

Analisis Kinerja Algoritma Damerau Levenshtein Distance dan Peter Norvig dalam Koreksi Ejaan Bahasa Indonesia

Mohammad Hafiz Ar Rafi¹, Iqbal Al-Ayyubi², Muhammad Reza Fahlefi³,
Anggraini Puspita Sari^{*4}

^{1,2,3,4}Informatika, Ilmu Komputer, UPN Veteran Jawa Timur

E-mail: ¹22081010031@student.upnjatim.ac.id, ²22081010217@student.upnjatim.ac.id,

³22081010223@student.upnjatim.ac.id, ^{*4}anggraini.puspita.if@upnjatim.ac.id

Abstrak

Damerau Levenshtein Distance (DLD) dan metode Peter Norvig (PN) merupakan dua buah jenis algoritma yang digunakan untuk melakukan koreksi ejaan. Kemudahan penerapan dari kedua algoritma ini menjadikan banyak penelitian sebelumnya menggunakannya untuk koreksi ejaan kata. Untuk itu, penelitian ini bertujuan untuk menganalisis dan membandingkan kinerja dari kedua algoritma tersebut. Analisis ini hanya mengukur besar akurasi dan waktu komputasi yang diperlukan untuk melakukan koreksi ejaan. Dari pengujian 500 kata yang salah dan sekali pengeditan, algoritma DLD memiliki akurasi 64.4% dengan waktu komputasi 67 detik. Sedangkan, metode PN memiliki akurasi 84% dengan waktu komputasi 0.21 detik. Untuk kata dengan dua kali pengeditan, keduanya mengalami penurunan. Namun, metode PN tetap unggul lebih unggul dibandingkan DLD baik dari akurasi maupun waktu komputasi.

Kata Kunci: Damerau Levenshtein Distance, Peter Norvig, Koreksi Ejaan, Teks Bahasa Indonesia, Analisis Algoritma.

Abstract

Damerau Levenshtein Distance (DLD) and the Peter Norvig (PN) method are two types of algorithms used to perform spelling correction. The ease of application of these two algorithms has made many previous studies use them for word spelling correction. Therefore, this research aims to analyze and compare the performance of the two algorithms. This analysis only measures the accuracy and computation time required to perform spelling correction. From testing 500 wrong words and one edit, the DLD algorithm has an accuracy of 64.4% with a computation time of 67 seconds. Meanwhile, the PN method has an accuracy of 84% with a computation time of 0.21 seconds. For words with two edits, both have decreased. However, the PN method remains superior to DLD in both accuracy and computation time.

Keywords: Damerau Levenshtein Distance, Peter Norvig, spelling correction, Indonesian text, Algorithm Analysis.

1. PENDAHULUAN

Bahasa Indonesia sangat berkaitan erat dengan masyarakat karena bahasa ini memprioritaskan sikap cinta tanah air dan menghormati tanah air, serta tidak ada perbedaan dalam berkomunikasi. Bahasa memberikan fasilitas komunikasi untuk semua aktivitas kita. Sebagai makhluk sosial, manusia tidak bisa terhindar dari komunikasi. Komunikasi adalah suatu tindakan yang dilakukan dengan tujuan memahami maksud dari informasi yang diberikan oleh orang lain. Sebagai warga negara Indonesia, sebaiknya kita menggunakan bahasa Indonesia yang benar untuk mewujudkan persatuan dan kesatuan bangsa. Bahasa Indonesia menjadi salah satu identitas dan menjadi sebuah perantara dalam

berkomunikasi [1]. Karena menulis merupakan cara komunikasi tidak langsung, maka agar komunikasi dapat terjalin dengan baik, setiap individu harus memperhatikan setiap kata yang ditulisnya.

