LAPORAN PENELITIAN

Stemmer Bahasa Jawa Ngoko dengan Metode *Affix Removal Stemmers* *(Rule Based Approach)*

Oleh :

Fatkhul Amin, S.T.,M.Kom	0624097201 (Ketua)
Purwatiningtyas, SE, M.Kom	0617096601 (Anggota)
Pudji Utomo, SE., M. Kom	0626016601 (Anggota)
Satria Ramadhanu	14.01.53.0026 (Anggota)
Septian Eka Cahya	14.01.53.0027 (Anggota)

FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS STIKUBANK (UNISBANK) SEMARANG
JANUARI 2016

HALAMAN PENGESAHAN
LAPORAN HASIL PENELITIAN

1. Judul Penelitian	: *Stemmer* Bahasa Jawa Ngoko dengan Metode *Affix Removal Stemmers* (*Rule Based Approach*)
2. Jenis Penelitian	: Penelitian Terapan (Applied Research)
3. a. Bidang Penelitian	: *Engineering and technology*
b. Kelompok	: 2 / 2.18 *Information Technology*
4. a. Tujuan Sosial Ekonomi	: *Advancement of knowledge*
b. Kelompok	: 20 / 20.05 *Information, Computer and Communications technologies*
5. Ketua Peneliti	
a. Nama Lengkap	: Fatkhul Amin, S.T.,M.Kom
b. Jenis Kelamin	: Laki-laki
c. NIDN / NIY	: 0624097401 / YU.2.02.10.044
d. Disiplin Ilmu	: Sistem Informasi
e. Pangkat / Golongan	: Penata Muda / IIIB
f. Jabatan Fungsional	: Asisten Ahli
g. Fakultas / Prodi	: Teknologi Informasi / Teknik Informatika
h. Alamat Kampus	: Jl. Tri Lomba Juang 1, Semarang
i. Telp/Faks/Email	: 0248311668/-/info@unisbank.ac.id
j. Alamat Rumah	: Jl. Candi Pawon Timur VI / 15 , Semarang
k. Telp/Faks/Email	: 081215156265/-/ fatkhulamin@gmail.com
6. Jumlah Anggota Peneliti	: 4 orang
a. Nama Anggota I	: Purwatiningtyas, SE, M.Kom / 0617096601
b. Nama Anggota II	: Pudji Utomo, SE., M. Kom / 0626016601
c. Mahasiswa yang terlibat	: Satria Ramadhanu /14.01.53.0026
d. Mahasiswa yang terlibat	: Septian Eka Cahya / 14.01.53.0027
7. Lokasi Penelitian	: Universitas Stikubank (Unisbank)
8. Jangka Waktu Penelitian	: Desember 2015 sd Januari 2016
9. Jumlah Biaya yang diusulkan	: Rp. 3.000.000,-

Semarang, 20 Januari 2016

Mengetahui,
Dekan Fakultas Teknologi Informasi

(Dr. Drs. Y. Suhari, M.MSI)
NIDN. 0620106502

Ketua Peneliti,

(Fatkhul Amin, S.T.,M.Kom)
NIDN. 0624097401

Menyetujui,
Kepala LPPM Unisbank

(Dr. Endang Tjahjaningsih, S.E., M.Kom)
NIDN. 0622056601

KATA PENGANTAR

Segala Puji kehadirat Allah SWT atas segala petunjukNya kepada penulis sehingga laporan penelitian berjudul "*Stemmer* Bahasa Jawa Ngoko dengan Metode *Affix Removal Stemmers* (*Rule Based Approach)* " dapat diselesaikan. Penulisan penelitian ini dapat terselesaikan karena bantuan dari berbagai pihak, serta dorongan baik berupa pikiran, ide dan sumbang saran. Oleh karena itu, pada kesempatan yang berbahagia ini penulis menyampaikan terima kasih yang sebesar-besarnya kepada;

1. Bapak **Dr. H. Hasan Abdul Rozak, S.H., C.N.,M.M.** selaku Rektor Universitas Stikubank (Unisbank) Semarang.
2. Ibu **Dr. Endang Tjahjaningsih, S.E., M.Kom,** selaku Ketua Lembaga Penelitian dan pengabdian Masyarakat (LPPM) Universitas Stikubank (Unisbank) Semarang.
3. Bapak **Dr. Drs. Y. Suhari, M.MSI,** selaku Dekan fakultas Teknologi Informasi Universitas Stikubank (Unisbank) Semarang.
4. Rekan-rekan dosen yang telah memberikan masukan-masukan untuk perbaikan dan kesempurnaan penulisan laporan ini.

