

APLIKASI PENENTUAN KATA DASAR DARI KATA BERIMBUHAN PADA KALIMAT BAHASA INDONESIA DENGAN ALGORITMA STEMMING

Julianto Wibowo

Mahasiswa Program Studi Teknik Informatika, STMIK Budi Darma Medan
 Jl. Sisimangaraja No.338 Simpang Limun Medan
 http : //www.stmik-budidarma.ac.id // Email : wibowojulianto@yahoo.com

ABSTRAK

Bahasa Indonesia kaya akan imbuhan. Kurang lebih ada sekitar 35 imbuhan resmi yang disebutkan dalam kamus Besar Bahasa Indonesia. Imbuhan-imbuhan ini dapat berupa prefix (awalan), sufiks (akhiran), konfiks maupun infiks (sisipan) yang diserap dari Bahasa Jawa. Algoritma stemming yaitu suatu teknik pencarian bentuk dasar dari suatu term. Yang dimaksud dengan term itu sendiri adalah tiap kata yang berada pada suatu dokumen teks. Di dalam algoritma stemming terdapat algoritma Nazief dan Adriani yaitu salah satu algoritma yang digunakan dalam tahap stemming. Algoritma Nazief dan Adriani merupakan algoritma untuk mengubah kata yang memiliki sufiks, prefiks dan konfiks menjadi bentuk kata dasar. Algoritma Nazief dan Adriani digunakan dalam penelitian ini sebagai algoritma pendukung dalam proses penentuan kata dasar dari kata berimbuhan dalam sebuah kalimat. Untuk menentukan kata dasar dari kata berimbuhan ini dengan menghilangkan semua imbuhan (afiks) baik yang terdiri dari awalan (prefiks), sisipan (infiks), akhiran (sufiks) dan awalan-akhiran (konfiks) pada kata berimbuhan dengan algoritma stemming Nazief dan Adriani serta dapat mengetahui kelas kata dari kata dasar yang berimbuhan tersebut.

Kata Kunci : Bahasa Indonesia, Algoritma Stemming, Nazief dan Adriani

1. Pendahuluan

1.1 Latar Belakang Masalah

Bahasa Indonesia kaya akan imbuhan. Kurang lebih ada sekitar 35 imbuhan resmi yang disebutkan dalam kamus Besar Bahasa Indonesia. Imbuhan-imbuhan ini dapat berupa prefix (awalan), sufiks (akhiran), konfiks maupun infiks (sisipan) yang diserap dari Bahasa Jawa. Satu hal yang unik dari Bahasa Indonesia adalah kecenderungan pemakaian imbuhan secara bebas, secara gamblang, dapat dikatakan bahwa imbuhan-imbuhan dalam bahasa Indonesia dapat digunakan pada semua kata dan imbuhan tersebut dapat dikombinasikan satu dengan lainnya.

Dalam suatu bahasa proses pembentukan kata dapat dilakukan melalui lima cara yaitu afiksasi (misalnya ajar menjadi belajar), duplikasi (misalnya meja menjadi meja-meja), modifikasi Intern (misalnya perubahan dalam bahasa Arab *dhoroba-yadhribu-idhrib*), komposisi (misalnya mata menjadi matahari) dan klitisasi (misalnya walau menjadi walaupun).

Dalam dunia pemrograman terdapat sebuah algoritma *stemming* yang dapat digunakan untuk penentuan kata dasar dari sebuah kata berimbuhan.

Adapun pengertian dari algoritma *stemming* yaitu suatu teknik pencarian bentuk dasar dari suatu *term*. Yang dimaksud dengan *term* itu sendiri adalah tiap kata yang berada pada suatu dokumen teks. *Stemming* dilakukan pada saat pembuatan indeks dari suatu dokumen. Pembuatan indeks dilakukan karena suatu dokumen tidak dapat dikenali langsung

oleh suatu sistem. Dokumen tersebut terlebih dahulu perlu dipetakan kedalam suatu representasi dengan menggunakan teks yang berada di dalamnya.