Beberapa tahun terakhir ini, perkembangan teknologi digital tumbuh semakin cepat dan secara langsung berdampak pada peningkatan pemanfaatan teknologi yang terus meningkat. Perkembangan teknologi dapat memberikan dampak positif dan negatif pada masyarakat. Dalam dunia pendidikan, internet mempunyai manfaat dalam mendukung proses belajar dan mengajar serta sebagai sarana untuk mendapatkan informasi [2]. Dalam menulis dokumen, penulisan kata menjadi hal penting untuk diperhatikan agar makna dalam sebuah kata dapat tersampaikan dengan jelas [3]. Untuk itu, penulisan kata yang baik harus didasarkan pada Kamus Besar Bahasa Indonesia (KBBI). Namun, penggunaan KBBI secara langsung tidak begitu efektif karena penulis harus melakukan pengecekan kata per kata pada dokumen. Kesalahan penulisan yang umum terjadi dalam menulis dokumen adalah terkait dengan ejaan kata yang salah. Pengetikan kata yang salah bisa terjadi karena satu sampai dua karakter dalam sebuah kata tertukar, ditambah, terhapus, atau terganti [4]. Kesalahan ini sering kali disebabkan oleh kurangnya ketelitian dalam mengetik. Oleh karena itu, untuk meningkatkan kualitas dokumen, koreksi ejaan berperan penting dalam memproses kata-kata yang mengalami kesalahan ejaan [5].

Untuk mengatasi kesalahan ejaan, algoritma Damerau-Levenshtein Distance (DLD) dan metode Peter Norvig (PN) merupakan dua algoritma yang sering digunakan untuk mengoreksi kesalahan ejaan. Kedua algoritma ini terkenal karena mudah diterapkan dan tidak memerlukan data pelatihan yang rumit. Algoritma DLD mampu menangani berbagai jenis kesalahan ejaan umum seperti penyisipan, penghapusan, substitusi, dan transposisi, sehingga memberikan saran koreksi yang akurat untuk berbagai jenis kesalahan. Keunggulan ini membuat DLD sangat andal dalam aplikasi yang memerlukan fleksibilitas tinggi. Namun, algoritma ini memiliki kelemahan dalam hal komputasi yang intensif dan memori yang diperlukan terutama untuk kamus yang besar [6][7].

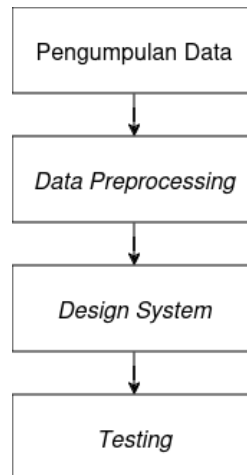
Di sisi lain, metode Peter Norvig menggunakan model probabilistik berdasarkan frekuensi urutan karakter dalam korpus besar. Hal ini menjadikannya efisien dan mudah diimplementasikan. Metode ini tidak memerlukan pelatihan awal pada dataset, memungkinkan penerapan langsung setelah kamus disiapkan. Namun, efektivitas koreksi dalam metode ini sangat bergantung pada kelengkapan kamus yang digunakan, dan algoritma ini mungkin mengalami kesulitan dengan kesalahan berbasis konteks karena terutama berfokus pada pencocokan string dan analisis frekuensi daripada pemahaman semantik [6][8].

Penelitian ini bertujuan untuk menganalisis kinerja kedua algoritma tersebut berdasarkan waktu komputasi dan akurasi dalam mengoreksi ejaan. Dengan demikian, pemilihan algoritma yang tepat dapat lebih dipertimbangkan sehingga sistem atau aplikasi yang dibangun dapat menjadi lebih efektif dan efisien. Kajian mengenai kinerja algoritma DLD menunjukkan bahwa algoritma ini memberikan kinerja yang baik dalam berbagai skenario, namun memerlukan efisiensi komputasi yang tinggi [6]. Sementara itu, analisis terhadap metode Peter Norvig menunjukkan bahwa pendekatan ini efektif dalam aplikasi dunia nyata, namun tetap memerlukan kamus yang komprehensif untuk hasil yang optimal [8].

