

DETEKSI PLAGIARISME TINGKAT KEMIRIPAN JUDUL SKRIPSI DENGAN ALGORITMA WINNOWING

Nur Alamsyah

Fakultas Teknologi Informasi

Universitas Islam Kalimantan Muhammad Arsyad Al Banjari Banjarmasin

uniskalam@gmail.com

ABSTRAK

Untuk menentukan kemiripan teks judul skripsi yang di ajukan dengan judul-judul skripsi yang telah ada sebelumnya maka diperlukan penerapan suatu algoritma untuk menentukan kemiripan teks, salah satunya algoritma untuk menentukan kemiripan teks ialah algoritma WInnowing.

Algoritma WInnowing adalah sebuah cara yang digunakan untuk mendeteksi kesamaan kata/kalimat (*common subsequence*) dalam dua atau lebih teks yang dibandingkan. Dua teks diketahui memiliki kesamaan kata/kalimat apabila di dalam dokumen tersebut dijumpai *fingerprint*, *fingerprint* inilah yang akan dijadikan dasar pembandingan antara teks, algoritma ini akan mencari *fingerprint* (kesamaan di dalam dua teks) dengan cara mengubah *n-gram* dari sebuah teks ke dalam bentuk nilai angka yang disebut dengan nilai *hash*, teknik untuk mencari nilai tersebut adalah *Hashing*.

Dengan sistem ini, Ketua Program Studi atau Koordinator Tugas Akhir nantinya hanya memasukkan judul yang akan diajukan mahasiswa, kemudian sistem akan mengecek secara otomatis dan menampilkan hasilnya. Hasil tersebut bisa dijadikan sebagai pertimbangan dalam pengambilan keputusan dan dapat menentukan apakah diterima atau ditolak judul skripsi tersebut.

Kata Kunci : *Algoritma WInnowing, fingerprint, n-gram, Hashing*

PENDAHULUAN

Bagi mahasiswa tingkat akhir, pencarian judul skripsi merupakan suatu hal yang sangat penting dan dapat dikatakan sulit namun dapat juga dikatakan mudah, dikatakan mudah karena dengan semakin cepatnya akses internet saat ini, mahasiswa dapat dengan mudah menemukan judul-judul skripsi untuk dijadikan acuan atau referensi, menjadi sulit karena dengan banyaknya judul-judul skripsi yang mudah diakses ada kemungkinan memiliki judul yang mirip bahkan sama sehingga bisa terjebak kedalam plagiarisme.

Seiring berjalannya waktu telah banyak judul-judul skripsi yang telah diterima dan tersimpan di akademis. Dalam pengajuan judul tidak sedikit mahasiswa yang berulang kali mengajukan judul skripsi, hal ini dikarenakan

mahasiswa tersebut tidak mengetahui judul yang diajukan telah ada sebelumnya karena penentuan diterima atau ditolaknya judul-judul yang diajukan oleh mahasiswa masih dilakukan secara manual yaitu dengan mengecek satu persatu berdasarkan daftar judul-judul skripsi angkatan yang telah lulus yang tersimpan di file Microsoft excel.

Judul skripsi yang sama, tidak menutup kemungkinan isi dari skripsi tersebut juga sama namun tidak menutup kemungkinan juga isi skripsi berbeda walaupun judul skripsi sama sehingga mahasiswa bingung dalam menentukan judul skripsinya. Dalam menentukan diterima atau tidaknya sebuah judul skripsi yang sekarang ini dilakukan adalah dengan mengecek atau membandingkan judul tersebut dengan daftar judul-judul yang

telah ada, tentu saja hal ini akan membutuhkan waktu yang cukup lama belum lagi jika pengaju judul skripsi berjumlah banyak, tidak menutup kemungkinan ada judul yang terulang atau sama dengan judul-judul yang tersimpan di akademis bahkan waktu yang dibutuhkan juga akan semakin lama.

Di samping itu, pengelolaan daftar judul-judul yang telah ada juga masih manual sehingga tidak menutup kemungkinan data akan hilang dan sulit dalam mendistribusikan informasi tersebut kepada mahasiswa. Oleh karena itu, diperlukan suatu sistem yang mampu mendeteksi persentase kemiripan teks judul secara cepat dan tepat dan mampu menyediakan informasi tersebut kepada seluruh mahasiswa.

RUMUSAN MASALAH

Berdasarkan uraian dari latar belakang, maka dapat dirumuskan permasalahannya yaitu dalam menentukan diterima atau tidaknya sebuah judul skripsi masih dilakukan dengan cara manual yakni dengan mengecek atau membandingkan judul tersebut dengan daftar judul-judul yang telah ada, tidak menutup kemungkinan data akan hilang dan sulit dalam mendistribusikan informasi tersebut kepada mahasiswa.

