

TESIS

ALGORITMA STEMMING UNTUK TEKS BAHASA KAILI



Disusun oleh:

Nama : Tamrizal A.M
NIM : 20.55.1378
Konsentrasi : Business Intelligence

PROGRAM STUDI PJJ S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2022

TESIS

ALGORITMA STEMMING UNTUK TEKS BAHASA KAILI

STEMMING ALGORITHM FOR KAILI LANGUAGE TEXT

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister



Disusun oleh:

Nama : Tamrizal A.M
NIM : 20.55.1378
Konsentrasi : Business Intelligence

PROGRAM STUDI PJJ S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA

2022

HALAMAN PENGESAHAN

ALGORITMA STEMMING UNTUK TEKS BAHASA KAILI

STEMMING ALGORITHM FOR KAILI LANGUAGE TEXT

Dipersiapkan dan Disusun oleh

Tamrizal A.M

20.55.1378

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi PJJ S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Rabu, 4 Januari 2023

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer.

Yogyakarta, 4 Januari 2023

Rektor

Prof. Dr. M. Suyanto, M.M.

NIK. 190302001

HALAMAN PERSETUJUAN
ALGORITMA STEMMING UNTUK TEKS BAHASA KAILI
STEMMING ALGORITHM FOR KAILI LANGUAGE TEXT

Dipersiapkan dan Disusun oleh

Tamrizal A.M
20.55.1378

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi PJJ S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Rabu, 4 Januari 2023

Pembimbing Utama

Prof. Dr. Kusriani, M.Kom
NIK. 190302106

Pembimbing Pendamping

M. Rudyanto Arief, M.T
NIK. 190302098

Anggota Tim Penguji

Dhani Ariatmanto, M.Kom., Ph.D.
NIK. 190302197

Dr. Kumara Ari Yuana, S.T., M.T.
NIK. 190302575

Prof. Dr. Kusriani, M.Kom
NIK. 190302106

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 4 Januari 2023
Direktur Program Pascasarjana

Dr. Kusriani, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : **Tamrizal A.M**
NIM : **20.55.1378**
Konsentrasi : **Business Intelligence**

Menyatakan bahwa Tesis dengan judul berikut:
Algoritma Stemming Untuk Teks Bahasa Kawi

Dosen Pembimbing Utama : **Prof. Dr. Kusriani, M.Kom**
Dosen Pembimbing Pendamping : **M. Rudyanto Arief, M.T**

1. Karya tulis ini adalah benar-benar **ASLI** dan **BELUM PERNAH** diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian **SAYA** sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab **SAYA**, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini **SAYA** buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka **SAYA** bersedia menerima **SANKSI AKADEMIK** dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 4 Januari 2023
Yang Menyatakan,



Tamrizal A.M

HALAMAN PERSEMBAHAN

Persembahan utama tentunya kepada Allah SWT atas segala ridhanya sehingga dimudahkan dalam perjalanan menuntut ilmu hingga penelitian tesis dapat terselesaikan.

Kemudian kepada keluarga besar, istri dan anak-anakku tersayang atas segala dukungan dan keceriaannya hingga lelah dan letih dapat berubah menjadi semangat dalam menuntut ilmu.



HALAMAN MOTTO

“Maka sesungguhnya beserta kesulitan ada kemudahan.” - **(QS. Al-Insyirah: 5)**

“Siapa yang menempuh jalan untuk mencari ilmu, maka Allah akan mudahkan baginya jalan menuju surga.” - **(HR Muslim, no. 2699).**

“Ilmu adalah kehidupan bagi pikiran.” - **Abu Bakar**

“Ilmu itu lebih baik dari kekayaan, karena kekayaan itu harus dijaga, sedangkan ilmu menjaga kamu.” - **Ali bin Abi Thalib**

“Jika kamu tidak sanggup menahan lelahnya belajar maka kamu harus sanggup menahan perihnya kebodohan.” - **Imam Syafi'i**

KATA PENGANTAR

Segala puji bagi Allah SWT tuhan semesta alam atas limpahan rahmat dan ridhanya sehingga penulis dapat menyelesaikan penelitian tesis dengan judul “Algoritma Stemming Untuk Teks Bahasa Kaili” sebagai salah satu syarat memperoleh derajat magister pada program studi S2 PJJ Teknik Informatik Universitas AMIKOM Yogyakarta. Penulis juga ingin menyampaikan sebesar-besarnya ucapan terimakasih kepada:

1. Bapak Prof. Dr. M. Suyanto, M.M, sebagai rektor Universitas AMIKOM Yogyakarta yang selalu enerjik dan bersemangat.
2. Ibu Prof. Dr. Kusriani, M.Kom, selaku Direktur Pascasarjana Universitas AMIKOM Yogyakarta serta sebagai pembimbing utama penelitian tesis saya, yang telah banyak memberikan arahan dalam penyusunan penelitian tesis ini.
3. Bapak M. Rudyanto Arief, M.T, sebagai pembimbing pendamping atas masukannya sehingga hasil akhir dari penelitian tesis yang disusun menjadi semakin baik.
4. Bapak Dr. Kumara Ari Yuana, S.T., M.T., selaku anggota tim penguji ujian tesis dan Seminar Proposal Tesis atas masukannya yang membangun untuk penelitian tesis saya.
5. Bapak Dhani Ariatmanto, M.Kom., Ph.D., selaku anggota tim penguji ujian tesis dan Seminar Hasil Penelitian Tesis atas koreksi dan sarannya untuk perbaikan tesis saya.
6. Bapak Hanif Al Fatta, M.Kom. selaku anggota tim penguji Seminar Hasil Penelitian Tesis atas dorongan dan masukannya untuk penelitian tesis saya.

7. Bapak Emha Taufiq Luthfi, S.T., M.Kom. selaku anggota tim penguji Seminar Proposal Tesis atas koreksinya untuk perbaikan penelitian tesis saya.
8. Bu Tutut, Mba Laily dan semua petugas di admisi yang tidak bisa saya sebut satu per satu, yang telah banyak membantu selama saya menuntut ilmu di kampus AMIKOM.
9. Bapak M. Muslami, S.T, M.H serta seluruh rekan-rekan di Bagian Keuangan dan SIMRS RSUD Ratu Aji Putri Botung atas dukungan dan pengertiannya selama proses perkuliahan online.
10. Istri tercinta (Febriana Shooan Perdani) dan anak-anakku tersayang (Inara dan Akhtar) serta seluruh keluarga besar atas segala dukungan dan doa.
11. Rekan-rekan seperjuangan prodi S2 PJJ Teknik Informatika angkatan 2020 dari berbagai wilayah Indonesia yang saling mendukung walau jarak memisahkan dalam perjuangan perkuliahan online.
12. Masyarakat Suku Kaili di Sulawesi Tengah yang menjadi sumber inspirasi dalam penelitian tesis ini.

Penulis sangat menyadari bahwa sebagai manusia yang tak luput dari khilaf dan salah, penelitian tesis ini masih banyak terdapat kekurangan. Sehingga kritik dan saran akan sangat diterima sebagai masukan yang membangun. Semoga penelitian tesis ini dapat memberi manfaat khususnya bagi masyarakat suku Kaili.

Yogyakarta, 4 Januari 2023

Penulis

DAFTAR ISI

HALAMAN JUDUL	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERSETUJUAN	iv
HALAMAN PERNYATAAN KEASLIAN TESIS	v
HALAMAN PERSEMBAHAN	vi
HALAMAN MOTTO	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
INTISARI	xv
<i>ABSTRACT</i>	xvi
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah	4
1.4. Tujuan Penelitian	4
1.5. Manfaat Penelitian	4
BAB II TINJAUAN PUSTAKA	5
2.1. Tinjauan Pustaka	5
2.2. Keaslian Penelitian	8
2.3. Landasan Teori	13



BAB III METODE PENELITIAN	20
3.1. Jenis, Sifat, dan Pendekatan Penelitian	20
3.2. Metode Pengumpulan Data	20
3.3. Metode Analisis Data	21
3.4. Alur Penelitian	21
BAB IV HASIL PENELITIAN DAN PEMBAHASAN	25
4.1. Pengumpulan Data	25
4.2. Analisis Data	26
4.3. Pengujian	63
4.4. Evaluasi	112
BAB V PENUTUP	118
5.1. Kesimpulan	118
5.2. Saran	119
DAFTAR PUSTAKA	120
LAMPIRAN	122

DAFTAR TABEL

Tabel 2.1. Matriks literatur review dan posisi penelitian Algoritma Stemming Untuk Teks Bahasa Kaili	8
Tabel 4.1 Daftar Imbuhan Bahasa Kaili	26
Tabel 4.2 Aturan Imbuhan Bahasa Kaili	27
Tabel 4.3 Urutan algoritma berdasarkan tingkat akurasi	80
Tabel 4.4 Hasil stemming Topomeka	83
Tabel 4.5 Hasil stemming Sikola	87
Tabel 4.6 Hasil stemming Sapeda	93
Tabel 4.7 Hasil stemming Noasu Bavu	100
Tabel 4.8 Hasil stemming Nodanda	107
Tabel 4.9 Data kesalahan stemming algoritma 14	112
Tabel 4.10 Data kesalahan pengujian	115

DAFTAR GAMBAR

Gambar 2.1 Algoritma Nazief & Adriani	19
Gambar 3.1 Alur Penelitian.....	22
Gambar 4. 1 Algoritma pengecekan awalan	32
Gambar 4.2 Pengecekan imbuhan awalan 'no'	33
Gambar 4. 3 Algoritma Pengecekan akhiran	34
Gambar 4.4 Pengecekan imbuhan akhiran 'pa'	34
Gambar 4. 5 Algoritma pengecekan imbuhan apit	35
Gambar 4.6 Pengecekan imbuhan apit 'nomba ... taka'	35
Gambar 4. 7 Algoritma pengecekan imbuhan sisipan	36
Gambar 4.8 Pengecekan imbuhan sisipan 'imb'	36
Gambar 4.9 Pengecekan kata dasar	37
Gambar 4.10 Pengujian Algoritma 1	64
Gambar 4.11 Pengujian Algoritma 2	65
Gambar 4.12 Pengujian Algoritma 3	65
Gambar 4.13 Pengujian Algoritma 4	66
Gambar 4.14 Pengujian Algoritma 5	67
Gambar 4.15 Pengujian Algoritma 6	67
Gambar 4.16 Pengujian Algoritma 7	68
Gambar 4.17 Pengujian Algoritma 8	69
Gambar 4.18 Pengujian Algoritma 9	69
Gambar 4.19 Pengujian Algoritma 10	70

Gambar 4.20 Pengujian Algoritma 11	71
Gambar 4.21 Pengujian Algoritma 12	71
Gambar 4.22 Pengujian Algoritma 13	72
Gambar 4.23 Pengujian Algoritma 14	72
Gambar 4.24 Pengujian Algoritma 15	73
Gambar 4.25 Pengujian Algoritma 16	74
Gambar 4.26 Pengujian Algoritma 17	75
Gambar 4.27 Pengujian Algoritma 18	75
Gambar 4.28 Pengujian Algoritma 19	76
Gambar 4.29 Pengujian Algoritma 20	77
Gambar 4.30 Pengujian Algoritma 21	77
Gambar 4.31 Pengujian Algoritma 22	78
Gambar 4.32 Pengujian Algoritma 23	79
Gambar 4.33 Pengujian Algoritma 24	79
Gambar 4.34 Grafik tingkat akurasi algoritma stemming	80

INTISARI

Indonesia merupakan sebuah bangsa majemuk yang terdiri dari berbagai Suku Bangsa dan Bahasa. Berdasarkan Penelitian untuk pemetaan bahasa di Indonesia yang dilakukan oleh Badan Pengembangan Bahasa dan Perbukuan, Kementerian Pendidikan dan Kebudayaan, hingga 2019 teridentifikasi sebanyak 718 bahasa. Dari ratusan Bahasa yang telah teridentifikasi, salah satunya adalah Bahasa Kaili yang digunakan oleh orang-orang dari suku kaili yang banyak mendiami wilayah Sulawesi Tengah.

Dalam bidang teknologi informasi, ilmu yang berfokus pada pengolahan bahasa dikenal dengan Natural Language Processing (NLP) yang merupakan cabang dari kecerdasan buatan untuk membantu komputer dalam memahami, menafsirkan dan memanipulasi bahasa manusia. Dari berbagai terapan NLP, stemming merupakan proses dasar yang paling sering dibutuhkan. Stemming adalah proses pemetaan atau penguraian bentuk dari suatu kata menjadi kata dasar.

Saat ini banyak algoritma telah dikembangkan untuk melakukan stemming seperti algoritma porter, algoritma tala, algoritma nazief dan adriani, algoritma arifin dan setiono, algoritma idris dan mustapha, algoritma vega, algoritma ahmad yusuf dan sembok, algoritma paice-husk dan masih banyak lagi. Berdasarkan penelitian yang pernah dilakukan mengenai tingkat akurasi dari berbagai algoritma yang ada, algoritma nazief dan adriani akan digunakan sebagai dasar dengan menyesuaikan morfologi dari bahasa Kaili khususnya dialek Ledo dalam melakukan penyusunan algoritma stemming untuk teks bahasa Kaili.

Dalam penelitian ini dibuat 24 algoritma yang dapat disusun berdasarkan aturan imbuhan. Kemudian diuji menggunakan 1000 kata dalam bahasa kaili untuk mengetahui algoritma dengan tingkat akurasi yang terbaik. Hasil pengujian menunjukkan tiga algoritma terbaik adalah algoritma 14 dengan tingkat akurasi mencapai 96,6%, kemudian algoritma 17 dengan tingkat akurasi mencapai 96,5% dan algoritma 18 dengan tingkat akurasi mencapai 96,5%. Selanjutnya 3 algoritma teratas yaitu algoritma 14, algoritma 17 dan algoritma 18 diuji menggunakan 5 cerita dalam bahasa kaili, hasil pengujian menunjukkan tingkat akurasi dari ketiga algoritma tersebut mencapai 93,6%.

Kata kunci: algoritma, Nazief&Adriani, stemming, kaili, ledo

ABSTRACT

Indonesia is a pluralistic nation consisting of various ethnic groups and languages. Based on research for language mapping in Indonesia conducted by the Language and Book Development Agency, Ministry of Education and Culture, as of 2019 there were 718 languages identified. Of the hundreds of languages that have been identified, one of them is the Kaili language which is used by people from the Kaili tribe who inhabit the Central Sulawesi region.

In the field of information technology, the science that focuses on language processing is known as Natural Language Processing (NLP) which is a branch of artificial intelligence to help computers understand, interpret and manipulate human language. Of the various NLP applications, stemming is the basic process that is most often needed. Stemming is the process of mapping or parsing the form of a word into a basic word.

Currently, many algorithms have been developed to perform stemming such as Porter's algorithm, tuning algorithm, Nazief and Adriani algorithm, Arifin and Setiono algorithm, Idris and Mustapha algorithm, Vega algorithm, Ahmad Yusof and Sembok algorithm, Paice-Husk algorithm and many more. Based on research that has been done regarding the accuracy of various existing algorithms, the nazief and adriani algorithms will be used as a basis by adjusting the morphology of the Kaili language, especially the Ledo dialect in compiling the stemming algorithm for the Kaili language text.

In this study, 24 algorithms were created which can be arranged based on affix rules. Then tested using 1000 words in kaili language to find out the algorithm with the best level of accuracy. The test results show that the three best algorithms are algorithm 14 with an accuracy rate of 96.6%, then algorithm 17 with an accuracy rate of 96.5% and algorithm 18 with an accuracy rate of 96.5%. Furthermore, the top 3 algorithms were tested using 5 stories in Kaili language, the test results showed the accuracy level of the three algorithms reached 93.6%.

Keyword: *algorithm, Nazief&Adriani, stemming, kaili, ledo*

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Indonesia merupakan sebuah bangsa majemuk yang terdiri dari berbagai Suku Bangsa dan Bahasa. Berdasarkan Penelitian untuk pemetaan bahasa di Indonesia yang dilakukan oleh Badan Pengembangan Bahasa dan Perbukuan, Kementerian Pendidikan dan Kebudayaan, hingga 2019 teridentifikasi sebanyak 718 bahasa (sumber: <https://petabahasa.kemdikbud.go.id/>). Jumlah bahasa tersebut menjadikan Indonesia sebagai salah satu negara yang memiliki Bahasa daerah terbanyak di dunia. Keragaman tersebut tentu perlu dirawat dan dikembangkan sebagai bagian dari kebhinekaan bangsa.

Dari ratusan Bahasa yang telah teridentifikasi, salah satunya adalah Bahasa Kaili yang digunakan oleh orang-orang dari suku kaili yang banyak mendiami wilayah Sulawesi Tengah. Bahasa Kaili terdiri atas beberapa dialek diantaranya Tara, Taje, Ledo, Da'a, Rai, Unde, Unde Kabonga, Kori, Njedu dan Pendau, diantara berbagai dialek tersebut, dialek Ledo adalah yang paling banyak digunakan oleh penutur Bahasa Kaili. Dialek Ledo merupakan dialek standar karena selain dituturkan di pusat pemerintahan/ibu kota provinsi, sebaran geografisnya luas dan jumlah penuturnya lebih besar, dialek Ledo juga digunakan dalam media massa lokal baik cetak maupun elektronik. (sumber: <https://petabahasa.kemdikbud.go.id/infobahasa2.php?idb=159&idp=Sulawesi%20Tengah>). Sciring dengan perkembangan zaman, bahasa Kaili mulai jarang

digunakan masyarakat suku Kaili karena sudah banyak yang menggunakan bahasa Indonesia dalam komunikasi sehari-hari.

Dalam bidang teknologi informasi, ilmu yang berfokus pada pengolahan bahasa dikenal dengan Natural Language Processing (NLP) yang merupakan cabang dari kecerdasan buatan untuk membantu komputer dalam memahami, menafsirkan dan memanipulasi bahasa manusia (Liddy, E.D, 2001). NLP mempunyai berbagai terapan seperti pengolah kata, chatbot, summarization, translation tools, interactive voice response (IVR), stemming dan sebagainya.

Dari berbagai terapan NLP, stemming merupakan proses dasar yang paling sering dibutuhkan. Stemming adalah proses pemetaan atau penguraian bentuk dari suatu kata menjadi kata dasar (Muchtari, M.A dkk, 2019). Proses stemming biasanya dibutuhkan pada kegiatan information retrieval (pencarian informasi) untuk meningkatkan kualitas informasi yang didapatkan. Cara kerja stemming umumnya dilakukan dengan membandingkan kata dengan kamus kata dasar ataupun dengan melakukan proses menghilangkan imbuhan yang mengiringi kata dasar menggunakan aturan morfologi bahasa.

Saat ini banyak algoritma telah dikembangkan untuk melakukan stemming seperti algoritma porter, algoritma tala, algoritma nazief dan adriani, algoritma arifin dan setiono, algoritma idris dan mustapha, algoritma vega, algoritma ahmad yusuf dan sembok, algoritma paice-husk dan masih banyak lagi. Diantara berbagai algoritma yang telah disebutkan, di Indonesia algoritma nazief dan adriani adalah yang paling populer karena tingkat akurasi dalam melakukan stemming bahasa Indonesia yang cukup tinggi. Beberapa penelitian yang pernah dilakukan

menunjukkan bahwa tingkat akurasi algoritma nazief dan adriani lebih baik dari algoritma yang lain, seperti penelitian yang dilakukan oleh Jumadi dkk yang membandingkan algoritma nazief dan adriani dengan algoritma paice-husk (Jumadi dkk, 2021) ataupun penelitian yang dilakukan oleh Prasadhatama dkk yang melakukan perbandingan algoritma nazief dan adriani dengan algoritma idris dan Mustapha (Prasadhatama dkk, 2018). Kedua penelitian tersebut menunjukkan bahwa algoritma nazief dan adriani mempunyai tingkat akurasi yang lebih baik dalam melakukan stemming bahasa Indonesia.

Berdasarkan penelitian yang pernah dilakukan mengenai tingkat akurasi tersebut diatas, algoritma nazief dan adriani akan digunakan sebagai dasar dengan menyesuaikan morfologi dari bahasa Kaili khususnya dialek Ledo dalam melakukan penyusunan algoritma stemming untuk teks bahasa Kaili.

1.2. Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dijabarkan diatas, maka pada penelitian ini diharapkan dapat memberikan kontribusi yang meliputi:

- a. Apakah dapat dihasilkan dataset berupa kumpulan kata dasar dalam bahasa Kaili-Ledo?.
- b. Apakah dapat dihasilkan algoritma hasil modifikasi dari algoritma nazief dan adriani yang bisa digunakan untuk melakukan stemming bahasa Kaili Ledo?.
- c. Apakah tingkat akurasi dari algoritma yang dihasilkan cukup baik untuk melakukan stemming bahasa Kaili-Ledo?.

1.3. Batasan Masalah

Batasan masalah dalam penelitian yang dilakukan adalah:

- a. Bahasa kaili yang digunakan dalam penelitian ini adalah dialek ledo.
- b. Kata dasar yang digunakan diambil dari kamus Kaili-Ledo – Indonesia – Inggris (Evans, 2003).
- c. Tata bahasa berdasarkan buku Morfologi dan Sintaksis Bahasa Kaili (Inghuong dkk, 1979).
- d. Algoritma yang akan dimodifikasi adalah algoritma nazief dan adriani.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah:

- a. Dapat menghasilkan algoritma stemming untuk bahasa kaili khususnya dialek ledo.
- b. Mengetahui akurasi dari algoritma stemming yang diusulkan.

1.5. Manfaat Penelitian

Manfaat yang diharapkan dari penelitian ini adalah:

- a. Menghasilkan algoritma yang dapat digunakan untuk melakukan stemming bahasa kaili.
- b. Mendukung program pemerintah dalam melestarikan bahasa daerah.

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Penelitian yang dilakukan oleh Jumadi dkk (2021) membandingkan algoritma stemming Nazief-Adriani and Paice-Husk untuk mengetahui tingkat akurasi dari masing-masing algoritma dalam melakukan stemming bahasa Indonesia. Pengujian yang dilakukan menggunakan 200 kalimat dalam bahasa Indonesia dengan 1958 kata berimbuhan. Hasil penelitian yang dilakukan menunjukkan tingkat akurasi algoritma Nazief-Adriani mencapai 91,87%, lebih baik dari algoritma Paice-Husk yang tingkat akurasi hanya 64,43%.

Wardani dkk (2020) melakukan penelitian stemming teks bahasa Bali dengan algoritma Enhanced Confix Stripping (ECS). Pengujian dilakukan dengan menggunakan 376 kata dalam bahasa Bali dengan rincian 240 kata prefix, 17 kata infiks dan 119 kata sufiks. Dari hasil pengujian didapatkan tingkat akurasi algoritma ECS mencapai 96,94% dari 376 kata yang diuji.