Dengan adanya harapan untuk melanjutkan pendidikan ke jenjang yang lebih tinggi, kebutuhan akan penulisan yang baik dan benar menjadi semakin krusial. Oleh karena itu, mengadopsi algoritma koreksi ejaan yang efektif dapat memberikan kontribusi nyata dalam memastikan kualitas dokumen yang dihasilkan, baik dalam konteks akademik maupun profesional. Selain itu, penggunaan bahasa Indonesia yang baik dan benar juga memperkuat identitas nasional dan memperlancar komunikasi antar warga negara, mendukung terciptanya persatuan dan kesatuan bangsa. Komunikasi yang efektif dapat meningkatkan motivasi, memungkinkan informasi yang disampaikan diterima dengan baik, sehingga pada gilirannya dapat meningkatkan kinerja [9]. Hal ini menunjukkan bahwa penguasaan bahasa Indonesia yang baik tidak hanya penting dalam konteks pendidikan, tetapi juga dalam kehidupan sehari-hari sebagai bagian dari masyarakat Indonesia.

2. METODE PENELITIAN

Penelitian ini bertujuan untuk menganalisis kinerja dua algoritma koreksi ejaan, yaitu DLD dan PN. Analisis dilakukan berdasarkan dua parameter utama: waktu komputasi dan akurasi ejaan. Metode penelitian ini melibatkan beberapa tahapan penting mulai dari pengumpulan data, preprocessing data, perancangan sistem, hingga pengujian. Setiap tahapan dilakukan secara sistematis untuk memastikan validitas dan reliabilitas hasil penelitian. Berikut adalah penjelasan rinci dari setiap tahapan dalam metode penelitian ini.



Gambar 1. Diagram Alir Metode Penelitian

2.1. Pengumpulan Data

Data yang digunakan dalam penelitian ini berasal dari beberapa sumber utama. Repository Sastrawi di GitHub menyediakan kata dasar sebanyak 29.932 entri [10]. Sementara itu, GitHub Nolimitid memberikan kata berimbuhan dalam jumlah 758 kata [11]. Selain itu, data dari Wikipedia dengan tema sejarah, khususnya mengenai Abad Pertengahan, diambil sebanyak 3.483 kata [12]. Namun, data dari Wikipedia memerlukan proses preprocessing untuk memastikan keakuratannya dan sesuai dengan kebutuhan program yang digunakan dalam penelitian ini.

2.2. Data Preprocessing

Pra-proses data, atau yang dikenal sebagai "Data preprocessing", merupakan serangkaian langkah untuk memproses atau membersihkan data sebelum data benar-benar digunakan, seperti mengidentifikasi dan mengisi nilai yang tidak diperlukan [13]. Untuk kata yang berasal dari wikipedia, perlu dilakukan Word Preprocessing terlebih dahulu. Pemrosesan kata pada tahap ini adalah dengan case folding atau mengubah huruf kapital menjadi huruf kecil dan menghapus karakter yang tidak dibutuhkan yaitu karakter yang tidak termasuk ke dalam alfabet [14]. Selain itu, setiap kata yang masuk kedalam kamus akan diperiksa maknanya dengan menggunakan website KBBI. Apabila kata tersebut tidak terdaftar dalam KBBI maka kata akan dihapus dalam kamus kata (Gambar 1). Dari proses pembersihan data, banyak kata yang berasal dari Wikipedia sejumlah 3.382 kata, sehingga jumlah data yang masuk kamus sebagai data uji sebanyak 34.074 kata.



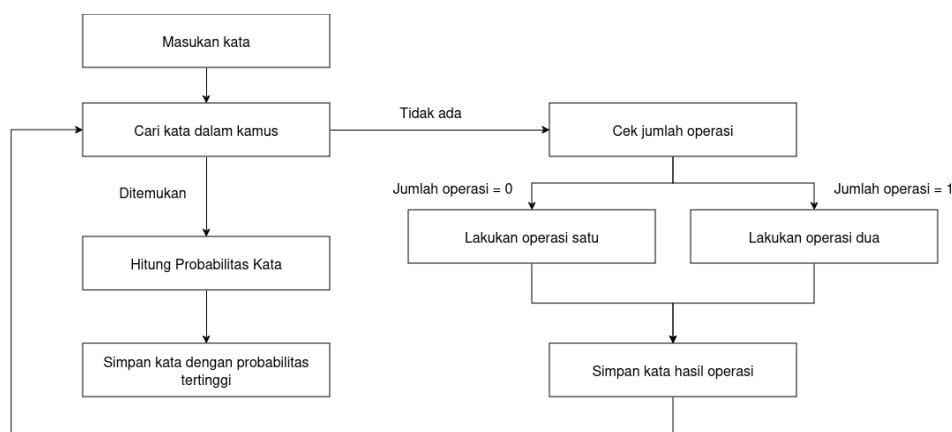
Gambar 2. Diagram Alir preprocessing data