Semoga laporan ini dapat bermanfaat dan menambah ilmu bagi semua, serta dapat mendukung kemajuan ilmu pengetahuan khususnya di bidang Teknologi informasi.

Semarang, 20 Januari 2016

Penulis

Stemmer* Bahasa *Jawa Ngoko
dengan Metode *Affix Removal Stemmers* (*Rule Based Approach)*

ABSTRAK

Proses pengembalian kata dasar bahasa *Jawa ngoko* dari kata jadian *(tembung andhahan)* menjadi kata dasar yang salah akan mengakibatkan kata dasar menjadi rusak, tidak memiliki arti yang benar dan tidak bisa digunakan untuk proses seteleah *Stemming*. Bahasa *Jawa Ngoko* memiliki morfologi dalam penyusunan kata yang didalamnya terdapat awalan *(ater-ater)*, sisipan *(seselan)* dan akhiran *(penambang)*. Penguasaan morfologi Bahasa Jawa ngoko membantu menciptakan cara untuk pengambilan kata dasa dalam bahasa *jawa ngoko* secara benar. *Stemmer* Bahasa *Jawa ngoko* metode *Affix Removal* digunakan untuk mendapatkan kata dasar dari hasil proses pengurangan awalan, sisipan dan akhiran secara benar. Hasil pengembalian kata dasar menggunakan *Stemmer* Bahasa *Jawa ngoko* metode *Affix Removal* mampu membuat kata dasar dalam bahasa Jawang ngoko dengan hasil benar mencapai 62 %. Kemampuan *stemmer* ini masih harus ditingkatkan sampai mencapai tingkat kebenaran dalam pengembalian kata dasar mencapai 100%.

Kata-Kunci: *Stemmer, Stemmer Jawa, Affix Removal*

DAFTAR ISI

Halaman :

DAFTAR GAMBAR

Halaman :

DAFTAR TABEL

Hal

BAB I
PENDAHULUAN

1.1. Latar Belakang

Proses pencarian pada mesin pencari terbukti efektif dan efisien dalam melaksanakan kerjanya sebagai penyedia informasi bagi penggunanya. Proses pencarian informasi menggunakan mesin pencari bisa menghasilkan hasil pencarian yang banyak namun juga bisa menghasilkan hasil pencarian yang sedikit. Hal ini karena algoritma yang digunakan berbeda. Salah satu komponen dalam proses pencarian adalah proses *stemmer*. Proses Stemmer dalam sistem temu kembali informasi *(information retrieval system)* akan berdampak kepada hasil pencarian kata yang akan sangat menentukan pada database yang dihasilkan. Proses stemmer yang tidak benar akan menghasilkan sejumlah kata yang terambil tidak benar dan tidak bisa dilakukan proses selanjutnya. Proses stemmer yang tidak benar bisa terjadi karena terlalu sedikit awalan, sisipan atau akhiran yang diambil dalam sebuah kata. Proses stemmer juga bisa salah karena terlalu banyak awalan, sisipan atau akhiran yang diambil terlalu banyak. Proses stemmer mengharuskan dibuat dengan cara mempelajari morfologi dari suatu bahasa dengan benar sehingga akan didapatkan proses pengambilan awalan, sisipan, akhiran atau kombinasinya dengan benar.

Bahasa Jawa *ngoko* memiliki morfologi yang berbeda dengan bahasa Indonesia atau bahasa negara lain dalam hal penyusunan kata. Proses penyusunan kata dalam bahasa jawa ngoko pada proses penyusunannya akan menggunakan *ater-ater* (awalan), *Seselan* (sisipan) dan *penambang* (akhiran). Jika sebuah kata dasar dalam bahasa jawa ngoko sudah mendapatkan awalan, sisipan, akhiran atau kombinasinya maka kata tersebut disebut *tembung andhahan* (kata jadian). Tembung andhahan adalah kata yang telah mendapatkan awalan, sisipan, akhiran atau kombinasi dari awalan, sisipan, akhiran. Berikut ini adalah Ater-ater (awalan) terdiri dari; *ater-ater Hanuswara* (m, n, ng, ny), *ater-ater tripurasa* (dak, ko, di) dan *ater-ater liyane* (a, ma, ka, ke, sa, pa, pi, pra, kuma, kami, kapi, tar). Seselan (sisipan) terdiri dari: *Seselan* (-um, -in, -el, lan -er). *Penambang* (akhiran) terdiri dari: *Penambang* (-i, -ake, -e, -ane, -ke, -a, -ana, -na, -ku, -mu, ,-en). Morfologi bahasa jawa *ngoko* yang berbeda dengan bahasa Indonesia memiliki keunikan dan kesulitan tersendiri dalam

proses stemmingnya. Begitu banyaknya awalan, sisipan dan akhiran dalam bahasa jawa *ngoko* membuat tingkat kesulitan yang semakin komplek dalam proses pembuatan stemmer bahasa jawa ngoko.

