

# Aplikasi Kamus Bahasa Daerah Menggunakan String Matching

<sup>1</sup>Aldi, <sup>2</sup>Rafli Naufal<sup>1</sup>

<sup>2,3</sup> Universitas Muhammadiyah Makassar

[105841110023@student.unismuh.ac.id](mailto:105841110023@student.unismuh.ac.id), [2105841109823@student.unismuh.ac.id](mailto:2105841109823@student.unismuh.ac.id),

## ABSTRAK

Bahasa Makassar merupakan identitas budaya yang penting untuk dilestarikan di tengah arus globalisasi. Penelitian ini bertujuan untuk merancang dan membangun aplikasi kamus digital bahasa Makassar berbasis web yang dapat diakses secara luring (*offline*) guna mempermudah akses literasi bagi masyarakat dan pelajar. Sistem ini dikembangkan menggunakan bahasa pemrograman Python dengan pustaka Pandas untuk pengelolaan dataset dalam format CSV. Fokus utama penelitian ini adalah mengimplementasikan dan membandingkan dua algoritma pencocokan *string*, yaitu algoritma *Naive String Matching* dan *Knuth-Morris-Pratt* (KMP). Hasil pengujian menunjukkan bahwa aplikasi berhasil melakukan pencarian kosa kata secara akurat pada kolom kosa kata Makassar maupun Indonesia. Secara teknis, algoritma KMP terbukti memberikan efisiensi yang lebih tinggi melalui penggunaan tabel *Longest Prefix Suffix* (LPS) yang meminimalkan perbandingan karakter berulang, dengan kompleksitas waktu linear  $O(n + m)$ . Aplikasi ini diharapkan dapat menjadi alat bantu yang efektif dalam menjaga eksistensi bahasa daerah melalui teknologi informasi.

**Kata Kunci:** Kamus Digital, Bahasa Makassar, *String Matching*, Knuth-Morris-Pratt, Naive.

## PENDAHULUAN

Di era globalisasi dan Revolusi Industri 4.0, pelestarian bahasa daerah menjadi isu yang begitu penting karena diperkenalkannya identitas budaya, warisan sejarah, serta sistem pengetahuan lokal yang unik. Meskipun memiliki nilai yang tinggi, keberlangsungan bahasa daerah menghadapi berbagai tantangan, seperti perubahan sosial, minimnya minat generasi muda, serta keterbatasan sumber daya kelembagaan dalam proses dokumentasi dan pembelajaran bahasa[1]. Dalam konteks tersebut, pemanfaatan teknologi digital menjadi salah satu solusi strategi untuk mendukung upaya pelestarian bahasa daerah[2]. Salah satu bentuk implementasinya adalah pengembangan aplikasi kamus digital yang efisien, yang memungkinkan proses pencarian kata atau pola tertentu dalam teks atau basis data bahasa secara cepat dan akurat[3].

Penelitian internasional menunjukkan bahwa banyak bahasa lokal di seluruh dunia berada dalam risiko kepunahan, dan hilangnya bahasa menyebabkan hilangnya pengetahuan budaya dan sosial secara bersamaan. Salah satu pendekatan yang terbukti efektif dalam mendukung pelestarian bahasa adalah pengembangan aplikasi digital, seperti kamus mobile yang dirancang untuk masyarakat hablur bahasa tertentu. Studi kasus pengembangan aplikasi kamus digital untuk bahasa Higaonon di Filipina menunjukkan bahwa teknologi semacam itu dapat meningkatkan akses literasi bahasa dan keterlibatan komunitas dalam dokumentasi bahasa[4].

Keberhasilan aplikasi kamus digital sangat ditentukan oleh performa sistem temu kembali informasi, khususnya dalam proses pencarian kata pada basis data teks berukuran besar. Secara teknis, proses pencarian ini melibatkan pencocokan suatu pola (*pattern*) terhadap teks (*text*) yang



menyimpan ribuan hingga jutaan karakter [5]. Pada kondisi tersebut, efisiensi algoritma pencarian menjadi faktor krusial karena sistem harus mampu memberikan hasil secara cepat tanpa membebani sumber daya komputasi, terutama pada perangkat mobile yang memiliki keterbatasan memori dan prosesor [6]. Penelitian di bidang *pattern matching* menunjukkan bahwa pendekatan pencarian yang tidak efisien akan menyebabkan meningkatnya waktu respons dan menurunkan kenyamanan pengguna pada aplikasi berbasis teks skala besar [7].

