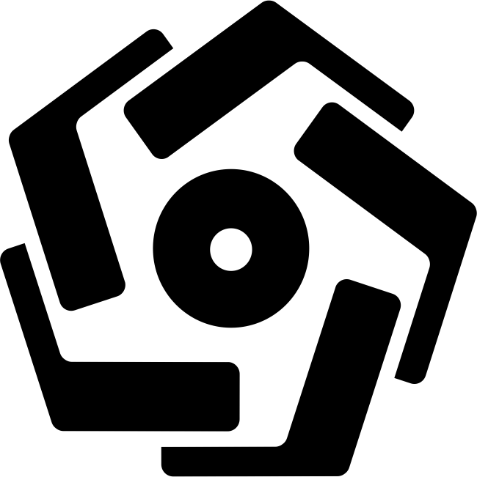
FINAL PROJECT AI

CHAT BOT MENGGUNAKAN ALGORITMA BOYER MOORE



Dosen : Sumarni Adi, S.Kom M.Cs

NAMA ANGGOTA KELOMPOK:

1. Ignatius Andika S 16.11.0001
2. Ichwanuttaqwa Fauzan Anwar 16.11.0002
3. Riko Ajiasa 16.11.0007
4. Rizdhani Maulana S 16.11.0037

**PROGAM STUDI INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS AMIKOM YOGYAKARTA**

**2018**

**Kata Pengantar**

Puji syukur penulis panjatkan atas kehadirat ALLAH SWT yang telah memberikan rahmat-Nya, sehingga kami dapat menyelesaikan Pipeline ini dengan judul “ Chat Bot menggunakan algoritma boyer moore ”.

Penulisan Pipeline ini dimaksudkan untuk memenuhi syarat Ujian Akhir Semester mata kuliah Kecerdasan Buatan di Universitas Amikom Yogyakarta.

Selesainya tugas akhir ini tidak terlepas dari dukungan berbagai pihak yang telah memberikan dorongan moril maupun spiritual dan juga bimbingan ilmu pengetahuan. Oleh karena itu penulis mengucapkan terimakasih kepada:

1. Ibu Sumarni Adi, S.Kom, M.Cs selaku dosen mata kuliah Kecerdasan Buatan.

2. Teman-teman 16-S1 IF-01.

3. Serta semua pihak yang telah membantu dan bekerjasama dalam pembuatan pipeline ini.

Kami menyadari masih begitu banyak kekurangan dalam penyusunan pipeline ini. Untuk itu, kritik dan saran adalah sesuatu yang sangat kami harapkan demi kemajuan bersama dan peningkatan ilmu pengetahuan di Indonesia.

Yogyakarta, 29 Oktober 2018

Penulis

**DAFTAR ISI**

1. **Halaman Cover.................................................................................... 1**
2. **Kata Pengantar.................................................................................... 2**
3. **Daftar Isi.............................................................................................. 3**
4. **Pendahuluan………………………………………………………… 4**
5. **Latar Belakang........................................................................................... 4**
6. **Pengertian Algoritma Booyer Moore........................................................ 5**
7. **Pembahasan…………………………………………………………. 6**
8. **Konsep Dasar Algoritma Boyer Moore..................................................... 6**
9. **Perhitungan Algoritma Boyer Moore........................................................ 7**
10. **Screenshot Implementasi Boyer Moore pada Chat Bot............................11**
11. **Kesimpulan........................................................................................... 12**
12. **Referensi……………………………………………………………... 12**

**PENDAHULUAN**

**Latar Belakang**

**Chatterbot** (disebut juga chatbot atau bots) adalah sebuah program computer yang dirancang untuk menyimulasikan percakapan intelektual dengan satu atau lebih manusia baik secara audio maupun teks.

