

**PERBANDINGAN ALGORITMA *JARO-WINKLER DISTANCE* DAN  
*LEVENSHTTEIN DISTANCE* PADA SISTEM PENGOREKSI EJAAN  
KATA BERBAHASA INDONESIA**

*Diajukan Untuk Menyelesaikan  
Pendidikan Program Sarjana Sastra-1 Pada  
Jurusan Teknik Informatika Fakultas Ilmu Komputer UNSRI*



Oleh :

Achma Desvania  
09021181520009

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER UNIVERSITAS SRIWIJAYA  
2020**

## LEMBAR PENGESAHAN SKRIPSI

### PERBANDINGAN ALGORITMA *JARO-WINKLER DISTANCE* DAN *LEVENSHTEIN DISTANCE* PADA SISTEM PENGOREKSI EJAAN KATA BERBAHASA INDONESIA

Oleh :

Achma Desvania  
NIM : 09021181520009

Indralaya, Agustus 2020

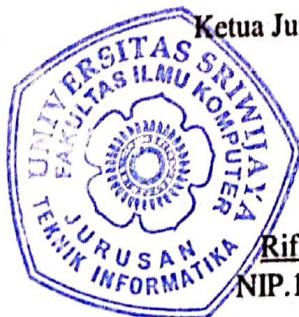
Pembimbing I,

Pembimbing II,

Novi Yulistiani, S.Kom., M.T.  
NIP. 198211082012122001

Desty Bodiah, M.T.  
NIP. 1671016112890005

Mengetahui,  
Ketua Jurusan Teknik Informatika



Rifkie Primartha, M.T.  
NIP.197706012009121004

## TANDA LULUS UJIAN SIDANG SKRIPSI

Pada hari Kamis tanggal 23 Juli 2020 telah dilaksanakan ujian sidang skripsi oleh Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya.

Nama : Achma Desvania  
NIM : 09021181520009  
Judul : Perbandingan Algoritma Jaro-Winkler Distance dan Levenshtein Distance Pada Sistem Pengoreksian Ejaan Kata Berbahasa Indonesia

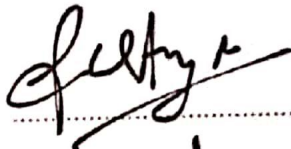
1. Pembimbing I

Novi Yusliani, S.Kom., M.T.  
NIP. 198211082012122001



2. Pembimbing II

Desty Rodiah, M.T.  
NIP. 1671016112890005



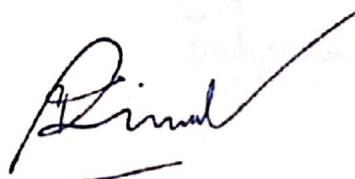
3. Penguji I

Dr. Abdiansah, S.Kom., M.CS.  
NIP. 198410012009121005



4. Penguji II

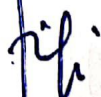
Mastura Diana Marieska, M.T.  
NIP. 198603212018032001



Mengetahui,  
Ketua Jurusan Teknik Informatika



Rifkie Primartha S.T. M.T.  
NIP 197706012009121004



## HALAMAN PERNYATAAN

Yang bertanda tangan di bawah ini :

Nama : Achma Desvania

NIM : 09021181520009

Program Studi : Teknik Informatika

Judul : Perbandingan Algoritma *Jaro-Winkler Distance* dan *Levenshtein Distance* Pada Sistem Pengoreksi Ejaan Kata Berbahasa Indonesia

Hasil Pengecekan Software *iThenticate/Turnitin* : 10%

Menyatakan bahwa Laporan Projek saya merupakan hasil karya sendiri dan bukan hasil penjiplakan atau plagiat. Apabila ditemukan unsur penjiplakan atau plagiat dalam tugas akhir ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya sesuai dengan ketentuan yang berlaku.

Demikian, pernyataan ini saya buat dengan sebenarnya dan tidak ada paksaan oleh siapapun.

Indralaya, Agustus 2020



Achma Desvania

NIM. 09021181520009

## MOTTO DAN PERSEMBAHAN

“There are two types of people who will tell you that you cannot make a difference in this world: those who are afraid to try and those who are afraid you will succeed.” —Ray

Goforth

“I am me. You are you. I’m just living my life, you should just live your own life, so take back those obscure preaching and do what you are supposed to do.” —Kim Hee Chul

“So indeed, along with hardship, there is ease. Indeed, along with hardship, there is ease. So when you have finished [your duties], then stand up [for worship]. And to your Lord turn [your] attention.” — QS: Al-Insyirah, 5-8

“If you are feeling blue because you are the last person to touch the finish line, it’s okay. Being slow is not a sin tho, there’s always a flower in every single path.” — Achma

Desvania

Kupersembahkan karya tulis ini kepada :

- Allah SWT
- Kedua Orang Tua yang kusayangi
- Adik-adikku yang terbaik
- Sahabat dan teman sejahwatku
- Idolaku yang memotivasi
- Almamaterku