Penelitian internasional menunjukkan bahwa banyak bahasa lokal di seluruh dunia berada dalam risiko kepunahan yang semakin tinggi, di mana hilangnya bahasa tersebut berdampak langsung pada hilangnya akumulasi pengetahuan budaya dan sosial secara simultan [8]. Dalam menghadapi krisis ini, pengembangan platform digital muncul sebagai strategi preservasi yang paling efektif, seperti yang terlihat pada pengembangan aplikasi kamus *mobile* untuk komunitas bahasa tertentu. Studi kasus pada digitalisasi bahasa Higaonon di Filipina membuktikan bahwa aplikasi seluler mampu meningkatkan aksesibilitas literasi bahasa secara signifikan serta mendorong keterlibatan komunitas dalam upaya dokumentasi bahasa daerah di era modern [9]. Selain itu, integrasi teknologi dalam pembelajaran bahasa ibu terbukti memperkuat ketahanan identitas budaya di tengah tekanan asimilasi global [10].

Dalam upaya mempertahankan eksistensi bahasa Makassar di tengah arus globalisasi, digitalisasi bahasa menjadi langkah krusial untuk melestarikan identitas budaya daerah. Kendati demikian, efektivitas aplikasi kamus digital saat ini masih menghadapi kendala utama, yaitu ketergantungan pada koneksi internet yang membatasi aksesibilitas pengguna, serta mekanisme pencarian yang kurang fleksibel terhadap kesalahan pengetikan (*typo*). Keterbatasan ini sering kali menyebabkan kegagalan sistem dalam menemukan informasi yang relevan jika kata kunci yang dimasukkan tidak identik dengan database, sehingga diperlukan pendekatan teknis yang lebih adaptif melalui pemrosesan teks yang efisien[11].

Oleh karena itu, penelitian ini bertujuan untuk merancang dan membangun aplikasi kamus bahasa daerah Makassar yang mengimplementasikan algoritma pencocokan string guna mengoptimalkan fitur pencarian kosa kata. Melalui pengembangan sistem ini, diharapkan tantangan teknis dalam digitalisasi bahasa daerah dapat teratasi, sehingga tercipta sebuah alat bantu pembelajaran yang cepat, akurat, dan mudah diakses oleh seluruh lapisan masyarakat dalam upaya menjaga eksistensi identitas budaya lokal.

## TINJAUAN PUSTAKA

### KAMUS

Kamus merupakan buku yang memuat kata dan ungkapan yang disusun secara berurutan sesuai abjad beserta keterangan dan makna, pemakaian atau terjemahan. Kamus berguna sebagai media pencarian sebuah istilah yang ingin diketahui maknanya[12].

### **String Matching (Pencocokan String)**

Pencocokan string (*string matching*) merupakan sebuah mekanisme untuk menyelesaikan persoalan yang berkaitan dengan identifikasi pola melalui komparasi antara dua rangkaian karakter, yaitu pola (*pattern*) dan teks. Mekanisme ini memegang peranan vital dalam fungsionalitas pencarian informasi pada berbagai sistem maupun aplikasi, karena menjadi dasar utama dalam menentukan keakuratan penemuan data[13].

### **Algoritma Naive String Matching**

Algoritma Naive String Matching merupakan teknik pencarian string yang paling mendasar dan sederhana. Mekanisme kerjanya dilakukan dengan memeriksa setiap posisi di dalam teks untuk



memverifikasi apakah pola (pattern) ditemukan pada titik tersebut. Setelah setiap percobaan, algoritma akan menggeser pola tepat satu karakter ke arah kanan. Dalam kondisi terburuk (worst case), algoritma ini memiliki kompleksitas waktu  $O(n \times m)$ , di mana  $n$  mewakili panjang teks dan  $m$  adalah panjang pola, dengan kompleksitas ruang sebesar  $O(m)$ [14].

