode yang bisa digunakan seperti LSB (*Least Significant Bit*) dan EOF (*End Of File*). Kedua algoritma ini mempunyai konsep yang berbeda dalam proses penyamaran dan penyembunyian data[2]. Kedua algoritma ini juga masih digunakan sebagai sarana pengamanan dan penyisipan data didalam ilmu steganografi. Didalam perkembangan IT telah hadir sebuah mesin pencarian didalam sistem komputer yang merupakan sebuah fasilitas yang dapat memudahkan *user* untuk melakukan pencarian data-data dan informasi yang diinginkan pada saat ini. Beberapa mesin pencarian yang dapat memudahkan *user* untuk pencarian data-data dan informasi dengan menambahkan saran-saran kata sebagai kata kunci dan memadukannya dengan kosa kata-kosa kata yang biasa digunakan oleh *user* didalam kesehariannya berinteraksi dengan sistem komputer[3].

Dalam penggunaan pemberian saran kata terhadap kosa kata yang akan digunakan ini membutuhkan algoritma pencocokan *string*, yang dapat dijadikan sebagai alternatif dalam penyelesaian masalah tersebut. Banyak algoritma *string matching* yang telah dikembangkan dan digunakan dalam model pencarian kata-kata, diantaranya *Brute Force*[4]. *Brute Force* akan melakukan pencarian data-data dari setiap karakter yang dimulai dari sebelah kiri karakter ke sebelah kanan karakter[5], hal ini membuktikan bahwa dalam perancangannya pencarian kata dapat dibuat menggunakan algoritma *String Matching* dalam pencarian karakter yang dibutuhkan oleh *user*.

Algoritma pencocokan *string* pada dasarnya telah melakukan pengimplementasian terhadap beberapa perangkat lunak yang sudah ada secara praktis. Pencarian kata digunakan untuk mencari satu atau lebih *string* yang sering disebut dengan *pattern* (*string* yang akan dicocokkan kedalam teks) dalam *string* yang disebut dengan *text* (*string* yang disisipkan)[6]. Dua teknik utama yang digunakan dalam algoritma pencocokan *string* yang benar-benar harus persis/serupa, yaitu *exact matching* yang melakukan pencocokan *string* dan *approximate matching* dengan melakukan pencocokan *string* yang tidak harus sama persis/serupa.

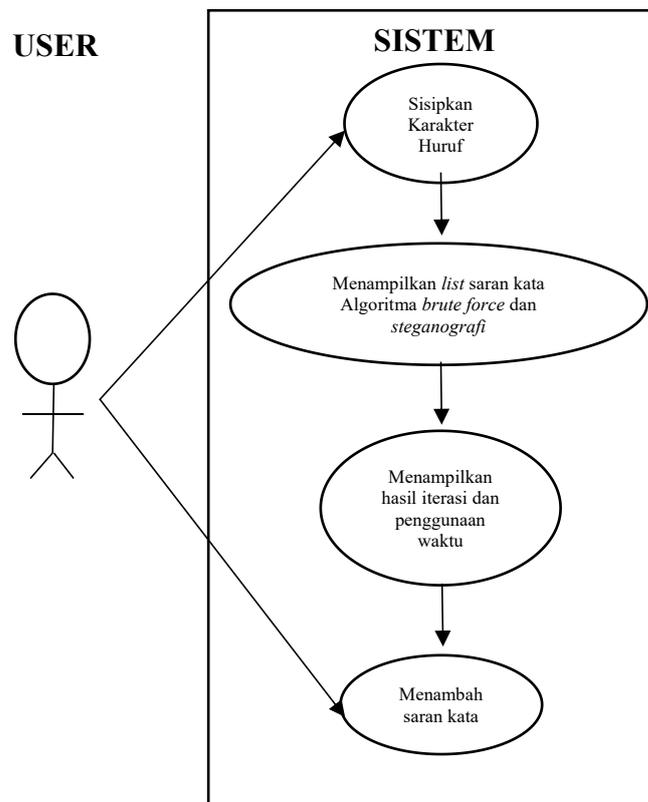
Algoritma *brute force* melakukan pengekan pada setiap kedudukan teks dari karakter pertama hingga karakter terakhir. Sesuai dengan pengecekan karakter pertama terjadi, maka untuk proses *shift* dilakukan dengan berpindah tempat satu posisi ke sebelah kanan (saling berpindah posisi sebanyak 1 karakter ke sebelah kanan dan seterusnya)[6]. Algoritma *brute force* tidak memerlukan tahapan pra-proses. Selama tahap pencarian, perbandingan karakter pada teks dapat selesai pada posisi dimanapun. Untuk proses pencarian teks digambarkan kedalam *coding* pencarian sebagai berikut :

```
Void BF (char *x, y; int m, n)
{
    Int i, j;
    /* Proses Pencarian */
    For (j = 0; j <= n - m)
    {
        For (i = 0; i < m && x[i] == y[i + j]; ++i);
        If (i >= m) OUTPUT (j);
    }
}
```