Kemudian penelitian yang dilakukan oleh Arifin dkk (2020) yang melakukan penelitian tentang stemming bahasa Jawa dengan mengadopsi algoritma stemming Nazief-Adriani yang biasa digunakan untuk stemming bahasa Indonesia. Dalam penelitian ini, untuk menguji algoritma yang dihasilkan digunakan 13.011 kata berimbuhan dalam bahasa Jawa. Hasil pengujian menunjukkan algoritma yang dihasilkan mempunyai tingkat akurasi mencapai 94,72%.

Penelitian serupa dilakukan oleh Wibawa dkk (2020), penelitian yang dilakukan adalah dengan melakukan modifikasi terhadap algoritma Nazief-Adriani agar sesuai dengan morfologi bahasa Jawa sehingga dapat digunakan dalam melakukan stemming bahasa Jawa. Untuk melakukan tes terhadap algoritma hasil modifikasi, digunakan 366 kata dalam bahasa Jawa untuk menguji keakuratan algoritma yang dihasilkan. Setelah dilakukan pengujian didapatkan tingkat akurasi sebesar 95,9%.

Penelitian lain dilakukan oleh Yusliani dkk (2019) yang bertujuan untuk mempercepat waktu proses yang dibutuhkan algoritma Enhanced Confix Stripping (ECS) dalam melakukan stemming. Pada penelitian yang dilakukan, digunakan pendekatan multiprocessing saat menjalankan proses stemming. Hasil penelitian yang dilakukan menunjukkan bahwa pendekatan multiprocessing dapat menurunkan waktu rata-rata proses stemming hingga 98,45%.

Selanjutnya penelitian yang dilakukan oleh Muchtar dkk (2019) yang melakukan penelitian stemming bahasa Batak Toba menggunakan algoritma Porter yang telah disesuaikan. Pada penelitian ini untuk menguji algoritma Porter yang telah disesuaikan, digunakan 20 dokumen berbahasa Batak Toba untuk dilakukan stemming. Hasil dari penelitian yang dilakukan menunjukkan tingkat akurasi mencapai 87,503%.

Kemudian Prasadhatama, A & Suryaningrum, K M, yang melakukan penelitian tentang perbandingan algoritma Nazief-Adriani dengan algoritma Idris dalam melakukan stemming. Pengujian dilakukan dengan menggunakan 5 buah dokumen cerita rakyat berbahasa Indonesia yang berisi total 3.174 kata. Dari

pengujian yang dilakukan didapatkan bahwa algoritma Nazief-Adriani mempunyai tingkat akurasi lebih baik yang mencapai 97,50%.

Kemudian penelitian yang dilakukan oleh Maylawati dkk (2018) yang melakukan modifikasi pada algoritma porter agar dapat melakukan stemming terhadap bahasa gaul Indonesia. Pengujian terhadap algoritma yang telah dimodifikasi dilakukan menggunakan dua skenario, yang pertama menggunakan 379 kata dengan berbagai imbuhan termasuk dalam bentuk bahasa gaul, kemudian yang kedua menggunakan 20 text data yang berasal dari twitter yang mengandung bahasa gaul Indonesia. Dari pengujian yang dilakukan didapatkan tingkat akurasi dari algoritma hasil modifikasi mencapai 88,65%.

Selanjutnya Putra dkk (2018) melakukan penelitian algoritma stemming untuk bahasa Indonesia non-formal. Pada penelitian ini dilakukan modifikasi terhadap algoritma Nazief-Adriani dengan menambahkan aturan afiks non formal. Pengujian dilakukan dengan menggunakan 60 kata non-formal, dari hasil pengujian didapatkan tingkat akurasi mencapai 73,33%.

Dari tinjauan pustaka, berbagai pendekatan dilakukan untuk dapat melakukan stemming bahasa yang berbeda-beda, kontribusi dari penelitian adalah menemukan algoritma yang sesuai untuk melakukan stemming bahasa Kaili.

2.2. Keastian Penelitian

Tabel 2.1. Matriks literatur review dan posisi penelitian
Algoritma Stemming Untuk Teks Bahasa Kaili

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1	Stemming Teks Bahasa Bali dengan Algoritma Enhanced Confix Stripping.	Wardani, N.W. & Nugraha, P.G.S.C., International Journal of Natural Science and Engineering, 2020.	Melakukan modifikasi algoritma Enhanced Confix Stripping (ECS) agar dapat digunakan untuk melakukan stemming teks bahasa Bali.	Tingkat akurasi dari stemming teks bahasa Bali menggunakan algoritma Enhanced Confix Stripping (ECS) yang telah disesuaikan mencapai 96,94% dari 376 kata yang diuji.	Pengujian yang dilakukan hanya untuk mengetahui tingkat akurasi dari algoritma hasil modifikasi dari algoritma ECS, akan lebih baik jika waktu proses yang dibutuhkan algoritma untuk menyelesaikan proses stemming juga diukur untuk mengetahui kecepatan dari algoritma tersebut.	Pada penelitian ini dilakukan modifikasi algoritma ECS agar dapat melakukan stemming bahasa Bali, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang dapat digunakan melakukan stemming bahasa Kaili Ledo.
2	Stemming Javanese: Another Adaptation of the Nazief-Adriani Algorithm	Nq, M.A., Manik, L.P. & Widiyatmoko, D., 3rd International Seminar on Research of Information Technology and Intelligent	Melakukan penyesuaian algoritma Nazief-Adriani agar dapat digunakan untuk melakukan stemming teks bahasa Jawa	Tingkat akurasi dari stemming teks bahasa Jawa menggunakan algoritma Nazief-Adriani yang telah disesuaikan mencapai 94,72% dari 13.011 kata yang diuji.	Pengujian yang dilakukan hanya untuk mengetahui tingkat akurasi dari algoritma hasil modifikasi dari algoritma Nazief-andriani, akan lebih baik jika waktu proses yang dibutuhkan algoritma untuk menyelesaikan	Pada penelitian ini dilakukan modifikasi algoritma nazief-andriani agar dapat melakukan stemming bahasa Jawa, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang dapat digunakan melakukan stemming bahasa Kaili Ledo.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
		Systems (ISRITI), 2020			proses stemming juga diukur untuk mengetahui kecepatan dari algoritma tersebut.	
3	Stemming javanese affix words using nazief and adriani modifications.	Wibawa, A.P., Dwiyanto, F.A., Zaeni, I.A.E., Nurrohman, R.K. & Afandi, A.N., J. Inform., 2020	Melakukan modifikasi algoritma Nazief-Adriani agar dapat digunakan untuk melakukan stemming teks bahasa Jawa	Tingkat akurasi dari stemming teks bahasa Jawa menggunakan algoritma Nazief-Adriani yang telah disesuaikan mencapai 95,9% dari 366 kata yang diuji.	Pengujian yang dilakukan hanya untuk mengetahui tingkat akurasi dari algoritma hasil modifikasi dari algoritma Nazief-andriani, akan lebih baik jika waktu proses yang dibutuhkan algoritma untuk menyelesaikan proses stemming juga diukur untuk mengetahui kecepatan dari algoritma tersebut.	Pada penelitian ini dilakukan modifikasi algoritma nazief-andriani agar dapat melakukan stemming bahasa Jawa, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang dapat digunakan melakukan stemming bahasa Kaili Ledo.
4	Implementation of Porter Stemmer Algorithm to Obtain Basic Words in Toba Batak Language Documents with the Two-Level Morphological Method.	Muchtar, M.A., Nababan, E.B., Nababan, M., Andayani, U., Simanjuntak, T. & Sitompul, O.S., IOP Conference Series: Materials Science and	Melakukan modifikasi algoritma Porter agar dapat digunakan untuk melakukan stemming teks bahasa Jawa	Tingkat akurasi dari stemming teks bahasa Batak Toba menggunakan algoritma Porter yang telah disesuaikan mencapai 87,503% dari 20 dokument berbahasa Batak Toba yang diuji serta rata-rata waktu pengujian adalah	Dapat dilakukan percobaan menggunakan algoritma lain untuk meningkatkan akurasi.	Pada penelitian ini dilakukan modifikasi algoritma porter agar dapat melakukan stemming bahasa Batak Toba, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang dapat digunakan melakukan stemming bahasa Kaili Ledo.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
5	An improved of stemming algorithm for mining Indonesian text with slang on social media.	Maylawati, D.S.A., Zulfikar, W.B., Slamet, C., Ramdhani, M.A. & Gerhana, Y.A., 6th International Conference on Cyber and IT Service Management (CITSM), 2018	Mengusulkan algoritma yang dapat digunakan untuk melakukan stemming bahasa gaul Indonesia kemudian dibandingkan dengan hasil dari algoritma porter, Nazief-Adriani dan Lucene.	0.000460798 detik per kata. Algoritma yang diusulkan menunjukkan tingkat akurasi mencapai 88,65% lebih baik dari tingkat akurasi algoritma porter yang mencapai 64,12%, algoritma Nazief-Adriani 40,11% dan Lucene 43,27% tetapi waktu proses yang dibutuhkan masih lebih lama dibandingkan algoritma yang lain, serta penggunaan memori yang lebih besar dibandingkan algoritma yang lain.	Hasil pengujian yang menunjukkan penggunaan memori yang sangat besar memungkinkan untuk dilakukan modifikasi dari algoritma yang diusulkan untuk mengurangi penggunaan memori	Pada penelitian ini dilakukan modifikasi algoritma nazief-andriani agar dapat melakukan stemming bahasa gaul Indonesia, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang dapat digunakan melakukan stemming bahasa Kaili Ledo.
6	Non-formal affixed word stemming in Indonesian language.	Putra, R.B.S. and Utami, E., International Conference on Information & Communications Technology	Melakukan modifikasi algoritma Nazief-Adriani agar dapat mendeteksi kata-kata non-formal dalam bahasa Indonesia	Tingkat akurasi dari stemming teks non formal bahasa Indonesia menggunakan algoritma Nazief-Adriani yang telah dimodifikasi	Pengujian dari algoritma yang diusulkan dapat dilakukan dengan menggunakan jumlah kata yang lebih besar untuk menilai tingkat akurasi.	Pada penelitian ini dilakukan modifikasi algoritma nazief-andriani agar dapat melakukan stemming bahasa non formal Indonesia, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
7	Multiprocessing Stemming: A Case Study of Indonesian Stemming.	(ICOACT), 2018 Yusliani, N., Primurtha, R. & Mariessa, M.D., International Journal Computer & Applications (IJCA), 2019	Melakukan pengujian algoritma stemming menggunakan metode multiprocessing untuk meningkatkan kecepatan proses	mencapai 73,33% dari 60 kata yang diuji. Hasil pengujian penggunaan metode multiprocessing secara signifikan dapat meningkatkan kecepatan proses sampai 98,45%	Penelitian hanya dilakukan untuk mengukur kecepatan proses, tidak ditunjukkan besar memori yang dibutuhkan untuk melakukan proses tersebut, akan lebih baik jika ditampilkan untuk membuat perbandingan yang lebih baik.	dapat digunakan melakukan stemming bahasa Kaili Ledo. Pada penelitian ini dilakukan pengujian algoritma stemming menggunakan metode multiprocessing untuk meningkatkan kecepatan stemming, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang dapat digunakan melakukan stemming bahasa Kaili Ledo.
8	Comparison of Nazief-Adriani and Paice-Husk algorithm for Indonesian text stemming process.	Jumadi, J., Maylawati, D.S., Pratiwi, L.D. and Ramadhani, M.A., IOP Conference Series: Materials Science and Engineering, 2021	Membandingkan algoritma Nazief-Adriani dengan algoritma Paice-Husk untuk mengetahui tingkat akurasi dari masing-masing algoritma dalam melakukan stemming bahasa Indonesia	Algoritma Nazief-Adriani mempunyai tingkat akurasi mencapai 91,87% lebih baik dari algoritma Paice-Husk yang tingkat akurasinya mencapai 64,43%.	Algoritma Paice-Husk dibuat untuk melakukan stemming dalam bahasa Inggris, sehingga diperlukan penyesuaian agar dapat efektif melakukan stemming dalam bahasa Indonesia.	Pada penelitian ini dilakukan pengujian algoritma Nazief-Adriani dan Paice-Husk untuk mengetahui tingkat akurasi dari masing-masing algoritma. Tingkat akurasinya kemudian dibandingkan untuk mengetahui algoritma yang lebih baik, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang dapat digunakan melakukan stemming bahasa Kaili Ledo.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
9	Perbandingan Algoritma Nazief & Adriani Dengan Algoritma Idris Untuk Pencarian Kata Dasar.	Prasidhatama, A. and Suryaningrum, K.M., Jurnal Teknologi dan Manajemen Informatika, 2018	Membandingkan algoritma Nazief-Adriani dengan algoritma Idris untuk mengetahui tingkat akurasi dari masing-masing algoritma dalam melakukan stemming bahasa Indonesia	Algoritma Nazief-Adriani mempunyai tingkat akurasi mencapai 97,50% lebih baik dibandingkan dengan algoritma Idris yang tingkat akurasinya mencapai 91,37%. Tetapi algoritma idris mempunyai kecepatan proses 0,027 detik/kata lebih baik dari algoritma Nazief-Adriani yang mempunyai kecepatan proses 0,035 detik/kata.	Dokumen yang digunakan untuk melakukan pengujian adalah cerita rakyat, dapat dicoba dilakukan dengan dokumen Bahasa Indonesia yang lain untuk pengujian lebih lanjut terutama algoritma Idris yang dibuat untuk stemming bahasa melayu.	Algoritma Nazief-Adriani dan Idris untuk mengetahui tingkat akurasi dari masing-masing algoritma. Tingkat akurasinya kemudian dibandingkan untuk mengetahui algoritma yang lebih baik, sedangkan penelitian yang dilakukan untuk mendapatkan algoritma yang dapat digunakan melakukan stemming bahasa Kaili Ledo.

2.3. Landasan Teori

1. Bahasa Kaili

Bahasa Kaili merupakan bahasa yang bertanah asal di Kabupaten Donggala, Parigi, dan Kota Palu, Provinsi Sulawesi Tengah. Berdasarkan penghitungan dialektometri bahasa ini memiliki sepuluh dialek (petabahasa.kemdikbud.go.id), yaitu :

- a. Dialek Tara dituturkan di Desa Olaya, Kecamatan Parigi dan Desa Dolago, Kecamatan Parigi Selatan, Kabupaten Parigi Moutong; Kelurahan Lasoani, Kecamatan Mantikulore, Kota Palu; Desa Tinggede, Kecamatan Marawola dan Desa Sibalaya Selatan, Kecamatan Tanambulava, Kabupaten Sigi.
- b. Dialek Taje dituturkan di Desa Petapa, Kecamatan Parigi Tengah dan Desa Sidole, Kecamatan Ampibabo, Kabupaten Parigi Moutong.
- c. Dialek Ledo dituturkan di Kelurahan Lolu Utara, Kecamatan Palu Timur; Kelurahan Kayumalue Pajeko, Kecamatan Palu Utara, Kota Palu; Desa Kotarindau, Kecamatan Dolo dan Desa Sintuwu, Kecamatan Palolo, Kabupaten Sigi; Desa Towale, Kecamatan Banawa Tengah, Kabupaten Donggala; Dialek Ledo ini merupakan dialek yang paling banyak penuturnya dibandingkan dengan kesembilan dialek yang lain. Dialek Ledo merupakan dialek standar karena selain dituturkan di pusat pemerintahan/ibu kota provinsi,

sebaran geografisnya luas dan jumlah penuturnya lebih besar, dialek Ledo juga digunakan dalam media massa cetak dan elektronik.

- d. Dialek Daa dituturkan di Desa Panturabate, Kecamatan Dolo; Desa Uwemanje, Kecamatan Kinovaro; Desa Lebanu, Kecamatan Marawola, Kabupaten Sigi; dan Desa Mbuwu, Kecamatan Banawa Selatan, Kabupaten Donggala.
- e. Dialek Rai dituturkan di Desa Lende, Kecamatan Sirenja; Desa Toaya, Kecamatan Sindue, Kabupaten Donggala; Desa Sibowi, Kecamatan Tinambulava, Kabupaten Sigi; Desa Mpanau, Kecamatan Sigi Biromaru, Kabupaten Sigi; Kelurahan Baiya, Kecamatan Tawaeli, Kota Palu; dan Desa Toini, Kecamatan Poso Pesisir, Kabupaten Poso.
- f. Dialek Unde dituturkan di Kelurahan Watusampu, Kecamatan Ulujadi, Kota Palu dan Desa Dalaka, Kecamatan Sindue, Kabupaten Donggala.
- g. Dialek Unde Kabonga dituturkan di Kelurahan Kabonga Besar, Kecamatan Banawa, Kabupaten Donggala.
- h. Dialek Kori dituturkan di Desa Taripa, Kecamatan Sindue, Kabupaten Donggala.
- i. Dialek Njedu dituturkan di Desa Enu, Kecamatan Sindue, Kabupaten Donggala.
- j. Dialek Pendau dituturkan di Desa Tambu, Kecamatan Balaesang, Kabupaten Donggala.

Ledo, Taje, Rai, Daa (Da'a), Tara, Unde, Unde Kabonga, Kori, Njedu, dan Pendau merupakan dialek yang berbeda dari bahasa yang sama karena penghitungan dialektometri menunjukkan adanya perbedaan yang berkisar antara 51%—75%. Bahasa Kaili juga memiliki perbedaan dialek dengan Sedoa, Bada, Togian, Bare'e (Bare'e), Pamona, Saluan, dan Tado (perbedaannya berkisar antara 71%—79%).

2. Morfologi

Morfologi merupakan salah satu dari cabang ilmu bidang bahasa yang menerangkan tentang awal terbentuk sebuah kata (Suparno, 2015). Morfologi bahasa tidak hanya mempelajari tentang awal terbentuk kata, namun juga mempelajari pergeseran makna.

2.1. Morfem

Morfem adalah unsur-unsur terkecil yang masing-masing mempunyai makna dalam tutur sebuah bahasa (Rumilah, 2020). Morfem dapat dibedakan atas dua macam, yaitu morfem bebas dan morfem terikat. Morfem bebas ialah semua bentuk bebas dari suatu morfem yang belum berkombinasi dengan bentuk lain, serta mendukung makna sendiri. Morfem terikat ialah morfem yang tidak dapat menduduki posisi tersendiri dalam kalimat, melainkan hanya membantu kedudukan morfem bebas.

2.2. Kata

Kata merupakan morfem atau kombinasi morfem, sebagai satuan terkecil yang dapat diujarkan sebagai bentuk bebas (Rumilah, 2020). Patokan yang umumnya dipakai untuk menentukan ciri kata yaitu stabilitas fonologis dan mobilitas sintagmatis. Stabilitas fonologis berarti kecenderungan unit fonologis memperlihatkan ketetapan sistem yang terdapat dalam struktur kata. Bahasa Kaili mempunyai ciri yang mudah dikenal, yaitu tidak terdapatnya fonem konsonan diakhir kata (Sofyan, 1979). Sedangkan mobilitas sintagmatis yaitu sekurang-kurangnya memiliki satu dari empat kemungkinan berikut: dapat disendirikan, dapat digantikan posisinya oleh yang lain, dapat dipisahkan oleh satuan lain, serta letaknya di dalam deretan bisa berbeda-beda.

Pada umumnya berdasarkan linguistik struktural kata digolongkan atas empat atau lima kelas kata dengan berbagai variasinya berdasarkan bahasa masing-masing. Pada bahasa Kaili terbagi atas lima kelas kata yaitu:

- Kelas I - Benda
- Kelas II - Kerja
- Kelas III - Sifat
- Kelas IV - Bilangan
- Kelas V - Tugas

2.3. Pengimbuhan

Proses pengimbuhan merupakan suatu proses yang terjadi apabila sebuah morfem terikat dibubuhkan atau dilekatkan pada sebuah morfem bebas secara urutan lurus (Sofyan, 1979). Imbuhan menurut posisinya terbagi atas awalan (prefix), sisipan (infix), akhiran (sufiks) serta penggunaan awalan dan akhiran secara bersamaan (konfiks).

3. Information Retrieval (IR)

Information retrieval atau sistem temu kembali informasi adalah suatu proses menemukan kembali informasi yang cocok dan sesuai dengan keinginan user dari kumpulan informasi (Kodimala, 2010). Pada dasarnya sistem temu kembali informasi bekerja dengan mencocokkan query dengan kata yang tersimpan dalam database. Nilai balikan dari query tersebut adalah kumpulan dokumen yang relevan serta membuang dokumen yang tidak relevan.

Dalam information retrieval kata yang mengandung morfologi terkadang dapat mengurangi akurasi. Hal ini menyebabkan perlunya untuk membuang imbuhan yang mengiringi kata, sehingga kata kembali ke bentuk awalnya.