2.3. Design System

Penerapan algoritma baik itu DLD, maupun metode PN menggunakan bahasa pemrograman python. Python dipilih karena memiliki banyak pustaka yang dapat digunakan untuk memudahkan pemrosesan dan pengolahan data [15]. Adapun beberapa pustaka yang digunakan untuk memudahkan pembuatan program diantaranya,

- A. Regex, Regular Expression (regex) merupakan sebuah cara untuk mencari kata berdasarkan pola yang ditulis dengan notasi tertentu [16]. Dalam kasus ini, regex berguna untuk mengambil kata-kata dalam kamus dan mengubahnya kedalam bentuk larik.
- B. Collections, pustaka ini menyediakan beberapa fungsi salah satunya yaitu counter. Counter merupakan fungsi untuk menghitung frekuensi sebuah kata dalam suatu larik.

Algoritma PN bekerja dengan cara merubah kata salah dengan kata yang baru dengan menambah, menghapus, mengubah, dan menukar sebuah karakter. Dengan cara ini, sebuah kata yang salah dapat dikoreksi menjadi kata yang ada di kamus [17]. Setiap operasinya akan menghasilkan himpunan kata baru sesuai dari jumlah karakter kata tersebut. Namun, jika operasi dilakukan hanya sekali, maka perbaikan hanya terdeteksi satu kali operasi. Untuk itu, penelitian ini menggunakan dua kali operasi pengeditan sehingga batas maksimal perbaikan sebanyak dua kali operasi. Selain itu, algoritma ini juga menerapkan probabilitas pada setiap kandidatnya. Jadi, setiap kata akan dihitung frekuensinya dibagi dengan banyaknya kata dalam kamus tersebut. Kata yang memiliki nilai probabilitas tertinggi yang akan menjadi koreksi dari perbaikan kata yang salah. Secara umum, alur dari algoritma ini dapat dilihat pada gambar 3.



Gambar 3. Diagram Alir Metode PN

Algoritma yang digunakan selanjutnya yaitu DLD. Dalam algoritma ini, jumlah jarak setiap kata digunakan untuk menentukan seberapa mirip kedua kata tersebut. Levenshtein Distance adalah metode untuk menghitung jumlah operasi yang diperlukan untuk mengubah satu kata menjadi kata lainnya. Metode ini mencakup operasi seperti penghapusan, penambahan, dan substitusi karakter, sehingga sangat cocok untuk koreksi ejaan. Variasi dari Levenshtein Distance adalah Damerau-Levenshtein Distance, yang menambahkan operasi transposisi, yaitu pertukaran dua huruf yang berdekatan, untuk menangani kesalahan ejaan yang umum terjadi. Berdasarkan penelitian sebelumnya, rumus untuk Levenshtein Distance dapat dilihat pada persamaan (1), (2), dan (3), sementara rumus untuk DLD dijelaskan pada persamaan (4) [3].

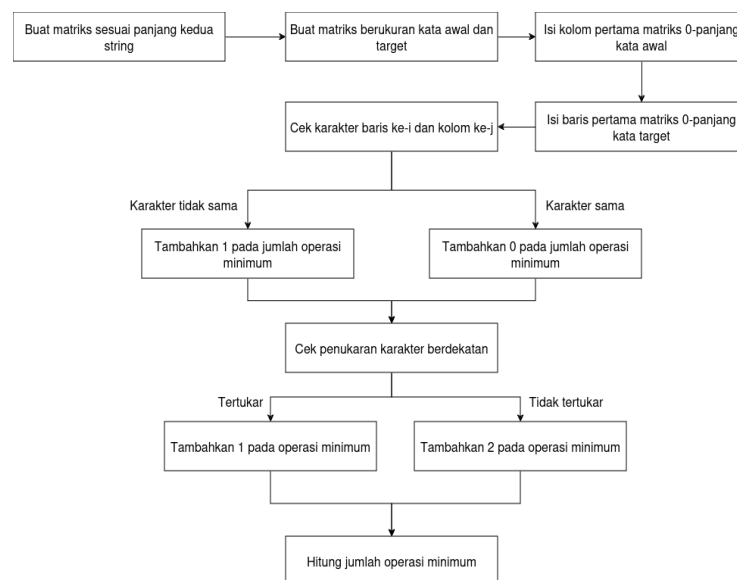
$$d_{i0} = \sum_{k=1}^i w_{del}(b_k) \quad , \quad for \ 1 \leq i \leq m \quad (1)$$

