

TESIS

**STUDI DAN ANALISIS ALGORITMA SIMILARITY UNTUK
IDENTIFIKASI KEMIRIPAN JUDUL SKRIPSI
(Studi Kasus : Fakultas Teknologi Informasi Universitas Islam Balitar)**



Disusun oleh:

Nama : Harry Eko Ujlantoro Adi Basuki
NIM : 22.55.1259
Konsentrasi : Digital Transformation Intelligence

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2023**

TESIS

**STUDI DAN ANALISIS ALGORITMA SIMILARITY UNTUK
IDENTIFIKASI KEMIRIPAN JUDUL SKRIPSI
(Studi Kasus : Fakultas Teknologi Informasi Universitas Islam Balitar)**

**STUDY AND ANALYSIS OF SIMILARITY ALGORITHM TO IDENTIFY
SIMILARITY THESIS TITLE**

(Case Study : Faculty of Information Technology, Islamic University of Balitar)

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister



Disusun oleh:

Nama : Harry Eko Ujlantoro Adi Basuki
NIM : 22.55.1259
Konsentrasi : Digital Transformation Intelligence

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2023

HALAMAN PENGESAHAN

**STUDI DAN ANALISIS ALGORITMA SIMILARITY UNTUK IDENTIFIKASI
KEMIRIPAN JUDUL SKRIPSI
(Studi Kasus : Fakultas Teknologi Informasi Universitas Islam Balitar)**

**STUDY AND ANALYSIS OF SIMILARITY ALGORITHM TO IDENTIFY
SIMILARITY THESIS TITLE
(Case Study : Faculty of Information Technology, Islamic University of Balitar)**

Dipersiapkan dan Disusun oleh

Harry Eko Ujiantoro Adi Basuki

22.55.1259

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Jumat, 1 Desember 2023

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 1 Desember 2023
Rektor

Prof. Dr. M. Suvanto, M.M.
NIK. 190302001

HALAMAN PERSETUJUAN

STUDI DAN ANALISIS ALGORITMA SIMILARITY UNTUK IDENTIFIKASI KEMIRIPAN JUDUL SKRIPSI

(Studi Kasus : Fakultas Teknologi Informasi Universitas Islam Balitar)

STUDY AND ANALYSIS OF SIMILARITY ALGORITHM TO IDENTIFY SIMILARITY THESIS TITLE

(Case Study : Faculty of Information Technology, Islamic University of Balitar)

Dipersiapkan dan Disusun oleh

Harry Eko Ujiantoro Adi Basuki

22.55.1259

Telah Ditujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Jumat, 1 Desember 2023

Pembimbing Utama

Prof. Dr. Kusriani, M.Kom.
NIK. 190302106

Pembimbing Pendamping

Drs. Asro Nasiri, M.Kom.
NIK. 190302152

Anggota Tim Penguji

Tonny Hidayat, M.Kom., Ph.D.
NIK. 190302182

Dhani Ariatmanto, M.Kom, Ph.D.
NIK. 190302197

Prof. Dr. Kusriani, M.Kom.
NIK. 190302106

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 1 Desember 2023
Direktur Program Pascasarjana

Prof. Dr. Kusriani, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Harry Eko Ujiantoro Adi Basuki
NIM : 22.55.1259
Konsentrasi : Digital Transformation Intelligence

Menyatakan bahwa Tesis dengan judul berikut:
**STUDI DAN ANALISIS ALGORITMA SIMILARITY UNTUK
IDENTIFIKASI KEMIRIPAN JUDUL SKRIPSI**
(Studi Kasus : Fakultas Teknologi Informasi Universitas Islam Balitar)

Dosen Pembimbing Utama : Prof. Dr. Kusriani, M.Kom.
Dosen Pembimbing Pendamping : Drs. Aaro Nasir, M.Kom.

1. Karya tulis ini adalah benar-benar **ASLI** dan **BELUM PERNAH** diajukan untuk mendapatkan gelar akademik, baik di Universitas AMKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian **SAYA** sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jalan dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab **SAYA**, bukan tanggung jawab Universitas AMKOM Yogyakarta
5. Pernyataan ini **SAYA** buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka **SAYA** bersedia menerima **SANKSI AKADEMIK** dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 1 Desember 2023

Yang Menyatakan,



Harry Eko Ujiantoro Adi Basuki

HALAMAN PERSEMBAHAN

Penulisan tesis dipersembahkan untuk Almarhum Bapakku Abdul Halik bin Joeliastro dan Ibuku Almarhumah Sulastri binti Joyo Kasmin serta istriku Kesi serta anak gantengku Lucky dan cah ayuku Putri serta sahabat kehidupanku yang tidak dapat disebut satu persatu khususnya Sang Penyemangat dan Pendorong kebangkitan hidupku. Terimakasih telah mewarnai hidupku. Perjuangan ini tidak lepas dari rintisan Almarhum Bapak Ibu Penulis menjadikan Penulis Sarjana Teknik di Telkom University (d/h STT Telkom) yang berlanjut berkesempatan menjadi Master Komputer di Kampus Kebanggaan Universitas AMIKOM Yogyakarta dengan harapan anak keturunan Penulis minimal memiliki pendidikan sederajat sama atau lebih. Insha Allah sebagai pengejawantahan kemampuan dan keilmuan yang dimiliki serta sesuai jalur karir yang dipilih.

Penulis dalam menyusun Tesis ini pada usia yang tidak muda lagi yakni 49 tahun namun terinspirasi oleh pepatah belajarlal sampai batas waktu ke liang lahat. Harta, Tahta dan Keluarga akan berpisah dengan sendirinya disaat kita tiada namun tulisan ini menjadi bermanfaat dan amal jariyah Penulis di masa depan dan dapat menjadi motivasi untuk memberikan karya lebih baik dan bermanfaat.

Dan semoga tesis ini dan gelar M.Kom yang insha allah Penulis sandang mampu menjadi penyemangat dan peningkat profesionalisme serta keterampilan Penulis baik sebagai ICT Consultant, Auditor dan Asesor Kompetensi. Amin YRA.

HALAMAN MOTTO

Hidup adalah proses perjalanan menuju kebaikan, keamanan, berkemampuan, terampil dan bijaksana yang tidak berhenti sampai akhir hidup di dunia sebagai bekal kehidupan di akhirat dan dapat menjadikan kita agar termasuk makhluk terbaik di Haribaan Allah SWT yakni bermanfaat bagi manusia lain di dunia fana ini.

“ Sekali bergerak tetap berhasil - Besuki, Djoeliasastro 1965 “



KATA PENGANTAR

Bismillahirrohmanirrohim, Alhamdulillah wa Syukurillah saya panjatkan kehadiran Allah SWT yang memberikan karunia dan rahmat-Nya kepada Penulis. Sholawat serta salam tak henti penulis panjatkan kehadiran junjungan Nabi Agung Sayyidina Rasulullah Muhammad SAW, semoga kita semua mendapatkan syafaatnya di hari Akhir. Amin Ya Rabbal Alamin.

Terima kasih saya ucapkan kepada Prof. Dr. Kusriani, M.Kom, dan Drs. Asro Nasiri, M.Kom selaku Dosen Pembimbing, Tim Dosen Penguji, dan semua pihak yang telah mendukung dan memberikan support atas penyelesaian tesis dengan segala duka dan waktu yang tidak pendek.

Spesial disampaikan terimakasih untuk istri penulis Mama Kesi, Anakku ganteng yang penuh wibawa Lucky Giga di Kampus Supernya dan cantikku Nisrina Putri di jannati baiti penjaga spirit Papa-nya, yang telah ikut serta memberikan dukungan dan doa yang pada akhirnya saya mampu menyelesaikan tesis ini.

Yogyakarta, 17 Oktober 2023

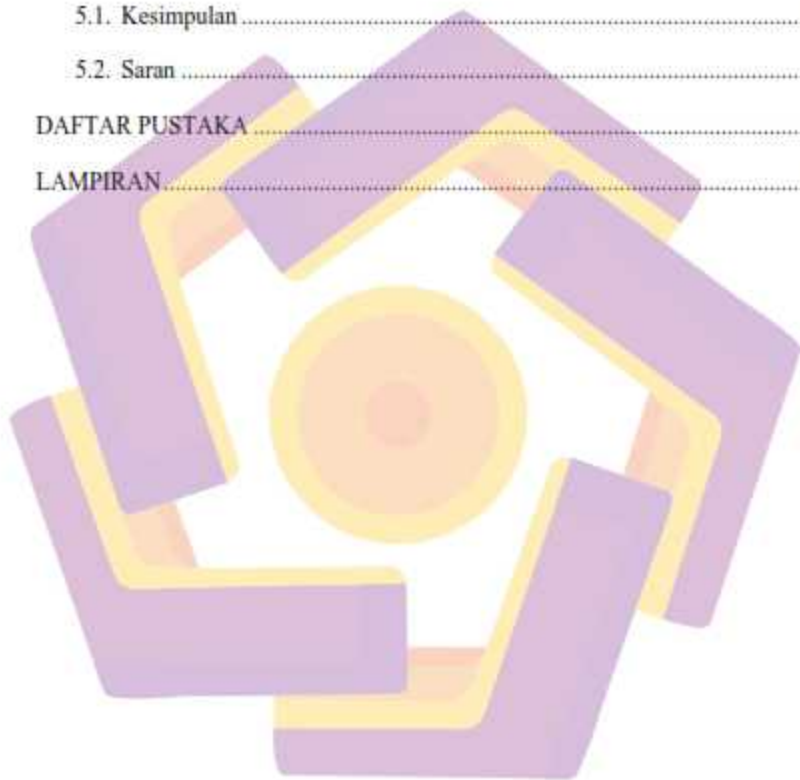
Penulis

DAFTAR ISI

HALAMAN JUDUL.....	ii
HALAMAN PENGESAHAN.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PERNYATAAN KEASLIAN TESIS.....	v
HALAMAN PERSEMBAHAN.....	vi
HALAMAN MOTTO.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiv
INTISARI.....	xv
<i>ABSTRACT</i>	xvi
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah.....	1
1.2. Rumusan Masalah.....	5
1.3. Batasan Masalah.....	6
1.4. Tujuan Penelitian.....	7
1.5. Manfaat Penelitian.....	8
BAB II TINJAUAN PUSTAKA.....	9
2.1. Tinjauan Pustaka.....	9
2.2. Keaslian Penelitian.....	13

2.3. Landasan Teori.....	27
2.3.1. Algoritma <i>Cosine Similarity</i>	27
2.3.2. Pembobotan <i>Cosine Similarity</i>	28
2.3.3. Algoritma <i>Vector Space model</i>	29
2.3.4. Pengukuran nilai kemiripan.....	31
2.3.5. Turnitin.....	32
2.3.6. Pemrograman Python.....	33
2.3.7. NLP.....	35
2.3.8. Flask.....	36
2.3.9. Diagram Fishbone.....	37
2.3.10. Confusion Matrix.....	37
BAB III METODE PENELITIAN.....	40
3.1. Jenis, Sifat, dan Pendekatan Penelitian.....	40
3.2. Metode Pengumpulan Data.....	41
3.3. Metode Analisis Data.....	41
3.4. Alur Penelitian.....	42
BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....	46
4.1. Pengumpulan Data.....	46
4.2. Tahap Pre-Processing.....	47
4.3. Proses Penghitungan <i>Cosine Similarity</i>	51
4.4. Pembobotan <i>Cosine Similarity</i>	53
4.5. Pengujian Sistem.....	55
4.5.1. Pengujian Algoritma VSM.....	55

4.5.2.	Pengujian Algoritma <i>Cosine Similarity</i>	59
4.5.3.	Pengujian Judul Skripsi Menggunakan Layanan Turnitin	63
4.5.4.	Perbandingan Kemiripan Algoritma terhadap layanan <i>turnitin</i>	66
BAB V PENUTUP		72
5.1.	Kesimpulan	72
5.2.	Saran	73
DAFTAR PUSTAKA		76
LAMPIRAN		81



DAFTAR TABEL

Tabel 1.1. Data Skripsi Jurusan Informatika Universitas Islam Balitar.....	4
Tabel 2.1. Matriks literatur review dan posisi penelitian STUDI DAN ANALISIS ALGORITMA SIMILARITY UNTUK IDENTIFIKASI KEMIRIPAN JUDUL SKRIPSI.....	13
(Studi Kasus : Fakultas Teknologi Informasi Universitas Islam Balitar)	13
Tabel 2.1. Lanjutan	14
Tabel 2.1. Lanjutan	15
Tabel 2.1. Lanjutan	16
Tabel 2.1. Lanjutan	17
Tabel 2.1. Lanjutan	18
Tabel 2.1. Lanjutan	19
Tabel 2.1. Lanjutan	20
Tabel 2.1. Lanjutan	21
Tabel 2.1. Lanjutan	22
Tabel 2.1. Lanjutan	23
Tabel 2.1. Lanjutan	24
Tabel 2.1. Lanjutan	25
Tabel 2.1. Lanjutan	26
Tabel 2.2 Makna Pembobotan.....	29
Tabel 4.1 Proses Integrasi	47
Tabel 4.2 Proses <i>Case Folding</i>	48

Tabel 4.3 proses <i>tokenizing</i>	49
Tabel 4.4 proses <i>stopword/filtering</i>	50
Tabel 4.6 Perbandingan Judul	52
Tabel 4.7. Hasil <i>Cosine Similarity</i>	53
Tabel 4.8 Perbandingan Judul	54
Tabel 4.9. Tes Kemiripan Judul Skripsi dengan VSM	54
Tabel 4.10 Tes perbandingan <i>similarity</i>	66
Tabel 4.10 Lanjutan	67
Tabel 4.10 Lanjutan	68
Tabel 4.10 Lanjutan	69
Tabel 4.10 Lanjutan	70
Tabel 4.10 Lanjutan	71

DAFTAR GAMBAR

Gambar 2.1. Metode VSM.....	30
Gambar 2.2. Elemen NLP.....	35
Gambar 2.3. Logo Flask.....	36
Gambar 3.1. Alur Penelitian.....	42
Gambar 3.2. Alur Pengujian Metode <i>Vector Space Model</i> dan <i>Cosine Similarity</i>	44
Gambar 3.3 Ilustrasi Pengecekan <i>Similarity</i> Menggunakan VSM berbasis Vektor	45
Gambar 4.1. Data training judul skripsi.....	46
Gambar 4.2 Proses <i>Stemming</i>	51
Gambar 4.3. Tampilan awal sistem deteksi plagiarisme Algoritma VSM.....	58
Gambar 4.4 Hasil pengecekan plagiarisme metode VSM.....	59
Gambar 4.5. Tampilan awal sistem deteksi plagiarisme Algoritma <i>Cosine Similarity</i>	60
Gambar 4.6 Hasil pengecekan plagiarisme metode <i>Cosine Similarity</i>	62
Gambar 4.7 Pengujian judul skripsi menggunakan layanan <i>turnitin</i>	64
Gambar 4.8 Laporan plagiasi judul yang diusulkan menggunakan <i>turnitin</i>	64
Gambar 4.9 Pengujian 30 judul skripsi menggunakan <i>turnitin</i>	65

INTISARI

Langkah awal dalam proses pengajuan skripsi adalah proses validasi oleh pembimbing skripsi. Masalah utama yang dihadapi dosen pembimbing skripsi di Fakultas Teknologi Informasi Universitas Islam Balitar ini adalah banyaknya proses validasi pengajuan skripsi yang dilakukan oleh mahasiswa. Hal ini disebabkan belum adanya sistem validasi pengajuan judul, baik yang sudah pernah diajukan maupun belum.

Pada penelitian ini, penulis melakukan penelitian untuk membuat sistem micro framework berbasis Python yang memiliki kemiripan dengan mesin pencari informasi. Sistem tersebut bertujuan untuk mendeteksi kesamaan judul skripsi yang diajukan oleh mahasiswa Fakultas Teknologi Informasi Universitas Islam Balitar .

Metode yang digunakan dalam sistem ini adalah *Cosine Similarity* yang merupakan salah satu metode yang digunakan untuk mendeteksi kesamaan pada suatu kata. Hasil dari penelitian ini adalah sebuah sistem yang dapat digunakan sebagai alat untuk pencarian informasi. Sistem menghasilkan informasi berupa persentase kesamaan judul skripsi, yang dapat digunakan sebagai alat bantu validasi judul skripsi di Fakultas Teknologi Informasi Universitas Islam Balitar

Kata kunci: *Cosine Similarity, Flask, Similarity, Micro Framework*

ABSTRACT

The first step in the thesis submission process is the validation process by the thesis supervisor. The main problem faced by thesis supervisors at the Faculty of Information Technology, Balitar Islamic University is the large number of validation processes for thesis submissions made by students. This is due to the lack of a system to validate the thesis submission, whether it has been submitted or not.

In this study, the authors conducted research to create a Python-based microframework system that has similarities with information search engines. The system aims to detect the similarity of thesis titles submitted by students of the Faculty of Information Technology, Balitar Islamic University.

The method used in this system is cosine similarity, which is one of the methods used to detect similarity in a word. The result of this research is a system that can be used as a tool for information retrieval. The system produces information in the form of a percentage of similarity of dissertation titles, which can be used as a tool for validating dissertation titles at the Faculty of Information Technology, Balitar Islamic University.

Keyword: Cosine Similarity, Flask, Similarity, MicroFramework



BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Perkembangan teknologi yang cukup pesat saat ini menyebabkan keterbukaan informasi untuk digunakan baik secara legal maupun ilegal semakin marak. Hal ini juga berdampak pada dunia pendidikan, di mana plagiat menurut Prasetya, dkk (2018) menjelaskan bahwa sebuah bentuk penjiplakan yang melanggar hak cipta, semakin marak terjadi. Penjiplakan dokumen, baik itu naskah kuliah, tugas, laporan praktikum, skripsi, tesis, bahkan disertasi, juga semakin marak terjadi.

Program Studi Teknik Informatika merupakan salah satu program studi yang ada di Fakultas Teknologi Informasi Universitas Islam Balitar. Lokasi kampus ini terletak di Jalan Majapahit No. 2-4, Sananwetan, Kecamatan Sananwetan, Kabupaten Blitar 66137. Dalam penyelenggaraan skripsi di Fakultas Teknologi Informasi, Universitas Islam Balitar berusaha untuk menjunjung tinggi nilai kejujuran yang melandasi setiap kebijakan dan layanan pendidikan yang dilakukan sesuai dengan kebijakan Fakultas Teknologi Informasi, khususnya Program Studi Teknik Informatika.

Menurut UU No. 20 tahun 2003 tentang Sistem Pendidikan Nasional ("UU Sisdiknas") menjelaskan bahwa setiap perguruan tinggi harus menetapkan syarat kelulusan untuk mendapatkan gelar akademik, profesi, atau vokasi. Apabila karya ilmiah yang digunakan untuk mendapatkan gelar akademik, profesi, atau vokasi

terbukti sebagai hasil jiplakan, maka gelarnya akan dicabut. Selain itu, Menteri Pendidikan juga telah menerbitkan Permendiknas No. 17 Tahun 2010 tentang Pencegahan dan Penanggulangan Plagiat di Perguruan Tinggi yang mengatur tentang penjiplakan karya ilmiah di perguruan tinggi.

Prosentase Plagiarisme berdasarkan peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia tentang Pencegahan dan Penanggulangan Plagiat di perguruan Tinggi. Pada halaman 5 pasal 5 mengatur pencegahan plagiasi dilakukan selama mahasiswa menjalani proses pembelajaran dan menurut pasal 6 yang melakukan pencegahan adalah Pimpinan Perguruan Tinggi. Selanjutnya pada pasal 7 diperkuat dengan kewajiban setiap mahasiswa memberikan surat pernyataan bahwa karya tersebut bebas dari plagiat dan apabila di kemudian hari terbukti terdapat plagiat dalam karya ilmiah tersebut, maka penyusunnya bersedia menerima sanksi sesuai ketentuan peraturan perundang-undangan.