PERBANDINGAN ALGORITMA *JARO-WINKLER DISTANCE* DAN *LEVENSHTEIN DISTANCE* PADA SISTEM PENGOREKSI EJAAN KATA BERBAHASA INDONESIA

Oleh :  
Achma Desvania  
09021181520009

ABSTRAK

Ejaan merupakan salah satu bagian terpenting dalam sebuah teks tulisan. Kerap dijumpai kesalahan ejaan pada kalimat yang memengaruhi makna yang terkandung dan mengakibatkan pembaca kesulitan dalam mengartikannya. Pengoreksian ejaan yang dilakukan secara manual akan memakan waktu yang lama dikarenakan proses pemeriksaan dilakukan berulang-ulang agar memperoleh hasil yang akurat. Dibutuhkan sebuah sistem dengan tujuan mempermudah proses pengoreksian ejaan kata yang salah. Beberapa algoritma dapat digunakan untuk membantu kerja sistem yang hendak dibangun. Pada penelitian ini, algoritma yang digunakan adalah algoritma *Levenshtein Distance* dan *Jaro-Winkler Distance* dimana performansi keduanya akan dibandingkan dalam sistem pengoreksi kata berejaan salah. Perbandingan menggunakan nilai *Mean Reciprocal Rank* dan dibantu oleh metode *N-Gram* untuk membagi kata per kata ke bentuk *Bigram*, pembobotan menggunakan *tf-idf* dan mengecek kemiripan antara dua buah kata dengan *Cosine Similarity*. Hasil yang diperoleh dari penelitian ini menunjukkan bahwa performansi algoritma *Levenshtein Distance* untuk jenis kesalahan *insertion* lebih unggul dibanding *Jaro-Winkler Distance*, kemudian performansi *Jaro-Winkler Distance* sedikit lebih unggul untuk jenis kesalahan *substitution* dan *transposition*, sedangkan untuk jenis kesalahan *deletion* performansi keduanyaimbang.

**Kata kunci:** *Levenshtein Distance, Jaro-Winkler Distance, Spelling Checker, N-Gram, tf-idf, Cosine Similarity, Mean Reciprocal Rank.*

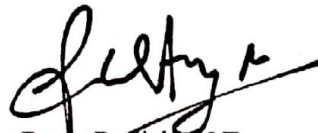
Indralaya, Agustus 2020

Pembimbing I,



Novy Yuskani, S.Kom., M.T.  
NIP. 198211082012122001

Pembimbing II,



Desly Rodiah, M.T.  
NIP. 1671016112890005

Mengetahui,  
Ketua Jurusan Teknik Informatika



Rifkie Primartha, M.T.  
NIP.197706012009121004

# COMPARISON OF JARO-WINKLER DISTANCE AND LEVENSHTTEIN DISTANCE ALGORITHM IN INDONESIAN WORD SPELLING CORRECTION SYSTEM

By :  
Achma Desvania  
09021181520009


## ABSTRACT

Spelling is one of the most noteworthy parts of a written text. Spelling errors often encountered in sentences that affect the meaning contained and make the reader have difficulty interpreting them. Correction of spelling done manually will take a long time because the checking process repeated to obtain accurate results. A system is needed to simplify the process of correcting the spelling of wrong words. We can use several algorithms to help the system work to build. In this study, the author will compare the Levenshtein Distance and Jaro-Winkler Distance algorithm to see the best performance of the two in a spelling checker system for incorrect words. The comparison will be using the Mean Reciprocal Rank value and assisted by the N-Gram method to divide word by word into Bigram form, weighting using tf-idf, and checking the similarity between two words with Cosine Similarity. The results obtained from this study indicate that the performance of the Levenshtein Distance algorithm for the insertion error type is better compared to Jaro-Winkler Distance. Jaro-Winkler Distance performance is slightly better for the substitution and transposition error type, and for deletion, the performance of both are balanced.

**Keywords:** Levenshtein Distance, Jaro-Winkler Distance, Spelling Checker, N-Gram, tf-idf, Cosine Similarity, Mean Reciprocal Rank.

Indralaya, Agustus 2020

Supervisor I,



Novi Yuskumi, S.Kom., M.T.  
NIP. 198211082012122001

Supervisor II,



Desty Rodiah, M.T.  
NIP. 1671016112890005

Approved,  
Chairman of The Informatics Engineering Departements,



Rifkie Primartha, M.T.  
NIP.197706012009121004

## KATA PENGANTAR



Alhamdulillah, puji dan syukur kehadiran Allah SWT atas segala nikmat, rahmat, dan karunia-Nya, sehingga penulis dapat merampungkan tugas akhir dengan baik. Tugas akhir ini disusun guna memenuhi salah satu syarat dalam menyelesaikan pendidikan tingkat sarjana (S1) pada Program Studi Teknik Informatika Fakultas Ilmu Komputer Program di Universitas Sriwijaya.