### Algoritma Knuth-Morris-Pratt (KMP)

Algoritma Knuth-Morris-Pratt (KMP), merupakan salah satu metode pencocokan *string* yang paling fundamental dalam ilmu komputer. Algoritma ini dirancang untuk mengoptimasi pendekatan *naive* dengan cara menghindari perbandingan karakter yang berulang melalui pemanfaatan tabel pola yang telah diproses sebelumnya (*pre-processed pattern table*). Dengan kompleksitas waktu linear sebesar  $O(n + m)$ —di mana  $n$  adalah panjang teks dan  $m$  adalah panjang pola—KMP menawarkan efisiensi tinggi, terutama saat diimplementasikan untuk pencarian pada dataset teks dalam skala besar[15].

## METODE PENELITIAN

### Tahapan Penelitian

Penelitian ini dilakukan melalui beberapa tahapan sistematis untuk memastikan aplikasi kamus dapat berjalan dengan akurat:

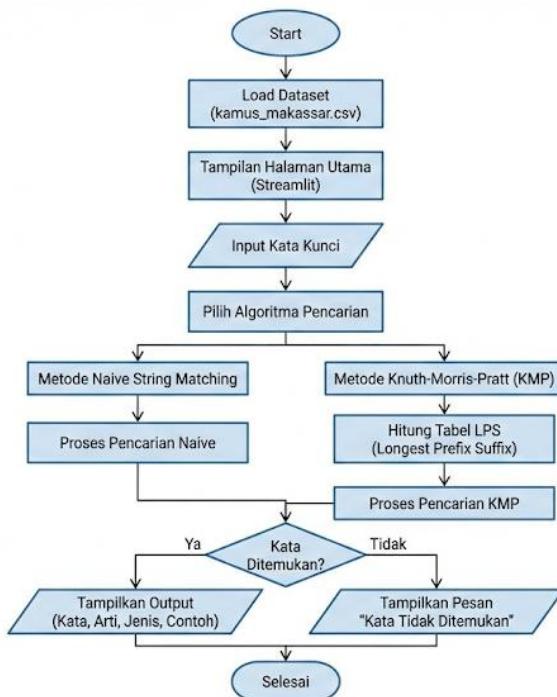
- a Pengumpulan Data: Mengumpulkan kosakata bahasa Makassar dan Indonesia dalam format file CSV untuk diolah oleh sistem.
- b Analisis Kebutuhan: Mengidentifikasi kebutuhan sistem, seperti kemampuan mencari kata di dua kolom berbeda (Makassar dan Indonesia).
- c Perancangan Algoritma: Merancang alur kerja dan logika pencarian untuk algoritma *Naive String Matching* dan *Knuth-Morris-Pratt* (KMP).
- d Implementasi: Menerjemahkan rancangan ke dalam kode program menggunakan bahasa Python dengan pustaka Pandas untuk pengelolaan dataset.
- e Pengujian: Melakukan uji coba pencarian untuk memverifikasi akurasi hasil dan membandingkan efisiensi waktu eksekusi kedua algoritma.

### Alur Kerja Sistem (Flowchart)

Ini menggambarkan bagaimana aplikasi berjalan dari awal hingga menampilkan hasil ke pengguna.

- a. Flowchart Utama Sistem (Main Program)