Khusus pada Fakultas Teknik Informatika Universitas Islam Balitar telah memiliki Buku Pedoman Penyusunan Karya Ilmiah II (Skripsi) yang menjelaskan bahwa mahasiswa wajib membuat pernyataan keaslian dokumen agar terhindar dari praktek plagiasi karya ilmiah. Namun prosentase uji plagiasi nya pada Buku Pedoman penyusunan tersebut diatur maksimal 25% (dua puluh lima prosen) untuk jenjang Sarjana. Selain itu di Universitas Islam Balitar sudah ada regulasi batasan plagiarisme tulisan ilmiah dalam bentuk Jurnal pada LPPM Universitas Islam Balitar sebagai pengelola Jurnal Supremasi milik Fakultas Hukum Unisba Blitar mengatur bahwa artikel sebelum dikirimkan harus memenuhi ketentuan : belum pernah dipublikasikan , atau belum ada sebelum jurnal lain untuk dipertimbangkan,

artikel yang diterbitkan dalam konferensi perlu diubah sebelum diserahkan dengan setidaknya 35% perbedaan dengan makalah aslinya, dan lulus pemeriksaan *reviewer*.

Berkaitan dengan peraturan Internal Fakultas Teknik Informatika Unisba yang mengatur Plagiarisme Skripsi maksimal 25% dan LPPM Unisba Menentukan maksimal plagiarisme Jurnal 35 maka Penulis menentukan batas maksimal tingkat kemiripan Judul Skripsi pada Fakultas Teknik Informatika Universitas Balitar untuk diterima sebagai Usulan Judul Tugas Akhir adalah dua puluh lima persen (25 %).

Dari penjelasan diatas perbuatan plagiat dalam penulisan karya ilmiah merupakan suatu tindak pidana yang sudah diatur oleh Undang -undang serta peraturan pelaksanaannya. Orang yang terbukti melakukan plagiat dalam penulisan karya ilmiah untuk mendapatkan gelar akademik, profesi, atau vokasi terancam sanksi pencabutan gelar, pembatalan ijazah, bahkan hingga ancaman pidana penjara.

Dari latar belakang diatas penulis ingin melakukan penelitian dengan judul **STUDI DAN ANALISIS ALGORITMA *SIMILARITY* UNTUK IDENTIFIKASI KEMIRIPAN JUDUL SKRIPSI**. Untuk mendukung penelitian ini agar dapat berjalan dengan baik diperlukan dataset dari Fakultas Teknologi Informasi Universitas Islam Balitar. Berikut jumlah skripsi yang telah dihasilkan oleh mahasiswa Program Sarjana Teknik Informatika Universitas Islam Balitar selama tahun 2019, 2020 dan 2021 ditunjukkan pada Tabel 1.1.

Tabel 1.1. Data Skripsi Jurusan Informatika Universitas Islam Balitar

No	Periode Kelulusan	Jumlah Skripsi (Data Latih)
1	2019	80
2	2020	90
3	2021	30
	Jumlah	200 (Data Set)

Dalam Penelitian ini akan melakukan evaluasi kinerja dari algoritma *Cosine Similarity* dengan *Vector Space Model* dalam melakukan proses pencocokan apakah judul yang dipilih memiliki kemiripan dengan database skripsi di Fakultas Teknologi Informasi Universitas Islam Balitar atau tidak. Untuk penerapan metode ini dibutuhkan ilmu mengenai *Natural Language Processing*. Menurut Wang dan Dong (2020) Implementasi *Natural Language Processing* dapat digunakan untuk *information retrieval*, *automatic question answering*, *machine translation*, *dialogue systems*, dan *document matching*. Namun, Menurut penelitian yang dilakukan oleh Triana (2014) dengan judul Pemanfaatan Metode *Vector Space Model* dan Metode *Cosine Similarity* pada Fitur Deteksi Hama dan Penyakit Tanaman Padi, terdapat kelemahan pada metode *Cosine Similarity* maupun *Vector Space Model* dalam melakukan pendeteksian kemiripan. Banyaknya suku kata dalam satu gejala menjadi salah satu kekurangan metode *VSM* untuk ditemukannya gejala yang relevan dengan query pada saat proses identifikasi input. Untuk menangani permasalahan tersebut penulis mencoba melakukan uji coba menggunakan bobot pada masing-masing metode untuk meningkatkan hasil akurasi pendeteksian.

Aplikasi yang digunakan untuk menguji metode *Cosine Similarity* dan *Vector Space Model* menggunakan Aplikasi Turnitin. Menurut Eka (2022) dalam penelitian yang berjudul "Persepsi Penggunaan Aplikasi Turnitin Untuk Mencegah Tindakan Plagiarisme Pada Mahasiswa Universitas Udayana", menjelaskan bahwa

penggunaan Turnitin sangat bermanfaat untuk membantu mahasiswa terhindar dari plagiat. Dengan adanya penggunaan aplikasi Turnitin, mahasiswa dapat meminimalisir kasus plagiat dan merasa lebih terbantu dalam menyelesaikan tugas akademik mereka.

Algoritma pencarian yang digunakan Turnitin bisa dikatakan sangat detail. Turnitin akan menampilkan hasil jumlah tingkat persentase plagiat yang ditemukan dari kesamaan tersebut. Turnitin akan membandingkan tulisan yang akan dideteksi dengan tulisan lain dalam database miliknya dan terlihatlah sumber artikel mana yang diambil oleh penulis sekaligus berisi sumber alamat websitenya.

Hasil evaluasi dari kinerja kedua algoritma di atas diharapkan dapat menjadi salah satu rekomendasi pembaca dalam memilih metode yang tepat untuk sistem pencocokan kemiripan judul skripsi.

1.2. Rumusan Masalah

Berdasarkan latar belakang, dapat dirumuskan perumusan masalah sebagai berikut :

- Bagaimana proses Pre-processing dilakukan pada Judul Skripsi yang akan diuji setelah diberikan nilai threshold dengan menggunakan Algoritma *Cosine Similarity* dan *Vector Space Model* secara terpisah
- Berapakah Nilai Uji Algoritma *Cosine Similarity* dan *Vector Space Model* dalam mengidentifikasi plagiasi judul skripsi menggunakan Framework Flask setelah dilakukan Pre-processing ?

- c. Bagaimana memilih nilai terdekat antara hasil uji algoritma *similarity* antara kinerja *Algoritma Cosine Similarity* dan *Vector Space Model* diperbandingkan (*benchmarking*) terhadap Uji Plagiasi Judul Skripsi dengan menggunakan Aplikasi Turnitin secara manual dengan tabulasi.

1.3. Batasan Masalah

Batasan pada penelitian ini dilakukan supaya penulisan pada penelitian lebih terfokus dan terarah sehingga tujuan dari penelitian akan tercapai. Adapun batasan masalah ini adalah:

- a. Judul yang diuji diambil dari database pengujian.
- b. Pengujian *prototype* penelitian yang telah dibuat dibandingkan (*benchmarking*) dengan Aplikasi Turnitin
- c. Penulis mengabaikan tanda baca dan huruf khusus (spesial karakter)
- d. Penulis melakukan *tokenizing* atas huruf-huruf yang ada
- e. Penulis membandingkan *Algoritma Cosine Similarity* dengan *Vector Space Model* yang digunakan untuk mengukur nilai kemiripan pada dokumen teks
- f. Penulis tidak melakukan uji kemiripan dalam bentuk gambar (*image*).
- g. Studi kasus dilakukan pada data file skripsi Program Sarjana Teknik Informatika Universitas Islam Balitar dengan jangka waktu 3 tahun terakhir
- h. Hasil penelitian dalam bentuk prototipe pengujian kemiripan naskah tesis berbasis Pemrograman *Framework* Python serta *Framework* Flask untuk tampilannya berbasis web

- i. Penelitian di running pada *PC* atau *Notebook* peneliti, tidak pada *server* dan spesifikasi mengikuti spesifikasi Laptop penulis.

1.4. Tujuan Penelitian

Tujuan dari penelitian sesuai dengan rumusan masalah yang ada adalah sebagai berikut :

- a. Menggunakan teknis metoda *semantic rule* pada proses *preprocessing data text* yang efektif yang akan digunakan sebagai bahan pemrosesan uji kesamaan (*Similarity*) naskah skripsi berbentuk teks dengan *tokenizing*.
- b. Membangun desain prototipe uji kesamaan kemiripan pada naskah skripsi Program Sarjana Teknik Informatika Universitas Islam Balitar berbasis Pemrograman Python,
- c. Mengetahui prosentase kinerja prototipe dengan algoritma *VSM* dengan menggunakan data naskah skripsi sesuai periode yang ditentukan pada batasan masalah dan diperbandingkan dengan Aplikasi Uji Kemiripan dokumen Turnitin yang sudah jamak digunakan oleh banyak organisasi termasuk perguruan tinggi.
- d. Mengetahui persentase kinerja prototipe dengan algoritma *Cosine Similarity* dengan menggunakan data naskah skripsi yang sama dan diperbandingkan dengan Aplikasi Uji Kemiripan dokumen Turnitin yang sudah jamak digunakan oleh banyak organisasi termasuk perguruan tinggi.

- e. Menyajikan rekomendasi algoritma yang cocok dalam melakukan proses analisa kemiripan naskah skripsi dengan memperhatikan banyaknya data serta kecepatan pengolahannya apabila jumlah skripsi dibuat rentang skala tertentu.

1.5. Manfaat Penelitian

Manfaat Penelitian yang diperoleh dari penelitian yang dilakukan oleh peneliti adalah :

- a. Mengetahui bahwa proses *semantic rule* merupakan proses yang terbaik untuk *preprocessing text mining* dengan *tokenizing*
- b. Mengetahui algoritma yang lebih baik dalam pengolahan data skripsi setelah dilakukan proses *preprocessing* untuk menghasilkan data yang mudah diberikan nilai
- c. Hasil nilai dari algoritma *text processing* selanjutnya diukur kemiripan (kemiripan) untuk di analisa kemiripan naskah judul skripsi yang diuji
- d. Menghasilkan statistik prosentase kemiripan naskah judul skripsi untuk dijadikan nilai minimum bagi Program Sarjana Teknik Informatika Universitas Islam Balitar
- e. Mengetahui kehandalan algoritma yang dipilih di uji dari sisi jumlah data serta waktu pemrosesan
- f. Mempermudah pihak Program Sarjana Teknik Informatika Universitas Islam Balitar menentukan judul skripsi mahasiswa dikategorikan sebagai plagiat/penjiplak atau tidak.

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Penelitian yang dilakukan oleh penulis terlebih dahulu dilakukan tinjauan pustaka dari penelitian-penelitian sebelumnya. Dari penelitian-penelitian sebelumnya penulis mendapatkan perbedaan dalam hal penggunaan metode. Adapun tujuan penulis mengambil beberapa penelitian terdahulu supaya bisa mengambil referensi terhadap penelitian tersebut serta bisa menambah pengetahuan serta memperkaya ilmu untuk bahan kajian yang dilakukan oleh penulis dalam melakukan penelitian ini.

Dibawah ini merupakan beberapa penelitian yang terkait dengan masalah yang dilakukan oleh penulis :

A. Syah, F., dkk (2022)

Pada penelitian dengan judul "Sistem Deteksi Kemiripan Judul Skripsi dengan Metode *Term Weight* dan *Word Embedding* di Program Studi Informatika Universitas PGRI Yogyakarta" yang dilakukan oleh Syah, F., dkk (2022), menunjukkan bahwa hasil pengujian deteksi tingkat kemiripan judul skripsi menunjukkan akurasi sistem dengan metode *Term Weight* dan *Word Embedding* sebesar 54%. Tujuan dari penelitian ini adalah untuk menentukan tingkat kemiripan judul skripsi mahasiswa dengan menggunakan metode *Term Weight* dan *Word Embedding*, dengan bahasa pemrograman PHP dan database *MySQL*.

B. Maake, dkk (2018)

Penelitian selanjutnya dilakukan oleh Maake, dkk (2018) dengan judul "*A Comparative Analysis of Text Similarity Measures and Algorithms in Research Paper Recommender Systems*". Penelitian ini menjelaskan bahwa penggunaan metode *Boosted Machine Learning* menunjukkan akurasi sebesar 83,2%, yang lebih baik daripada penggunaan metode *Random Forest* yang menunjukkan akurasi sebesar 80,3%. Tujuan Maake dalam melakukan penelitian ini adalah untuk menentukan kombinasi algoritma dan metrik kemiripan yang dapat digunakan untuk mengoptimalkan pencarian dan rekomendasi artikel dalam sistem rekomendasi penelitian. Penelitian ini menggunakan algoritma klasifikasi non-linear dengan matrik kemiripan teks.

C. Mutiara, dkk (2008)

Pada penelitian selanjutnya yang dilakukan oleh Mutiara, dkk (2008) dengan judul "*Aplikasi Anti Plagiarisme dengan Algoritma KARP-RABIN pada Penulisan Ilmiah Universitas Gunadarma*" menunjukkan bahwa tingkat kemiripan di atas 15% sudah dianggap mirip dengan karya sebelumnya. Tujuan dari penelitian ini adalah untuk membuat aplikasi yang dapat mendeteksi kemiripan penulisan karya ilmiah di Universitas Gunadarma menggunakan bahasa pemrograman PHP dengan metode Karp-Rabin.

D. J Wang dan Y Dong (2020)

Penelitian yang dilakukan oleh Wang dan Dong (2020) dengan judul "*Measurement of Text Similarity : A Survey*" menjelaskan bahwa Penggunaan Metode *similarity* pada sistem pendeteksi kemiripan cukup mudah diterapkan

dalam penerapan ini menghasilkan hasil yang cukup baik dari beberapa metode yakni *Cosine Similarity* dan *VSM*.

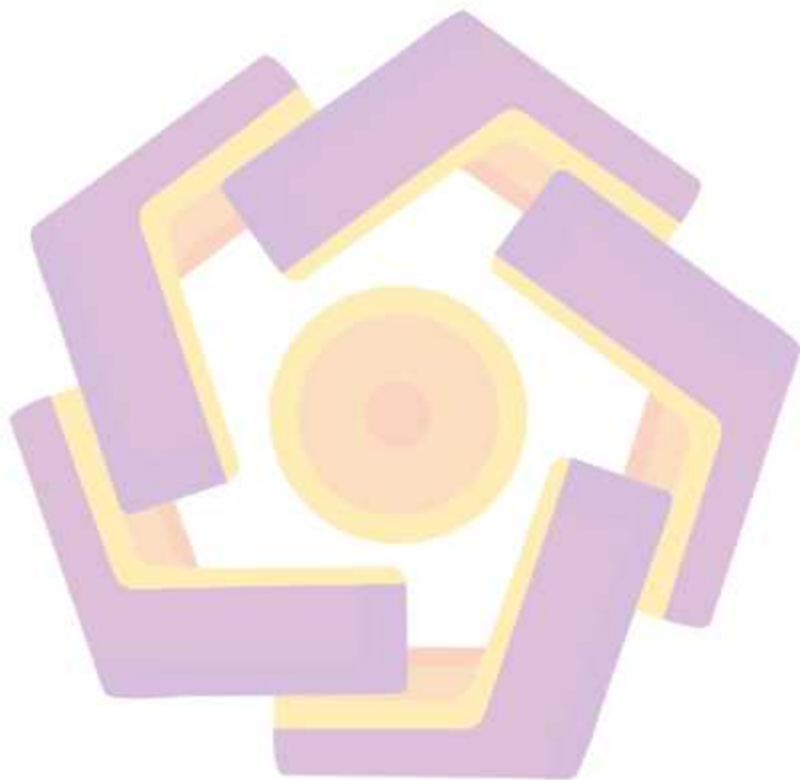
E. Rismayani, dkk (2022)

Penelitian selanjutnya dilakukan oleh Rismayani, dkk (2022) dengan judul "Implementasi Algoritma *Text Mining* dan *Cosine Similarity* untuk Desain Sistem Aspirasi Publik Berbasis *Mobile*", kombinasi metode *Text Mining* dan *Cosine Similarity* dapat digunakan untuk mengukur kesamaan fungsional masing-masing komisi dengan aspirasi yang diinput oleh masyarakat, sehingga aspirasi dapat diarahkan ke komisi yang tepat. Persyaratan untuk penetapan multikomisi adalah harus sama atau lebih besar dari 75% dari nilai maksimum. Dari kondisi tersebut, hasil untuk Komisi A dan Komisi D yang memenuhi persyaratan akan diarahkan ke Komisi A dan Komisi B

F. Didik, dkk (2019)

Dalam penelitian yang berjudul "Penerapan Algoritma *Cosine Similarity* pada *Text Mining* Terjemah Al-Qur'an Berdasarkan Keterkaitan Topik" oleh Didik, dkk (2019), *Algoritma Cosine Similarity* digunakan untuk menghitung tingkat kemiripan antara ayat Al-Qur'an dan menghasilkan beberapa kelompok ayat yang dibandingkan dengan *index* Al-Qur'an. Hasil penelitian menunjukkan bahwa tingkat kemiripan antara ayat sebesar 20% memberikan hasil terbaik pada pengelompokan *index* Al-Qur'an dengan rata-rata 46,42%. Tingkat kemiripan antara terjemah ayat Al-Qur'an sebesar 40% memberikan rata-rata sebesar 15,39% pada pengelompokan *index* Al-Qur'an.

Namun, untuk tingkat kemiripan antar ayat di atas 40%, terdapat kelompok ayat yang mirip tetapi tidak masuk dalam *index* Al-Qur'an.



2.2. Keaslian Penelitian

Tabel 2.1. Matriks literatur review dan posisi penelitian
STUDI DAN ANALISIS ALGORITMA SIMILARITY UNTUK IDENTIFIKASI KEMIRIPAN JUDUL SKRIPSI
(Studi Kasus : Fakultas Teknologi Informasi Universitas Islam Balitar)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1	Sistem Deteksi Kemiripan Judul Skripsi dengan <i>Metode Term Weight dan Word Embedding</i> di Program Studi Informatika Universitas PGRI Yogyakarta	Firdiyan Syah, Nurirwan Saputra, dan Selma Maharani. Jurnal Dinamika Informatika, 2022	Untuk menentukan tingkat kemiripan judul skripsi mahasiswa dengan menggunakan metode Term Weight dan Word Embedding, dengan bahasa pemrograman PHP dan database MySQL	Hasil pengujian deteksi tingkat kemiripan judul skripsi menunjukkan akurasi sistem dengan metode Term Weight dan Word Embedding sebesar 54%	Pada penelitian tersebut sistem yang dibuat hanya hanya memberikan persentase tingkat kemiripan judul, akan tetapi tidak memberikan rekomendasi judul yang diajukan dan aplikasi yang dibangun berbasis website.	Perbedaan terletak pada metode yang digunakan, pada penelitian sebelumnya metode yang digunakan adalah metode <i>Metode Term Weight dan Word Embedding</i> sedangkan pada penelitian yang saya lakukan metode yang digunakan adalah <i>Cosine Similarity</i>
2	A Comparative Analysis of Text Similarity Measures and Algorithms in Research Paper Recommender Systems	Maake Benard Magara, Sunday O. Ojo, and Tranos Zuva. IEEE, 2018	Tujuan Maake dalam melakukan penelitian ini adalah untuk menentukan kombinasi algoritma dan metrik kemiripan yang dapat digunakan untuk	Penelitian ini menjelaskan bahwa penggunaan metode <i>Boosted Machine Learning</i> menunjukkan akurasi sebesar 83,2%, yang lebih baik daripada penggunaan	Penggunaan pohon keputusan dalam melakukan klasifikasi teks terkadang menghasilkan kinerja yang buruk dibanding dengan algoritma clustering. oleh karena	Perbedaan terletak pada metode yang digunakan, pada penelitian sebelumnya metode yang digunakan adalah <i>boosted machine learning</i> dan <i>Random Forest</i> sedangkan pada penelitian yang saya lakukan metode

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
			<p>mengoptimalkan pencarian dan rekomendasi artikel dalam sistem rekomendasi penelitian. Penelitian ini menggunakan algoritma klasifikasi non-linear dengan matrik kemiripan teks</p>	<p><i>metode Random Forest</i> yang menunjukkan akurasi sebesar 80,3%</p>	<p>itu kami mengusulkan sistem yang akan secara otomatis mendeteksi makalah penelitian yang sedang dibaca atau dilihat oleh seorang peneliti, dan berdasarkan aktivitas pada makalah penelitian tersebut, fitur-fitur penting dari makalah diekstraksi dan digunakan untuk menghitung seberapa mirip mereka dengan dokumen lainnya. Model-model yang lebih canggih seperti analisis semantik laten akan digunakan dalam eksperimen masa depan karena mereka dapat mengklasifikasikan dokumen sebagai milik kelas yang sama bahkan ketika mereka tidak memiliki kata dan istilah yang serupa</p>	<p>yang digunakan adalah <i>Cosine Similarity</i></p>

