

# Analisis Deteksi Kemiripan Dokumen Tugas Mahasiswa pada LMS Undiknas Menggunakan Metode *K-Shingling* dan *Cosine Similarity*

Komang Nova Artawan<sup>1</sup>, Made Sudarma<sup>2</sup>, Nyoman Gunantara<sup>3</sup>

[Submission: 30-01-2024, Accepted: 25-03-2024]

**Abstract**— Learning Management System (LMS) application at one of the universities in Bali, namely Undiknas, began to be developed since there was an obligation to carry out online learning when the COVID pandemic occur, and for now, LMS Undiknas has been used to support the implementation of the digitalization of distance learning process. One things that needs to be considered regarding the distance learning method is how to ensure that students understand the learning material. This can be done by giving assignments that must be done by students. However, giving assignments in LMS can also be an opening for students to cheat, as it is often found that students submission is duplicating another students submission. So, in this research authors aims to overcome this problem by analyzing the similarity detection feature of student assignment documents on the Undiknas LMS using the *K-Shingling* and *Cosine Similarity* methods, so that it can be used by lecturers to detect the similarities percentage of each student's assignment documents. Based on the *training* and *testing* stages, it is concluded that with data partition ratio of 70% (*training*) and 30% (*testing*), similarities detection in student assignment documents using text *pre-processing* and parameter value  $K = 9$  for the *K-Shingling* method obtain an accuracy value of 73.55% at the *testing* stage which shows the performance and success level of the system in detecting similarities in student assignment answer documents, and this value obtain higher accuracy value within 8.24% compared to the accuracy value at the *training* stage.

**Intisari**— Aplikasi LMS (*Learning Management System*) pada salah satu perguruan tinggi swasta di Bali yaitu Universitas Pendidikan Nasional (Undiknas) mulai dikembangkan sejak adanya kewajiban untuk melakukan pembelajaran secara daring saat pandemi COVID melanda, dan hingga saat ini penggunaan LMS Undiknas digunakan untuk menunjang implementasi dari proses pembelajaran jarak jauh agar dapat dilakukan secara digital. Salah satu hal yang perlu diperhatikan dari metode pembelajaran jarak jauh tersebut adalah terkait bagaimana memastikan bahwa mahasiswa telah paham dengan materi pembelajaran yang diberikan secara daring. Hal tersebut dapat dilakukan dengan memberikan tugas yang harus dikerjakan oleh mahasiswa. Namun, pemberian tugas melalui LMS juga dapat menjadi celah untuk mahasiswa melakukan kecurangan dengan kerap ditemukannya bahwa

antar mahasiswa melakukan duplikasi jawaban tugas dari mahasiswa yang lain. Sehingga, dalam penelitian ini dilakukan upaya untuk mengatasi permasalahan tersebut dengan melakukan analisis fitur deteksi kemiripan dokumen tugas mahasiswa pada LMS Undiknas dengan metode *K-Shingling* dan *Cosine Similarity* agar dapat digunakan oleh dosen untuk mendeteksi persentase kemiripan dari dokumen pengumpulan tugas tiap mahasiswa. Berdasarkan tahap *training* dan tahap *testing* yang telah dilakukan, didapatkan kesimpulan bahwa pada rasio partisi data 70% (*training*) dan 30% (*testing*), deteksi kemiripan dokumen tugas mahasiswa dengan menggunakan *pre-processing* teks dan nilai parameter  $K = 9$  pada metode *K-Shingling* diperoleh nilai akurasi sebesar 73,55% pada tahap *testing* yang menunjukkan performa dan tingkat keberhasilan sistem dalam melakukan deteksi kemiripan dokumen jawaban tugas mahasiswa, dan nilai akurasi pada tahap *testing* ini lebih tinggi 8.24% dibandingkan nilai akurasi pada tahap *training*.

**Kata Kunci**— LMS Undiknas, Kemiripan Dokumen Tugas, *K-Shingling*, *Cosine Similarity*

## I. PENDAHULUAN

*Learning Management System* (LMS) merupakan sebuah aplikasi perangkat lunak yang digunakan untuk mengatur dan memberikan konten pembelajaran yang mencakup materi, silabus, tugas, kuis dan ujian [1]. Penggunaan LMS mulai banyak diterapkan dalam bidang pendidikan terlebih sejak pandemi COVID melanda, dimana seluruh kegiatan terpaksa harus dilakukan secara daring [2], dimana hal tersebut juga berdampak pada aktivitas pembelajaran, mulai dari penyampaian materi, hingga pengumpulan tugas harus dapat tersinkronisasi secara daring.

Pada salah satu perguruan tinggi swasta di Bali, yakni Universitas Pendidikan Nasional (Undiknas), juga sudah mulai melakukan pengembangan aplikasi LMS sejak adanya kewajiban untuk melakukan pembelajaran secara daring saat pandemi COVID. Hingga saat ini, LMS yang telah dikembangkan juga digunakan untuk menunjang implementasi dari pembelajaran jarak jauh. Dengan adanya metode pembelajaran jarak jauh ini, dosen harus dapat mengunggah materi pembelajaran yang dapat diakses oleh mahasiswa, menyediakan ruang sebagai forum untuk diskusi dengan mahasiswa, serta dapat mengevaluasi mahasiswa dengan memberikan beberapa tugas dan kuis, sedangkan mahasiswa harus dapat mengakses materi pembelajaran yang telah disediakan dosen, berpartisipasi pada forum diskusi yang tersedia, dan dapat mengerjakan kuis dan juga melakukan pengumpulan tugas. Berdasarkan beberapa kebutuhan umum pada proses pembelajaran tersebut, maka pada aplikasi LMS

<sup>1</sup>Mahasiswa, Magister Teknik Elektro Fakultas Teknik Universitas Udayana, Jln. P.B Sudirman Denpasar-Bali 80234 INDONESIA (telp: 089662176056; e-mail: [komang.nopha@gmail.com](mailto:komang.nopha@gmail.com))

<sup>2, 3</sup> Magister Teknik Elektro Fakultas Teknik Universitas Udayana, Jln. P.B Sudirman Denpasar-Bali 80234 INDONESIA (e-mail: [msudarma@unud.ac.id](mailto:msudarma@unud.ac.id); [gunantara@unud.ac.id](mailto:gunantara@unud.ac.id))



Undiknas juga telah dikembangkan fitur – fitur untuk dapat menunjang proses pembelajaran jarak jauh agar dapat berjalan dengan lancar.

Salah satu aspek yang perlu diperhatikan dan menjadi konsen dari metode pembelajaran jarak jauh ini adalah terkait bagaimana memantau proses belajar dari mahasiswa, bagaimana cara memastikan bahwa mahasiswa telah mengakses materi pembelajaran yang diberikan, dan juga bagaimana memastikan bahwa mahasiswa telah paham dengan materi pembelajaran yang diberikan. Oleh karena itu, salah satu cara yang dapat dilakukan oleh dosen adalah dengan memberikan tugas terhadap mahasiswa terkait dengan materi pembelajaran yang diberikan secara daring, disamping itu pemberian tugas juga berpengaruh terhadap penilaian mahasiswa itu sendiri [3]. Namun metode pengumpulan tugas secara daring dengan mengunggah berupa dokumen ini membuka peluang munculnya plagiarisme [4], yang dapat dilakukan oleh mahasiswa dengan meniru dan menduplikasi jawaban tugas dari mahasiswa yang lain, sehingga jawaban tugas mereka memiliki persentase kemiripan yang tinggi. Plagiarisme yang dilakukan tersebut tidak hanya merugikan penulis yang karyanya telah disalin, tetapi juga menyebabkan kurangnya kreativitas dan membentuk mental yang buruk bagi mereka yang melakukannya [5]. Oleh karena itu, maka pada penelitian ini dilakukan upaya untuk mengatasi permasalahan yang muncul tersebut dengan mengimplementasikan fitur pada menu tugas di aplikasi LMS Undiknas, sehingga dosen dapat melakukan deteksi persentase kemiripan dokumen tugas mahasiswa secara otomatis, satu sama lain ketika mahasiswa telah melakukan pengumpulan jawaban tugas dengan mengunggah jawaban mereka dalam bentuk file dokumen *.pdf*.