Penulis hendak mengucapkan banyak terima kasih kepada beberapa pihak yang telah membantu penulis dalam bentuk bimbingan, doa, dan motivasi dalam perjalanan penyusunan tugas akhir ini, yaitu :

1. Bapak Achfas Decca dan Ibu Nyayu Sallamah selaku orang tua penulis tercinta, Mardiana Meisie Safitri dan Siti Amanda Sebrina selaku saudara penulis beserta seluruh keluarga besar yang tak henti memberi dukungan berupa kasih sayang, didikan, nasihat, motivasi dan doa;
2. Bapak Jaidan Jauhari, S.Pd., M.T. selaku Dekan Fakultas Ilmu Komputer Universitas Sriwijaya;
3. Bapak Rifkie Primartha, M.T. selaku ketua Jurusan Teknik Informatika Universitas Sriwijaya;
4. Bapak Samsuryadi, M.Kom., Ph.D. selaku pembimbing akademik yang memberi nasihat dan bimbingan kepada penulis selama masa perkuliahan;
5. Ibu Novi Yusliani, M.T. dan Ibu Desty Rodiah, M.T. selaku pembimbing tugas akhir yang memberi bimbingan, nasihat, kritik, dan ilmu pengetahuan dalam proses penyelesaian tugas akhir ini;



6. Bapak Dr. Abdiansah, S.Kom., M.Cs. dan Ibu Mastura Diana Marieska, M.T. selaku penguji yang memberi nasihat, kritik, saran dan bimbingan guna menyelesaikan tugas akhir ini;
7. Seluruh Bapak dan Ibu Dosen Pengajar Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya atas ilmu dan pelajaran yang diberikan kepada penulis selama kegiatan akademik;
8. Seluruh jajaran staf Program Studi Ilmu Komputer Universitas Sriwijaya yang telah membantu dalam kelancaran proses administrasi dan akademik selama masa perkuliahan;
9. Sahabat seperjuangan IF Reguler A 2015 yang bahu-membahu dalam memberi motivasi dan dukungan sejak awal perkuliahan hingga tugas akhir ini selesai;
10. Sahabat baikku, Kintan Semi Melinda, yang selalu bersama penulis sejak awal perkuliahan hingga tugas akhir untuk memberi motivasi, dukungan, nasihat serta ikatan persahabatan yang tak pernah putus;
11. Kak Malian Zikri, kak Nur Hamidah, kak Sarafina dan kakak-kakak tingkat lain yang telah merelakan waktu juga tenaga untuk membantu penulis dalam mengajari dan memberi arahan dalam pembuatan tugas akhir;
12. Ari Susanto, Billy Dipta Yulio, Ewaldo Haryoseno, Farhan Evrizal, Rizka Nabillah, Rizqi Septian Dwinanda, Sabatia, Annisa Agustin, Afrilia, Anggi Juwita, Imas Syaibun Nisa, Melani Puspa Sari, Oktaria Permata Sari, Nadiya Fahada, Ricko Prayudha, Nedi Novriansyah dan Raden Roro

Ayu Laraswati, sahabat penulis yang selalu membantu penulis sejak awal masa perkuliahan hingga menyusun tugas akhir ini;

13. Saudara KAMSUJUPAY yaitu Fathin Sabrina, Nurma Junita, Pamela Young, Fany Khairani, Maulida Melvina Putri dan Kak Jungsoo yang selalu memberi kasih sayang, motivasi, dukungan, nasihat, bimbingan dan doa kepada penulis;

14. Idola penulis, Super Junior, yang tanpa sadar memberi penulis motivasi dan menghibur penulis selama penyelesaian tugas akhir;

15. Seluruh pihak yang turut membantu dan memberikan semangat kepada penulis dalam penulisan tugas akhir ini.

Penulis mengetahui bahwa tugas akhir ini belum mendekati kata sempurna. Dengan kata lain, penulis mengharapkan kritik maupun saran yang membangun guna menyempurnakan tugas akhir ini di masa yang akan datang.

Inderalaya, Agustus 2020

Penulis

## DAFTAR ISI

	<b>Halaman</b>
HALAMAN JUDUL .....	i
HALAMAN LEMBAR PENGESAHAN .....	ii
HALAMAN PERSETUJUAN KOMISI PENGUJI.....	ivi
HALAMAN PERNYATAAN .....	iv
HALAMAN MOTTO DAN PERSEMBAHAN .....	v
ABSTRACT .....	xi
ABSTRAK .....	vvi
KATA PENGANTAR.....	viii
DAFTAR ISI .....	xi
DAFTAR TABEL .....	xv
DAFTAR GAMBAR.....	xvi
DAFTAR LAMPIRAN .....	xvi

### **BAB I PENDAHULUAN**

1.1	Pendahuluan.....	I1
1.2	Latar Belakang Masalah.....	I1
1.3	Rumusan Masalah .....	I3
1.4	Tujuan Penelitian .....	I4
1.5	Manfaat Penelitian .....	I4
1.6	Batasan Masalah .....	I5
1.7	Sistematika Penulisan.....	I5
1.8	Kesimpulan .....	I7