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
3	Aplikasi Anti Plagiarisme dengan Algoritma KARP-RABIN pada Penulisan Ilmiah Universitas Gunadarma	A. Benny Mutiara, Sinta Agustina, Anzdoc, 2008	Tujuan dari penelitian ini adalah untuk membuat aplikasi yang dapat mendeteksi kemiripan penulisan karya ilmiah di Universitas Gunadarma menggunakan bahasa pemrograman PHP dengan metode Karp-Rabin	Hasil menunjukkan bahwa tingkat kemiripan di atas 15% sudah dianggap mirip dengan karya sebelumnya	Diharapkan aplikasi yang dibuat digunakan pada dunia pendidikan, aplikasi yang dibuat supaya dosen bisa memeriksa hasil tugas mahasiswanya maupun lingkungan yang lebih luas seperti proses penulisan ilmiah.	Perbedaan terletak pada metode yang digunakan, pada penelitian sebelumnya metode yang digunakan adalah KARP-RABIN sedangkan pada penelitian yang saya lakukan metode yang digunakan adalah <i>Cosine Similarity</i>
4	<i>Measurement of Text Similarity : A Survey</i>	Jiapeng Wang, dan Yihong Dong, Jurnal Information, 2020	Melakukan deteksi terhadap dokumen <i>similarity</i>	Hasil menunjukan bahwa Penggunaan Metode <i>similarity</i> pada sistem pendeteksi kemiripan cukup mudah diterapkan dalam penerapan ini menghasilkan hasil yang cukup baik dari beberapa metode yakni <i>Cosine Similarity</i> dan VSM	Dari survei ini jelas bahwa setiap metode memiliki kelebihan dan kekurangan, dan sulit untuk memilih model terbaik; namun, metode berbasis representasi teks dan jarak teks yang sesuai yang paling populer telah menunjukkan hasil yang menjanjikan dibandingkan	Perbedaan terletak pada metode yang digunakan, pada penelitian sebelumnya metode yang digunakan adalah CNN MF-LSTM sedangkan pada penelitian yang saya lakukan

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
					dengan model independen lainnya. Survei ini akan memberikan dasar yang baik bagi para peneliti untuk menemukan metode baru dalam mengukur kemiripan semantik.	digunakan adalah <i>Cosine Similarity</i>
5	Implementasi Algoritma <i>Text Mining</i> dan <i>Cosine Similarity</i> untuk Desain Sistem Aspirasi Publik Berbasis Mobile	Rismayani, Hasyrif SY, Tommy Darwansyah, Irsan Mansyur, komputika, 2022	Penelitian ini bertujuan untuk membangun sebuah aplikasi yang dapat menampung aspirasi masyarakat dan kemudian dapat mengklasifikasikannya dan meneruskannya ke komisi berbasis mobile yang sesuai, untuk menyediakan sistem yang dapat digunakan DPRD untuk memberikan masukan.	Kombinasi metode <i>Text Mining</i> dan <i>Cosine Similarity</i> dapat digunakan untuk mengukur kesamaan fungsional masing-masing komisi dengan aspirasi yang diinput oleh masyarakat, sehingga aspirasi dapat diarahkan ke komisi yang tepat. Persyaratan untuk penetapan multikomisi adalah harus sama atau lebih besar dari 75% dari nilai maksimum. Dari kondisi tersebut, hasil untuk Komisi A dan Komisi D yang memenuhi persyaratan akan diarahkan ke Komisi A dan Komisi B	Dalam pengembangan kedepannya disarankan agar sistem digunakan pada tingkat lebih tinggi lagi dengan menggunakan algoritma yang berbeda dengan hasil akurasi yang lebih baik.	Perbedaan terletak pada metode yang digunakan, pada penelitian sebelumnya metode yang digunakan menggunakan dua metode yaitu <i>Text Mining</i> dan <i>Cosine Similarity</i> sedangkan pada penelitian yang saya lakukan metode yang digunakan hanya satu yaitu <i>Cosine Similarity</i>

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
6	Penerapan Algoritma <i>Cosine Similarity</i> pada <i>Text Mining</i> Terjemah Al-Qur'an Berdasarkan Keterkaitan Topik	M. Didik R. Wahyudi, Semesta Teknika, 2019	Melakukan perancangan sistem dalam mengenali, mencari topik, dan mengelompokkan suatu permasalahan Al-Quran dengan metode <i>Cosine Similarity</i>	Hasil penelitian menunjukkan bahwa tingkat kemiripan antara ayat sebesar 20% memberikan hasil terbaik pada pengelompokan index Al-Qur'an dengan rata-rata 46,42%. Tingkat kemiripan antara terjemah ayat Al-Qur'an sebesar 40% memberikan rata-rata sebesar 15,39% pada pengelompokan index Al-Qur'an. Namun, untuk tingkat kemiripan antar ayat di atas 40%, terdapat kelompok ayat yang mirip tetapi tidak masuk dalam index Al-Qur'an.	Tingkat kemiripan tertinggi dalam pengelompokan Al-Quran masih diatas 40%, untuk kemiripan antar ayat pun masih 40%, dibutuhkan metode lain untuk mendapatkan hasil yang lebih baik.	Persamaan terdapat pada metode yang digunakan yaitu metode <i>Cosine Similarity</i> , perbedaannya terdapat pada tujuan penelitian, penelitian yang dilakukan sebelumnya melakukan perancangan untuk mencari topik serta pengelompokan ayat Al-Quran, sedangkan pada penelitian yang saya lakukan adalah untuk melakukan identifikasi kesamaan judul Skripsi.

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
7	A proposed approach for plagiarism detection in Article documents	Saeed, A. A. M., & Taqa, A. Y. <i>Sinkron: jurnal dan penelitian teknik informatika</i> , 2022	Menguji kemiripan dokumen proposal artikel dengan algoritma K-Means dan <i>Cosine Similarity</i> dengan diberikan pembobotan metoda TF-IDF.	Hasil uji kemiripan menunjukkan prosentase kemiripan pada jenis artikel tergantung pada nilai bobot.	Aplikasi sudah didesain apabila ratio kemiripan $\leq 25\%$ maka dokumen diterima dan sisanya ditolak.	Aplikasi menerima semua jenis dokumen artikel. Dengan pembobotan pada spesifikasi komputer yang sama menghasilkan kemiripan dokumen yang berbeda dan waktu pemrosesan. Pada penelitian penulis menggunakan <i>Cosine Similarity</i> dan VSM bukan <i>Cosine</i> dan K-Means
8	Analytical Study of Traditional and Intelligent textual Plagiarism Detection Approaches	Ayoub Ali and Atna Yaseen Taqa, <i>Journal of education and</i>	Penggunaan representasi numerik dokumen dalam bentuk vektorisasi yang dituangkan	Setelah menggunakan beberapa pendekatan perhitungan kemiripan dokumen, terdapat tradisional perhitungan dengan pendekatan vektor. Penggunaan	Intelligence <i>Similarity</i> lebih bagus dalam mendeteksi kemiripan namun membutuhkan waktu lebih lama pemrosesannya	Penelitian jurnal banyak menggunakan tradisional dan intelligence

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
		Science (ISSN 1812-125x), Vol.31 No. 01, 2022	dalam VSM selanjutnya di hitung kemiripannya dengan Cosine dan Jackard. Algoritma bisa dijadikan pendekatan perhitungan plagiasi secara tradisional. Penelitian ini mencoba membuat perhitungan kemiripan dengan pendekatan intelijen yakni menerapkan Convolutional Neural Network (CNN) dan Long Short Term Memory (LSTM)	intelen pendekatan perhitungan kemiripan seperti Plagiarism, Authorship, Social Software Misure (PAN), Conference and Labs of the Evaluation Forum (CLEF), Corpus of english Novel (CEN) and Open Source Arabic Corpus (OSAC). Perhitungan kemiripan tidak membutuhkan kecepatan waktu, hanya fokus pada kebutuhan nilai kemiripannya. Namun penelitian menunjukkan <i>Cosine Similarity</i> banyak dan populer digunakan sebagai perhitungan kemiripan dan memiliki catatan keberhasilan.		algoritma dan processing pada dokumen di website. Sedangkan penelitian penulis fokus pada penggunaan <i>Cosine Similarity</i> dan Vector Space Model pada Judul Skripsi
9	Optimization of Plagiarism Detection Using vector Space Model on CUDA Architecture	Cader, A. J. M. A., Gamaarachchi, H., & Ragel, R. G., International	Penelitian menawarkan software berbasis percepatan pada deteksi plagiarisim dengan menggunakan model	Dalam rangka mempercepat bersuam dengan pengembangan GPU dengan CUDA setelah menguji dataset hanya memerlukan 36detik dan kecepatan naik45x ketika	Pengujian ini sangat tergantung pada komputer cpu yang digunakan padahal variasi komputer dengan spesifikasnya banyak sekali. Sehingga faktor faktor yang	Penelitian cader fokus pada penggunaan CUDA Arsitekture untuk menguji cepat tidaknya dibanding

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
		Journal of Innovative Computing and Application (4), pp 232-244, 2022.	yang cocok sesuai CPU/GPU. Software berbasis VSM dan mencari model CPU terbaik sesuai jumlah text/dokumen yang diujikan dengan Arsitektur CUDA (Compute Unified Device Architecture).	mewnggunakan CPU biasa , namun saat optimasi dijalankan bisa kecepatan proses hanya 4s dan kecepatan menjadi 389x ketika serial CUDA di implementasikan	mempengaruhi kecepatan pemrosesan sangat banyak	tradisional processing dengan pendekatan VSM pada pengujian data set yang besar. Sedangkan penelitian penulis menguji antara cosine simailaty dan VSM mana yang lebih baik dalam pendeteksian kemiripan judul skripsi
10	Automatic Detection of Plagiarism in Writing	Davoodifard, M, Studies in Applied Linguistic & TESOL at Teacher college, Colombia	Peneliti a ini bertujuan memberikan tools langsung pada sistem algoritma deteksi kemiripan dengan lebih cepat dan mengenali ketidak normalan pada text	Penggunaan UTF-8 encoding untuk normalisasi text membantu percepatan proses stemming , lemmatization, untuk memperkecil infleksi dan adding kata.pada akhirnya VSM merupakan cara terbaik dalam mendeteksi kemiripan text	Disarankan batasan 0,2 atau 20% adalah batasan maksimal kemiripan text sebagai indikasi plagiasi.	Penelitian Davoodifard menggunakan encoding UTF-8 serta VSM yang perhitungannya <i>similaritynya</i>

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
		Unibersity, Vol.21 Hal 54,60, 2022	secara langsung. Normalisasi text menggunakan UTF-8 Encoding.			dengan <i>Cosine Similarity</i> Sedangkan penulis melakukan penelitian dengan menggunakan metode pembobotan pada proses VSM dan finalisasi penilaian kemiripan dengan pendekatan cosine.
11	Evaluation of Different Plagiarism Detection Methods : A Fuzzy MCDM Perspective	Jambi, K.M, Khan, I.H, Shiddiqi M.A, Applied Science, 2022	Penelitian ini bertujuan mengevaluasi beberapa metode deteksi plagiasi dengan perspetif MCDM (Multi Criteria Decision Making), menggunakan pendekatan NLF	Penggunaan Metoda MCDM yang memiliki sistem hirearki meningkatkan efektifitas deteksi kemiripan dokumen. Penelitian ini memberikan rekomendasi blue print penggunaan MCDM sebagai salah satu Machine Learning (ML) untuk masa depan uji plagiasi yang lebih efektif.	Penggunaan MCDM sebagai salah satu metoda pengujian kemiripan dokumen efektif namun prosesnya panjang pada pre-processing.	Penelitian ini menitikberatkan pada penggunaan MCDM pada proses preprocessing sebelum digunakan hasilnya pada berbagai macam metoda algoritma <i>similarity</i> testing. Penelitian penulis sudah jelas preprocessingnya menggunakan pendekatan NLF dan algoritma yang digunakan hanya 2 yakni <i>Cosine Similarity</i> dan Vector Space Model (VSM).

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
12	Penerapan Algoritma <i>Cosine Similarity</i> pada Aplikasi Bank Soal	Daniel Nugraha, Jurnal Sains Komputer dan Teknologi Infomarmasi Vol.3 No. 2, 2021	Penelitian riset dan pengembangan terapan dengan UML, dengan bahasa html, css, javascript, web server XAMPP dan MySQL Dan algoritma yang digunakan adalah <i>Cosine Similarity</i>	Hasilnya pengujian kemiripan dokumen dipermudah dengan adanya aplikasi berbasis web. Hasil uji menunjukkan pada bank soal terdapat kemiripan yang mutlak dan ketidak mutlak serta kemiripan antara 9,6% sd 20,1% pada pengujian 5 jenis soal pada bank data soal.	Penelitian hanya menggunakan satu algoritma yakni <i>Cosine Similarity</i>	Penelitian hanya dengan <i>Cosine Similarity</i> algoritmh. Sedangkan penulis memperbandingkan dengan 2 algoritma yakni <i>Cosine Similarity</i> dan VSM. Selain itu penulis membuat aplikasi berbasis web dengan microframework Flask dan database dalam json.
13	<i>Cosine Similarity</i> Retrieval untuk Prediksi Masa Studi Mahasiswa Berbasis Case Based Reasoning	Ramadhanif R, dkk. Jurnal Transistor Elektro dan Informatika (Transistor EI), Vol 4 No. 2, 2022	Penggunaan Case based Reasoning yang merupakan salah satu teknik penalaran berdasarkan kejadian atau catatan masa lampau, yang diperbandingkan dengan data masa kini menggunakan algoritma <i>Cosine Similarity</i> pada Fakultas Teknologi	Penggunaan metoda penalaran dengan data masa lampau untuk menguji masa studi mahasiswa masa kini menggunakan algoritma <i>Cosine Similarity</i> efektif dan hasilnya bisa mendeteksi 8,6% dan tingkat akurasi nya 91,4%.	Ternyata dalam uji coba ditemukan anomali ketepatan 99% masa studi dalam kenyataannya mahasiswa lulus dengan 10 semester, jauh dari standar yang ditetapkan. Saran perlu ada pre processing dalam	Penelitian ini mengukur ketepatan atas ramalan masa studi mahasiswa dengan pendekatan pengalaman masa lampau (CBR) dengan satu algoritma <i>Cosine Similarity</i> , Sedangkan penulis menguji kemiripan text judul skripsi atas judul

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
			Industri Unissula Semarang.		berikut data cleaning agar anomali tidak menjadikan error perhitungan dan pengambilan keputusan	skripsi yang sudah ada data latih dengan dua algoritma yakni <i>Cosine Similarity</i> dan VSM.
14	Implementasi Algoritma Text Mining dan <i>Cosine Similarity</i> untuk Desain Sistem Aspirasi Publik berbasis Mobile	Rismayani dkk, Komputika : Jurnal Sistem Komputer, Vol 9 No. 2, 2022	Penelitian ini bertujuan untuk membangun sebuah aplikasi yang dapat menampung aspirasi masyarakat dan kemudian dapat mengklasifikasikannya dan meneruskannya ke komisi yang sesuai berbasis mobile dengan metoda Text Mining dan <i>Cosine Similarity</i> .	Hasilnya mencapai lebih dari 75% mencapai sasaran yang ditetapkan yakni penyaluran aspirasi menuju pada komisi yang terkait sesuai text data latih yang disimpan.	Penggunaan kombinasi text mining dan <i>Cosine Similarity</i> dianggap masih perlu perbaikan dengan kedepan dilakukan pengujian dengan algoritma lain yang lebih relevan karena ada proses klasifikasi. Selain untuk meningkatkan akurasi ketepatan penyaluran aspirasi yang relevan.	Penelitian ini menggunakan kombinasi text mining dan algoritma <i>Cosine Similarity</i> . Namun titik berat tujuannya adalah menghasilkan klasifikasi yang relevan dengan akurasi yang tinggi. Sedangkan penelitian penulis adalah menguji sebuah kalimat text dicari kemiripan yang paling baik dengan text kalimat yang ada pada data latih dengan dua algoritma yakni <i>Cosine Similarity</i> dan VSM.

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
15	Text Summarization pada Artkel Berita Menggunakan Vector Space Model dan <i>Cosine Similarity</i>	Utomo, M.S dkk, <i>Dinamika Informatika</i> , Vol.14, No.1, 2022	Penelitian mencoba meringkas sebuah artikel yang panjang dengan otomatis menggunakan metoda Vector Space Model untuk memberikan bobot pada kata dasar artikel dan <i>Cosine Similarity</i> untuk mengukur <i>similarity</i> -nya.	Hasil dari proses peringkasan berupa sebuah paragraf yang diambil dari beberapa kalimat yang memiliki nilai kemiripan dengan judul paling tinggi. Dari 104 kalimat yang diuji pada judul ada 5 kalimat yang paling mirip atau 5% yang dijadikan <i>summary</i> atau ringkasan.	Pada artikel bab 5 kesimpulan algoritma yang digunakan adalah <i>Jaccard Similarity</i> padahal di abstraksi disebutkan menggunakan VSM dan <i>Cosine Similarity</i> . Sehingga penelitian menjadi bias apalagi disebutkan menghasilkan kelas dari suatu kalimat.	Penelitian ini menggunakan algoritma <i>Cosine Similarity</i> dan VSM serta <i>Jaccard</i> namun tidak jelas pastinya. Untuk membuat resume artikel berita berdasarkan judul yang ditentukan. Penulis membuat penelitian menguji judul yang diinput paling mirip dengan judul sebelumnya di data latih dengan 2 metoda pasti yakni <i>Cosine Similarity</i> dan VSM.
16	Pengukuran Kemiripan makna Menggunakan <i>Cosine Similarity</i> dan Basis Data Sinonim Kata	Sanjaya, Ardi dkk. <i>Jurnal Teknologi Informasi dan Ilmu</i>	Penelitian ini bertujuan memberikan alternatif dalam menguji kemiripan makna antar 2 kalimat. Dengan membentuk database sinonim kata serta	Rata-rata kemiripan pada penggunaan ID sinonim sebagai vektor hitung adalah 94,48% dan rata-rata nilai kemiripan pada metode atau	Dengan membuat sinonim database sebagai vektor perbandingan menyebabkan kualitas kemiripan menjadi	Penelitian ini membuat database sinonim dulu lalu membandingkan kemiripan makna dengan <i>Cosine Similarity</i> dan hasilnya tidak maksimal

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
		Komputer (JTIK) Vol 10 No.4, 2023	menggunakan algoritma <i>Cosine Similarity</i> .	alur pembandingan sebesar 69,96%.	lebih tinggi yang mana maksimal menyentuh kemiripan 30,04 % karena pada prosesnya sudah dikendalikan dengan database sinonim yang menghasilkan kemiripan 94,48%.	karena tergantung isi database sinonim. Sedangkan penulis membuat penelitian membandingkan judul skripsi dengan database judul skripsi yang memiliki variasi sangat banyak . metoda menggunakan VSM dan <i>Cosine Similarity</i> dimana VSM lebih baik dari sekedar <i>Cosine Similarity</i> saja sebagai metoda uji kemiripannya.
17	Aplikasi Pencarian hadis Menggunakan Vector Space Model dengan Pembobotan TF-IDF dan Confix Stripping Stemmer	Yunita, Novi Prima. Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK) Vol. 10 No. 2 2023	Membangun aplikasi pencarian hadis yang dapat digunakan untuk menemukan hasil pencarian yang relevan dengan keyword pencarian. Aplikasi dibangun dengan metoda retrieval informasi menggunakan VSM dan pembobotan TF-IDF menghasilkan vektor. Selanjutnya di stemming dengan Confix-Strimming Stemmer yang text	Penggunaan metoda stemmer dan pembobotan hasil VSM menghasilkan tingkat keberhasilan stemmer 78,6% dan matrix word yang dihasilkan dengan ordo 673 x 162. Hasil pengujian keyword yang diketik dapat menghasilkan kemiripan pencarian dokumen dengan	Aplikasi yang dibangun berbasis web namun dengan pendekatan Monolingual Information Retrieval (MIR) dimana dalam hadis terdapat kombinasi bahasa latin dan arabic. Sehingga dimungkinkan adanya error rate yang tidak kecil.	Penelitian ini menggunakan proses pre processing yang spesifik yakni Confix Stripping Stemmer sebelum dihasilkan vektor kata dan matrix dimensi oleh algoritma VSM. Matrik berisi hasil pembobotannya yang selanjutnya digunakan untuk data latih uji <i>similarity</i> . data latih tidak akan berubah atau berkembang karena datanya sumbernya tetap. Penelitian penulis fokus pada memperbandingkan satu kalimat judul entry data pada aplikasi berbasis web

Tabel 2.1. Lanjutan

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
			akhirnya diuji kemiripannya dengan Inner Product Similarity measurement.	nilai inner produk yang tertinggi.		dengan dua metoda algoritma yakni <i>Cosine Similarity</i> dan VSM.
18	Aplikasi Ruang Tugas dengan Deteksi Kemiripan Teks pada Dokumen Tugas Menggunakan <i>Cosine Similarity</i>	Prismadana, TA. Jurnal Informatika & Multimedia, Vol. 15 No. 1, 2023	Menggunakan <i>Cosine Similarity</i> untuk menguji kemiripan tugas-tugas mahasiswa dengan data artikel dan studi kepustakaan.	Hasil dari penelitian ini adalah aplikasi yang memberikan hasil pengecekan kemiripan text dengan akurasi tinggi dan waktu yang lebih efisien dibandingkan metoda manual. Hasil pengujian adalah mengarahkan inputan data ke halaman output yang sesuai data inputan.	Tidak dijelaskan ruang lingkup datanya dan sumber data yang dijadikan data latih. Metoda pengujian dengan blackbox Perlu diukur efisiensi dan efektifitas terhadap perbedaan jika diuji dengan metoda blackbox atau whitebox atau greybox yang sudah ditentukan dulu data latih tujuan pengujian.	Penelitian ini tidak menghasilkan prosentase kemiripan namun langsung melakukan jump page pada halaman yang sesuai dengan topik, kalimat atau artikel yang di masukkan. Sedangkan penulis melakukan penelitian pada penggunaan dua algoritma kemiripan yakni <i>Cosine Similarity</i> dan vector space model untuk membantu memetakan kemiripan dokumen yang paling efektif dan efisien dalam bentuk prosentase kemiripan.