Beberapa hal yang menjadi batasan dari penelitian yang dilakukan ini adalah penelitian ini mengambil studi kasus pada aplikasi LMS Undiknas sebagai sumber data yang digunakan dan juga pengembangan fitur deteksi kemiripan dokumen tugas dari penelitian ini akan diaplikasikan pada LMS Undiknas. Kemudian data yang akan digunakan dalam penelitian ini adalah berupa file dokumen berekstensi *.pdf*, dimana data dokumen tugas mahasiswa yang diproses hanya yang berisi teks saja, tidak termasuk dokumen yang berisi jawaban dalam bentuk gambar yang dimasukkan ke dalam file *.pdf*. Proses perbandingan kemiripan yang diterapkan dalam penelitian ini dilakukan antar dokumen tugas mahasiswa saja, tidak membandingkan kemiripan dokumen tugas mahasiswa dengan dokumen lain yang bersumber dari internet, dan hasil deteksi kemiripan dokumen tugas mahasiswa yang ditampilkan adalah berupa persentase kemiripan dari masing – masing dokumen.

Sebagai referensi pada penelitian ini, telah dilakukan tinjauan pustaka dari beberapa jurnal penelitian terkait, salah satunya adalah penelitian yang dilakukan oleh Salmuasih, dkk pada tahun 2013 dengan judul Implementasi Algoritma Rabin Karp untuk Pendeteksian Plagiat Dokumen Teks Menggunakan Konsep Similarity, dengan menggunakan metode Rabin Karp mendapatkan hasil bahwa penggunaan *stemming* dalam *pre-processing* dapat mengurangi waktu pemrosesan, tetapi menghasilkan akurasi *similarity* yang rendah, sedangkan nilai modulo dan *k-gram* yang semakin kecil akan menghasilkan

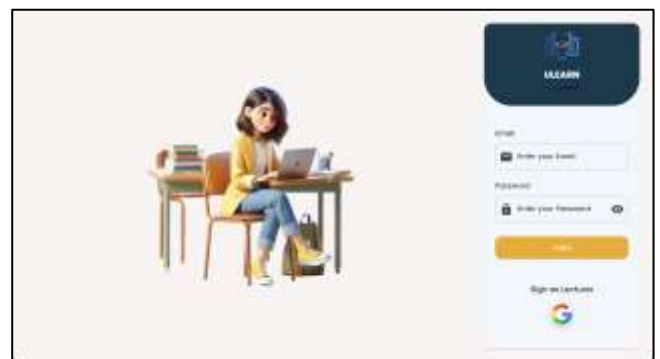
nilai *similarity* yang lebih baik [6]. Kemudian penelitian yang dilakukan oleh Manaa, dkk pada tahun 2018 dengan judul *Web Documents Similarity Using K-Shingle Tokens and Minhash Technique* menggunakan metode *K-Shingle* yang dikombinasikan dengan metode hashing *LSH (Locally Sensitive Hashing)* yang bertujuan untuk mengecilkan ukuran matriks vektor nilai yang dihasilkan dari proses *shingle* dokumen berdasarkan jumlah hashing yang akan dilakukan. Hasil evaluasi yang dilakukan pada beberapa nilai *k* menunjukkan bahwa nilai *k* yang terbaik untuk menentukan *similarity* dokumen adalah *k = 5*. [7]

Berdasarkan referensi dari beberapa jurnal tersebut, maka pada penelitian ini akan dilakukan analisis terhadap metode *K-Shingling* dan *Cosine Similarity* yang digunakan untuk mendeteksi kemiripan dokumen tugas mahasiswa sehingga hasil analisis ini dapat menjadi dasar untuk pengembangan fitur yang akan diimplementasikan pada LMS Undiknas. Analisis dalam penelitian ini akan dilakukan dengan menggunakan beberapa percobaan untuk mendapatkan nilai parameter yang terbaik, diantaranya adalah percobaan dengan skema menggunakan dan tanpa menggunakan *pre-processing* teks, percobaan dengan menggunakan beberapa rasio partisi data, dan percobaan menggunakan beberapa nilai *K* pada metode *K-Shingling*.

## II. STUDI PUSTAKA

### A. LMS Undiknas

LMS Undiknas dikembangkan dengan menggunakan framework *cross-platform* yaitu flutter, saat ini LMS Undiknas digunakan sebagai aplikasi berbasis web yang dapat diakses oleh dosen dan mahasiswa pada alamat berikut : <https://lms.undiknas.ac.id>



Gambar 1: Tampilan Login LMS Undiknas

Gambar 1 menunjukkan tampilan awal yakni halaman login LMS Undiknas, mahasiswa dapat login menggunakan email dan password yang telah didaftarkan, sedangkan dosen dapat login dengan menggunakan akun email google, setelah itu pengguna dapat mengakses fitur didalam LMS untuk mendukung proses pembelajaran secara daring. Beberapa fitur yang terdapat pada LMS Undiknas antara lain adalah sebagai berikut :

### 1. Kelas Daring

LMS Undiknas mendukung beberapa fitur seperti yang terdapat pada LMS umumnya yaitu *moodle* dan *google classroom*, sehingga LMS Undiknas dapat digunakan untuk menunjang proses pembelajaran secara daring. Pada LMS Undiknas, dosen dan mahasiswa akan terdaftar secara otomatis pada kelas matakuliah mereka, dimana dosen terdaftar sesuai dengan matakuliah yang mereka ajar, sedangkan mahasiswa terdaftar sesuai dengan matakuliah yang mereka tempuh dalam satu semester.



Gambar 2: Halaman Daftar Seluruh Kelas/Course

Pada gambar 2 terlihat halaman dosen dan mahasiswa yang berisi daftar keseluruhan matakuliah mereka. Dosen dan mahasiswa dapat masuk ke halaman *dashboard* matakuliah, setelah itu dosen dapat mengisi konten dengan mengunggah materi pembelajaran yang berupa materi dalam bentuk teks, file dokumen, dan juga video. Sedangkan mahasiswa akan dapat mengakses seluruh materi pembelajaran, mengumpulkan tugas, dan kuis yang telah diberikan oleh dosen. Gambar 3 dibawah ini menunjukkan halaman *dashboard* yang terlihat oleh dosen dan mahasiswa.



Gambar 3: Halaman *Dashboard* Suatu Kelas/Course

### 2. Forum Diskusi

Pada menu ini, dosen dengan mahasiswa dan juga antar mahasiswa dapat saling berinteraksi satu sama lain dalam kelas daring untuk membahas terkait suatu materi

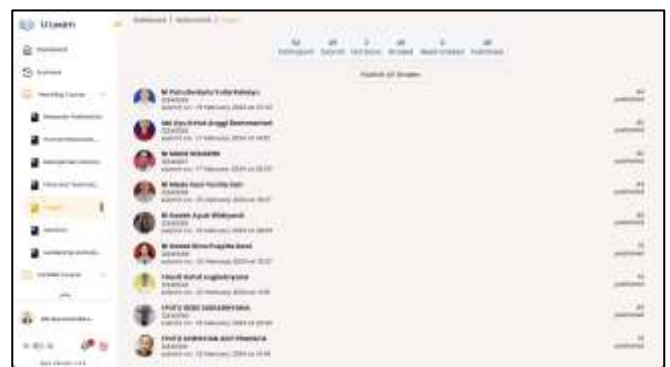
pembelajaran. Gambar 4 dibawah ini menunjukkan halaman forum diskusi yang dapat digunakan oleh dosen dan mahasiswa untuk berinteraksi dalam proses pembelajaran.