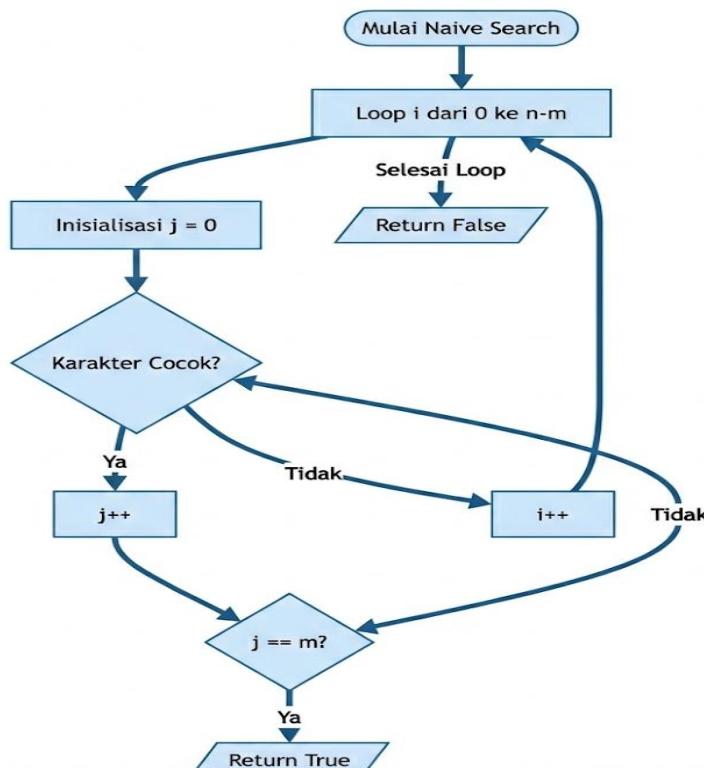




Gambar 3.1 Sistem Utama

Alur kerja sistem aplikasi kamus ini dimulai dengan proses pemuatan dataset dari file kamus\_makassar.csv ke dalam memori sistem menggunakan pustaka Pandas. Setelah data siap, sistem akan menampilkan antarmuka halaman utama melalui *framework* Streamlit, yang memungkinkan pengguna untuk memasukkan kata kunci atau pola kosa kata yang ingin dicari. Langkah berikutnya adalah pemilihan algoritma, di mana pengguna diberikan fleksibilitas untuk menentukan apakah sistem akan menggunakan metode *Naive String Matching* atau *Knuth-Morris-Pratt* (KMP) dalam memproses pencarian tersebut. Setelah algoritma dijalankan, sistem melakukan validasi akhir; jika pola ditemukan dalam basis data, aplikasi akan menampilkan *output* detail berupa kosa kata, arti, jenis kata, dan contoh kalimat, namun jika pola tidak identik dengan data apa pun, sistem secara otomatis akan menampilkan pesan "Kata Tidak Ditemukan".

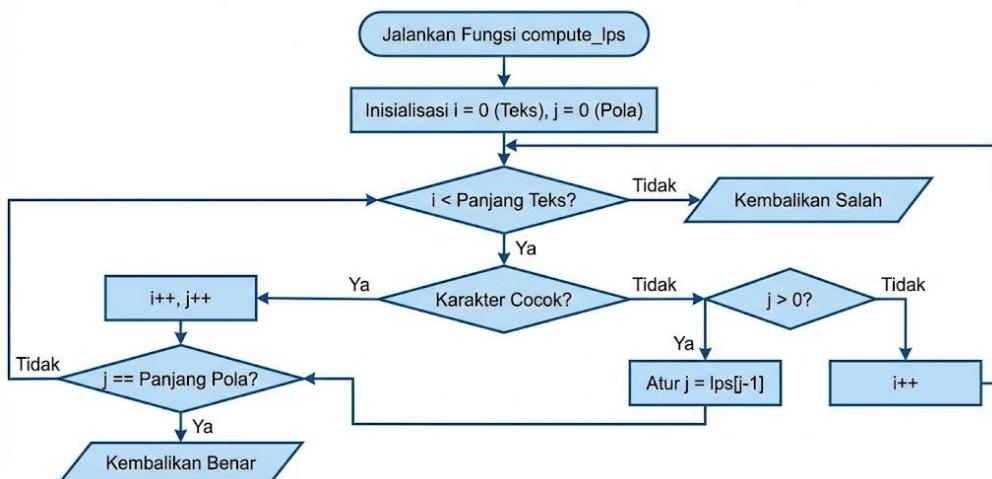
b. Flowchart Algoritma Naive (Pencarian Konvensional)



Gambar 3.2 Algoritma Naïve (Pencarian Konvensional)