Pada mulanya, program komputer (*bots*) ini diuji melalui Turing Test, yaitu dengan merahasiakan identitasnya sebagai [mesin](https://id.wikipedia.org/wiki/Mesin) sehingga dapat mengelabui orang yang bercakap-cakap dengannya. Jika pengguna tidak dapat mengidentifikasi *bots* sebagai suatu program komputer, maka *chatterbot* tersebut dikategorikan sebagai [kecerdasan buatan](https://id.wikipedia.org/wiki/Kecerdasan_buatan" \o "Kecerdasan buatan) (atau *artificial intelligence*). Dewasa ini, *chatterbot* telah dimanfaatkan untuk tujuan praktis seperti bantuan online, layanan personal, atau akuisisi informasi, dalam hal ini dapat dilihat fungsi program sebagai suatu jenis agen percakapan (atau *conversational agent*). Yang membedakan *chatterbot* dengan sistem [pemrosesan bahasa alami](https://id.wikipedia.org/wiki/Pemrosesan_bahasa_alami" \o "Pemrosesan bahasa alami) (atau *Natural Language Processing System*) adalah kesederhanaan [algoritme](https://id.wikipedia.org/wiki/Algoritme" \o "Algoritme) yang digunakan.

Meskipun banyak *bots* yang tampaknya dapat mengartikan dan menanggapi [input](https://id.wikipedia.org/wiki/Input) manusia, sebenarnya *bots* tersebut hanya memindai [kata kunci](https://id.wikipedia.org/wiki/Kata_kunci) dalam input dan membalasnya dengan kata kunci yang paling cocok, atau pola kata-kata yang paling mirip dari [basis data](https://id.wikipedia.org/wiki/Basis_data) tekstual.

**Algoritma adalah** urutan atau langkah-langkah untuk penghitungan atau menyelesaikan suatu masalah yang ditulis secara berurutan. Dalam pemrograman, yang penting untuk dipahami adalah logika kita dalam berpikir bagaimana cara untuk memecahkan masalah pemrograman yang akan dibuat. Dalam hal ini, algoritma dan logika pemrograman akan sangat penting dalam pemecahan masalah.

**Algoritma Boyer-Moore** adalah salah satu algoritma pencarian string, dipublikasikan oleh Robert S. Boyer, dan J.Strother Moore pada tahun 1977. Algoritme ini dianggap sebagai algoritma yang paling efisien pada aplikasi umum. Tidak seperti algoritma pencarian string yang ditemukan sebelumnya, algoritma Boyer-Moore mulai mencocokkan karakter dari sebelah kanan pattern. Ide di balik algoritma ini adalah bahwa dengan memulai pencocokan karakter dari kanan, dan bukan dari kiri, maka akan lebih banyak informasi yang didapat.

**PEMBAHASAN**

**Konsep Dasar Algoritma Boyer Moore**

Konsep Dasar dari Algoritma Boyer Moore adalah mencari string dengan melakukan pembandingan karakter mulai dari karakter paling kanan dari string yang dicari. Tentu hal ini memudahkan proses pencarian akan menjadi lebih cepat jika di bandikan dengan program algoritma lainnya.

Langkah – langkah :

1. Algoritma boyer moore mulai mencocokkan pattern pada awal teks
2. Dari kanan ke kiri, algoritma ini akan mencocokan karakter per karakter pattern dengan karakter di teks yang bersesuaian, sampai salah satu kondisi terpenuhi
3. Algoritma kemudian menggeser pattern dengan memaksimalkan nilai penggeseran good-suffix dan perggeseran bad-character, lalu mengulang langkah ke 2 sampai pattern berada di ujung teks

**PERHITUNGAN**

1. Konsep Algoritma Boyer Moore

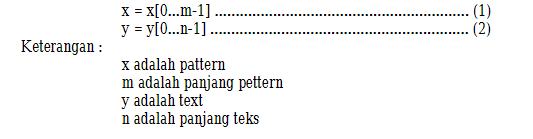
Algoritma ini melakukan pencocokan string dengan bergerak dari kanan ke kiri. Algoritma Boyer-Moore melakukan pencocokan string berdasarkan pada dua teknik, yaitu teknik looking-glass dan teknik character jump. Teknik looking-glass melakukan pencarian pattern pada teks denan bergerak mundur, dimulai dari bagian akhir teks. Pada teknik character-jump, ketika ketidakcocokan terjadi pada T[i] == x, maka karakter P[j] pada pattern tidak akan sama dengan T[i].

