

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Stemming merupakan proses pembuangan imbuhan untuk menemukan akar kata (*root*) atau kata dasar dari kata berimbuhan. *Stemming* Bahasa Indonesia perlu diterapkan pada tahap *preprocessing text* untuk meningkatkan akurasi *similarity*, semakin tinggi tingkat akurasi *similarity* maka akurasi pada sebuah aplikasi yang menggunakan algoritma *similarity* lebih akurat.

Pencarian kata dasar (*stemming*) khususnya bahasa Indonesia terdapat beberapa algoritma yang telah dikembangkan sebelumnya, diantaranya algoritma Nazief & Adriani (Asian, 2007) dan algoritma Porter (Tala, 2003).

Algoritma Nazief dan Adriani dikembangkan berdasarkan aturan morfologi Bahasa Indonesia yang mengelompokkan imbuhan menjadi awalan (*prefix*), sisipan (*infix*), akhiran (*suffix*) dan gabungan awalan akhiran (*confixes*). Algoritma ini menggunakan kamus kata dasar untuk menghilangkan imbuhan sehingga akurasi algoritma Algoritma Nazief dan Adriani lebih tinggi dari pada Algoritma Porter, kelemahan algoritma Algoritma Nazief dan Adriani waktu proses yang lebih lama. Algoritma Porter menghilangkan imbuhan berdasarkan aturan morfologi dalam Bahasa Indonesia, tanpa menggunakan kamus. Sehingga waktu proses algoritma Porter lebih cepat.

Algoritma Nazief & Adriani mendapatkan akurasi yang tinggi dengan mencocokkan kata yang akan diubah menjadi kata dasar yang harus melewati tahap

pengecekan kata pada database kamus dasar, sehingga akan membuat waktu proses lebih lama.

Metode filtering *stopword* berfungsi menghilangkan semua kata umum (*common words*) yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna. Seperti apa, adalah, yang, aku, kamu, dan lain-lain. Menambah filtering *stopword* pada preprocessing text dapat mengurangi jumlah kata yang akan diproses.

Pada penelitian ini metode filtering *stopword* yang mampu menghilangkan semua kata umum dapat diterapkan pada Algoritma Nazief & Adriani untuk mengurangi waktu proses stemming.

1.2 Rumusan Masalah

Berdasarkan latar belakang maka dirumuskan permasalahan yang akan diuraikan solusinya yaitu:

1. Bagaimana menerapkan algoritma Nazief & Adriani pada preprocessing text algoritma Rabin Karp.
2. Bagaimana menerapkan *filtering stop word* untuk mengurangi waktu proses algoritma stemming Nazief dan Adriani.

1.3 Batasan Masalah

Batasan masalah yang akan digunakan untuk mempersempit pembahasan adalah sebagai berikut:

1. Penelitian berfokus pada aplikasi berbasis *website*.

2. Bahasa pemrograman yang di gunakan *PHP* dan *MySQL*.
3. Menggunakan algoritma Rabin Karp dan rumus *dice coefficient similarity* untuk pengujian *similarity*.
4. Melakukan perbandingan setelah penerapan Teknik *filtering stopword* pada *preprocessing text*.
5. Perangkat lunak yang digunakan yaitu Brackets (*text editor*), Google Chrome (*Browser*), Virtual Linux, Apache dan phpMyAdmin.
6. Hasil dari penelitian ini akan berupa file berformat php.

1.4 Tujuan Penelitian

Tujuan penelitian yang dilakukan adalah sebagai berikut:

1. Mengetahui penerapan algoritma Nazief & Adriani dapat meningkatkan akurasi *similarity*.
2. Membuat waktu proses pada algoritma Nazief & Adriani lebih cepat dengan menambah Teknik *filtering stop word* pada *preprocessing text*.

1.5 Manfaat Penelitian

Manfaat dari penelitian yang dilakukan adalah sebagai berikut:

1. Mengetahui dan memahami cara mengimplementasikan algoritma stemming Nazief & Adriani pada aplikasi berbasis website.
2. Mengetahui perbandingan waktu proses algoritma Nazief & Adriani setelah diterapkan *filtering stopword*.

3. Sebagai referensi programmer untuk penggunaan / penerapan algoritma *stemming* Nazief & Adriani dan Teknik *filtering stopwords*.
4. referensi untuk melakukan penelitian lebih lanjut tentang *stemming* khususnya Bahasa Indonesia.

1.6 Metode Penelitian

Penelitian ini menggunakan metode penelitian *Research and Development* (penelitian dan pengembangan). Metode ini adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu dan menguji keefektifan produk tersebut. Langkah-langkah yang dilakukan peneliti adalah sebagai berikut:

1.6.1 Metode Pengumpulan data

Pengumpulan data yang dilakukan untuk memperoleh data-data yang dilakukan dalam penelitian ini yaitu:

1. Metode Study Literatur

Mempelajari data-data dari jurnal, buku dan website yang berkaitan dengan algoritma yang akan diteliti. Tujuan dari metode ini adalah untuk memperkuat permasalahan dan juga sebagai dasar teori dalam melakukan penelitian.

2. Metode study pustaka

mempelajari data-data dan arsip yang sudah ada sebagai acuan atau bahan referensi dalam memperoleh data informasi yang dibutuhkan.

1.6.2 Metode Analisis

Metode analisis yang digunakan dalam penelitian ini yaitu analisis kebutuhan sistem yang meliputi kebutuhan fungsional dan non-fungsional dan analisis kelayakan sistem yang meliputi kelayakan teknologi, operasional dan hukum.

1.6.3 Metode Perancangan

Metode perancangan yang dilakukan yaitu membuat prototype terlebih dahulu dengan melakukan simulasi desain kebutuhan sistem seperti *interface* dan *flowchart*.

1.6.4 Metode Pengembangan

Pada tahap metode pengembangan akan mulai membuat kode program berdasarkan desain yang telah dibuat sebelumnya.

1.6.5 Metode Testing

Pada tahap ini dilakukan perbandingan akurasi dan kecepatan pada algoritma Nazief dan Adriani setelah menerapkan Teknik *filtering stopword* pada *preprocessing text* pada algoritma rabin karp dan untuk menghitung *similarity* menggunakan *dice similarity coefficient*.

1.7 Sistematika Penulisan

Penulisan skripsi ini ditulis secara sistematis yang terdiri dari beberapa sub bab. Berikut uraian sistematika penulisan:

1.7.1 BAB I : Pendahuluan

Pada bab ini dijelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian, dan sistematika penulisan.

1.7.2 BAB II : Landasan Teori

Pada bab ini dijelaskan tentang teori, definisi-defenisi, atau model yang berkaitan dengan ilmu dan masalah yang sedang diteliti.

1.7.3 BAB III : Analisis dan Perancangan

Pada bab ini akan menganalisis tentang desain sistem berdasarkan tabel algoritma Nazief dan Adriani (Asian, 2007) yang telah dikembangkan sebelumnya.

1.7.4 BAB IV : Implementasi dan Pembahasan

Pada bab ini akan membahas tentang hasil analisis dan penerapan algoritma Nazief & Adriani dalam proses *string matching* menggunakan algoritma rabin karap dan untuk menghitung *similarity* menggunakan *dice similarity coefficient*.

1.7.5 BAB V : Penutup

Bab ini berisikan uraian kesimpulan dan saran dari perumusan masalah yang disampaikan.