2.3. Landasan Teori

2.3.1. Algoritma *Cosine Similarity*

Cosine Similarity merupakan salah satu metode yang digunakan untuk menghitung kemiripan dua buah *vektor*. Amrizal, (2018). Pengukuran metode ini dilakukan dengan menghitung nilai sudut cosinus antara dua *vektor* yang merepresentasikan dokumen dan query. Laxmi et al, (2019). *Cosine Similarity* dapat diterapkan untuk menghitung nilai kemiripan antar kalimat dan merupakan teknik untuk mengukur kemiripan teks populer.

Perhitungan *Cosine Similarity* menggunakan persamaan 1 berikut:

$$\cos \alpha = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (1)$$

Keterangan :

A = Vektor A, yang akan dibandingkan kemiripannya

B = Vektor B, yang akan dibandingkan kemiripannya

A • B = dot product antara vektor A dan vektor B

|A| = panjang vektor A

|B| = panjang vektor B

|A||B| = cross product antara |A| dan |B|

Metode pengklasifikasian yang digunakan pada sistem ini adalah dengan cara membandingkan kesamaan atau similaritas antara judul dokumen dengan kata kunci pertama, kemudian cara membandingkan kesamaan atau similaritas antara judul dokumen dengan kata kunci kedua, begitu seterusnya hingga kata kunci kedelapan. Kemudian dicari jumlah similaritas yang tertinggi antara kedelapan kata kunci. Selain *Cosine Similarity* terdapat pula istilah *Cosine coefficient* sebenarnya *cosien coefisient* adalah istilah yang umumnya digunakan secara seragam dengan *Cosine Similarity*. Dalam konteks *Cosine Similarity*, *coefficient* biasanya mengacu pada nilai hasil perhitungan *Cosine Similarity*. Adapun untuk penggunaan dalam

literatur dan implementasi, istilah "*Cosine Similarity*" lebih umum digunakan daripada *cosine coefficient*, tetapi keduanya dapat merujuk pada konsep yang sama. Jadi, secara umum, *Cosine Similarity* dan *cosine coefficient* mengacu pada konsep yang sama, yaitu mengukur sejauh mana dua vektor berada dalam hubungan kosinus satu sama lain. Nilai yang lebih tinggi menunjukkan tingkat kesamaan atau korelasi yang lebih besar antara vektor tersebut.

2.3.2. Pembobotan *Cosine Similarity*

Pada dasarnya, pembobotan dalam *Cosine Similarity* dilakukan dengan memberikan bobot pada masing-masing dimensi vektor sebelum menghitung *Cosine Similarity*. Ini dapat diwujudkan dengan mengalikan masing-masing nilai dalam vektor dengan bobot yang sesuai. Misalnya, jika terdapat vektor $A=[a_1, a_2, a_3]$ dan bobotnya adalah $w=[w_1, w_2, w_3]$, maka vektor yang dihasilkan setelah pembobotan ($A_{weighted}$) adalah:

$$A_{weighted}=[a_1 \cdot w_1, a_2 \cdot w_2, a_3 \cdot w_3] \quad (4)$$

Keterangan :

- $A_{weighted}$ = Vektor hasil pembobotan
- A_i = Nilai dalam dimensi ke-i dari vektor A
- w_i = Bobot yang diberikan pada dimensi ke-i dari vektor A

Setelah itu, *Cosine Similarity* dapat dihitung menggunakan vektor $A_{weighted}$ dan vektor lainnya. Adapun untuk makna perbedaan bobot dari tiap nilai yang digunakan dapat dilihat pada tabel 2.2 berikut ini :

Tabel 2.2 Makna Pembobotan

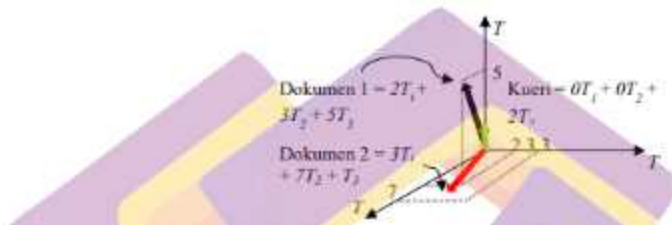
Pengujian	Bobot nilai	Keterangan
Judul skripsi 1	0.9	Pengujian judul skripsi dengan bobot nilai 0.9
Judul skripsi 2	0.8	Pengujian judul skripsi yang sama dengan bobot 0.8
Judul skripsi 3	0.7	Pengujian judul skripsi dengan bobot 0.7

Pada tabel 2.2 diatas makna dari pembobotan yang digunakan Uji 1 menggunakan bobot 0.9 untuk term judul skripsi yang diuji, yang menunjukkan bahwa dalam konteks ini, kata-kata dalam judul dianggap sangat penting dan memberikan kontribusi besar dalam mengukur kesamaan dokumen. Uji 2 menggunakan bobot 0.8, menunjukkan bahwa pada uji ini, kontribusi term judul skripsi sedikit berkurang dibandingkan dengan Uji 1, tetapi masih dianggap memiliki tingkat kepentingan yang tinggi. Uji 3 menggunakan bobot 0.7, menunjukkan bahwa pada uji ini, term judul skripsi diberikan bobot yang lebih rendah dibandingkan dengan Uji 1 dan Uji 2. Kontribusinya dianggap lebih rendah dalam mengukur kesamaan dokumen.

2.3.3. Algoritma *Vector Space model*

Vector Space Model merupakan metode yang dilakukan setelah proses pemberian nilai bobot pada term atau kata, di mana kumpulan dokumen direpresentasikan ke dalam sebuah matriks vektor. Matriks tersebut akan dapat menghasilkan sebuah koordinat tertentu. Kumpulan kata-kata (*string*) ditransformasi ke dalam representasi gambar yang sesuai pada pengklasifikasian teks di dalam dokumen untuk proses learning untuk disusun menjadi rangkaian kata. Pengukuran nilai kesamaan atau kemiripan antar dokumen dapat

menggunakan *Cosine Similarity*. *Query* dokumen dikatakan semakin relevan ketika nilai kesamaan antara *query* dan vektor dokumen juga semakin besar. Fauzi (2018). Representasi dokumen dan query pada *Algoritma Vector Space Model* ditunjukkan pada Gambar 2.1.



Gambar 2.1. Metode VSM

Kemiripan antara kedua vektor *query* dan dokumen dapat dihitung dengan melihat sudut yang paling kecil. Dua buah vektor membentuk suatu sudut yang dapat dihitung dengan melakukan *inner product*. Cara kerja *Vector Space Model* dimulai dengan *case folding*, *cleaning data*, *indexing*, *filtering*, *stemming*, dan tokenisasi yaitu tahap pemotongan *string input* berdasarkan tiap kata yang menyusunnya dan memecah dokumen ke dalam tabel frekuensi kata. Seluruh kata dalam dokumen dibentuk menjadi satu yang disebut sebagai *term*. Tiap dokumen ditampilkan sebagai vektor yang akan dibandingkan dengan term yang telah dibentuk. *Similarity Analysis* untuk mengukur kemiripan dokumen dilakukan dengan menghitung cosinus jarak antara dokumen tersebut.

Berikut rumus dari *Vector Space Model* dengan pendekatan *Cosine Similarity* yang ditunjukkan pada persamaan 2.

$$\text{Sim}(D, D_i) = \cos \theta = \frac{D \cdot D_i}{|D||D_i|} = \frac{\sum W_{q,j} W_{i,j}}{\sqrt{\sum W_{q,j}^2} \sqrt{\sum W_{i,j}^2}} \quad [2]$$

Di mana:

- D : Dokumen acuan
- D_i : Dokumen ke-i
- W_{q,j} : Bobot term j pada dokumen acuan
- W_{i,j} : Bobot term j pada dokumen i

2.3.4. Pengukuran nilai kemiripan

Untuk mendapatkan tingkat persentase kesamaan sebuah dokumen dengan dokumen lain menggunakan persamaan *Cosine Coefficient* yang ditunjukkan pada persamaan 3 :

$$\text{Similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \quad [3]$$

Untuk menentukan jenis kesamaan antara dokumen yang diuji, ada 5 jenis penilaian persentase kemiripan Mutiara (2008):

- a. 0% : Hasil uji 0% berarti kedua dokumen tersebut benar-benar berbeda baik dari segi isi dan kalimat secara keseluruhan
- b. < 15% : Hasil uji 1-14% berarti kedua dokumen tersebut hanya mempunyai sedikit kesamaan
- c. 15-50%: Hasil uji 15-50% berarti menandakan dokumen tersebut termasuk plagiat tingkat sedang
- d. >50% : Hasil uji lebih dari 51-99% berarti dapat dikatakan bahwa dokumen tersebut mendekati penjiplakan

- e. 100% : Hasil uji 100% menandakan bahwa dokumen tersebut adalah plagiat karena dari awal sampai akhir mempunyai isi yg sama persis.

2.3.5. Turnitin

Turnitin adalah salah satu platform yang dibuat untuk mendeteksi plagiarisme pada skripsi, tesis, maupun disertasi. Aplikasi ini diciptakan oleh iParadigms LLC. Saat pertama kali diluncurkan pada tahun 1997. Turnitin terhubung dengan dosen dan mahasiswa untuk melakukan scanning karya ilmiah dengan dosen yang bersangkutan. Aplikasi turnitin tidak hanya melakukan pengecekan similaritas, namun turnitin juga menyediakan layanan berupa manajemen aplikasi secara lebih terstruktur. Menurut Penelitian yang dilakukan oleh Rama, dkk (2022) dengan judul Analisis Penggunaan *Software* Turnitin sebagai Alat Pendeteksi Plagiarisme menjelaskan bahwa penelitian ini dibuat untuk mengetahui persepsi mahasiswa terhadap penggunaan *software* turnitin agar mencegah tindak plagiarisme di Universitas Udayana. Objek penelitian pada penelitian ini yaitu mahasiswa Universitas Udayana. Metode penelitian yang digunakan di penelitian ini yaitu penelitian kuantitatif dengan analisis deskriptif. Data yang digunakan pada penelitian ini adalah sebanyak 100 orang responden. Hasil dari penelitian ini responden menyatakan bahwa *software* turnitin sangat bermanfaat serta membantu mahasiswa untuk meminimalisir tindak plagiarisme.

Untuk menggunakan aplikasi ini pelanggan diwajibkan untuk berlangganan. Untuk menggunakan aplikasi ini pengguna dapat mengakses langsung dari situs

turnitin.com. aplikasi ini bekerja dengan cara melakukan proses scanning terhadap ide pengguna dan membandingkan kemiripan terhadap ide atau teks yang ada pada database turnitin dan seluruh ide yang beredar di internet.

Penggunaan turnitin pada penelitian ini adalah sebagai Benchmarking terhadap *prototype* aplikasi yang telah dibuat oleh penulis menggunakan 2 metode yang berbeda yakni *Algoritma Cosine Similarity* dan *Vector Space Model*.

2.3.6. Pemrograman Python

Python merupakan Bahasa pemrograman yang digunakan untuk berbagai macam keperluan pengembangan software. Kelebihan dari Bahasa pemrograman ini salah satunya adalah memungkinkan Bahasa pemrograman ini dapat berjalan pada multi *platform* (*Windows / Linux / Mac*). Penulisan sintaks yang lebih fleksibel dibandingkan dengan bahasa pemrograman lainnya membuat *Python* menjadi mudah dipelajari.

Bahasa pemrograman Python ini memiliki efisiensi tinggi untuk struktur data level tinggi, pemrograman berorientasi objek lebih sederhana tetapi efektif, dapat bekerja pada multi platform, dan dapat digabungkan dengan bahasa pemrograman lain untuk menghasilkan aplikasi yang diinginkan. Hal ini didukung oleh pernyataan Carbonelle (2016) yang menjelaskan bahwa Python merupakan Bahasa pemrograman yang mudah dipelajari, *literature* yang tersedia juga banyak dan mengikuti tren saat ini. Berdasarkan survei yang dilakukan oleh Pierre Carbonelle pada tahun 2016 menunjukkan bahwa bahasa pemrograman Python

berada pada peringkat ke-2 setelah Java, dan *trend* ini terus meningkat sebesar 4% dalam 10 tahun terakhir.

Menurut pendapat Saragih (2018), Python memiliki beberapa keunggulan. *Keunggulan* dari Python meliputi:

1. Python mudah digunakan

Python merupakan bahasa pemrograman tingkat tinggi, artinya Bahasa Python lebih mendekati bahasa manusia daripada bahasa mesin. Python juga memiliki struktur dasar yang mengharuskan pengguna untuk meng-include sistemnya sebelum membuat program. Perintah-perintah yang digunakan di Python menggunakan bahasa Inggris yang digunakan sehari-sehari seperti *print, input, export, dll.*

2. Kompabilitas dan Kemampuan Tinggi

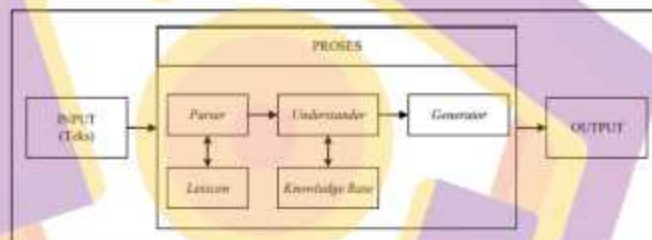
Python memiliki kemampuan yang tinggi dan mampu menciptakan aplikasi yang sederhana hingga aplikasi yang kompleks. Bahasa pemrograman Python mendukung pemrograman berbasis grafis (*GUI Programming*). Python juga memiliki kelebihan berupa pengalokasian memori secara dinamis.

3. Platform Independent

Maksudnya yaitu *program* yang dibuat bisa dijalankan di sistem operasi apa saja selama di sistem operasi tersebut terdapat *platform* Python (*Interpreter Python*).

2.3.7. NLP

Natural language processing (NLP) merupakan sebuah teknik pemrograman dimana komputer dapat memahami dan memberikan luaran dalam bentuk Bahasa manusia atau memudahkan komunikasi manusia dengan mesin Iswandi,dkk (2013). Tujuan *NLP* adalah memberikan jawaban atau respon yang sesuai pemahaman makna Bahasa manusia yang dilakukan oleh mesin Alamanda,dkk (2016). Penggunaan *NLP* sudah diterapkan di berbagai bidang kehidupan manusia. Hal ini dikarenakan *NLP* lebih mudah digunakan untuk menjadi tampilan antar muka komputer dibandingkan mempelajari Bahasa perintah komputer.



Gambar 2.2. Elemen NLP

Gambar 2.2 menjelaskan bahwa elemen dalam *natural language processing* yaitu *parser*, *lexicon*, *understander*, *knowledge base*, dan *generator*. *Parser* merupakan bagian yang mengidentifikasi setiap kata. *Lexicon* merupakan kumpulan kata yang dikenali oleh program, *Understander* merupakan bagian yang menentukan makna dari sebuah kalimat. *Knowledge base* merupakan basis pengetahuan yang berisi kata maupun frasa. *Generator* adalah luaran yang dihasilkan berdasarkan masukan yang telah diolah. Lisangan (2013).

2.3.8. Flask

Flask merupakan *web framework* yang ditulis dengan Bahasa pemrograman python dan tergolong sebagai jenis *micro framework*. Flask berfungsi sebagai *framework* aplikasi dan tampilan pada suatu *web*. Dengan menggunakan flask dan Bahasa python, pengembang dapat membuat sebuah *web* yang terstruktur dan dapat mengatur *behavior* suatu *web* dengan lebih mudah. Logo flask *framework* dapat dilihat pada Gambar 2.3.



Gambar 2.3. Logo Flask

Penggunaan *framework* ini bertujuan untuk mempercepat pembangunan aplikasi karena dalam flask sudah disediakan *library* dan kumpulan kode program yang siap digunakan untuk membangun sebuah *web app* tanpa harus membuatnya dari awal. Selain itu jika menggunakan flask maka akan mengurangi penggunaan *resource memory* karena flask tergolong dalam *micro-framework*.

Microframework disini berarti bahwa Flask bermaksud untuk membuat *core* dari aplikasi ini sesederhana mungkin tapi tetap dapat dengan mudah ditambahkan. Dengan begitu, fleksibilitas serta skalabilitas dari Flask dapat dikatakan cukup tinggi dibandingkan dengan *framework* lainnya. *Web framework* Flask ditulis menggunakan bahasa Python, sehingga sebelum Flask dapat digunakan, maka *developer* harus menginstall Python pada perangkat yang akan digunakan. Oleh

sebab itu, *web developer* yang akan menggunakan Flask sebagai *web framework* untuk *web development* harus setidaknya mempelajari bahasa pemrograman Python terlebih dahulu, sebelum dapat menggunakan Flask seutuhnya.

Dalam melakukan instalasi Flask pada sebuah perangkat, dibutuhkan PIP yang biasanya sudah terinstal pada Python versi 3.4 ke atas. PIP adalah sebuah *package management system* yang biasa digunakan untuk mengatur dan menginstall *package* yang berisi modul-modul Python. PIP digunakan untuk menginstall Flask karena Flask ditulis dan dikembangkan dengan bahasa dan *modul-modul* pemrograman Python. Dengan menggunakan PIP, semua hal yang dibutuhkan untuk instalasi Flask akan diunduh dan dipasang dalam satu perintah.