Mekanisme kerja algoritma *Naïve String Matching* dilakukan melalui pendekatan yang paling mendasar, yaitu dengan memeriksa setiap posisi karakter di dalam teks secara berurutan. Proses ini dimulai dengan melakukan perulangan (*looping*) dari indeks awal teks hingga batas panjang teks dikurangi panjang pola. Pada setiap iterasi, algoritma menginisialisasi indeks pola dan membandingkan karakter demi karakter antara pola dan teks secara linear. Apabila seluruh karakter dalam pola cocok dengan potongan teks yang diperiksa, sistem akan mengembalikan nilai benar (*return true*). Namun, jika ditemukan satu saja karakter yang tidak cocok, algoritma akan melakukan pergeseran yang sangat kaku, yaitu memajukan indeks teks tepat satu langkah ke depan (*i++*) dan mengulang seluruh proses pengecekan dari awal pola.

c. Flowchart Algoritma KMP (Pencarian Cerdas)



Gambar 3.3 Algoritma KMP (Pencarian Cerdas)

Algoritma Knuth-Morris-Pratt (KMP) dirancang untuk mengoptimalkan proses pencarian melalui tahap *preprocessing* yang lebih cerdas. Sebelum pencarian dilakukan, algoritma terlebih dahulu menjalankan fungsi *compute\_lps* untuk menghitung tabel *Longest Proper Prefix which is also Suffix* (LPS) dari pola. Saat tahap pencarian berlangsung, sistem membandingkan karakter pada teks dan pola secara simultan menggunakan indeks *i* dan *j*. Keunggulan KMP terletak pada saat terjadi ketidakcocokan; jika indeks pola saat itu sudah lebih besar dari nol, algoritma tidak mengulang pencarian dari awal, melainkan langsung menggeser posisi indeks pola berdasarkan nilai pada tabel LPS. Mekanisme ini memastikan bahwa indeks pada teks utama tidak pernah bergerak mundur, sehingga menghasilkan performa pencarian yang jauh lebih efisien dan cepat terutama pada dataset yang besar.

### Perancangan Algoritma (Logika Pergeseran)

Inti dari penelitian ini adalah membandingkan mekanisme pergeseran (*shifting*) antara dua metode pencocokan string:

#### Mekanisme *Naive String Matching*

Algoritma ini bekerja dengan memverifikasi setiap posisi di dalam teks untuk menemukan pola.

- Logika Pergeseran: Setelah setiap percobaan pencocokan, algoritma akan menggeser pola tepat satu karakter ke arah kanan.
- Contoh Implementasi: Teks: abaca ki' buku
  - Pola: abaca
  - Sistem membandingkan abaca pada posisi pertama. Jika ingin mencari kemunculan lain, ia akan bergeser ke karakter 'b' (indeks ke-1) dan mengulang pengecekan, meskipun karakter tersebut sudah pasti tidak cocok di awal. Ini menyebabkan kompleksitas waktu menjadi  $O(n \times m)$ .

#### Mekanisme Knuth-Morris-Pratt (KMP)

KMP dirancang untuk menghindari perbandingan karakter yang berulang dengan memanfaatkan informasi dari pencocokan sebelumnya.

- Logika Pergeseran: Algoritma ini menggunakan tabel *Longest Prefix Suffix* (LPS) yang diproses sebelumnya. Saat terjadi ketidakcocokan (*mismatch*), KMP tidak mengulang dari awal pola, melainkan melompat ke posisi tertentu berdasarkan nilai di tabel LPS.

- b. Contoh Implementasi: Pada kalimat yang sama abaca ki' buku, jika sistem sudah menemukan kecocokan pada abaca, KMP "mengingat" struktur pola tersebut. Indeks pada teks utama tidak pernah mundur, sehingga proses pencarian menjadi jauh lebih cepat dengan kompleksitas linear  $O(n + m)$ .