Solusi untuk mengatasi masalah stemmer bahasa jawa ngoko adalah dengan membuat *Stemmer* Bahasa Jawa Ngoko dengan Metode *Affix Removal Stemmers (Rule Based Approach)*. Metode *Affix Removal Stemmers (Rule Based Approach)* dipilih karena menghasilkan stemmer bahasa jawa ngoko yang tepat.

1.2. Perumusan Masalah

Bagaimana membuat *Stemmer* Bahasa Jawa Ngoko dengan Metode *Affix Removal Stemmers (Rule Based Approach)?*

BAB II
TUJUAN DAN MANFAAT PENELITIAN

2.1. Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah;

a. Membuat *Stemmer* Bahasa Jawa Ngoko dengan Metode *Affix Removal Stemmers*
b. Membuat panduan stemmer bahasa jawa ngoko yang benar sehingga bisa digunakan untuk penelitian-penelitian terkait dengan Sistem Temu Kembali Informasi.

2.2. Manfaat Penelitian

Manfaat yang dari penelitian ini adalah;

a. Memelihara Bahasa Jawa ngoko sebagai aset bangsa
b. Menanamkan kecintaan bahasa jawa kepada anak cucu
c. Mengembangkan mesin pencari dengan berbasis bahasa Jawa
d. *Stemmer* Bahasa Jawa *Ngoko* bisa digunakan untuk riset pendukung Information Retrieval System
e. Menghemat waktu pencarian informasi untuk mendapatkan dokumen yang diinginkan.

BAB III
TELAAH PUSTAKA

3.1. Penelitian Terdahulu

Penelitian terdahulu dengan judul "*Strength and Accuracy Analysis of Affix Removal Stemming Algorithms*" oleh Sandeep R. Sirsat, Dkk (2013). Telah menyimpulkan bahwa semua algoritma yang berasal dibahas dalam makalah ini relatif kuat dan agresif, tapi kurang akurat. Semua cenderung menghasilkan baik lebih-berasal dan di bawah kesalahan berasal. Namun, terjadinya bawah berasal kesalahan dalam Paice / Husk stemmer relatif rendah. The ACWF diperoleh Lovins dan Porter1 stemmer menunjukkan persentase negatif. Hal ini karena jumlah kata yang batang ke salah kata-kata adalah lebih dari benar berasal kata. Jadi dalam kedua kasus di atas-yang berasal dan di bawah-yang berasal kesalahan terjadi lebih dari yang lain. Memajukan AWCF dari *Stemmer Paice* / sekam relatif positif; masih memiliki masalah terjadinya over-berasal kesalahan, sebagai ICF dan WSF relatif tinggi. CSF dan AWCF diperoleh dengan Porter2 stemmer cukup baik, tapi itu menghasilkan lebih-berasal kesalahan karena dibandingkan dengan bawah-berasal kesalahan.

Penelitian terkait *stemmer* dengan judul "*Stemming Algorithms: A Comparative Study and their Analysis*", oleh Deepika Sharma (2012). Dalam beberapa tahun terakhir, jumlah informasi di Web telah tumbuh secara eksponensial. Informasi ini di Web praktis pada semua topik dan dalam berbagai bahasa. Beberapa bahasa ini belum menerima banyak perhatian dan untuk yang sumber daya bahasa yang langka. Untuk membuat ini informasi yang tersedia berguna, itu harus diindeks dan dibuat dicari oleh *Information Retrieval System*. Stemming adalah salah satu pendekatan tersebut digunakan dalam proses pengindeksan Kami telah menyajikan studi banding dari berbagai berasal metode. Dalam hal ini kita belajar bahwa berasal signifikan meningkatkan hasil pencarian untuk kedua berdasarkan aturan dan statistik pendekatan. Hal ini juga berguna dalam mengurangi ukuran file indeks sebagai jumlah kata untuk diindeks dikurangi untuk umum bentuk atau disebut batang. Kinerja statistik stemmers jauh lebih unggul beberapa yang terkenal berbasis aturan stemmers dan di antara

stemmers berdasarkan statistik GRAS memiliki mengungguli Yass yang merupakan pengelompokan berdasarkan akhiran stripping algoritma. Tapi kelemahan utama yang telah kita lihat di ini stemmers statistik adalah cakupan miskin bahasa yaitu mereka tidak termasuk semua dokumen di corpus untuk membuat analisis statistik seperti yang sangat memakan waktu agak mereka menganggap sampel dokumen dari corpus untuk ini analisis dan pengumpulan kecil ini dapat mengakibatkan cakupan miskin kata-kata. Kinerja GRAS juga tergantung pada kepadatan grafik tetapi penelitian telah menunjukkan bahwa ia mampu penanganan kelas yang menarik dari bahasa dan meningkatkan kinerja pencarian informasi Mono-lingual secara signifikan dengan biaya komputasi rendah dan di relatif waktu pemrosesan rendah.