### **BAB II KAJIAN LITERATUR**

2.1	Pendahuluan.....	II-1
2.2	Landasan Teori.....	II-1

2.2.1	<i>Spelling Checker</i> .....	II-1
2.2.2	<i>Preprocessing</i> .....	II-3
2.2.3	<i>Levenshtein Distance</i> .....	II-4
2.2.4	<i>Jaro-Winkler Distance</i> .....	II-6
2.2.5	<i>N-Gram</i> .....	II-9
2.2.6	<i>Term Frequency-Inverse Document Frequency (tf-idf)</i> .....	II-9
2.2.7	<i>Cosine Similarity</i> .....	II-10
2.2.8	<i>Mean Reciprocal Rank (MRR)</i> .....	II-11
2.2.9	<i>Rational Unified Process (RUP)</i> .....	II-12
2.3	Penelitian Relevan .....	II-14
2.4	Kesimpulan .....	II-15

### **BAB III METODOLOGI PENELITIAN**

3.1	Pendahuluan .....	III-1
3.2	Metode Pengumpulan Data .....	III-1
3.3	Tahapan Penelitian .....	III-2
3.3.1.	Menetapkan Kerangka Kerja/Framework .....	III-2
3.3.2.	Menetapkan Kriteria Pengujian .....	III-4
3.3.3.	Menetapkan Format Data Pengujian .....	III-5
3.3.4.	Menentukan Alat yang Digunakan dalam Pelaksanaan Penelitian....	III-6
3.3.5.	Melakukan Pengujian Penelitian .....	III-7
3.3.6.	Analisis Hasil Pengujian dan Membuat Kesimpulan Penelitian .....	III-7
3.4	Metode Pengembangan Perangkat Lunak .....	III-8
3.4.1.	Fase Insepsi .....	III-8
3.4.2.	Fase Elaborasi .....	III-8
3.4.3.	Fase Konstruksi .....	III-9
3.4.4.	Fase Transisi .....	III-9
3.5	Manajemen Proyek Penelitian .....	III-10

3.6	Kesimpulan .....	III-22
-----	------------------	--------

## **BAB IV PENGEMBANGAN PERANGKAT LUNAK**

4.1	Pendahuluan.....	IV-1
4.2	Fase Insepsi.....	IV-1
4.2.1	Pemodelan Bisnis.....	IV-1
4.2.2	Kebutuhan Sistem .....	IV-3
4.2.3	Analisis dan Desain.....	IV-4
4.2.3.1	Analisis Kebutuhan Perangkat Lunak.....	IV-4
4.2.3.2	Analisis Data .....	IV-5
4.2.3.3	Analisis <i>Preprocessing</i> .....	IV-6
4.2.3.4	Analisis Identifikasi Kata.....	IV-8
4.2.3.5	Analisis Pencarian Kandidat Kata dengan <i>Levenshtein Distance</i> ..	IV-8
4.2.3.6	Analisis Pencarian Kandidat Kata dengan <i>Jaro-Winkler</i> .....	IV-9
4.2.3.7	Analisis Pemecahan Kandidat Kata menggunakan <i>Bigram</i> .....	IV-10
4.2.3.8	Analisis Pembobotan Kandidat Kata menggunakan <i>tf-idf</i> .....	IV-11
4.2.3.9	Analisis Pemingkatan Kandidat Kata menggunakan <i>Cosine Similarity</i> .....	IV-13
4.2.3.10	Desain Perangkat Lunak.....	IV-15
4.3	Fase Elaborasi .....	IV-19
4.3.1	Pemodelan Bisnis.....	IV-20
4.3.2	Perancangan Data.....	IV-20
4.3.3	Perancangan Antarmuka .....	IV-20
4.3.4	Kebutuhan Sistem .....	IV-21
4.3.5	Diagram Aktivitas .....	IV-22
4.3.6	Diagram Alur ( <i>Sequence Diagram</i> ).....	IV-24
4.4	Fase Konstruksi.....	IV-26
4.4.1	Kebutuhan Sistem .....	IV-26
4.4.2	Diagram Kelas .....	IV-26

4.4.3	Implementasi.....	IV-28
4.4.2.1	Implementasi Kelas .....	IV-28
4.4.2.2	Implementasi Antarmuka .....	IV-31
4.5	Fase Transisi .....	IV-32
4.5.1	Pemodelan Bisnis.....	IV-32
4.5.2	Kebutuhan Sistem .....	IV-32
4.5.3	Rencana Pengujian.....	IV-33
4.5.3.1	Rencana Pengujian <i>Use Case</i> Melakukan Muat Data .....	IV-33
4.5.4	Implementasi.....	IV-34
4.5.4.1	Pengujian <i>Use Case</i> Melakukan Muat Data .....	IV-36
4.5.4.2	Pengujian <i>Use Case</i> Melakukan Proses Pengoreksian Data.....	IV-37
4.6	Kesimpulan .....	IV-38