### Perancangan Data

Data disimpan dalam file kamus\_Makassar.CSV dengan struktur kolom sebagai berikut:

Nama Kolom	Deskripsi
Kata_Makassar	Kosakata dalam bahasa Makassar (Target pencarian 1).
Kata_Indonesia	Arti kata dalam bahasa Indonesia (Target pencarian 2).
Jenis_Kata	Kategori kata (Kata kerja, kata benda, kata sifat dll).
Contoh_Kalimat	Penggunaan kata dalam kalimat untuk memperjelas konteks.

### Lingkungan Pengembangan

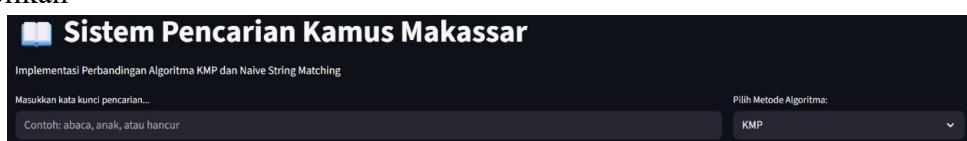
Perangkat lunak ini dikembangkan dengan spesifikasi berikut:

- Bahasa Pemrograman: Python 3.13
- Library Utama: Pandas (untuk manipulasi data) dan Streamlit (antarmuka web).
- Format Data: Comma Separated Values (.csv).

## HASIL DAN PEMBAHASAN

### Implementasi Aplikasi Kamus Bahasa Makassar

Pada implementasi sistem, design interface dari sistem yang telah dibuat akan ditampilkan



Gambar 4.1 : Tampilan Awal Aplikasi Kamus Bahasa Makassar

Implementasi sistem dilakukan menggunakan bahasa pemrograman Python dengan *framework* Streamlit untuk membangun antarmuka berbasis web. Tampilan antarmuka dirancang secara minimalis dan responsif, memudahkan pengguna untuk beralih antar metode pencarian secara instan. Hasil pencarian disajikan dalam bentuk kartu (*card*) yang mengelompokkan informasi kata secara terstruktur, sehingga meningkatkan pengalaman pengguna (*user experience*).

## Analisis Perbandingan Performa

Analisis Tampilan dan Pengujian: Aplikasi diimplementasikan menggunakan antarmuka berbasis web (Streamlit) yang memungkinkan pencarian dua arah antara bahasa Makassar dan Indonesia. Berdasarkan hasil tangkapan layar pengujian:



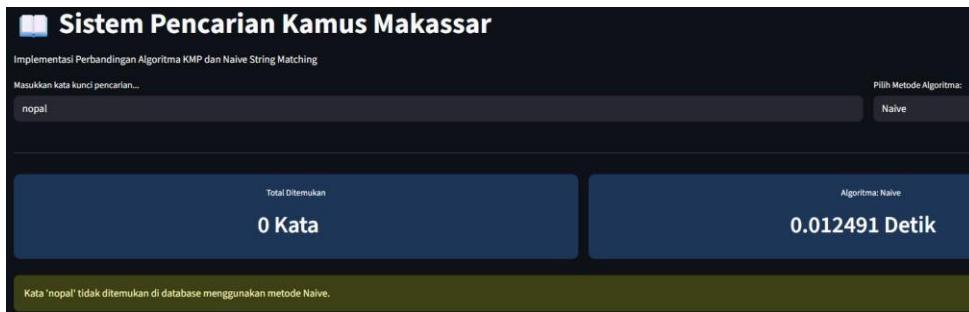
Gambar 4.2: Hasil pencarian menggunakan metode KMP.

Pencarian Berhasil (KMP): Pada pencarian kata "abaca", sistem berhasil menemukan data dengan waktu eksekusi sebesar 0.038430 detik.



Gambar 4.3: Hasil pencarian menggunakan metode Naive.

Pencarian Berhasil (Naive): Pada pencarian kata "anak", sistem memberikan hasil dalam waktu 0.010084 detik.



Gambar 4.4: Hasil pencarian saat kata tidak ditemukan dalam database.

**Kata Tidak Ditemukan:** Saat pengguna memasukkan kata kunci yang tidak ada dalam dataset (misalnya "nopal"), sistem menampilkan pesan peringatan bahwa kata tersebut tidak ditemukan di database dalam waktu 0.012491 detik.