2.3.9. Diagram Fishbone

Diagram fishbone merupakan salah satu metode untuk melakukan analisis terhadap suatu masalah. Diagram ini lebih dikenal sebagai diagram sebab akibat atau *cause effect diagram*. Penulis memilih diagram fishbone dikarenakan memudahkan penulis dalam memecahkan permasalahan terkait kebutuhan dalam membangun dan menyelesaikan penelitian terkait system pendeteksi kemiripan judul skripsi pada studi kasus Universitas Islam Balitar mulai dari kebutuhan mencari studi literatur, kebutuhan alat dan desain sistem yang dipersiapkan, implementasi, pengujian yang dilakukan dan hasil penelitian yang didapat

2.3.10. Confusion Matrix

Salah satu cabang disiplin ilmu *artificial intelligence (AI)* yang membahas tentang bagaimana suatu sistem dibangun berdasarkan pada suatu data adalah *Machine Learning*. *Machine learning* adalah proses mesin/komputer untuk belajar

dari suatu data (*learn from data*). Bila tidak ada data, maka mesin/komputer tidak dapat belajar. Teknik yang ada pada *machine learning* ada beberapa macam diantaranya adalah *supervised learning* dan *unsupervised learning*. *supervised learning* biasanya dikenal dengan klasifikasi, klasifikasi merupakan model prediksi dimana hasil prediksinya bersifat diskrit hasil dari pembagian kelas-kelas yang dibangun. Untuk mengukur performa atau akurasi dari model klasifikasi maka digunakan *confusion matrix*. *Confusion Matrix* adalah pengukuran akurasi untuk *machine learning* berupa klasifikasi, dimana hasil *output* bisa berupa dua *class* (kelas) ataupun lebih. *Confusion Matrix* adalah tabel dengan empat kombinasi berbeda dari nilai prediksi dan nilai aktual. Terdapat empat istilah yang merupakan representasi hasil proses klasifikasi pada *confusion matrix*, yaitu *True Positive*, *False Positif*, *True Negatif*, dan *False Negatif*.

True Positive (TP) bisa dikatakan bahwa pengguna memprediksi suatu hal akan bernilai positif dan ketika terjadi ternyata hal itu benar terjadi. Contohnya seorang perempuan melihat laki-laki yang sedang memegang perut dan kelihatan kesakitan kemudian perempuan tersebut menebak bahwa laki-laki itu penyakit maag nya sedang kambuh, dan ketika ditanya oleh temannya ternyata benar laki-laki itu sedang kambuh penyakit maag nya.

True Negative (TN) bisa dikatakan bahwa pengguna memprediksi suatu hal akan bernilai negatif dan ketika terjadi ternyata hal itu benar terjadi. Contohnya seorang perempuan melihat laki-laki sedang memegang perut dan terlihat kesakitan, wanita tersebut memprediksi atau menebak bahwa laki-laki itu pasti

tidak sedang kesakitan karena datang bulan, ketika ditanya ternyata benar dia kesakitan bukan karena datang bulan karena dia adalah seorang laki-laki.

False Positive (FP)) bisa dikatakan bahwa pengguna memprediksi suatu hal akan bernilai positif dan ketika terjadi ternyata hal itu tidak terjadi. Contohnya seorang perempuan melihat laki-laki yang sedang memegang perut dan kelihatan sedang kesakitan kemudian perempuan tersebut menebak bahwa laki-laki itu penyakit maag nya sedang kambuh, dan ketika ditanya oleh temannya ternyata laki-laki itu sedang kelaparan karena belum makan dari pagi.

False Negative (FN)) bisa dikatakan bahwa pengguna memprediksi suatu hal akan bernilai negatif dan ketika terjadi ternyata hal itu salah. Contohnya seorang perempuan melihat laki-laki yang sedang memegang perut dan kelihatan kesakitan kemudian perempuan tersebut menebak bahwa laki-laki itu tidak sedang sakit maag, dan ketika ditanya oleh temannya ternyata laki-laki itu sedang kambuh penyakit maag nya dan malahan maagnya kronis.

Nilai Prediksi adalah hasil dari *output* program dimana nilainya *output* nya berupa nilai Positif dan nilai Negatif sedangkan nilai Aktual adalah nilai sebenarnya dimana nilainya adalah bernilai *True* dan *False*.

BAB III

METODE PENELITIAN

3.1. Jenis, Sifat, dan Pendekatan Penelitian

3.1.1. Jenis Penelitian

Jenis penelitian yang digunakan oleh peneliti merupakan penelitian Eksperimen, dimana Penelitian yang dilakukan dengan menerapkan serangkaian tindakan untuk membuktikan suatu konsep. Dalam studi kasus ini, penulis membandingkan kinerja antara *Algoritma Cosine Similarity* dan *Algoritma Vector Space Model* dengan menggunakan data uji 170 Judul Skripsi Jurusan Teknik Informatika Universitas Islam Balitar. Hasil kinerja tersebut kemudian dibandingkan dengan aplikasi Turnitin sebagai kalibrator pengujian.

3.1.2. Sifat Penelitian

Sifat penelitian yang akan dilakukan adalah penelitian secara eksperimental dimana peneliti melakukan Sebuah eksperimen dilakukan oleh peneliti untuk menguji performa penerapan *Algoritma Vector Space Model* dan *Cosine Similarity* dalam mendeteksi tingkat kemiripan judul skripsi mahasiswa sarjana Teknik Informatika Universitas Islam Balitar. Pengujian dilakukan dengan menggunakan aplikasi Turnitin untuk membandingkan akurasi dan efisiensi kedua algoritma tersebut.

3.1.3. Pendekatan Penelitian

Pendekatan kuantitatif digunakan peneliti dimana penelitian akan melakukan penelitian sesuai alur yang telah peneliti lakukan.

3.2. Metode Pengumpulan Data

Data tersebut kemudian dikumpulkan dalam bentuk file json. Selanjutnya, dilakukan proses *case folding* untuk mengubah huruf kapital menjadi huruf kecil. Setelah itu, dilakukan proses *tokenizing* untuk memecah dokumen menjadi kumpulan kata, menghilangkan karakter spesial, dan memisahkan per spasi. Tahap selanjutnya adalah *filtering/stopword removal* untuk menghilangkan kata yang tidak diperlukan dengan menggunakan *stoplist*. Jika kata yang ditemukan termasuk pada daftar *stoplist*, maka kata tersebut dianggap tidak penting atau tidak menjadi *keyword*.

3.3. Metode Analisis Data

Analisis data dilakukan setelah melakukan pengumpulan 200 sebagai dataset yang dibagi 170 sebagai data latih (*training data*) dan 30 judul sebagai data uji (*testing data*) untuk menguji kinerja algoritma *similarity Cosine* dan *VSM* dalam mengidentifikasi Kemiripan Judul Skripsi Program Studi Teknik Informatika Universitas Islam Balitar. Untuk melakukan proses data uji Mahasiswa akan melakukan penginputan untuk judul skripsi yang akan diajukan. Setelah itu data uji pengajuan judul skripsi akan dicocokkan dengan 170 data *training skripsi* apakah terdapat kemiripan atau tidak. Hasil dari kinerja tersebut akan dibandingkan

dengan aplikasi turnitin sebagai pembanding hasil kinerja metode *Cosine Similarity*, *Vector Space Model* dengan Algoritma dari aplikasi Turnitin dalam mendeteksi kemiripan skripsi dengan menggunakan data set yang sama.

3.4. Alur Penelitian

Pada penelitian tentunya memiliki alur untuk memudahkan penulis dalam melakukan penelitian. Alur dalam menyusun penelitian ini menggunakan diagram fishbone yang terdapat pada Gambar 3.1. Untuk alur pengujian metode mengenai *Cosine Similarity* dan *Vector Space Model* ditunjukkan pada Gambar 3.2:



Gambar 3.1. Alur Penelitian

Tahapan- tahapan yang ada pada gambar 3.1 diatas selanjutnya dijabarkan dalam penelitian sebagai berikut :

a. Tahap Studi Literatur

Pada tahap ini dilakukan pengumpulan jurnal, buku mengenai penelitian terkait: *Metode Cosine Similarity*, *Vector Space Model*, Bahasa pemrograman Python, *Framework Flask Python*, dan Turnitin.

b. Analisa Desain

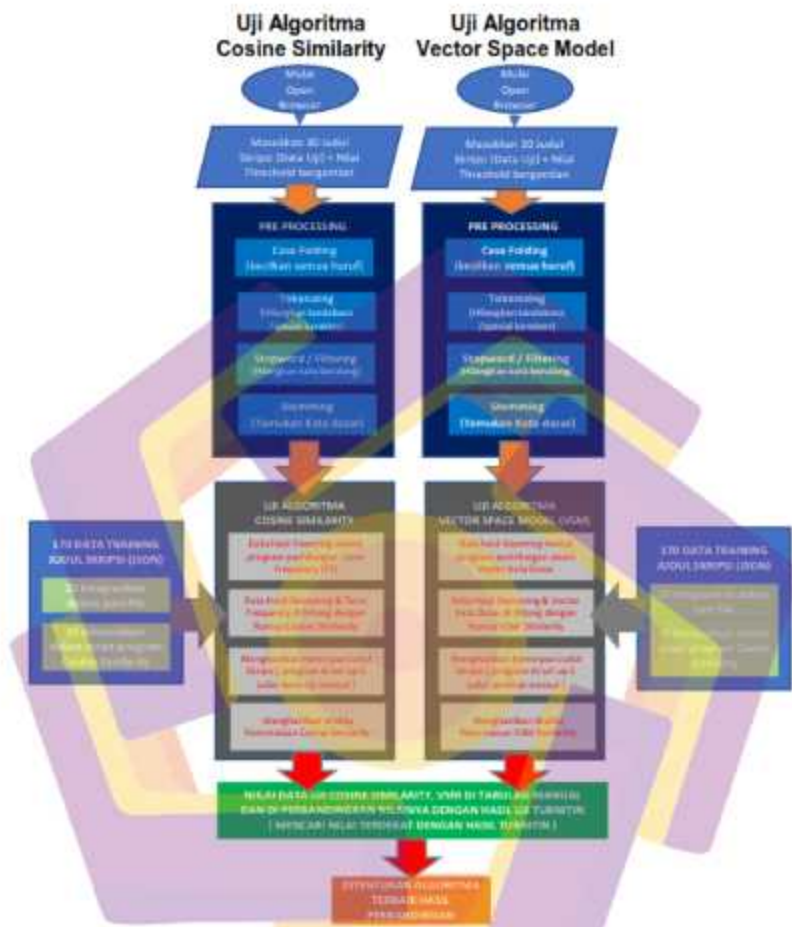
Pada tahap ini dilakukan analisis mengenai desain sistem deteksi kemiripan judul skripsi, dan melakukan analisis mengenai Aplikasi Turnitin sebagai Uji Validasi terhadap sistem deteksi yang telah dibuat penulis menggunakan *Cosine Similarity* dan *Vector Space Model*.

c. Implementasi

Pada tahap ini dilakukan implementasi sistem deteksi menggunakan metode *Cosine Similarity* dan *Vector Space Model*.

d. Pengujian

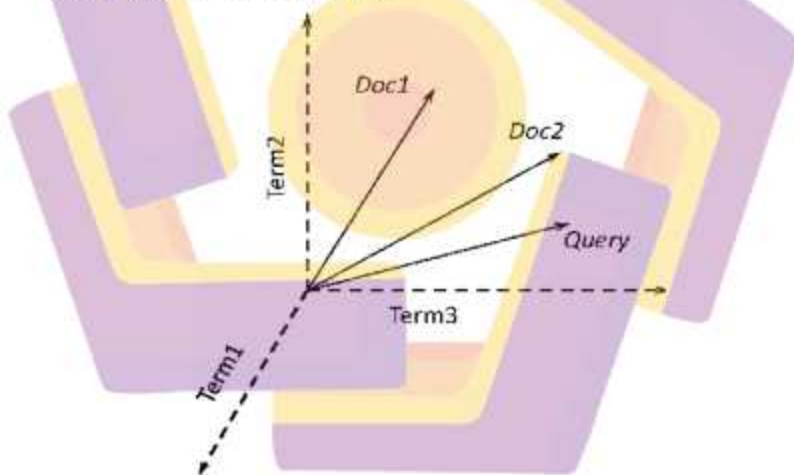
Setelah aplikasi berhasil dibuat, dilakukan proses pengujian aplikasi mengenai kinerja deteksi kemiripan antara metode *Cosine Similarity* dengan *Vector Space Model*. Hasil kinerja akan dibandingkan dengan Aplikasi Turnitin Sebagai Alat uji validasi.



Gambar 3.2. Alur Pengujian Metode *Vector Space Model* dan *Cosine Similarity*

Dari gambar 3.2 di atas merupakan alur pengujian penggunaan metode *Cosine Similarity* dan *alur metode VSM* dimulai pemberian Nilai Threshold atau pembobotan dari pembacaan judul skripsi kemudian dilanjutkan dengan melakukan *preprocessing text* dari judul skripsi yang telah dimasukkan kedalam sistem. Setelah data judul skripsi dilakukan *preprocessing* dan dilakukan pengujian dengan masing-masing metode yang digunakan yaitu *metode VSM* dan metode

Cosine Similarity. Hasil dari pengujian masing-masing metode selanjutnya dilakukan analisa kinerja terbaik dari tiap metode. Perbedaan proses preprocessing antara Cosine dan VSM terletak pada proses preprocessing TF-IDF. Metode cosine terdapat proses Term Frequency untuk melakukan pencocokan terkait kemiripan judul skripsi. Sedangkan pada metode VSM pencocokan kata diilustrasikan dalam bentuk vektor. Dimana setiap sudut menggambarkan kecocokan data pada dataset judul skripsi. Kemudian kemiripan antara setiap judul akan dihitung dengan mengukur jarak kemiripan antara vektor yang muncul pada judul skripsi dilakukan data uji. Pada gambar 3.3 menunjukkan proses pencocokan *similarity* VSM berbasis vektor. Dimana Query adalah judul skripsi yang dimasukkan. Term 1, 2, 3 merupakan pemecahan kata dasar pada judul skripsi kemudian dicocokkan dengan dataset doc 1 dan 2 pada database skripsi Teknik Informatika Universitas Islam Balitar



Gambar 3.3 Ilustrasi Pengecekan *Similarity* Menggunakan VSM berbasis Vektor

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

Berikut tahapan dari hasil perancangan sistem deteksi kemiripan judul skripsi pada Jurusan Teknik Informatika Fakultas Teknologi Informasi Universitas Islam Balitar :

4.1. Pengumpulan Data

Proses pengumpulan data diambil dari data skripsi yang terdapat pada Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Islam Balitar sejumlah 200 judul skripsi sebagai data set yang dibagi 30 sebagai data uji dan 170 data training yang telah dikumpulkan dapat dilihat pada gambar 4.1 berikut ini.

"Sistem Prediksi Jumlah Permisian Darah Menggunakan Metode Double Exponential Smoothing",
"Penerapan Algoritma Naive Bayes Pada Klasifikasi Pekerjaan Alumni Perguruan Tinggi",
"Penerapan Metode Profile Matching Dalam Penentuan Promosi Jabatan Karyawan PT Sumber Alfaria Trijaya",
"Sistem Pendukung Keputusan Seleksi Calon Penerima Program Indonesia Pintar Menggunakan Metode Simple Additive Weighting",
"Sistem Pendukung Keputusan Sepeda Motor Untuk Dijual Kembali Dengan Metode AHP berbasis Web",
"Sistem Rekomendasi Produk Menggunakan Metode Association Rule Mining",
"Sistem Penjadwalan Ujian Skripsi Menggunakan Algoritma Genetika",
"Perancangan Dan Implementasi Virtual Server Menggunakan Proxmox Dengan Metode High Availability",
"Sistem penentuan Oli Motor Terbaik Menggunakan Pengujian White Box",
"Aplikasi penentuan MPASI Berdasarkan Jumlah Rakor Pada Bayi dengan Metode TOPSIS",
"Sistem Pendukung Keputusan Pemilihan Kayu Untuk Mebel Menggunakan Metode AHP",
"Klasifikasi Tumbuhan Khat Umami Rasm Indonesia Dan Madinah Pada Algoritma Menggunakan Back Propagation",
"Sistem Pendukung Keputusan Pemilihan Produk Asuransi Jiwa tambahan Menggunakan Metode Profile Matching",
"Sistem Informasi Pencarian Administrasi Surat Masuk dan Keluar dengan algoritma String Matching Knuth Morph".

Gambar 4.1. Data training judul skripsi

4.2. Tahap Pre-Processing

Pada tahap ini merupakan awal proses data teks akan diolah menjadi data berbasis numerik. Menurut Wahyuni (2017), menjelaskan bahwa Proses Preprocessing merupakan tahapan dimana deskripsi akan diolah dan siap diproses pada tahap teks mining. Setelah data terkumpul proses selanjutnya terdiri :

a. Data Integrasi

Pada penelitian ini data judul skripsi akan dijadikan 1 dalam bentuk file json. Untuk melihat contoh data integritas sesuai *sample* dataset yang digunakan dapat dilihat pada tabel 4.1 dibawah ini.

Tabel 4.1 Proses Integrasi

No	judul skripsi
1	Sistem Prediksi Jumlah Permintaan Darah Menggunakan Metode Double Exponential Smoothing
2	Penerapan Algoritma Naive Bayes Pada Klasifikasi Pekerjaan Alumni Perguruan Tinggi
3	Penerapan Metode Profile Matching Dalam Penentuan Promosi Jabatan Karyawan PT Sumber Alfaria Trijaya
4	Sistem Pendukung Keputusan Seleksi Calon Penerima Program Indonesia Pintar Menggunakan Metode Simple Additive Weighting
5	Sistem Pendukung Keputusan Sepeda Motor Untuk Dijual Kembali Dengan Metode AHP berbasis Web
6	Sistem Rekomendasi Produk Menggunakan Metode Association Rule Mining
7	Sistem Penjadwalan Ujian Skripsi Menggunakan Algoritma Genetika
8	Perancangan Dan Implementasi Virtual Server Menggunakan Proxmox Dengan Metode High Availability
9	Sistem penentuan Oli Motor Terbaik Menggunakan Pengujian White Box
10	Aplikasi penentuan MPASI Berdasarkan Jumlah Kalori Pada Bayi dengan Metode TOPSIS

Pada tabel 4.1 diatas data yang ada berupa judul skripsi dari hasil pengumpulan data kemudian dimasukan kedalam tabel dan dijadikan satu file data berbentuk json yang siap dilakukan preprocessing ke tahap selanjutnya.

b. Case Folding

Pada proses *case folding* judul skripsi akan dilakukan konversi semua judul menjadi *lowercase* agar memudahkan proses deteksi kemiripan judul skripsi menggunakan *Cosine Similarity* dan *VSM*.

Tabel 4.2 Proses *Case Folding*

No	judul skripsi	<i>Case Folding</i>
1	Sistem Prediksi Jumlah Permintaan Darah Menggunakan Metode Double Exponential Smoothing	sistem prediksi jumlah permintaan darah menggunakan metode double exponential smoothing.
2	Penerapan Algoritma Naive Bayes Pada Klasifikasi Pekerjaan Alumni Perguruan Tinggi	penerapan algoritma naive bayes pada klasifikasi pekerjaan alumni perguruan tinggi
3	Penerapan Metode Profile Matching Dalam Penentuan Promosi Jabatan Karyawan PT Sumber Alfaria Trijaya	penerapan metode profile matching dalam penentuan promosi jabatan karyawan pt sumber alfaria trijaya
4	Sistem Pendukung Keputusan Seleksi Calon Penerima Program Indonesia Pintar Menggunakan Metode Simple Additive Weighting	sistem pendukung keputusan seleksi calon penerima program indonesia pintar menggunakan metode simple additive weighting
5	Sistem Pendukung Keputusan Sepeda Motor Untuk Dijual Kembali Dengan Metode AHP berbasis Web	sistem pendukung keputusan sepeda motor untuk dijual kembali dengan metode ahp berbasis web
6	Sistem Rekomendasi Produk Menggunakan Metode Association Rule Mining	sistem rekomendasi produk menggunakan metode association rule mining
7	Sistem Penjadwalan Ujian Skripsi Menggunakan Algoritma Genetika	sistem penjadwalan ujian skripsi menggunakan algoritma genetika
8	Perancangan Dan Implementasi Virtual Server Menggunakan Proxmox Dengan Metode High Availability	perancangan dan implementasi virtual server menggunakan proxmox dengan metode high availability
9	Sistem penentuan Oli Motor Terbaik Menggunakan Pengujian White Box	sistem penentuan oli motor terbaik menggunakan pengujian white box
10	Aplikasi penentuan MPASI Berdasarkan Jumlah Kalori Pada Bayi dengan Metode TOPSIS	aplikasi penentuan mpasi berdasarkan jumlah kalori pada bayi dengan metode topsis

Pada tabel 4.2 dapat dilihat bahwa proses *preprocessing* dengan menggunakan *case folding* dilakukan untuk merubah tulisan yang tadinya berupa campuran

antara huruf besar (*uppercase*) dan juga huruf kecil (*lowercase*) menjadi hanya tulisan berupa *lowercase* (huruf kecil).

c. *Tokenizing*

Pada Proses *Tokenizing*, data yang sebelumnya dilakukan konferensi pada case folding selanjutnya akan dilakukan pengecekan terkait tanda baca, karakter khusus akan dihapus.

