Sains Data Jurnal Studi Matematika dan Teknologi
ISSN 2986-903X
v2i2(2024); p. 68-79

Deteksi Kemiripan Dokumen Abstrak Skripsi menggunakan Metode Jaro-Winkler Distance dan Synonym Recognition

Muhammad Syahrullah [1] , Fika Hastarita Rachman [2] , Ika Oktavia Suzanti [3]
[1,2,3] Program Studi Teknik Informatika, Universitas Trunojoyo Madura, Bangkalan - Indonesia
E-mail: [1] 180411100006@student.trunojoyo.ac.id, [2] fika.rachman@trunojoyo.ac.id,
[3]iosuzanti@trunojoyo.ac.id

DOI : https://doi.org/10.52620/sainsdata.v2i2.136

Abstrak

Natural Language Processing (NLP) terus berkembang hingga saat ini. Dalam 10 tahun terakhir, NLP berkembang pesat seiring meningkatnya ketersediaan teks elektronik saat ini. Salah satu contoh aplikasi yang mengimplementasikan pendekatan NLP adalah *Similarity Detection* atau deteksi kemiripan. Deteksi kemiripan digunakan untuk mengetahui seberapa mirip dokumen teks satu dengan lainnya. Dokumen teks merupakan sebuah tulisan yang tercetak yang bertujuan untuk menerangkan atau memberikan sebuah informasi tertentu. Pada penelitian ini, metode *Jaro-Winkler Distance* dikombinasikan dengan *Synonym Recognition* untuk mendeteksi nilai persentase kemiripan dari dokumen abstrak skripsi. Abstrak skripsi yang digunakan adalah abstrak skripsi dari Program Studi Infromatika Fakultas Teknik Universitas Trunojoyo Madura dengan jumlah 110 abstrak. Dari uji coba yang telah dilakukan, diperoleh hasil bahwa dengan menggunakan kombinasi metode *Jaro-Winkler Distance* dengan *Synonym Recognition* dinilai kurang efektif karena *score* yang dihasilkan lebih rendah. Uji coba dilakukan menggunakan data sintetis potongan dan data sintetis gabungan. Tujuan dari dibuatnya data sintetis untuk menjadi *ground truth* atau acuan peneliti terhadap nilai *similarity* yang asli dari *query* yaitu agar dapat menghasilkan nilai *Error Rate* dari kinerja metode *Jaro-Winkler Distance* dan *Synonym Recognition*. *Error Rate* yang diperoleh tanpa menggunakan *Synonym Recognition* memiliki nilai sebesar 0.005511, sedangkan menggunakan *Synonym Recognition* diperoleh nilai sebesar 0.0397.

Kata Kunci: *NLP, Similarity Detection, Synonym Recognition, Jaro-Winkler Distance*

Abstract

Natural Language Processing (NLP) continues to develop today. In the last 10 years, NLP has developed rapidly along with the increasing availability of electronic texts today. One example of an application that implements the NLP approach is Similarity Detection. Similarity detection is used to find out how similar text documents are to each other. A text document is a printed piece of writing that aims to explain or provide certain information. In this research, the Jaro-Winkler Distance method is combined with Synonym Recognition to detect the similarity percentage value of the thesis abstract document. The thesis abstracts used are thesis abstracts from the Informatics Study Program, Faculty of Engineering, Trunojoyo University, Madura, with a total of 110 abstracts. From the trials that have been carried out, the results obtained are that using a combination of the Jaro-Winkler Distance method with Synonym Recognition is considered less effective because the resulting score is lower. Trials were carried out using chunk synthetic data and combined synthetic data. The purpose of creating synthetic data is to become ground truth or a researcher's reference for the original similarity value of the query, namely to be able to produce Error Rate values from the performance of the Jaro-Winkler Distance and Synonym Recognition methods. The error rate obtained without using Synonym Recognition has a value of 0.005511, while using Synonym Recognition the value obtained is 0.0397.

Keywords: *NLP, Similarity Detection, Synonym Recognition, Jaro-Winkler Distance*

PENDAHULUAN

Natural Language Processing (NLP) teknik machine learning yang memungkinkan komputer untuk memahami dan mengolah teks manusia dalam berbagai bahasa[1]. Penelitian NLP telah berlangsung selama beberapa dekade terakhir dimulai dari penelitian tentang mesin terjemahan awal pada tahun 1946 [2]. Perkembangan penelitian NLP terus meningkat hingga sampai pada saat ini. Dalam 10 tahun terakhir ini perkembangan NLP berkembang dengan pesat dapat dilihat dengan peningkatan ketersediaan teks elektronik dalam jumlah besar [2]. NLP banyak diimplementasikan pada aplikasi yang menggunakan teks. Beberapa contoh aplikasi yang munggunakan NLP adalah Information Retrival, Question-Answering, Summarization, mesin terjemahan, sistem dialog, Similarity Detection dan lain-lain [2].

Similarity Detection atau deteksi kemiripan merupakan salah satu implementasi dari NLP. Tujuan dari similarity detection adalah untuk dapat mengukur tingkat kemiripan dua buah objek. Salah satu objek yang bisa digunakan pada Similarity Detection adalah dokumen teks [3]. Dokumen teks merupakan sebuah tulisan yang tercetak dan bertujuan untuk menerangkan atau memberikan sebuah informasi tertentu [4].