Gambar 4: Halaman Forum Diskusi

### 3. Tugas dan Kuis

Fitur ini memungkinkan dosen untuk memberikan tugas dan kuis kepada mahasiswa dengan tujuan untuk mengevaluasi tingkat pemahaman siswa terhadap materi pembelajaran yang telah diberikan. Ini membantu siswa meningkatkan pemahaman mereka tentang topik pembelajaran dan meningkatkan kemampuan mereka untuk menyampaikan informasi secara verbal [8]. Dosen juga dapat menggunakan fitur ini untuk melihat hasil pengumpulan jawaban tugas mahasiswa, dan dari sisi mahasiswa mereka dapat menggunakan fitur ini untuk melakukan pengumpulan tugas dengan mengunggah jawaban mereka.



Gambar 5: Halaman Dosen Melihat Pengumpulan Tugas Mahasiswa

Gambar 5 menunjukkan halaman menu yang dapat diakses oleh dosen untuk melihat pengumpulan tugas mahasiswa. Halaman ini yang akan dikembangkan lebih lanjut pada penelitian ini dengan mengimplementasikan fitur untuk dapat melihat kemiripan dari masing – masing jawaban mahasiswa, sehingga dosen akan mengetahui apabila ada jawaban tugas dari mahasiswa yang melakukan plagiarisme.



## B. K-Shingling

Algoritma *Shingling* melihat dokumen sebagai satu set *n-gram* dalam bentuk sirap atau rangkaian kata pendek (*shingle*) dalam teks yang saling tumpang tindih [9].

*Shingle* mengacu pada sekelompok kata berurutan yang berdekatan dalam dokumen. Algoritma berbasis sirap mengharuskan serangkaian sirap dipilih dari dokumen, dan kemudian sirap dipetakan ke dalam tabel nilai [10].

Algoritma ini bekerja dengan membentuk *shingle* yang terdiri dari sejumlah kata yang telah ditentukan [11], kemudian untuk setiap *shingle* akan dipetakan menjadi suatu nilai *shingle fingerprint* [12] yang menandakan *shingle* tersebut apakah berada dalam suatu dokumen atau tidak. Dalam membentuk *shingle*, terdapat parameter yang dalam penelitian ini disebut sebagai variabel *K*, variabel ini akan menentukan jumlah kata dalam satu *shingle*.

Analisis terhadap pemilihan nilai variabel *K* ini menjadi salah satu hal yang berpengaruh terhadap hasil deteksi kemiripan dokumen. Sebab jika nilai *k* terlalu kecil, maka sebagian besar sirap dengan rangkaian *k* karakter akan muncul di sebagian besar dokumen. Jika demikian, maka kita memiliki dokumen yang rangkaian sirapnya memiliki kemiripan yang tinggi, namun dokumen tersebut tidak memiliki kalimat atau bahkan frasa yang sama [13]. Sebagai contoh, jika kita menggunakan nilai  $k = 1$ , maka dokumen akan memiliki sebagian besar karakter umum yang sama dan memiliki tingkat kesamaan yang tinggi. Pemilihan nilai *k* seharusnya bergantung pada berapa panjang dokumen dan jumlah kumpulan karakter didalamnya, dan hal penting yang perlu diperhatikan adalah nilai *k* yang dipilih harus cukup besar sehingga akan rendah kemungkinan munculnya sirap tertentu dalam suatu dokumen. [7].

Nilai *shingle fingerprint* yang berbentuk vektor nilai akan dibandingkan untuk setiap dokumen yang akan dibandingkan [14], sehingga nilai *shingle fingerprint* ini menandakan hubungan antar dokumen yang dapat digunakan untuk menghitung kemiripan dokumen. *Proses yang dilakukan dalam algoritma K-Shingling sebagai berikut :*

### 1. Pembentukan *shingle*

*Shingle* yang dibentuk merupakan rangkaian kata yang saling tumpang tindih, oleh karena itu proses pembentukan ini akan berjalan secara berurutan dimulai dari kata pertama sampai *K* kata setelahnya, dilanjutkan dengan kata kedua sampai *K* kata setelahnya, dan seterusnya sampai kata terakhir pada dokumen, sehingga akan terbentuk sejumlah *shingle* dari masing-masing dokumen yang terdiri dari *K* kata.

Misalnya terdapat dokumen A dan dokumen B yang akan dibandingkan, dalam dokumen A terdapat teks “*Value suatu service adalah sebuah value mengacu pada hubungan antara nilai aktual produk dan nilai yang dirasakan produk dalam kaitannya dengan produk lain dalam kategori yang sama*”, dan dalam dokumen B terdapat teks “*Value dari suatu layanan adalah suatu nilai atau kualitas yang membedakan suatu produk dari produk yang lain*” jika akan dibentuk *shingle* yang terdiri dari 5 kata (*K*

= 5) maka akan terbentuk *shingle* dari dokumen A seperti berikut [['Value', 'suatu', 'service', 'adalah', 'sebuah'], ['suatu', 'service', 'adalah', 'sebuah', 'value'], ['service', 'adalah', 'sebuah', 'value', 'mengacu'], ['adalah', 'sebuah', 'value', 'mengacu', 'pada'], ['sebuah', 'value', 'mengacu', 'pada', 'hubungan'], ... , ['kaitannya', 'dengan', 'produk', 'lain', 'dalam'], ['dengan', 'produk', 'lain', 'dalam', 'kategori'], ['produk', 'lain', 'dalam', 'kategori', 'yang'], ['lain', 'dalam', 'kategori', 'yang', 'sama.']], sedangkan pada dokumen B akan terbentuk *shingle* seperti berikut [['Value', 'dari', 'suatu', 'layanan', 'adalah'], ['dari', 'suatu', 'layanan', 'adalah', 'suatu'], ['suatu', 'layanan', 'adalah', 'suatu', 'nilai'], ['layanan', 'adalah', 'suatu', 'nilai', 'atau'], ['adalah', 'suatu', 'nilai', 'atau', 'kualitas'], ... , ['membedakan', 'suatu', 'produk', 'dari', 'produk'], ['suatu', 'produk', 'dari', 'produk', 'yang'], ['produk', 'dari', 'produk', 'yang', 'lain.']].

Setelah terbentuk *shingle* dari setiap dokumen yang akan dibandingkan, kemudian bentuk *shingle* gabungan yang bersifat unik dan tidak ada duplikasi *shingle* didalamnya.

### 2. Hitung nilai *shingle fingerprint*

Setelah *shingle* telah terbentuk dari setiap dokumen yang dibandingkan, kemudian dibangkitkan nilai *shingle fingerprint* yang menjadi nilai vektor dokumen.