## **BAB V HASIL DAN ANALISIS PENELITIAN**

5.1	Pendahuluan.....	V-1
5.2	Data Hasil Percobaan / Penelitian.....	V-1
5.2.1	Konfigurasi Percobaan .....	V-1
5.2.2	Data Hasil Konfigurasi.....	V-3
5.3	Analisis Hasil Penelitian .....	V-12
5.4	Kesimpulan .....	V-15

## **BAB VI KESIMPULAN DAN SARAN**

6.1	Kesimpulan .....	VI-1
6.2	Saran .....	VI-2

<b>DAFTAR PUSTAKA</b> .....	xix
-----------------------------	-----

## DAFTAR TABEL

	<b>Halaman</b>
Tabel II-1. Contoh Proses Bigram .....	II-9
Tabel II-2. Contoh Perhitungan <i>Mean Reciprocal Rank</i> .....	II-12
Tabel III-1. Rancangan Tabel Hasil Pengujian .....	III-5
Tabel III-2. Rancangan Tabel Keseluruhan Hasil Pengujian Sistem.....	III-6
Tabel III-3. Perencanaan <i>Work Breakdown Structure</i> (WBS).....	III-11
Tabel IV-1. Kebutuhan Fungsional.....	IV-3
Tabel IV-2. Kebutuhan Non-fungsional.....	IV-3
Tabel IV-3. Definisi Aktor.....	IV-17
Tabel IV-4. Definisi Use Case .....	IV-17
Tabel IV-5. Skenario Use Case Memuat Data .....	IV-18
Tabel IV-6. Skenario Use Case Proses Pengecekan Data .....	IV-18
Tabel IV-7. Rancangan Data.....	IV-20
Tabel IV-8. Implementasi Kelas .....	IV-29
Tabel IV-9. Rencana Pengujian <i>Use Case</i> Memuat Data.....	IV-33
Tabel IV-10. Rencana Pengujian <i>Use Case</i> Proses Pengoreksian Data .....	IV-33
Tabel IV-11. Pengujian <i>Use Case</i> Proses Memuat Data .....	IV-36
Tabel IV-12. Pengujian <i>Use Case</i> Melakukan Proses Pengecekan Data .....	IV-36
Tabel V-1. Hasil Algoritma <i>Levenshtein Distance</i> dan <i>Jaro-Winkler Distance</i> .....	V-3
Tabel V-2. Hasil Pengurutan Kandidat Kata Dengan <i>Cosine Similarity</i> .....	V-7
Tabel V-3. Hasil <i>Mean Reciprocal Rank</i> .....	V-12

## DAFTAR GAMBAR

	<b>Halaman</b>
Gambar II-1. Perbaikan Kalimat dari Jenis Kesalahan Penghapusan Huruf.....	II-2
Gambar II-2. Perbaikan Kalimat dari Jenis Kesalahan Penambahan Huruf .....	II-2
Gambar II-3. Perbaikan Kalimat dari Jenis Kesalahan Penggantian Huruf .....	II-3
Gambar II-4. Perbaikan Kalimat dari Jenis Kesalahan Penukaran Posisi Huruf.....	II-3
Gambar II-5. Contoh Proses <i>Case Folding</i> .....	II-4
Gambar II-6. Contoh Proses <i>Tokenizing</i> .....	II-4
Gambar II-7. <i>Pseudocode</i> algoritma <i>Levenshtein Distance</i> .....	II-6
Gambar II-8. Contoh Perhitungan Algoritma <i>Jaro-Winkler Distance</i> .....	II-8
Gambar II-9. Arsitektur <i>Rational Unified Process</i> .....	II-13
Gambar III-1. Diagram Kerangka Kerja Sistem.....	II-2
Gambar III-2. Penjadwalan Penelitian Tahap Menentukan Ruang Lingkup dan Unit Penelitian.....	III-15
Gambar III-3. Penjadwalan Penelitian Tahap Menentukan Dasar Teori yang Berkaitan dengan Penelitian .....	III-16
Gambar III-4. Penjadwalan Penelitian Tahap Menentukan Kriteria Pengujian ....	III-16
Gambar III-5. Penjadwalan Penelitian Tahap Menentukan Alat yang Digunakan untuk Pelaksanaan Penelitian pada Fase Inception .....	III-17
Gambar III-6. Penjadwalan Penelitian Tahap Menentukan Alat yang Digunakan untuk Pelaksanaan Penelitian pada Fase Elaboration.....	III-18
Gambar III-8. Penjadwalan Penelitian Tahap Menentukan Alat yang Digunakan untuk Pelaksanaan Penelitian pada Fase Transition.....	III-19
Gambar III-9. Penjadwalan Penelitian Tahap Melakukan Pengujian Penelitian...	III-19
Gambar III-10. Penjadwalan Penelitian Tahap Melakukan Analisa Hasil Pengujian dan Membuat Kesimpulan .....	III-19
Gambar IV-1. Diagram Proses <i>Preprocessing</i> .....	IV-6
Gambar IV-2. Contoh Teks Berejaan Salah.....	IV-7
Gambar IV-3. Hasil Proses <i>Case Folding</i> .....	IV-7
Gambar IV-4. Hasil Proses <i>Tokenizing</i> .....	IV-7