4. Stemming

Stemming adalah proses menemukan kata dasar dari sebuah kata dengan menghilangkan imbuhan yang menyertai kata tersebut (Kodimala, 2010). Proses stemming sering digunakan pada topik information retrieval. Hal ini bertujuan untuk meningkatkan nilai akurasi dari informasi yang

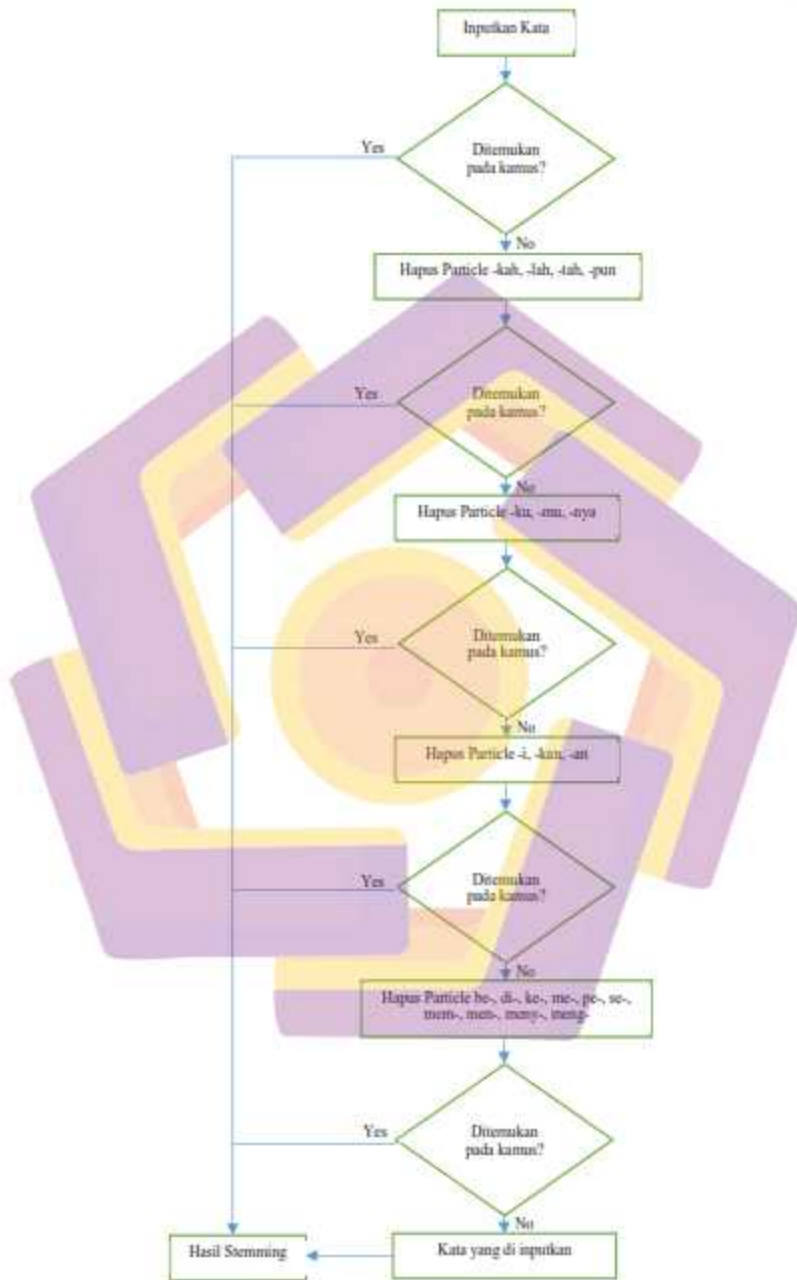
diperoleh. Stemming memiliki fungsi untuk menemukan kata dasar dari kata yang diberikan.

Dalam proses stemming sering ditemukan kesalahan dalam proses pemenggalan imbuhan dari kata dasarnya. Kesalahan ini dapat dibagi kedalam 4 kategori (Mandala dkk, 2004) yaitu overstemming yaitu proses pemenggalan imbuhan yang melebihi dari yang seharusnya, understemming yaitu proses pemenggalan imbuhan yang kurang dari yang seharusnya, unchange yaitu proses yang seharusnya terjadi pemenggalan imbuhan tetapi pada prosesnya tidak terjadi pemenggalan imbuhan sama sekali dan spelling exception pemenggalan imbuhan yang tidak benar karena kesalahan dalam proses stemming dalam menentukan aturan yang digunakan.

Banyak penelitian mengenai algoritma yang dipakai dalam proses stemming. Contoh algoritma yang dipakai untuk stemming teks bahasa Indonesia adalah algoritma enhanced confix stripping (ECS), algoritma nazief-andriani, algoritma arifin-setiono, algoritma idris-mustofa, algoritma vega, algoritma confix stripping (CS) dan lain sebagainya. Untuk teks berbahasa asing diantaranya algoritma porter stemmer untuk teks bahasa Inggris.

5. Algoritma Nazief dan Adriani

Algoritma Nazief dan Adriani merupakan salah satu algoritma yang populer dan banyak digunakan untuk proses stemming bahasa Indonesia (Jumadi dkk, 2021). Pada algoritma Nazief dan Adriani langkah-langkah stemming yang dilakukan adalah sebagai berikut (Adriani dkk, 2007).



Gambar 2.1 Algoritma Nazief & Adriani

BAB III

METODE PENELITIAN

3.1. Jenis, Sifat, dan Pendekatan Penelitian

Metode penelitian yang digunakan adalah eksperimental, karena akan dilakukan pengujian untuk mengetahui tingkat akurasi dari algoritma yang disusun pada penelitian ini. Adapun sifat penelitian yang dilakukan menggunakan metode deskriptif analisis dengan menggunakan kumpulan kata dan imbuhan dalam bahasa Kaili-Ledo yang dianalisis agar mendapatkan susunan algoritma yang dapat digunakan untuk melakukan stemming bahasa Kaili-Ledo.

Dalam melakukan penelitian ini digunakan buku Morfologi dan Sintaksis Bahasa Kaili (Inghuon dkk, 1979) dan Kamus Kaili-Ledo – Indonesia – Inggris (Evans, 2003) sebagai acuan utama dalam menentukan daftar kata dasar dan aturan imbuhan yang digunakan dalam bahasa Kaili-Ledo. Kemudian daftar kata dasar dan aturan imbuhan tersebut akan digunakan sebagai dasar dalam menyusun algoritma stemming untuk bahasa Kaili-Ledo.

3.2. Metode Pengumpulan Data

Penelitian dilakukan melalui studi kepustakaan, dengan mengumpulkan bahan referensi yang terkait dengan penelitian, baik melalui buku, artikel, jurnal, paper dan mengunjungi situs – situs di internet terkait dengan algoritma stemming, morfologi dan tata bahasa dalam bahasa Kaili khususnya dialek Ledo. Data yang digunakan adalah kumpulan morfem, kata dan imbuhan dalam bahasa Kaili

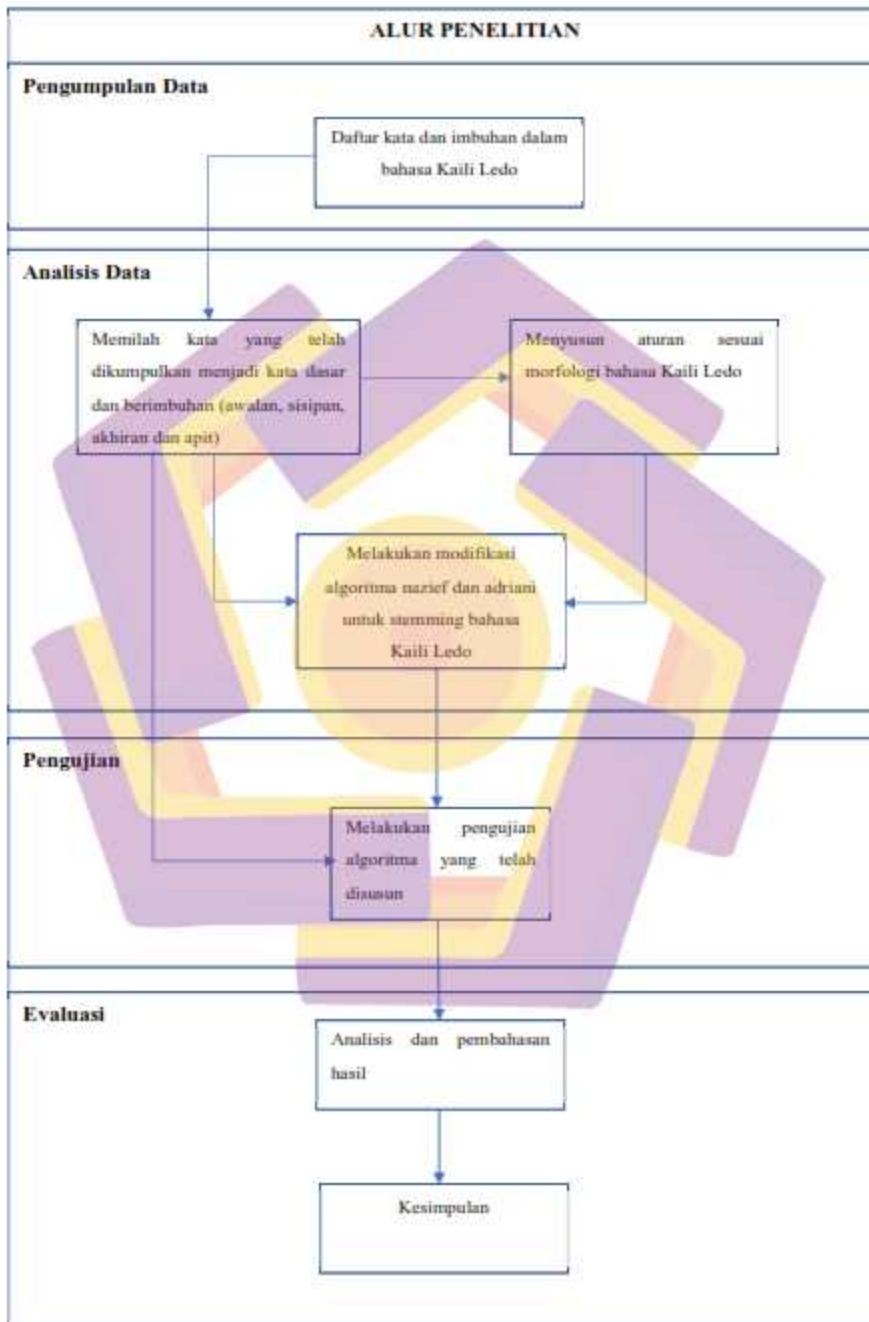
khususnya dialek Ledo untuk dianalisis sehingga dapat ditentukan algoritma yang cocok untuk melakukan stemming.

3.3. Metode Analisis Data

Pada penelitian ini akan dikumpulkan data berupa kata-kata dan jenis imbuhan yang digunakan dalam bahasa Kaili-Ledo, kemudian dianalisis untuk mendapatkan daftar kata dasar serta aturan dari penggunaan imbuhan. Kemudian berdasarkan daftar kata dasar dan aturan imbuhan tersebut, disusun sebuah algoritma dengan melakukan modifikasi dari algoritma Nazief dan Adriani yang dapat digunakan dalam melakukan stemming kata-kata dalam bahasa Kaili-Ledo. Selanjutnya algoritma yang telah disusun, diuji dengan sejumlah data berupa kata-kata dalam bahasa Kaili-Ledo untuk mengetahui tingkat akurasi dan kesalahan dari algoritma yang telah disusun.

3.4. Alur Penelitian

Dalam melakukan penelitian, diperlukan adanya tahapan-tahapan yang diurutkan secara sistematis agar pelaksanaan penelitian dapat berjalan dengan baik. Adapun alur pada penelitian ini dapat dilihat pada gambar 1 sebagai berikut:



Gambar 3.1 Alur Penelitian

Adapun secara garis besar urutan penelitian adalah sebagai berikut:

a. Tahapan Pengumpulan Data

Tahap ini bertujuan untuk mengumpulkan data yang berkaitan dengan penelitian ini yang berupa daftar kata dan imbuhan dalam bahasa Kaili Ledo.

b. Tahap Analisis

Pada tahap ini data yang didapatkan dipilah menjadi daftar kata dasar yang akan digunakan sebagai acuan dalam algoritma yang akan disusun, kata berimbuhan yang diperlukan pada tahap pengujian, daftar awalan (prefix), sisipan (infix), akhiran (suffix) dan gabungan (konfix) yang akan digunakan dalam proses penyusunan aturan-aturan imbuhan.

Langkah selanjutnya adalah menyusun aturan-aturan imbuhan berdasarkan data imbuhan prefix, infix, suffix dan konfix sesuai dengan morfologi bahasa Kaili Ledo. Penyusunan aturan ini akan memperhatikan perubahan bunyi dari sebuah kata saat bertemu dengan sebuah imbuhan. Hal ini diperlukan agar algoritma yang dihasilkan mempunyai akurasi yang tinggi.

Kemudian dilakukan modifikasi algoritma nazief dan adriani berdasarkan aturan-aturan yang telah terbentuk, proses modifikasi algoritma juga akan memanfaatkan daftar kata dasar sebagai acuan pembandingan saat proses pemisahan kata dasar dari imbuhan.

c. Tahap Pengujian

Pada tahap ini, algoritma stemming yang telah disusun akan diuji dengan melakukan stemming terhadap 969 kata dalam bahasa Kaili Ledo. Jumlah kata yang berhasil distemming akan dihitung untuk kemudian dibandingkan dengan jumlah kata yang diuji untuk menghitung tingkat akurasi dan kesalahan, perhitungan akurasi dan kesalahan dirumuskan dalam formula berikut.

$$\text{Akurasi} = \frac{\text{Jumlah kata yang berhasil distemming}}{\text{Jumlah Kata yang distemming}} \times 100\%$$

$$\text{Kesalahan} = \frac{\text{Jumlah kata yang gagal distemming}}{\text{Jumlah Kata yang distemming}} \times 100\%$$

Sehingga dari proses pengujian yang dilakukan akan didapatkan persentase akurasi dan kesalahan dari algoritma stemming yang telah disusun.

d. Tahap Evaluasi

Pada tahap ini akan dilakukan evaluasi dari algoritma yang telah disusun berdasarkan hasil pada tahap pengujian untuk kemudian disimpulkan hasil dari penelitian yang telah dilakukan.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1. Pengumpulan Data

Pengumpulan data ini dilakukan untuk mengetahui daftar kata dan imbuhan yang digunakan dalam bahasa Kaili Ledo. Untuk keperluan tersebut digunakan Kamus Kaili-Ledo – Indonesia – Inggris yang disusun oleh Donna Evans dan secara resmi diterbitkan oleh Dinas Kebudayaan dan Pariwisata Pemerintah Daerah Propinsi Sulawesi Tengah, sebagai acuan kata-kata yang digunakan dalam bahasa Kaili serta buku Morfologi dan Sintaksis Bahasa Kaili yang disusun oleh Inghuong alias Sofyan, Syahrudin Kaseng, M. Sikki dan Patuko Pepy yang diterbitkan oleh Pusat Pembinaan dan Pengembangan Bahasa Departemen Pendidikan dan Kebudayaan sebagai acuan imbuhan yang digunakan pada bahasa Kaili.

Berdasarkan daftar kata yang terdapat dalam Kamus Kaili-Ledo diketahui bahwa dalam bahasa Kaili-Ledo terdapat kata yang diawali lima huruf vokal a, e, i, o dan u, sedangkan kata yang diawali huruf konsonan terdiri atas huruf b, c, d, g, h, j, k, l, m, n, p, r, s, t, v, w dan y, serta tidak terdapat kata yang diawali huruf f, q, w, x dan z.

Untuk imbuhan yang digunakan dalam bahasa Kaili-Ledo terdapat 93 imbuhan yang terdiri atas 50 imbuhan awalan, 4 imbuhan sisipan, 6 imbuhan akhiran dan 33 imbuhan apit.

4.2. Analisis Data

Berdasarkan data yang telah dikumpulkan, dilakukan analisis untuk memilah data kata dasar yang digunakan dalam bahasa Kaili. Dari hasil pemilahan yang dilakukan, didapatkan 4127 kata dasar (data terlampir).

Selain kata dasar, dilakukan juga analisis terhadap 93 imbuhan yang terdiri dari 50 imbuhan awalan, 4 imbuhan sisipan, 6 imbuhan akhiran dan 33 imbuhan apit. Berikut daftar imbuhan yang terdapat pada bahasa Kaili-Ledo.

Tabel 4.1 Daftar Imbuhan Bahasa Kaili

1	Awalan mombapaka-	32	Awalan rape-	63	Apit nomba- -taka
2	Awalan nombapaka-	33	Awalan nipe-	64	Apit momba- -taka
3	Awalan nosipopo-	34	Awalan pari-	65	Apit nomba- -saka
4	Awalan mosipopo-	35	Awalan topo-	66	Apit momba- -saka
5	Awalan mompaka-	36	Awalan mpo-	67	Apit nomba- -ka
6	Awalan nompaka-	37	Awalan ma-	68	Apit momba- -ka
7	Awalan mosipaka-	38	Awalan mo-	69	Apit nomba- -si
8	Awalan nosipaka-	39	Awalan me-	70	Apit momba- -si
9	Awalan nosipari-	40	Awalan ne-	71	Apit nomba- -ti
10	Awalan mosipari-	41	Awalan ni-	72	Apit momba- -ti
11	Awalan nomposi-	42	Awalan na-	73	Apit nomba- -i
12	Awalan momposi-	43	Awalan no-	74	Apit momba- -i
13	Awalan nompari-	44	Awalan pa-	75	Apit nosi- -raka
14	Awalan mompari-	45	Awalan po-	76	Apit mosi- -raka
15	Awalan rapaka-	46	Awalan ra-	77	Apit nosi- -taka
16	Awalan nipaka-	47	Awalan sa-	78	Apit mosi- -taka
17	Awalan nipari-	48	Awalan ka-	79	Apit nosi- -saka
18	Awalan nipopo-	49	Awalan pe-	80	Apit mosi- -saka
19	Awalan sangga-	50	Awalan to-	81	Apit nosi- -si
20	Awalan momba-	51	Sisipan -in-	82	Apit mosi- -si
21	Awalan nomba-	52	Sisipan -um-	83	Apit nosi- -ka
22	Awalan nosi -	53	Sisipan -imb-	84	Apit mosi- -ka
23	Awalan mosi -	54	Sisipan -il-	85	Apit na- -ka
24	Awalan posi -	55	Akhiran -a	86	Apit ma- -ka
25	Awalan meti-	56	Akhiran -i	87	Apit pe- -na
26	Awalan neti-	57	Akhiran -si	88	Apit ka- -na
27	Awalan ngga-	58	Akhiran -ti	89	Apit pa- -a
28	Awalan paka-	59	Akhiran -pa	90	Apit ma- -i

Tabel 4.1 (Lanjutan)

29	Awalan naka-	60	Akhiran -mo	91	Apit na- i
30	Awalan maka-	61	Apit nomba- -raka	92	Apit pe- a
31	Awalan popo-	62	Apit momba- -raka	93	Apit ka- a

Selanjutnya menyusun aturan sesuai ketentuan imbuhan yang telah ditemukan. Tujuan dari aturan yang dibuat adalah untuk menentukan format dari imbuhan dan peluluhan katanya agar mempermudah memisahkan imbuhan dengan kata dasarnya. Berdasarkan analisis yang dilakukan terhadap imbuhan yang terdapat dalam Bahasa Kaili, diusulkan 113 aturan imbuhan dengan rincian sebagai berikut.

Tabel 4.2 Aturan Imbuhan Bahasa Kaili

No	Aturan	Format	Peluluhan	Contoh
Awalan ma-				
1	1a	ma...	ma - ...	mause => ma + ase
2	1b	man/d/	man - d...	mandu => man + du
3	1c	man/t/	man - t...	manbe => man + teba
4	1d	manjV...	manj - s...	manjili => manj + sili
5	1e	mangV	mang - V...	mangelo => mang + elo
Awalan mo-				
6	2a	moK...	mo - K...	mokolu => mo + kolu
7	2b	mom/p/	mom - p...	mompe => mom + pe
8	2c	mom/b/	mom - b...	mombe => mom + be
Awalan me-				
9	3	me...	me - ...	mesani => me + sani
Awalan ne-				
10	4	ne...	ne - ...	nemange => ne + mänge
Awalan ni-				
11	5	ni...	ni - ...	niome => ni + ome
Awalan na-				
12	6a	na...	na - ...	nauda => na + uda
13	6b	nanjV...	nanj - /s/...	nanjyo => nanj + sayo
14	6c	nangV...	nang - V...	nangelo => nang + elo
15	6d	nanK...	nan - K...	nandu => nan + du

Tabel 4.2 (Lanjutan)

	Awalan no-		
16	7	no...	no - ... noana => no + ana
	Awalan pa-		
17	8a	pa/k/...	pa - /k/... pakande => pa + kande
18	8b	pan/d/...	pan - /d/... pandiu => pan + diu
19	8c	pan/v/...	pan - /v/... pantora => pan + tora
20	8d	pangV...	pang - V... pangala => pang + ala
21	8e	paniV...	pani - /s/... panjambei => pani + sambei
	Awalan po-		
22	9a	po...	po - ... podau => po + dau
23	9b	pon...	pon - ... pumboliaka => pon + holiaka
	Awalan ra-		
24	10	ra...	ra - ... ritula => ra + rula
	Awalan sa-		
25	11a	sa...	sa - ... saei => sa + ei
26	11b	san/b/...	san - /b/... santanga => san + tanga
27	11c	san/d/...	san - /d/... sandanga => san + danga
28	11d	sanjV...	sanj - /n/... sanjiku => sanj + iku
29	11e	sangV...	sang - /k/... sanggeke => sang + keke
30	11f	samb/...	sam - /b/... sambana => sam + bana
	Awalan ka-		
31	12	ka...	ka - ... kauntu => ka + untu
	Awalan momba-		
32	13a	momba...	momba - ... mombaopa => momba + opa
33	13b	mba...	mba - ... mbaala => mba + ala
	Awalan nomba-		
34	14	nomba...	nomba - ... nombairaka => nomba + airaka
	Awalan nosi-		
35	15	nosi...	nosi - ... nosimpu => nosi + umpu
	Awalan mosi-		
36	16	mosi...	mosi - ... mosikande => mosi + kande
	Awalan meti-		
37	17	meti...	meti - ... metigila => meti + gila
	Awalan neti-		
38	18	neti...	neti - ... netingu => neti + ingu
	Awalan ngga-		
39	19	ngga...	ngga - ... nggaupa => ngga + upu
	Awalan paka-		
40	20	paka...	paka - ... pakaturu => paka + turu
	Awalan naka-		
41	21	naka...	naka - ... nakadosa => naka + dosa

Tabel 4.2 (Lanjutan)

	Awalan maka-			
42	22	maka...	maka - ...	makadua => maka + dua
	Awalan popo-			
43	23	popo...	popo - ...	popoana => popo + ana
	Awalan rape-			
44	24	rape...	rape - ...	rapetuda => rape + tuda
	Awalan nipe-			
45	25	nipe...	nipe - ...	nipesua => nipe + sua
	Awalan nipari-			
46	26	nipari...	nipari - ...	niparipuri => nipari + puri
	Awalan pari-			
47	27	pari...	pari - ...	parikanto => pari + kanto
	Awalan nombapaka-			
48	28	nombapaka...	nombapaka - ...	nombapakadua => nombapaka + dua
	Awalan nompaka-			
49	29	nompaka...	nompaka - ...	nompakambela => nompaka + mbela
	Awalan mombapaka-			
50	30	mombapaka...	mombapaka - ...	mombapakamate => mombapaka + mate
	Awalan mompaka-			
51	31	mompaka...	mompaka - ...	mompakambela => mompaka + mbela
	Awalan rapaka-			
52	32	rapaka...	rapaka - ...	rapakadua => rapaka + dua
	Awalan nipaka-			
53	33	nipaka...	nipaka - ...	nipakadua => nipaka + dua
	Awalan nosipaka-			
54	34	nosipaka...	nosipaka - ...	nosipakambela => nosipaka + mbela
	Awalan mosipaka-			
55	35	mosipaka...	mosipaka - ...	mosipakadua => mosipaka + dua
	Awalan nosipari-			
56	36	nosipari...	nosipari - ...	nosiparipuri => nosipari + puri
	Awalan mosipari-			
57	37	mosipari...	mosipari - ...	mosiparipuri => mosipari + puri
	Awalan nompari-			
58	38	nompari...	nompari - ...	nomparikanto => nompari + kanto
	Awalan mompari-			
59	39	mompari...	mompari - ...	momparipuri => mompari + puri
	Awalan momposi-			
60	40	momposi...	momposi - ...	momposiala => momposi + ala
	Awalan momposi-			
61	41	momposi...	momposi - ...	momposikande => momposi + kande