Misalnya dari *shingle* dokumen A dan dokumen B diatas telah terbentuk gabungan *shingle* seperti berikut ini [['Value', 'suatu', 'service', 'adalah', 'sebuah'], ['suatu', 'service', 'adalah', 'sebuah', 'value'], ['service', 'adalah', 'sebuah', 'value', 'mengacu'], ['adalah', 'sebuah', 'value', 'mengacu', 'pada'], ['sebuah', 'value', 'mengacu', 'pada', 'hubungan'], ... , ['membedakan', 'suatu', 'produk', 'dari', 'produk'], ['suatu', 'produk', 'dari', 'produk', 'yang'], ['produk', 'dari', 'produk', 'yang', 'lain.']] kemudian nilai *shingle fingerprint* dari setiap dokumen akan didapatkan seperti pada tabel berikut :

TABEL 1

NILAI SHINGLE FINGERPRINT SUATU DOKUMEN

<i>Shingle</i> (K=5)	<i>Shingle Fingerprint Dokumen A</i>	<i>Shingle Fingerprint Dokumen B</i>
'Value', 'suatu', 'service', 'adalah', 'sebuah'	1	0
'suatu', 'service', 'adalah', 'sebuah', 'value'	1	0
'service', 'adalah', 'sebuah', 'value', 'mengacu'	1	0
'adalah', 'sebuah', 'value', 'mengacu', 'pada'	1	0
'sebuah', 'value', 'mengacu', 'pada', 'hubungan'	1	0
'Value', 'dari', 'suatu', 'layanan', 'adalah'	1	1
'dari', 'suatu', 'layanan', 'adalah', 'suatu'	0	1
'suatu', 'layanan', 'adalah', 'suatu', 'nilai'	0	1

'layanannya', 'adalah', 'suatu', 'nilai', 'atau'	1	1
'adalah', 'suatu', 'nilai', 'atau', 'kualitas'	0	1
'membedakan', 'suatu', 'produk', 'dari', 'produk'	0	1

Pada Tabel 1 menunjukkan nilai *shingle fingerprint* dari masing – masing dokumen yang dibandingkan, nilai 1 diberikan apabila *shingle* tersebut ditemukan dalam suatu dokumen, sedangkan nilai 0 diberikan jika *shingle* tidak ditemukan dalam dokumen. Sehingga, hasil yang didapatkan dari proses pembentukan *shingle fingerprint* ini adalah sebuah vektor nilai dari masing – masing dokumen yang menandakan hubungan dari dokumen, semakin banyak kesamaan nilai *shingle fingerprint* maka semakin besar kemiripan dokumen tersebut.

### C. Cosine Similarity

*Cosine Similarity* adalah metode yang dapat digunakan untuk mengukur kesamaan (*similarity*) antar dua buah objek dalam bentuk vektor [15], yang dinyatakan dalam sebuah ruang dimensi yang didapat dari nilai cosinus sudut dari perkalian dua buah vektor yang akan dibandingkan [8]. Rumus dari *Cosine Similarity* dapat dilihat pada rumus sebagai berikut [16] :

$$\cos(d_j, q_k) = \frac{\sum_{i=1}^n (td_i \times tq_i)}{\sqrt{\sum_{i=1}^n td_i^2 \times \sum_{i=1}^n tq_i^2}} \quad (1)$$

Berdasarkan persamaan 1, untuk mendapatkan nilai *Cosine Similarity* dilakukan proses penjumlahan dari seluruh hasil perkalian setiap elemen dalam vektor nilai *shingle* setiap dokumen (*dot product*), kemudian dibagi dengan nilai akar dari penjumlahan seluruh hasil perkalian dari nilai kuadrat setiap elemen vektor nilai *shingle*. Dalam penelitian ini, metode *Cosine Similarity* diterapkan setelah melakukan metode *K-Shingling*, sehingga yang akan diproses pada *Cosine Similarity* merupakan nilai *shingle fingerprint* yang terbentuk dari kedua dokumen yang akan dibandingkan.

*Cosine Similarity* menghitung kesamaan antar dua buah dokumen berdasarkan besar sudut cosinus [17], sehingga nilai yang dihasilkan dari perhitungan *Cosine Similarity* memiliki rentang nilai dari 0 sampai dengan 1, yang menandakan bahwa semakin mirip suatu dokumen, maka nilai kemiripannya mendekati 1.

Sebagai contoh berdasarkan nilai *shingle fingerprint* yang telah terbentuk dari proses *K-Shingling* diatas, dokumen A memiliki vektor nilai *shingle fingerprint* yaitu [1,1,1,1,1,1,0,0,1,0,0] dan dokumen B memiliki vektor nilai *shingle fingerprint* yakni [0,0,0,0,0,1,1,1,1,1], maka  $n = 11$  (panjang vektor nilai), , sehingga nilai *Cosine Similarity* yang dihasilkan adalah sebagai berikut :

$$\cos(A, B) = \frac{(1 \times 0) + (1 \times 0) + (1 \times 0) + \dots + (1 \times 1) + (0 \times 1) + (0 \times 1)}{\sqrt{(1^2 + 1^2 + 1^2 + \dots + 0^2 + 0^2)} \times \sqrt{(0^2 + 0^2 + 0^2 + \dots + 1^2 + 1^2)}}$$

Komang Nova Artawan: Analisis Kemiripan Dokumen Tugas...

$$\cos(A, B) = \frac{0 + 0 + 0 + \dots + 1 + 0 + 0}{\sqrt{7} \times \sqrt{6}} = \frac{2}{\sqrt{42}} = 0.3086067$$

Nilai hasil dari perhitungan *Cosine Similarity* ini akan menunjukkan persentase kemiripan dari dua dokumen, maka nilai *Cosine Similarity* akan dikalikan dengan 100% sehingga persentase kemiripan dokumen menjadi 30.86%.

## III. METODOLOGI

Untuk melakukan implementasi dan analisis hasil dalam penelitian ini, dilakukan beberapa tahapan sebagai metodologi yang melalui beberapa tahapan sebagai berikut :

### A. Data Penelitian

Data yang digunakan dalam penelitian ini adalah data primer yang diperoleh secara langsung dari sumbernya yakni LMS Undiknas, dimana data tersebut merupakan file dokumen yang merupakan pengumpulan tugas essay mahasiswa yang diunggah pada setiap tugas yang berkaitan. File dokumen yang digunakan adalah dalam bentuk pdf, dan hanya dokumen yang berisi jawaban dalam teks saja, sehingga dokumen yang berisi jawaban dalam bentuk gambar tidak akan digunakan pada penelitian ini.

Jumlah dokumen pdf jawaban tugas mahasiswa yang telah dikumpulkan pada sistem LMS Undiknas saat ini berjumlah kurang lebih 1000 dokumen tugas, yang tersebar pada 36 matakuliah. Dokumen pdf tersebut akan diseleksi untuk mendapatkan dokumen yang hanya berisi teks saja agar dapat diolah dalam penelitian ini. Kemudian dari total keseluruhan data tersebut akan dibagi menjadi data pelatihan (*training*) dan data pengujian (*testing*), untuk rasio perbandingan data *training* dan *testing* yang akan digunakan juga akan melalui skema evaluasi, dimana terdapat beberapa acuan rasio partisi data, yakni 60% data *training* – 40% data *testing*, 70% data *training* – 30% data *testing*, dan 80% data *training* – 20% data *testing* yang akan diuji coba untuk mengetahui rasio partisi data yang menghasilkan nilai evaluasi terbaik.

Beberapa acuan rasio partisi data tersebut berdasarkan pada beberapa penelitian lain yang telah dilakukan, diantaranya adalah penelitian yang dilakukan oleh (Ratniasih et al., 2017) yang menyatakan hasil bahwa persentase akurasi sistem tertinggi dihasilkan dengan perbandingan jumlah data pelatihan dan pengujian adalah 70% : 30% pada kasus spam filtering [18], dan penelitian yang dilakukan oleh (Ulgasesa et al., 2022) menyatakan bahwa rasio partisi data latih dan data uji 80% : 20% memperoleh hasil performa tertinggi pada kasus klasifikasi sentiment [19].