Beberapa metode similarity detection yang dapat menghitung persentase kemiripan dokumen adalah metode Jaccard Coeffision, Levenshtein Distance, dan Jaro-Winkler Distance. Terdapat beberapa penelitian similarity detection yang sudah dilakukan [5]. Pada penelitian tahun 2022 [6] metode Jaro-Winkler Distance guna mengukur tingkat kemiripan pada dokumen berita online. Dari uji coba yang sudah dilakukan menggunakan 55 data, menyatakan terdapat dokumen yang seharusnya masih bisa terdeteksi nilai similarity tinggi akan tetapi tidak terdeteksi dikarenakan terdapat kata yang memiliki arti yang sama tetapi memiliki kata yang berbeda[6]. Dalam penelitian yang dilakukan pada tahun 2019 [7], metode Rabin-Karp dikombinasikan dengan Synonym Recognition untuk mengantisipasi plagiarisme dalam dokumen skripsi. Dari hasil uji coba yang sudah dilakukan yaitu Metode Rabin-Karp yang dikombinasikan dengan pendekatan Synonym Recognition menghasilkan nilai akurasi pemutusan persentase kemiripan yang lebih baik daripada tanpa dikombinasikan dengan pendekatan Synonym Recognition [7].

Pada penelitian ini metode yang digunakan untuk deteksi nilai persentase similarity adalah metode Jaro-Winkler Distance yang dikombinasikan dengan Synonym Recognition. Dimana pendekatan Synonym Recognition melakukan proses pengecekan dengan menambahkan sinonim yang ada pada kamus Thesaurus kedalam dokumen [7]. Pengukuran kedekatan antar dokumen uji menggunakan metode Jaro-Winkler Distance. Dalam metode Distance, semakin tinggi nilai dari Jaro-Winkler Distance untuk dua kata pada dokumen maka semakin tinggi juga nilai dari Similarity kedua dokumen tersebut [8]. Penelitian pada tahun 2022 [8] yang membandingkan antara metode Jaro-Winkler Distance dengan Levenshtein Distance menyatakan bahwa perhitungan dari metode Jaro-Winkler Distance memiliki tingkat Similarity lebih baik daripada menggunakan Levenshtein Distance [8]. Dari hasil penelitian [6] dapat diketahui bahwa metode Jaro-Winkler Distance tidak mampu mendeteksi kata yang mempunyai makna sama tetapi mempunyai kata yang berbeda [6]. Dokumen yang digunakan pada penelitian ini adalah abstrak skripsi program studi informatika fakultas teknik Universitas Trunojoyo Madura (PSIF FT-UTM) sebanyak 110 abstrak skripsi. Dimana pada Universitas Trunojoyo Madura, skripsi merupakan salah satu persyaratan dasar untuk gelar sarjana. Banyak mahasiswa menganggap skripsi sangat sulit. Sehingga banyak mahasiswa yang sengaja menyalin karya orang lain karena tidak paham dengan materi kuliah yang diajarkan, sengaja menulis karya orang lain karena tidak tahu bagaimana cara mengutip dan mencantumkan sumber informasi dengan baik dan benar, atau mengganti tulisan orang lain dengan sinonimnya [9]. Berdasarkan masalah yang disebutkan, metode Jaro-Winkler Distance yang dikombinasikan dengan pendekatan Synonym Recognition dipilih agar dapat mengetahui berapa nilai similarity yang didapat.

METODE

Jaro-Winkler Distance

Jaro-Winkler adalah sebuah algoritma String-Matching guna menghitung kemiripan antara dua kata. Metode Jaro-Winkler biasa digunakan untuk mendeteksi kemiripan [11]. Metode Jaro-Winkler mempunyai 3 dasar yaitu:

1. Menghitung Panjang dari sebuah string
2. Menemukan jumlah karakter yang sama dalam 2 string
3. Menemukan jumlah transposisi

Jaro-Winkler memiliki rumus untuk menentukan nilai jarak (dj) terhadap 2 string:

$$dj = \frac{1}{3} \times \left(\frac{m}{|s1|} + \frac{m}{|s2|} + \frac{m-t}{m} \right) \quad (1)$$

m = Jumlah karakter yang serupa

S1 = Panjang dari string 1

S2 = Panjang dari string 2

t = Jumlah Transposisi

Untuk menentukan nilai dari transposisi diperlukan menghitung jumlah karakter yang tidak sama di indeks (n) yang sama[11].

$$Transpositions(t) = n/2 \quad (2)$$

n = jumlah *index* sampai ditemukan *non-matching characters*

Jaro-winkler (d_w) memanfaatkan skala *prefix (p)* agar mendapatkan tingkat penilaian yang lebih, berikut ini merupakan rumus dari *Jaro Winkler distance*:

$$d_w = d_j + (lp(1-d_j)) \quad (3)$$

d_j = Nilai dari *Jaro Distance* untuk *string* 1 dan *string* 2

l = Panjang *prefix* umum di awal *string* nilai maksimalnya 4 karakter (Panjang karakter yang sama sebelum ditemukan ketidaksamaan maksimal 4)

p = Konstanta *scaling factor*. Memiliki nilai standar menurut winkler adalah 0,1[11].

