# *Penerapan String Matching dalam Penggunaan Spam Filter*

## *Menggunakan Algoritma Boyer-Moore-Horsepool*

Samantha Olivia Tandri - 13517123
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13517123@std.stei.itb.ac.id

***Abstrak*—Pencegahan spam yang semakin lama semakin banyak menyerang pengguna surat elektronik dapat dicegah dengan melakukan penyaringan surat elektronik dengan menggunakan beberapa metode. Salah satunya adalah dengan menggunakan keyword filtering dan diterapkan dengan menggunakan salah satu algoritma string matching yaitu boyermoore horspool, Terdapat beberapa kelebihan dan kekurangan dalam menggunakan algoritma ini dalam pencocokan string yang akan dibahas.**

***Keywords—spam, filter, boyermoore, string-matching, exact.***

## I. ↑ PENDAHULUAN

Penggunaan surat elektronik kini sudah tidak dapat lepas lagi dari kehidupan masyarakat dunia. Dari berbagai layanan komunikasi digital seperti berbagai *chat messanger*, Facebook, maupun Twitter, penggunaan surat elektronik tetap menjadi fasilitas yang kerap kali digunakan baik untuk berkomunikasi yang bersikap resmi maupun komunikasi santai antar teman. Hingga kini tercatat bahwa jumlah penyedia layanan surat elektronik telah mencapai 255 penyedia layanan dan pasti akan terus berkembang seiring dengan waktu.

Ketika sebuah surat elektronik dikirim dan diterima oleh seseorang maka suatu komunikasi terjalin. Berdasarkan data statistic yang disediakan oleh Statista, tercatat bahwa hingga tahun ini ada 293.6 Miliar surat elektronik yang terkirim setiap harinya. Sehingga dapat dikatakan sejumlah itulah terjadi komunikasi dengan surat elektronik setiap harinya. Namun ternyata pada kenyataannya hampir 49,7 persen dari jumlah surat elektronik tersebut merupakan spam.

Spam menyerang dengan penggunaan system surat elektronik untuk mengirim pesan yang tidak sesuai, tidak relevan, atau tidak sesuai dalam jumlah yang banyak. Target yang sangat mudah dalam menerima spam ini adalah surat elektronik. Dewasa ini banyak penyedia layanan *spam filtering*. Layanan ini akan menyaring semua surat elektronik yang masuk ke dalam server dan menyeleksi surat-surat elektronik yang diduga adalah spam. Namun, efisiensi layanan penyaringan spam ini masih bisa dipertanyakan karena pada kenyataannya sering kali surat elektronik yang isinya merupakan hal penting yang harus dilihat oleh pengguna dikategorikan sebagai spam dan sebaliknya surat elektronik yang berisi spam tidak tersaring dan masuk ke dalam *inbox* surat elektronik.

Maka dari itu, diperlukan aplikasi penyaringan yang dapat memproses surat elektronik dan mendeteksi spam hanya dengan kata pemicu spam umunya namun disesuaikan dengan konten dan konteks surat elektronik dari sisi pengguna masing-masing.

## II. TEORI DASAR

### *A. Pencocokan String*

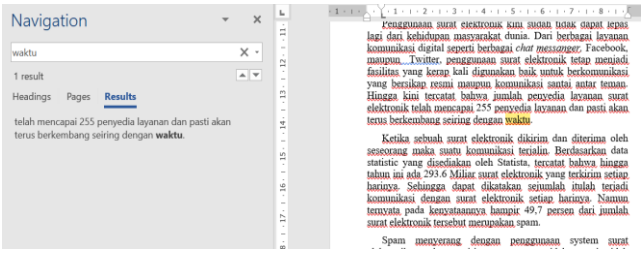
String matching (pencocokan string) adalah suatu algoritma yang digunakan untuk memecahkan masalah pencocokan suatu teks terhadapap teks lain(pattern) . Persoalah string matching dirumuskan sebagai berikut:

Diberikan:

1. Teks (text), yaitu (long) string yang panjangnya n karakter.

2. Pattern, yaitu string dengan panjang m karakter (m < n) yang akan dicari di dalam teks.

Implementasi dari string matching antara lain pencarian suatu kata di dalam dokumen (misalnya menu Find di dalam Microsoft Word).




Contoh :

Pattern : rumah

Teks : Hari ini kami tidak pulang ke rumah karena macet.