TUJUAN PENELITIAN

Tujuan penelitian ini adalah untuk mengetahui kemampuan algoritma winnowing dalam mendeteksi tingkat plagiarisme kemiripan judul skripsi dengan menggunakan perubahan nilai n-gram dan window dari algoritma winnowing untuk menghasilkan persentase kemiripan judul skripsi.

TARGET LUARAN DAN MANFAAT PENELITIAN

Target Luaran pada penelitian ini adalah untuk memberikan kontribusi kepada Koordinator Tugas Akhir atau Ketua Program Studi dalam menentukan tingkat plagiarisme pada judul-judul skripsi yang diajukan mahasiswa agar terhindar dari plagiarisme.

Sedangkan manfaat dari penelitian adalah untuk memudahkan mendeteksi tingkat plagiarisme pada judul-judul skripsi yang ada difakultas dengan judul skripsi yang diajukan dengan menggunakan algoritma winnowing.

METODE PENELITIAN

Metode yang digunakan adalah dengan metode eksperimen, dengan tahapan penelitian sebagai berikut:

1. Pengumpulan Data

Data yang diperlukan berupa dokumen teks yaitu judul-judul skripsi mahasiswa yang telah lulus kemudian disimpan pada Microsoft excel.

2. Pengolahan Data Menggunakan Algoritma Winnowing

Winnowing adalah algoritma yang digunakan untuk melakukan proses document *fingerprinting* (Schleimer, Wilkerson, & Aiken, 2003: 4). Document *fingerprinting* merupakan metode yang digunakan untuk mendeteksi keakuratan salinan antar dokumen atau hanya sebagian teks saja. Prinsip kerja dari metode *document fingerprinting* ini adalah dengan menggunakan teknik *hashing*. Teknik *hashing* adalah sebuah fungsi yang mengkonversi setiap string menjadi bilangan.

Algoritma *winnowing* melakukan penghitungan nilai-nilai *hash* dari setiap *k-gram*, untuk mencari nilai *hash* selanjutnya digunakan fungsi *rolling hash*. Kemudian dibentuk *window* dari nilai-nilai *hash* tersebut. Dalam setiap *window* dipilih nilai *hash* minimum. Jika ada lebih dari satu *hash* dengan nilai minimum, dipilih nilai *hash* yang paling kanan. Kemudian semua nilai *hash* terpilih disimpan untuk dijadikan *fingerprint* dari suatu dokumen.

Input dari proses document *fingerprinting* adalah file teks. Kemudian outputnya akan berupa sekumpulan nilai *hash* yang disebut *fingerprint*. *Fingerprint* ini yang akan dijadikan dasar pembandingan kesamaan antara teks yang telah dimasukkan.

Syarat dari algoritma deteksi plagiarisme (Schleimer, Wilkerson, & Aiken, 2003: 2) yaitu;

- a. **Whitespace insensitivity**, yaitu pencocokan teks *file* seharusnya tidak terpengaruh oleh spasi, jenis huruf kapital, tanda baca dan sebagainya.
- b. **Noise surpression**, menghindari pencocokan teks *file* dengan panjang kata yang terlalu kecil atau kurang relevan dan bukan merupakan kata yang umum digunakan.
- c. **position independence**, yaitu pencocokan teks *file* seharusnya tidak bergantung pada posisi kata-kata sehingga kata dengan urutan posisi berbeda masih dapat dikenali jika terjadi kesamaan.

Winnowing telah memenuhi syarat-syarat tersebut dengan cara membuang seluruh karakter-karakter yang tidak relevan misal: tanda baca, spasi dan juga karakter lain, sehingga nantinya hanya karakter-karakter yang berupa huruf atau angka yang akan diproses lebih lanjut.

Langkah-langkah dalam penerapan Algoritma *Winnowing* adalah sebagai berikut (Sariyanti Astutik, Andharini Dwi Cahyani, Mochammad Kautsar Sophan, 2014):

Langkah I *Pembuangan Karakter yang Tidak Relevan.*

Yaitu penghapusan tanda baca, spasi dan symbol-simbol seperti =, #, %, &, (,), -, _, \$, @, !, /, ", seperti contoh dibawah ini:

Hello !!! Saya Alam, Apa kabarmu?