Similarity

Similarity adalah pengukuran antara dua objek yang bertujuan untuk mengetahui informasi kemiripan dari objek tersebut. *Similarity* memiliki 3 konsep yang bertujuan untuk menentukan nilai *similarity* (kemiripan) diantaranya [3]:

1. *Distance-based similarity measure*

   Distance-based similarity adalah metode untuk mengukur kemiripan dokumen dengan cara membandingkan 2 objek dan menghitung jarak antar string yang dibandingkan. Metode *distance-based similarity* terdapat pada : *jaccard distance, levenshtein disctance, dice's coefficient*, dan lain-lain.

2. *Feature-based similarity measure*

   Feature-based similarity adalah metode yang melakukan perhitungan tingkat *similarity* dengan cara mengubah objek menjadi bentuk *feature* yang akan di perbandingkan. Metode *feature-based similarity* biasa digunakan untuk pengklasifikasian dan juga *pattern matching* untuk gambar dan teks.
3. *Probabilistic-based similarity measure*

   Probabilistic-based similarity adalah metode untuk menghitung tingkat *similarity* dari 2 objek dengan menampilkan 2 objek yang akan dibandingkan dalam bentuk *probability*. Contoh metode yang ada pada *Probabilistic-based similarity* adalah *Kullback Leibler Distance* dan *Posterior Probability*.

Terdapat 5 tingkat penilaian dalam menentukan kemiripan pada dokumen yang diuji yaitu[3]:

1. Tingkat 0%

   Pada tingkatan ini menyatakan bahwa kedua dokumen tersebut tidak terdapat kemiripan sama sekali.
2. Tingkat <15%

   Pada tingkatan ini menyatakan bahwa kedua dokumen yang dibandingkan memiliki sedikit kemiripan didalamnya.
3. Tingkat 15%-50%

   Pada tingkatan ini menyatakan bahwa kedua dokumen yang dibandingkan mendekati kemiripan.
4. Tingkat >50%

   Pada tingkatan ini menyatakan bahwa kedua dokumen yang dibandingkan mirip.
5. Tingkat 100%

   Pada tingkatan ini menyatakan bahwa kedua dokumen yang dibandingkan memiliki isi yang sama persis.

Synonym Recognition

Synonym Recognition adalah sebuah algoritma untuk mendeteksi *similarity* pada teks dengan memanfaatkan pendekatan sinonim [7]. Sinonim merupakan kata yang mempunyai makna sama namun katanya berbeda. Kata yang terdeteksi memiliki arti yang sama akan diubah menjadi kata yang sesuai pada database [7]. Dalam penelitian ini dokumen akan dikomparasikan dengan dokumen lainnya dengan mendeteksi kata-kata antar dokumen yang memilik kata arti sama sehingga tingkat kemiripan bisa lebih akurat.

Text Pre-processing

Text Preprocessing merupakan suatu tahapan yang bertujuan untuk membersihkan teks dengan menghapus kata-kata yang tidak terpakai [10]. Juga dilakukan proses penstrukturan ulang kata dengan cara membedakan tiap kata, menghapus imbuhan di setiap kata, serta menghilangkan kata yang tidak bermakna pada dataset sehingga menciptakan data yang bersih dan siap diproses melalui tahapan berikut.

1. *Case Folding*

   Case Folding merupakan tahapan yang memiliki tujuan untuk mengubah kata sepenuhnya menjadi huruf kecil [10].

2. *Tokenizing*

   Tokenizing adalah tahap untuk memotong string sesuai dengan setiap kata yang menyusunnya [10]. Pemisahan teks dilakukan sehingga menghasilkan potongan token, yang dapat berupa huruf, kata, atau kalimat, sebelum dianalisis lebih lanjut.
3. *Stopword*

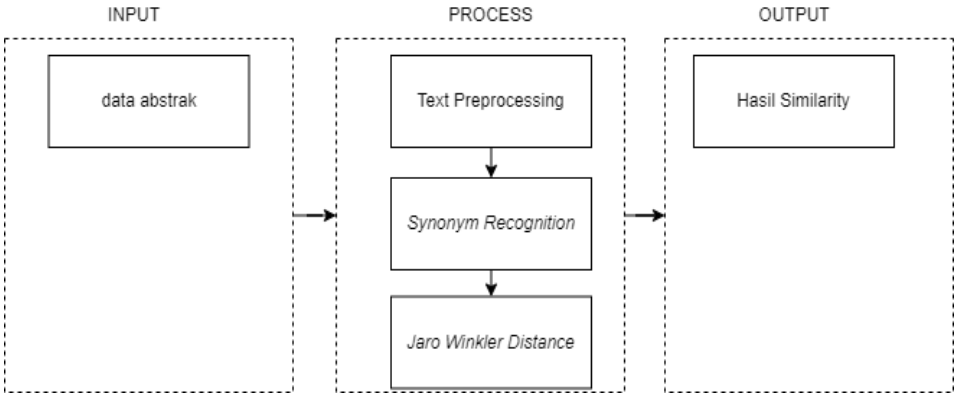
   Stopword adalah tahap menghapus kata-kata yang tidak relevan [10].
4. *Stemming*

   Stemming merupakan tahapan untuk menurunkan jumlah variasi indeks dalam satu dokumen serta melakukan perubahan kata menjadi kata dasar dan arti yang sejenis tetapi bentuknya berbeda akibat adanya imbuhan yang berbeda.