Di dalam algoritma *stemming* terdapat algoritma Nazief dan Adriani yaitu salah satu algoritma yang digunakan dalam tahap *stemming*. Algoritma Nazief dan Adriani merupakan algoritma untuk mengubah kata yang memiliki sufiks, prefiks dan konfiks menjadi bentuk kata dasar. Algoritma Nazief dan Adriani digunakan dalam penelitian ini sebagai algoritma pendukung dalam proses penentuan kata dasar dari kata berimbuhan dalam sebuah kalimat.

Teknik *stemming* diperlukan selain untuk memperkecil jumlah indeks yang berbeda dari suatu dokumen, juga untuk melakukan pengelompokan kata-kata lain yang memiliki kata dasar dan arti yang serupa namun memiliki bentuk atau form yang berbeda karena mendapatkan imbuhan yang berbeda.

1.2 Perumusan Masalah

Adapun rumusan yang diuraikan pada penelitian ini adalah :

1. Bagaimana proses pembentukan kata dasar menjadi kata berimbuhan pada kalimat Bahasa Indonesia?
2. Bagaimana menerapkan algoritma *stemming* dalam penentuan kata dasar dan kelas kata dari kata berimbuhan pada kalimat Bahasa Indonesia?
3. Bagaimana merancang aplikasi penentuan kata dasar dari kata berimbuhan pada kalimat Bahasa Indonesia dengan algoritma *stemming*?

1.3 Batasan Masalah

Beberapa batasan masalah yang digunakan untuk memfokuskan lingkup pembahasan dan mempertajam pemahaman tentang sistem yang akan dibuat, adalah sebagai berikut:

1. Pembahasan hanya fokus pada algoritma *stemming* yaitu Nazief dan Adriani.
2. Proses *stemming* yang dilakukan adalah untuk penghilang awalan dan akhiran saja tidak untuk penghilangan imbuhan sisipan.
3. Penentuan kelas kata dibatasi hanya untuk kata kerja, kata sifat, dan kata benda.
4. Aplikasi program ini menggunakan Visual Basic Net 2008 dengan Database *MySQL*.

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

1. Menjelaskan proses pembentukan kata dasar menjadi kata berimbuhan pada kalimat Bahasa Indonesia.
2. Menerapkan algoritma *stemming* dalam penentuan kata dasar dan kelas kata dari kata berimbuhan pada kalimat Bahasa Indonesia.
3. Merancang aplikasi penentuan kata dasar dari kata berimbuhan pada kalimat Bahasa Indonesia dengan algoritma *stemming*.

1.4.2 Manfaat Penelitian

Sedangkan manfaat dari penelitian ini adalah :

1. Penelitian ini diharapkan dapat membantu dalam hal penentuan kata dasar dari suatu kata berimbuhan untuk melakukan pemilihan kata yang benar dan tepat dalam penggunaan kata dasar dan penentuan kelas kata yang merupakan hasil dari proses *stemming*.
2. Penelitian ini juga diharapkan dapat membantu pelajar dalam menyelesaikan karya tulis dengan mengetahui kata dasar dari kata berimbuhan dan mengetahui kelas kata dari kata tersebut.

2. Landasan Teori

2.1 Aplikasi

Aplikasi adalah program komputer yang di pakai untuk melakukan pekerjaan yang dapat membantu manusia dalam pekerjaannya yang berhubungan dengan komputer, dengan tujuan agar pekerjaan tersebut bisa lebih cepat, tepat dan efisien.

2.2 Pengertian Kata Dasar

Kata dasar diartikan sebagai suatu kata yang berdiri sendiri, kata yang berdiri sendiri disini merupakan suatu kata yang tidak mengandung imbuhan baik itu berupa awalan maupun berupa akhiran.