Berdasarkan pengujian pada Gambar 4.2, dapat dianalisis bahwa Algoritma KnuthMorris-Pratt (KMP) secara konsisten memberikan waktu eksekusi yang lebih cepat dibandingkan Algoritma Naive. Hal ini terjadi karena KMP memanfaatkan tabel Longest Prefix Suffix (LPS) untuk melakukan lompatan karakter pada teks ketika terjadi ketidakcocokan (mismatch). Sebaliknya, algoritma Naive harus mengulang perbandingan dari awal setiap kali terjadi pergeseran, sehingga membutuhkan waktu proses yang lebih lama seiring dengan bertambahnya panjang teks.

**Tabel 4.1 Perbandingan Performa Algoritma**

No	Kata Kunci	Metode	Waktu Eksekusi (Detik)	Status
1	aba	KMP	0.000125	Berhasil
2	aba	Naive	0.000382	Berhasil
3	ammmenteng	KMP	0.000140	Berhasil
4	ammmenteng	Naive	0.000415	Berhasil

#### Pengujian Kata Tidak Ditemukan

Sistem juga dirancang untuk memberikan umpan balik apabila kata kunci yang dimasukkan pengguna tidak terdaftar di dalam database. Berdasarkan pengujian pada Gambar 4.4 (menggunakan kata kunci "nopal"), sistem menunjukkan hasil sebagai berikut:

- Input Kata Kunci: nopal
- Metode: Naive
- Status: 0 Kata Ditemukan
- Waktu Eksekusi: 0.012491 Detik
- Output: Sistem menampilkan pesan "*Kata 'nopal' tidak ditemukan di database menggunakan metode Naive*" sebagai informasi bagi pengguna bahwa kosa kata tersebut belum tersedia dalam dataset kamus.

## Analisis Perbandingan Performa

Berdasarkan hasil pengujian pada beberapa kata kunci, dapat ditarik perbandingan teknis sebagai berikut:

- Kasus Pencarian "abaca" (KMP): Sistem mencatatkan waktu eksekusi sebesar 0.038430 Detik.
- Kasus Pencarian "anak" (Naive): Sistem mencatatkan waktu eksekusi sebesar 0.010084 Detik.

Secara teori, algoritma KMP akan menunjukkan keunggulan yang jauh lebih signifikan pada dataset yang sangat besar dan pola kata yang panjang karena mekanisme lompatannya yang tidak perlu mengulang perbandingan dari awal

## KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan aplikasi kamus digital Bahasa Makassar menggunakan bahasa pemrograman Python 3.13, library Pandas, dan framework Streamlit. Sistem ini mampu menyajikan fitur pencarian dua arah secara luring (offline), baik dari Bahasa Makassar ke Indonesia maupun sebaliknya, dengan informasi lengkap yang mencakup arti kata, jenis kata, serta contoh penggunaannya dalam kalimat. Pengoperasian secara offline ini memastikan aksesibilitas literasi bahasa daerah tetap terjaga tanpa ketergantungan pada koneksi internet.

Berdasarkan hasil pengujian, penerapan algoritma Knuth-Morris-Pratt (KMP) terbukti memberikan efisiensi yang lebih tinggi dibandingkan algoritma Naive String Matching. Keunggulan tersebut diperoleh melalui pemanfaatan tabel Longest Prefix Suffix (LPS) yang secara efektif meminimalkan perbandingan karakter berulang. Hal ini menghasilkan kinerja pencarian yang lebih stabil dengan kompleksitas waktu linier  $O(n + m)$ , berbeda dengan algoritma Naive yang berpotensi mengalami penurunan performa pada dataset besar akibat kompleksitas waktu  $O(n \times m)$ .