Akan dirubah menjadi

hellosayaalamapakabarmu

Langkah II *Pembentukan Rangkaian n-gram.*

Yaitu dengan cara membentuk rangkaian karakter sepanjang *n* dari hasil pembuangan karakter yang tidak relevan, dari

teks diatas yang telah dibersihkan dengan ukuran *k*, misal ukuran *k* = 5

hello ellos llosa losay osaya sayaa ayaal
yaala aalam alama lamap amapa mapak
apaka pakab akaba kabar abarm barmu

Langkah III *Perhitungan Fungsi Hash untuk Setiap n-gram.*

Yaitu melakukan perhitungan-perhitungan nilai-nilai *hash* dari setiap *gram*, fungsi yang digunakan untuk menghasilkan nilai *hash* dari rangkaian *gram* dalam algoritma *Winnowing* adalah *rolling hash*.

Rolling Hash adalah suatu cara untuk mentransformasi 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*.

Fungsi hash $H_{(c1..ck)}$ didefinisikan sebagai berikut:

$$H_{(ck)} = c_1 * b^{(k-1)} + c_2 * b^{(k-2)} + ... + c_k * b^{(k-k)}$$

Keterangan :

c = nilai ascii karakter

b = basis (bilangan prima)

k = banyak karakter

hasil *rolling hash* dari kalimat diatas adalah:

6462 6498 6726 6782 6846 6782 6420 6826
6126 6238 6492 6266 6538 6294 6576 6178
6376 6122 6270

Langkah IV *Pembentukan Window dari Nilai Hash.*

Pembentukan *nilai hash* dari window dengan ukuran *w* = 4 yaitu :

W-1 : { 6462 6498 6726 6782 }

W-2 : { 6498 6726 6782 6846 }

W-3 : { 6726 6782 6846 6782 }

- W-4 : { 6782 6846 6782 **6420** }
- W-5 : { 6846 6782 **6420** 6826 }
- W-6 : { 6782 6420 6826 **6126** }
- W-7 : { 6420 6826 **6126** 6238 }
- W-8 : { 6826 **6126** 6238 6492 }
- W-9 : { **6126** 6238 6492 6266 }
- W-10 : { **6238** 6492 6266 6538 }
- W-11 : { 6492 **6266** 6538 6294 }
- W-12 : { **6266** 6538 6294 6576 }
- W-13 : { 6538 6294 6576 **6178** }
- W-14 : { 6294 6576 **6178** 6376 }
- W-15 : { 6576 6178 6376 **6122** }
- W-16 : { 6178 6376 **6122** 6270 }

Langkah V Pemilihan Fingerprint dari Setiap Window

Langkah terakhir yaitu memilih nilai terkecil dari setiap window untuk dijadikan *fingerprint*, hasil dari *nilai fingerprint*nya sebagai berikut;

- [6462, 1], [6498, 2], [6726, 3], [6420, 4], [6126, 6], [6238, 10], [6266, 12], [6178, 14], [6122,15]

Persamaan Jaccard Coefficient.

Nilai fingerprint yang dibentuk dari algoritma winnowing digunakan untuk mengukur prosentase kemiripan teks pada persamaan *Jaccard Coefficient*.

Persamaan *Jaccard Coefficient* digunakan untuk menghitung kemiripan (*similarity*) dari kumpulan kata-kata yang telah dihitung nilai *hash* nya. Berikut ini rumus persamaan *Jaccard Coefficient*.

$$\text{Similarity} = \frac{\text{Jumlah_fingerprint_sama}}{\text{Total_seluruh_fingerprint}} \times 100$$

ANALISIS HASIL DAN PEMBAHASAN

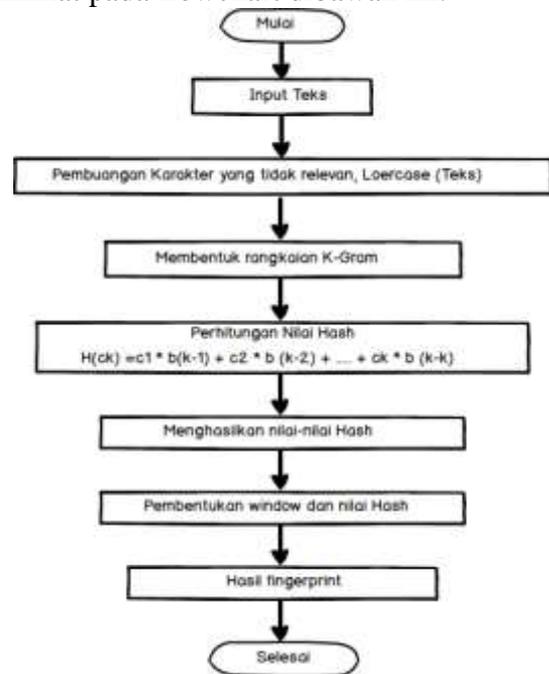
Analisis Algoritma Winnowing

Algoritma Winnowing merupakan salah satu metode document *fingerprinting* yang digunakan untuk mendeteksi kemiripan antar teks dokumen dengan menggunakan teknik *hashing*. Algoritma ini dipilih karena