2.3 Kata Berimbuhan

Kata berimbuhan adalah kata dasar yang mendapatkan tambahan imbuhan baik awalan, akhiran, sisipan atau gabungan yang biasa disebut *morfem terikat*.

2.4 Algoritma Stemming

Stemming adalah salah satu cara yang digunakan untuk meningkatkan performa IR (*Information Retrieval*) dengan cara mentransformasi kata-kata dalam sebuah dokumen teks ke bentuk kata dasarnya.

2.5 Algoritma Stemming Nazief dan Adriani

Algoritma Nazief & Adriani sebagai algoritma *stemming* untuk teks berbahasa Indonesia yang memiliki kemampuan prosentase keakuratan (presisi) lebih baik dari algoritma lainnya. Algoritma ini sangat dibutuhkan dan menentukan dalam proses IR dalam dokumen Indonesia.

Pada umumnya kata dasar pada bahasa Indonesia terdiri dari kombinasi:

DP + DP + ROOT WORD + DS + PP

Algoritma Nazief & Adriani yang dibuat oleh Bobby Nazief dan Mirna Adriani ini memiliki tahap-tahap sebagai berikut:

1. Cari kata yang akan diistem dalam kamus. Jika ditemukan maka diasumsikan kata adalah *root word*. Maka algoritma berhenti.
2. *Inflection Suffixes* (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) dibuang. Jika berupa *particles* (“-lah”, “-kah”, “-tah” atau “-pun”) maka langkah ini diulangi lagi untuk menghapus *Possesive Pronouns* (“-ku”, “-mu”, atau “-nya”), jika ada.
3. Hapus *Derivation Suffixes* (“-i”, “-an” atau “-kan”). Jika kata ditemukan di kamus, maka algoritma berhenti. Jika tidak maka ke langkah 3a
 - a. Jika “-an” telah dihapus dan huruf terakhir dari kata tersebut adalah “-k”, maka “-k” juga ikut dihapus. Jika kata tersebut ditemukan dalam kamus maka algoritma berhenti. Jika tidak ditemukan maka lakukan langkah 3b.
 - b. Akhiran yang dihapus (“-i”, “-an” atau “-kan”) dikembalikan, lanjut ke langkah 4.
4. Hapus *Derivation Prefix*. Jika pada langkah 3 ada sufiks yang dihapus maka pergi ke langkah 4a, jika tidak pergi ke langkah 4b.
 - a. Periksa tabel kombinasi awalan-akhiran yang tidak diijinkan. Jika ditemukan maka algoritma berhenti, jika tidak
 - b. pergi ke langkah 4b.
 - c. For $i = 1$ to 3, tentukan tipe awalan kemudian hapus awalan. Jika *root word* belum juga ditemukan lakukan langkah 5, jika sudah maka algoritma berhenti. Catatan: jika awalan kedua sama dengan awalan pertama algoritma berhenti.
5. Melakukan Recoding.
6. Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai *root word*. Proses selesai.

Tipe awalan ditentukan melalui langkah-langkah berikut:

1. Jika awalannya adalah: “di-”, “ke-”, atau “se-” maka tipe awalannya secara berturut-turut adalah “di-”, “ke-”, atau “se-”.

2. Jika awalannya adalah “te-”, “me-”, “be-”, atau “pe-” maka dibutuhkan sebuah proses tambahan untuk menentukan tipe awalannya.
3. Jika dua karakter pertama bukan “di-”, “ke-”, “se-”, “te-”, “be-”, “me-”, atau “pe-” maka berhenti.
4. Jika tipe awalan adalah “none” maka berhenti. Jika tipe awalan adalah bukan “none”.