$$d_{0j} = \sum_{k=1}^j w_{ins}(a_k) \quad , \quad for \ 1 \leq j \leq n \quad (2)$$

$$d_{ij} = \begin{cases} d_{i-1,j-1} & for \ a_j = b_i \\ \min \begin{cases} d_{i-1,j} + w_{del}(b_i) \\ d_{i,j-1} + w_{ins}(a_j) \\ d_{i-1,j-1} + w_{sub}(a_j, b_i) \end{cases} & for \ a_j \neq b_i \end{cases} \quad (3)$$

$$d_{ij} = \begin{cases} d_{i-1,j-1} & for \ a_j = b_i \\ \min \begin{cases} d_{i-1,j} + w_{del}(b_i) \\ d_{i,j-1} + w_{ins}(a_j) \\ d_{i-1,j-1} + w_{sub}(a_j, b_i) \\ d_{i-2,j-2} + w_{tra} \end{cases} & if \ i,j > 1, a_i = b_{j-1}, a_{i-1} = b_j, for \ a_j \neq b_i \\ \min \begin{cases} d_{i-1,j} + w_{del}(b_i) \\ d_{i,j-1} + w_{ins}(a_j) \\ d_{i-1,j-1} + w_{sub}(a_j, b_i) \end{cases} & for \ 1 \leq i \leq m, 1 \leq j \leq n, for \ a_j \neq b_i \end{cases} \quad (4)$$

Lebih jelasnya, Alur pada gambar 4 merepresentasikan proses dan tahapan penghitungan jarak pada algoritma DLD. Pada gambar 4 proses perulangan terjadi dua kali sehingga setiap karakter pada kedua kata tersebut dapat dibandingkan.



Gambar 4. Diagram Alir Algoritma DLD

2.4. Testing

Untuk melakukan testing, akan dibagi menjadi 2 kelompok operasi dibedakan dengan jumlah karakter yang akan dikoreksi oleh program. untuk Operasi 1 akan melakukan Koreksi dengan total 1

karakter dan untuk Operasi 2 akan melakukan koreksi dengan total 2 karakter di setiap tipe Sub-Operasi nya.

Tabel 1. Operasi 1

Tipe Operasi	Contoh Penerapan		Keterangan
	Kata Awal	Kata koreksi	
Penghapusan	selamma	selama	menghapus “m”
Penambahan	istimwa	istimewa	menambahkan “e”
Penggantian	awalnya	awalnya	mengganti “i” dengan “y”
Pertukaran	menjaid	menjadi	menukar posisi “i” dengan “d”

Tabel 2. Operasi 2

Tipe Operasi	Contoh Penerapan		Keterangan
	Kata Awal	Kata koreksi	
Penghapusan	mempeirbanyyak	memperbanyak	menghapus “i” dan “y”
Penambahan	mhasiwa	mahasiswa	menambahkan “a” dan “s”
Penggantian	dilaramk	dilarang	mengganti “m” dengan “n” dan “k” dengan “g”
Pertukaran	nuasntaar	nusantara	menukar posisi “a” dengan “s” dan “a” dengan “r”

Kedua algoritma akan diuji untuk melakukan koreksi ejaan kata yang salah seperti pada tabel 1 dan 2. Banyaknya kata yang digunakan dalam pengujian ini yaitu sebesar 500 kata secara bertahap dengan detail pada tabel 3 dan 4. Kumpulan kata yang salah tersebut akan disimpan ke dalam larik. Selanjutnya, tiap-tiap kata akan dikoreksi menggunakan algoritma DLD dan metode PN. Hasil koreksi kata tersebut akan menjadi nilai akurasi dari kedua algoritma tersebut. Selain itu, waktu komputasi juga akan dihitung untuk mengetahui seberapa besar efisiensi kedua algoritma tersebut.

