BAB 2

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Beberapa penelitian terdahulu yang menjadi ide dasar dalam pembuatan skripsi ini diantaranya :

1. "Implementasi Algoritma Rabin Karp Untuk Mendeteksi Kemiripan Dokumen Stmik Bandung" oleh Siti Yuliyanti dan Rizky(Yuliyanti & Rizky, 2020).

   Pada tulisan ilmiah tersebut dijelaskan pada proses mencari kemiripan dokumen dilakukan dengan menerapkan metode rolling hash menggunakan k-gram, text mining(case folding, tokenizing, filtering, stemming) dan perhitungan nilai similarity(dice's similarity coeficients, modulus).

2. "Implementasi Algoritma Rabin-Karp untuk Membantu Pendeteksian Plagiat pada Karya Ilmiah" oleh Doddi Aria Putra, Herry Sujaini, dan Helen Sasty Pratiwi(Putra & Sujaini, 2015).

   Pada tulisan ilmiah tersebut dijelaskan system pendeteksian yang dibangun berbasis *web* dengan menggunakan bahasa pemrograman PHP dan *database* MySQL. Proses pendeteksian karya tulis dilakukan melalui proses *case folding*, *tokenizing*, *filtering*, *stemming*, k-gram dan hashing. Dilanjutkan dengan menggunakan algoritma Rabin-Karp

dan untuk menghitung persentase dengan menggunakan *Dice's Similarity Coeficient*.

3. "Aplikasi Pendeteksi KemiripanPada Dokumen Menggunakan Algoritma Rabin Karp" oleh Inta Widiastuti, Cahya Rahmad, dan Yuri Ariyanto(Widiastuti et al., 2017).

   Pada penulisan ilmiah tersebut dijelaskan proses mendeteksi kemiripan menggunakan algoritma Rabin-Karp dengan menggunakan fungsi hash dan K-gram.

4. "RABIN-CARP IMPLEMENTATION IN MEASURING SIMALIRITY OF RESEARCH PROPOSAL OF STUDENTS" oleh Herman, Lukeman Syafie, Tasmil, dan Muhammad Resha (Herman et al., 2020).

   Pada penulisan ilmiah tersebut dijelaskan proses pengujian yang dilakukan berdasarkan nilai k-gram. Dengan nilai k-gram yang diuji adalah 4, 5, dan 6. Nilai tersebut didasarkan pada penelitian yang sebelumnya yang sudah dilakukan. Dan penulisan ilmiah ini menjadi referensi bagi penulis dalam melakukan pengujian aplikasi.

Dari referensi yang telah ada, system yang akan dibuat penulis memiliki fitur-fitur yaitu dapat memasukkan file dokumen, dapat melakukan proses komparasi antar dokumen, dan dapat melihat hasil dari komparasi yang telah dilakukan.

2.2 Dasar Teori

Berikut adalah beberapa dasar teori yang terkait dengan skripsi yang penulis lakukan :

2.2.1 Kemiripan

Kemiripan dalam Kamus Besar Bahasa Indonesia adalah hal(keadaan) mirip.

2.2.2 Algoritma Rabin-Karp

Algoritma Rabin-Karp adalah sebuah algoritma pencarian *string* yang dikembangkan oleh Richard M. Karp dan Michael O. Rabin pada tahun 1987 yang menggunakan *hashing* untuk menemukan *pattern string* dalam teks(*Rabin–Karp Algorithm - Wikipedia*, n.d.). Rabin dan karp mengusulkan algoritma pencocokan string yang bekerja dengan baik dalam praktik dan juga menggeneralisasi ke algoritma lain untuk masalah yang terkait, seperti pencocokan pola dua dimensi(Cormen et al., 2001). Pada algoritma ini mencocokkan nilai *hash* dari pola dengan nilai *hash* dari *substring* teks, dan apabila nilainya sama maka akan dilakukan pencocokan karakter(*Rabin-Karp Algorithm for Pattern Searching - GeeksforGeeks*, n.d.).