Winnowing merupakan salah satu algoritma terbaik untuk mendapatkan nilai *similarity* antar teks dokumen baik dalam segi akurasi ataupun performansi.

Penerapan Algoritma Winnowing dalam sebuah system membutuhkan beberapa inputan yaitu dokumen teks berekstensi .txt, jumlah karakter, jumlah Gram dan jumlah Window, sedangkan output yang akan dihasilkan berupa Persentase kemiripan dari teks dokumen yang dibandingkan.

Tahapan-tahapan Algoritma dapat dilihat pada flowchart dibawah ini:



Flowchart Algoritma Winnowing

Berdasarkan gambar flowchart diatas, menjelaskan tahapan Algoritma Winnowing mulai dari input teks (processing), kemudian pembuangan karakter yang tidak relevan seperti spasi, tanda baca dan symbol-simbol. Setelah itu pembagian kata untuk membentuk nilai gram, kemudian menghitung nilai-nilai hash melalui proses tolling hash akan menghasilkan nilai hash yang dibag menurut window dan sampai pada proses menghasilkan nilai fingerprint.

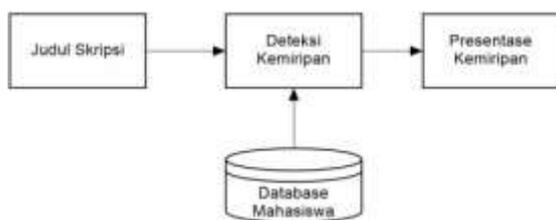
Tahapan tersebut dilakukan pada dokumen pertama, setelah itu dilakukan

tahapan yang sama untuk dokumen kedua sehingga menghasilkan dua nilai-nilai fingerprints dari kedua dokumen. Setelah itu proses dilanjutkan dengan perhitungan persentasi kemiripan dari kedua dokumen berdasarkan nilai fingerprints yang telah diperoleh. Persentasi kemiripan dihitung menggunakan metode kesamaan nilai fingerprint (*Jaccard similarity coefficient*).

Jadi dapat disimpulkan Input dari proses document fingerprinting adalah file teks. Kemudian output-nya berupa sekumpulan nilai hash yang disebut fingerprint. Fingerprint inilah yang akan dijadikan dasar pembandingan antara file-file teks yang telah dimasukkan.

Analisis Deteksi Kemiripan Judul Skripsi dengan Algoritma Winnowing

Proses deteksi kemiripan judul dapat dilakukan dengan menerapkan berbagai metode untuk menghasilkan presentasi kemiripan. Tujuan utama dari deteksi kemiripan judul adalah untuk menentukan presentase kemiripan judul satu dengan judul lainnya.



Skema deteksi kemiripan judul skripsi

Deteksi kemiripan judul skripsi dapat dideteksi dengan salah satu algoritma yaitu dengan algoritma winnowing dimana hasil presentase kemiripan judul ditampilkan. Proses deteksi kemiripan dilakukan dengan menampilkan presentase dengan menggunakan Persamaan *Jaccard Coeficient* pada hasil Fingerprint.

Adapun langkah-langkah deteksi kemiripan judul skripsi adalah sebagai berikut:

1. Masukan judul skripsi yang akan dilihat tingkat presentase nya dengan judul skripsi yang sudah ada sebelumnya.
2. Memasukan nilai n-gram, untuk memebentuk rangkaian gram pada judul yang dimasukan dan judul yang dibandingkan.
3. Masukan nilai Window, untuk menentukan pembentukan window dari nilai Hash.
4. Proses deteksi kemiripan judul skripsi dengan algoritma winnowing yang mana tujuanya dari proses deteksi kemiripan judul tersebut ada untuk menampilkan tingkat presentase dengan judul-judul yang sudah ada sebelumnya. Berdasarkan analisa di atas, dapat disimpulkan bahwa proses deteksi kemiripan judul skripsi berdasarkan algoritma winnowing menghasilkan presentase kemiripan dengan kumpulan judul skripsi yang sudah ada, karena semakin tinggi presentase kemiripan maka akan semakin mirip dengan judul yang dibandingkan.