### B. Prosedur Penelitian

Dalam penelitian ini akan dilakukan dalam dua tahapan yaitu tahap *training* dan *testing* yang memiliki tujuan sebagai berikut :

- Tahap *training* bertujuan untuk mengetahui nilai terbaik dari parameter nilai *K* pada metode *K-Shingling* berdasarkan data *training* yang akan digunakan. Pada p-ISSN:1693 – 2951; e-ISSN: 2503-2372



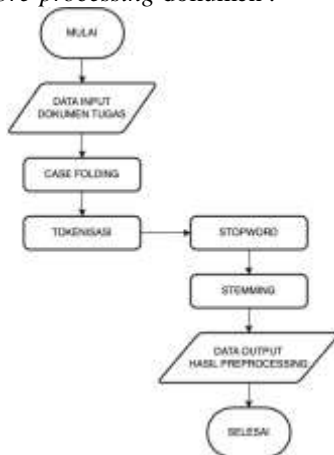
tahap ini dilakukan beberapa skenario uji coba, diantaranya uji coba skema dengan menggunakan dan tanpa menggunakan *pre-processing* teks, uji coba menggunakan beberapa rasio partisi data (60 : 40, 70 : 30, dan 80 : 20), dan uji coba nilai  $K$  (3, 5, 7, 9, 11). Untuk setiap skenario uji coba, akan dihitung nilai akurasi dan waktu pemrosesan dari proses deteksi kemiripan dokumen tugas mahasiswa sebagai acuan untuk menentukan nilai parameter yang terbaik.

- Tahap *testing* bertujuan untuk mendapatkan nilai akurasi dari proses deteksi kemiripan dokumen tugas mahasiswa pada data *testing* dan menggunakan nilai parameter terbaik yang telah didapatkan pada tahap *training*. Nilai akurasi ini menjadi hasil akhir dari penelitian yang dinyatakan sebagai tingkat keberhasilan sistem dalam melakukan deteksi kemiripan dokumen tugas mahasiswa.

Pada tahap *training* dan tahap *testing* akan dilakukan proses deteksi kemiripan dokumen tugas mahasiswa pada data *training* dan *testing* untuk mengetahui nilai akurasi pada tiap tahapannya. Proses deteksi kemiripan dokumen ini akan melalui beberapa langkah yang akan dijabarkan sebagai berikut :

### 1. Pre-processing dokumen

Langkah ini bertujuan untuk membersihkan dan mengubah format data teks dalam dokumen sebelum diproses pada metode *K-Shingling*. Berikut adalah *flowchart* dari *pre-processing* dokumen :



Gambar 6: Flowchart Pre-Processing Teks

Gambar 6 menunjukkan tahapan yang dilakukan untuk *pre-processing* teks dalam penelitian ini. Setiap tahapan dalam *pre-processing* teks memiliki pengaruh terhadap hasil evaluasi klasifikasi teks [19]. *Case folding* dilakukan untuk mengubah format teks menjadi format yang seragam yakni mengubah semua teks menjadi huruf kecil (*lowercase*), menghilangkan angka, menghapus karakter ganda dan menghapus spasi yang berlebihan, tokenisasi digunakan untuk memisahkan teks menjadi unit – unit yang lebih kecil, yaitu menjadi kata – kata yang disebut dengan *token*, kemudian *filtering* atau *stopword* untuk menghapus kata tidak memberikan informasi khusus dalam makna

suatu kalimat, dan *stemming* atau *lemmatization* untuk mengubah kata ke dalam bentuk dasar.

### 2. K-Shingling

Proses *K-Shingling* dilakukan pada masing - masing teks dari suatu dokumen yang akan diuji kemiripannya, dan hasil keluaran dari metode ini berupa vektor nilai sebagai matriks identitas atau *fingerprint* setiap dokumen, sehingga masing – masing vektor nilai tersebut dapat dihitung untuk mendapatkan suatu nilai yang menyatakan persentase kemiripan dokumen. Berikut adalah *flowchart* dari proses *K-Shingling*.



Gambar 7: Flowchart Metode K-Shingling

Pada gambar 7 terlihat bahwa yang menjadi parameter awal dari metode *K-Shingling* ini adalah nilai  $K$  yang digunakan untuk menentukan berapa jumlah kata yang akan dikelompokkan menjadi satu komponen *shingle* [20]. Setelah seluruh *shingle* terbentuk dari tiap dokumen, kemudian *shingle* tersebut digabungkan dan dibersihkan dari duplikasi. Gabungan *shingle* tersebut kemudian akan digunakan untuk mendapatkan suatu vektor nilai sebagai *shingle fingerprint* dari setiap dokumen sebagai nilai *output* dari metode ini. Selanjutnya nilai vektor tersebut akan diproses pada metode *Cosine Similarity* untuk mendapatkan nilai persentase kemiripan dokumen.

### 3. Cosine Similarity

*Cosine Similarity* digunakan untuk menghitung nilai kemiripan dari dokumen yang dibandingkan, data masukan yang digunakan untuk menentukan nilai kemiripan dokumen adalah vektor nilai dari *shingle fingerprint* masing-masing dokumen yang merupakan output dari metode *K-Shingling*. Berikut adalah *flowchart* dari metode *Cosine Similarity*.



Gambar 8: Flowchart Metode Cosine Similarity

Gambar 8 menunjukkan bahwa hasil akhir dari metode *Cosine Similarity* adalah label klasifikasi status kemiripan dokumen yang ditentukan berdasarkan nilai persentase kemiripannya, yang pada penelitian ini menggunakan rentang nilai ambang batas (*threshold*) dengan acuan sebagai berikut [21] :

- persentase 0% dilabelkan sebagai dokumen yang tidak memiliki kemiripan (*zero similarity*).
- persentase 1% - 24% dilabelkan sebagai dokumen memiliki sedikit kemiripan (*low degree of similarity*),
- persentase diantara 25% - 49% dilabelkan sebagai dokumen dengan tingkat kemiripan sedang (*moderate degree of similarity*).
- persentase 50% - 74% dilabelkan sebagai dokumen yang memiliki tingkat kemiripan tinggi (*high degree of similarity*).
- persentase 75% - 100% dilabelkan sebagai dokumen yang memiliki tingkat kemiripan sangat tinggi (*very high degree of similarity*).

#### 4. Pengukuran dan Evaluasi

Metrik evaluasi yang digunakan untuk mengukur performa dari implementasi metode didalam penelitian ini adalah dengan menggunakan *confusion matrix* dan nilai *accuracy*. Berikut adalah penjelasan dari metrik evaluasi yang digunakan dalam penelitian ini [20] :

- *True Positive (TP)* merupakan jumlah dokumen duplikat yang benar diklasifikasi oleh sistem dinyatakan sebagai dokumen duplikat.
- *False Positive (FP)* merupakan jumlah dokumen duplikat yang salah diklasifikasi oleh sistem dinyatakan sebagai dokumen non duplikat.
- *True Negative (TN)* merupakan jumlah dokumen yang tidak duplikat yang benar diklasifikasi oleh sistem dinyatakan sebagai dokumen non duplikat.

• *False Negative (FN)* merupakan jumlah dokumen yang tidak duplikat yang salah diklasifikasi oleh sistem dinyatakan sebagai dokumen duplikat.

• *Accuracy* merupakan hasil dari rasio jumlah dokumen yang diklasifikasi benar oleh sistem terhadap keseluruhan jumlah dokumen yang diuji. Persamaan untuk mendapatkan nilai ini adalah sebagai berikut :

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

Berdasarkan persamaan 2, maka untuk bisa melakukan pengukuran ini terlebih dahulu akan dilakukan pemberian label klasifikasi status kemiripan dokumen berdasarkan nilai kemiripannya. Setiap dokumen data yang telah dikumpulkan terlebih dahulu akan dibuat bentuk pasangan data (dokumen A – dokumen B), dan kemudian akan dihitung nilai kemiripannya dengan menggunakan web berikut <https://www.check-plagiarism.com/plagiarism-comparison-search>. Nilai kemiripan dari pasangan data dokumen yang didapat melalui web tersebut akan diubah menjadi label klasifikasi sesuai dengan acuan rentang nilai batas yang digunakan, kemudian label klasifikasi tersebut dinyatakan sebagai kelas sebenarnya (*Actual Class*).