↑ *match.*

Dengan asumsi teks berada di dalam memori (bila mencari string di dalam arsip, maka semua isi arsip - atau potongan besar data arsip - perlu dibaca terlebih dahulu, dan menyimpannya di dalam memori). Jika pattern muncul lebih dari sekali di dalam teks, maka pencarian hanya memberikan keluaran berupa lokasi pattern ditemukan pertama ka1i. Selain itu, untuk membedakan arrtara pattern masukan dengan pattern di dalam teks, maka pattern yang berada di dalam teks dinamakan target.

### *B. Algoritma Brute Force*

Dengan asumsi bahwa teks berada di dalam array T[1..n] dan pattern berada di dalam array P[1..m], maka algoritma brute force pencocokan string adalah sebagai berikut :

1. Mula-mula pattern P dicocokkan pada awal teks T.
2. Deng bergerak dari kiri ke kanan, bandingkan setiap karakter di dalam patter P dengan karakter yang bersesuaian di dalam teks T samapi :

   a. Semua karakter yang dibandingkan cocok atau sama (pencarian berhasil), atau
   b. Dijumpai sebuah ketidakcocokan karakter (pencarian belum berhasil).
3. Bila pattern P belum ditemukan kecocokannya dan teks T belum habis, geser pattern P satu karakter ke kanan dan ulangi langkah 2.
4.

```
Procedure BruteForceSearch (input m, n :
integer, input P :
array[1..m] of char, input T :
array[1..n] of char, output idx :
integer)

{ mencari kecocokan pattern P di dalam
teks T. Jika ditemukan P di dalam T,
lokasi awal kecocokan disimpan di dalam
peubah idx.

Masukkan : pattern P yang panjangnya m
dan teks T yang panjangnya n. Teks T
direpresentasikan sebagai string (array
or character).
Keluaran : posisi awal kecocokan (idx).
Jika P tidak ditemukan, idx = -1.}

Deklarasi
s, j : integer
ketemu : Boolean

Algoritma
S ← 0
ketemu ← false
while (s ≤ n-m) and (not ketemu) do
    j ← 1
    while (j ≤ m) and (P[j] = T[s+j]) do
        j ← j + 1
    endwhile
{ j > m or P[j] ≠ T[s+j] }
    if (j = m) then {kecocokan ditemukan}
        ketemu ← true
    else
        s ← s + 1
    endif
endwhile
{ s > n – m or ketemu }
if (ketemu) then
    idx ← s + 1
else
    idx ← -1
endif
```

Kompleksitas algoritma pencocokan string dihitung dari jumlah operasi perbandingan yang dilakukan. Kompleksitas kasus terbaik adalah O(n). Kasus terbaik terjadi jika karakter pertama pattern P tidak pernah sama dengan karakter teks T yang dicocokkan (kecuali pada pencocokan terakhir). Pada kasus ini, jumlah perbandingan yang dilakukan paling banyak n kali misalnya :

Teks : Ini adalah string panjang yang berakhir dengan zz
*Pattern* : zz

Kasus terburuk membutuhkan m(n – m + 1) perbandingan, yang mana kompleksitasnya adala O(mn), misalnya :

Teks : aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaab
*Pattern* : aaaab

### *C. Boyer-Moore Algorithm*

Algoritma Boyer-Moore merupakan variasi lain dari pencarian string dengan melompat maju sejauh mungkin. Tetapi, algoritma BM berbeda dengan algoritma KMP, dimana algoritma BM melakukan perbandingan pattern mulai dari kanan (belakang), sedangkan algoritma KMP melakukan perbandingan mulai dari kiri (depan). Pencocokan string algoritma Boyer-Moore dilakukan dengan 2 teknik yaitu:

a) *The looking-glass technique*: cari pattern dalam text dari belakang text.
b) *The character-jump technique*: ketika ketidakcocokan terjadi, geser pattern pada text sebanyak suatu nilai untuk mencegah pencocokan yang sia-sia.