Implementasi Algoritma Winnowing Terhadap Kemiripan Judul Skripsi

Tahapan-tahapan untuk mengimplementasikan Algoritma Winnowing untuk mendeteksi presentase plagiarisme pada judul skripsi adalah sebagai berikut;

1. **Ambil 2 judul skripsi yang akan dibandingkan.**

Teks Judul 1:

“APLIKASI LAYANAN KEPENDUDUKAN ELEKTRONIK PADA DINAS KEPENDUDUKAN DAN PENCATATAN SIPIL KOTA BANJARMASIN”

Teks Judul 2:

“SISTEM INFORMASI MANAJEMEN LAYANAN KEPENDUDUKAN PADA DISCAPIL KOTA BANJARMASIN”

2. Pembuangan Karakter yang Tidak Relevan

Hapus semua huruf yang bukan A-Z, a-z, 0-9. Kemudian ubah menjadi huruf kecil semua.

Teks Judul 1 akan terbentuk menjadi:

aplikasilayanankependudukanelektronikp
adadinaskependudukanpencatatan
sipilkotabanjarmasin

Teks Judul 2 akan terbentuk menjadi :

sisteminformasimanajemenlayanankepend
udukanpadadiscapilkotabanjarmasin

3. Pembentukan Rangkaian n-gram

Buat *n-gram* untuk masing-masing judul kalimat. Jumlah data pengelompokan *n-gram* ini bisa dimulai dari 2, 3, 5, 7 dan seterusnya misal *n-gram*=3.

Pada teks Judul 1 akan terbentuk 87 rangkaian n-gram yaitu:

apl pli lik ika kas asi sil ila lay aya yan
ana nan ank nke kep epe pen end ndu
dud udu duk uka kan ane nel ele lek ekt
ktr tro ron oni nik ikp kpa pad ada dad
adi din ina nas ask ske kep epe pen end
ndu dud udu duk uka kan and nda dan
anp npe pen enc nca cat ata tat ata tan
ans nsi sip ipi pil ilk lko kot ota tab aba
ban anj nja jar arm rma mas asi sin

Sedangkan pada teks Judul 2 terbentuk 66 rangkaian n-gram yaitu:

sis ist ste tem emi min inf nfo for orm
rma mas asi sim ima man ana naj aje
jem eme men enl nla lay aya yan ana nan
ank nke kep epe pen end ndu dud udu
duk uka kan anp npa pad ada dad adi
dis isc sca cap api pil ilk lko kot ota tab
aba ban anj nja jar arm rma mas asi sin

4. Perhitungan Fungsi Hash untuk Setiap n-gram

Buat Rolling Hash untuk masing-masing N-Gram , Perhitungan nilai hash pada rangkaian n-gram pada teks judul 1 bagian pertama “apl” dengan nilai basis (b) = 2, panjang rangkaian ngram(n) = 3

$$H_{(apl)} = \text{asci}_{(a)} * 2^3 + \text{asci}_{(p)} * 2^2 + \text{asci}_{(l)} * 2^1$$

$$= 97 * 8 + 112 * 4 + 108 * 2$$

$$= 1440$$

Hasil semua perhitungan nilai *hash* pada teks judul 1 yaitu:

1440 1538 1498 1462 1474 1446 1556 1466
1494 1454 1576 1410 1488 1430 1510 1484
1458 1520 1448 1514 1468 1570 1482 1558
1464 1418 1500 1442 1482 1468 1548 1606
1576 1538 1514 1492 1498 1484 1370 1388
1386 1440 1474 1498 1450 1550 1484 1458
1520 1448 1514 1468 1570 1482 1558 1464
1416 1474 1408 1440 1530 1520 1446 1470
1412 1434 1548 1434 1536 1446 1550 1564
1498 1532 1486 1514 1532 1546 1512 1362
1392 1428 1498 1464 1450 1542 1490 1446
1560

Kemudian rangkaian n-gram pada teks judul 2 bagian pertama “sis” dengan nilai basis (b) = 2, panjang rangkaian ngram(n) = 3

$$H_{(sis)} = \text{asci}_{(s)} * 2^3 + \text{asci}_{(i)} * 2^2 + \text{asci}_{(s)} * 2^1$$

$$= 115 * 8 + 105 * 4 + 115 * 2$$

$$= 1570$$

Hasil semua perhitungan nilai *hash* pada teks judul 2 yaitu:

1570 1532 1586 1550 1454 1512 1484 1510
1488 1562 1542 1490 1446 1558 1470 1480
1410 1480 1402 1470 1446 1496 1464 1506
1494 1454 1576 1410 1488 1430 1510 1484
1458 1520 1448 1514 1468 1570 1482 1558
1464 1440 1522 1484 1370 1388 1386 1450
1498 1510 1404 1434 1532 1486 1514 1532
1546 1512 1362 1392 1428 1498 1464 1450
1542 1490 1446 1560

5. Pembentukan Window dari Nilai Hash

Kelompokkan (windowing) untuk masing-masing hasil *hash*, langkahnya mirip seperti *n-gram*. Pembentukan window dari hasil perhitungan nilai *hash* dengan ukuran lebar window (w) = 3 pada teks judul 1 yaitu:

W-1 : {1440 1538 1498}	W-61 : {1530 1520 1446}
W-2 : {1538 1498 1462}	W-62 : {1520 1446 1470}
W-3 : {1498 1462 1474}	W-63 : {1446 1470 1412}
W-4 : {1462 1474 1446}	W-64 : {1470 1412 1434}
W-5 : {1474 1446 1556}	W-65 : {1412 1434 1548}
W-6 : {1446 1556 1466}	W-66 : {1434 1548 1434}
W-7 : {1556 1466 1494}	W-67 : {1548 1434 1536}
W-8 : {1466 1494 1454}	W-68 : {1434 1536 1446}
W-9 : {1494 1454 1576}	W-69 : {1536 1446 1550}
W-10 : {1454 1576 1410}	W-70 : {1446 1550 1564}
W-11 : {1576 1410 1488}	W-71 : {1550 1564 1498}
W-12 : {1410 1488 1430}	W-72 : {1564 1498 1532}
W-13 : {1488 1430 1510}	W-73 : {1498 1532 1486}
W-14 : {1430 1510 1484}	W-74 : {1532 1486 1514}
W-15 : {1510 1484 1458}	W-75 : {1486 1514 1532}
W-16 : {1484 1458 1520}	W-76 : {1514 1532 1546}
W-17 : {1458 1520 1448}	W-77 : {1532 1546 1512}
W-18 : {1520 1448 1514}	W-78 : {1546 1512 1362}
W-19 : {1448 1514 1468}	W-79 : {1512 1362 1392}
W-20 : {1514 1468 1570}	W-80 : {1362 1392 1428}
W-21 : {1468 1570 1482}	W-81 : {1392 1428 1498}
W-22 : {1570 1482 1558}	W-82 : {1428 1498 1464}
W-23 : {1482 1558 1464}	W-83 : {1498 1464 1450}
W-24 : {1558 1464 1418}	W-84 : {1464 1450 1542}
W-25 : {1464 1418 1500}	W-85 : {1450 1542 1490}
W-26 : {1418 1500 1442}	W-86 : {1542 1490 1446}
W-27 : {1500 1442 1482}	W-87 : {1490 1446 1560}

Kemudian pada teks judul 2 hasilnya yaitu;

W-1 : {1570 1532 1586}	W-49 : {1498 1510 1404}
W-2 : {1532 1586 1550}	W-50 : {1510 1404 1434}
W-3 : {1586 1550 1454}	W-51 : {1404 1434 1532}
W-4 : {1550 1454 1512}	W-52 : {1434 1532 1486}
W-5 : {1454 1512 1484}	W-53 : {1532 1486 1514}
W-6 : {1512 1484 1510}	W-54 : {1486 1514 1532}
W-7 : {1484 1510 1488}	W-55 : {1514 1532 1546}
W-8 : {1510 1488 1562}	W-56 : {1532 1546 1512}
W-9 : {1488 1562 1542}	W-57 : {1546 1512 1362}
W-10 : {1562 1542 1490}	W-58 : {1512 1362 1392}
W-11 : {1542 1490 1446}	W-59 : {1362 1392 1428}
W-12 : {1490 1446 1558}	W-60 : {1392 1428 1498}
W-13 : {1446 1558 1470}	W-61 : {1428 1498 1464}
W-14 : {1558 1470 1480}	W-62 : {1498 1464 1450}
W-15 : {1470 1480 1410}	W-63 : {1464 1450 1542}
W-16 : {1480 1410 1480}	W-64 : {1450 1542 1490}
W-17 : {1410 1480 1402}	W-65 : {1542 1490 1446}
W-18 : {1480 1402 1470}	W-66 : {1490 1446 1560}