Tabel 4.3 proses *tokenizing*

No	Case Folding	Tokenizing
1	sistem prediksi jumlah permintaan darah menggunakan metode double exponential smoothing.	sistem prediksi jumlah permintaan darah menggunakan metode double exponential smoothing.
2	penerapan algoritma naive bayes pada klasifikasi pekerjaan alumni perguruan tinggi	penerapan algoritma naive bayes pada klasifikasi pekerjaan alumni perguruan tinggi
3	penerapan metode profile matching dalam penentuan promosi jabatan karyawan pt sumber alfaria trijaya	penerapan metode profile matching dalam penentuan promosi jabatan karyawan pt sumber alfaria trijaya
4	sistem pendukung keputusan seleksi calon penerima program indonesia pintar menggunakan metode simple additive weighting	sistem pendukung keputusan seleksi calon penerima program indonesia pintar menggunakan metode simple additive weighting
5	sistem pendukung keputusan sepeda motor untuk dijual kembali dengan metode ahp berbasis web	sistem pendukung keputusan sepeda motor untuk dijual kembali dengan metode ahp berbasis web
6	sistem rekomendasi produk menggunakan metode association rule mining	sistem rekomendasi produk menggunakan metode association rule mining
7	sistem penjadwalan ujian skripsi menggunakan algoritma genetika	sistem penjadwalan ujian skripsi menggunakan algoritma genetika
8	perancangan dan implementasi virtual server menggunakan proxmox dengan metode high availability	perancangan dan implementasi virtual server menggunakan proxmox dengan metode high availability
9	sistem penentuan oli motor terbaik menggunakan pengujian white box	sistem penentuan oli motor terbaik menggunakan pengujian white box
10	aplikasi penentuan mpasi berdasarkan jumlah kalori pada bayi dengan metode topsis	aplikasi penentuan mpasi berdasarkan jumlah kalori pada bayi dengan metode topsis

Pada tabel 4.3 dapat dilihat bahwa proses *tokenizing* yang dilakukan untuk menghilangkan karakter-karakter yang ada pada judul skripsi.

d. *Stopword / Filtering*

Proses stopword digunakan untuk menghapus pengulangan kata yang mungkin terjadi pada saat penginputan dataset judul.

Tabel 4.4 proses *stopword/filtering*

No	<i>Tokenizing</i>	<i>stopword/filtering</i>
1	sistem prediksi jumlah permintaan darah menggunakan metode double exponential smoothing.	sistem prediksi jumlah permintaan darah metode double exponential smoothing
2	penerapan algoritma naive bayes pada klasifikasi pekerjaan alumni perguruan tinggi	penerapan algoritma naive bayes klasifikasi pekerjaan alumni perguruan tinggi
3	penerapan metode profile matching dalam penentuan promosi jabatan karyawan pt sumber alfaria trijaya	penerapan metode profile matching penentuan promosi jabatan karyawan pt sumber alfaria trijaya
4	sistem pendukung keputusan seleksi calon penerima program indonesia pintar menggunakan metode simple additive weighting	sistem pendukung keputusan seleksi calon-penerima program indonesia pintar metode simple additive weighting
5	sistem pendukung keputusan sepeda motor untuk dijual kembali dengan metode ahp berbasis web	sistem pendukung keputusan sepeda motor dijual metode ahp berbasis web
6	sistem rekomendasi produk menggunakan metode association rule mining	sistem rekomendasi produk metode association rule mining
7	sistem penjadwalan ujian skripsi menggunakan algoritma genetika	sistem penjadwalan ujian skripsi algoritma genetika
8	perancangan dan implementasi virtual server menggunakan proxmox dengan metode high availability	perancangan implementasi virtual server proxmox metode high availability
9	sistem penentuan oli motor terbaik menggunakan pengujian white box	sistem penentuan oli motor pengujian white box
10	aplikasi penentuan mpasi berdasarkan jumlah kalori pada bayi dengan metode topsis	aplikasi penentuan mpasi jumlah kalori bayi metode topsis

Pada tabel 4.4 proses *stopword* telah dilakukan dalam rangka menghapus pengulangan kata ataupun kata yang dianggap tidak penting yang ada pada judul skripsi.

e. *Stemming*

Proses stemming berguna untuk menemukan kata dasar dari setiap kata yang disaring. Data yang dihasilkan dari *stopword* akan digunakan untuk

melakukan proses penghitungan Term Frequency / Penghitungan banyaknya kemiripan kata dasar.

- 1) sistem => sistem
- 2) prediksi => prediksi
- 3) jumlah => jumlah
- 4) permintaan => minta
- 5) darah => darah
- 6) metode => metode
- 7) double => double
- 8) exponential => exponential
- 9) smoothing => smooth

Gambar 4.2 Proses *Stemming*

Pada gambar 4.2 diatas proses stemming dilakukan dengan menghilangkan awalan dan akhiran pada tiap kata menjadi kata dasar.

4.3. Proses Penghitungan *Cosine Similarity*

Data testing judul skripsi mahasiswa dimasukkan kedalam Aplikasi yang selanjutnya diproses (di-running) menghasilkan angka prosentase kemiripan dan kecepatan proses yang berbeda beda. Selanjutnya hasil uji beberapa judul skripsi tersebut dimasukkan ke dalam tabel untuk dianalisis untuk menentukan skenario mana setelah data testing di treatment menghasilkan metoda uji kemiripan judul skripsi yang terbaik (prosentase kemiripannya) dan tingkat sensitifitasnya dalam mendeteksi kemiripan judul skripsi yang diuji.

Adapun script untuk melakukan pengujian kemiripan judul skripsi dengan metoda algoritma *Cosine Similarity* sesuai dalam kode bawah ini.


```

def calculate_cosine_similarity(doc1, doc2):
    # Define weights for each term
    weights = {'title': 0.9}
    # Convert documents to vectors using TF-IDF
    vectorizer=
    TfidfVectorizer(stop_words='english')
    vectors = vectorizer.fit_transform([doc1,
    doc2]).toarray()
    # Apply weights to the vectors
    weights_array = np.array([weights['title'],
    weights['title']])
    weighted_vectors = vectors *
    np.expand_dims(weights_array, axis=1)
    # Calculate cosine similarity
    similarity =
    cosine_similarity(weighted_vectors[0].reshape
    (1, -1), weighted_vectors[1].reshape(1, -1))
    return similarity[0][0]

```

Kode diatas adalah baris *code* untuk melakukan perhitungan kemiripan judul skripsi dengan metode *Cosine Similarity*. Setelah mahasiswa memasukan judul yang akan diajukan selanjutnya sistem akan membandingkan dan melakukan pencocokan dengan dataset yang ada. Adapun pendefinisian pada *doc1* pada kode diatas menunjukan sebagai input judul oleh mahasiswa dan *doc2* sebagai definisi untuk dataset yang tersedia pada sistem. Berikut proses penghitungan dari *Cosine Similarity* dalam melakukan proses pendeteksian kemiripan judul skripsi.

Tabel 4.6 Perbandingan Judul

No	Judul 1	Judul 2
1	Perancangan Sistem Informasi Penjualan Online Dengan Metode Ecommerce	Sistem Informasi Penjualan Online Dengan Metode E-commerce

Tabel 4.6 diatas merupakan tabel perbandingan antara judul yang akan diujikan dengan judul yang ada pada database sistem.

Tabel 4.7. Hasil *Cosine Similarity*

Term	D1
Perancangan	0
Sistem	1
Informasi	1
Penjualan	1
Online	1
Dengan	1
Metode	1
Ecommerce	1
TF	7
IDF	0,0579919
TF*IDF	0,4059436
Vector Q > D (Test Data > Training Dataset)	2,0102119
Vector Q	2,6457513
<i>Similarity</i>	0,537251722

Dari tabel 4.7 diatas dapat kita lihat nilai dari tiap kata dari judul skripsi dengan nilai yang ada pada tiap baris, untuk nilai *true positif* adalah 7 dan nilai dari IDF adalah 0,0579919. Selanjutnya hasil perkalian dari TF x IDF menghasilkan 0,4059436, dan nilai dari *Similarity* adalah 0,537251722 atau sebesar 53%. Selanjutnya dapat dilihat hasil pengujian pengajuan judul skripsi menggunakan algoritma VSM.

4.4. Pembobotan *Cosine Similarity*

Dalam penelitian yang dilakukan pembobotan dilakukan menggunakan nilai pembobotan 0.9, 0.8 dan 0.7 untuk mengetahui seberapa besar tingkat kemiripan antar judul yang diuji pada penelitian. Adapun untuk penghitungan bobot dalam pengujian *Cosine Similarity* adalah sebagai berikut.

```
def calculate_cosine_similarity(doc1, doc2):
    # Define weights for each term
    weights = {'title': 0.9}
```

kode diatas menjelaskan rumus penghitungan bobot dengan menggunakan *variabel weights* dengan melihat judul skripsi yang akan diuji yang dideskripsikan dengan *title* serta nilai bobot yang digunakan yaitu 0.9, 0.8 dan 0.7 untuk menguji nilai lainnya tinggal mengganti nilai 0.9 dengan nilai bobot yang akan kita uji.

Jika nilai bobot yang digunakan semakin tinggi maka tingkat kontribusi dan kepentingan pada judul yang diuji sangat tinggi, dan semakin kecil bobot yang digunakan maka dianggap bahwa judul yang diuji menunjukkan tingkat kepentingan lebih rendah dalam menentukan kesamaan relevansi dokumen.

Tabel 4.8 Perbandingan Judul

No	Judul 1	Judul 2
1	Perancangan Sistem Informasi Penjualan Online Dengan Metode Ecommerce	Sistem Informasi Penjualan Online Dengan Metode E-commerce

Tabel 4.8 diatas merupakan tabel perbandingan antara judul yang akan diujikan dengan judul yang ada pada database dan kemudian akan dilakukan pengujian menggunakan metode VSM.

Tabel 4.9. Tes Kemiripan Judul Skripsi dengan VSM

Term	D1
Perancangan	0
Sistem	0
Informasi	0
Penjualan	1
Online	1
Dengan	0
Metode	0
Ecommerce	0
TF	2
IDF	0,1760913
TF*IDF	0,3521825
Vector Q -> D	2,0310081
Vector Q	1,4142136
Similarity	0,089758726 atau 8%

Dari tabel 4.9 diatas dapat dilihat hasil perhitungan untuk tes kemiripan menggunakan judul skripsi dengan hasil *similarity* sebesar 8%.

4.5. Pengujian Sistem

Dalam pembuatan melakukan pengujian terhadap sistem deteksi *plagiarisme* sistem yang dibangun menggunakan *microframework flask*. Sistem yang dibuat menggunakan dua metode yang berbeda, yang pertama menggunakan algoritma VSM dan yang kedua menggunakan algoritma *Cosine Similarity*.

4.5.1. Pengujian Algoritma VSM

Metode yang pertama yang dibangun dalam melakukan cek kemiripan judul adalah algoritma VSM. Langkah pertama dalam melakukan pengujian kemiripan judul adalah dengan melakukan *import* terhadap *library* yang dibutuhkan, adapun perintah yang digunakan dapat dilihat pada kode dibawah ini.

```
from flask import Flask, render_template,
request
import json
import numpy as np
from sklearn.feature_extraction.text import
TfidfVectorizer
from sklearn.metrics.pairwise import
cosine_similarity
import time
from datetime import datetime
```

kode diatas adalah proses melakukan import terhadap library yang dibutuhkan oleh program dalam melakukan perbandingan tingkat kemiripan judul skripsi. Selanjutnya adalah proses import database berupa file json yang dapat dilihat pada kode dibawah ini.

```

@staticmethod
def load_json_data(file_path):
    with open(file_path, 'r') as file:
        data = json.load(file)
    return data

```

Selanjutnya setelah data diload kedalam sistem proses selanjutnya adalah melakukan proses penghitungan VSM dengan menggunakan *Cosine Similarity*. Untuk melihat lebih jelasnya bisa dilihat pada kode dibawah ini.

```

class VectorSpaceModel:
    def __init__(self, data_path):
        self.json_data =
self.load_json_data(data_path)
        self.titles = self.json_data['titles']
        self.vectorizer =
TfidfVectorizer(stop_words='english')
        self.vectors =
self.vectorizer.fit_transform(self.titles).toa
rray()
        @staticmethod
        def load_json_data(file_path):
            with open(file_path, 'r') as file:
                data = json.load(file)
            return data
        def calculate_cosine_similarity(self,
query):
            query_vector =
self.vectorizer.transform([query]).toarray()
            similarity_scores =
cosine_similarity(query_vector,
self.vectors)[0]
            return similarity_scores

```

Selanjutnya data yang sudah diload akan dilakukan pengujian perbandingan antara database json yang ada dengan judul yang akan diuji. Proses tersebut dapat dilihat pada kode dibawah ini.

```

@app.route('/search', methods=['POST'])
def search():
    query = request.form['query']
    start_time = time.time()
    vsm = VectorSpaceModel('data.json')
    similarity_scores =
vsm.calculate_cosine_similarity(query)
    end_time = time.time()
    processing_time = end_time - start_time
    results_df = pd.DataFrame({'title':
vsm.titles, 'similarity': similarity_scores})
    top_similarity_scores = results_df.nlargest(5,
'similarity')

```

Proses terakhir dari pengujian adalah menampilkan proses pengujian, untuk proses output dapat dilihat pada kode dibawah ini.

```

return render_template('result.html',
query=query,
similarity_scores=top_similarity_scores.values,
processing_time=processing_time)

```

Pada kode yang telah dibuat digunakan model VSM yang digabungkan dengan metode *Cosine Similarity*. Hubungan antara VSM dan *Cosine Similarity* sangat erat hubungannya karena VSM digunakan untuk menghasilkan representasi vektor dari dokumen atau query dalam ruang vektor sedangkan *Cosine Similarity* digunakan untuk mengukur sejauh mana dua vektor tersebut berada dalam hubungan kosinus, mencerminkan tingkat kesamaan atau relevansinya.

Setelah kode pengujian selesai dibuat selanjutnya *interface prototype* yang telah dibuat bisa langsung diuji coba. Berikut ini adalah penerapan *prototype* sistem yang telah dibuat menggunakan algoritma VSM untuk mengetahui sejauh mana tingkat kemiripan judul skripsi yang diajukan oleh mahasiswa dengan data yang ada pada database.



Gambar 4.3. Tampilan awal sistem deteksi plagiarisme Algoritma VSM

Ketika pengguna mulai membuka tampilan awal sistem deteksi plagiarisme dengan algoritma VSM maka tampilan awal akan terlihat seperti pada gambar 4.3 diatas. Pada gambar 4.3 tampilan yang disajikan sangat sederhana, sehingga pengguna akan lebih mudah dalam menggunakan sistem yang telah dirancang. Pada tampilan awal untuk menggunakan sistem ini pengguna tinggal memasukan judul skripsi pada kolom judul yang tersedia kemudian tentukan bobot yang akan digunakan dengan ketentuan nilai bobot yang dimasukan adalah rentang antara 0 sampai dengan 1 dan nilai default dari bobot pada sistem yang telah dibuat adalah 0.9. Jika bobot yang dimasukan bernilai kecil maka hasil dari sistem yang dirancang tidak terlalu sensitif dalam membaca judul yang dimasukan, akan tetapi jika bobot yang dimasukan semakin besar maka tingkat sensitif akan semakin tinggi. Adapun hasil dari pengecekan *plagiarisme* menggunakan metode VSM dengan judul skripsi “Perancangan Sistem Informasi Penjualan Online Dengan Metode Ecommerce” dapat dilihat pada gambar 4.4 dibawah ini.

Search Result

Query Perancangan Sistem Informasi Penjualan Online Dengan Metode Ecommerce

- Perancangan Sistem Informasi Penjualan Online dengan Metode E-commerce - (13 30479)
- Perancangan dan Implementasi Sistem Informasi Penjualan Online dengan Metode E-commerce - (49 18 898)
- Perancangan dan Implementasi Sistem Informasi Penjualan Online berbasis web - (40 140470)
- Perancangan dan Implementasi Sistem Informasi Manajemen Logistik dengan Metode Customer Relationship Management (CRM) - (42 11 1096)
- Perancangan dan Implementasi Sistem Informasi Manajemen Penjualan dengan Metode Customer Relation Management (CRM) - (42 11 1071)

Proceding 11th IJCAI/11/AM/09 (2008)

Gambar 4.4 Hasil pengecekan plagiarisme metode VSM

Pada gambar 4.4 diatas hasil pengecekan judul skripsi ditampilkan menggunakan persentase kemiripan antara judul yang diajukan dengan judul lain yang sudah ada, disamping itu juga selain menampilkan berdasarkan persentase sistem juga menampilkan perangkingan 5 data teratas yang memiliki kemiripan judul. Hasil dari pengujian judul skripsi dengan judul "Perancangan Sistem Informasi Penjualan Online Dengan Metode Ecommerce" menghasilkan nilai persentase tertinggi sebesar 88.43% menggunakan algoritma VSM. Sedangkan nilai persentase terendah dari 5 data teratas menghasilkan nilai persentase kemiripan sebesar 41.63%.

4.5.2. Pengujian Algoritma *Cosine Similarity*

Metode selanjutnya yang dibangun dalam mengecek kemiripan judul adalah algoritma *Cosine Similarity*. Berikut ini adalah penerapan *prototype* sistem yang telah dibuat menggunakan algoritma *Cosine Similarity*.



Gambar 4.5. Tampilan awal sistem deteksi plagiarisme Algoritma *Cosine Similarity*

Ketika pengguna mulai membuka tampilan awal sistem deteksi plagiarisme dengan algoritma *Cosine Similarity* maka tampilan awal akan terlihat seperti pada gambar 4.5 diatas. Pada gambar 4.5 tampilan yang disajikan sangat sederhana, sehingga pengguna akan lebih mudah dalam menggunakan sistem yang telah dirancang.

Proses dalam melakukan input judul yang akan dilakukan pengujian dimulai dengan melakukan import database file json terlebih dahulu kedalam sistem. Proses *import data json* pada *Cosine Similarity* dapat dilihat pada kode dibawah ini.

```
def load_json_data(file_path):
    with open(file_path, 'r') as file:
        data = json.load(file)
    return data
```

Pada kode diatas merupakan tampilan kode untuk melakukan import data json yang sebelumnya telah dilakukan perintah import dengan perintah *import json* pada bagian deskripsi header. *Path* pada file yang akan diimport

dilakukan pembacaan json dengan perintah *r* sebagai file. Selanjutnya proses import data json ditunjukkan dengan perintah pada kode dibawah ini.

```
json_data = load_json_data('data.json')
titles = json_data['titles']
```

Pada kode proses load data json sebagai database untuk perbandingan *similarity* (kemiripan) terhadap judul skripsi di import kedalam sistem, kemudian data yang telah diimport selanjutnya dilakukan pemilihan kunci yang akan dilakukan perbandingan kemiripan, adapun pada gambar 4.8 diatas kunci yang dipilih adalah *titles*. Tahapan selanjutnya setelah database diimport adalah melakukan perbandingan dengan data judul yang diinput, pada sistem yang dibuat judul skripsi yang diinput ditangkap menggunakan variabel *query* dengan metode *post*, dapat dilihat pada kode dibawah ini.

```
@app.route('/search', methods=['POST'])
def search():
    query = request.form['query']

    json_data = load_json_data('data.json')
    titles = json_data['titles']
    similarity_scores = []
    start_time = time.time()
    for title in titles:
        similarity_score =
        calculate_cosine_similarity(query, title)
        similarity_scores.append((title,
        similarity_score))
    end_time = time.time()
    processing_time = end_time - start_time
    similarity_scores.sort(key=lambda x: x[1],
    reverse=True)
    top_similarity_scores = similarity_scores[:5]
```

Pada kode diatas proses pembacaan input judul skripsi ditandai dengan *form* dengan nama inputan *query* dan metode yang digunakan adalah metode *post*. Selanjutnya variabel *query* yang diinput akan dilakukan perbandingan

dengan variabel *titles* dan kemudian hasil perhitungan akan diambil 5 teratas dengan yang ada pada database. Hasil yang telah dilakukan perbandingan selanjutnya akan ditampilkan ke halaman *result*, untuk kodenya dapat dilihat seperti pada kode dibawah ini.

```
return render_template('result.html',
    query=query,
    similarity_scores=top_similarity_scores,
    processing_time=processing_time
```

Pada kode hasil output akan dikirimkan ke file *result.html* dengan menampilkan judul yang diinput, skor kemiripan 5 judul teratas dan waktu pemrosesan yang dibutuhkan.