Dataset

Penelitian ini menggunakan dataset abstrak yang menggunakan bahasa Indonesia dari jurusan Teknik Informatika yang diperoleh dari Portal Tugas Akhir Universitas Trunojoyo Madura (*https://pta.trunojoyo.ac.id/*) pada 5 desember 2022 [16]. Data yang digunakan berjumlah 110 abstrak.

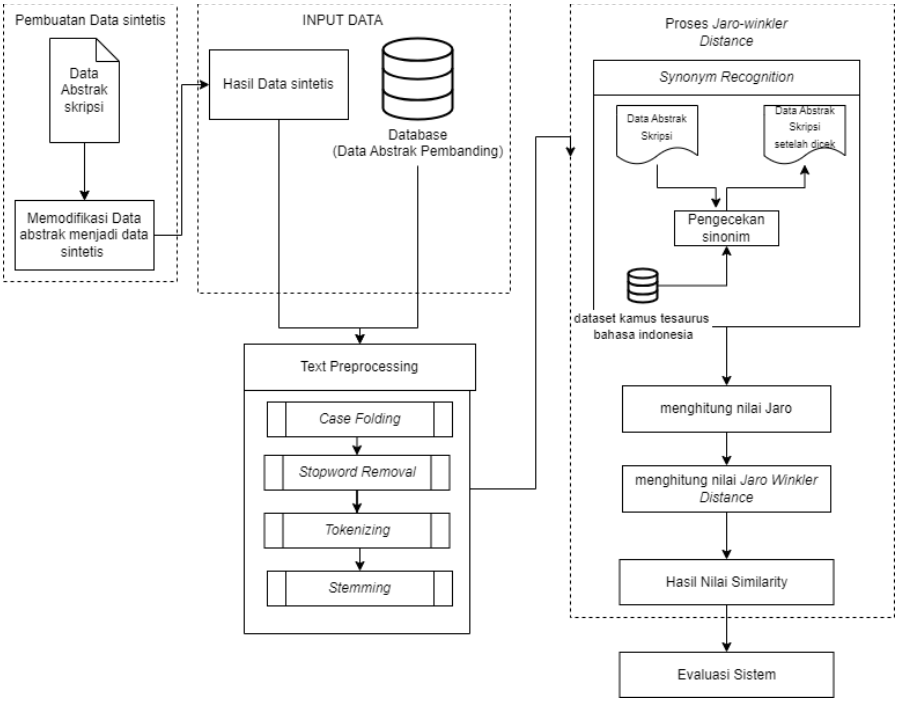
Perancangan dan Arsitektur Sistem



Gambar 1 Diagram IPO

Gambar 1 menunjukkan diagram IPO dari sistem deteksi kemiripan dokumen abstrak dengan menerapkan metode *Jaro-Winkler Distance* dan *Synonym Recognition*. Terdapat 3 tahapan yaitu *input, process*, dan *output*. Pada tahapan pertama yaitu *input*, data yang diinput berupa data abstrak skripsi. Tahapan selanjutnya yaitu *process*, dilakukan proses pada data yang sudah diinputkan sebelumnya yaitu *text preprocessing, Synonym Recognition*, dan *Jaro-Winkler Distance*. Tahapan terakhir yaitu *output*, setelah dilakukan tahap process, didapatkan hasil output berupa nilai *similarity*.

v2i2(2024); p. 68-79



Gambar 2 Arsitektur Sistem

Pada Gambar 2 proses pertama yang dilakukan membuat data sintetis dari data abstrak skripsi. Selanjutnya menginputkan data abstrak. Proses selanjutnya dilakukan *text preprocessing* data abstrak dan data pembanding, yang mana terdapat beberapa tahapan pada *text preprocessing* yaitu *Casefolding, Stopword Removal, Tokenizing* dan *stemming*. Setelah melakukan *text preprocessing*, selanjutnya data abstrak diproses menggunakan *Synonym Recognition* untuk melakukan pengecekan kesamaan kata yang memiliki arti sama atau sinonim. Sedangkan data pembanding diproses tanpa melalui pengecekan sinonim. Selanjutnya data dihitung menggunakan metode *Jaro-Winkler Distance* untuk mendapatkan nilai *similarity* dari data yang sudah diproses.