Selanjutnya, pada tiap pasangan data diberikan label hasil klasifikasi kemiripan dokumen dengan menggunakan metode *K-Shingling* dan *Cosine Similarity* yang digunakan pada penelitian ini, dan dinyatakan sebagai kelas prediksi (*Predicted Class*). Sehingga, berdasarkan perbandingan label *Actual Class* dan *Predicted Class* dari tiap pasangan data dokumen tugas, maka akan didapatkan nilai pengukuran dari metrik evaluasi sebagai hasil dari penelitian ini.

## IV. HASIL DAN PEMBAHASAN

### A. Hasil Pengumpulan Data

Data yang berhasil dikumpulkan merupakan dokumen jawaban dari pengumpulan tugas mahasiswa yang didapatkan melalui metode penggunaan secara langsung (observatif) pada LMS Undiknas.

Bentuk dataset yang didapat dari LMS Undiknas seperti yang terlihat pada gambar 9 dibawah ini berisi alamat url tempat file dokumen tugas mahasiswa tersimpan pada *Cloud Storage* Undiknas. Dataset tugas mahasiswa tersebut sudah dikelompokkan berdasarkan setiap matakuliahnya. Berikut adalah bentuk dataset yang digunakan dalam penelitian ini.



The image shows a list of document pairs. Each pair consists of a document ID (e.g., '15403') and a corresponding URL (e.g., 'https://cdn.unidkknas.ac.id/bahanajar/24568/fagasa/submission/15403/fagasa-1662019040582000.pdf'). The list continues with similar pairs for IDs 15404 through 16654.

Gambar 9: Bentuk Dataset Dokumen Tugas Mahasiswa

Keseluruhan dataset dokumen tugas mahasiswa tersebut selanjutnya diseleksi agar mendapat dokumen tugas yang hanya berisi jawaban dalam bentuk teks saja. Setelah melalui seleksi data, maka didapatkan data dokumen tugas dari beberapa matakuliah yang bisa digunakan dalam penelitian ini. Berikut adalah tabel jumlah data dokumen tugas mahasiswa yang telah diseleksi :

TABEL 2  
JUMLAH DATA DOKUMEN TUGAS MAHASISWA YANG DIGUNAKAN

Matakuliah	Jumlah Dokumen Tugas
Manajemen Layanan Teknologi Informasi	64
Kecerdasan Buatan	62
Komputer Forensik	65
Sistem Operasi	53
Komunikasi Bisnis	62
Pengantar Ilmu Komunikasi	22
Mikroekonomi	48
<b>Total</b>	<b>376</b>

Kemudian untuk setiap dokumen tugas dari masing - masing matakuliah pada tabel diatas akan dijadikan bentuk pasangan data dimana setiap dokumen dalam satu matakuliah akan dipasangkan dengan dokumen lainnya dalam satu matakuliah yang sama, sehingga apabila terdapat  $n$  jumlah dokumen tugas dalam satu matakuliah, maka akan terdapat  $\frac{n}{2}(n + 1)$  pasang data. Misalkan pada matakuliah Manajemen Layanan Teknologi Informasi terdapat 64 Dokumen Tugas, maka akan terdapat  $\frac{64}{2}(64 + 1) = 2080$  pasang data. Berikut adalah bentuk pasangan data yang dihasilkan :

The image shows a list of document pairs, similar to Gambar 9, with document IDs ranging from 15403 to 16654 and corresponding URLs.

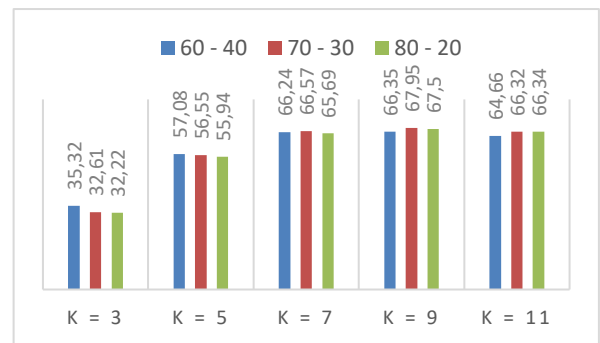
Gambar 10: Bentuk Pasangan Data Uji Dari Dokumen Tugas Mahasiswa Dalam Satu Matakuliah

Pada gambar 10 terlihat bentuk pasangan data yang telah dihasilkan dalam satu matakuliah, tiap pasangan data tersebut juga terlebih dahulu sudah diberikan nilai kemiripan yang akan dinyatakan sebagai label klasifikasi kemiripan dokumen dari *Actual Class*. Kemudian untuk keseluruhan pasangan data dokumen tugas mahasiswa yang dihasilkan dari semua matakuliah akan dipisah menjadi data *training* dan data *testing* sesuai dengan rasio partisi data yang akan diuji coba dalam penelitian ini. Selanjutnya keseluruhan data *training* dan data *testing* tersebut akan diukur nilai metrik evaluasinya berdasarkan perbandingan dari label klasifikasi status kemiripan dokumen pada *Actual Class* dan *Predicted Class*.

B. Hasil Penelitian

1. Hasil Tahap *Training*

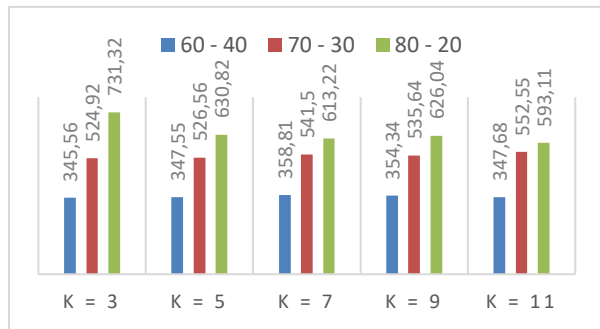
Pada tahap *training* ini dilakukan dengan uji coba pada skenario tanpa *pre-processing* teks (tanpa menggunakan *stopword* dan *stemming*) dan dengan menggunakan *pre-processing* teks, kemudian pada tiap skenario akan dilakukan perbandingan nilai pengukuran metrik evaluasi dan waktu proses untuk beberapa nilai  $K$  yaitu 3, 5, 7, 9, dan 11, dan juga dilakukan perbandingan rasio partisi data (*training* : *testing*) yang akan digunakan yaitu 60 : 40, 70 : 30, dan 80 : 20, sehingga setelah melalui tahapan *training* ini akan dapat diketahui parameter terbaik untuk nilai  $K$ , rasio partisi data dan skenario tanpa *pre-processing* teks atau dengan menggunakan *pre-processing* teks Berikut adalah hasil yang didapatkan dari tahap *training* yang telah dilakukan :



Gambar 11. Grafik Nilai Akurasi Berdasarkan Nilai K dan Rasio Partisi Data pada Skenario Tanpa *Pre-Processing* Teks

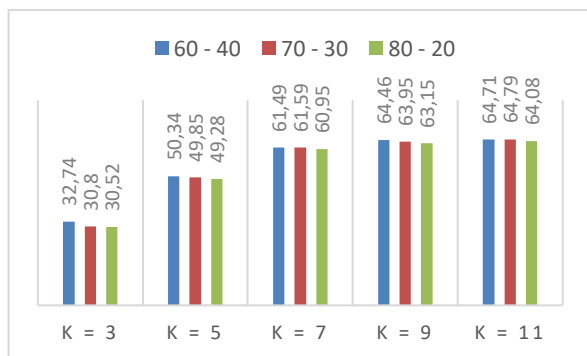


Pada gambar 11 terlihat hasil grafik bahwa pada nilai  $K$  yang semakin besar cenderung terdapat peningkatan nilai akurasi untuk deteksi kemiripan dokumen, dimana nilai akurasi tertinggi didapatkan pada nilai  $K = 9$  dan pada rasio partisi data 70 – 30 yakni dengan nilai akurasi 67,95%.