Algoritma Boyer-Moore menggunakan fungsi kemunculan terakhir (Last Occurrence Function) L() dimana pattern akan diproses bersama kumpulan karakter A terlebih dahulu sebelum dilakukan pencarian pattern pada text. L() memetakan setiap karakter dalam A sebagai integer. Fungsi L(x) dimana x bagian dari A didefinisikan sebagai indeks terbesar dimana pattern == x, atau -1 jika tidak ada indeks

tersebut. Biasanya L() disimpan dalam suatu array.

```
Algoritm BMMatch(T, P):
Input: String T (text with n characters
and P (pattern) with m characters
Output: Starting index offirst substring
of T matching p, or an indication that P
is not a substring of T

i <-- m - 1
j <-- m - 1

repeat

if P[j] = T[i] then

if j = 0 then

return i {a match!}

else {cek karakter selanjutnya }

i <-- i - 1
j <-- j - 1

else { P[j] <> T[i] pattern gerak}

i <-- i + m - j - 1
i <-- i + max(j - last(T[i]), match(j))
j <-- m-1

until i > n - 1

return "There is no substring of T
matching P."
```

*D. Boyer-Moore- Horsepool*

Algoritma Horspool merupakan penyederhanaan dari algoritma Boyer-Moore. Perbedaan antara keduanya adalah pada metode penggeseren 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), seangkan untuk metode praproses nya adalah O(m+σ).

Algoritma Horspool hampir mirip dengan algoritma Boyer-Moore, pergeseran pada Algoritma Horspool adalah :

- Menggunakan pencocokan pattern indeks terakhir dengan text.
- Jika pattern indek terakhir terjadi kesaaman dengan text maka akan dilakukan pengecekan unutk indek pattern sebelumnya.
- Hal itu dilakukan secara terus menerus sampai ditemukanya pattern pada teks.

.

Berikut merupakan contoh kode dari algoritma BM-Horspool.

```
procedure horspoolInitocc()
{
    i traversal[0..alphabetsize]
        occ[a] <-- occ[a]-1

    j traversal[0..m-1]
    {
        a<--p[j]
        occ[a] <--j
    }
}
procedure horspoolSearch()
{
    integer i<--0
    integer j;
    while (i<=n-m)
    {
        j<--m-1;
        while (j>=0 && p[j]==t[i+j]) j--;
        if (j<0) report(i);
        i<-- i +m-1;
        i<-- i - occ[t[i]];
    }
}
```

*E. Spam*

Spam adalah penyalahgunaan sistem pesan elektronik (termasuk media penyiaran dan sistem pengiriman digital) untuk mengirim pesan massal yang tidak diinginkan tanpa harus memperhatikan kepentingannya apa. Yang paling dikenal bentuk spam adalah spam email.

a. Jenis – jenis spam

a) Pengalihan licik dan/atau penyelubungan

Situs sepertinya menyelubungi (menampilkan konten berbeda pada pengguna manusia, berbeda dari apa yang ditampilkan ke mesin telusur) atau mengarahkan ulang pengguna ke laman berbeda selain apa yang dilihat Google.

b) Situs yang diretas

Sebagian laman pada situs ini mungkin telah diretas oleh pihak ketiga untuk menampilkan konten atau tautan berisi spam. Pemilik situs web harus mengambil tindakan segera untuk membersihkan situs mereka dan memperbaiki kerentanan keamanan yang ada.

c) Teks tersembunyi dan/atau penjejalan kata kunci

Beberapa laman Anda mungkin berisi teks tersembunyi dan/atau penjejalan kata kunci.

d) Domain yang diparkir

Domain yang diparkir adalah situs placeholder dengan sedikit konten unik, jadi Google biasanya tidak memasukkan domain seperti ini ke dalam hasil penelusuran.

e) Spam Murni

Situs tampaknya menggunakan teknik spam agresif seperti omong kosong yang dibuat secara otomatis, penyelubungan, konten yang dicuri dari situs web lain, dan/atau pelanggaran berat maupun berulang dari Pedoman Webmaster Google.

f) Penyedia DNS dinamis dan hosting gratis berisi spam

Situs ini dihosting oleh penyedia DNS dinamis atau layanan hosting gratis yang memiliki sebagian besar konten berisi spam.

g) Konten tipis dengan sedikit atau tanpa nilai tambah

Situs tampaknya terdiri dari laman dangkal atau berkualitas rendah yang tidak menyediakan nilai tambah yang banyak kepada pengguna (seperti laman afiliasi tipis, laman pintu, situs pemotong cookie, konten yang dibuat secara otomatis, atau konten salinan).

h) Tautan tidak wajar ke situs

Google telah mendeteksi pola tautan tidak alami, palsu, menipu, atau manipulatif yang menunjuk ke situs ini. Hal ini mungkin disebabkan oleh pembelian tautan yang lolos PageRank atau partisipasi dalam skema tautan.

i) Spam buatan pengguna

Situs tampaknya berisi konten berisi spam yang dibuat pengguna. Konten yang bermasalah mungkin akan muncul pada laman forum, laman buku tamu, atau profil pengguna.