Tabel 1. Kombinasi Awalan Akhiran Yang Tidak Diiijinkan

Awalan	Akhiran yang tidak diizinkan
be-	-i
di-	-an
ke-	-i, -kan
me-	-an
se-	-i, -kan

Tabel 2 :Cara Menentukan Tipe Awalan Untuk Awalan “te-“

Following Characters				Tipe Awalan
Set 1	Set 2	Set 3	Set 4	
“-x-“	“-x-“	–	–	None
“-x-“	–	–	–	ter-luluh
“-x-“	not (vowel or “-x-“)	“-er-“	Vowel	Ter
“-x-“	not (vowel or “-x-“)	“-er-“	not vowel	ter-
“-x-“	not (vowel or “-x-“)	not “-er-“	–	Ter
not (vowel or “-x-“)	“-er-“	Vowel	–	None
not (vowel or “-x-“)	“-er-“	not vowel	–	Te

Tabel 3 .Jenis Awalan Berdasarkan Tipe Awalannya

Tipe Awalan	Awalan yang harus dihapus
di-	di-
ke-	ke-
se-	se-
te-	te-
ter-	ter-
ter-luluh	Ter

Untuk mengatasi keterbatasan pada algoritma di atas, maka ditambahkan aturan-aturan dibawah ini:

1. Aturan untuk duplikasi.
 - a. Jika kedua kata yang dihubungkan oleh kata penghubung adalah kata yang sama maka *root word* adalah bentuk tunggalnya, contoh : “buku-buku” *root word*-nya adalah “buku”.
 - b. Kata lain, misalnya “bolak-balik”, “berbalas-balasan, dan ”seolah-olah”. Untuk mendapatkan *root word*-nya, kedua kata diartikan secara terpisah. Jika keduanya memiliki *root word* yang sama maka diubah menjadi bentuk tunggal, contoh: kata “berbalas-balasan”, “berbalas” dan “balasan” memiliki *root word* yang sama yaitu “balas”, maka *root word* “berbalas-balasan” adalah

“balas”. Sebaliknya, pada kata “bolak-balik”, “bolak” dan “balik” memiliki *root word* yang berbeda, maka *root word*-nya adalah “bolak-balik”.

2. Tambahan bentuk awalan dan akhiran serta aturannya.
 - a. Untuk tipe awalan “mem-“, kata yang diawali dengan awalan “memp-” memiliki tipe awalan “mem-”.
 - b. Tipe awalan “meng-“, kata yang diawali dengan awalan “mengk-” memiliki tipe awalan “meng-”.

3. Analisa dan Pembahasan

3.1 Analisa Kata Dasar

Pada aplikasi ini kata dasar dibuat sebagai database kata dasar yang merupakan pusat dari suatu daftar kata dasar yang nantinya akan digunakan sebagai acuan untuk penentuan kata dasar dari sebuah kata yang dilakukan proses stemming. Adapun hasil dari penentuan kata dasar ini tergantung dari ada tidaknya kata tersebut dalam database kata dasar. Sehingga untuk pembatasan pada bagian kata dasar ini berupa tidak semua kata yang mengalami proses *stemming* dapat berhasil dicarikan kata dasarnya. Untuk keberhasilan penentuan kata dasar tersebut dipengaruhi oleh daftar dari kata dasar yang ada database kata dasar dengan proses hasil *stemming* itu sendiri.

Database kata dasar disini berfungsi sebagai daftar kumpulan-kumpulan kata dasar yang akan digunakan pada proses pencarian kata dasar dengan menggunakan algoritma *stemming*. Pada database kata dasar ini terdiri dari beberapa bagian yang berfungsi untuk mendukung dalam hal penentuan bentuk kata. Contoh kata memakan memiliki kata dasar makan.

3.2 Penerapan Algoritma Stemming

Algoritma ini menggunakan algoritma *stemming* yang dibahas pada bagian sebelumnya dari algoritma Nazief dan Adriani. Selain itu adanya penambahan langkah-langkah pada proses *stemming* untuk mendapatkan kata dasar dari kata berimbuhan. Adapun penambahan itu berupa pembagian kata berimbuhan yang dikelompokkan menjadi empat bagian sesuai dengan imbuhan yang dimiliki oleh kata berimbuhan tersebut.