Pada tampilan awal untuk menggunakan sistem ini pengguna tinggal memasukan judul skripsi pada kolom judul yang tersedia kemudian tentukan bobot yang akan digunakan dengan ketentuan nilai bobot yang dimasukan adalah rentang antara 0 sampai dengan 1 dan nilai default dari bobot pada sistem yang telah dibuat adalah 0.9. jika bobot yang dimasukan kecil maka tidak terlalu sensitif, akan tetapi jika bobot yang dimasukan semakin besar maka tingkat sensitif sangat tinggi. Adapun hasil dari pengecekan *plagiarisme* menggunakan metode *Cosine Similarity* dapat dilihat pada gambar 4.6 dibawah ini.



Search Result

Query: Penerapan Sistem Informasi Berbasis Online Dengan Metode Cosine Similarity

- Penerapan Sistem Informasi Berbasis Online Dengan Metode Cosine Similarity - 171 (17.81%)
- Penerapan dan Implementasi Sistem Informasi Berbasis Online Dengan Metode Cosine Similarity - 049 (50.02%)
- Analisis dan Penerapan Sistem Informasi Berbasis Online Dengan Metode Cosine Similarity - 044 (40.77%)
- Analisis dan Penerapan Sistem Informasi Berbasis Online Dengan Metode Cosine Similarity - 049 (50.02%)
- Penerapan dan Implementasi Sistem Informasi Berbasis Online Dengan Metode Cosine Similarity - 045 (42.77%)

Processing Time: 0.01907675166414088

Gambar 4.6 Hasil pengecekan plagiarisme metode *Cosine Similarity*

Pada gambar 4.6 diatas hasil pengecekan judul skripsi “Perancangan Sistem Informasi Penjualan Online Dengan Metode Ecommerce” dengan skenario pertama menggunakan bobot 0.9 ditampilkan menggunakan persentase kemiripan antara judul yang diajukan dengan judul lain yang sudah ada, disamping itu juga selain menampilkan berdasarkan persentase sistem juga menampilkan perangkangan 5 data teratas yang memiliki kemiripan judul. Hasil persentase tertinggi menggunakan algoritma *Cosine Similarity* menghasilkan nilai sebesar 77.99% sedangkan nilai terendah dari hasil 5 data yang ditampilkan adalah sebesar 42.72%.

Hasil dari pengujian prototype menggunakan algoritma VSM dan algoritma *Cosine Similarity* menunjukkan bahwa hasil dari pengecekan judul skripsi menggunakan algoritma *Cosine Similarity* menghasilkan nilai persentase lebih kecil dibanding dengan algoritma VSM. Hal ini berarti algoritma VSM tampil lebih baik dibandingkan dengan algoritma *Cosine Similarity*.

4.5.3. Pengujian Judul Skripsi Menggunakan Layanan Turnitin

Selanjutnya untuk melihat perbandingan model yang dibuat dengan pengujian menggunakan layanan turnitin dilakukan pengujian judul skripsi yang sama untuk melihat seberapa besar nilai *similarity* dari layanan tersebut dibanding dengan model yang dibuat menggunakan algoritma *Cosine Similarity* dan juga VSM. Adapun hasil dari pengujian menggunakan layanan *turnitin* dapat dilihat pada gambar 4.7 dibawah ini.

Submission date: 04-Oct-2023 06:35AM (UTC-0700)
 Submission ID: 2184176679
 File name: 2.pdf (187.39K)
 Word count: 24
 Character count: 187



Gambar 4.7 Pengujian judul skripsi menggunakan layanan *turnitin*

Dari gambar 4.7 diatas adalah hasil upload judul skripsi yang diusulkan menggunakan 3 judul yang sama dengan pengetikan karakter yang berbeda, hasil dari pengujian judul skripsi tersebut dapat dilihat pada gambar 4.8 dibawah ini.



Gambar 4.8 Laporan plagiasi judul yang diusulkan menggunakan *turnitin*

Dari gambar 4.8 diatas kita bisa melihat hasil yang didapat dari layanan *turnitin* sebesar 0% hal ini berarti judul yang diusulkan belum terdeteksi pada database *turnitin*. Selanjutnya dilakukan pengujian 30

sample judul skripsi menggunakan layanan turnitin untuk mengetahui nilai *similarity* dari judul tersebut. Hasil pengujian dari judul menggunakan layanan turnitin dapat dilihat pada gambar 4.9 dibawah ini.

1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10
11	11	11	11
12	12	12	12
13	13	13	13
14	14	14	14
15	15	15	15
16	16	16	16
17	17	17	17
18	18	18	18
19	19	19	19
20	20	20	20
21	21	21	21
22	22	22	22
23	23	23	23
24	24	24	24
25	25	25	25
26	26	26	26
27	27	27	27
28	28	28	28
29	29	29	29
30	30	30	30

Gambar 4.9 Pengujian 30 judul skripsi menggunakan *turnitin*

Pada gambar 4.9 diatas dapat dilihat 30 judul skripsi telah diuji tingkat *similarity* (kemiripan) menggunakan layanan turnitin, hasil yang didapat dari 30 judul tersebut hampir semua menghasilkan *similarity* sebesar 100% hanya pada judul no 20 yang menghasilkan 22% dan judul nomor 23 menghasilkan 0%.

4.5.4. Perbandingan Kemiripan Algoritma terhadap layanan *turnitin*

Tabel 4.10 Tes perbandingan *similarity*

NO	JUDUL SKRIPSI	NILAI UJI DENGAN COSINE			NILAI UJI DENGAN VSM			NILAI UJI DENGAN TURNITIN			KESIMPULAN YG DEKAT TURNITIN
		% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	
1	Pengembangan Aplikasi Penentuan Media Promosi Pelaku UMKM Berbasis Website	20,02	1,88	20,02	29,90	0,03	30,00	100,00	1,00	100,00	VSM
2	User Experience Evaluation of Botani Mobile Application using User Experience Questionnaire	4,72	1,70	4,72	31,58	0,04	31,58	100,00	1,00	100,00	VSM
3	Prediksi Pergerakan Saham BBRI ditengah Issue Ancaman Resesi 2023 dengan Pendekatan Machine Learning	15,33	2,03	15,33	71,78	0,03	71,78	100,00	1,00	100,00	VSM
4	Enhancement of Coronary Heart Disease Prediction using Stacked Long Short Term Memory	0	2,19	0	0	0,04	0	100,00	1,00	100,00	Tidak ada
5	Klasifikasi Cacat Biji Kopi Menggunakan Metode Transfer Learning dengan Hyperparameter Tuning Gridsearch	15,18	1,67	15,18	40,05	0,04	40,05	100,00	1,00	100,00	VSM

Tabel 4.10 Lanjutan

NO	JUDUL SKRIPSI	NILAI UJI DENGAN COSINE			NILAI UJI DENGAN VSM			NILAI UJI DENGAN TURNITIN			KESIMPULAN YG DEKAT TURNITIN
		% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	
6	Penerapan Metode Simple Additive Weighting (SAW) untuk Penentuan Muazin Terbaik	20,61	1,68	20,61	40,42	0,05	40,42	100,00	1,00	100,00	VSM
7	Sistem Informasi Manajemen Aset Bendung Serdang BWS Sumatera II Menggunakan Metode Straight Line dan Simple Additive Weighting	24,40	1,67	24,40	54,32	0,04	54,32	100,00	1,00	100,00	VSM
8	Penerapan Model Kesuksesan Sistem Informasi HOT-Fit terhadap Penggunaan e-Belajar sebagai Media Pembelajaran Daring Berbasis Website (Studi Kasus: STIKI)	14,21	1,71	14,21	32,02	0,03	32,02	100,00	1,00	100,00	VSM
9	Implementasi NodeMCU sebagai Sistem Deteksi Dini Bahaya Banjir di Purwodadi Berbasis Web dengan Notifikasi Bot Telegram	22,03	1,75	22,03	32,83	0,03	32,83	100,00	1,00	100,00	VSM
10	Penerapan TOPSIS sebagai Penentu Kenaikan Gaji Karyawan di Perusahaan Alpha	9,73	1,83	9,73	27,51	0,03	27,51	100,00	1,00	100,00	VSM

Tabel 4.10 Lanjutan

NO	JUDUL SKRIPSI	NILAI UJI DENGAN COSINE			NILAI UJI DENGAN VSM			NILAI UJI DENGAN TURNITIN			KESIMPULAN YG DEKAT TURNITIN
		% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	
11	Analisis Sistem Pembelajaran Daring Berbasis Gamification Collaboration untuk Mendukung Merdeka Belajar Menggunakan Octalysis Framework	15,65	1,67	15,65	47,93	0,03	47,93	100,00	1,00	100,00	VSM
12	Sistem Informasi Bimbingan Konseling Menggunakan Framework PHP Codeigniter 4.0 (Studi pada SMAK Frateran Malang)	15,33	1,70	15,33	26,74	0,04	26,74	100,00	1,00	100,00	VSM
13	Analisis Metode Euclidean Distance dalam Menentukan Koordinat Peta pada Alamat Rumah	14,55	1,70	14,55	34,30	0,03	34,30	100,00	1,00	100,00	VSM
14	Sistem Informasi Point of Sale Berbasis Web pada Toko Afridah Cake	29,12	1,65	29,12	54,02	0,04	54,02	100,00	1,00	100,00	VSM
15	Pengaruh Kualitas Sistem Informasi Sumber Daya Manusia, Pelatihan dan Pengembangan terhadap Kinerja Karyawan melalui Disiplin Kerja pada PT. Era Mulia Abadi Sejahtera	13,46	1,73	13,46	42,55	0,04	42,55	100,00	1,00	100,00	VSM

Tabel 4.10 Lanjutan

NO	JUDUL SKRIPSI	NILAI UJI DENGAN COSINE			NILAI UJI DENGAN VSM			NILAI UJI DENGAN TURNITIN			KESIMPULAN YG DEKAT TURNITIN
		% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	
16	Studi Literatur Analisis Potensi Pasar Marketplace terhadap Penjualan	12,74	1,68	12,74	40,03	0,03	40,03	100,00	1,00	100,00	VSM
17	Perancangan Aplikasi Sistem Informasi Distribusi Bantuan Bencana Alam dengan Memanfaatkan Metode Rational Unified Process	31,69	1,96	31,69	50,88	0,04	50,88	100,00	1,00	100,00	VSM
18	Rekomendasi Supplement Learning Resources dalam E-Learning berbasis Fuzzy AHP	19,64	1,7	19,64	52,14	0,04	52,14	100,00	1,00	100,00	VSM
19	Implementasi Algoritma K-Means untuk Menentukan Persediaan Barang pada Poultry Shop	0,82	1,67	0,82	38,04	0,03	38,04	100,00	1,00	100,00	VSM
20	Analisis dan Perancangan Sistem Manajemen Inventaris Menggunakan Metode Fishbone	43,6	1,78	43,60	58,19	0,03	58,19	22,00	1,00	22,00	COSINE
21	Perbandingan Model Deep Learning untuk Klasifikasi Sentiment Analysis dengan Teknik Natural Language Processing	13,18	1,8	13,18	34,49	0,03	34,49	100,00	1,00	100,00	VSM

Tabel 4.10 Lanjutan

NO	JUDUL SKRIPSI	NILAI UJI DENGAN COSINE			NILAI UJI DENGAN VSM			NILAI UJI DENGAN TURNITIN			KESIMPULAN YG DEKAT TURNITIN
		% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	
22	Peramalan Stock Barang Dagangan Menggunakan Metode Single Exponential Smoothing	18,98	1,65	18,98	34,35	0,04	34,35	100,00	1,00	100,00	VSM
23	Pengembangan Aplikasi Android MVTE dengan Metode RAD	34,08	1,86	34,08	49,72	0,04	49,72	0	1,00	0	COSINE
24	Klasifikasi Tanaman Beringin (Ficus Bernjamina) berdasarkan Citra Daun Menggunakan Algoritma K-Nearest Neighbors	12,44	1,68	12,44	29,93	0,04	29,93	100,00	1,00	100,00	VSM
25	Rancang Bangun Sistem Informasi Point of Sales (POS) Pada CV. Arema Alam Abadi	20,2	1,67	20,20	36,71	0,03	36,71	100,00	1,00	100,00	VSM
26	Penerapan Algoritma K-Means untuk Segmentasi Konsumen Menggunakan R	24,12	1,66	24,12	28,04	0,03	28,04	100,00	1,00	100,00	VSM
27	Analisis Sentimen Terhadap Produk The Body Shop Tea Tree Oil	10,78	1,65	10,78	34,25	0,03	34,25	100,00	1,00	100,00	VSM
28	Pengenalan Ikan Hias Laut Pada Anak Usia 3 Tahun Dengan Metode Marker Based Tracking Berbasis Augmented Reality	13,66	1,76	13,66	61,58	0,04	61,58	100,00	1,00	100,00	VSM

Tabel 4.10 Lanjutan

NO	JUDUL SKRIPSI	NILAI UJI DENGAN COSINE			NILAI UJI DENGAN VSM			NILAI UJI DENGAN TURNITIN			KESIMPULAN YG DEKAT TURNITIN
		% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	% KEMIRIPAN	WAKTU PROSES	AKURASI	
29	SALES AND PURCHASE OF WEBSITE BASED HEAVY EQUIPMENT SPAREPART	5,61	1,76	5,61	43,19	0,04	43,19	100,00	1,00	100,00	VSM
30	SISTEM CERDAS DALAM KLASIFIKASI KEMATANGAN BUAH JERUK BERDASARKAN FITUR EKSTRAKSI GLCM DENGAN METODE NAÏVE BAYES	16,18	1,68	16,18	34,38	0,03	34,38	100,00	1,00	100,00	VSM

Dari tabel 4.10 Tes perbandingan *similarity*, hasil perbandingan uji kemiripan judul skripsi antara *Algoritma Cosine Similarity* dan VSM terhadap pengujian menggunakan layanan turnitin menghasilkan kenyataan bahwa penggunaan algoritma VSM unggul dengan 27 *sample* judul skripsi yang ada lebih mendekati terhadap hasil pengujian menggunakan layanan *turnitin*, sedangkan penggunaan *Algoritma Cosine Similarity* hanya menghasilkan 2 judul yang paling mendekati terhadap pengujian layanan *turnitin*, dan satu sisanya tidak masuk kepada algoritma manapun karena sama-sama menghasilkan nilai 0 pada pengujian judul skripsi. Untuk hasil pengujian VSM, *Cosine Similarity* dan *turnitin* terlampir.

BAB V

PENUTUP

5.1. Kesimpulan

Hasil analisis dari penelitian yang telah dilakukan didapatkan beberapa kesimpulan, adapun kesimpulan yang didapat adalah

1. Hasil pengujian judul skripsi menggunakan *Algoritma Cosine Similarity* menggunakan bobot 0.9 menghasilkan nilai kemiripan judul sebesar 77.94%. hal ini menunjukkan bahwa model yang dibuat memberikan hasil yang sangat baik dalam melakukan pengujian tingkat kemiripan judul skripsi.
2. Hasil pengujian judul skripsi menggunakan *Algoritma Vector Space Model* menggunakan bobot 0.9 menghasilkan nilai kemiripan sebesar 88.43%. artinya model yang dibuat menggunakan algoritma VSM bekerja lebih baik dibanding dengan *Algoritma Cosine Similarity*. Tingkat keakuratan model ini dalam mendeteksi kemiripan judul sangat baik karena menghasilkan nilai akurasi yang tinggi mencapai lebih dari 80%.
3. Tingkat kinerja *Algoritma Cosine Similarity* dan *Algoritma VSM* jika dilakukan komparasi dengan layanan *turnitin* tampil lebih baik karena hasil akurasi *Algoritma Cosine Similarity* dan algoritma *VSM* menghasilkan nilai lebih dari 70% sedangkan hasil pengujian menggunakan layanan *turnitin* menghasilkan nilai sebesar 0% yang berarti judul yang disarankan tidak terdeteksi pada database tersebut.

5.2. Saran

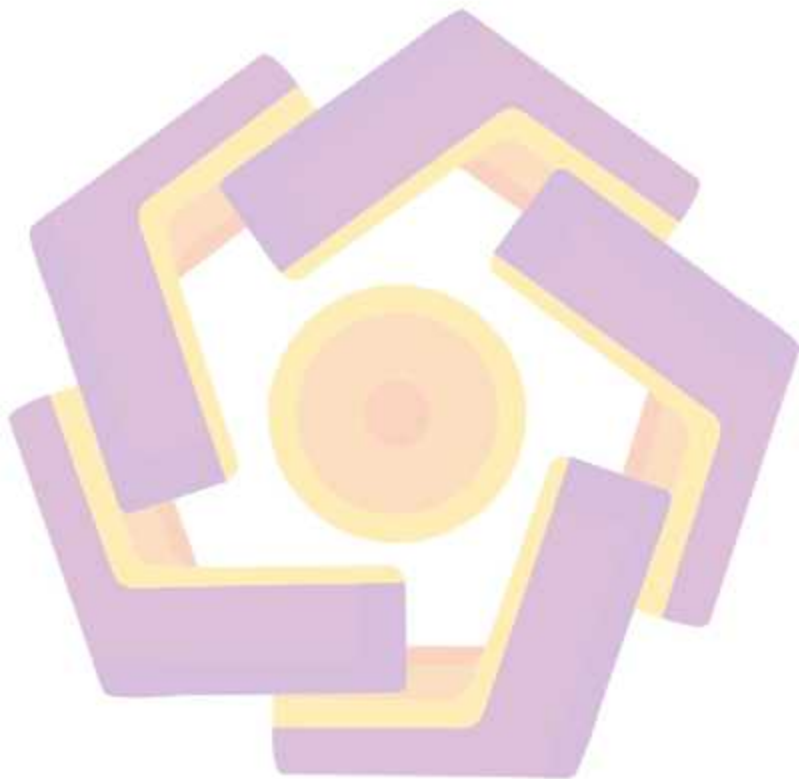
Dalam melakukan penelitian ini penulis menyadari masih banyak kekurangan yang perlu ditingkatkan sehingga kedepannya penelitian yang serupa bisa menghasilkan tingkat *akurasi* dan juga *reliabilitas* yang tinggi. Adapun beberapa saran dari penelitian yang sudah dilakukan adalah :

1. Untuk mendapatkan akurasi yang lebih tinggi diharapkan untuk penelitian berikutnya menggunakan dataset lebih banyak lagi dari dataset yang ada sekarang.
2. Untuk penelitian berikutnya diharapkan bukan hanya judul yang dilakukan pengujian *similarity* akan tetapi pada pengujian berikutnya bisa dilakukan pengujian terhadap dokumen skripsi dalam hal plagiasi
3. Disarankan ada penelitian lanjutan yang di uji adalah isi skripsinya dalam menguji kehandalan dua algoritma uji plagiarisme *Cosine Similarity* dan *VSM*.
4. Penelitian lanjutan disarankan menguji efektifitas pemberian threshold pada sebelum proses preprocessing algoritma kemiripan bisa menunjukkan perlu tidaknya ada threshold
5. Penelitian lanjutan juga diharapkan dapat menguji kecepatan rata-rata pemrosesan antara kedua algoritma
6. Penelitian lanjutan pada Uji Kemiripan Text (*Similarity Text Testing*) Judul Skripsi menggunakan Algoritma Vector Space Model (*VSM*) yang sudah teruji lebih baik dari *Cosine Similarity* tidak hanya memasukan Judul saja sebagai data uji dan data latihnya namun termasuk abstraksi. Judul skripsi tabpa adanya

abstraksi yang akan diujikan pada data latih akan memiliki kecenderungan kemiripan lebih besar karena kombinasi term judulnya lebih sedikit (karena kalimatnya pendek pendek).