Tabel 4.2 (Lanjutan)

	Awalan nosipopo-		
62	42	nosipopo...	nosipopo - ... nosipoponturo => nosipopo + nturo
	Awalan mosipopo-		
63	43	mosipopo...	mosipopo - ... mosipopodava => mosipopo + dava
	Awalan nipopo-		
64	44	nipopo...	nipopo - ... nipopoberet => nipopo + beret
	Awalan sangga-		
65	45	sangga...	sangga - ... sanggapande => sangga + pande
	Awalan to-		
66	46	tu...	tu - ... tofare => to + fare
	Awalan pe-		
67	47	pe...	pe - ... petevai => pe + tevai
	Awalan posi-		
68	48	posi...	posi - ... posilumba => posi + lumba
	Awalan topo-		
69	49	topo...	topo - ... topomore => topo + more
	Awalan mpo-		
70	50	mpo...	mpo - ... mpobumba => mpo + bumba
	Sinipan - in -		
71	51	...in...	...in - ... inole => s + in + ole => sole
	Sinipan - um -		
72	52	...um...	...um - ... tumangit => t + um + angit => tangit
	Sinipan - imb -		
73	53	...imb...	...imb - ... imbala => t + imb + ala => tala
	Sinipan - il -		
74	54	...il...	...il - ... hilau => h + il + au => hau
	Akhiran -a		
75	55	...a	...a - ... kanda => kande + a
	Akhiran -i		
76	56	...i	...i - ... itritai => itrita + i
	Akhiran -si		
77	57	...si	...si - ... duasi => dua + si
	Akhiran -ti		
78	58	...ti	...ti - ... kuniti => kuni + ti
	Akhiran -pa		
79	59	...pa	...pa - ... inupa => inu + pa
	Akhiran -mo		
80	60	...mo	...mo - ... aginamo => agina + mo
	Apat pa--a		
81	61	pa...a	pa - ... a paturua => pa + uru + a

Tabel 4.2 (Lanjutan)

	Apti ma- -i			
82	62	man...i	man-...-i	mantalingai => man + talinga + i
	Apti na- -i			
83	63	nan...i	nan-...-i	nantalingai => nan + talinga + i
	Apti nomba- -i			
84	64	nomba...i	nomba-...-i	nombavatu => nomba + vatu + i
	Apti momba- -i			
85	65	momba...i	momba-...-i	mombavatu => momba + vatu + i
	Apti nomba- -si			
86	66	nomba...si	nomba-...-si	nombaduasi => nomba + dua + si
	Apti momba- -si			
87	67	momba...si	momba-...-si	mombaveulasi => momba + veulu + si
	Apti nomba- -ti			
88	68	nomba...ti	nomba-...-ti	nombabuyati => nomba + buya + ti
	Apti momba- -ti			
89	69	momba...ti	momba-...-ti	mombabuyati => momba + buya + ti
	Apti na- -ka			
90	70	nang...ka	nang-...-ka	nangulika => nang + uli + ka
	Apti ma- -ka			
91	71	mang...ka	mang-...-ka	mangulika => mang + uli + ka
	Apti nosi- -si			
92	72	nosi...si	nosi-...-si	nosidiasi => nosi + dua + si
	Apti mosi- -si			
93	73	mo...si	mo-...-si	mosidiasi => mosi + dua + si
	Apti nomba- -ka			
94	74	nomba...ka	nomba-...-ka	nombakenika => nomba + keni + ka
	Apti momba- -ka			
95	75	momba...ka	momba-...-ka	mombakenika => momba + keni + ka
	Apti nosi- -ka			
96	76	nosi...ka	nosi-...-ka	nosikenika => nosi + keni + ka
	Apti mosi- -ka			
97	77	mosi...ka	mosi-...-ka	mosikenika => mosi + keni + ka
	Apti nomba- -raka			
98	78	nomba...raka	nomba-...-raka	nombasuaraka => nomba + sua + raka
	Apti momba- -raka			
99	79	momba...raka	momba-...-raka	mombasuaraka => momba + sua + raka
	Apti nomba- -taka			
100	80	nomba...taka	nomba-...-taka	nombarempetaka => nomba + rempe + taka

Tabel 4.2 (Lanjutan)

	Apti momba- -taka			
101	81	momba ... taka	momba- ... -taka	mombanpetaka => momba + rempe + taka
	Apti nomba- -saka			
102	82	nomba ... saka	nomba- ... -saka	nombanavusaka => nomba + navu + saka
	Apti momba- -saka			
103	83	momba ... saka	momba- ... - saka	mombanavusaka => momba + navu + saka
	Apti nosi- -raka			
104	84	nosi ... raka	nosi- ... -raka	nosisaraka => nosi + sua + raka
	Apti mosi- -raka			
105	85	mosi ... raka	mosi- ... -raka	mosisaraka => mosi + sua + raka
	Apti nosi- -taka			
106	86	nosi ... taka	nosi- ... -taka	nosirempetaka => nosi + rempe + taka
	Apti mosi- -taka			
107	87	mosi ... taka	mosi- ... -taka	mosirempetaka => mosi + rempe + taka
	Apti nosi- -saka			
108	88	nosi ... saka	nosi- ... -saka	nosinavusaka => nosi + navu + saka
	Apti mosi- -saka			
109	89	mosi ... saka	mosi- ... -saka	mosinavusaka => mosi + navu + saka
	Apti pe- -a			
110	90	pe ... a	pe- ... -a	pevava => pe + vava + a
	Apti pe- -na			
111	91	pe ... na	pe- ... -na	peambana => pe + amba + na
	Apti ka- -a			
112	92	ka ... a	ka- ... -a	kalengea => ka + lenge + a
	Apti ka- -na			
113	93	ka ... na	ka- ... -na	kaalirina => ka + alirna + na

Berikutnya disusun perintah dari masing-masing aturan imbuhan yang dapat digunakan untuk menemukan imbuhan pada sebuah kata. Untuk imbuhan awalan, perintah disusun dengan menghitung jumlah kata imbuhan yang dicek kemudian kata yang akan diperiksa dipotong dari sebelah kiri sebanyak jumlah kata awalan. Selanjutnya hasil potongan kata dibandingkan dengan kata imbuhan, jika cocok maka imbuhan awalan telah ditemukan. Algoritma pengecekan awalan adalah sebagai berikut.



Berikut contoh perintah pengecekan salah satu imbuhan awalan.

```

function TForm1.cekawalno(kata: string): boolean;
const jml=2;
const awl='no';
begin
  Result:= False;
  kataawalan:='';
  if LeftStr(kata,jml)=awl then
  begin
    kataawalan:= awl;
    jmlawalan:=jml;
    Result:= True;
  end;
end;
  
```

Gambar 4.2 Pengecekan imbuhan awalan 'no'

Pada pengecekan imbuhan akhiran, langkah yang digunakan hampir sama dengan pengecekan awalan, perbedaannya pemotongan kata yang diperiksa dilakukan dari sebelah kanan. Jika hasil potongan kata yang dibandingkan dengan kata imbuhan cocok maka imbuhan akhiran telah ditemukan. Algoritma pengecekan akhiran adalah sebagai berikut.



Gambar 4.3 Algoritma Pengecekan akhiran

Berikut contoh perintah pengecekan salah satu imbuhan akhiran.

```

function TForm1.ccekakhirpa(kata: string): boolean;
const jml=2;
const akr='pa';
begin
  Result:= False;
  kataakhiran:='';
  if RightStr(kata,jml)=akr then
  begin
    kataakhiran:= akr;
    jmlakhiran:=jml;
    Result:= True;
  end;
end;
  
```

Gambar 4.4 Pengecekan imbuhan akhiran 'pa'

Untuk pengecekan imbuhan apit memanfaatkan fungsi pengecekan imbuhan awalan dan imbuhan akhiran. Jika hasil pengecekan imbuhan awalan dan akhiran yang dilakukan terhadap kata yang diperiksa ditemukan secara bersamaan, maka imbuhan apit ditemukan pada kata yang diperiksa. Algoritma pengecekan imbuhan apit adalah sebagai berikut.



Gambar 4.5 Algoritma pengecekan imbuhan apit

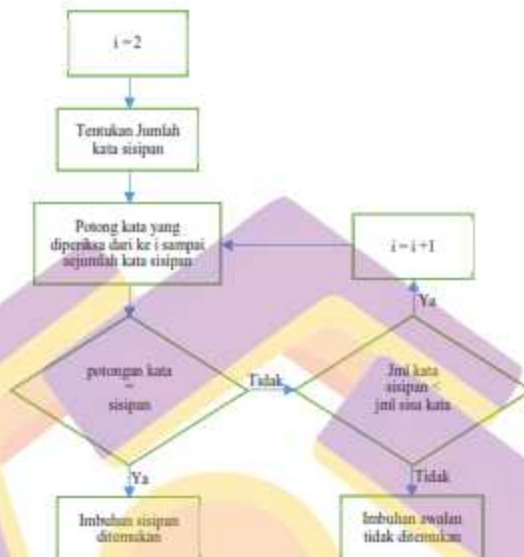
Berikut contoh perintah pengecekan salah satu imbuhan apit.

```

function TForm1.cekapitnomba_taka(kata: string): boolean;
begin
  Result:= False;
  if(cekawalnomba(kata) and(cekakhirtaka(kata)) then
    Result:= True;
end;
  
```

Gambar 4.6 Pengecekan imbuhan apit 'nomba ... taka'

Selanjutnya untuk pengecekan imbuhan sisipan dilakukan dengan memotong bagian tengah dari kata yang diperiksa dimulai dari huruf kedua sebanyak jumlah kata imbuhan. Kemudian hasil potongan kata dibandingkan dengan kata imbuhan, jika sama maka telah ditemukan imbuhan sisipan pada kata tersebut. Jika tidak cocok maka langkah akan diulangi pada huruf berikutnya sampai jumlah huruf yang tersisa lebih kecil dari jumlah huruf imbuhan yang diperiksa. Algoritma pengecekan imbuhan sisipan adalah sebagai berikut.



Gambar 4. 7 Algoritma pengecekan imbuhan sisipan

Berikut contoh perintah pengecekan salah satu imbuhan sisipan.

```

function TForm1.ceksisip_imb(kata: string): boolean;
const jml=3;
const sisip='imb';
var i: Integer;
begin
  Result:= False;
  katasisipan:='';
  for i:= 2 to (Length(kata)-jml+1) do
  begin
    if MidStr(kata,i,jml)=sisip then
    begin
      katasisipan:=sisip;
      jmlsisipan:=jml;
      Result:=True;
      awalsisipan:=i;
      akhirsisipan:=i+jml-1;
      Exit;
    end;
  end;
end;

```

Gambar 4.8 Pengecekan imbuhan sisipan 'imb'

Selain pemeriksaan kata imbuhan, pemeriksaan kata dasar juga perlu dilakukan dengan mengikuti ketentuan dari aturan yang telah disusun. Pada pengecekan kata dasar, jika pada kata yang diperiksa ditemukan awalan 'manj', 'nanj', 'panj' atau 'sanj' dengan diikuti huruf vokal maka kata tersebut ditambahkan huruf 's'. Tetapi jika awalan 'sangg' yang diikuti huruf vokal, maka ditambahkan huruf 'k'. Berikut perintah pengecekan kata dasar.

```
function TForm1.CekKataDasar(kata: string): Boolean;
var X: Integer;
    hurufawal: Char;
    cekkata: String;
begin
    Result := False;
    cekkata := LowerCase(kata);
    if ((kataakhiran='') and ((kataawalan='manj') or (kataawalan='nanj')
    or (kataawalan='panj') or (kataawalan='sanj'))) then
    begin
        hurufawal := cekkata[1];
        if (hurufawal in ['a','i','u','e','o']) then
            cekkata := 's'+cekkata;
    end;

    if ((kataakhiran='') and (kataawalan='sangg')) then
    begin
        hurufawal := cekkata[1];
        if (hurufawal in ['a','i','u','e','o']) then
            cekkata := 'k'+cekkata;
    end;

    for X := 0 to S.Count-1 do
    begin
        if S[X] = cekkata then
        begin
            katadasar := S[X];
            Result := True;
            exit;
        end;
    end;
end;
```

Gambar 4.9 Pengecekan kata dasar

Kemudian berdasarkan aturan yang telah disusun, dilakukan modifikasi algoritma Nazief dan Adriani agar dapat melakukan stemming bahasa Kaili – Ledo. Modifikasi algoritma dapat disusun sebagai berikut:

Langkah 1: Lakukan pencarian pada daftar kata dasar jika ditemukan maka kata tersebut dianggap sebagai kata dasar dan algoritma dihentikan, jika tidak lanjutkan ke langkah 2

Langkah 2: Cek imbuhan sisipan dengan menghilangkan partikel berikut pada tengah kata

1. "imb",
2. "in",
3. "um",
4. "il"

Kemudian dari kata yang dihasilkan dari masing-masing proses penghilangan partikel, lakukan pencarian pada daftar kata dasar jika ditemukan maka kata tersebut dianggap sebagai kata dasar dan algoritma dihentikan, jika tidak lanjutkan ke langkah berikutnya. Jika seluruh langkah telah dilakukan dan kata dasar belum ditemukan lanjutkan pada langkah 3

Langkah 3: Cek imbuhan apit dengan menghilangkan partikel dengan urutan berikut:

1. "nomba" pada awal kata dan "raka" pada akhir kata,
2. "momba" pada awal kata dan "raka" pada akhir kata,
3. "nomba" pada awal kata dan "taka" pada akhir kata,
4. "momba" pada awal kata dan "taka" pada akhir kata,
5. "nomba" pada awal kata dan "saka" pada akhir kata,
6. "momba" pada awal kata dan "saka" pada akhir kata,
7. "nosi" pada awal kata dan "raka" pada akhir kata,
8. "mosi" pada awal kata dan "raka" pada akhir kata,
9. "nosi" pada awal kata dan "taka" pada akhir kata,
10. "mosi" pada awal kata dan "taka" pada akhir kata,

11. "nosi" pada awal kata dan "saka" pada akhir kata,
12. "mosi" pada awal kata dan "saka" pada akhir kata,
13. "nomba" pada awal kata dan "si" pada akhir kata,
14. "momba" pada awal kata dan "si" pada akhir kata,
15. "nomba" pada awal kata dan "ti" pada akhir kata,
16. "momba" pada awal kata dan "ti" pada akhir kata,
17. "nomba" pada awal kata dan "ka" pada akhir kata,
18. "momba" pada awal kata dan "ka" pada akhir kata,
19. "nomba" pada awal kata dan "i" pada akhir kata,
20. "momba" pada awal kata dan "i" pada akhir kata,
21. "nang" pada awal kata dan "ka" pada akhir kata,
22. "mang" pada awal kata dan "ka" pada akhir kata,
23. "nosi" pada awal kata dan "si" pada akhir kata,
24. "mosi" pada awal kata dan "si" pada akhir kata,
25. "nosi" pada awal kata dan "ka" pada akhir kata,
26. "mosi" pada awal kata dan "ka" pada akhir kata,
27. "mang" pada awal kata dan "i" pada akhir kata,
28. "nang" pada awal kata dan "i" pada akhir kata,
29. "man" pada awal kata dan "i" pada akhir kata,
30. "nan" pada awal kata dan "i" pada akhir kata,
31. "pe" pada awal kata dan "na" pada akhir kata,
32. "ka" pada awal kata dan "na" pada akhir kata,
33. "pa" pada awal kata dan "a" pada akhir kata,
34. "ma" pada awal kata dan "i" pada akhir kata,
35. "na" pada awal kata dan "i" pada akhir kata,
36. "pe" pada awal kata dan "a" pada akhir kata,
37. "ka" pada awal kata dan "a" pada akhir kata

Kemudian dari kata yang dihasilkan dari masing-masing proses penghilangan partikel, lakukan pencarian pada daftar kata dasar jika ditemukan maka kata tersebut dianggap sebagai kata dasar dan algoritma dihentikan, jika tidak lanjutkan ke langkah berikutnya. Jika

seluruh langkah telah dilakukan dan kata dasar belum ditemukan
 lanjutkan pada langkah 4

Langkah 4: Cek imbuhan awalan dengan menghilangkan partikel pada awal kata
 dengan urutan berikut:

1. "nombapaka"
2. "mombapaka"
3. "nosipopo"
4. "mosipopo"
5. "nosipari"
6. "mosipari"
7. "nosipaka"
8. "mosipaka"
9. "nomposi"
10. "momposi"
11. "nompari"
12. "mompari"
13. "nompaka"
14. "mompaka"
15. "rapaka"
16. "nipaka"
17. "sangga"
18. "nipopo"
19. "nipari"
20. "sangg", jika huruf berikutnya adalah huruf vocal maka
 ditambahkan huruf "k" pada awal kata yang dihasilkan
21. "momba"
22. "nomba"
23. "pari"
24. "nipe"
25. "rape"
26. "popo"

27. "maka"
28. "naka"
29. "paka"
30. "ngga"
31. "neti"
32. "meti"
33. "sanj", jika huruf berikutnya adalah huruf vocal maka ditambahkan huruf "s" pada awal kata yang dihasilkan
34. "panj", jika huruf berikutnya adalah huruf vocal maka ditambahkan huruf "s" pada awal kata yang dihasilkan
35. "pang"
36. "nanj", jika huruf berikutnya adalah huruf vocal maka ditambahkan huruf "s" pada awal kata yang dihasilkan
37. "manj", jika huruf berikutnya adalah huruf vocal maka ditambahkan huruf "s" pada awal kata yang dihasilkan
38. "nosi"
39. "mosi"
40. "posi"
41. "topo"
42. "mang"
43. "nang"
44. "man"
45. "nan"
46. "pan"
47. "mba"
48. "sam"
49. "san"
50. "pom"
51. "mom"
52. "mpo"
53. "ma"

54. "mo"
55. "po"
56. "sa"
57. "ra"
58. "no"
59. "na"
60. "ni"
61. "ne"
62. "me"
63. "pa"
64. "pe"
65. "ka"
66. "to"

Kemudian dari kata yang dihasilkan dari masing-masing proses penghilangan partikel, lakukan pencarian pada daftar kata dasar jika ditemukan maka kata tersebut dianggap sebagai kata dasar dan algoritma dihentikan, jika tidak lanjutkan ke langkah berikutnya. Jika seluruh langkah telah dilakukan dan kata dasar belum ditemukan lanjutkan pada langkah 5

Langkah 5: Cek imbuhan akhiran dengan menghilangkan partikel pada akhir kata dengan urutan berikut:

1. "pu",
2. "ti",
3. "si",
4. "mo",
5. "a",
6. "i"

Kemudian dari kata yang dihasilkan dari masing-masing proses penghilangan partikel, lakukan pencarian pada daftar kata dasar jika ditemukan maka kata tersebut dianggap sebagai kata dasar dan

algoritma dihentikan, jika tidak lanjutkan ke langkah berikutnya. Jika seluruh langkah telah dilakukan dan kata dasar belum ditemukan lanjutkan pada langkah 6

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Pada algoritma yang disusun, proses penghapusan partikel imbuhan dibagi dalam 4 kelompok berdasarkan jenis imbuhan yaitu awalan, akhiran, apit dan sisipan, sehingga urutan pemeriksaan imbuhan dapat disusun dengan 24 cara, yaitu:

Algoritma 1 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 2 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$

Langkah 2: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$((k - (B = \{z|z \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$

Langkah 3: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$((k - (B = \{z|z \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$

Langkah 4: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$((k - (B = \{z|z \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$

Langkah 5: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$((k - (B = \{z|z \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 3 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k \in (B = \{z|z \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k \in (B = \{z|z \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k \in (B = \{z|z \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k \in (B = \{z|z \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 4 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan sisipan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan awalan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan akhiran})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika

ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan apit})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap

sebagai kata dasar.

Algoritma 5 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan

kata tersebut adalah hasil stemming

$$(k \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan sisipan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika
ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap
sebagai kata dasar.

Algoritma 6 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan
kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan apit})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan awalan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap
sebagai kata dasar.

Algoritma 7 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan
kata tersebut adalah hasil stemming

$$(k \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika
ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan apit})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan sisipan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan awalan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}x \text{ imbuhan akhiran})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 8 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}x \text{ imbuhan apit})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}x \text{ imbuhan sisipan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}x \text{ imbuhan akhiran})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}x \text{ imbuhan awalan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 9 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 10 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika
ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap
sebagai kata dasar.

Algoritma 11 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan
kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika
ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 12 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan sisipan})) = \{y\}y \in k, y \notin B)) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan awalan})) = \{y\}y \in k, y \notin B)) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 13 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan awalan})) = \{y\}y \in k, y \notin B)) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan apit})) = \{y\}y \in k, y \notin B)) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan sisipan})) = \{y\}y \in k, y \notin B)) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan akhiran})) = \{y \mid y \in k, y \notin B\}) \in (A = \{x \mid x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 14 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan

kata tersebut adalah hasil stemming

$$(k \in (A = \{x \mid x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan awalan})) = \{y \mid y \in k, y \notin B\}) \in (A = \{x \mid x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika

ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan apit})) = \{y \mid y \in k, y \notin B\}) \in (A = \{x \mid x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan akhiran})) = \{y \mid y \in k, y \notin B\}) \in (A = \{x \mid x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan sisipan})) = \{y \mid y \in k, y \notin B\}) \in (A = \{x \mid x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap

sebagai kata dasar.

Algoritma 15 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 16 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika

ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap

sebagai kata dasar.

Algoritma 17 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan

kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan akhiran})) = \{y | y \in k, y \notin B\}) \in (A = \{x | x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan sisipan})) = \{y | y \in k, y \notin B\}) \in (A = \{x | x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika

ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan apit})) = \{y | y \in k, y \notin B\}) \in (A = \{x | x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap

sebagai kata dasar.

Algoritma 18 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan

kata tersebut adalah hasil stemming

$$(k \in (A = \{x | x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan awalan})) = \{y | y \in k, y \notin B\}) \in (A = \{x | x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\} \text{ imbuhan akhiran})) = \{y | y \in k, y \notin B\}) \in (A = \{x | x \text{ kata dasar}\}) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika
ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap
sebagai kata dasar.

