



BAB II

TINJAUAN PUSTAKA

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Berdasarkan penelitian yang dilakukan oleh Irawan, dkk (2021) mengenai Deteksi Kemiripan Judul Skripsi Menggunakan Algoritma *Levenshtein Distance* pada Kampus Stmik Mic Cikarang, Irawan, dkk menggunakan metode *Levenshtein Distance* untuk mendeteksi kemiripan judul skripsi. Perbedaan dengan penelitian penulis yaitu nama penulis, tahun penelitian dan penelitian penulis untuk kesamaan gambar sedangkan penelitian Irawan, dkk untuk kemiripan judul skripsi.

Berdasarkan penelitian yang dilakukan oleh Julianda (2020) mengenai Analisis Sistem Informasi Pendeteksi Kemiripan Isi Teks Dokumen Sebagai Acuan Penentuan Kelayakan Judul Skripsi Menggunakan Metode *Levenshtein Distance*, Julianda menggunakan metode *Levenshtein Distance* untuk mendeteksi kemiripan isi teks dokumen sebagai acuan penentuan kelayakan judul skripsi. Perbedaan dengan penelitian penulis yaitu nama penulis, tahun penelitian dan penelitian penulis untuk kesamaan gambar sedangkan penelitian Julianda untuk penentuan kelayakan judul skripsi.

Berdasarkan penelitian yang dilakukan oleh Kurniawan, dkk (2018) mengenai Analisis Pencocokan Nama dengan Nama Arab Terjemahan Bahasa Indonesia Menggunakan metode *Levenshtein Distance*, Kurniawan, dkk menggunakan metode RSA untuk Pengamanan Acakan Biss, sedangkan penelitian ini menggunakan metode *Levenshtein Distance* untuk pencocokan nama dengan

nama arab terjemahan bahasa indonesia. Perbedaan dengan penelitian penulis yaitu nama penulis, tahun penelitian dan penelitian penulis untuk kesamaan gambar sedangkan penelitian kurniawan, dkk untuk pencocokan nama dengan nama arab.

Berdasarkan penelitian yang dilakukan oleh Lupitasari (2019) mengenai Aplikasi Pendeteksi Plagiarisme Berdasarkan Isi Teks Dokumen Menggunakan Metode *Levenshtein Distance*, Lupitasari menggunakan metode *Levenshtein Distance* untuk mendeteksi plagiarisme berdasarkan isi teks dokumen. Perbedaan dengan penelitian penulis yaitu nama penulis, tahun penelitian dan penelitian penulis untuk kesamaan gambar sedangkan penelitian Lupitasari untuk pendeteksi plagiarisme isi teks dokumen.

Berdasarkan penelitian yang dilakukan oleh Syahputra, dkk (2021) mengenai Penerapan Metode *Levenshtein Distance* Dalam Mendeteksi Hasil Jawaban Peserta Kelas Daring Dengan Teknik *Web Scraping* Manual, Syahputra, dkk menggunakan metode *Levenshtein Distance* untuk mendeteksi hasil jawaban peserta kelas daring. Perbedaan dengan penelitian penulis yaitu nama penulis, tahun penelitian dan penelitian penulis untuk kesamaan gambar sedangkan penelitian Syahputra, dkk untuk mendeteksi hasil jawaban peserta kelas daring.

II.2. Landasan Teori

Beberapa landasan teori yang dikutip dari beberapa referensi penelitian yang berkaitan dengan penelitian ini yaitu:

II.2.1. Deteksi

Deteksi adalah suatu proses untuk memeriksa atau melakukan pemeriksaan terhadap sesuatu dengan menggunakan cara dan teknik tertentu. Deteksi dapat digunakan untuk berbagai masalah, misalnya dalam sistem pendeteksi sebuah mentimun. Di mana sistem mengidentifikasi masalah-masalah yang berhubungan dengan buah mentimun. Tujuan dari deteksi adalah memecahkan suatu masalah dengan berbagai cara tergantung metode yang diterapkan sehingga menghasilkan sebuah solusi. Deteksi merupakan teknik yang menemukan suatu objek pada citra dengan cara mendeteksi perubahan tingkat kecerahan yang signifikan atau memiliki diskontinuitas. (Berutu, 2020: 457).