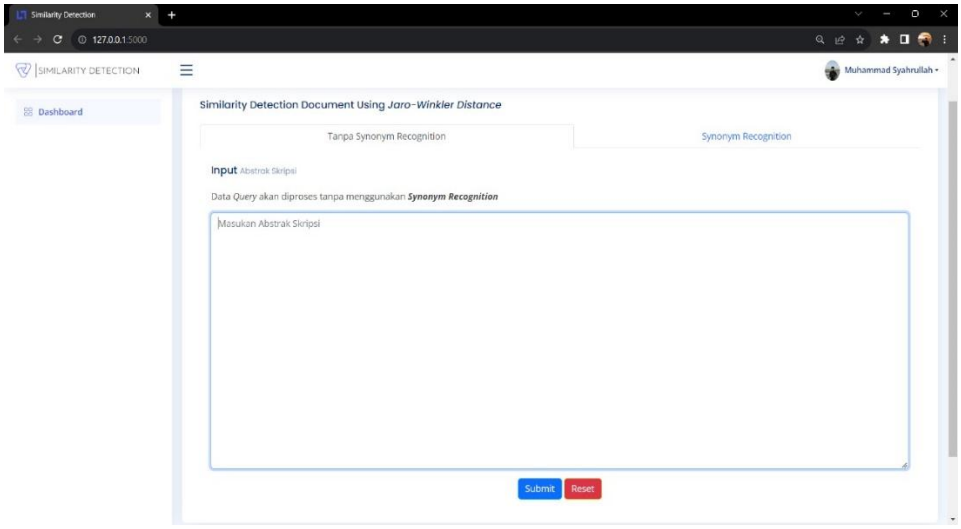
HASIL DAN PEMBAHASAN

Hasil Implementasi Antarmuka

Berikut adalah hasil implementasi antarmuka berbasis web dari sistem deteksi kemiripan dokumen abstrak skripsi yang terdiri dari halaman awal, hasil similarity tanpa synonym recognition, dan hasil similarity menggunakan synonym recognition.

v2i2(2024); p. 68-79

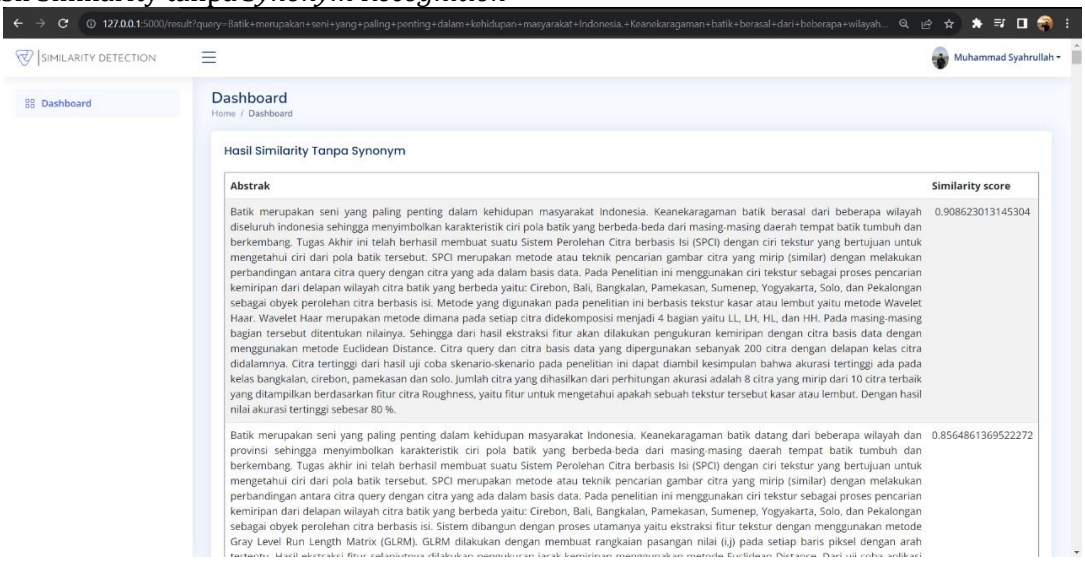
1. Halaman Awal



Gambar 3 Tampilan Halaman Awal

Gambar 3 menunjukkan tampilan halaman awal pada sistem yang digunakan untuk mengetahui nilai *similarity*. Sistem yang dibuat memiliki navigasi dengan merujuk pada nilai *similarity* menggunakan *Synonym Recognition* dan tanpa *Synonym Recognition* dari data abstrak skripsi. Masing-masing navigasi ini berisi tabel input untuk dokumen yang diketahui nilai *similarity* dan terdapat tombol submit dan reset.

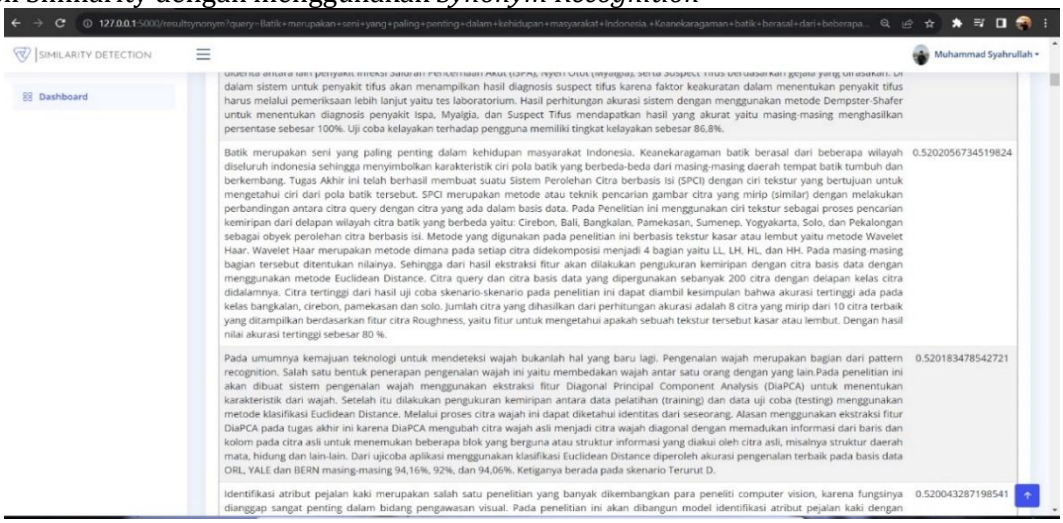
2. Hasil Similarity tanpa *Synonym Recognition*