## **DAFTAR LAMPIRAN**

1. Dokumen yang Digunakan
2. Kode Program

# **BAB I**

## **PENDAHULUAN**

### **1.1 Pendahuluan**

Bab ini menjabarkan keseluruhan penelitian secara umum, meliputi latar belakang masalah, rumusan masalah, tujuan dan manfaat masalah, batasan masalah, sistematika penulisan, dan kesimpulan.

Dalam pendahuluan ini pun terdapat penjabaran mengenai algoritma *Levenshtein Distance* dan *Jaro-Winkler Distance* yang akan diterapkan pada sistem pengecekan kata berejaan salah.

### **1.2 Latar Belakang Masalah**

Ejaan merupakan salah satu bagian terpenting dalam sebuah teks tulisan. Kerap dijumpai kesalahan ejaan pada kalimat yang memengaruhi makna yang terkandung dan mengakibatkan pembaca kesulitan dalam mengartikannya. Kesalahan ejaan tersebut dapat dipicu oleh beberapa faktor seperti keteledoran yang dilakukan oleh pengguna maupun kata yang dimasukkan ke dalam tulisan belum terdaftar ke dalam kamus basis data (Braddley, Fachrurrozi, & Novi, 2017).

Pengoreksian ejaan yang dilakukan secara manual akan memakan waktu yang lama dikarenakan proses pemeriksaan dilakukan berulang-ulang agar memperoleh hasil yang akurat. Dibutuhkan sebuah sistem yang bertujuan memudahkan proses pengoreksian ejaan yang dikenal dengan *spelling checker*. Sistem ini bekerja dengan memeriksa kata berejaan salah dan memberi saran dalam bentuk kandidat kata yang mendekati benar (Soleh & Purwarianti, 2011).

Salah satu contoh pencocokan *string* yang sering digunakan dalam beberapa kasus untuk algoritma tertentu adalah *Approximate String Matching*. Pencocokan ini digunakan untuk menemukan perkiraan *string* yang cocok dengan pola sesuai dengan kata yang terdapat dalam basis data. Metode ini dapat dipakai untuk mencari kata tidak baku karena mampu mengidentifikasi *string* yang sama dan memiliki kemiripan penulisan (Kusumaningrum & Rochmawati, 2016).

Penelitian yang dilakukan oleh (Yulianto, Arifudin, & Alamsyah, 2018) menggunakan metode *Approximate String Matching*, menyatakan bahwa algoritma *Levenshtein Distance* yang diterapkan pada dokumen Bahasa Indonesia berejaan salah menghasilkan sistem pengoreksi ejaan yang bisa mendeteksi kesalahan dengan tiga operasi pencocokan *string* yakni penghapusan, penambahan dan penggantian dan juga memberi sugesti pada setiap kategori penyebab kesalahan ejaan. Tingkat akurasi yang didapat dari penelitian sebesar 86%.

Tak hanya *Levenshtein Distance*, beberapa algoritma lain juga digunakan untuk menghasilkan sistem pengecekan ejaan kata yang kurang tepat. (Rochmawati & Kusumaningrum, 2015) melakukan sebuah penelitian yang

membandingkan empat algoritma pencarian *string* menggunakan pendekatan *Approximate String Matching* yaitu *Levenshtein Distance*, *Hamming Distance*, *Damerau Levenshtein Distance* dan *Jaro-Winkler Distance*. Hasil dari evaluasi yang dilakukan menggunakan *user relevance judgement*. Hasil penelitian menggunakan keempat algoritma pencarian *string* tersebut menunjukkan bahwa algoritma *Jaro-Winkler Distance* memiliki nilai kinerja tertinggi dalam melakukan pengecekan kata dengan nilai *Mean Average Precision* (MAP) sebesar 0,87.

Berdasarkan uraian diatas, penelitian kali ini akan membandingkan algoritma *Levenshtein Distance* dengan algoritma *Jaro-Winkler Distance* untuk pengecekan kata berejaan salah dalam Bahasa Indonesia dengan menggunakan nilai *Mean Reciprocal Rank* (MRR) sebagai indikator.

### 1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan, maka rumusan masalah dari penelitian ini adalah bagaimana perbandingan dari kinerja algoritma *Levenshtein Distance* dan algoritma *Jaro-Winkler Distance* dalam memperbaiki kata salah ejaan. Untuk menyelesaikan permasalahan tersebut, penelitian ini dibagi dalam beberapa pertanyaan penelitian (*Research Question*), yaitu :

1. Bagaimana mekanisme pengembangan perangkat lunak dengan menggunakan algoritma *Levenshtein Distance* dan *Jaro-Winkler Distance* dalam mengecek kesalahan dalam ejaan kata?