Dari Kamus Besar Bahasa Indonesia, deteksi adalah “usaha menemukan dan menentukan keberadaan, anggapan atau kenyataan”. Deteksi merupakan proses untuk memeriksa sesuatu sesuatu yang menggunakan cara dan teknik tertentu.

Menurut artikel, jenis-jenis alat detektor kebakaran terbagi atas alat deteksi asap (*smoke detector*), alat deteksi panas (*heat detector*), alat deteksi api (*flame detector*), dan alat deteksi gas (*gas detector*). Alat-alat tersebut memiliki fungsi untuk mendeteksi. (Waworundeng, 2020: 118).

II.2.2. Gambar

Gambar merupakan media visual dua dimensi di atas bidang yang tidak transparan. Media gambar adalah media yang berfungsi untuk menyampaikan pesan melalui gambar yang menyangkut indera penglihatan. Pesan yang disampaikan dituangkan melalui simbol-simbol komunikasi visual. Media gambar mempunyai tujuan untuk

menarik perhatian, memperjelas materi, mengilustrasikan fakta dan informasi. Media gambar merupakan suatu perantara atau pengantar pesan berbasis visual yang disajikan melalui gambar, simbol-simbol, titik dan garis, untuk memberi gambaran secara konkret dan jelas mengenai suatu materi, gagasan, ide atau peristiwa. (Jayanti dan Fachrurazi, 2020: 333).

Gambar merupakan suatu alat untuk berkomunikasi dalam menyampaikan pesan tertentu dari pembuat gambar, sehingga gambar dapat digunakan menjadi bahasa dalam bidang teknik. Bentuk penyampaian informasi merupakan suatu fungsi yang penting dalam membuat gambar. Oleh karena itu gambar diharapkan dapat menjelaskan tentang data dan keterangan yang tepat dan obyektif. Keterangan yang tertulis dalam gambar yang tidak dapat dinyatakan dengan bahasa lisan, namun harus dibuat dalam bentuk simbol atau lambang tertentu, sehingga untuk menjaga kualitas keterangan dan informasi data yang akan disampaikan akan dipengaruhi kemampuan dari juru gambar (*drafter*). (Dewi, dkk, 2021: 253).

II.2.3. Citra

Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan. Ditinjau dari sudut pandang matematis, citra merupakan fungsi menerus (*continue*) dari intensitas cahaya pada bidang dwimatra. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut.

Pantulan cahaya ini ditangkap oleh alat-alat optik, misalnya mata pada manusia, kamera, pemindai (*scanner*), dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam. (Berutu, 2020: 457).

Sebuah citra dapat diartikan sebagai gambaran secara visual terdiri suatu objek yang tidak mudah untuk diterjemahkan oleh mata dan diinterpretasikan oleh otak, terutama pada objek-objek yang memiliki bentuk atau dimensi yang tidak teratur khususnya apabila objek itu berupa citra keabuan. Berbeda apabila citra tersebut merupakan citra warna, kita dengan mudah dapat menentukan objek-objek yang ada pada citra yang kita amati. Hal ini dikarenakan visualisasi citra warna lebih jelas. Kita dapat menentukan warna-warna yang terkandung didalamnya, begitu juga objek apa yang terkandung di dalam citra warna tersebut. (Sipan, dkk, 2019: 15).

II.2.4. Citra Digital

Citra digital adalah suatu gambaran atau kemiripan dari suatu objek dan dapat diolah oleh komputer. Dalam tinjauan matematis, citra merupakan fungsi kontinu dari intensitas cahaya pada bidang dua dimensi. Citra digital merupakan suatu matriks dimana indeks baris dan kolomnya menyatakan suatu titik pada citra tersebut dan elemen matriksnya (yang disebut sebagai elemen gambar/*pixel*/ piksel/*pels*/ *picture element*) menyatakan tingkat keabuan pada titik tersebut. (Purba, dkk, 2019: 300).