Pada algoritma ini, terdapat 3 kasus yang mungkin terjadi.

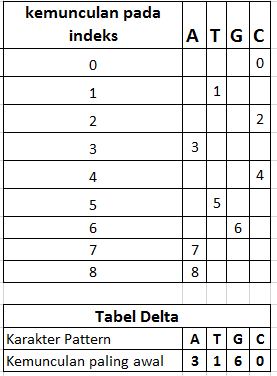
Kasus 1: Jika pattern berisi “X” di posisi tertentu, maka geser pattern ke kanan sesuai dengan kemunculan “X” di pattern, yaitu teks[i].

Kasus 2: Jika pattern berisi “X” di posisi tertentu, tapi menggeser pattern ke kanan tidak mungkin dilakukan, maka geser P ke kanan sebesar 1 karakter, yaitu teks [i+1].

Kasus 3: Jika kasus 1 dan 2 tidak mungkin terjadi, maka geser pattern ke pattern [1] dengan T[i+1].

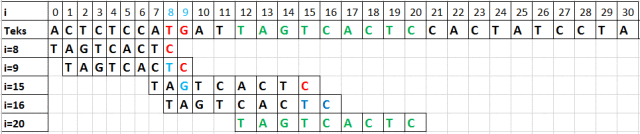


1. Mencari Tabel Delta atau Occurrence Heuristic Table



abel delta didapat dari kemunculan pertama sebuah pattern yang dihitung dari kanan. perhitungan dimulai dari angka nol.

Berikut adalah Tabel pencocokan String dengan menggunakan Algoritma Boyer-Moore



Penjelasan:

1. Ketika dicocokan pertama kali, Indeks ke-8 pada teks (T) tidak cocok dengan pattern pertama paling kanan (C). Hal ini membuat pattern digeser sejauh nilai maksimal dari tabel delta atau sejumlah pergeseran teks. Dicari nilai maksimal dari nilai delta huruf T yakni 1 (lihat tabel delta) dan pergeseran pattern masih 1 kali. Sehingga pattern digeser sejauh 1 langkah.

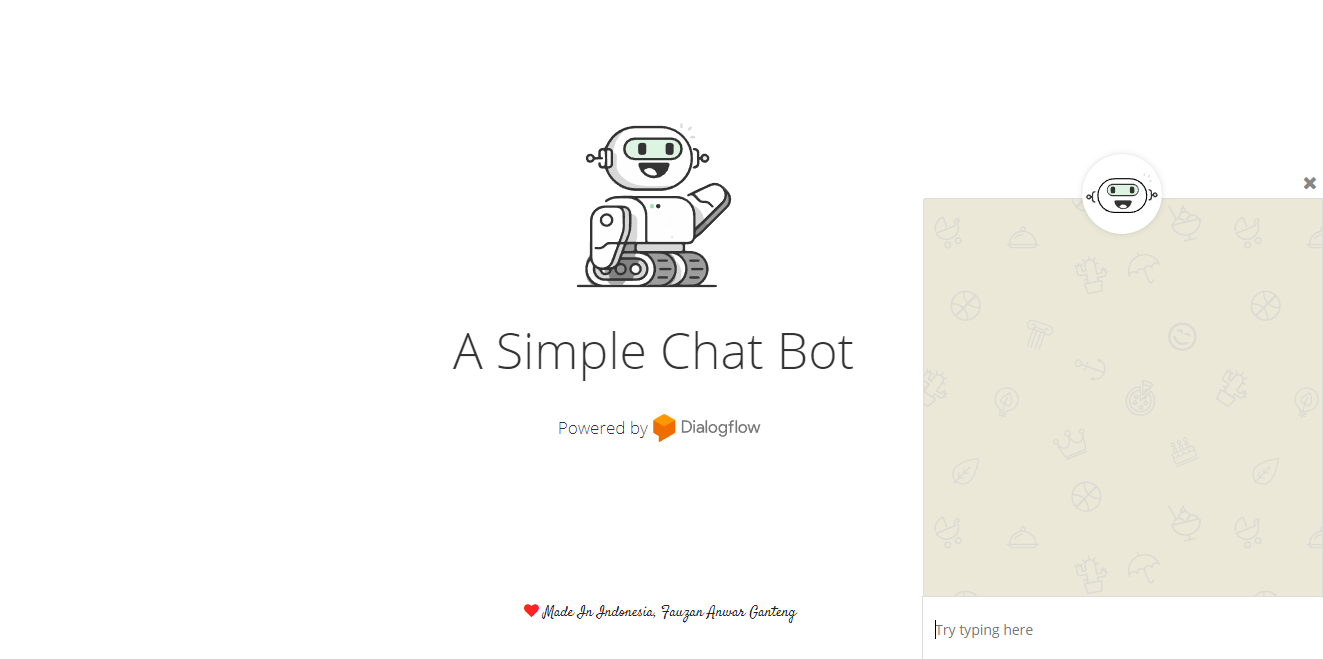
2. Pencocokan kedua dengan menggeser pattern satu langkah ke kanan (i=9). indeks ke-9 pada teks (G) juga tidak cocok dengan pattern (C). Pattern kembali digeser sejauh  nilai maksimal dari tabel delta atau sejumlah pergeseran teks. Dicari nilai maksimal dari nilai delta huruf G yakni 6 dan pergesedan pattern masih 1 kali, sehingga pattern digeser sejauh 6 langkah.