Gambar 4 Hasil Similarity Tanpa Synonym Recognition

Gambar 4 merupakan tampilan hasil *similarity* abstrak tanpa menggunakan *Synonym Recognition* dengan menampilkan hasil *score similarity* dari yang terbesar hingga yang terkecil.

3. Hasil Similarity dengan menggunakan *Synonym Recognition*



Gambar 5 Hasil Similarity dengan Synonym Recognition

Gambar 5 merupakan tampilan hasil *similarity* abstrak dengan menggunakan *Synonym Recognition* dengan menampilkan hasil *score similarity* dari yang terbesar hingga yang terkecil.

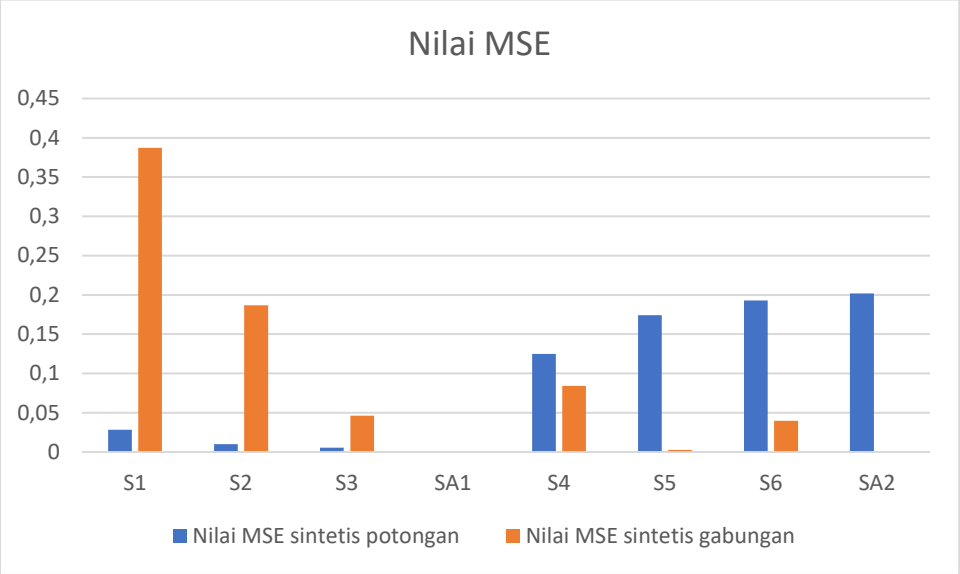
Hasil Uji Coba

Hasil dari pengujian sistem yang dilakukan dengan masing-masing data sintetis didapatkan nilai sebagai berikut:

Tabel 1 Hasil Uji Coba

Kode skenario	Nilai MSE sintetis potongan	Nilai MSE sintetis gabungan
S1	0.028352	0.387364
S2	0.009928	0.187004
S3	0.005511	0.046442
SA1	0	
S4	0.125111	0.08403
S5	0.174175	0.002559
S6	0.192919	0.03978
SA2	0.201940421	

Visualilasi diagram batang berdasarkan data dari Tabel 1 adalah sebagai berikut.



Berdasarkan hasil skenario uji coba yang telah dilakukan, diperoleh hasil nilai *error* dari masing-masing data uji. Nilai tersebut diperoleh dari data uji yang diinputkan ke sistem untuk mendapatkan *score* dari data uji. Setelah mendapatkan *score*, dilakukan perhitungan dengan menggunakan rumus *mean squared error* untuk mendapatkan nilai *error rate* seperti pada tabel 1. Nilai yang dibutuhkan adalah *score* sistem, *groundtruth* dan jumlah data uji. Berdasarkan hasil uji coba nilai *error rate* terendah didapatkan pada skenario S3 dengan menggunakan data sintetis potongan 75% tanpa menggunakan *Synonym Recognition* sebesar 0.005511. sedangkan dengan data asli nilai *error rate* terendah sebesar 0.