Gambar 12. Grafik Waktu Proses (detik) Berdasarkan Nilai  $K$  dan Rasio Partisi Data pada Skenario Tanpa *Pre-Processing* Teks

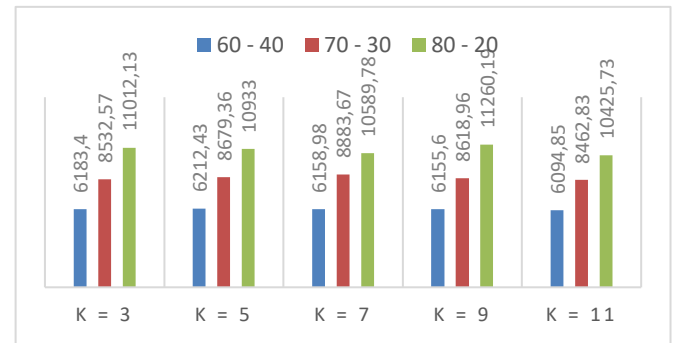
Pada gambar 12 diatas terlihat hasil grafik bahwa waktu proses akan semakin tinggi untuk setiap rasio partisi data nya, ini karena pada tiap rasio partisi data nya terdapat semakin banyak data *training* yang diproses (60%, 70%, dan 80% dari keseluruhan data), oleh karena itu pada rasio partisi data 60 : 40 memiliki waktu proses yang paling kecil, sedangkan rasio partisi data 80 : 20 memiliki waktu proses yang paling besar. Sehingga, untuk mendapatkan pengaruh nilai  $K$  terhadap waktu proses ini akan dihitung nilai rata – rata nya pada tiap rasio partisi data, dan didapatkan rata – rata terkecil untuk waktu proses adalah pada nilai  $K = 11$  dengan rata – rata sebesar 470,395 detik.



Gambar 13. Grafik Nilai Akurasi Berdasarkan Nilai  $K$  dan Rasio Partisi Data pada Skenario Dengan *Pre-Processing* Teks

Pada gambar 13, grafik menunjukkan hasil akurasi dengan menggunakan *pre-processing* juga cenderung terdapat peningkatan nilai akurasi seiring dengan meningkatnya nilai  $K$ , dimana nilai akurasi tertinggi yang didapatkan pada skenario dengan menggunakan *pre-processing* teks adalah pada nilai  $K = 11$  dan rasio partisi data 70 : 30 yakni dengan nilai akurasi 64,79%, tetapi rata – rata nilai akurasi yang dihasilkan pada skenario dengan

*pre-processing* teks ini lebih kecil dibandingkan dengan tanpa *pre-processing* teks.



Gambar 14. Grafik Waktu Proses (detik) Berdasarkan Nilai  $K$  dan Rasio Partisi Data pada Skenario Tanpa *Pre-Processing* Teks

Pada hasil grafik diatas, terlihat bahwa dengan menggunakan *pre-processing* teks akan memakan waktu proses yang sangat tinggi jika dibandingkan dengan tanpa *pre-processing* teks, dimana rata – rata waktu proses yang paling rendah adalah 8260,29 detik yakni pada nilai  $K = 11$ . Hal ini dikarenakan tahapan yang dilakukan lebih banyak jika dibandingkan dengan tanpa *pre-processing* teks, yakni dengan tambahan dilakukannya proses filter *stopword* dan *stemming*. Terlihat juga bahwa semakin besar nilai  $K$  maka waktu prosesnya akan semakin kecil, hal ini dapat disebabkan karena jumlah *shingle* yang terbentuk akan semakin sedikit apabila jumlah kata dalam satu *shingle* yang tersebut berjumlah banyak. Selain itu dalam penelitian ini hasil waktu proses juga dipengaruhi oleh beberapa faktor lain diantaranya adalah karena ukuran file tiap dokumen yang beragam tergantung dari banyaknya teks dalam dokumen, dan juga dapat dipengaruhi oleh kecepatan koneksi internet saat membaca dokumen karena dokumen yang diproses pada penelitian ini tersimpan pada *cloud storage* LMS Undiknas

Setelah dilakukan beberapa skenario pada tahap *training* ini, maka ditentukan bahwa nilai parameter terbaik yang dihasilkan dari tahap *training* ini dilihat berdasarkan nilai akurasi yang didapatkan yakni nilai akurasi tertinggi 67,95% didapat pada skenario tanpa *pre-processing* teks, menggunakan rasio partisi data 70 : 30, dengan nilai  $K = 9$ . Parameter terbaik ini selanjutnya akan digunakan untuk mendapatkan hasil pada tahap *testing*.

## 2. Hasil Tahap *Testing*

Tahap *testing* dilakukan setelah mendapatkan nilai parameter terbaik pada tahap *training*, kemudian hasil pengukuran metrik evaluasi berupa akurasi dari tahap *testing* ini akan dinyatakan sebagai tingkat keberhasilan sistem dalam mendeteksi kemiripan dokumen tugas mahasiswa. Berikut adalah parameter yang digunakan pada tahap *testing* :



TABEL 3

PARAMETER YANG DIGUNAKAN PADA TAHAP TESTING

Skenario	Rasio Partisi Data	Nilai K
Tanpa <i>Pre-Processing</i> Teks	70% ( <i>training</i> ) : 30% ( <i>testing</i> )	$K = 9$

Setelah melalui tahap *testing* yang dilakukan pada 30 % dari keseluruhan data dengan menggunakan parameter terbaik sesuai pada tabel 3 diatas, didapatkan hasil pengukuran sebagai berikut :

TABEL 4

HASIL PENGUKURAN TAHAP TESTING

Akurasi	Waktu Proses
73.55%	102.49 detik

Hasil pengukuran pada tabel 4 menunjukkan nilai akurasi yang didapatkan adalah sebesar 73.55% dengan total waktu proses sebesar 102.49 detik. Nilai akurasi pada tahap *testing* ini menunjukkan tingkat keberhasilan sistem untuk mendeteksi kemiripan dokumen tugas mahasiswa memiliki nilai yang lebih tinggi 8,24% jika dibandingkan dengan nilai akurasi tertinggi pada tahap *training* yaitu 67,95%.

### C. Hasil Implementasi Fitur Deteksi Kemiripan Dokumen Tugas Pada LMS Undiknas

Setelah melalui tahapan *training* dan *testing*, selanjutnya dilakukan implementasi dari fitur deteksi kemiripan dokumen tugas mahasiswa pada LMS Undiknas, fitur ini dapat diakses dan digunakan oleh dosen ketika berada pada halaman menu pengumpulan tugas mahasiswa, berikut adalah tampilan halaman pengumpulan tugas mahasiswa yang dapat diakses oleh dosen :



Gambar 15: Menu Pengumpulan Tugas Mahasiswa

Pada setiap tampilan dari pengumpulan tugas mahasiswa, terdapat sebuah tombol menu yang dapat digunakan oleh dosen untuk mengakses fitur deteksi kemiripan dokumen tugas mahasiswa tersebut. Deteksi kemiripan dokumen tugas dilakukan dengan membandingkan dokumen mahasiswa tersebut dengan dokumen mahasiswa yang lainnya yang juga telah mengumpulkan tugas. Berikut adalah tampilan dari hasil implementasi fitur untuk mendeteksi kemiripan dokumen tugas mahasiswa.



Gambar 16: Halaman Hasil Deteksi Kemiripan Dokumen Tugas Mahasiswa

Pada gambar diatas menunjukkan hasil implementasi dari fitur deteksi kemiripan dokumen tugas mahasiswa pada LMS Undiknas, yang menampilkan informasi berupa nilai persentase kemiripan dokumen tugas dari salah satu mahasiswa dengan dokumen tugas mahasiswa lainnya. Berdasarkan masing – masing nilai persentase kemiripan yang ditampilkan tersebut, maka dosen akan dapat mengetahui apabila ada indikasi plagiarisme yang dilakukan oleh salah satu mahasiswa pada pengumpulan tugas mereka.