7. Seiring bertambahnya jumlah lulusan dan waktu sejak berdirinya Fakultas Teknik Informatika Universitas Islam Balitar terdapat penambahan data latih akan meningkatkan kualitas pengujian dan memperkecil kemiripan Judul Skripsi yang diajukan mahasiswa apabila memasukkan abstraksi yang memuat kalimat atau text tentang jenis penelitian, metoda algoritma yang digunakan, atau kombinasi beberapa algoritma penelitian, lokasi objek penelitian, data penelitian dan waktu penelitian.
8. Direkomendasikan kepada Fakultas Teknologi Informasi Universitas Islam Balitar melakukan maksimal 1 tahun sekali update data latih skripsi dan abstraksi dimasukkan dalam aplikasi. Jika dimungkinkan setiap akhir semester setelah masa persidangan Skripsi mahasiswa dilakukan update pada data latih aplikasi uji kemiripan Judul Skripsi.
9. Selanjutnya penelitian berikutnya menyajikan hasil Uji Kemiripan Algoritma *VSM* dengan pendekatan *Cosine Similarity* dengan menempatkan nilai vektor kata per kata setiap Judul Skripsi dan Abstraksi yang diuji diperbandingan dengan titik-titik vektor Judul Skripsi dan Abstraksi data latih yang digambarkan atau di visualisasikan pada gambar 3 dimensi sumbu x-y-z. Visualiasi tersebut dapat menunjukkan titik resultant vektor masing-masing kata yang mirip pada judul skripsi dan isi abstraksi. Hal ini akan memudahkan pembaca memaknai tentang konsep *Vector Space Model (VSM)*.

- Minimal 1 tahun sekali dicari referensi untuk memastikan apakah algoritma *Cosine Similarity* dan *VSM* hasilnya masih relevan bagus sebagai metodologi Uji *Similarity* judul skripsi untuk meng-evaluasi kehandalan aplikasi yang dibuat.



DAFTAR PUSTAKA

PUSTAKA BUKU

- Kusrini & Emha L. T., 2009. *Algoritma Data Mining*, Penerbit Andi Yogyakarta
- B. Santosa , 2007. *Data Mining : Teknik Pemanfaatan Data Untuk Keperluan Bisnis*, Graha Ilmu Yogyakarta

PUSTAKA MAJALAH, JURNAL ILMIAH ATAU PROSIDING

- Prasetya, Didik & Wibawa, Aji & Hirashima, Tsukasa. (2018). *The performance of text similarity algorithms. International Journal of Advances in Intelligent Informatics*. 4. 10.26555/ijain.v4i
- Fajar Rohman Hariri, Ema Utami, Armadyah Amborowati, *Learning Vector Quantification* untuk Klasifikasi Abstrak Tesis, Magister Teknik Informatika, STMIK AMIKOM Yogyakarta,
- Wang, Jiapeng & Dong, Yihong. (2020). *Measurement of Text Similarity: A Survey. Information*. 11. 421. 10.3390/info11090421.
- Maake, Benard & Zuva, Tranos. (2018). *A Comparative Analysis of Text Similarity Measures and Algorithms in Research Paper Recommender Systems*. 10.1109/ICTAS.2018.8368766. P. Made, R. C. Dinatha, and N. A. Rakhmawati, "Komparasi Term"
- P. Made, R. C. Dinatha, and N. A. Rakhmawati. (2020). *Komparasi Term Weighting dan Word Embedding pada Klasifikasi Tweet Pemerintah Daerah (Comparison of Term Weighting and Word Embedding on Local Government Tweet Classification)*," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 9, no. 2, 2020
- M. Muliadi, M. R. Said, and E. Sofyan, "Sistem Pendeteksi Kemiripan Judul Skripsi Menggunakan Algoritma Winnowing,"(2021). *J. Pegguruang Conf. Ser.*, vol. 3, no. 1, p. 144, 2021, doi: 10.35329/jp.v3i1.1261.
- A. Filcha and M. Hayaty, "Implementasi Algoritma Rabin-Karp untuk Pendeteksi Plagiarisme pada Dokumen Tugas Mahasiswa,"(2019). *JUITA J. Inform.*, vol. 7, no. 1, p. 25, 2019, doi: 10.30595/juita.v7i1.4063
- M. Selma, S. Firdiyan, S. Nurirwan, "Sistem deteksi kemiripan judul skripsi dengan metode term weight dan word embedding"(2022). Vol 11 No 2 (2022): *Jurnal Dinamika Informatika* Vol.11 No.2

- Mutiara, Achmad & Agustina, Sinta. (2008). APLIKASI ANTI PLAGIATISME DENGAN ALGORITMA KARP-RABIN PADA PENULISAN ILMIAH UNIVERSITAS GUNADARMA. 10.13140/RG.2.1.1565.4169
- Basmalah, V. Dkk (2015). Analisis Perbandingan Metode *Vector Space Model* dan *Weighted Tree Similarity* dengan *Cosine Similarity* pada Kasus Pencarian Informasi Pedoman Pengobatan Dasar di Puskesmas. Vol. 4 No. 2 Desember 2015 : Jurnal ITSMART. ISSN : 2301-7201.
- Hussein, N., dkk (2022). *Short Text Semantic Measurement Approach Based on Semantic Network*. Baghdad Science Journal : 2022, 19 (6) : 1581-1591.
- Singh, Ritika & Singh, Satwinder (2021). *Text Similarity Measures in News Articles by Vector Space Model Using NLP*. J.Inst.Eng.India.Ser.B (April 2021) 102(2):329-338.
- Widhiani, N., Suhartika, I., & Haryanti, N. (2022). Persepsi Penggunaan Aplikasi Turnitin untuk mencegah tindakan plagiarisme pada mahasiswa Universitas Udayana. *Jurnal Ilmiah Ilmu Perpustakaan Dan Informasi*.
- Sophan, Mochammad & Kurniawati, Arik. (2018). Perancangan Aplikasi Learning By Doing Interaktif Untuk Mendukung Pembelajaran Bahasa Pemrograman. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 5. 163. 10.25126/jtiik.201852608.
- Wijayanto, Candra & Alfa, Yerima (2022). Implementasi Flask Framework pada Pembangunan Aplikasi Sistem Informasi *Helpdesk* (SIH). *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, Volume 07, Nomor 03.
- Utami, E., 2010. JawaTeX Web: Web Based Latin to Javanese Characters Transliteration System, *International Journal of Computer Science & Emerging Technologies (IJCSET)*, ISSN: 2044-6004, Vol. 1 Issue.1 June, 2010
- Zuliarso, Eri. 2010. "Aplikasi Web Crawler Berbasis Breadth First Search Dan Back-Link." *Fakultas Teknologi Informasi Universitas Stikubank Semarang XV(1): 52-56.*
- Sugiyamta. 2015. "Sistem Deteksi Kesamaan Dokumen Dengan Algoritma *Cosine Similarity* dan Single Pass Clustering." *Dinamika Informatika 7(2): 7.*
- Nurdiana, Ogie, Jumadi, dan Dian Nursantika. 2016. "Perbandingan Metode *Cosine Similarity* dengan Metode *Jaccard Similarity* pada Aplikasi Pencarian Terjemah Al-Qur'an Bahasa Indonesia." *Jurnal Online Informatika (GABUNG) 1(1): 59-63.*
- Santos, Hari. 2015. "Pencegahan dan Pengendalian Plagiat dalam Penulisan Karya Ilmiah di Perpustakaan Perguruan Tinggi Oleh: Drs. Hari Santoso, S.Sos. I." *Perpustakaan UM Malang (1): 1-23.*

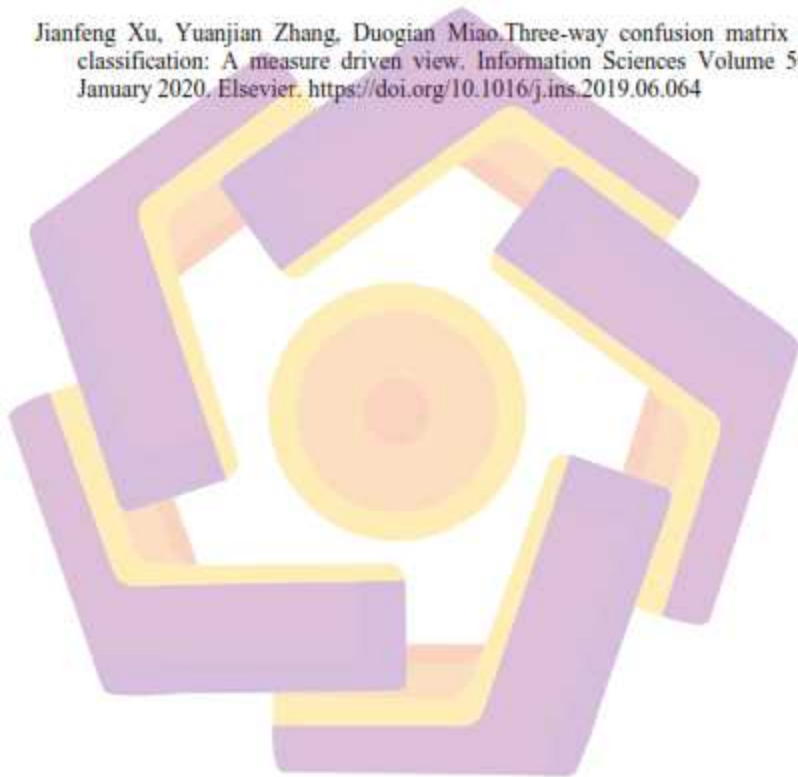
- Kumar, Manish, Ankit Bindal, Robin Gautam, dan Rajesh Bhatia. 2018. "Perayap Web Berfokus Kueri Berbasis Kata Kunci." *Ilmu Komputer Procedia* 125:584–90. <http://linkinghub.elsevier.com/retrieve/pii/S1877050917328399>.
- Rungsawang, Arnon, dan Niran Angkawattanawit. 2005. "Perayap Web Khusus Topik yang Dapat Dipelajari." *Jurnal Aplikasi Jaringan dan Komputer* 28(2): 97–114.
- Hutabarat, Suetha Ronarumata. (2016). "Tingkat Plagiat Pada Skripsi Mahasiswa Lulusan Program Studi Ilmu Perpustakaan Tahun 2015 Berdasarkan Plagiarism Checker X Scanner." *Medan: Program Studi Ilmu Perpustakaan dan Informasi Fakultas Ilmu Budaya Universitas Sumatera Utara*.
- Triana, A., Saptono, R., & Sulisty, ME 2014. Pemanfaatan Metode Vector Space Model dan Metode *Cosine Similarity* pada Fitur Deteksi Hama dan Penyakit Tanaman Padi. *ITSMART: Jurnal Teknologi dan Informasi*, 3(2), hlm .90-95.
- Ma'arif, Abdul Aziz. 2015. Pengaplikasian Algoritma Tf-Idf Untuk Pencarian Karya Ilmiah. *Jurnal. Jurusan Teknik Informatika. Fakultas Ilmu Komputer. Universitas Dian Nuswantoro Semarang*.
- Jiffriya, MAC, Jahan, MA, & Rage, RG 2014. Deteksi plagiarisme pada penugasan berbasis teks elektronik menggunakan model ruang vektor. *Dalam Konferensi Internasional ke-7 tentang Informasi dan Otomasi untuk Keberlanjutan* (hlm. 1-5). IEEE.
- Rozqa, A., Hardinto, RK, Yumida, R., & Faslah, R. (2018). Sistem Informasi Penelitian dan Pengabdian Masyarakat Guna Meningkatkan Kinerja P3M Politeknik Negeri Banjarmasin. *Jurnal Sains Terapan Teknologi Informasi*, 10(2), 36-41.
- Pratama, R. P., Faisal, M., & Hanani, A. (2019). Deteksi Plagiarisme pada Artikel Jurnal Menggunakan Metode *Cosine Similarity*. *SMARTICS Journal*, Vol.5 No. 1 2019, 22-26.
- Saeed, A. A. M., & Taqa, A. Y. (2022). A proposed approach for plagiarism detection in Article documents. *Sinkron: jurnal dan penelitian teknik informatika*, 7(2), 568-578.
- Ali, A., & Taqa, A. Y. (2022). Analytical Study of Traditional and Intelligent Textual Plagiarism Detection Approaches. *Journal of Education and Science*, 31(1), 8-25.
- Cader, J. M. A., Cader, A. J. M. A., Gamaarachchi, H., & Ragel, R. G. (2022). Optimisation of plagiarism detection using vector space model on CUDA architecture. *International Journal of Innovative Computing and Applications*, 13(4), 232-244.

- Davoodifard, M. (2022). Automatic Detection of Plagiarism in Writing. *Studies in Applied Linguistics & TESOL at Teachers College*, Vol. 21, No. 2, pp. 54–60
- Jambi, K.M, Khan, I.H. & Siddiqui, M.A. (2022). Evaluation of Different Plagiarism Detection Methods : A Fuzzy MCDM Perspective, *Applied Science*, 2022, 12 , 4580
- Nugraha, D. (2021). Penerapan Algoritma *Cosine Similarity* pada Aplikasi Bank Soal. *Jurnal Sains Komputer dan Teknologi Informasi*, Vol. 5, No. 2, pp. 81-86
- Ramadhanif, N., Subroto, I.M.I & Widiharini, A (2022). *Cosine Similarity* Retrieval untuk Prediksi Masa Studi Mahasiswa Berbasis Case Base Reasoning. *Jurnal Transistor Elektro dan Informatika (Transistor EI)*, Vol. 4, No. 2, pp. 142-150
- Rismayani dkk. (2022). Implementasi Algoritma Text Mining dan *Cosine Similarity* untuk Desain Sistem Aspirasi Publik berbasis Mobile. *Komputika : Jurnal Sistem Komputer*, Vol. 9, No. 2, pp. 169-176
- Utomo, Mardi S. dkk. (2023). Text Summarization Pada Artikel Berita Menggunakan Vector Space Model dan *Cosine Similarity*. *Dinamika Informatika*, Vol. 14, No. 1, pp. 11-24
- Sanjaya, Ardi dkk. (2023). Pengukuran Kemiripan Makna menggunakan *Cosine Similarity* dan basis Data Sinonim Kata. *Jurnal Teknologi Informasi dan Ilmu Komputer*, Vol. 10, No. 4, pp. 747-752
- Yunita, N.P.. (2023). Aplikasi Pencarian hadis menggunakan Vector Space Model dengan Pembobotan TF-IDF dan Confix Stripping Stemmer. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, Vol. 10, No. 3, pp. 665-676
- Prismadana, T.A. (2023). Aplikasi Ruang Tugas dengan Deteksi Kemiripan Teks pada Dokumen Tugas menggunakan *Cosine Similarity*. *Jurnal Informatika & Multimedia*, Vol. 15, No. 1, pp. 31-37

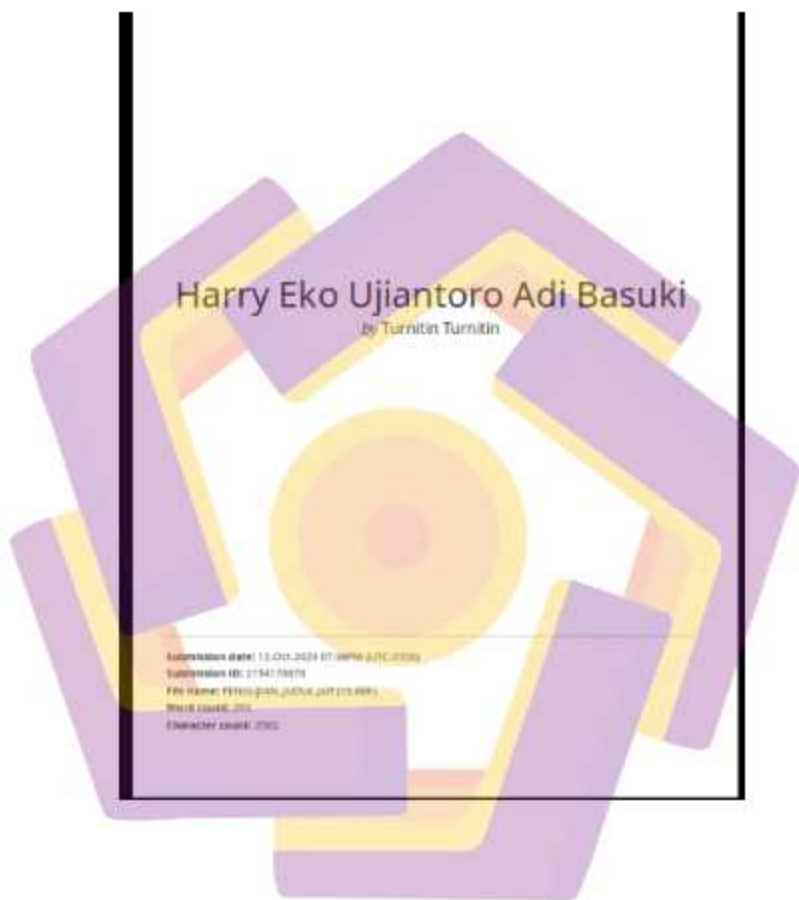
PUSTAKA LAPORAN PENELITIAN

- Pahlevi, Irfan, Moch Arief Bijaksana, dan M Ir Tech. "Perhitungan Kemiripan Dokumen Berbahasa Indonesia Menggunakan Metode *Cosine Similarity* (Studi Kasus: Abstrak Tugas Akhir Fakultas Informatika Universitas Telkom)."

- Rizki, Dhidik, dan Eko Suprpto. 2017. Pengaplikasian Algoritma *Cosine Similarity* dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen. Skripsi. Jurusan Teknik Elektro. Fakultas Teknik. Universitas Negeri Semarang Kampus Sekaran, Gunungpati, Semarang
- Wahyuni, RT, Prastiyanto, D., & Suprpto, E. (2017). Penerapan Algoritma *Cosine Similarity* dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi. Jurnal Teknik Elektro Vol. 9 No.1, 18-23.
- Jianfeng Xu, Yuanjian Zhang, Duogian Miao. Three-way confusion matrix for classification: A measure driven view. Information Sciences Volume 507, January 2020. Elsevier. <https://doi.org/10.1016/j.ins.2019.06.064>



LAMPIRAN



1. Pengembangan Aplikasi Perencanaan Risiko Perencanaan Usaha UMKM Sektor Usaha Jasa: Eksperimen Evaluasi Efektivitas Model Risiko Aplikasi menggunakan Metode Kualitatif
2. Perilaku Pengisian Saluran BMM sebagai Jasa Asuransi: Studi 2022 dengan Pendekatan Metode Learning
3. Enhancement of Customer Trust Online Purchase using Market Long Short-Term Memory
4. Analisis Tawar-Pilih Kupon Menggunakan Metode Two-Step Learning dengan Hyperparameter Tuning Gridsearch
5. Persepsi Mekanik Sepeda Adiktif Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda
6. Analisis Persepsi Masyarakat Saat Melakukan Kegiatan BWR Terhadap Efektivitas Metode Self-Test dan Self-Administered Wasting
7. Persepsi Mekanik Sepeda Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test
8. Pengembangan Sistem Informasi Sistem Informasi Manajemen Berbasis Web untuk Meningkatkan Efektivitas dan Efisiensi Kerja
9. Persepsi COVID-19 terhadap Perilaku Berbasis Self-Test dan Self-Administered Wasting
10. Analisis Persepsi Masyarakat Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test
11. Persepsi Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test
12. Analisis Persepsi Masyarakat Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test
13. Analisis Persepsi Masyarakat Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test
14. Analisis Persepsi Masyarakat Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test
15. Pengembangan Aplikasi Sistem Informasi Manajemen Berbasis Web untuk Meningkatkan Efektivitas dan Efisiensi Kerja
16. Analisis Persepsi Masyarakat Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test
17. Pengembangan Aplikasi Sistem Informasi Manajemen Berbasis Web untuk Meningkatkan Efektivitas dan Efisiensi Kerja
18. Analisis Persepsi Masyarakat Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test
19. Pengembangan Aplikasi Sistem Informasi Manajemen Berbasis Web untuk Meningkatkan Efektivitas dan Efisiensi Kerja
20. Analisis Persepsi Masyarakat Terhadap Penggunaan Jasa Perawatan Mekanik Sepeda dengan Metode Persepsi dan Perilaku Berbasis Kuesioner dan Self-Test

Harry Eko Ujiantoro Adi Basuki

UNIVERSITY REPORT

89%	89%	30%	18%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

RECENT SOURCES

1	www.scilit.net Internet Source	68%
2	www.jurnal.unmer.ac.id Internet Source	16%
3	www.sciencegate.app Internet Source	3%
4	garuda.kemdikbud.go.id Internet Source	2%

Exclude spaces

Exclude matches

Exclude bibliography