Algoritma 19 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan
kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika
ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar
jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan sisipan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 20 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan

kata tersebut adalah hasil stemming

$$(k \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan akhiran})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika

ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan apit})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan sisipan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\}z \text{ imbuhan awalan})) = \{y\}y \in k, y \notin B) \in (A = \{x\}x \text{ kata dasar})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 21 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|z \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 22 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan sisipan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika

ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

Algoritma 23 :

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$(k \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan akhiran}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\mid z \text{ imbuhan sisipan}\}) = \{y\mid y \in k, y \notin B\}) \in (A = \{x\mid x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\mid z \text{ imbuhan awalan}\}) = \{y\mid y \in k, y \notin B\}) \in (A = \{x\mid x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika

ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\mid z \text{ imbuhan apit}\}) = \{y\mid y \in k, y \notin B\}) \in (A = \{x\mid x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap

sebagai kata dasar.

Algoritma 24 :

Kata Dasar \Rightarrow Akhiran \Rightarrow Sisipan \Rightarrow Apit \Rightarrow Awalan

Langkah 1: Lakukan pemeriksaan kata dasar jika ditemukan hentikan proses dan

kata tersebut adalah hasil stemming

$$(k \in (A = \{x\mid x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 2: Lakukan pemeriksaan imbuhan akhiran jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\mid z \text{ imbuhan akhiran}\}) = \{y\mid y \in k, y \notin B\}) \in (A = \{x\mid x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 3: Lakukan pemeriksaan imbuhan sisipan jika ditemukan cek kata dasar

jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z\mid z \text{ imbuhan sisipan}\}) = \{y\mid y \in k, y \notin B\}) \in (A = \{x\mid x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 4: Lakukan pemeriksaan imbuhan apit jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan apit}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 5: Lakukan pemeriksaan imbuhan awalan jika ditemukan cek kata dasar jika ditemukan hentikan proses dan kata tersebut adalah hasil stemming

$$((k - (B = \{z|x \text{ imbuhan awalan}\}) = \{y|y \in k, y \notin B\}) \in (A = \{x|x \text{ kata dasar}\})) \rightarrow (k = x)$$

Langkah 6: Kata awal yang belum distemming akan dikembalikan dan dianggap sebagai kata dasar.

4.3. Pengujian

Pada tahap ini dilakukan pengujian terhadap 24 alternatif algoritma yang disusun, untuk itu disiapkan data uji berupa 1000 kata dalam bahasa kaili sebagai bahan untuk melakukan pengujian. Data yang disiapkan terdiri atas kata beserta kata dasarnya. Setelah dilakukan proses stemming, hasilnya akan dibandingkan dengan kata dasar yang seharusnya untuk mengetahui hasil dari proses stemming telah benar atau salah.

Microsoft Excel - Pengujian Hasil Berhasil

Data Uji			Algoritma 1		Algoritma 2		Hasil Algoritma 1			
No.	Data Uji	Hasil Benar	Algoritma 1	Algoritma 2	No.	Data	Hasil Benar	Status	Status	Status
1	1000	1000	Algoritma 1	Algoritma 2	1	1000	1000	Benar	Benar	Benar
2	1000	1000	Algoritma 3	Algoritma 4	2	1000	1000	Benar	Benar	Benar
3	1000	1000	Algoritma 5	Algoritma 6	3	1000	1000	Benar	Benar	Benar
4	1000	1000	Algoritma 7	Algoritma 8	4	1000	1000	Benar	Benar	Benar
5	1000	1000	Algoritma 9	Algoritma 10	5	1000	1000	Benar	Benar	Benar
6	1000	1000	Algoritma 11	Algoritma 12	6	1000	1000	Benar	Benar	Benar
7	1000	1000	Algoritma 13	Algoritma 14	7	1000	1000	Benar	Benar	Benar
8	1000	1000	Algoritma 15	Algoritma 16	8	1000	1000	Benar	Benar	Benar
9	1000	1000	Algoritma 17	Algoritma 18	9	1000	1000	Benar	Benar	Benar
10	1000	1000	Algoritma 19	Algoritma 20	10	1000	1000	Benar	Benar	Benar
11	1000	1000	Algoritma 21	Algoritma 22	11	1000	1000	Benar	Benar	Benar
12	1000	1000	Algoritma 23	Algoritma 24	12	1000	1000	Benar	Benar	Benar
13	1000	1000			13	1000	1000	Benar	Benar	Benar
14	1000	1000			14	1000	1000	Benar	Benar	Benar
15	1000	1000			15	1000	1000	Benar	Benar	Benar
16	1000	1000			16	1000	1000	Benar	Benar	Benar
17	1000	1000			17	1000	1000	Benar	Benar	Benar
18	1000	1000			18	1000	1000	Benar	Benar	Benar
19	1000	1000			19	1000	1000	Benar	Benar	Benar
20	1000	1000			20	1000	1000	Benar	Benar	Benar
21	1000	1000			21	1000	1000	Benar	Benar	Benar
22	1000	1000			22	1000	1000	Benar	Benar	Benar
23	1000	1000			23	1000	1000	Benar	Benar	Benar
24	1000	1000			24	1000	1000	Benar	Benar	Benar
25	1000	1000			25	1000	1000	Benar	Benar	Benar
26	1000	1000			26	1000	1000	Benar	Benar	Benar
27	1000	1000			27	1000	1000	Benar	Benar	Benar
28	1000	1000			28	1000	1000	Benar	Benar	Benar
29	1000	1000			29	1000	1000	Benar	Benar	Benar
30	1000	1000			30	1000	1000	Benar	Benar	Benar

Jumlah Data : 300
Jumlah Data Benar : 284
Jumlah Data Salah : 16

Persentase Data Benar : 94,7%
Persentase Data Salah : 5,3%

Gambar 4.10 Pengujian Algoritma 1

Pengujian yang dilakukan menggunakan algoritma 1 menghasilkan 945 data benar dan 55 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 94,5%

Microsoft Excel - Pengujian Hasil Berhasil

Data Uji			Algoritma 1		Algoritma 2		Hasil Algoritma 2			
No.	Data Uji	Hasil Benar	Algoritma 1	Algoritma 2	No.	Data	Hasil Benar	Status	Status	Status
1	1000	1000	Algoritma 1	Algoritma 2	1	1000	1000	Benar	Benar	Benar
2	1000	1000	Algoritma 3	Algoritma 4	2	1000	1000	Benar	Benar	Benar
3	1000	1000	Algoritma 5	Algoritma 6	3	1000	1000	Benar	Benar	Benar
4	1000	1000	Algoritma 7	Algoritma 8	4	1000	1000	Benar	Benar	Benar
5	1000	1000	Algoritma 9	Algoritma 10	5	1000	1000	Benar	Benar	Benar
6	1000	1000	Algoritma 11	Algoritma 12	6	1000	1000	Benar	Benar	Benar
7	1000	1000	Algoritma 13	Algoritma 14	7	1000	1000	Benar	Benar	Benar
8	1000	1000	Algoritma 15	Algoritma 16	8	1000	1000	Benar	Benar	Benar
9	1000	1000	Algoritma 17	Algoritma 18	9	1000	1000	Benar	Benar	Benar
10	1000	1000	Algoritma 19	Algoritma 20	10	1000	1000	Benar	Benar	Benar
11	1000	1000	Algoritma 21	Algoritma 22	11	1000	1000	Benar	Benar	Benar
12	1000	1000	Algoritma 23	Algoritma 24	12	1000	1000	Benar	Benar	Benar
13	1000	1000			13	1000	1000	Benar	Benar	Benar
14	1000	1000			14	1000	1000	Benar	Benar	Benar
15	1000	1000			15	1000	1000	Benar	Benar	Benar
16	1000	1000			16	1000	1000	Benar	Benar	Benar
17	1000	1000			17	1000	1000	Benar	Benar	Benar
18	1000	1000			18	1000	1000	Benar	Benar	Benar
19	1000	1000			19	1000	1000	Benar	Benar	Benar
20	1000	1000			20	1000	1000	Benar	Benar	Benar
21	1000	1000			21	1000	1000	Benar	Benar	Benar
22	1000	1000			22	1000	1000	Benar	Benar	Benar
23	1000	1000			23	1000	1000	Benar	Benar	Benar
24	1000	1000			24	1000	1000	Benar	Benar	Benar
25	1000	1000			25	1000	1000	Benar	Benar	Benar
26	1000	1000			26	1000	1000	Benar	Benar	Benar
27	1000	1000			27	1000	1000	Benar	Benar	Benar
28	1000	1000			28	1000	1000	Benar	Benar	Benar
29	1000	1000			29	1000	1000	Benar	Benar	Benar
30	1000	1000			30	1000	1000	Benar	Benar	Benar

Jumlah Data : 300
Jumlah Data Benar : 284
Jumlah Data Salah : 16

Persentase Data Benar : 94,7%
Persentase Data Salah : 5,3%

Summary of data from the 'Hasil Uji' table in Gambar 4.13:

No.	Algoritma	Benar	Salah	Tingkat Akurasi
1	Algoritma 1	933	67	93.3%
2	Algoritma 2	933	67	93.3%
3	Algoritma 3	933	67	93.3%
4	Algoritma 4	953	47	95.3%
5	Algoritma 5	933	67	93.3%
6	Algoritma 6	933	67	93.3%
7	Algoritma 7	933	67	93.3%
8	Algoritma 8	933	67	93.3%
9	Algoritma 9	933	67	93.3%
10	Algoritma 10	933	67	93.3%
11	Algoritma 11	933	67	93.3%
12	Algoritma 12	933	67	93.3%
13	Algoritma 13	933	67	93.3%
14	Algoritma 14	933	67	93.3%
15	Algoritma 15	933	67	93.3%
16	Algoritma 16	933	67	93.3%
17	Algoritma 17	933	67	93.3%
18	Algoritma 18	933	67	93.3%
19	Algoritma 19	933	67	93.3%
20	Algoritma 20	933	67	93.3%
21	Algoritma 21	933	67	93.3%
22	Algoritma 22	933	67	93.3%
23	Algoritma 23	933	67	93.3%
24	Algoritma 24	933	67	93.3%

Gambar 4.13 Pengujian Algoritma 4

Pengujian yang dilakukan menggunakan algoritma 4 menghasilkan 953 data benar dan 47 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 95,3%

Summary of data from the 'Hasil Uji' table in Gambar 4.14:

No.	Algoritma	Benar	Salah	Tingkat Akurasi
1	Algoritma 1	933	67	93.3%
2	Algoritma 2	933	67	93.3%
3	Algoritma 3	933	67	93.3%
4	Algoritma 4	953	47	95.3%
5	Algoritma 5	933	67	93.3%
6	Algoritma 6	933	67	93.3%
7	Algoritma 7	933	67	93.3%
8	Algoritma 8	933	67	93.3%
9	Algoritma 9	933	67	93.3%
10	Algoritma 10	933	67	93.3%
11	Algoritma 11	933	67	93.3%
12	Algoritma 12	933	67	93.3%
13	Algoritma 13	933	67	93.3%
14	Algoritma 14	933	67	93.3%
15	Algoritma 15	933	67	93.3%
16	Algoritma 16	933	67	93.3%
17	Algoritma 17	933	67	93.3%
18	Algoritma 18	933	67	93.3%
19	Algoritma 19	933	67	93.3%
20	Algoritma 20	933	67	93.3%
21	Algoritma 21	933	67	93.3%
22	Algoritma 22	933	67	93.3%
23	Algoritma 23	933	67	93.3%
24	Algoritma 24	933	67	93.3%

Gambar 4.14 Pengujian Algoritma 5

Pengujian yang dilakukan menggunakan algoritma 5 menghasilkan 946 data benar dan 54 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 94,6%

No	Nama	Eksa Dasar	Hasil	Skor	Status
1	algoritma 1	algoritma 1	Benar	1000	Benar
2	algoritma 2	algoritma 2	Benar	1000	Benar
3	algoritma 3	algoritma 3	Benar	1000	Benar
4	algoritma 4	algoritma 4	Benar	1000	Benar
5	algoritma 5	algoritma 5	Benar	1000	Benar
6	algoritma 6	algoritma 6	Salah	946	Salah
7	algoritma 7	algoritma 7	Benar	1000	Benar
8	algoritma 8	algoritma 8	Benar	1000	Benar
9	algoritma 9	algoritma 9	Benar	1000	Benar
10	algoritma 10	algoritma 10	Benar	1000	Benar
11	algoritma 11	algoritma 11	Benar	1000	Benar
12	algoritma 12	algoritma 12	Benar	1000	Benar
13	algoritma 13	algoritma 13	Benar	1000	Benar
14	algoritma 14	algoritma 14	Benar	1000	Benar
15	algoritma 15	algoritma 15	Benar	1000	Benar
16	algoritma 16	algoritma 16	Benar	1000	Benar
17	algoritma 17	algoritma 17	Benar	1000	Benar
18	algoritma 18	algoritma 18	Benar	1000	Benar
19	algoritma 19	algoritma 19	Benar	1000	Benar
20	algoritma 20	algoritma 20	Benar	1000	Benar
21	algoritma 21	algoritma 21	Benar	1000	Benar
22	algoritma 22	algoritma 22	Benar	1000	Benar
23	algoritma 23	algoritma 23	Benar	1000	Benar
24	algoritma 24	algoritma 24	Benar	1000	Benar

Gambar 4.15 Pengujian Algoritma 6

Pengujian yang dilakukan menggunakan algoritma 6 menghasilkan 934 data benar dan 66 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 93,4%

Monitoring Sistem No. 1

Pengujian Data Berantakan

Data Uji			Algoritma 1	Algoritma 2	Hasil Algoritma 1				
No.	Data Uji	Hasil Uji	Algoritma 3	Algoritma 4	No.	Data	Hasil	Stabilitas	Status
1	10000	10000	Algoritma 5	Algoritma 6	1	10000	10000	100%	Benar
2	10000	10000	Algoritma 7	Algoritma 8	2	10000	10000	100%	Benar
3	10000	10000	Algoritma 9	Algoritma 10	3	10000	10000	100%	Benar
4	10000	10000	Algoritma 11	Algoritma 12	4	10000	10000	100%	Benar
5	10000	10000	Algoritma 13	Algoritma 14	5	10000	10000	100%	Benar
6	10000	10000	Algoritma 15	Algoritma 16	6	10000	10000	100%	Benar
7	10000	10000	Algoritma 17	Algoritma 18	7	10000	10000	100%	Benar
8	10000	10000	Algoritma 19	Algoritma 20	8	10000	10000	100%	Benar
9	10000	10000	Algoritma 21	Algoritma 22	9	10000	10000	100%	Benar
10	10000	10000	Algoritma 23	Algoritma 24	10	10000	10000	100%	Benar

Jumlah Data : 1000
 Jumlah Data Benar : 945
 Jumlah Data Salah : 55

Persentase Data Benar : 94,5%
 Persentase Data Salah : 5,5%

Gambar 4.16 Pengujian Algoritma 7

Pengujian yang dilakukan menggunakan algoritma 7 menghasilkan 945 data benar dan 55 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 94,5%

Monitoring Sistem No. 1

Pengujian Data Berantakan

Data Uji			Algoritma 1	Algoritma 2	Hasil Algoritma 2				
No.	Data Uji	Hasil Uji	Algoritma 3	Algoritma 4	No.	Data	Hasil	Stabilitas	Status
1	10000	10000	Algoritma 5	Algoritma 6	1	10000	10000	100%	Benar
2	10000	10000	Algoritma 7	Algoritma 8	2	10000	10000	100%	Benar
3	10000	10000	Algoritma 9	Algoritma 10	3	10000	10000	100%	Benar
4	10000	10000	Algoritma 11	Algoritma 12	4	10000	10000	100%	Benar
5	10000	10000	Algoritma 13	Algoritma 14	5	10000	10000	100%	Benar
6	10000	10000	Algoritma 15	Algoritma 16	6	10000	10000	100%	Benar
7	10000	10000	Algoritma 17	Algoritma 18	7	10000	10000	100%	Benar
8	10000	10000	Algoritma 19	Algoritma 20	8	10000	10000	100%	Benar
9	10000	10000	Algoritma 21	Algoritma 22	9	10000	10000	100%	Benar
10	10000	10000	Algoritma 23	Algoritma 24	10	10000	10000	100%	Benar

Jumlah Data : 1000
 Jumlah Data Benar : 954
 Jumlah Data Salah : 46

Persentase Data Benar : 95,4%
 Persentase Data Salah : 4,6%

Gambar 4.17 Pengujian Algoritma 8

Pengujian yang dilakukan menggunakan algoritma 8 menghasilkan 934 data benar dan 66 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 93,4%

No	Data	Error Data	Hasil	Skor	Status
1	1000	1000	1000	1000	Benar
2	1000	1000	1000	1000	Benar
3	1000	1000	1000	1000	Benar
4	1000	1000	1000	1000	Benar
5	1000	1000	1000	1000	Benar
6	1000	1000	1000	1000	Benar
7	1000	1000	1000	1000	Benar
8	1000	1000	1000	1000	Benar
9	1000	1000	1000	1000	Benar
10	1000	1000	1000	1000	Benar
11	1000	1000	1000	1000	Benar
12	1000	1000	1000	1000	Benar
13	1000	1000	1000	1000	Benar
14	1000	1000	1000	1000	Benar
15	1000	1000	1000	1000	Benar
16	1000	1000	1000	1000	Benar
17	1000	1000	1000	1000	Benar
18	1000	1000	1000	1000	Benar
19	1000	1000	1000	1000	Benar
20	1000	1000	1000	1000	Benar
21	1000	1000	1000	1000	Benar
22	1000	1000	1000	1000	Benar
23	1000	1000	1000	1000	Benar
24	1000	1000	1000	1000	Benar

Gambar 4.18 Pengujian Algoritma 9

Pengujian yang dilakukan menggunakan algoritma 9 menghasilkan 950 data benar dan 50 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 95%

Testing Algoritma 10

Data Asli

No	Daftar Kata	Kata Asli
1	berita	berita
2	berita	berita
3	berita	berita
4	berita	berita
5	berita	berita
6	berita	berita
7	berita	berita
8	berita	berita
9	berita	berita
10	berita	berita
11	berita	berita
12	berita	berita
13	berita	berita
14	berita	berita
15	berita	berita
16	berita	berita
17	berita	berita
18	berita	berita
19	berita	berita
20	berita	berita
21	berita	berita
22	berita	berita
23	berita	berita

Algoritma 1 - Algoritma 23

Hasil Algoritma 10

No	Kata	Kata Asli	Hasil Memanggil (Benar)
1	berita	berita	Benar
2	berita	berita	Benar
3	berita	berita	Benar
4	berita	berita	Benar
5	berita	berita	Benar
6	berita	berita	Benar
7	berita	berita	Benar
8	berita	berita	Benar
9	berita	berita	Benar
10	berita	berita	Benar
11	berita	berita	Benar
12	berita	berita	Benar
13	berita	berita	Benar
14	berita	berita	Benar
15	berita	berita	Benar
16	berita	berita	Benar
17	berita	berita	Benar
18	berita	berita	Benar
19	berita	berita	Benar
20	berita	berita	Benar
21	berita	berita	Benar
22	berita	berita	Benar
23	berita	berita	Benar

Jumlah Data : 1000
Jumlah Data Benar : 954
Jumlah Data Salah : 46
Persentase Data Benar : 95,4%
Persentase Data Salah : 4,6%

Gambar 4.19 Pengujian Algoritma 10

Pengujian yang dilakukan menggunakan algoritma 10 menghasilkan 954 data benar dan 46 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 95,4%

Testing Algoritma 11

Data Asli

No	Daftar Kata	Kata Asli
1	berita	berita
2	berita	berita
3	berita	berita
4	berita	berita
5	berita	berita
6	berita	berita
7	berita	berita
8	berita	berita
9	berita	berita
10	berita	berita
11	berita	berita
12	berita	berita
13	berita	berita
14	berita	berita
15	berita	berita
16	berita	berita
17	berita	berita
18	berita	berita
19	berita	berita
20	berita	berita
21	berita	berita
22	berita	berita
23	berita	berita

Algoritma 1 - Algoritma 23

Hasil Algoritma 11

No	Kata	Kata Asli	Hasil Memanggil (Benar)
1	berita	berita	Benar
2	berita	berita	Benar
3	berita	berita	Benar
4	berita	berita	Benar
5	berita	berita	Benar
6	berita	berita	Benar
7	berita	berita	Benar
8	berita	berita	Benar
9	berita	berita	Benar
10	berita	berita	Benar
11	berita	berita	Benar
12	berita	berita	Benar
13	berita	berita	Benar
14	berita	berita	Benar
15	berita	berita	Benar
16	berita	berita	Benar
17	berita	berita	Benar
18	berita	berita	Benar
19	berita	berita	Benar
20	berita	berita	Benar
21	berita	berita	Benar
22	berita	berita	Benar
23	berita	berita	Benar

Jumlah Data : 1000
Jumlah Data Benar : 942
Jumlah Data Salah : 58
Persentase Data Benar : 94,2%
Persentase Data Salah : 5,8%

Gambar 4.20 Pengujian Algorithm 11

Pengujian yang dilakukan menggunakan algoritma 11 menghasilkan 943 data benar dan 57 data salah dari 1000 data yang diuji sehingga tingkat akurasiya mencapai 94,3%

Algoritma 1	Algoritma 2	Hasil Aplikasi 12					
Algoritma 3	Algoritma 4	No.	Kata	Exit Case	Hasil	Stemming	Status
Algoritma 5	Algoritma 6	1	terasa	terasa	terasa	terasa	Benar
Algoritma 7	Algoritma 8	2	terasa	terasa	terasa	terasa	Benar
Algoritma 9	Algoritma 10	3	terasa	terasa	terasa	terasa	Benar
Algoritma 11	Algoritma 12	4	terasa	terasa	terasa	terasa	Benar
Algoritma 13	Algoritma 14	5	terasa	terasa	terasa	terasa	Benar
Algoritma 15	Algoritma 16	6	terasa	terasa	terasa	terasa	Benar
Algoritma 17	Algoritma 18	7	terasa	terasa	terasa	terasa	Benar
Algoritma 19	Algoritma 20	8	terasa	terasa	terasa	terasa	Benar
Algoritma 21	Algoritma 22	9	terasa	terasa	terasa	terasa	Benar
Algoritma 23	Algoritma 24	10	terasa	terasa	terasa	terasa	Benar
		Jumlah Data Benar: 943		Jumlah Data Salah: 57		Jumlah Data Benar: 943%	
		Jumlah Data Benar: 943		Jumlah Data Salah: 57		Jumlah Data Benar: 94,3%	

Gambar 4.21 Pengujian Algorithm 12

Pengujian yang dilakukan menggunakan algoritma 12 menghasilkan 938 data benar dan 62 data salah dari 1000 data yang diuji sehingga tingkat akurasiya mencapai 93,8%