Tabel 3. Pengujian

No. Pengujian	Jumlah Kata Uji
Pengujian 1	125
Pengujian 2	250
Pengujian 3	375
Pengujian 4	500

3. HASIL DAN PEMBAHASAN

Dalam penelitian ini, dua algoritma koreksi ejaan yaitu DLD dan metode PN diuji untuk mengoreksi kata-kata yang salah. Pengujian dilakukan dengan mengukur akurasi dan waktu komputasi dari kedua algoritma tersebut. Berikut ini adalah analisis dari hasil pengujian berdasarkan data yang ditampilkan dalam Tabel 4 dan 5. dan contoh pengujian yang ada di (Gambar 5)

Enakm kesatriab beramni melawzan kejarans jmusuh, membuktikqn kesetiawan dwan keberankian msereka dzalam memlpertahankan negra. Raj emerintah degan tngan bei, mengunang pertkaian atar bangsawn tang merasz fertekan oley pajal berlehihan. Melebarnyq pengqruh kekhasaan taja terhaadp esluruh wilayahnya snagat dibatnu loeh sistme militre ayng kuta.

enam kesatria berani melawan kejaran musuh, membuktikkan kesetiaan dan keberanian mereka dalam mempertahankan negara. raja memerintah dengan tangan baik, mengundang pertikaian antar bangsawan tang merasa tertekan oleh pajak berlebihan. melebarnya pengaruh kekuasaan taja terhadap seluruh wilayahnya sangat dibantu oleh sistem militer yang

Gambar 5. Contoh Pengujian

Tabel 4. Hasil pengujian operasi 1

Pengujian	Jumlah Kata	Jumlah Koreksi Benar		Akurasi (%)		Waktu (s)	
		DLD	PN	DLD	PN	DLD	PN
1	125	90	113	72	90.4	67.674	0.216
2	250	183	226	73.2	90.4	133.802	0.157
3	375	252	330	67.2	88.	196.148	1.272
4	500	231	423	64.2	84.6	233.316	2.925

Tabel 5. Hasil pengujian operasi 2

Pengujian	Jumlah Kata	Jumlah Koreksi Benar		Akurasi (%)		Waktu (s)	
		DLD	PN	DLD	PN	DLD	PN
1	125	5	86	4	68.8	88.279	8.421
2	250	15	170	6	68	163.873	12.795
3	375	17	241	4.5	64.2	243.328	15.469
4	500	20	306	6	61.2	303.298	20.324

3.1. Analisis Akurasi

Berdasarkan hasil pengujian, terlihat bahwa metode PN memiliki akurasi yang lebih tinggi dibandingkan dengan DLD pada hampir setiap pengujian. Pada pengujian pertama (Tabel 4) dengan jumlah kata sebanyak 125, metode PN mampu mengoreksi 113 kata dengan akurasi 90.4%, sedangkan DLD hanya mengoreksi 90 kata dengan akurasi 72%. Hal ini menunjukkan bahwa metode PN lebih efektif dalam mengoreksi kata-kata yang salah.

Namun, seiring dengan meningkatnya jumlah kata yang diuji, akurasi metode PN cenderung menurun. Pada pengujian keempat (Tabel 4) dengan jumlah kata sebanyak 500, akurasi metode PN turun menjadi 84.6%, sedangkan akurasi DLD turun menjadi 64.2%. Penurunan akurasi ini mungkin disebabkan oleh kompleksitas dan jumlah kata yang semakin banyak sehingga mempengaruhi performa algoritma dalam melakukan koreksi ejaan.

Selain mengalami penurunan saat kata yang diuji semakin banyak, keduanya juga mengalami penurunan pada pengujian operasi 2 (Tabel 5). Salah satu contohnya yaitu pada pengujian 1 (Tabel 4) metode PN yang memiliki tingkat akurasi sebesar 90.4% turun menjadi 68.8% pada saat pengujian 1 (Tabel 5) dengan jumlah kata yang sama. Penurunan akurasi ini terjadi karena metode PN melakukan pengeditan kata sebanyak dua kali, apabila kata belum ditemukan dalam kamus saat pengeditan pertama.