Adapun rumusan dari algoritma *stemming* ini yaitu Nazief dan Adriani:

$$DP + DP + \text{ROOT WORD} + DS + PP$$

Keterangan :

DP: derivational prefix

DS: derivational suffix

PP: prossesive pronoun

Mem-pe-kerja-kan-nya

Keterangan:

1. Derivation prefiks I = Mem

2. Derivation prefiks II = pe

3. Root word = kerja
4. Derivation suffiks = kan
5. Possessive pronoun = nya

Adapun proses dari tahap pemotongan kata pada kata berimbuhan diatas dapat dilihat dari penjelasan pada table dibawah ini :

Tabel 4 . Tahapan Algoritma Nazief dan Adriani

Tahap	Proses	Kata asal	Hasil potong	Kata akhir
1	Cek Database	Mempekerjakannya	-	Mempekerjakannya
2	Buang partikel (-nya)	Mempekerjakannya	Mempekerjakan	Mempekerjakan
3	Buang partikel (-kan)	Mempekerjakan	Mempekerja	Mempekerja
4	Buang awalan I (-mem)	Mempekerja	Pekerja	Pekerja
5	Buang awalan II (-per)	Pekerja	Kerja	Kerja
6	Cek Database	Kerja	-	Kerja

4. Algoritma dan Implementasi

4.1 Algoritma

Algoritma merupakan urutan langkah-langkah logis dalam memecahkan masalah yang disusun secara sistematis. Algoritma merupakan suatu cara yang digunakan untuk memperoleh dan menerangkan suatu keadaan tertentu sehingga dapat lebih dimengerti dalam suatu perancangan.

4.2 Algoritma penentuan kata dasar dari kata berimbuhan

DP + DP + ROOT WORD + DS + PP

Input : i : Input teks.

Output : k : Kata

Rw : Root word (Kata dasar)

kk : kelas kata

Dp :Derivation Prefiks (imbuhan awal)

Ds : Derivation Surfiks (imbuhan akhir)

Dp = Dp + Root Word

Ds = Root Word + Ds

Proses : Do while i

If i= Root word then

Rw=i

Elseif i <> Root word then

If i termasuk kelas kata 1 then

If i terdapat Dp then

i= hapus Dp

Elseif

i = hapus Ds

End if

If i termasuk Rw then

Rw = i

End if

Elseif i termasuk kelas kata 2 then

If i terdapat Dp then

Rw = sisa pemotongan

Dp = hasil pemotongan

Elseif i terdapat Ds then

Rw = sisa pemotongan

Ds = hasil pemotongan

End if

Elseif i termasuk kelas kata 3 then

Rw =sisa pemotongan

Dp= hasil pemotongan Dp

Ds = hasil pemotongan Ds

End if

If Rw=Root word then

Rw=Rw

End if

k = i

End while

4.3 Desain Implementasi Program

Desain implementasi system merupakan gambaran program ketika dirancang didalam bahasa pemrograman, disini aplikasi dirancang menggunakan Visual Basic 2008.

1. Halaman Form Masuk

Halaman form masuk ini adalah halaman yg akan tampil pertama kali apabila aplikasi dijalankan.



Gambar 1 : Form Masuk

2. Form Aplikasi Stemming

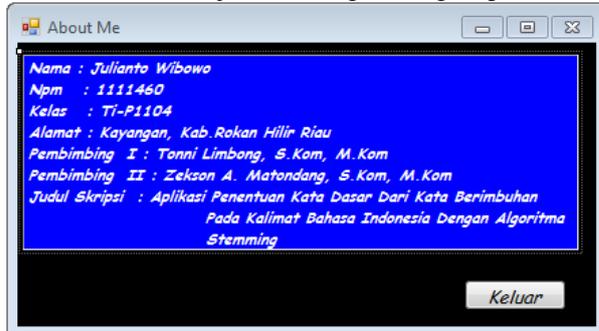
Halaman ini menampilkan form aplikasi penentuan kata dasar dari kata berimbuhan dari sistem yang dirancang.