Pada fungsi *brute force* panjang teks disisipkan kedalam variabel  $n$ , panjang *pattern* disisipkan kedalam variabel  $m$ , selanjutnya dilakukan perulangan untuk melakukan perbandingan karakter yang dimulai dari 0 sampai  $(n-m)$ , dalam pengulangan tersebut dilakukan perbandingan karakter *text* dan *pattern*[7]. Jika ditemukan kecocokan, maka nilai  $i$  akan dikembalikan, jika tidak maka nilai  $i$  akan menjadi -1.

## 2. METODE PENELITIAN

Dalam melakukan pemodelan sistem, diperoleh gambaran tentang objek apa saja yang berinteraksi dengan sistem, serta hal apa saja yang harus dilakukan oleh sebuah sistem agar sistem tersebut berfungsi dengan baik. *Use case* dari penggunaan sistem ini sebagai berikut



**Gambar 1.** *Use Case Diagram Sistem*

**Tabel 1.** Proses pencarian saran kata

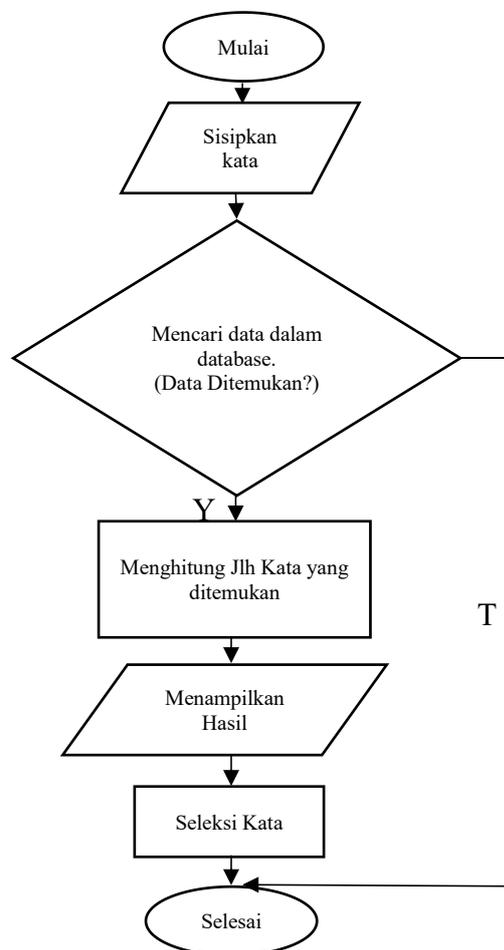
Nama	Proses Pecarian Saran Kata
User	Pengguna yang sudah ditentukan
Deskripsi	<i>Use case</i> ini mendeskripsikan pencarian saran kata menggunakan algoritma <i>brute force</i> dan <i>Steganografi</i>
Dasar Pencarian	Pengguna menyisipkan karakter huruf yang akan dicari
Saran/alternatif kata	Pengguna dapat memilih atau menggunakan saran kata yang ditampilkan oleh sistem
Menampilkan kondisi	Pengguna mengetahui kondisi dan hasil iterasi dan penggunaan waktu pencarian saran kata

Data saran kata yang digunakan berdasarkan pada kata-kata yang telah disisipkan dan tersimpan sebelumnya pada media basis data, data-data tersebut berisi kata yang berdasarkan pada istilah kamus komputer, seperti sampel kata-kata berikut :

**Tabel 2.** Sampel data saran kata

No	Saran Kata
1.	Kita
2.	Kata
3.	Kiat
4.	Kamu
5.	Aku
6.	Dst..