### *F. Spam Filter*

Hingga saat ini, belum ad acara pasti untuk menghilangkan *spam* . Namun cara lain yang dapat mengatasinya adalah dengan mengotomatisasi proses pemilahan(*filtering*) antara email *spam* dan yang bukan.

Teknik- Teknik yang dapat diterapkan dalam spam filtering adalah sebagai berikut.

- Keyword filtering

Metode ini merupakan Application Layer Filtering (ALF). Dengan metode ini, spam di-blok berdasarkan kata-kata tertentu yang sering dituliskan pada spam-mail misalnya : “sexy”, atau “menangkan 1000000 dollar”.

- Address blocking

Metode ini memblok spam-mail berdasarkan IP atau domain atau alamat e-mail tertentu yang telah dikategorikan sebagai alamat spammer.

- Black Listing

Metode ini hampir sama dengan address blocking, yaitu mem-blok spam berdasarkan list alamat spammers yang telah diketahui. Biasanya black listing ini dikerjakan oleh beberapa sukarelawan dan dibuat dalam bentuk database spam-mail , sehingga dapat digunakan oleh semua orang. Salah satu black listing yang dapat diakses adalah Open Relay Data Base, ORDB.org

- White Listing

Kebalikan dengan Black listing, white listing berisi daftar alamat yang dikategorikan sebagai pengirim e-mail yang sah (legitimate mail). Alamat pengirim mail yang tidak termasuk dalam daftar ini akan diasumsikan sebagai spam-mail

- Bayesian Filtering

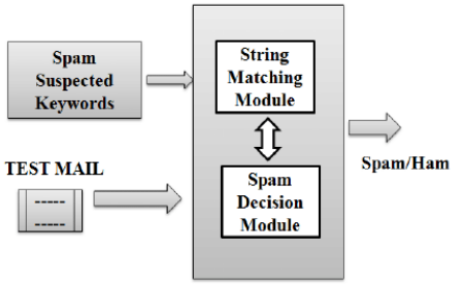
Metode Bayesian Filtering merupakan metode anti spam filter yang terbaru. Metode ini mengenali spam berdasarkan kata-kata (token) yang terkandung pada sebuah e-mail. Metode filter ini pertama kali perlu di-“training” menggunakan dua koleksi e-mail, satu koleksi merupakan spam-mail, dan koleksi yang lain merupakan legitimate mail. Dengan cara seperti ini, pada setiap e-mail baru yang diterima, Bayesian filter dapat memperkirakan probabilitas spam berdasarkan kata-kata yang sering muncul di koleksi spam-mail atau di koleksi legitimate mail. Bayesian filter efektif untuk mem-blok spam karena filter ini dapat secara otomatis mengkategorikan spam-mail atau legitimate mail.

- Rule based(heuristic) filtering

Filter ini mem-blok spam-mail dengan mencari pola karakteristik tertentu yang mengindikasikan spam contohnya : kata-kata “kotor”, kata dengan banyak huruf besar atau banyak tanda seru, atau tanggal pengiriman yang tidak tepat. Kekurangan dari metode ini adalah rule (aturan) yang digunakan bersifat statis, sehingga jika spammers menggunakan pola baru untuk mengirim spam-mail, aturan yang baru harus diberikan pada filter. Sedangkan pada Bayesian filter, kita cukup memberitahu filter bahwa pengklasifikasian e-mail yang dilakukannya salah, maka Bayesian filter akan secara otomatis mempelajari pola yang terdapat pada e-mail tersebut.

## III. CARA KERJA SPAM FILTERING DENGAN MENGGUNAKAN KEYWORD FILTERING

Seperti yang telah dijelaskan sebelumnya spam membawa banyak pengaruh buruk terhadap pengguna email. Selain memakan banyak *network bandwidth*, menyebabkan malware, virus yang dapat menyebabkan kerugian finansial. Sehingga penggunaan *spam filters* dapat mengidentifikasi dan membuang spam dari *inbox* surat elektronik.