6. Pemilihan Fingerprint dari Setiap Window

Ambil angka terkecil dari masing-masing window. Untuk selanjutnya angka terkecil ini diistilahkan *Fingerprints*. Pemilihan *nilai fingerprint* dari hasil pembentukan window pada tahap sebelumnya adalah ;

nilai fingerprint pada teks judul 1 adalah **78 window**

nilai fingerprint pada teks judul 2 adalah **66 window**

Fingerprints dari kelompok (window) kalimat teks judul 1 dengan fingerprints (window) kalimat teks judul 2 inilah yang nanti akan dihitung tingkat koefisien plagiarisme-nya

7. Persamaan Jaccard Coefficient

Perhitungan kesamaan dengan menggunakan persamaan *jaccard coefficient* yaitu;

$$\text{Similarity (kemiripan)} = \frac{65}{88} * 100\% = 73.86 \%$$

Jumlah Fingerprints pada teks judul 1 = 87

Jumlah Fingerprints teks judul 2 = 66

Union (Gabungan) Fingerprints 1 dan 2 = 153

Intersection (fingerprints yang sama) = 65

(Union - Intersection) = 88

Prosentase Plagiarisme

$$\text{Koefisien Jaccard} = \left(\frac{\text{Intersection}}{\text{Union} - \text{Intersection}} \right) * 100$$

$$(65/88) * 100 = \mathbf{73.86 \%$$

Jadi dapat disimpulkan dari kedua judul diatas memiliki tingkat plagiarisme sebesar **73.86 %**

Pengujian Metode Winnowing

Untuk menentukan jenis kesamaan antara dokumen yang diuji dalam kasus ini adalah teks judul skripsi, ada 5 jenis penilaian persentase similarity (Mutiara-Agustina, 2008):

1. **0%** : Hasil uji 0% berarti kedua dokumen tersebut benar-benar berbeda baik dari segi isi dan kalimat secara keseluruhan
2. **< 15%**: Hasil uji 1-14% berarti kedua dokumen tersebut hanya mempunyai sedikit kesamaan
3. **15-50%**: Hasil uji 15-50% berarti menandakan dokumen tersebut termasuk plagiat tingkat sedang
4. **>50%** : Hasil uji lebih dari 51-99% berarti dapat dikatakan bahwa dokumen tersebut mendekati plagiarisme
5. **100%** : Hasil uji 100% menandakan bahwa dokumen tersebut adalah plagiat karena dari awal sampai akhir mempunyai isi yg sama persis.

Proses pengujian dilakukan mulai dari pegujian jumlah gram, jumlah window dan basis bilangan prima. Dapat dilihat pada tabel dibawah ini;

Tabel hasil pengujian deteksi tingkat plagiarism pada judul skripsi

No	n-gram (n)	Window(w)	Nilai basis	Presentase %
1	3	3	2	73.86 %
2	5	3	2	36.7 %
3	7	3	2	29.46 %
4	3	5	2	73.26 %
5	5	5	2	36.79 %
6	7	5	2	29.36 %
7	3	7	2	55.91 %
8	5	7	2	30.56 %
9	7	7	2	23.42 %
10	3	9	2	50%
11	5	9	2	31.73 %
12	7	9	2	19.82 %

Berdasarkan dari tabel hasil Pengujian diatas dalam menentukan tingkat plagiarisme dengan menggunakan n-gram, window dan bilangan prima menghasilkan **73.86 % tingkat plagiarisme tinggi** artinya mendekati plagiarisme dengan n-gram = 3 , window = 3 dan bilangan prima = 2 sedangkan dengan menggunakan n-gram = 7 , window = 9 dan bilangan prima = 2 menghasilkan **19.82 % artinya hanya mempunyai tingkat plagiarisme sedang.**

PENUTUP

Kesimpulan

Berdasarkan hasil pengujian teks judul skripsi dengan menggunakan algoritma winnowing dapat ditarik kesimpulan sebagai berikut;

1. Penerapan Algoritma Winnowing dalam sebuah sistem membutuhkan beberapa inputan yaitu dokumen teks berekstensi .txt, jumlah karakter, jumlah Gram dan jumlah Window, sedangkan output yang akan dihasilkan berupa Persentase kemiripan dari teks dokumen yang dibandingkan.
2. Setelah membandingkan teks judul 1 dengan teks judul 2 menghasilkan Jumlah Fingerprints pada teks judul 1 = 87, Jumlah Fingerprints teks judul 2 = 66, Union (Gabungan) Fingerprints 1 dan 2 = 153, Intersection (fingerprints yang sama) = 65, (Union - Intersection) = 88, Prosentase Plagiarisme, Koefisien Jaccard = $(\text{Intersection} / (\text{Union} - \text{Intersection})) * 100$ jadi $(65/88) * 100 = 73.86 \%$
3. Berdasarkan hasil Pengujian dalam menentukan tingkat plagiarisme dengan menggunakan n-gram, window, bilangan prima menghasilkan **73.86 % tingkat plagiarisme tinggi** artinya mendekati plagiarisme dengan n-gram = 3 , window = 3 dan bilangan prima = 2 sedangkan dengan menggunakan n-gram = 3 , window = 3 dan bilangan prima = 2 menghasilkan **19.82 % artinya hanya mempunyai tingkat plagiarisme sedang.**