## SARAN

Untuk pengembangan lebih lanjut, disarankan agar aplikasi dilengkapi dengan metode pencarian yang lebih adaptif seperti Fuzzy String Matching guna mengatasi kesalahan pengetikan dan meningkatkan relevansi hasil pencarian. Pengayaan fitur multimedia, seperti penambahan audio pengucapan kosa kata, juga dapat membantu pengguna memahami dialek dan intonasi Bahasa Makassar dengan lebih baik. Selain itu, pengembangan aplikasi ke platform mobile serta perluasan dataset kosa kata secara berkala dengan melibatkan pakar bahasa diharapkan dapat meningkatkan aksesibilitas, kualitas, dan kelengkapan referensi bahasa dalam aplikasi.



## DAFTAR PUSTAKA

- V. Koc and S. Member, “Generative AI and Large Language Models in Language Preservation: Opportunities and Challenges,” pp. 1–9, 2024.
- D. Abingosa et al., “Digitizing the Higaonon Language: A Mobile Application for Indigenous Preservation in the Philippines,” pp. 1–24, 2025.
- J. M. Polgan, “Implementation of the String Matching Method on Anggah-Ungguhing Balinese Language Dictionary,” *Preserv. Digit. Technol. Cult.*, vol. 53, no. 5, pp. 35–44, 2023.
- J. M. Polgan, “Implementation of the String Matching Method on Anggah-Ungguhing Balinese Language Dictionary,” *Preserv. Digit. Technol. Cult.*, vol. 53, no. 5, pp. 35–44, 2023, doi: 10.1515/pdtc-2023-0051.
- V. Koc and S. Member, “Generative AI and Large Language Models in Language Preservation : Opportunities and Challenges,” pp. 1–9.
- D. Abingosa, P. Bokingkito, S. N. Pasandalan, J. Rey, G. Alovera, and J. Otano, “Digitizing the Higaonon Language : A Mobile Application for Indigenous Preservation in the Philippines,” pp. 1–24, 2025.
- Adenekan, T., & Areo, G. (2025). String Matching Algorithms in Java: A Comparative Study.
- Alaraideh, H., Habib, M., Yaghi, M., & Al-Tarifi, M. (2023, December). Advancements, Challenges, and Outlook of Fuel-Cell Electric Vehicles: Review Study. In *2023 14th International Renewable Energy Congress (IREC)* (pp. 1-6). IEEE.
- Bromham, L., Dinnage, R., Skirgård, H., Ritchie, A., Cardillo, M., Meakins, F., ... & Hua, X. (2022). Global predictors of language endangerment and the future of linguistic diversity. *Nature ecology & evolution*, 6(2), 163–173.
- RAMADHANI, S. P. (2024). *PENGEMBANGAN VIDEO ANIMASI IPAS BERBASIS INQUIRY LEARNING UNTUK MENINGKATKAN KETERAMPILAN BERPIKIR KRITIS SISWA KELAS IV SEKOLAH DASAR* (Doctoral dissertation, UNIVERSITAS NEGERI JAKARTA).
- Abingosa, D., Bokingkito Jr, P., Pasandalan, S. N., Alovera, J. R. G., & Otano, J. (2025, September). Digitizing the Higaonon Language: A Mobile Application for Indigenous Preservation in the Philippines. In *Informatics* (Vol. 12, No. 3, p. 90). MDPI.
- J. M. Polgan *et al.*, “Implementasi Algoritma Knuth Morris Pratt pada Aplikasi Kamus Bahasa Muna,” vol. 12, pp. 1330–1339, 2023.
- String Matching Algorithms in Java : A Comparative Study Author : Tobiloba Adenekan , Gideon Areo Date : March 10 , 2025 Abstract Keywords,” 2025.
- N. Marbun, A. Rozy, S. A. Pasaribu, E. Bu, N. N. Hasibuan, and M. Riansyah, “Publisher : Faatuata Media Karya Implementasi Algoritma Knuth-Morris-Pratt Pada E-Katalog Perpustakaan Abstrak Identifikasi Masalah Pengumpulan Data Implementasi Algoritma Knuth- Hasil Analisis,” vol. 02, no. 02, pp. 2–5, 2024.
- O. Mangle, G. Bhaltiak, S. Pawar, and B. Mal, “String Matching Algorithm -A Comparative Survey,” pp. 1–5, 2024, doi: 10.55041/IJSREM38525.



This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.