Melalui metode *keyword triggerword* surat elektronik akan di difilter dengan spam decision module yang telah dibuat dan akan dideteksi pola yang dicurigai sebagai spam *email.* Semua konten yang akan difilter menggunakan *exact string matching.*

1. Surat elektronik yang akan diperiksa diinput ke dalam spam detection module.
2. Data Keyword yang akan digunakan untuk mendeteksi pola diinput ke dalam spam detection module.
3. Dari kedua input tersebut akan diidentifikasi surat elektronik yang diinput dan akan dipilah mana surat yang dikategorikan sebagai spam dan yang tidak

## IV. IMPLEMENTASI

Pada implementasi *spam filter* ini , penulis menggunakan Bahasa python dan sebuah database trigger keyword dengan kata dasar Bahasa Inggris yang dapat diakses dari spam keyword umum.

Pencocokan string yang digunakan dengan algoritma Boyer Moore Horspool.

Dari database maka akan dicari apakah ada kecocokan dari teks surat elektronik yang diinput. Dimana tiap terjadi kecocokan maka penghitung akan bertambah satu. Diakhir pemeriksaan akan dicari apakah jumlah spam keyword yang dideteksi telah mencapai batas maksimum yang telah ditentukan.

Contoh :

Kata *Spam Triggerword:*

| | |
|---|---|
| *No claim form* | *No gimmick* |
| *No catch* | *Offer* |
| *Prizes* | *See it yourself* |
| *Get* | *Give it away* |
| *Cancet at any time* | *Winning* |
| *Special* | *Announcement* |
| *Guaranteed* | *100% real* |
| *Giving* | *Announced* |
| *Special* | *Jackpot* |
| *Win* | *Million* |
| *Secretly* | *Without any* |
| *Purchase* | *Contact us* |

Pesan surat elektronik:

winning Jackpot !!

Hello Mr Dodododo

Now We have finally anounced that you have win the special jackpot prize

from our company. it is guaranteed 100% real. You can contact us with the

number below and we will give you 1 million without any purchase.

Regards, Us

Dari kedua data diatas maka tiap kalimat dimasukan ke dalam array text dan tiap keyword akan dimasukan ke dalam array keyword.

Dari data yang telah disiapkan, tiap kalimat akan dicocokan dengan string keyword sebagai patternya dan diperiksa jumlah keyword yang ada dalam surat elektronik tersebut

```
input : Mail1.txt
----------------
data :
['No claim form', 'No gimmick', 'No catch', '
Offer', 'Prizes', 'See it yourself', 'Get', '
Give it away', 'Cancet at any time', 'Winning
', 'Special', 'Announcement', 'Guaranteed\t',
 '100% real', 'Giving', 'Announced', 'Special
', 'Jackpot', 'Win', 'Million', 'Secretly', '
, 'Without any', 'Purchase', 'Contact us']
['winning Jackpot !!', 'Hello Mr Dodododo', 'Now We have finally anounced that y
ou have win the special jackpot prize', 'from our company. it is guaranteed 100%
 real. You can contact us with the', 'number below and we will give you 1 millio
n without any purchase.', 'Regards, Us']
----------------
RESULT :
This mail is a spam! contains more than 50 percent spam trigger word
---------------
EXECUTION TIME :
0.0019958019256591797
```

## V. ANALISIS

*a) Percobaan*

Untuk mengetahui seberapa akurat *spam filter* yang telah diimplementasikan maka akan dilakukan *testcase* lain yang akan menguji beberapa kasus lain termasuk jika kasus surat elektronik yang diinput bukan merupakan spam sehingga hasil yang diberikan sesuai dengan yang diharapkan oleh pengguna

1. Kasus pertama yaitu surat elektronik yang bukan merupakan spam, penambahan database spam keyword yang lebih akurat.berisi data spam triggerword umum sebanyak 70 data.

```
PS C:\Users\Samantha Olivia\Documents\STIMA> python  bmhorspool.py
input : Mail1.txt
----------------
data :
['No claim form', 'No gimmick', 'No catch', 'Offer', 'Prizes', 'See it yourse
Special', 'Announcement', 'Guaranteed\t', '100% real', 'Giving', 'Announced',
ecial', 'Jackpot', 'Win', 'Million', 'Secretly', 'Without any', 'Purchase', '
act us', 'You have been selected', 'Free Offer', 'Free Access', 'Free Gift',
e Money', 'Free sample', 'Free instant', 'Free consultation', 'Free investmen
'All new', 'All Acurate', 'For Free', 'Amazing', 'Promise You', 'Do it today'
nly', 'take action now', 'phone', 'bonus', 'casino', 'rolex', 'as seen onlyor
status', 'buy', 'clearance', 'only', 'shopper', 'meet singles', 'double your', 'in
come from home', 'make Money', 'extra cash', 'homebased business', 'online degree'
, 'work from homeopportunity', 'bargain', 'big bucks', 'check', 'cheap', 'affordab
le', 'claims', 'collect', 'compare rates', 'bargain', 'why pay more', 'full refung
', 'stock pick']
['Dear Mr Piper,', 'I am writing to thank you for all your help.', 'I look forward
d to seeing you next week.', 'With best wishes,', 'John Smith']
----------------
RESULT :
This mail is not a spam!
---------------
EXECUTION TIME :
0.007602214813232422
```