Daftar Data

No.	Kata Kunci	Kata Kunci
1	lemba	lemba
2	lemba	lemba
3	lemba	lemba
4	lemba	lemba
5	lemba	lemba
6	lemba	lemba
7	lemba	lemba
8	lemba	lemba
9	lemba	lemba
10	lemba	lemba
11	lemba	lemba
12	lemba	lemba
13	lemba	lemba
14	lemba	lemba
15	lemba	lemba
16	lemba	lemba
17	lemba	lemba
18	lemba	lemba
19	lemba	lemba

Hasil Pengujian

No.	Kata Kunci	Hasil	Memang Sederhana
1	lemba	lemba	Benar
2	lemba	lemba	Benar
3	lemba	lemba	Benar
4	lemba	lemba	Benar
5	lemba	lemba	Benar
6	lemba	lemba	Benar
7	lemba	lemba	Benar
8	lemba	lemba	Benar
9	lemba	lemba	Benar
10	lemba	lemba	Benar
11	lemba	lemba	Benar
12	lemba	lemba	Benar
13	lemba	lemba	Benar
14	lemba	lemba	Benar
15	lemba	lemba	Benar
16	lemba	lemba	Benar
17	lemba	lemba	Benar
18	lemba	lemba	Benar
19	lemba	lemba	Benar
20	lemba	lemba	Benar
21	lemba	lemba	Benar
22	lemba	lemba	Benar
23	lemba	lemba	Benar
24	lemba	lemba	Benar

Jumlah Data : 100
 Jumlah Data Benar : 95
 Jumlah Data Salah : 5
 Persentase Data Benar : 95%
 Persentase Data Salah : 5%

Gambar 4.22 Pengujian Algorithm 13

Pengujian yang dilakukan menggunakan algoritma 13 menghasilkan 95 data benar dan 5 data salah dari 100 data yang diuji sehingga tingkat akurasi mencapai 95,9%

Daftar Data

No.	Kata Kunci	Kata Kunci
1	lemba	lemba
2	lemba	lemba
3	lemba	lemba
4	lemba	lemba
5	lemba	lemba
6	lemba	lemba
7	lemba	lemba
8	lemba	lemba
9	lemba	lemba
10	lemba	lemba
11	lemba	lemba
12	lemba	lemba
13	lemba	lemba
14	lemba	lemba
15	lemba	lemba
16	lemba	lemba
17	lemba	lemba
18	lemba	lemba
19	lemba	lemba
20	lemba	lemba
21	lemba	lemba
22	lemba	lemba
23	lemba	lemba
24	lemba	lemba

Hasil Pengujian

No.	Kata Kunci	Hasil	Memang Sederhana
1	lemba	lemba	Benar
2	lemba	lemba	Benar
3	lemba	lemba	Benar
4	lemba	lemba	Benar
5	lemba	lemba	Benar
6	lemba	lemba	Benar
7	lemba	lemba	Benar
8	lemba	lemba	Benar
9	lemba	lemba	Benar
10	lemba	lemba	Benar
11	lemba	lemba	Benar
12	lemba	lemba	Benar
13	lemba	lemba	Benar
14	lemba	lemba	Benar
15	lemba	lemba	Benar
16	lemba	lemba	Benar
17	lemba	lemba	Benar
18	lemba	lemba	Benar
19	lemba	lemba	Benar
20	lemba	lemba	Benar
21	lemba	lemba	Benar
22	lemba	lemba	Benar
23	lemba	lemba	Benar
24	lemba	lemba	Benar

Jumlah Data : 100
 Jumlah Data Benar : 99
 Jumlah Data Salah : 1
 Persentase Data Benar : 99%
 Persentase Data Salah : 1%

Gambar 4.23 Pengujian Algorithm 14

Pengujian yang dilakukan menggunakan algoritma 14 menghasilkan 966 data benar dan 34 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 96,6%

No	Data Asli	Kategori
1	100000	100000
2	100000	100000
3	100000	100000
4	100000	100000
5	100000	100000
6	100000	100000
7	100000	100000
8	100000	100000
9	100000	100000
10	100000	100000
11	100000	100000
12	100000	100000
13	100000	100000
14	100000	100000
15	100000	100000
16	100000	100000
17	100000	100000
18	100000	100000
19	100000	100000

No	Data	Kategori	Hasil Berkelompok
1	100000	100000	Benar
2	100000	100000	Benar
3	100000	100000	Benar
4	100000	100000	Benar
5	100000	100000	Benar
6	100000	100000	Benar
7	100000	100000	Benar
8	100000	100000	Benar
9	100000	100000	Benar
10	100000	100000	Benar
11	100000	100000	Benar
12	100000	100000	Benar
13	100000	100000	Benar
14	100000	100000	Benar
15	100000	100000	Benar
16	100000	100000	Benar
17	100000	100000	Benar
18	100000	100000	Benar
19	100000	100000	Benar

Jumlah Data: 1000
 Jumlah Data Benar: 966
 Jumlah Data Salah: 34
 Akurasi Data: 96,6%
 Prepresensi Data Benar: 96,6%

Gambar 4.24 Pengujian Algoritma 15

Pengujian yang dilakukan menggunakan algoritma 15 menghasilkan 959 data benar dan 41 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 95,9%

Hasil Algoritma 16

No.	Data	Data Asli	Hasil	Status
1	10000	10000	10000	Berisi
2	10000	10000	10000	Berisi
3	10000	10000	10000	Berisi
4	10000	10000	10000	Berisi
5	10000	10000	10000	Berisi
6	10000	10000	10000	Berisi
7	10000	10000	10000	Berisi
8	10000	10000	10000	Berisi
9	10000	10000	10000	Berisi
10	10000	10000	10000	Berisi
11	10000	10000	10000	Berisi
12	10000	10000	10000	Berisi
13	10000	10000	10000	Berisi
14	10000	10000	10000	Berisi
15	10000	10000	10000	Berisi
16	10000	10000	10000	Berisi
17	10000	10000	10000	Berisi
18	10000	10000	10000	Berisi
19	10000	10000	10000	Berisi
20	10000	10000	10000	Berisi
21	10000	10000	10000	Berisi
22	10000	10000	10000	Berisi
23	10000	10000	10000	Berisi

Jumlah Data Uji : 1000
 Jumlah Data Benar : 958
 Jumlah Data Salah : 42
 Akurasi Data Uji : 95,8%

Gambar 4.25 Pengujian Algoritma 16

Pengujian yang dilakukan menggunakan algoritma 16 menghasilkan 958 data benar dan 42 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 95,8%

Hasil Algoritma 16

No.	Data	Data Asli	Hasil	Status
1	10000	10000	10000	Berisi
2	10000	10000	10000	Berisi
3	10000	10000	10000	Berisi
4	10000	10000	10000	Berisi
5	10000	10000	10000	Berisi
6	10000	10000	10000	Berisi
7	10000	10000	10000	Berisi
8	10000	10000	10000	Berisi
9	10000	10000	10000	Berisi
10	10000	10000	10000	Berisi
11	10000	10000	10000	Berisi
12	10000	10000	10000	Berisi
13	10000	10000	10000	Berisi
14	10000	10000	10000	Berisi
15	10000	10000	10000	Berisi
16	10000	10000	10000	Berisi
17	10000	10000	10000	Berisi
18	10000	10000	10000	Berisi
19	10000	10000	10000	Berisi
20	10000	10000	10000	Berisi
21	10000	10000	10000	Berisi
22	10000	10000	10000	Berisi
23	10000	10000	10000	Berisi

Jumlah Data Uji : 1000
 Jumlah Data Benar : 958
 Jumlah Data Salah : 42
 Akurasi Data Uji : 95,8%

Gambar 4.26 Pengujian Algorithm 17

Pengujian yang dilakukan menggunakan algoritma 17 menghasilkan 965 data benar dan 35 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 96,5%

Data Data			Hasil Kognitif					
No	Data Asli	Data Hasil	No	Algoritma	Data Asli	Data Hasil	Hasil	Status
1	10000	10000	1	Algoritma 1	10000	10000	Benar	Benar
2	10000	10000	2	Algoritma 2	10000	10000	Benar	Benar
3	10000	10000	3	Algoritma 3	10000	10000	Benar	Benar
4	10000	10000	4	Algoritma 4	10000	10000	Benar	Benar
5	10000	10000	5	Algoritma 5	10000	10000	Benar	Benar
6	10000	10000	6	Algoritma 6	10000	10000	Benar	Benar
7	10000	10000	7	Algoritma 7	10000	10000	Benar	Benar
8	10000	10000	8	Algoritma 8	10000	10000	Benar	Benar
9	10000	10000	9	Algoritma 9	10000	10000	Benar	Benar
10	10000	10000	10	Algoritma 10	10000	10000	Benar	Benar
11	10000	10000	11	Algoritma 11	10000	10000	Benar	Benar
12	10000	10000	12	Algoritma 12	10000	10000	Benar	Benar
13	10000	10000	13	Algoritma 13	10000	10000	Benar	Benar
14	10000	10000	14	Algoritma 14	10000	10000	Benar	Benar
15	10000	10000	15	Algoritma 15	10000	10000	Benar	Benar
16	10000	10000	16	Algoritma 16	10000	10000	Benar	Benar
17	10000	10000	17	Algoritma 17	10000	10000	Benar	Benar
18	10000	10000	18	Algoritma 18	10000	10000	Benar	Benar
19	10000	10000	19	Algoritma 19	10000	10000	Benar	Benar
20	10000	10000	20	Algoritma 20	10000	10000	Benar	Benar

Gambar 4.27 Pengujian Algorithm 18

Pengujian yang dilakukan menggunakan algoritma 18 menghasilkan 965 data benar dan 35 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 96,5%

Gambar 4.29 Pengujian Algorithma 20

Pengujian yang dilakukan menggunakan algoritma 20 menghasilkan 938 data benar dan 62 data salah dari 1000 data yang diuji sehingga tingkat akurasiya mencapai 93,8%

No	Data	Data Dasar	Hasil	Skor	Status
1	1000	1000	1000	1000	Benar
2	1000	1000	1000	1000	Benar
3	1000	1000	1000	1000	Benar
4	1000	1000	1000	1000	Benar
5	1000	1000	1000	1000	Benar
6	1000	1000	1000	1000	Benar
7	1000	1000	1000	1000	Benar
8	1000	1000	1000	1000	Benar
9	1000	1000	1000	1000	Benar
10	1000	1000	1000	1000	Benar
11	1000	1000	1000	1000	Benar
12	1000	1000	1000	1000	Benar
13	1000	1000	1000	1000	Benar
14	1000	1000	1000	1000	Benar
15	1000	1000	1000	1000	Benar
16	1000	1000	1000	1000	Benar
17	1000	1000	1000	1000	Benar
18	1000	1000	1000	1000	Benar
19	1000	1000	1000	1000	Benar
20	1000	1000	1000	1000	Benar
21	1000	1000	1000	1000	Benar
22	1000	1000	1000	1000	Benar
23	1000	1000	1000	1000	Benar
24	1000	1000	1000	1000	Benar

Gambar 4.30 Pengujian Algorithma 21

Pengujian yang dilakukan menggunakan algoritma 21 menghasilkan 955 data benar dan 45 data salah dari 1000 data yang diuji sehingga tingkat akurasiya mencapai 95,5%

Gambar 4.32 Pengujian Algoritma 23

Pengujian yang dilakukan menggunakan algoritma 23 menghasilkan 950 data benar dan 50 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 95%

No	Nama	Data Dasar	Hasil	Status
1	Algoritma 1	1000	1000	Benar
2	Algoritma 2	1000	1000	Benar
3	Algoritma 3	1000	1000	Benar
4	Algoritma 4	1000	1000	Benar
5	Algoritma 5	1000	1000	Benar
6	Algoritma 6	1000	1000	Benar
7	Algoritma 7	1000	1000	Benar
8	Algoritma 8	1000	1000	Benar
9	Algoritma 9	1000	1000	Benar
10	Algoritma 10	1000	1000	Benar
11	Algoritma 11	1000	1000	Benar
12	Algoritma 12	1000	1000	Benar
13	Algoritma 13	1000	1000	Benar
14	Algoritma 14	1000	1000	Benar
15	Algoritma 15	1000	1000	Benar
16	Algoritma 16	1000	1000	Benar
17	Algoritma 17	1000	1000	Benar
18	Algoritma 18	1000	1000	Benar
19	Algoritma 19	1000	1000	Benar
20	Algoritma 20	1000	1000	Benar
21	Algoritma 21	1000	1000	Benar
22	Algoritma 22	1000	1000	Benar
23	Algoritma 23	1000	1000	Benar
24	Algoritma 24	1000	1000	Benar

Gambar 4.33 Pengujian Algoritma 24

Pengujian yang dilakukan menggunakan algoritma 24 menghasilkan 938 data benar dan 62 data salah dari 1000 data yang diuji sehingga tingkat akurasi mencapai 93,8%

Dari pengujian yang telah dilakukan terhadap ke 24 algoritma yang telah disusun, didapatkan tingkat akurasi yang bervariasi dari yang terendah dengan tingkat akurasi sebesar 93,4% sampai yang tertinggi dengan tingkat akurasi mencapai 96,6%. Hasil dari pengujian yang dilakukan jika ditampilkan dalam grafik dapat disajikan sebagai berikut.



Gambar 4.34 Grafik tingkat akurasi algoritma stemming

Dari pengujian yang dilakukan terhadap 24 algoritma yang telah disusun, jika diurutkan berdasarkan tingkat akurasinya didapatkan urutan sebagai berikut.

Tabel 4.3 Urutan algoritma berdasarkan tingkat akurasi

Algoritma	Akurasi	Kesalahan
Algoritma 14	96,6%	3,4%
Algoritma 17	96,5%	3,5%
Algoritma 18	96,5%	3,5%
Algoritma 13	95,9%	4,1%
Algoritma 15	95,9%	4,1%
Algoritma 16	95,8%	4,2%
Algoritma 21	95,5%	4,5%
Algoritma 22	95,5%	4,5%
Algoritma 10	95,4%	4,6%

Tabel 4.3 (Lanjutan)

Algoritma 3	95,4%	4,6%
Algoritma 4	95,3%	4,7%
Algoritma 9	95,0%	5,0%
Algoritma 23	95,0%	5,0%
Algoritma 5	94,6%	5,4%
Algoritma 1	94,5%	5,5%
Algoritma 7	94,5%	5,5%
Algoritma 11	94,3%	5,7%
Algoritma 19	94,3%	5,7%
Algoritma 12	93,8%	6,2%
Algoritma 20	93,8%	6,2%
Algoritma 24	93,8%	6,2%
Algoritma 2	93,4%	6,6%
Algoritma 6	93,4%	6,6%
Algoritma 8	93,4%	6,6%

Berdasarkan hasil uji coba yang telah dilakukan terhadap 24 algoritma yang telah disusun, didapatkan algoritma 14 mempunyai tingkat akurasi paling baik yang mencapai 96,6%. Pada algoritma 14 pemeriksaan imbuhan yang pertama dilakukan adalah pemeriksaan imbuhan awalan yang secara berturut-turut dilanjutkan dengan pemeriksaan imbuhan apit, akhiran dan sisipan.

Urutan pemeriksaan tersebut sesuai dengan urutan imbuhan dalam morfologi bahasa Kaili dari yang terbanyak aturan imbuhan, yaitu aturan imbuhan awalan yang mempunyai 70 aturan imbuhan atau sebesar 61,95% dari total 113 aturan imbuhan, kemudian aturan imbuhan apit yang mempunyai 33 aturan imbuhan atau sebesar 29,20% dari total 113 aturan imbuhan, selanjutnya aturan imbuhan akhiran yang mempunyai 6 aturan imbuhan atau sebesar 5,31% dari total 113 aturan imbuhan dan yang terakhir aturan imbuhan sisipan yang mempunyai 4 aturan imbuhan atau sebesar 3,54% dari total 113 aturan imbuhan.

Untuk mengetahui tingkat akurasi dari algoritma yang telah disusun jika digunakan untuk melakukan stemming teks bahasa kaili, dilakukan pengujian menggunakan 5 cerita yang ditulis menggunakan bahasa kaili, yaitu:

1. Topomeka
2. Sikola
3. Sapeda
4. Noasu Bavu
5. Nodanda

Algoritma yang diuji dalam pengujian ini adalah algoritma 14, algoritma 17 dan algoritma 18. Algoritma-algoritma tersebut merupakan urutan pertama sampai dengan ketiga dari pengujian pertama. Pada pengujian kedua ini, sebelum melakukan stemming dilakukan proses membersihkan tanda baca agar kata dapat diproses dengan baik. Selain itu dilakukan juga pengecekan kata yang akan diperiksa agar tidak terjadi pengulangan atas kata yang telah distemming. Selanjutnya kata dasar yang dihasilkan dari proses stemming akan dicek dengan

kamus bahasa Kaili untuk mengetahui apakah kata dasar yang dihasilkan benar atau salah.

Pada pengujian algoritma 14, algoritma 17 dan algoritma 18 dari proses stemming yang dilakukan terhadap 5 cerita tersebut, didapatkan hasil sebagai berikut:

Tabel 4.4 Hasil Stemming Topomeka

No	Kata	Hasil Stemming Algoritma			Menurut Kamus
		14	17	18	
1	nana	ria	ria	ria	ria
2	sangu	sangu	sangu	sangu	sangu
3	levuto	levuto	levuto	levuto	levuto
4	nosangaka	sanga	sanga	sanga	sanga
5	nu	nu	nu	nu	nu
6	palu	palu	palu	palu	palu
7	ri	ri	ri	ri	ri
8	bivina	bivi	bivi	bivi	bivi
9	bara	bara	bara	bara	bara
10	sakuya	sakuya	sakuya	sakuya	sakuya
11	angu	angu	angu	angu	angu
12	banua	banua	banua	banua	banua
13	pura	pura	pura	pura	pura
14	tona	tona	tona	tona	tona
15	samai	samai	samai	samai	samai

Tabel 4.4 (Lanjutan)

16	ledo	ledo	ledo	ledo	ledo
17	ntanina	nina	nina	nina	nina
18	panggavia	panggavia	panggavia	panggavia	via
19	aga	aga	aga	aga	aga
20	pomeka	eka	eka	eka	eka
21	bau	bau	bau	bau	bau
22	tasi	tasi	tasi	tasi	tasi
23	nomparia	paria	paria	paria	paria
24	sakaya	sakaya	sakaya	sakaya	sakaya
25	nipesavira	nipesavira	nipesavira	nipesavira	savi
26	hilau	hau	hau	hau	hau
27	momeka	eka	eka	eka	eka
28	meka	meka	meka	meka	meka
29	balumba	balumba	balumba	balumba	balumba
30	mai	mai	mai	mai	mai
31	eo	eo	eo	eo	eo
32	nambaso	mbaso	mbaso	mbaso	mbaso
33	saito	saito	saito	saito	saito
34	i	i	i	i	i
35	polebalu	polebalu	polebalu	polebalu	polebalu
36	nombatomo	tompo	tompo	tompo	tompo
37	pebauna	bau	bau	bau	bau

Tabel 4.4 (Lanjutan)

38	nadea	dea	dea	dea	dea
39	naeka	eka	eka	eka	eka
40	nompeinta	peinta	peinta	peinta	peinta
41	nangulika	uli	uli	uli	uli
42	ia	ia	ia	ia	ia
43	hee	hee	hee	hee	hee
44	nemo	nemo	nemo	nemo	nemo
45	gaga	gaga	gaga	gaga	gaga
46	iko	iko	iko	iko	iko
47	mebau	bau	bau	bau	bau
48	kandalana	ndala	ndala	ndala	ndala
49	makaupua	kaupua	kaupua	kaupua	kaupua
50	belaka	belaka	belaka	belaka	belaka
51	maria	ria	ria	ria	ria
52	abalaa	abalaa	abalaa	abalaa	abalaa
53	manggava	manggava	manggava	manggava	kava
54	tatanganana	tatanganana	tatanganana	tatanganana	tatanganana
55	semamo	sema	sema	sema	sema
56	mompороa	roa	roa	roa	roa
57	ngena	ngena	ngena	ngena	ngena
58	apa	apa	apa	apa	apa
59	kami	kami	kami	kami	kami

Tabel 4.4 (Lanjutan)

60	tunggai	tunggai	tunggai	tunggai	tunggai
61	nakava	kava	kava	kava	kava
62	hii	hii	hii	hii	hii
63	nesana	sana	sana	sana	sana
64	nisaniku	sani	sani	sani	sani
65	muni	muni	muni	muni	muni
66	akupa	aku	aku	aku	aku
67	domo	domo	domo	domo	domo
68	mainga	inga	inga	inga	inga
69	pangane	pangane	pangane	pangane	pangane
70	koimo	koi	koi	koi	koi
71	tobelo	belo	belo	belo	belo
72	bo	bo	bo	bo	bo
73	niulika	uli	Uli	uli	uli

Dari proses stemming yang dilakukan terhadap cerita topomeka, baik algoritma 14, algoritma 17 maupun algoritma 18 mendapatkan hasil yang sama yaitu 69 kata benar dan 4 kata salah dari 73 kata sehingga tingkat akurasi sebesar 94,52%.