II.2.5. Metode *Levensthein Distance*

Salah satu langkah paling populer di *name matching* yang jumlah minimum operasi untuk mengkonversi nama. Operasi dasarnya yaitu:

1. Penyisipan karakter (misalnya Yeltsin → Yexltsin)
2. Penghapusan karakter (misalnya Yeltsin → Yeltn)
3. Penggantian dengan karakter yang lain (misalnya Yeltsin → Yqltsin)

Algoritma *levenshtein distance* memperbandingkan dua kata yang kemudian diukur berdasarkan berdasarkan kata masing-masing yang kemudian dihitung dengan menggunakan matrik yang dibangun dengan baris dan kolom dapat digambarkan dengan n dan m yang berisi kata per karakter masing-masing. Memakai fungsi yang mengembalikan nilai 0 jika karakter yang dicocokkan sama dan sebaliknya memberikan nilai 1 jika karakter yang dicocokkan tidak sama. Jarak Edit dikenal adalah Jarak *levenshtein distance* (setelah matematika Rusia Vladimir Levenshtein). Dan perhitungan dasarnya yaitu: $\text{Dist}(S1,$

$$S2) = 1.0 - \frac{\text{dist}(S1,S2)}{\max(|S1|,|S2|)} \dots\dots\dots(1)$$

$\max(|S1|,|S2|)$ Keterangan:

S1: Jarak Awal *Levensthein Distance*

S2: Jarak Akhir *Levensthein Distance*

|S1|: Determinan Dari Jarak Awal

|S2|: Determinan Dari Jarak Akhir

Di mana $\text{dist}(S1, S2)$ adalah yang sebenarnya jarak *levenshtein distance* yang mengandung nilai 0 jika string yang satu jumlahnya positif jika keduanya tidak

sama. Ukuran nilai di antara 0.0 dan 1.0 ketika ada nilai yang lebih besar akan semakin banyak yang didapat di antara dua string yang sama. (Kurniawan, dkk,

2018: 3).

Contoh:

Berikut dua string yang akan digunakan sebagai contoh:

BARU

BATU

Langkah 1:

Operasi Penyisipan Karakter (*Insertion*) Operasi penyisipan karakter berarti menyisipkan karakter ke dalam suatu *string*:

		B	A	R	U
	0	1	2	3	4
B	1				
A	2				
T	3				
U	4				

Langkah 2:

Operasi Penghapusan Karakter (*Deletion*) yang tidak sama:

- Kalimat BATU dan BARU memiliki jumlah karakter yang sama, jadi tidak ada penghapusan karakter

Langkah 3:

Operasi Penukaran Karakter (*Subtitution*) Operasi penukaran karakter

merupakan operasi menukar sebuah karakter dengan karakter lain:

		B	A	R	U
	0	1	2	3	4
B	1	0			
A	2				
T	3				
U	4				

Lakukan langkah tersebut sampai akhirnya mendapatkan hasil.

		B	A	R	U
	0	1	2	3	4
B	1	0	1	2	3
A	2	1	0	1	2
T	3	2	1	1	2
U	4	3	2	2	0

Dari sini bisa kelihatan perbedaan antara kedua kata:

x: BARU y:

BATU

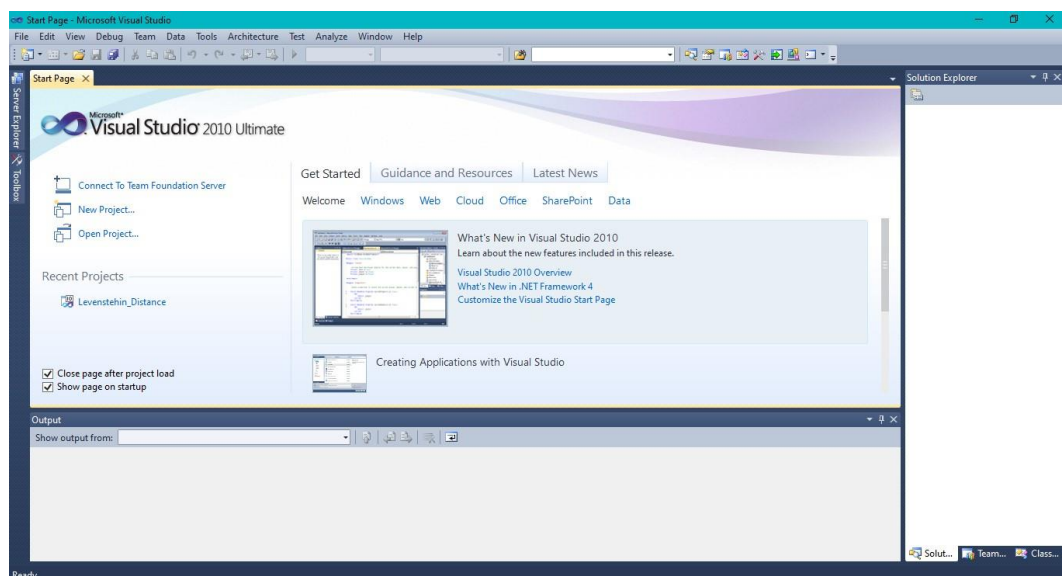
Hasil perbedaan: 25%

Hasil kemiripan: 75%

Jumlah kata yang tidak mirip adalah **1 huruf**.

II.2.6. *Visual Basic 2010*

Visual basic 2010 adalah inkarnasi dari bahasa *Visual Basic* yang sangat populer dan telah dilengkapi dengan fitur serta fungsi yang setara dengan bahasa tingkat tinggi lainnya seperti C++. *Visual Basic* dapat digunakan untuk membuat aplikasi *Windows*, *mobile*, *web*, dan *office* yang kompleks dengan menggunakan kode yang anda tulis, atau kode yang telah ditulis oleh orang lain dan kemudian dimasukkan ke dalam program. (Putri dan Putra, 2018: 52).



Gambar II.1. Visual Basic 2010
(Sumber: Putri dan Putra, 2018: 52)

II.2.7. *Unified Modeling Language (UML)*

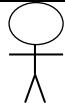
Unified Modeling Language (UML) adalah sebuah bahasa untuk menentukan, visualisasi, kontruksi, dan mendokumentasikan *artifact* (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat

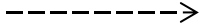
lunak. *Artifact* dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya.

1. Use Case Diagram

Use Case Diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut adalah simbol-simbol yang ada pada diagram *use case*. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.1.

Tabel II.1. Simbol Use Case

Notasi	Keterangan	Simbol
<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .	
<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).	

<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).	
<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.	
<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.	
<i>Association</i>	Apa yang mnghubungkan antara objek satu dengan objek lainnya.	
<i>System</i>	Menspesifikan paket yang menampilkan sistem secara terbatas.	
<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .	
<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya.	

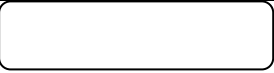
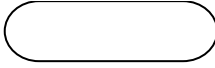
<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi	
-------------	---	---

(Sumber: Pitrawati dan Sanjaya, 2021: 157)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut adalah simbol-simbol yang ada pada diagram aktivitas. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.2.

Tabel II.2. Simbol *Activity Diagram*

Notasi	Keterangan	Simbol
<i>Activity</i>	Memperlihatkan bagaimana masingmasing kelas antarmuka saling berinteraksi satu sama lain.	
<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.	
<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.	

<i>Activity Final</i>	Bagaimana objek dibentuk dan dihancurkan	
<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran	

(Sumber: Pitrawati dan Sanjaya, 2021: 157)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.3.

Tabel II.3. Simbol *Sequence Diagram*

Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang terjadi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang

		terjadi.
--	--	----------

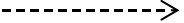
(Sumber: Pitrawati dan Sanjaya, 2021: 157)

4. *Class Diagram* (Diagram Kelas)

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelaskelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Berikut adalah simbol-simbol yang ada pada diagram Kelas. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

Tabel II.4. *Class Diagram*

Notasi	Keterangan	Gambar
<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).	
<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.	
<i>Class</i>		

	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.	
<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.	
<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.	
<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang	
	bergantung padanya elemen yang tidak mandiri	
<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya	

(Sumber: Pitrawati dan Sanjaya, 2021: 157)