3.2. Analisis Waktu Komputasi

Dari sisi waktu komputasi, metode PN menunjukkan keunggulan yang signifikan dibandingkan dengan DLD. Pada semua pengujian, waktu komputasi metode PN jauh lebih cepat daripada DLD. Misalnya, pada pengujian pertama, waktu komputasi DLD adalah 67.674 detik, sedangkan metode PN hanya membutuhkan 0.216 detik. Perbedaan ini semakin terlihat pada pengujian dengan jumlah kata yang lebih banyak. Pada pengujian keempat dengan 500 kata, waktu komputasi DLD mencapai 233.316 detik, sementara metode PN hanya membutuhkan 2.925 detik.

Kecepatan komputasi yang dimiliki oleh metode PN membuatnya lebih efisien untuk digunakan dalam sistem atau aplikasi yang membutuhkan koreksi ejaan secara real-time atau dengan jumlah data yang besar.

4. KESIMPULAN

Berdasarkan hasil pengujian yang telah dilakukan, dapat disimpulkan bahwa metode PN memiliki akurasi yang lebih tinggi dan waktu komputasi yang lebih cepat dibandingkan dengan algoritma DLD. Metode PN lebih efektif dalam mengoreksi kata-kata yang salah dan lebih efisien dalam hal waktu komputasi, terutama saat menangani jumlah data yang besar. Meskipun akurasi kedua metode

cenderung menurun seiring dengan bertambahnya jumlah kata yang diuji, metode PN tetap menunjukkan performa yang lebih unggul. Dengan demikian, penerapan metode PN dalam sebuah sistem koreksi ejaan dapat menjadi pilihan yang lebih baik karena keunggulannya dalam akurasi dan efisiensi waktu komputasi.

5. SARAN

Penelitian ini masih memiliki beberapa kekurangan, salah satunya adalah penyusunan kamus kata. Susunan kamus sangat berpengaruh terhadap performa algoritma. Hal lain yang perlu diperhatikan adalah penggunaan pustaka tambahan. Beberapa pustaka mungkin memiliki waktu komputasi dan penggunaan memori yang berbeda-beda, sehingga dapat memberatkan proses program. Berdasarkan kekurangan-kekurangan tersebut, penelitian selanjutnya dapat mempertimbangkan untuk memperbaiki susunan kamus guna menganalisis kedua algoritma lebih lanjut. Selain itu, penggunaan pustaka tambahan perlu dipertimbangkan dengan cermat agar dapat memproses kedua algoritma dengan lebih efisien.