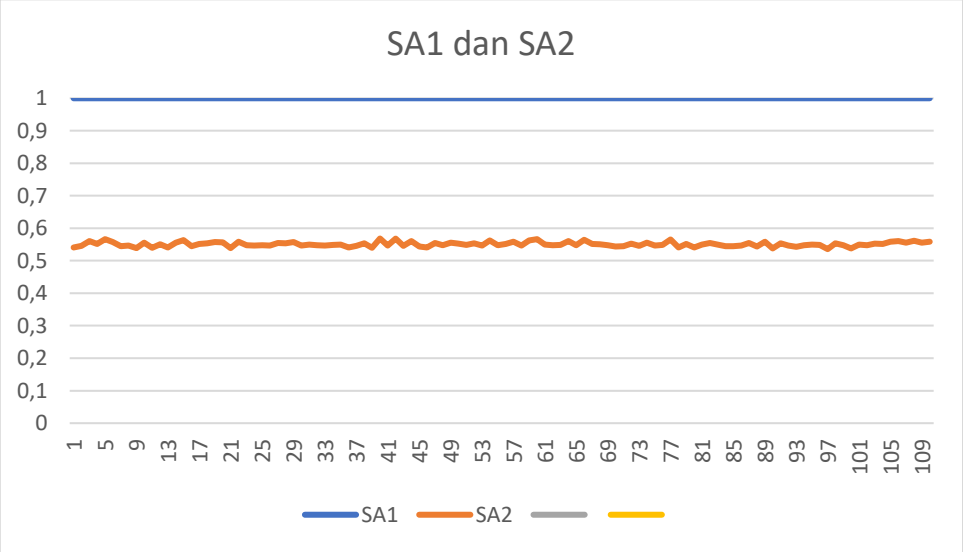
Analisa Uji Coba

Hasil nilai *score similarity* tanpa menggunakan *Synonym Recognition* dengan data sintetis potongan memiliki nilai *error rate* yang kecil sedangkan dengan menggunakan *synonym recognition* mendapatkan hasil *error rate* yang lebih tinggi. Hal tersebut terjadi karena perhitungan *Jaro-Winkler Distance* menghitung berdasarkan karakter yang sama pada data uji. Pada hal ini data uji atau *query* memiliki peran penting dalam pengujian sistem. Nilai *error* sistem deteksi kemiripan dokumen abstrak skripsi dapat diketahui dengan membuat data sintetis yang memiliki *score similarity* atau *score* asli yang sudah diketahui sebagai pembanding agar dapat dihitung nilai *errornya* menggunakan rumus *mean squared error*.

Pembuatan data sintetis juga digunakan sebagai *ground truth* atau acuan untuk mendapatkan nilai *error rate* data uji. Pembuatan data sintetis juga digunakan untuk mengetahui apakah panjang dari karakter mempengaruhi *score* yang dihasilkan dapat dilihat pada hasil skenario uji coba S1-S3 dengan menggunakan data sintetis gabungan memiliki nilai *error* yang lebih besar pada data sintetis potongan, karena panjang data sintetis gabungan merupakan panjang data asli. Berdasarkan nilai error yang diperoleh dapat mengetahui tingkat error dari sistem deteksi kemiripan dokumen dalam menampilkan *score* atau nilai *similarity*.

Penggunaan *Synonym Recognition* pada penelitian kali ini dinilai kurang efektif dikarenakan *score similarity* yang diperoleh pada sistem lebih rendah. Penulis menyimpulkan pemasalahan tersebut terjadi karena banyaknya sinonim yang ditambahkan kedalam data uji, sehingga

menyebabkan *score* yang diperoleh lebih rendah. Dapat dilihat pada gambar 4.8 *score similarity* yang dihasilkan tanpa menggunakan *synonym recognition* lebih baik dibandingkan menggunakan *synonym recognition*.



Gambar 4. 1 Chart S3 dan S6

Berdasarkan uji coba yang dilakukan ketika menggunakan *Synonym Recognition* mendapat nilai *error rate* pada hasil skenario uji coba SA1 yaitu 0, selanjutnya dilakukan uji coba dengan data yang sama tanpa menggunakan *synonym recognition* dan mendapatkan *error rate* pada hasil skenario uji coba SA2 yaitu 0.201940421. Pada gambar 4.8 dapat dilihat bahwa tanpa menggunakan *Synonym Recognition, score* yang diperoleh lebih dekat dengan *ground truth* dibandingkan dengan menggunakan *Synonym Recognition*.

KESIMPULAN DAN SARAN

Kesimpulan

Dalam penelitian ini, dilakukan pengujian metode *Jaro-Winkler Distance* dan *Synonym Recognition* dalam sistem deteksi kemiripan dokumen dengan menggunakan *dataset* abstrak skripsi dari PSIF FT-UTM. Hasil penelitian menunjukan bahwa metode *Jaro-Winkler Distance* yang dikombinasikan dengan *Synonym Recognition* kurang efektif dibandingkan tanpa menggunakan *Synonym Recognition.* Hal ini dinilai berdasarkan *score* yang dihasilkan tanpa menggunakan *synonym recognition* metode *Jaro-Winkler Distance* dapat mengukur *similariy* dengan sangat baik sehingga menhasilkan *score* sebesar 1 dan nilai *error rate* yaitu 0. Sedangkan dengan menambahkan *synonym recognition score* yang dihasilkan lebih rendah dengan rata-rata *score* 0.55. Sedangkan uji coba dilakukan menggunakan data sintetis potongan dan data sintetis gabungan. *Error Rate* yang diperoleh tanpa menggunakan *Synonym Recognition* memiliki nilai sebesar 0.005511, sedangkan menggunakan *Synonym Recognition* diperoleh nilai sebesar 0.0397. Maka dari hasil pengujian yang telah dilakukan didapatkan hasil metode *Jaro-Winkler Distance* kurang cocok dikombinasikan dengan *Synonym Recognition*.