2. Bagaimana perbandingan nilai *Mean Reciprocal Rank* (MRR) dari algoritma *Levenshtein Distance* dan algoritma *Jaro-Winkler Distance* untuk mengecek kesalahan dalam ejaan kata?

#### 1.4 Tujuan Penelitian

Tujuan penelitian ini adalah sebagai berikut ;

1. Mengembangkan perangkat lunak yang dapat membantu dalam pengecekan kesalahan ejaan kata berbahasa Indonesia dengan menggunakan algoritma *Levenshtein Distance* dan algoritma *Jaro-Winkler Distance*.
2. Mengetahui perbandingan dari nilai *Mean Reciprocal Rank* (MRR) dari algoritma *Levenshtein Distance* dan algoritma *Jaro-Winkler Distance* untuk mengecek kesalahan dalam ejaan kata.

#### 1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Sistem yang dibangun dapat membantu dalam memperbaiki kata yang ejaannya salah.
2. Hasil penelitian dapat digunakan sebagai acuan penelitian selanjutnya dan dikembangkan untuk melakukan perbandingan antara algoritma *Levenshtein Distance* dan algoritma *Jaro-Winkler Distance* untuk mengecek kesalahan dalam ejaan kata.

## 1.6 Batasan Masalah

Batasan masalah dari penelitian adalah :

1. Sistem yang dibangun hanya dapat mengecek kata dalam bahasa Indonesia.
2. Kandidat kata yang disarankan sebanyak 5 kata per algoritma dengan angka probabilitas yang paling tinggi.
3. Kamus yang digunakan adalah Kamus Besar Bahasa Indonesia (KBBI).
4. Metode yang digunakan untuk sistem yang dibangun antara lain. memecahkan kata dengan metode *Bigram*, pembobotan karakter menggunakan metode *tf-idf* dan penentuan ranking kandidat kata menggunakan *Cosine Similiarity*.
5. Sistem yang dibangun hanya dapat mengecek kesalahan kata; bukan singkatan, lambang, nama ataupun kata khusus lainnya.

## 1.7 Sistematika Penulisan

Sistematika penulisan dari penelitian ini adalah sebagai berikut :

### **BAB I PENDAHULUAN**

Bab ini menjabarkan lebih jelas mengenai latar belakang, perumusan masalah, tujuan dan manfaat penelitian, batasan masalah, metodologi penelitian, dan sistematika penulisan.

### **BAB II KAJIAN LITERATUR**

Bab ini membahas dasar-dasar teori dari algoritma dan metode yang digunakan dalam penelitian, objek penelitian serta tinjauan literatur dari penelitian yang akan dilakukan.

### **BAB III METODOLOGI PENELITIAN**

Bab ini menjelaskan mengenai tahapan peneliti dimana tiap rencana tahapan penelitian dipaparkan secara rinci dengan mengacu pada sebuah kerangka kerja. Di bagian akhir bab akan memuat perancangan manajemen proyek untuk pelaksanaan penelitian.

### **BAB IV PENGEMBANGAN PERANGKAT LUNAK**

Bab ini mengulas proses pengembangan dari perangkat lunak yang dipakai sebagai sarana pengujian dimana penulis membangun perangkat lunak dengan berlandaskan pemograman yang berorientasi objek dengan panduan *Rational Unified Process*. Ada empat fase yang termasuk dalam orientasi objek *Rational Unified Process* yaitu fase inepsi, elaborasi, konstruksi, dan trasisi dimana setiap fase memiliki pemodelan bisnis, analisis dan desain, kebutuhan, implementasi serta pengujian fase.

### **BAB V HASIL DAN ANALISIS PENELITIAN**

Bab ini menjabarkan hasil pengujian perangkat lunak berupa daftar kata berejaan salah dan kandidat yang dihasilkan serta analisis perbandingan

antara kedua algoritma yang dibandingkan menggunakan *Mean Reciprocal Rank* (MRR) sebagai indikator.

## **BAB VI KESIMPULAN DAN SARAN**

Bab ini berisi beberapa kesimpulan dan saran dari penelitian yang dilakukan.

### **1.8 Kesimpulan**

Penelitian ini berfokus pada perbandingan antara algoritma *Levenshtein Distance* dan *Jaro-Winkler Distance* untuk melakukan pengecekan pada kata dalam Bahasa Indonesia yang berejaan salah. Penjelasan mengenai algoritma *Levenshtein Distance* dan algoritma *Jaro-Winkler Distance* serta teori terkait dan penelitian yang relevan akan dijabarkan lebih jelas di bagian bab selanjutnya.



