Perancangan Aplikasi Deteksi Keberadaan Kesamaan Kalimat Sebagai Indikasi Penjiplakan Dengan Algoritma Hashing Berbasis N-Gram

Proposal Tugas Akhir

Oleh :

Adrian Fadillah Hidayat

115130008

PROGRAM STUDI INFORMATIKA

INSTITUT TEKNOLOGI INDONESIA

2017/2018

1. Latar Belakang

Suatu karya ilmiah dikatakan sebagai hasil penjiplakan apabila kutipan yang dilakukan tidak disertai penyebutan referensi secara benar. Sehingga dijumpai kalimat yang sama saja tidak berarti karya ilmiah tersebut dinyatakan sebagai hasil plagiat. Algoritma Winnowing [1] yang dibahas pada makalah ini adalah suatu cara untuk mendeteksi adanya kalimat – kalimat yang sama atau sering disebut juga sebagai problem *common subsequence* [2].

Aplikasi Dustball - *The Plagiarism Checker* yang dibuat oleh tim dari University of Maryland Dustball [3] menggunakan fasilitas mesin pencari dengan mencari kalimat yang diduga sebagai hasil penjiplakan dalam web, sedangkan Copyscape mendeteksi isi dari halaman web menurut alamat URL yang diisikan [4]. Kedua aplikasi tersebut mencurigai adanya penjiplakan berdasarkan urutan posisi kata dalam kalimat seperti penelitian di Universitas Gajah Mada (UGM) Yogyakarta dengan nama TESSY (*Test of Texts Similarity*) [5]. Urutan posisi kata juga dapat digunakan dalam pengaplikasian lain seperti pengecekan urutan kata dalam *spell checker* [6].

Pada tugas akhir ini kalimat yang sama dikenali melalui *fingerprint* dari dokumen [7,8]. Identifikasi penjiplakan dengan teknik *fingerprint* (sidik jari) akan merubah urutan kata dengan setiap panjang tertentu (*window*) menjadi suatu nilai yang dianggap sebagai sidik jari. Teknik *fingerprint* dapat mengenali frase yang dicurigai banyak dijiplak pada suatu dokumen teks meskipun telah sedikit mengalami perubahan dengan cara parafrase. Hal tersebut yang masih belum bisa dikenali dengan pendekatan urutan posisi kata dalam kalimat.

langkah – langkah untuk deteksi keberadaan kalimat sama yang menjadi indikasi penjiplakan dengan algoritma *Winnowing* sebagai algoritma *hashing* berbasis *n-gram* adalah pertama membuang karakter yang tidak *relevan* seperti spasi, kedua membentuk rangkaian N-Gram dari teks contoh N=5, ketiga melakukan fungsi *hash* pada setiap *N-Gram*, keempat memilih *fingerprint* dari hasil *hashing* dengan pembagian hasil *hash* berdasarkan satu nilai *window w*, dan kemudian dipilih nilai *hash* terkecil. Semisal $w = 4$

sehingga *window* yang dibentuk dari 4 nilai-nilai *hash* adalah sejumlah N *window*, kelima kemudian *fingerprint* yang dihasilkan adalah sejumlah N nilai *hash* dari N *window*, keenam lakukan perhitungan kesamaan (jumlah hash sama / total jumlah hash lalu * 100%) [9].

2. Rumusan Masalah

Berdasarkan latar belakang diatas maka dapat dirumuskan masalah sebagai berikut :

1. Bagaimana perancangan aplikasi deteksi keberadaan kesamaan kalimat sebagai indikasi penjiplakan dengan Algoritma Hashing berbasis N-Gram?
2. Bagaimana cara deteksi keberadaan kesamaan kalimat sebagai indikasi penjiplakan dengan Algoritma Hashing berbasis N-Gram?

3. Tujuan

Tujuan dari penulisan Tugas Akhir ini adalah untuk merancang aplikasi deteksi keberadaan kesamaan kalimat sebagai indikasi penjiplakan dengan Algoritma Hashing berbasis N-Gram. Merancang *interface* aplikasi, implementasi, dan evaluasi deteksi keberadaan kesamaan kalimat sebagai indikasi penjiplakan dengan Algoritma Hashing berbasis N-Gram, sasaran aplikasi tersebut adalah tugas mahasiswa berupa artikel, karya tulis, makalah atau tugas akhir.

4. Ruang Lingkup

Mengingat ruang lingkup yang akan dibahas begitu luas, maka dibutuhkan batasan masalah sebagai berikut :

1. Data yang akan di olah pada aplikasi deteksi keberadaan kesamaan kalimat sebagai indikasi penjiplakan dengan Algoritma Hashing berbasis N-Gram ini adalah untuk mempermudah dosen kampus Institut Teknologi Indonesia dalam melakukan deteksi indikasi penjiplakan pada artikel, karya tulis, makalah atau tugas akhir.

2. Aplikasi deteksi keberadaan kesamaan kalimat sebagai indikasi penjiplakan dengan Algoritma Hashing berbasis N-Gram ini dibangun menggunakan VB (Visual Basic), dan XML.