DAFTAR PUSTAKA

- [1] A. P. S and W. J. Hartono, "PENTINGNYA PENGGUNAAN BAHASA INDONESIA DI PERGURUAN TINGGI," *Jotika Journal in Education*, vol. 2, no. 2, pp. 57–64, Feb. 2023, doi: 10.56445/jje.v2i2.84.
- [2] A. P. Sari, D. A. Prasetya, Al Haromainy Muhammad Muharrom, Firza Prima Aditiawan, A. N. Sihananto, and W. S. Saputra, "Analisis Faktor Kesuksesan Penggunaan eBelajar Menggunakan Metode Hot-Fit di STIKI Malang," *Prosiding seminar nasional sains data*, vol. 2, no. 1, pp. 92–102, Nov. 2022.
- [3] F. Augusfian, V. C. Mawardi, J. Hendryli, and D. S. Naga, "SISTEM PENGOREKSAN EJAAN TEKS BAHASA INDONESIA DENGAN DAMERAU LEVENSHTAIN DISTANCE DAN RECURRENT NEURAL NETWORK," *Computatio: Journal of Computer Science and Information Systems*, vol. 3, no. 2, p. 144, Nov. 2019, doi: 10.24912/computatio.v3i2.6038.
- [4] A. P. Wibawa, P. Yuliawati, P. Santoso, R. Shalahuddin, and I. M. Wirawan, "Damerau Levenshtain Distance dengan Metode Empiris untuk Koreksi Ejaan Bahasa Indonesia," *ILKOM Jurnal Ilmiah*, vol. 12, no. 3, pp. 176–182, Dec. 2020, doi: 10.33096/ilkom.v12i3.600.176-182.
- [5] N. Zukarnain, B. S. Abbas, S. Wayan, A. Trisetyarso, and C. H. Kang, "Spelling checker algorithm methods for many languages," in *2019 International Conference on Information Management and Technology (ICIMTech)*, Aug. 2019. Accessed: May 30, 2024. [Online]. Available: <http://dx.doi.org/10.1109/icimtech.2019.8843801>
- [6] G. Navarro, "A guided tour to approximate string matching," *ACM Computing Surveys (CSUR)*, vol. 33, no. 1, pp. 31–88, Mar. 2001, doi: 10.1145/375360.375365.
- [7] K. Toutanova and R. C. Moore, "Pronunciation modeling for improved spelling correction," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2002, pp. 144–151, doi: 10.3115/1073083.1073119
- [8] A. I. Fahma, I. Cholissodin, and R. S. Perdana, "Identifikasi Kesalahan Penulisan Kata (Typographical Error) pada Dokumen Berbahasa Indonesia Menggunakan Metode N-gram dan Levenshtein Distance," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 1, pp. 53–62, Jan. 2018, doi: 10.18412/jiko.v2i1.53-62.
- [9] E. Zahara, "Peranan komunikasi organisasi bagi pimpinan organisasi," *J. Warta*, no. 56, Apr. 2018, Universitas Dharmawangsa, ISSN: 1829-7463.
- [10] Sastrawi. (2024). "Kata Dasar Repository." GitHub. [Online]. Available: <https://github.com/sastrawi/sastrawi/blob/master/data/kata-dasar.txt>. [Accessed: 10-Jun-2024].

- [11] Nolimitid. (2024). "Indonesian Stopwords Complete." GitHub. [Online]. Available: <https://github.com/nolimitid/nolimit-kamus/blob/master/indonesian-stopwords-complete.txt>. [Accessed: 10-Jun-2024].
- [12] Wikipedia. (2024). "Abad Pertengahan." Wikipedia bahasa Indonesia. [Online]. Available: https://id.wikipedia.org/wiki/Abad_Pertengahan. [Accessed: 10-Jun-2024]
- [13] T. Gori, A. Sunyoto, and H. AlFatta, "Preprocessing data dan klasifikasi untuk prediksi kinerja akademik siswa," J. Teknol. Inf. Ilmu Komput., vol. 11, no. 1, pp. 215-224, Feb. 2024, doi: 10.25126/jtiik.20241118074.
- [14] T. M. Fahrudin, I. Sa'diyah, L. Latipah, I. Z. Atha Illah, C. C. Bey Lirna, and B. S. Acarya, "KEBI 1.0: Indonesian Spelling Error Detection System for Scientific Papers using Dictionary Lookup and Peter Norvig Spelling Corrector," Lontar Komputer : Jurnal Ilmiah Teknologi Informasi, vol. 12, no. 2, p. 78, Aug. 2021, doi: 10.24843/lkjiti.2021.v12.i02.p02.
- [15] A. J. Dhruv, R. Patel, and N. Doshi, "Python: The most advanced programming language for computer science applications," in Proceedings of the International Conference on Culture Heritage, Education, Sustainable Tourism, and Innovation Technologies, 2020. Accessed: May 30, 2024. [Online]. Available: <http://dx.doi.org/10.5220/0010307902920299>
- [16] D. N. Fadhillah and A. Rachman, "IMPLEMENTASI REGEX PADA PEMBERIAN KOMENTAR KODE PROGRAM HTML," Jurnal Advanced Research Informatika, vol. 2, no. 1, pp. 1–5, Dec. 2023, doi: 10.24929/jars.v2i1.3078.
- [17] T. M. Fahrudin, I. Sa'diyah, Latipah, I. Z. Atha Illah, C. C. Beylirna, and B. S. Acarya, "Analysis and development of KEBI 1.0 Checker framework as an application of indonesian spelling error detection," Internasional Journal of Data Science, Engineering, and Anaylitics, vol. 1, no. 2, pp. 12–26, Nov. 2021, doi: 10.33005/ijdasea.v1i2.9.