Dalam pencarian informasi, proses kecepatan merupakan salah satu hal yang penting untuk proses pencarian informasi. Beberapa mesin pencarian kata menambahkan fitur saran kata, yang digunakan untuk memberikan saran-saran kata kepada pengguna agar dapat memerikan saran kata dari pada informasi yang dicari. Pengguna tidak perlu mengetikkan kata-kata yang akan dicari secara lengkap, sehingga proses ini dapat dianggap sangat efektif untuk meningkatkan proses pencarian informasi.

**Gambar 2.** Langkah Kerja Saran Kata

Dalam pencocokan saran kata ini membutuhkan algoritma *string matching* untuk mencocokkan karakter-karakter sisipan dari pengguna dengan basisda informasi agar menghasilkan sugesti kata yang tepat. Maka dari pada itu menganalisis pencarian saran kata dengan menggunakan algoritma *Brute Force* untuk mencocokkan karakter kata yang akan diberikan saran dapat mempengaruhi pengguna untuk membantu proses pencarian kata.

Algoritma *Brute Force* digunakan dalam pengecekan pada setiap kedudukan dalam teks dari karakter pertama hingga akhir karakter selesai. Sesuai dengan pengecekan karakter pertama terjadi, maka proses perpindahan dilakukan sebanyak satu posisi kearah kanan (dan seterusnya)[8], algoritma *Brute Force* tidak memerlukan tahapan praproses. Selama tahap pencarian, pembandingan karakter teks dapat selesai pada posisi dimanapun.

### 3. HASIL DAN PEMBAHASAN

Setelah melalui proses analisis, berikut tampilan antarmuka sistem dengan pengguna, rancangan digunakan untuk acuan *coding* setiap halaman-halaman pada *software*. Berikut tampilan antarmuka sistem dan pengguna.

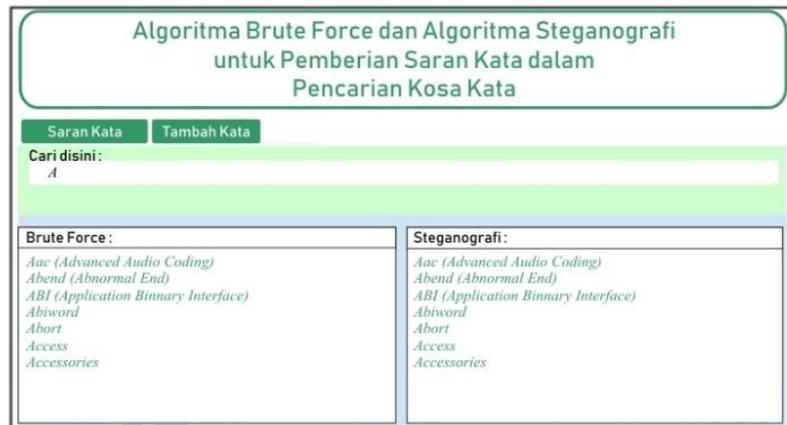
**Gambar 2.** Halaman Utama

Halaman ini akan menampilkan tampilan proses pencarian saran kata. Pada halaman ini, pengguna dapat langsung berinteraksi ke sistem dengan melakukan pencarian kata yang ingin dicari, maka setiap pengetikan karakter dari kata yang dicari, proses algoritma akan langsung memberikan saran kata yang dapat memudahkan pengguna untuk memilih kata-kata apa yang akan digunakan dalam proses pencarian kata.

No.	Kata	Action
1	a	edit delete
2	ab	edit delete
3	aba	edit delete
4	aba-aba	edit delete
5	abad	edit delete
6	abadi	edit delete
7	abadiah	edit delete
8	abah	edit delete
9	abai	edit delete
10	abaimana	edit delete

**Gambar 3.** Penambahan Pustaka Kata

Pada halaman ini merupakan halaman penambahan kosa kata yang apabila terdapat kosa kata yang baru, maka kata-kata tersebut dapat disisipkan kembali kedalam sistem sebagai pustaka pencarian kosa kata dan saran kata. Sebagai sampel pengujian saran kata, pengguna ingin mencari kata "*Accessories*" dan mengetikkannya kedalam sistem. Lalu saran kata akan memeriksa kedalam semua basis data informasi yang mengandung huruf "*A*".