Saran

Untuk mencapai hasil yang lebih baik, saran yang dapat diberikan untuk penelitian selanjutnya adalah untuk mengkombinasikan metode *Jaro-Winkler Distance* dengan metode atau ekstraksi fitur lain agar mendapatkan hasil yang lebih baik. Selain itu juga dapat megkombinasikan *Synonym Recognition* dengan metode *string matching* lain yang melakukan pencarian per-kata.

REFERENSI

[1] M. Amien, "Sejarah dan Perkembangan Teknik Natural Language Processing (NLP) Bahasa Indonesia: Tinjauan tentang sejarah, perkembangan teknologi, dan aplikasi NLP dalam bahasa Indonesia," Mar. 2023,

[2] E. D. Liddy, "Natural Language Processing Natural Language Processing Natural Language Processing 1," 2001

[3] K. Proposal Dan Isi Skripsi Dengan Algoritma Rabin-Karp, L. Juliana Purba, and L. Sitorus, "Perancangan Aplikasi Untuk Menghitung Persentase," 2018.

[4] G. E. I. Kambey, R. Sengkey, and A. Jacobus, "Penerapan Clustering pada Aplikasi Pendeteksi Kemiripan Dokumen Teks Bahasa Indonesia," *Jurnal Teknik Informatika*, vol. 15, no. 2, pp. 75–82.

[5] T. Bollé and E. Casey, "Using computed similarity of distinctive digital traces to evaluate non-obvious links and repetitions in cyber-investigations," in *DFRWS 2018 EU - Proceedings of the 5th Annual DFRWS Europe*, Digital Forensic Research Workshop, 2018, pp. S2–S9. doi: 10.1016/j.diin.2018.01.002.

[6] T. Efriyanto and M. Hayaty, "975 Jaro Winkler Algorithm For Measuring Similarity Online News," *Jurnal Teknik Informatika (JUTIF)*, doi: 10.20884/1.jutif.2022.3.4.152.

[7] N. Prima Putra and S. Sularno, "Penerapan Algoritma Rabin-Karp Dengan Pendekatan Synonym Recognition Sebagai Antisipasi Plagiarisme Pada Penulisan Skripsi," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 1, no. 2, pp. 48–58, Jul. 2019, doi: 10.47233/jteksis.v1i2.52.

[8] I. Syahputra and F. Syakti, "Perbandingan Algoritma Levenshtein dan Jaro Winkler Pada Sistem Informasi Pencarian Dokumen Perundang-Undangan (Studi Kasus : Diskominfo Lahat)," *SMATIKA JURNAL*, vol. 12, no. 02, pp. 176–186, Dec. 2022, doi: 10.32664/smatika.v12i02.696.

[9] P. C. S. Mahendra, "Deteksi Similarity Abstrak Skripsi Menggunakan Metode," 2022.

[10] R. Rosyadi and S. Al-Faraby, "Penerapan Question Answering System Pada Pembahasan Agama Islam Dengan Pendekatan Metode Pattern Based," vol. 2, no. 4, 2018.

[11] H. Nur Hanani, H. Jayadianti, H. Cahya Rustamaji, and U. Pembangunan Nasional Veteran Yogyakarta, "Fuzzy String Matching for Semi-Automatication of Words with Jaro Winkler Distance Algorithm on Microsoft Word Documents Fuzzy String Matching untuk Semi-Otomatisasi Pencocokan Kata dengan Algoritma Jaro Winkler Distance pada Dokumen Microsoft Word," pp. 13–2021, [Online]. Available: www.myvocabulary.com

[12] Bunga Dea Laraswati, "Data Sintetis: Apa Itu dan Apa Kegunaannya?," Algoritma.

[13] R. P. Pratama, M. Faisal, and A. Hanani, "Deteksi Plagiarisme pada Dokumen Jurnal Menggunakan Metode Cosine Similarity," *SMARTICS Journal*, vol. 5, no. 1, pp. 22–26, Apr. 2019, doi: 10.21067/smartics.v5i1.2848.

[14] D. Plagiarisme *et al.*, "Techno Xplore Jurnal Ilmu Komputer dan Teknologi Informasi." [Online]. Available: http://www.smallseotools.com/

[15] S. Fatonah, A. Hadinegoro, A. Hadinegoro, A. D. Hartanto, and A. D. Hartanto, "Deteksi Kemiripan Abstraksi Tugas Akhir Diploma Informatika Universitas AMIKOM Yogyakarta dengan Algoritma Rabin Karp," *JURIKOM (Jurnal Riset Komputer)*, vol. 7, no. 1, p. 1, Feb. 2020, doi: 10.30865/jurikom.v7i1.1927.

[16] "Portal Tugas Akhir Universitas Trunojoyo Madura.", https://pta.trunojoyo.ac.id/