5. Metodologi Penelitian

Metodologi penelitian yang digunakan dalam menyelesaikan tugas akhir ini terdiri dari:

1. Studi literatur

   Membaca buku-buku yang berkaitan dengan topik pada tugas akhir.

2. Observasi

   Pengamatan yang dilakukan secara langsung pada aplikasi yang serupa dengan topik tugas akhir. Observasi merupakan salah satu teknik pengumpulan data yang cukup akurat dan efektif untuk mempelajari suatu sistem yang sedang berjalan.

3. Analisa

   Analisa terhadap algoritma dan sumber data. Selain itu juga diperlukan analisa kebutuhan untuk perancangan aplikasi pada topik tugas akhir.

4. Pemodelan

   Model yang digunakan untuk desain sistem pada topik tugas akhir adalah Unified modeling language (UML).

5. Implementasi

   Pada tahap ini akan diimplementasikan menggunakan VB (Visual Basic).

6. Pengujian dan evaluasi

   Pada tahap ini dilakukan pengujian dan evaluasi terhadap aplikasi yang dibangun.

6. Jadwal Pelaksanaan

Kegiatan	Maret				April				Mei			
	1	2	3	4	1	2	3	4	1	2	3	4
Penulisan Proposal												
Pengumpulan Data												
Analisis Data												
Pemodelan												
Implementasi												
Testing & evaluasi												
Penulisan Laporan												

7. Teori Dasar Utama

1. Algoritma *Hashing* berbasis *N-Gram*

Algoritma adalah langkah-langkah yang disusun secara tertulis dan berurutan untuk menyelesaikan suatu masalah. *Hash* adalah suatu teknik "klasik" dalam Ilmu Komputer yang banyak digunakan dalam praktek secara mendalam. *Hash* merupakan suatu metode yang secara langsung mengakses *record-record* dalam suatu tabel dengan melakukan transformasi aritmatik pada *key* yang menjadi alamat dalam tabel tersebut. *Key*merupakan suatu input dari pemakai di mana pada umumnya berupa nilai atau string karakter.

Fungsi *Hash* adalah suatu fungsi yang mengubah *key* menjadi alamat dalam tabel. Fungsi *Hash* memetakan sebuah *key* ke suatu alamat dalam tabel. Idealnya, *key-key* yang berbeda seharusnya dipetakan ke alamat-alamat yang berbeda juga. Pada kenyataannya, tidak ada fungsi *Hash* yang sempurna. Kemungkinan besar yang terjadi adalah dua atau lebih *key* yang berbeda dipetakan ke alamat yang sama dalam table.

N-Gram adalah penghitungan jarak berbasis cepat dan mudah untuk menghitung, maka metode ini memerlukan sedikit waktu komputasi.

8. Daftar Pustaka

[1] S.Schleimer, D.Wilkerson, dan A.Aiken, "Winnowing: Local Algorithms for Document Fingerprinting", Proc. of the ACM SIGMOD Intl. Conf. on Management of Data, pp 76–85, 2003.

[2] J.Oetsch, J.Pührer, M.Schwengerer, dan H.Tompits, "The System Kato: Detecting Cases of Plagiarism for Answer-Set Programs", Theory and Practice of Logic Programming, 10(4-6), pp 759-775, 2010.

[3] B.Klug, "The Plagiarism Checker", 2002, http://www.dustball.com/cs/plagiarism.checker/, diakses tanggal 10 Januari 2011.

[4] G.Greenspan, "Copyscape", 2006, http://copyscape.com, diakses tanggal 02 Maret 2017.

[5] Editor The Jakarta Post, Tracing Plagiarism Makes Cheating Hard", 2008, http://www.thejakartapost.com/news/2008/12/26/tracing-plagiarism-makes-cheating- hard.html, diakses tanggal 02 Maret 2017.

[6] T. Ahmad, N. Jatmiko, dan M. Safii, "Perancangan dan Pembuatan Aplikasi SMS Spell Checker Berbahasa Indonesia dengan Menggunakan Algoritma Naive Bayes pada Mobile Device", Kursor, Vol.4 No.2 tahun 2008

[7] A.Kurniawati, dan I.W.S.Wicaksana, "Perbandingan Pendekatan Deteksi Plagiarism Dokumen dalam Bahasa Inggris", Seminar Ilmiah Nasional Komputer dan Sistem Inteligen KOMMIT 2008, Universitas Gunadarma, 2008.

[8] Editor Berita Institut Teknologi Bandung, "Pernyataan Sikap ITB Terhadap Plagiarisme Mochammad Zuliansyah", 2010, url http://www.itb.ac.id/news/2813.xhtml, diakses tanggal 02 Maret 2017.

[9] Purwitasari, D., Kusmawan, P.T, & Yuhana, U.L "Deteksi Keberadaan Kalimat Sama sebagai Indikasi Penjiplakan dengan Algoritma Hashing Berbasis N-Gram". Jurnal Ilmiah Kursor Vol 6, No 1, Jan 2011.