Tabel 4.5 Hasil Stemming Sikola

No	Kata	Hasil Stemming			Menurut Kamus
		14	17	18	
1	naria	ria	ria	ria	ria
2	saito	saito	saito	saito	saito
3	ngana	ngana	ngana	ngana	ngana
4	nosanga	sanga	sanga	sanga	sanga
5	surudi	surudi	surudi	surudi	surudi
6	nakavao	vao	vao	vao	vao
7	ponturona	nturo	nturo	nturo	nturo
8	bo	bo	bo	bo	bo
9	posikola	sikola	sikola	sikola	sikola
10	mpadondo	mpadondo	mpadondo	mpadondo	dondo
11	i	i	i	i	i
12	membangu	mbangu	mbangu	mbangu	mbangu
13	kana	kana	kana	kana	kana
14	masalisa	salisa	salisa	salisa	salisa
15	dako	dako	dako	dako	dako
16	nembangu	mbangu	mbangu	mbangu	mbangu
17	ia	ia	ia	ia	ia
18	kapola	pola	pola	pola	pola
19	pola	pola	pola	pola	pola
20	hilau	hau	hau	hau	hau

Tabel 4.5 (Lanjutan)

21	mandiu	diu	diu	diu	diu
22	ri	ri	ri	ri	ri
23	salura	salura	salura	salura	salura
24	nakava	kava	kava	kava	kava
25	nipakande	kande	kande	kande	kande
26	ntinanamo	ntina	ntina	ntina	ntina
27	pade	pade	pade	pade	pade
28	nidekeina	dekei	dekei	dekei	dekei
29	baju	baju	baju	baju	baju
30	naramba	ramba	ramba	ramba	ramba
31	puruka	puruka	puruka	puruka	puruka
32	muni	muni	muni	muni	muni
33	mai	mai	mai	mai	mai
34	ana	ana	ana	ana	ana
35	nu	nu	nu	nu	nu
36	guru	guru	guru	guru	guru
37	itumo	itu	itu	itu	itu
38	nompasua	sua	sua	sua	sua
39	kalupa	kalupa	kalupa	kalupa	kalupa
40	ane	ane	ane	ane	ane
41	sikola	sikola	sikola	sikola	sikola
42	nesavi	savi	savi	savi	savi

Tabel 4.5 (Lanjutan)

43	sapeda	sapeda	sapeda	sapeda	sapeda
44	kodi	kodi	kodi	kodi	kodi
45	nialika	ali	ali	ali	ali
46	ntuamana	ntuamana	ntuamana	ntuamana	ntuamana
47	moliu	liu	liu	liu	liu
48	riara	riara	riara	riara	riara
49	ngata	ngata	ngata	ngata	ngata
50	nipekutana	kutana	kutana	kutana	kutana
51	ntona	ntona	ntona	ntona	ntona
52	isema	isema	isema	isema	isema
53	nosapeda	sapeda	sapeda	sapeda	sapeda
54	noliu	liu	liu	liu	liu
55	pangane	pangane	pangane	pangane	pangane
56	nesina	sana	sana	sana	sana
57	tona	tona	tona	tona	tona
58	nangisanimo	isani	isani	isani	isani
59	nonturo	nturo	nturo	nturo	nturo
60	donggala	donggala	donggala	donggala	donggala
61	nanguli	uli	uli	uli	uli
62	totua	tua	tua	tua	tua
63	nisanikumo	sani	sani	sani	sani
64	masipato	sipato	sipato	sipato	sipato

Tabel 4.5 (Lanjutan)

65	mosapeda	sapeda	sapeda	sapeda	sapeda
66	sii	sii	sii	sii	sii
67	eo	eo	eo	eo	eo
68	nipeintaku	peinta	peinta	peinta	peinta
69	nekutanamo	kutana	kutana	kutana	kutana
70	umba	umba	umba	umba	umba
71	hau	hau	hau	hau	hau
72	nesanamo	sana	sana	sana	sana
73	sambote	bote	bote	bote	bote
74	ledo	ledo	ledo	ledo	ledo
75	naola	ola	ola	ola	ola
76	nosikola	kola	kola	kola	sikola
77	sampalepa	sampale	sampale	sampale	sampale
78	nadua	dua	dua	dua	dua
79	kaulu	ulu	ulu	ulu	ulu
80	guruna	guru	guru	guru	guru
81	nangulimo	guli	guli	guli	uli
82	ntaina	ina	ina	ina	ina
83	guruku	guru	guru	guru	guru
84	eva	eva	eva	eva	eva
85	hii	hii	hii	hii	hii
86	pombasana	pombasana	pombasana	pombasana	pombasana

Tabel 4.5 (Lanjutan)

87	ria	ria	ria	ria	ria
88	nakeku	keku	keku	keku	keku
89	ivesia	ivesia	ivesia	ivesia	ivesia
90	porekena	reke	reke	reke	reke
91	poukina	uki	uki	uki	uki
92	nuapa	nuapa	nuapa	nuapa	nuapa
93	niuli	uli	uli	uli	uli
94	aga	aga	aga	aga	aga
95	sanggani	ni	ni	ni	nggani
96	raepena	raepena	raepena	raepena	epe
97	nisanina	sani	sani	sani	sani
98	puramo	pura	pura	pura	pura
99	nipotove	tove	tove	tove	tove
100	mpuu	mpuu	mpuu	mpuu	mpuu
101	banua	banua	banua	banua	banua
102	ntomana	ntomana	ntomana	ntomana	ntoma
103	nosinggavaka	singgava	singgava	singgava	singgava
104	nangulika	uli	uli	uli	uli
105	ka	ka	ka	ka	ka
106	to	to	to	to	to
107	mana	mana	mana	mana	mana
108	anamiu	anamiu	anamiu	anamiu	anamiu

Tabel 4.5 (Lanjutan)

109	neliu	liu	liu	liu	liu
110	katauna	tau	tau	tau	tau
111	kupatudukika	kupatudukika	kupatudukika	kupatudukika	patuduki
112	nasaemo	sae	sae	sae	sae
113	ira	ira	ira	ira	ira
114	nturo	nturo	nturo	nturo	nturo
115	tuli	tuli	tuli	tuli	tuli
116	ninginumo	inu	inu	inu	inu
117	uve	uve	uve	uve	uve
118	mpane	mpane	mpane	mpane	mpane
119	nopalakanamo	palakana	palakana	palakana	palakana
120	manjili	ili	ili	ili	sili
121	banuana	banua	banua	banua	banua
122	naupu	upu	upu	upu	upu
123	nosiganggo	ganggo	ganggo	ganggo	ganggo
124	pale	pale	pale	pale	pale
125	randua	randua	randua	randua	randua
126	nanau	au	au	au	nau
127	tana	tana	tana	tana	tana
128	nikeni	keni	keni	keni	keni
129	ntoma	ntoma	ntoma	ntoma	ntoma
130	sanggoto	koto	koto	koto	koto

Tabel 4.5 (Lanjutan)

131	dala	dala	dala	dala	dala
132	notesa	tesa	tesa	tesa	tesa
133	tesamo	tesa	tesa	tesa	tesa
134	nggalako	nggalako	nggalako	nggalako	lako
135	lako	lako	lako	lako	lako
136	kaupunapa	kaupunapa	kaupunapa	kaupunapa	upu
137	tesara	tesa	tesa	tesa	tesa
138	nositabemo	tabe	tabe	tabe	tabe
139	nosintalikuri	ntalikuri	ntalikuri	ntalikuri	ntalikuri

Dari proses stemming yang dilakukan terhadap cerita Sikola, baik algoritma 14, algoritma 17 maupun algoritma 18 mendapatkan hasil yang sama yaitu 128 kata benar dan 11 kata salah dari 139 kata sehingga tingkat akurasi sebesar 92,09%.

Tabel 4.6 Hasil Stemming Sapeda

No	Kata	Hasil Stemming			Menurut Kamus
		14	17	18	
1	naria	ria	ria	ria	ria
2	tatalu	tatalu	tatalu	tatalu	tatalu
3	petoro	petoro	petoro	petoro	petoro
4	nosapeda	sapeda	sapeda	sapeda	sapeda
5	noliu	liu	liu	liu	liu

Tabel 4.6 (Lanjutan)

6	ri	ri	ri	ri	ri
7	donggala	donggala	donggala	donggala	donggala
8	kodi	kodi	kodi	kodi	kodi
9	hilau	hau	hau	hau	hau
10	mapola	pola	pola	pola	pola
11	duyu	duyu	duyu	duyu	duyu
12	apa	apa	apa	apa	apa
13	saito	saito	saito	saito	saito
14	noberei	berei	berei	berei	berei
15	pangane	pangane	pangane	pangane	pangane
16	bara	bara	bara	bara	bara
17	mangande	gande	gande	gande	gande
18	ngande	ngande	ngande	ngande	kande
19	banua	banua	banua	banua	banua
20	nu	nu	nu	nu	nu
21	roara	roa	roa	roa	roa
22	mai	mai	mai	mai	mai
23	nasaemo	sae	sae	sae	sae
24	dopa	dopa	dopa	dopa	dopa
25	nanjili	ili	ili	ili	sili
26	palu	palu	palu	palu	palu
27	nangulimo	guli	guli	guli	uli

Tabel 4.6 (Lanjutan)

28	to	to	to	to	to
29	umbamo	umba	umba	umba	umba
30	kapola	pola	pola	pola	pola
31	nanguli	uli	uli	uli	uli
32	tona	tona	tona	tona	tona
33	santanga	tanga	tanga	tanga	tanga
34	dala	dala	dala	dala	dala
35	pengavu	ngavu	ngavu	ngavu	ngavu
36	boyaoge	boyaoge	boyaoge	boyaoge	boyaoge
37	nasae	sae	sae	sae	sae
38	nariamo	ria	ria	ria	ria
39	dako	dako	dako	dako	dako
40	potomu	tomu	tomu	tomu	tomu
41	nipekutanamo	pekutana	pekutana	pekutana	pekutana
42	ntona	ntona	ntona	ntona	ntona
43	nesanamo	sana	sana	sana	sana
44	ia	ia	ia	ia	ia
45	da	da	da	da	da
46	hamai	hamai	hamai	hamai	hamai
47	nantunu	tunu	tunu	tunu	tunu
48	ntunu	ntunu	ntunu	ntunu	ntunu
49	dale	dale	dale	dale	dale

Tabel 4.6 (Lanjutan)

50	bo	bo	bo	bo	bo
51	loka	loka	loka	loka	loka
52	talumbaa	talumbaa	talumbaa	talumbaa	talumbaa
53	manu	manu	manu	manu	manu
54	nitunura	tunu	tunu	tunu	tunu
55	nikandekara	kande	kande	kande	kande
56	nitunu	tunu	tunu	tunu	tunu
57	rua	rua	rua	rua	rua
58	mbaa	mbaa	mbaa	mbaa	mbaa
59	muni	muni	muni	muni	muni
60	rapoutaka	rapoutaka	rapoutaka	rapoutaka	putaka
61	kandea	kande	kande	kande	kande
62	panguli	uli	uli	uli	uli
63	ntotua	ntotua	ntotua	ntotua	ntotua
64	nangongopa	ngongo	ngongo	ngongo	ngongo
65	nangongo	ongo	ongo	ongo	ongo
66	munimo	muni	muni	muni	muni
67	nangandemo	gande	gande	gande	gande
68	ira	ira	ira	ira	ira
69	pura	pura	pura	pura	pura
70	naupu	upu	upu	upu	upu
71	nangande	gande	gande	gande	gande

Tabel 4.6 (Lanjutan)

72	pade	pade	pade	pade	pade
73	naupupa	upu	upu	upu	upu
74	mbo	mbo	mbo	mbo	mbo
75	nekagole	kagole	kagole	kagole	kagole
76	golemo	gole	gole	gole	gole
77	tuli	tuli	tuli	tuli	tuli
78	namonjomo	monjo	monjo	monjo	monjo
79	tupu	tupu	tupu	tupu	tupu
80	nauya	uya	uya	uya	uya
81	rarantaira	rarantaira	rarantaira	rarantaira	rarantaira
82	namonjo	monjo	monjo	monjo	monjo
83	puramo	pura	pura	pura	pura
84	napanga	panga	panga	panga	panga
85	ntotomo	ntoto	ntoto	ntoto	ntoto
86	naturu	туру	туру	туру	туру
87	nematamo	mata	mata	mata	mata
88	niikenamo	like	like	like	like
89	roana	roa	roa	roa	roa
90	manjili	ili	ili	ili	ili
91	nelikemo	like	like	like	like
92	hee	hee	hee	hee	hee
93	pembangu	mbangu	mbangu	mbangu	mbangu

Tabel 4.6 (Lanjutan)

94	nabongimo	bongi	bongi	bongi	bongi
95	nembangu	mbangu	mbangu	mbangu	mbangu
96	nopalakanamo	palakana	palakana	palakana	palakana
97	pola	pola	pola	pola	pola
98	nesavi	savi	savi	savi	savi
99	sapedara	sapeda	sapeda	sapeda	sapeda
100	niosera	ose	ose	ose	ose
101	dalara	dala	dala	dala	dala
102	nakavapa	kava	kava	kava	kava
103	potuana	tua	tua	tua	tua
104	aginapa	agina	agina	agina	agina
105	kita	kita	kita	kita	kita
106	manau	au	au	au	au
107	sii	sii	sii	sii	sii
108	nesana	sana	sana	sana	sana
109	tumo	tumo	tumo	tumo	tumo
110	papolamo	pola	pola	pola	pola
111	nopoomo	poo	poo	poo	poo
112	nogiro	giro	giro	giro	giro
113	rampa	rampa	rampa	rampa	rampa
114	sapeda	sapeda	sapeda	sapeda	sapeda
115	tano	tano	tano	tano	tano

Tabel 4.6 (Lanjutan)

116	nombarumpa	rumpa	rumpa	rumpa	rumpa
117	bata	bata	bata	bata	bata
118	ntaipa	ntai	ntai	ntai	ntai
119	mbaso	mbaso	mbaso	mbaso	mbaso
120	bivi	bivi	bivi	bivi	bivi
121	sapedana	sapeda	sapeda	sapeda	sapeda
122	nakoto	koto	koto	koto	koto
123	balenggana	balengga	balengga	balengga	balengga
124	kada	kada	kada	kada	kada
125	sabingga	bingga	bingga	bingga	bingga
126	damo	damo	damo	damo	damo
127	nikova	kova	kova	kova	kova
128	nisimbuaka	simbuaka	simbuaka	simbuaka	simbuaka
129	napola	pola	pola	pola	pola
130	itumo	itu	itu	itu	itu
131	ane	ane	ane	ane	ane
132	kana	kana	kana	kana	kana
133	nanau	au	au	au	au
134	motua	tua	tua	tua	tua
135	nadaa	daa	daa	daa	daa
136	ntoto	ntoto	ntoto	ntoto	ntoto

Dari proses stemming yang dilakukan terhadap cerita Sapeda, baik algoritma 14, algoritma 17 maupun algoritma 18 mendapatkan hasil yang sama yaitu 132 kata benar dan 4 kata salah dari 136 kata sehingga tingkat akurasi sebesar 97,06%.

Tabel 4.7 Hasil Stemming Noasu Bavu

No	Kata	Hasil Stemming Algoritma			Menurut Kamus
		14	17	18	
1	pampa	pampa	pampa	pampa	pampa
2	ntona	ntona	ntona	ntona	ntona
3	domo	domo	domo	domo	domo
4	naria	ria	ria	ria	ria
5	najadi	jadi	jadi	jadi	jadi
6	katuvuna	tuvu	tuvu	tuvu	tuvu
7	apa	apa	apa	apa	apa
8	napara	para	para	para	para
9	gaga	gaga	gaga	gaga	gaga
10	bo	bo	bo	bo	bo
11	rakayu	rakayu	rakayu	rakayu	rakayu
12	nadea	dea	dea	dea	dea
13	ntoto	ntoto	ntoto	ntoto	ntoto
14	bavu	bavu	bavu	bavu	bavu
15	nesua	sua	sua	sua	sua

Tabel 4.7 (Lanjutan)

16	riara	riara	riara	riara	riara
17	nu	nu	nu	nu	nu
18	panipara	panipara	panipara	panipara	para
19	ane	ane	ane	ane	ane
20	nitudaramo	nitudaramo	nitudaramo	nitudaramo	tuda
21	dale	dale	dale	dale	dale
22	bara	bara	bara	bara	bara
23	kasubi	kasubi	kasubi	kasubi	kasubi
24	napiri	piri	piri	piri	piri
25	piri	piri	piri	piri	piri
26	matara	tara	tara	tara	tara
27	nompaturusi	turusi	turusi	turusi	turusi
28	nitudara	tuda	tuda	tuda	tuda
29	maupa	upa	upa	upa	upa
30	ivesia	ivesia	ivesia	ivesia	ivesia
31	kana	kana	kana	kana	kana
32	muni	muni	muni	muni	muni
33	nasumpupa	sumpu	sumpu	sumpu	sumpu
34	rarantaira	rarantaira	rarantaira	rarantaira	rarantaira
35	nompasiromumo	pasiromu	pasiromu	pasiromu	pasiromu
36	ira	ira	ira	ira	ira
37	nombajaritai	jarita	jarita	jarita	jarita

Tabel 4.7 (Lanjutan)

38	kadea	dea	dea	dea	dea
39	bongi	bongi	bongi	bongi	bongi
40	pampara	pampa	pampa	pampa	pampa
41	da	da	da	da	da
42	kapojarita	kapojarita	kapojarita	kapojarita	jarita
43	rita	rita	rita	rita	rita
44	nakavamo	kava	kava	kava	kava
45	saito	saito	saito	saito	saito
46	totua	tua	tua	tua	tua
47	nangulij	uli	uli	uli	uli
48	nabelopa	belo	belo	belo	belo
49	raasukita	raasukita	raasukita	raasukita	asu
50	mpadondo	mpadondo	mpadondo	mpadondo	mpadondo
51	nesana	sana	sana	sana	sana
52	puramo	pura	pura	pura	pura
53	ivetu	ivetu	ivetu	ivetu	ivetu
54	tano	tano	tano	tano	tano
55	pangulimiu	pangulimiu	pangulimiu	pangulimiu	uli
56	nabelo	belo	belo	belo	belo
57	nangulimo	guli	guli	guli	uli
58	pangane	pangane	pangane	pangane	pangane
59	masadondo	sadondo	sadondo	sadondo	sadondo

Tabel 4.7 (Lanjutan)

60	kita	kita	kita	kita	kita
61	mosinggava	singgava	singgava	singgava	singgava
62	ri	ri	ri	ri	ri
63	sii	sii	sii	sii	sii
64	kaeona	eo	eo	eo	eo
65	mpadondona	mpadondo	mpadondo	mpadondo	mpadondo
66	nosinggavamo	singgava	singgava	singgava	singgava
67	polibura	polibura	polibura	polibura	polibura
68	riavi	riavi	riavi	riavi	riavi
69	nanggeni	nanggeni	nanggeni	nanggeni	keni
70	asu	asu	asu	asu	asu
71	tavala	vala	vala	vala	vala
72	kanjai	kanjai	kanjai	kanjai	kanjai
73	dopa	dopa	dopa	dopa	dopa
74	nasae	sae	sae	sae	sae
75	nesuamo	sua	sua	sua	sua
76	tona	tona	tona	tona	tona
77	pade	pade	pade	pade	pade
78	netuntuni	tuntuni	tuntuni	tuntuni	tuntuni
79	neragamo	raga	raga	raga	raga
80	nijagaira	nijagaira	nijagaira	nijagaira	jaga
81	pura	pura	pura	pura	pura

Tabel 4.7 (Lanjutan)

82	dala	dala	dala	dala	dala
83	mesuvu	suvu	suvu	suvu	suvu
84	tumai	tumai	tumai	tumai	tumai
85	netabuni	tabuni	tabuni	tabuni	tabuni
86	puna	puna	puna	puna	puna
87	kayu	kayu	kayu	kayu	kayu
88	mbaso	mbaso	mbaso	mbaso	mbaso
89	napanga	panga	panga	panga	panga
90	noliumo	liu	liu	liu	liu
91	sangu	sangu	sangu	sangu	sangu
92	damo	damo	damo	damo	damo
93	nambaso	mbaso	mbaso	mbaso	mbaso
94	nibununamo	bunu	bunu	bunu	bunu
95	noliu	liu	liu	liu	liu
96	nambela	mbela	mbela	mbela	mbela
97	lovu	lovu	lovu	lovu	lovu
98	ntaina	ina	ina	ina	ina
99	ledo	ledo	ledo	ledo	ledo
100	kijina	kiji	kiji	kiji	kiji
101	namatemo	mate	mate	mate	mate
102	vaimo	vai	vai	vai	vai
103	sangupa	sangu	sangu	sangu	sangu

Tabel 4.7 (Lanjutan)

104	nibununa	bunu	bunu	bunu	bunu
105	vai	vai	vai	vai	vai
106	ruangumo	ruangu	ruangu	ruangu	ruangu
107	namate	mate	mate	mate	mate
108	umbanapa	umbanapa	umbanapa	umbanapa	umbanapa
109	ntanina	nina	nina	nina	nina
110	nadeapa	dea	dea	dea	dea
111	arakayu	arakayu	arakayu	arakayu	arakayu
112	hilau	hau	hau	hau	hau
113	niasukira	niasukira	niasukira	niasukira	niasukira
114	eo	eo	eo	eo	eo
115	saeo	eo	eo	eo	eo
116	mai	mai	mai	mai	mai
117	kabongina	bongi	bongi	bongi	bongi
118	napalavangarmo	palavanga	palavanga	palavanga	palavanga
119	sakide	sakide	sakide	sakide	sakide
120	pitu	pitu	pitu	pitu	pitu
121	nitipanakara	nitipanakara	nitipanakara	nitipanakara	tipa
122	hia	hia	hia	hia	hia
123	nisaka	saka	saka	saka	saka
124	mesua	sua	sua	sua	sua
125	nibunura	bunu	bunu	bunu	bunu

Tabel 4.7 (Lanjutan)

126	asura	asu	asu	asu	asu
127	aga	aga	aga	aga	aga
128	vesiamo	vesia	vesia	vesia	vesia
129	panggaviara	panggaviara	panggaviara	panggaviara	panggavia
130	tuli	tuli	tuli	tuli	tuli
131	poviara	poviara	poviara	poviara	via
132	munimo	muni	muni	muni	muni
133	natepiri	piri	piri	piri	piri
134	naturu	туру	туру	туру	туру
135	nipoviara	nipoviara	nipoviara	nipoviara	via
136	katuvu	tuvu	tuvu	tuvu	tuvu
137	dale	dale	dale	dale	dale
138	kasubira	kasubi	kasubi	kasubi	kasubi
139	tuda	tuda	tuda	tuda	tuda

Dari proses stemming yang dilakukan terhadap cerita Noasu Bavu, baik algoritma 14, algoritma 17 maupun algoritma 18 mendapatkan hasil yang sama yaitu 127 kata benar dan 12 kata salah dari 139 kata sehingga tingkat akurasinya sebesar 91,37%.