## DAFTAR PUSTAKA

- Braddley, M. O., Fachrurrozi, M., & Novi, Y. (2017). Pengoreksian Ejaan Kata Berbahasa Indonesia Menggunakan Algoritma Levenshtein Distance. *Prosiding Annual Research Seminar*, 3(1), 167–171.
- Christanti, V., Rudy, & Naga, D. S. (2018). Fast and accurate spelling correction using trie and Damerau-levenshtein distance bigram. *Telkomnika (Telecommunication Computing Electronics and Control)*, 16(2), 827–833. <https://doi.org/10.12928/TELKOMNIKA.v16i2.6890>
- Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3), 171–176. <https://doi.org/10.1145/363958.363994>
- Dwitiyastuti, R. N., Muttaqin, A., & Aswin, M. (2013). *Pengoreksi Kesalahan Ejaan Bahasa Indonesia Menggunakan Metode Levenshtein Distance*. 1–6.
- Gan, J. Q. (2015). ORIGINAL RESEARCH A query suggestion method combining TF-IDF and Jaccard Coefficient for interactive web search, 4(2), 119–125. <https://doi.org/10.5430/air.v4n2p119>
- Fahma, A. I., Cholissodin, I., & Perdana, R. S. (2018). Identifikasi Kesalahan Penulisan Kata ( Typographical Error ) pada Dokumen Berbahasa Indonesia Menggunakan Metode N-gram dan Levenshtein Distance. 2(1), 53–62.
- Hamidah, N. (2019). *Spelling Checker Menggunakan Algoritma Damerau Levenshtein Distance dan Cosine Similarity*.
- Kruchten, P. (2003). *The Rational Unified Process An Introduction*. <https://doi.org/10.1109/ICSE.2002.146346>
- Kurniawati, A., Puspitodjati, S., & Rahman, S. (2010). Implementasi Algoritma Jaro-Winkler Distance untuk Membandingkan Kesamaan Dokumen Berbahasa Indonesia. *Proceeding, Seminar Ilmiah Nasional Komputer Dan Sistem Intelijen KOMMIT 2008, Depok, Indonesia*.
- Kusumaningrum, R., & Rochmawati, Y. (2016). *Studi Perbandingan Algoritma Pencarian String dalam Metode Approximate Studi Perbandingan Algoritma Pencarian String dalam Metode Approximate String Matching untuk Identifikasi Kesalahan Pengetikan Teks*. May. <https://doi.org/10.24002/jbi.v7i2.491>
- Lahitani, A. R., Permasari, A. E., & Setiawan, N. A. (2016). *Cosine Similarity to Determine Similarity Measure : Study Case in Online Essay Assessment*.
- Lawaye, A. A., & Purkayastha, B. S. (2016). Design and Implementation of Spell Checker for Kashmiri. *International Journal of Scientific Research*, 5(7).
- Maghfira, T., Cholissodin, I., & Widodo, A. (2017). Deteksi Kesalahan Ejaan dan Penentuan Rekomendasi Koreksi Kata yang Tepat Pada Dokumen Jurnal JTIK Menggunakan Dictionary Lookup dan Damerau-Levenshtein Distance. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 1(6), 498–506.
- Nejja, M., & Yousfi, A. (2015). The Context in Automatic Spell Correction. *Procedia Computer Science*, 73(April), 109–114. <https://doi.org/10.1016/j.procs.2015.12.055>

- Novantara, P., & Pasruli, O. (2018). Implementasi Algoritma Jaro-Winkler Distance Untuk Sistem Pendeteksi Plagiarisme Pada Dokumen Skripsi. *Buffer Informatika*, 3(1). <https://doi.org/10.25134/buffer.v3i2.960>
- Prasanti, A. A., Fauzi, M. A., & Furqon, M. T. (2018). Klasifikasi Teks Pengaduan Pada Sambat Online Menggunakan Metode N- Gram dan Neighbor Weighted K-Nearest Neighbor ( NW-KNN ). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIHK) Universitas Brawijaya*, 2(2), 594–601.
- Rasyid, A., Kamayani, M., Reinanda, R., Simbolon, S., Soleh, M. Y., & Purwarianti, A. (2011). Application of document spelling checker for Bahasa Indonesia. *ICACISIS 2011 - 2011 International Conference on Advanced Computer Science and Information Systems, Proceedings, January*, 249–252.
- Rochmawati, Y., & Kusumaningrum, R. (2015). *Studi Perbandingan Algoritma Pencarian String dalam Metode Approximate String Matching untuk Identifikasi Kesalahan Pengetikan Teks*. 125–134.
- Soleh, M. Y., & Purwarianti, A. (2011). A non word error spell checker for Indonesian using morphologically analyzer and HMM. *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics, ICEEI 2011, July*. <https://doi.org/10.1109/ICEEI.2011.6021514>
- Yulianto, M. M., Arifudin, R., & Alamsyah, A. (2018). Autocomplete and Spell Checking Levenshtein Distance Algorithm To Getting Text Suggest Error Data Searching In Library. *Scientific Journal of Informatics*, 5(1), 75. <https://doi.org/10.15294/sji.v5i1.14148>