**Gambar 4.** Pencarian sampel data kata ke-1

Pada basis data, informasi yang didapat berjumlah 8, yaitu [A]AC, [A]bend, [A]BI, [A]biword, [A]bort, [A]ccess, [A]ccessories. Saran kata akan menampilkan seluruh kata pada basis data yang mengandung huruf “A” sebagai saran dari kata yang dicari oleh pengguna. Kemudian pengguna mengetikkan karakter kedua yaitu “Ac”, maka saran kata akan memeriksa ke basis data informasi yang mengandung huruf “Ac”.



**Gambar 5.** Pencarian sampel data kata ke-2

Selanjutnya basis data informasi akan memeriksa basis data yang mengandung huruf “Ac”, terdapat 7 data terdekat, yaitu [A]ccess, [A]ccess Method, [A]ccess Point, [A]ccess Provider, [A]ccess Time, [A]ccessibility, [A]ccessories. Lalu seterusnya setiap pengguna mengetikkan kata yang ingin dicari, maka sistem akan langsung memberikan saran kata yang ditampilkan sebagai pilihan terhadap pengguna kata yang akan dipilih dan digunakan.

Semua tabel dan gambar harus jelas/tidak kabur/buram. Ukuran huruf pada tabel dan gambar harus dapat dibaca oleh mata normal dengan mudah. Posisi tabel atau gambar disuatu halaman, sebaiknya terletak dibagian atas atau bawah halaman pada tiap kolom. Contoh dapat dilihat pada tabel 1 atau gambar 1. Meletakkan tabel atau gambar ditengah halaman atau paragraf supaya dihindari. Tabel dan gambar diletakkan pada posisi tengahsetiap kolom (*center alignment*).

#### 4. KESIMPULAN

Dalam melakukan pencarian kata melalui saran kata, algoritma *brute force* dan algoritma *steganografi* dapat diterapkan dengan baik. Kedua algoritma ini dapat dikombinasikan untuk proses

pencocokan karakter kata yang akan dicari dengan melakukan pemeriksaan terhadap setiap karakter. Hal ini dapat membuktikan dalam pencarian saran kata dapat di implementasikan dan dikombinasikan kedalam algoritma *String Matching*.

#### DAFTAR PUSTAKA

- [1] Gunawan, I., “*Fungsi Algoritma Hill Cipher untuk Pengamanan File Gambar dari Serangan Brute Force*”. JURIKOM PRIMA (Jurnal Sistem Informasi Ilmu Komputer Prima). Vol. 2, No. 1, pp-28-32, Juli 2018.
- [2] Gunawan, I., “*Penggunaan Algoritma Kriptografi Steganografi Least Significant Bit Untuk Pengamanan Pesan Teks dan Data Video*”. Jurnal Sains Komputer & Informatika (J-SAKTI), Vol. 2, No. 1, pp-57-65, Maret 2018.
- [3] Januardi, A., “*Analisa Perbandingan Algoritma Brute Force dan Boyer Moore dalam Pencarian Word Suggestion Menggunakan Metode Perbandingan Eksponensial*”. Skripsi. Sekolah Tinggi Manajemen Informatika dan Komputer Budidarma – Medan. 2013.
- [4] Lubis, M. R., “*Implementasi Algoritma Brute Force untuk Permasalahan dalam Penerjemahan Idiom Bahasa Inggris*”. Skripsi. Universitas Sumatera Utara – Medan. 2014.
- [5] Gunawan, I., “*Penggunaan Brute Force Attack dalam Penerapannya pada Crypt-8 dan CSA-Rainbow Tool untuk Mencari BISS*”. InfoTekjar (Jurnal Nasional Informatika dan Teknologi Jaringan). Vol. 1, No. 1, pp-52-55. September 2016.
- [6] Charras, C. & Lecroq, T., “*Handbook of Exact String-Matching Algorithm*”. London : King’s College Publication. 2004.
- [7] Pradana, F., “*Penerapan string matching pada fitur Auto correct dan fitur Auto text di smart phone*”. Institut Teknologi Bandung – Bandung. 2012.
- [8] Gunawan, I., “*Modifikasi Keamanan File dengan Algoritma Hill Cipher untuk Mengantisipasi dari Serangan Brute Force*” TECHSI (Jurnal Teknik Informatika). Vol. 11, No. 2, pp. 237-246, September 2019.