3. Pencocokan ketiga yakni pada indeks ke-15 pada teks (T) tidak cocok dengan pattern pertama paling kanan (C). Hal ini membuat pattern digeser sejauh nilai maksimal dari tabel delta atau sejumlah pergeseran teks. Dicari nilai maksimal dari nilai delta huruf T yakni 1 (lihat tabel delta) dan pergeseran pattern masih 1 kali. Sehingga pattern kembali digeser sejauh 1 langkah.

4. Pencocokan keempat yakni pada indeks ke-16 pada teks (C) cocok dengan pattern pertama paling kanan (C). Hal ini membuat indeks pattern dan teks dikurangi 1 untuk mencocokan pattern selanjutnya. Indeks ke-15 pada teks (T) juga cocok dengan pattern selanjutnya (T) sehingga indeks pattern dan teks dikurangi 1. Pada indeks ke-14 pada teks (G) tidak cocok pada pattern selanjutnya (G) sehingga terdapat pergeseran pattern kembali. Dicari nilai maksimal dari nilai delta huruf G yakni 6 dan pergesedan pattern sudah 1 kali, sehingga pattern digeser sejauh 6 langkah. dari indeks terakhir (indeks ke-14) sejauh 6 langkah (indeks-20).

5. pencocokan kelima yakni pada indeks ke-20. Setelah dilakukan pengecekan dari kanan ke kiri teks terhadap pattern, ditemukan kecocokan seluruh pattern pada teks. Ketika Seluruh pattern telah berhasil dicocokan, maka algoritma akan berhenti.

**Screenshot ChatBot dengan Algoritma boyer moore**



**Kesimpulan**

Algoritma Boyer-Moore untuk mencocokan sebuah String jauh lebih Efisien dibandingkan dengan Algoritma Brute Force karena Pada algoritma Brute-Force, string dicocokan satu per satu hingga selesai. Pada Algoritma Boyer-Moore ketika karakter pada teks yang dicocokan pada pattern tidak cocok, maka pattern akan digeser sejauh nilai maksimun dari nilai delta atau perpindahan pattern tersebut.

**Refrensi**

<http://kukuhtw.com/2016/09/27/membuat-chatbot-dengan-fitur-ai/>

<https://medium.com/kata-engineering/berkenalan-dengan-chatbot-part1-ea92147e0c5d>

<https://sibstudy217180382.blogspot.com/2017/11/algoritma-boyer-moore.html?m=1>

http://technoprast.blogspot.com/2011/04/kasus-pada-algoritma-boyer-moore.html?m=1