Saran

Deteksi Plagiarisme Tingkat Kemiripan Judul skripsi dengan Algoritma Winnowing ini tidak terlepas dari kekurangan dan kelemahan. Oleh Karena itu perlu perbaikan dan perkembangan sistem selanjutnya, maka saran yang diberikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Untuk mendeteksi plagiarisme atau kemiripan teks dokumen dengan metode *fingerprinting* dapat dilakukan dengan algoritma lainnya yang berkaitan pada

bidang text mining seperti Algoritma Rabin Karp atau Algoritma Manber.

2. Dapat dibuatkan perbandingan antar algoritma mana algoritma terbaik untuk mendeteksi plagiarisme pada dokumen teks atau judul skripsi.
3. Dapat dibuatkan aplikasi deteksi plagiarisme secara online yang sudah memiliki database judul-judul skripsi mahasiswa yang telah lulus, kemudian mahasiswa yang ingin mengambil judul bisa langsung cek di aplikasi tersebut sebelum mengusulkan judulnya.
4. Dapat dikembangkan untuk document teks yang lebih banyak atau kompleks, tidak hanya judul skripsi tapi dapat diterapkan untuk dokumen utuh skripsi atau jurnal penelitian.

DAFTAR PUSTAKA

- Astuti, W. (Agustus 2017). Analisis String Matching Pada Judul Skripsi dengan Algoritma Knuth-Morris Pratt(KMP). *ILKOM Jurnal Ilmiah Volume 9 Nomor 2 ISSN Cetak 2087-1716 ISSN Online 2548-7779*, 168 - 172.
- Astutik, S., Cahyani, A. D., & Sophan, M. K. (November 2014). Sistem Penilaian Esai Otomatis pada E-Learning dengan Algoritma Winnowing. *Jurnal Informatika, Vol 12, No.2 ISSN 1411-0105*, 47-52.
- Christian. (2013). Perancangan Sistem Deteksi Plagiarisme Dokumen Teks Menggunakan Algoritma Damerau Levenshtein Distance. *Jurnal Mahasiswa PTIIK UB Vol 1*.
- leonaerde, G. G. (2014). Penerapan algoritma Boyes Moore Pada Aplikasi Pengajuan Judul Skripsi Berbasis Web. *Seminar Informasi dan Teknologi Ilmiah (INTI)*.
- Mudafiq, P. r. (2011). Aplikasi Pendeteksi Duplikasi Dokumen Teks Bahasa Indonesia Menggunakan Algoritma Winnowing dengan metode K-gram dan Synonym Recognition. *Jurnal Tugas Akhir Jurusan Teknik Informatika Universitas Muhammadiyah Malang*.
- Nurdin, & Amin, M. (September 2017). Sistem Pendeteksian Kemiripan Judul Skripsi Menggunakan Algoritma Winnowing. *Jurnal Nasional Informatika dan Teknologi Jaringan (InfoTekJar) Vol 2, No 1 e- ISSN : 2540-7600, p-ISSN : 2540-7597*.
- Ridho, M. (2013). *Rancang Bangun Aplikasi Pendeteksi Penjiplakan Dokumen Menggunakan Algoritma Biword Winnowing*. Pekanbaru Riau: UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM .
- Setiawan, A. (Januari 2017). Implementasi Algoritma Winnowing Untuk Deteksi Kemiripan Judul Skripsi Studi Kasus STIMIK Budidarma. *Majalah Ilmiah INTI Volume: XII, Nomor : 1 ISSN : 2339-210X*.
- Soelistyo, H. (2011). *Plagiarisme , Pelanggaran Hak cipta dan Etika*. Jakarta: Kanisius.
- Syahputra, A. R. (Maret 2015). Implementasi Algoritma Winnowing Untuk Deteksi Kemiripan Text. *Pelita Informatika Budi Darma, Volume ; IX, Nomor; 1 ISSN 2301-9425*, 134-138.