Gambar 2 : Form Stemming

3. Form About Me

Form about me menjelaskan biografi singkat penulis.



Gambar 3 : Form About Me

5. Kesimpulan Dan Saran

5.1 Kesimpulan

Berdasarkan uraian dari bab-bab sebelumnya, maka penulis dapat memberikan kesimpulan sebagai berikut:

1. Pembentukan kata dasar menjadi kata berimbuhan dengan memberikan imbuhan awal dan di akhir kalimat.
2. Cara kerja pada algoritma *stemming* Nazief dan Adriani berupa pengambilan langkah pemotongan pertama pada semua akhiran terlebih dahulu dan selanjutnya dilakukan pemotongan awalan serta pengecekan database kata dasar pada awal dan akhir proses *stemming*.
3. Perancangan aplikasi penentuan kata dasar dari kata berimbuhan pada kalimat Bahasa Indonesia dengan algoritma *stemming* menggunakan visual studio.net 2008 dengan mysql.

5.2 Saran

Selain kesimpulan, penulis juga ingin memberikan saran yang mungkin dapat membantu dalam perbaikan penulisan adalah sebagai berikut :

1. Untuk menghasilkan kata dasar yang optimal maka penulis menyarankan agar adanya database kata dasar yang lengkap.
2. Penulis juga menyarankan agar mengembangkan aplikasi pencarian kata dasar ini dengan algoritma *stemming* lainnya lagi.
3. Untuk pengembangan aplikasi penentuan kata dasar dari kata berimbuhan ini sebaiknya menggunakan bahasa pemrograman berbasis online seperti website, web service, mobile dan lain sebagainya
4. Perancangan aplikasi yang dibangun masih banyak kekurangan dan kelemahan, oleh karena itu para pembaca dan peneliti dapat mengembangkan menjadi lebih baik lagi .

DAFTAR PUSTAKA

1. Agus Susatya, “Tata Bahasa Indonesia”, Depdiknas, 1989.
2. Alwi, “Tata Bahasa Indonesia”, 2003.
3. Andi , “Aplikasi database & cristal report pada visual basic “, 2007.

4. Andi, “Konsep dasar pengolahan dan pemrograman database”, Yogyakarta,2005.
5. Andi, “Perancangan Basis Data”, 2007.
6. Asrifin, “Intisari Bahasa dan Sastra Indonesia”, Agung Media Mulia, 2004.
7. Balai Pustaka, “Frekuensi penggunaan afiks”, 1991.
8. Dian Pratiningsih, “Ragam Bahasa Indonesia”, 2005.
9. Djon Irwanto, “Perancangan Object Oriented Software dengan UML”, Penerbit Andi, 2006.
10. Edhy Sutanta, “Pengantar Teknologi Informasi”, 2005.
11. Hendrayuni, , ”Pengertian Aplikasi”, pemrograman delphi 2008.
12. Ketut Darmayuda, *Pemrograman Aplikasi Database dengan MicrosoftVisual Basic.Net* 2008.
13. Nani Darmayanti, “Sejarah Bahasa Indonesia”, 2006.
14. Riri Hemiliawati, “Bahasa Indonesia”, Penerbit Erlangga, 2010.
15. Santoso, “Fungsi Bahasa Indonesia”, 2004.
16. Widjono Hs, “Bahasa Indonesia”, 2008.
17. Jurnal Azhar firdaus, “Aplikasi Pendeteksian Kemiripan Pada Documen Teks”, Teknologi Informasi, 2014.
18. Jurnal Ledy Agusta, “Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief dan Adriani”, Konfrensi Nasional Sistem dan Informatika, 2009.
19. Jurnal Nurfatimah Zainal, ”Kesalahan Morfologi Bahasa Indonesia”, Linguistika Akademia, 2009.