Dari hasil eksekusi tersebut dapat terlihat bahwa deteksi spam menyatakan bahwa surat elektronik tersebut bukan sebuah spam.

2. Kasus kedua yaitu surat elektronik yang seharusnya merupakan spam dengan penambahan database spam keyword yang lebih akurat.berisi data spam triggerword umum sebanyak 70 data.

```
PS C:\Users\Samantha Olivia\Documents\STIMA> python  bmhorspool.py
input : Mail1.txt
----------------
data :
['No claim form', 'No gimmick', 'No catch', 'Offer', 'Prizes', 'See it yourself',
'Get', 'Give it away', 'Cancet at any time', 'Winning', 'Special', 'Announcement',
 'Guaranteed\t', '100% real', 'Giving', 'Announced', 'Special', 'Jackpot', 'Win',
'Million', 'Secretly', 'Without any', 'Purchase', 'Contact us', 'You have been sel
ected', 'Free Offer', 'Free Access', 'Free Gift', 'Free Money', 'Free sample', 'Fr
ee instant', 'Free consultation', 'Free investment', 'All new', 'All Acurate', 'Fo
r Free', 'Amazing', 'Promise You', 'Do it today', 'only', 'take action now', 'phon
e', 'bonus', 'casino', 'rolex', 'as seen only', 'order status', 'buy', 'clearance'
, 'only', 'shopper', 'meet singles', 'double your', 'income from home', 'make Mone
y', 'extra cash', 'homebased business', 'online degree', 'work from home', 'opport
unity', 'bargain', 'big bucks', 'check', 'cheap', 'affordable', 'claims', 'collect
', 'compare rates', 'bargain', 'why pay more', 'full refung', 'stock pick']
['WELCOME TO THE BEST WORK OPPORTUNITY EVER', 'you have been selected to join our
business', 'we will guarantee you the best work from home OPPORTUNITY', 'you can d
ouble your income from our homebased business', 'no need go to office, but big buc
ks will come to you', 'contact our staff immediately', 'and you will know this ama
zing job']
----------------
RESULT :
This mail is a spam! contains more than 50 percent spam trigger word
---------------
EXECUTION TIME :
0.009661674499511719
```

Dari kasus diatas terlihat bahwa deteksi spam telah menyatakan bahwa surat elektronik tersebut adalah sebuah spam

Sehingga melalui testcase tersebut dan dengan penambahan database trigger keyword yang lebih disesuaikan maka spam filter dapat dilakukan dengan menggunakan algoritma Boyer-Moore- Horsepool.

## KESIMPULAN

Pencocokan string dapat dieksekusi dengan menggunakan algoritma BoyerMooreHorspool. Hasil yang didapatkan memang tidak lebih mangkus dari BoyerMoore namun implentasi algoritma lebih mudah dilakukan terhadap program spam filter ini.

## REFERENSI


[1] Importance of String Matching in Real Word Problems. Diakses 26 April 2019. Dari : https://www.researchgate.net/publication/304305210_Importance_of_String_Matching_in_Real_World_Problems

[2] Spam Filtering Diakses pada 25 April 2019 Dari :
https://www.academia.edu/7331627/Makalah_Spam_Filtering

[3] The Shocking Truth about How Many Emails Are Sent Diakses pada 26 April 2019 Dari:
https://www.campaignmonitor.com/blog/email-marketing/2018/03/shocking-truth-about-how-many-emails-sent/

[4] The Ultimate List of Email SPAM Trigger Words Diakses 25 April 2019. Dari :
https://blog.hubspot.com/blog/tabid/6307/bid/30684/the-ultimate-list-of-email-spam-trigger-words.aspx


## 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, 26 April 2019

Samantha Olivia Tandri
13517123