2.2.3 Hashing

Hashing adalah suatu cara untuk mentransformasikan sebuah string menjadi suatu nilai yang unik dengan panjang tertentu (*fixed-length*) yang berfungsi sebagai penanda string tersebut. Fungsi untuk menghasilkan nilai ini disebut fungsi *hash*, sedangkan nilai yang dihasilkan disebut nilai *hash*. Contoh sederhana *hashing* adalah (Firdaus, 2003):

Firdaus, Hari Munir, Rinaldi Rabin, Michael Karp, Richard

menjadi

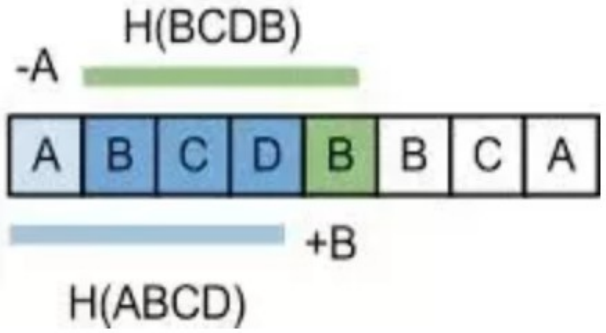
7864 Firdaus,Hari 9802 Munir, Rinaldi 1990 Rabin, Michael 8822 Karp, Richard

Contoh di atas adalah penggunaan *hashing* dalam pencarian pada *database*. Apabila tidak di-*hash*, pencarian akan dilakukan karakter per karakter pada nama-nama yang panjangnya bervariasi dan ada 26 kemungkinan pada setiap karakter. Namun pencarian akan menjadi lebih mangkus setelah di-*hash* karena hanya akan membandingkan empat digit angka dengan cuma 10 kemungkinan setiap angka.

Nilai *hash* pada umumnya digambarkan sebagai *fingerprint* yaitu suatu string pendek yang terdiri atas huruf dan angka yang terlihat acak (data biner yang ditulis dalam heksadesimal)(Firdaus, 2003).

2.2.4 Rolling Hash

Rolling Hash adalah salah satu metode hashing yang memberikan kemampuan untuk menghitung nilai hash tanpa mengulangi seluruh string(Wibowo & Hastuti, 2016). Berikut ini adalah contoh dari Rolling Hash :



Gambar 2.1: Rolling Hash

2.2.5 K-Gram

K-gram adalah rangkaian terms dengan panjang gram. Kebanyakan yang digunakan sebagai terms adalah kata. K-gram merupakan sebuah metode yang diaplikasikan untuk pembangkitan kata atau karakter. Metode ini digunakan untuk mengambil potongan-potongan karakter huruf sejumlah *gram* dari sebuah kata yang secara kontinuitas dibaca dari teks sumber hingga akhir dari dokumen.

Berikut ini adalah contoh K-grams dengan nilai k=5 :

- Teks : “A do run run run ,a do run run”
- kemudian dilakukan penghilangan spasi : “adorunrunrunadorunrun”
- sehingga dihasilkan rangkaian 5-grams yang diturunkan dari teks :“adoru dorun orunr runru unrun nrunr runru unrun nruna runad unado nador adoru dorun orunr runru unrun”(Prima Putra & Sularno, 2019).

2.2.6 Text Preprocessing

Tahap Text Preprocessing adalah tahapan di mana aplikasi melakukan seleksi data yang akan diproses pada setiap dokumen. Proses preprocessing ini meliputi Case Folding, Tokenizing, Filtering, dan Stemming (Tineges, 2021).

a) Case Folding

Peran Case Folding dibutuhkan dalam mengonversikan keseluruhan teks dalam dokumen menjadi suatu bentuk standar(biasanya huruf kecil atau lowercase).

b) Tokenizing

Tahap Tokenizing adalah tahap pemotongan string input berdasarkan tiap kata yang menyusunnya.

c) Filtering

Tahap Filtering adalah tahap mengambil kata-kata penting dari hasil token.

d) Stemming

Stemming merupakan suatu proses untuk menemukan kata dasar dari sebuah kata (Ismail, 2009).

2.2.7 Persamaan Dice's Similarity Coefficient

Pada algoritma Rabin-Karp, cara yang dapat digunakan dalam menghitung tingkat kesamaan antara dua pattern adalah dengan menggunakan Dice's Similarity Coefficients.

Mencari nilai kesamaan dengan cara menghitung nilai dari jumlah K-Gram dari hasil pencarian pada kedua dokumen, kemudian mencari nilai fingerprint yang didapatkan dari nilai K-Gram yang sama(Putra & Sujaini, 2015).

Rumus dalam mencari persentase kesamaan adalah sebagai berikut.

$$DSC = \frac{2|X \cap Y|}{|X| + |Y|}$$

Keterangan:

X : jumlah k-gram dari teks 1

Y : jumlah k-gram dari teks 2