Tabel 4.8 Hasil Stemming Nodanda

No	Kata	Hasil Stemming Algoritma			Menurut Kamus
		14	17	18	
1	naria	ria	ria	ria	ria
2	ngana	ngana	ngana	ngana	ngana
3	uvalu	uvalu	uvalu	uvalu	uvalu
4	nasiromu	siromu	siromu	siromu	siromu
5	ri	ri	ri	ri	ri
6	bivi	bivi	bivi	bivi	bivi
7	nu	nu	nu	nu	nu
8	banua	banua	banua	banua	banua
9	kami	kami	kami	kami	kami
10	nekutana	kutana	kutana	kutana	kutana
11	saito	saito	saito	saito	saito
12	mokuya	kuya	kuya	kuya	kuya
13	kita	kita	kita	kita	kita
14	nesana	sana	sana	sana	sana
15	ira	ira	ira	ira	ira
16	papitupa	papitu	papitu	papitu	papitu
17	modanda	danda	danda	danda	danda
18	panto	panto	panto	panto	panto
19	anumo	anumo	anumo	anumo	anumo
20	tano	tano	tano	tano	tano

Tabel 4.8 (Lanjutan)

21	alamo	ala	ala	ala	ala
22	mbo	mbo	mbo	mbo	mbo
23	sangu	sangu	sangu	sangu	sangu
24	vatu	vatu	vatu	vatu	vatu
25	niala	ala	ala	ala	ala
26	bo	bo	bo	bo	bo
27	hifau	hau	hau	hau	hau
28	nombatinda	tinda	tinda	tinda	tinda
29	nontindapa	nontindapa	nontindapa	nontindapa	tinda
30	nobagiino	nobagiino	nobagiino	nobagiino	bagi
31	ampa	ampa	ampa	ampa	ampa
32	sabingga	bingga	bingga	bingga	bingga
33	sanga	sanga	sanga	sanga	sanga
34	i	i	i	i	i
35	jali	jali	jali	jali	jali
36	roca	roca	roca	roca	roca
37	bali	bali	bali	bali	bali
38	laba	laba	laba	laba	laba
39	muni	muni	muni	muni	muni
40	nosanga	sanga	sanga	sanga	sanga
41	gasi	gasi	gasi	gasi	gasi
42	gala	gala	gala	gala	gala

Tabel 4.8 (Lanjutan)

43	palu	palu	palu	palu	palu
44	jama	jama	jama	jama	jama
45	nanguli	uli	uli	uli	uli
46	mai	mai	mai	mai	mai
47	isema	isema	isema	isema	isema
48	mosimpesavi	mosimpesavi	mosimpesavi	mosimpesavi	savi
49	aku	aku	aku	aku	aku
50	nasimbayu	simbayu	simbayu	simbayu	simbayu
51	kambaso	mbaso	mbaso	mbaso	mbaso
52	ivesia	ivesia	ivesia	ivesia	ivesia
53	tonabolina	tonabolina	tonabolina	tonabolina	tonabolina
54	niontiramo	niontiramo	niontiramo	niontiramo	niontiramo
55	konoana	konoana	konoana	konoana	kono
56	dopa	dopa	dopa	dopa	dopa
57	nasae	sae	sae	sae	sae
58	gei	gei	gei	gei	gei
59	jalimo	jalimo	jalimo	jalimo	jalimo
60	meamba	amba	amba	amba	amba
61	mokaulu	mokaulu	mokaulu	mokaulu	ulu
62	pade	pade	pade	pade	pade
63	puri	puri	puri	puri	puri
64	neamba	amba	amba	amba	amba

Tabel 4.8 (Lanjutan)

65	niamba	amba	amba	amba	amba
66	aga	aga	aga	aga	aga
67	napara	para	para	para	para
68	nambela	mbela	mbela	mbela	mbela
69	roana	roa	roa	roa	roa
70	randuapa	randua	randua	randua	randua
71	naavaraka	naavaraka	naavaraka	naavaraka	ava
72	pura	pura	pura	pura	pura
73	naupupa	upu	upu	upu	upu
74	komiupa	komiu	komiu	komiu	komiu
75	nompakambela	mbela	mbela	mbela	mbela
76	nokaulu	nokaulu	nokaulu	nokaulu	ulu
77	niambana	amba	amba	amba	amba
78	kaliu	liu	liu	liu	liu
79	liu	liu	liu	liu	liu
80	nitobena	tobe	tobe	tobe	tobe
81	amba	amba	amba	amba	amba
82	iko	iko	iko	iko	iko
83	topetobe	tobe	tobe	tobe	tobe
84	vai	vai	vai	vai	vai
85	nakavao	vao	vao	vao	vao
86	roa	roa	roa	roa	roa

Tabel 4.8 (Lanjutan)

87	tatalupa	tatalu	tatalu	tatalu	tatalu
88	namosu	mosu	mosu	mosu	mosu
89	nitompona	tompo	tompo	tompo	tompo
90	komiu	komiu	komiu	komiu	komiu
91	domo	domo	domo	domo	domo
92	ledo	ledo	ledo	ledo	ledo
93	nabali	bali	bali	bali	bali
94	peambana	amba	amba	amba	amba
95	nambelapa	mbela	mbela	mbela	mbela
96	nongare	ngare	ngare	ngare	ngare
97	nangggataka	anggataka	anggataka	anggataka	anggataka
98	palera	lera	lera	lera	lera
99	bavona	bavo	bavo	bavo	bavo
100	nesavi	savi	savi	savi	savi
101	jarara	jara	jara	jara	jara
102	vatura	vatu	vatu	vatu	vatu
103	sanggani	ni	ni	ni	nggani
104	netuu	tuu	tuu	tuu	tuu
105	ruanggani	nggani	nggani	nggani	nggani
106	saitoa	saitoa	saitoa	saitoa	saitoa
107	nadanda	danda	danda	danda	danda

Dari proses stemming yang dilakukan terhadap cerita Nodanda, baik algoritma 14, algoritma 17 maupun algoritma 18 mendapatkan hasil yang sama yaitu 100 kata benar dan 7 kata salah dari 107 kata sehingga tingkat akurasi sebesar 93,46%.

Berdasarkan pengujian yang telah dilakukan terhadap Algoritma 14, algoritma 17 dan algoritma 18 menggunakan 5 cerita dalam bahasa kaili, jika diakumulasikan pada masing-masing algoritma terdapat 556 kata yang distemming dengan benar dan 38 kata yang salah dari 594 kata. Sehingga secara keseluruhan tingkat akurasi dari masing-masing algoritma mencapai 93,60%.

4.4. Evaluasi

Pada pengujian pertama dilakukan pengujian terhadap 24 algoritma menggunakan 1000 kata dalam bahasa kaili untuk mengetahui algoritma yang tingkat akurasi yang paling tinggi. Dari hasil pengujian yang dilakukan, didapatkan algoritma 14 merupakan algoritma dengan tingkat akurasi yang paling baik yang mencapai 96,6% dengan 966 kata distemming dengan benar dan 34 kata yang mengalami kesalahan stemming. Setelah dilakukan analisis terhadap kata yang salah, didapatkan data sebagai berikut:

Tabel 4.9 Data kesalahan stemming algoritma 14

No	Kata	Kata Dasar	Hasil Stemming	Jenis Kesalahan
1	sangani	Nggani	ni	Overstemming
2	nombasa	Basa	sa	Overstemming
3	nakabakua	Kabaku	bakua	Spelling Exception
4	kadempena	Kadempe	dempe	Overstemming

Tabel 4.9 (Lanjutan)

5	nakadiki	Kadiki	diki	Overstemming
6	nakaja	Kaja	Ja	Overstemming
7	nakakata	Kakata	kata	Overstemming
8	nosigaligoti	Kaligoti	nosigaligoti	Unchange
9	nakalosu	Kalosu	losu	Overstemming
10	Kalua	Kalu	lua	Overstemming
11	nakamao	Kamao	mao	Overstemming
12	kambana	Kamba	mbana	Spelling Exception
13	nakambuu	Kambuu	mbuu	Overstemming
14	nakamumu	Kamumu	mumu	Overstemming
15	Nakana	Kana	na	Overstemming
16	pangande	Kande	gande	Spelling Exception
17	nakanonggo	Kanonggo	nonggo	Overstemming
18	Nakape	Kape	pe	Overstemming
19	Nakapo	Kapo	po	Overstemming
20	nakapoo	Kapoo	poo	Overstemming
21	nakaposu	Kaposu	posu	Overstemming
22	Nakara	Kara	ra	Overstemming
23	nakarama	Karama	rama	Overstemming
24	karavana	Karava	ravana	Spelling Exception
25	nakareke	Kareke	reke	Overstemming
26	nakasusu	Kasusu	susu	Overstemming
27	nakatano	Katano	tano	Overstemming
28	nakatepu	Katepu	tepu	Overstemming
29	Nakavao	Kavao	vao	Overstemming
30	Nakaya	Kaya	ya	Overstemming
31	Samata	Mata	ata	Overstemming
32	Nanasa	Nasa	asa	Overstemming
33	nanggola	Nggola	gola	Overstemming
34	nanjulebe	Sulebe	nanjulebe	Unchange

Berdasarkan data pada tabel 4.9 kesalahan overstemming sebanyak 28 kata atau 82% dari jumlah kesalahan stemming, spelling exception sebanyak 4 kata atau 12% dari jumlah kesalahan stemming serta unchange sebanyak 2 kata atau 6% dari jumlah kesalahan stemming. Jika dibandingkan dengan seluruh data maka kesalahan overstemming sebesar 2,8% dari keseluruhan data, kesalahan spelling

exception sebesar 0,4% dari keseluruhan data serta kesalahan unchange sebesar 0,2% dari keseluruhan data.

Hasil analisis terhadap data kesalahan stemming didapatkan bahwa kesalahan overstemming adalah kesalahan yang paling banyak terjadi, hal ini karena penempatan posisi aturan imbuhan yang menyebabkan terjadinya pemotongan imbuhan yang lebih panjang dari seharusnya. Tetapi penempatan posisi aturan imbuhan algoritma 14 adalah yang terbaik karena tingkat akurasi yang lebih baik dibandingkan dengan penempatan posisi aturan imbuhan pada algoritma yang lain.

Kesalahan lain yang terjadi adalah spelling exception dan unchange, kesalahan ini karena perubahan bunyi dari proses imbuhan sehingga kata dasar tidak dapat terdeteksi dengan benar. Perbedaannya adalah pada spelling exception terjadi perubahan pada hasil stemming dibandingkan kata awal tetapi kata yang dihasilkan tidak tepat, sedangkan pada kesalahan unchange proses stemming mendeteksi kata tersebut sebagai kata dasar secara utuh.

Kemudian dilakukan pengujian terhadap algoritma 14, algoritma 17 dan algoritma 18 menggunakan 5 cerita dalam bahasa kaili untuk mengetahui tingkat akurasi dari masing-masing algoritma jika digunakan untuk melakukan stemming terhadap tulisan dalam bahasa kaili. Pengujian yang dilakukan terhadap ketiga algoritma didapatkan hasil yang sama persis, akumulasi dari pengujian yang dilakukan terhadap 5 cerita dalam bahasa kaili tingkat akurasi mencapai 93,60% dengan 556 kata distemming dengan benar dan 38 kata yang mengalami kesalahan

stemming dari 594 kata. Setelah dilakukan analisis terhadap kata yang salah, didapatkan data sebagai berikut:

Tabel 4.10 Data kesalahan pengujian

No	Cerita	Kata	Hasil Stemming	Menurut Kamus	Jenis Kesalahan
1	Topomeka	panggavia	panggavia	via	Unchange
2		nipesavira	nipesavira	savi	Unchange
3		manggava	manggava	kava	Unchange
4		tatanganana	tatanganana	tatangana	Unchange
5	Sikola	mpadondo	mpadondo	dondo	Unchange
6		nosikola	kola	sikola	Overstemming
7		nangulimo	guli	uli	Overstemming
8		sanggani	ni	nggani	Overstemming
9		raepena	raepena	epe	Unchange
10		ntomana	ntomana	ntoma	Unchange
11		kupatudukika	kupatudukika	patuduki	Unchange
12		manjili	ili	sili	Overstemming
13		nanau	au	nau	Overstemming
14		nggalako	nggalako	lako	Unchange
15	kaupunapa	kaupunapa	upu	Unchange	
16	Sapeda	ngande	ngande	kande	Unchange
17		nanjili	ili	sili	Overstemming
18		nangulimo	guli	uli	Understemming
19		rapoutaka	rapoutaka	putaka	Unchange

Tabel 4.10 (Lanjutan)

20		panipara	panipara	para	Unchange
21		nitudaramo	nitudaramo	tuda	Unchange
22		kapojarita	kapojarita	jarita	Unchange
23		raasukita	raasukita	asu	Unchange
24		pangulimiu	pangulimiu	uli	Unchange
25	Noasu	nangulimo	guli	uli	Overstemming
	Bava	nanggeni	nanggeni	keni	Spelling Exception
26		nijagaira	nijagaira	jaga	Unchange
27		nitipanakara	nitipanakara	tipa	Unchange
28		panggaviara	panggaviara	panggavia	Unchange
29		poviara	poviara	via	Unchange
30		nipoviara	nipoviara	via	Unchange
31		nontindapa	nontindapa	tinda	Unchange
32		mosimpesavi	mosimpesavi	savi	Unchange
33		konoana	konoana	kono	Unchange
34	Nodanda	mokaulu	mokaulu	ulu	Unchange
35		naavaraka	naavaraka	ava	Unchange
36		nokaulu	nokaulu	ulu	Unchange
37		sangani	ni	nggani	Overstemming
38					

Berdasarkan data pada tabel 4.10 kesalahan overstemming sebanyak 8 kata atau 21,05% dari jumlah kesalahan stemming, spelling exception sebanyak 1 kata

atau 2,63% dari jumlah kesalahan stemming, under stemming sebanyak 1 kata atau 2,63% dari jumlah kesalahan stemming serta unchange sebanyak 28 kata atau 73,68% dari jumlah kesalahan stemming. Jika dibandingkan dengan seluruh data maka kesalahan overstemming sebesar 1,35% dari keseluruhan data, kesalahan spelling exception sebesar 0,17% dari keseluruhan data, kesalahan understemming sebesar 0,17% dari keseluruhan data serta kesalahan unchange sebesar 73,68% dari keseluruhan data.

Hasil analisis terhadap data kesalahan stemming didapatkan bahwa kesalahan paling banyak terjadi pada jenis kesalahan unchange, hal tersebut dikarenakan terdapat kata imbuhan yang digunakan pada cerita tetapi tidak terdapat pada jenis imbuhan di buku "Morfologi dan Sintaksis Bahasa Kaili" yang dijadikan sebagai acuan dalam penyusunan algoritma, sehingga tidak dapat dideteksi oleh algoritma stemming.

Kesalahan lain yang cukup banyak yaitu overstemming hal ini karena penempatan posisi aturan imbuhan yang menyebabkan terjadinya pemotongan imbuhan yang lebih panjang dari seharusnya. Sedangkan untuk kesalahan understemming dan spelling exception, kesalahan ini karena perubahan bunyi dari proses imbuhan sehingga kata dasar tidak dapat terdeteksi dengan benar.

BAB V

PENUTUP

5.1. Kesimpulan

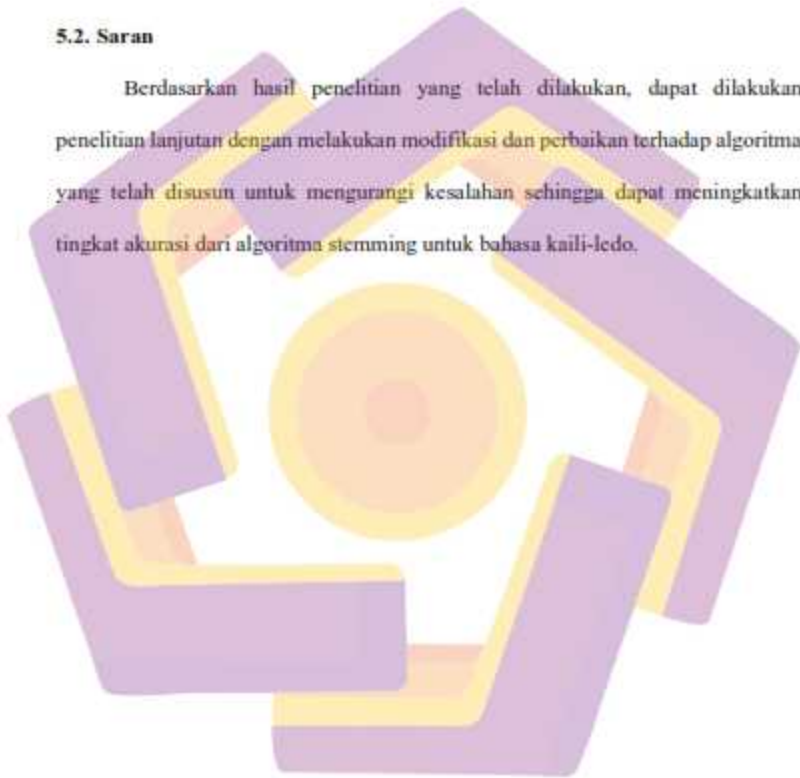
Dari penelitian yang telah dilakukan, dapat disimpulkan bahwa kontribusi yang dihasilkan adalah sebagai berikut:

- a. Dihasilkan dataset berupa kumpulan kata dasar dalam bahasa Kaili-Ledo sebanyak 4127 kata (data terlampir)
- b. Dari hasil analisis imbuhan dalam bahasa Kaili-Ledo, dihasilkan algoritma hasil modifikasi dari algoritma nazief dan adriani yang dapat disusun dalam 24 alternatif algoritma.
- c. Berdasarkan dua langkah pengujian yang dilakukan, didapatkan hasil sebagai berikut:
 1. Pengujian pertama dilakukan terhadap 24 algoritma yang diuji menggunakan 1000 kata dalam bahasa Kaili-Ledo, hasil yang didapatkan adalah algoritma 14 merupakan algoritma terbaik dengan tingkat akurasi sebesar 96,60%.
 2. Pengujian kedua dilakukan dengan menguji algoritma 14, algoritma 17 dan algoritma 18 yang merupakan tiga algoritma teratas dari hasil pengujian pertama dengan 5 cerita dalam bahasa Kaili-Ledo. Dari hasil pengujian yang dilakukan, ketiga algoritma mempunyai tingkat akurasi dari akumulasi hasil stemming mencapai 93,60%.

Dari hasil pengujian diatas dapat disimpulkan bahwa algoritma 14, algoritma 17 maupun algoritma 18 cukup baik digunakan untuk melakukan stemming bahasa Kaili-Ledo.

5.2. Saran

Berdasarkan hasil penelitian yang telah dilakukan, dapat dilakukan penelitian lanjutan dengan melakukan modifikasi dan perbaikan terhadap algoritma yang telah disusun untuk mengurangi kesalahan sehingga dapat meningkatkan tingkat akurasi dari algoritma stemming untuk bahasa kaili-ledo.



DAFTAR PUSTAKA

PUSTAKA BUKU

- Sofyan, A.I., Kaseng, S., Sikki, M and Pepy, P., 1979, Morfologi dan Sintaksis Bahasa Kaili, Pusat Pembinaan dan Pengembangan Bahasa Departemen Pendidikan dan Kebudayaan.
- Evans, D., 2003, Kamus Kaili-Ledo – Indonesia – Inggris, Pemerintah Daerah Propinsi Sulawesi Tengah, Dinas Kebudayaan dan Pariwisata.
- Suparno, D., 2015, Morfologi Bahasa Indonesia, Universitas Islam Negeri Syarif Hidayatullah Jakarta.
- Pantuobo, T., 1991, Bula Dongga, Balai Pustaka.
- Liddy, E.D. 2001. Natural Language Processing. In Encyclopedia of Library and Information Science, 2nd Ed. NY: Marcel Decker, Inc.

PUSTAKA MAJALAH, JURNAL ILMIAH ATAU PROSIDING

- Jumadi, J., Maylawati, D.S., Pratiwi, L.D. and Ramdhani, M.A., 2021, March. Comparison of Nazief-Adriani and Paice-Husk algorithm for Indonesian text stemming process. In IOP Conference Series: Materials Science and Engineering (Vol. 1098, No. 3, p. 032044). IOP Publishing.
- Wardani, N.W. and Nugraha, P.G.S.C., 2020. Stemming Teks Bahasa Bali dengan Algoritma Enhanced Confix Stripping. International Journal of Natural Science and Engineering, 4(3), pp.103-113.
- Nq, M.A., Manik, L.P. and Widiyatmoko, D., 2020, December. Stemming Javanese: Another Adaptation of the Nazief-Adriani Algorithm. In 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI) (pp. 627-631). IEEE.
- Wibawa, A.P., Dwiyanto, F.A., Zaeni, I.A.E., Nurrohman, R.K. and Afandi, A.N., 2020. Stemming javanese affix words using nazief and adriani modifications. J. Inform., 14(1), p.36.
- Rumilah, S. and Cahyani, I., 2020. STRUKTUR BAHASA; Pembentukan Kata dan Morfem sebagai Proses Morfemis dan Morfonemik dalam Bahasa Indonesia. Jurnal Pendidikan Bahasa Indonesia, 8(1), pp.70-87.

- Yusliani, N., Primartha, R. and Marieska, M.D., 2019. Multiprocessing Stemming: A Case Study of Indonesian Stemming. *International Journal Computer and Applications (IJCA)*, 182(40), pp.15-19.
- Muchtar, M.A., Nababan, E.B., Nababan, M., Andayani, U., Simanjuntak, T. and Sitompul, O.S., 2019, October. Implementation of Porter Stemmer Algorithm to Obtain Basic Words in Toba Batak Language Documents with the Two-Level Morphological Method. In *IOP Conference Series: Materials Science and Engineering* (Vol. 648, No. 1, p. 012025). IOP Publishing.
- Maylawati, D.S.A., Zulfikar, W.B., Slamet, C., Ramdhani, M.A. and Gerhana, Y.A., 2018, August. An improved of stemming algorithm for mining indonesian text with slang on social media. In *2018 6th International Conference on Cyber and IT Service Management (CITSM)* (pp. 1-6). IEEE.
- Putra, R.B.S. and Utami, E., 2018, March. Non-formal affixed word stemming in Indonesian language. In *2018 International Conference on Information and Communications Technology (ICOIACT)* (pp. 531-536). IEEE.
- Prasidhatama, A. and Suryaningrum, K.M., 2018. Perbandingan Algoritma Nazief & Adriani Dengan Algoritma Idris Untuk Pencarian Kata Dasar. *Jurnal Teknologi dan Manajemen Informatika*, 4(1).
- Adriani, M., Asian, J., Nazief, B., Tahaghoghi, S.M.M., and Williams, H.E. (2007). Stemming Indonesian: A Confix-Stripping Approach. *ACM Transactions on Asian Language Information Processing*, Vol. 6, No. 4, Article 13.
- Mandala, R., Koryanti, E., Munir, R., & Harlili, H. (2004). Sistem Stemming Otomatis untuk kata dalam Bahasa Indonesia. In *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*.

PUSTAKA LAPORAN PENELITIAN

- Kodimala, S., 2010, Study of stemming algorithms, Thesis, Master of Science in Computer Science, University of Nevada Las Vegas.

LAMPIRAN

Daftar Kata Dasar

<https://is.gd/5qXsxo>

Daftar 1000 Kata Pengujian

<https://is.gd/EdpOCw>

Daftar Cerita Dalam Bahasa Kaili

Topomeka

<https://is.gd/XZBw0q>

Sikola

<https://is.gd/TJYRRz>

Sapeda

<https://is.gd/UL9UXQ>

Noasu Bavu

<https://is.gd/u9Z6ho>

Nodanda

<https://is.gd/OJSTff>