## V. KESIMPULAN

Berdasarkan hasil skenario dari tahap *training* yang telah dilakukan, didapatkan kesimpulan bahwa parameter terbaik yang dihasilkan adalah pada skenario tanpa *pre-processing* teks, menggunakan rasio partisi data 70 : 30, dan nilai  $K = 9$  dengan nilai akurasi 69,75%. Sedangkan hasil pengukuran yang dilakukan pada tahap *testing* dengan menggunakan parameter terbaik mendapatkan nilai akurasi 73,55% dengan waktu proses sebesar 102.49 detik, nilai akurasi ini menunjukkan bahwa performa dan keberhasilan sistem untuk mendeteksi kemiripan dokumen tugas mahasiswa memiliki nilai yang lebih tinggi 8.24% jika dibandingkan dengan nilai akurasi tertinggi pada tahap *training*.

Hasil pengembangan fitur deteksi kemiripan dokumen tugas mahasiswa telah berhasil diimplementasikan pada LMS Undiknas dan dapat digunakan oleh dosen pada saat melihat pengumpulan tugas mahasiswa, sehingga dosen dapat mengetahui adanya indikasi plagiarisme yang dilakukan oleh mahasiswa ketika melakukan pengumpulan tugas berdasarkan informasi berupa nilai persentase kemiripan dokumen tugas yang ditampilkan.

## UCAPAN TERIMA KASIH

Artikel jurnal ini ditulis oleh Komang Nova Artawan beserta dosen pembimbing Made Sudarma dan Nyoman Gunantara berdasarkan hasil penelitian Deteksi Kemiripan Dokumen Tugas Mahasiswa pada LMS Menggunakan Metode *K-Shingling* dan *Cosine Similarity* (Studi Kasus : LMS Undiknas) yang digunakan sebagai syarat untuk mendapatkan gelar Magister pada Program Studi Magister Teknik Elektro. Isi sepenuhnya menjadi tanggung jawab penulis.

REFERENSI

- [1] Yauma, Alfath., Fitri, Iskandar., Ningsih, Sari. "Learning Management System (LMS) pada E-Learning Menggunakan Metode Agile dan Waterfall berbasis Website". Jurnal JTik (Jurnal Teknologi Informasi dan Komunikasi), 323-328, 2021.
- [2] Fitriani, Yuni. "Analisa Pemanfaatan Learning Management System (LMS) Sebagai Media Pembelajaran Online Selama Pandemi COVID-19". Journal of Information System, Informatics and Computing. Vol.4 No.2, Desember 2020.
- [3] Putra, Miftakhul Ilmi S. Mutaqin, Imam. "Analisis Deteksi Plagiarisme Pada LMS (Learning Management Systems) Untuk Pembelajaran Online Calon Guru Madrasah Ibtidaiyah". Jpdi: Jurnal Pendidikan Dasar Islam, Vol. 3, No. 1 : 01-15. April 2021.
- [4] Susanto, D., Basuki, A., Duanda, P. "Deteksi Plagiat Dokumen Tugas Daring Laporan Praktikum Mata Kuliah Desain Web Menggunakan Metode Naïve Bayes". Nusantara Journal of Computers and its Applications Volume 2, No. 1, Desember 2016.
- [5] Siswanto, Eric., Giap, Yo Ceng. "Implementasi Algoritma Rabin-Karp dan Cosine Similarity Untuk Pendeteksi Plagiarisme Pada Dokumen". Jurnal Algor-Vol.1 No.2. 2020.
- [6] Salmuasih, Sunyoto, Andi. "Implementasi Algoritma Rabin Karp untuk Pendeteksian Plagiat Dokumen Teks Menggunakan Konsep Similarity". 2013. Seminar Nasional Aplikasi Teknologi Informasi (SNATI) 2013.
- [7] Manaa, Mehdi Ebady., Abdulameer, Ghufra. "Web Documents Similarity Using K-Shingle Tokens and MinHash Technique". Journal of Engineering and Applied Sciences 13 (6): 1449-1505, 2018.
- [8] Firmansyah Fataruba. "Penerapan Metode Cosine Similarity Untuk Pengecekan Kemiripan Jawaban Ujian Siswa". JATI (Jurnal Mahasiswa Teknik Informatika) Vol. 2 No. 2, September 2018
- [9] Rodier, S., Carter, D. "Online Near-Duplicate Detection of News Articles". 2020. Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 1242 – 1249
- [10] Li, Peng., Qiao, Tianling., Guang, Yongxing., Zhang, Lan. "A New Shingling Similar Text Detection Algorithm" in Proceedings of the Second International Symposium on Simulation and Process Modelling, 2020, paper, page 83.
- [11] Sariwating, Vegard, A. "Perancangan dan Implementasi Aplikasi Deteksi Kemiripan Citra Digital Menggunakan Algoritma Shingling dan Redundant Pixel Removal". Skripsi. Universitas Kristen Satya Wacana. 2016
- [12] Simanullang, Irwan Saputra. "Perancangan Aplikasi Deteksi Kemiripan Dokumen Teks Menggunakan Algoritma Shingling". Jurnal Sistem Komputer dan Informatika (JSON) Volume 2, Nomor 1. Hal: 36-41, Sep 2020.
- [13] Leskovec, J., Rajaraman, A., & Ullman, J. D. (n.d.). *Mining of Massive Datasets*.
- [14] Adiansyah, Muhamad Yusuf. "Perancangan dan Implementasi Aplikasi Deteksi Kemiripan Dokumen Menggunakan Algoritma Shingling dan MD5 Fingerprint". S. Kom. Artikel Ilmiah. Universitas Kristen Satya Wacana, Des. 2014.
- [15] Setyadi, I Wayan Adi., Khrisne, Duman Care., Suyadnya, I Made Arsa. "Automatic Text Summarization Menggunakan Metode Graph dan Ant Colony Optimization". Majalah Ilmiah Teknik Elektro, Vol. 17, No. 1, Januari – April 2018.
- [16] Meilina, Lely., Kumara, I Nyoman Satya., Setiawan, Nyoman. "Literature Review Klasifikasi Data Menggunakan Metode Cosine Similarity dan Artificial Neural Network". Majalah Ilmiah Teknologi Elektro, Vol. 20, No. 2, Juli – Desember 2021.
- [17] Firdaus, Pasnur och Wabdillah. "Implementasi Cosine Similarity Untuk Peningkatan Akurasi Pengukuran Kesamaan Dokumen Pada Klasifikasi Dokumen Berita Dengan K Nearest Neighbour". Jurnal Teknologi Informasi dan Komunikasi, Vol. 1, Nomor 1, 2019.
- [18] Ratniasih, N. L., Sudarma, M., & Gunantara, N. (2017). Penerapan Text Mining dalam Spam Filtering untuk Aplikasi Chat. *Majalah Ilmiah Teknologi Elektro*, 16(3).
- [19] Ulgasesa, R., Negara, A. B. P., Tursina. "Pengaruh Stemming Terhadap Performa Klasifikasi Sentimen Masyarakat Tentang Kebijakan New Normal". Jurnal Sistem dan Teknologi Informasi Vol. 10, No. 3, Juli 2022.
- [20] Khan, L. A. R., Ahmed, M. S. I., Almistarihi, H. H. "A Novel Technique Using Multiple K-Shingling Based Weighted Dissimilarity Score for Web Content Outlier Mining". International Journal of Intelligent Engineering and Systems, Vol. 12, No. 4, 2019.
- [21] Agustiawan. (2022). Analisis Similarity/Kemiripan Artikel Jurnal Online Terbitan Tahun 2019-2020 di ISI Yogyakarta. Jurnal Perpustakaan Dan Kearsipan, 2(1), 29–43.



{Halamn ini sengaja dikosongkan}