Penelitian terdahulu juga pernah dilakukan dengan topik ”*Stemming Techniques for Arabic Words: A Comparative Study*” oleh May Y. Al-Nashashibi, D. (2010). Sebuah surat posisional pendekatan peringkat untuk ekstraksi akar diselidiki. Dua varian bersama dengan penyesuaian untuk itu juga diusulkan dan dilaksanakan di sini. Hasil menerapkan teknik tersebut dibandingkan dengan orang-orang dari satu berbasis aturan. Ditemukan bahwa algoritma Koreksi memang meningkatkan kinerja dari dua pendekatan. Disesuaikan metode AI-Shulabi terbukti menjadi yang tertinggi di akurasi antara semua fve algoritma asli. Namun, Algoritma aturan-Bused menjadi pendekatan dengan tertinggi akurasi antara semua sepuluh algoritma ketika Koreksi yang algoritma termasuk di dalamnya (peningkatan sekitar 14%). Percobaan menunjukkan fture menjanjikan untuk diusulkan Algoritma *Correcton* dilaksanakan untuk lainnya stemmers. Namun, memiliki beberapa keterbatasan dan 14% perbaikan dapat ditingkatkan dengan menambahkan aturan frther dan pembatasan. Juga, metode AI-Shulubi Disesuaikan dapat frther ditingkatkan dengan: a) penanganan dua huruf geminated kata-kata, melakukan normalisasi untuk menangani kata-kata yang lemah, dan mengekstrak beberapa kata Quadriliteral-akar berbasis (seperti diusulkan dalam varian kedua), b) penanganan khusus efect dari b sebagai surat tambahan (ketika di awal kata) dan c) mengusulkan kisaran nilai berat badan bukan specifc yang. Hal ini agar karena jelas dari eksperimental hasil bahwa kedua kelompok yang diusulkan huruf dan mereka bobot masing tidak memberikan yang lebih tinggi

umumnya nilai akurasi. Namun, ia mengamati bahwa setiap Metode yang diusulkan memberi akar yang benar untuk beberapa kata tapi gagal untuk orang lain, sedangkan metode AI-Shulubi Disesuaikan disediakan akar yang benar untuk banyak orang lain. Hal ini menunjukkan bahwa menggunakan rentang nilai bobot surat tersebut mungkin menyediakan akurasi yang lebih tinggi (bukan nilai-nilai specifc). Juga, mengakuisisi koleksi teks lebih besar akan menekankan hasil kinerja teknik tersebut.

Penelitian terkait lainnya dilakukan sebelumnya dengan judul " *A Rule Based Arabic Stemming Algorithm* " oleh Tengku Mohd T. Sembok, dkk (2010). Percobaan kami menunjukkan bahwa kami berasal baru algoritma melakukan lebih baik daripada Al-Omari. Mungkinkah ditingkatkan lebih lanjut? Analisis kami menunjukkan bahwa sebagian besar kesalahan yang tersisa karena urutan yang tepat di mana aturan diterapkan, dan kami sedang mempertimbangkan cara-cara di mana ini pemesanan dapat diterapkan terbaik.

Proses *Stemming* digunakan untuk mengubah *term* yang masih melekat dalam term tersebut awalan, sisipan, dan akhiran. Selanjutnya *term* tersebut diproses untuk dihilangkan awalan, sisipan dan akhiran sehingga menjadi term kata dasar. Proses membuat *term* dasar ini mengacu kepada bahasa Indonesia yang benar. Proses stemming dilakukan dengan cara menghilangkan semua imbuhan (*affixes*) baik yang terdiri dari awalan *(prefixes*), sisipan *(infixes)*, akhiran *(suffixes)* dan *confixes* (kombinasi dari awalan dan akhiran) pada kata turunan. *Stemming* digunakan untuk mengganti bentuk dari suatu kata menjadi kata dasar dari kata tersebut yang sesuai dengan struktur morfologi bahasa Indonesia yang benar (tala, 2003).

Imbuhan *(affixes)* pada Bahasa Indonesia lebih kompleks jika dibandingkan dengan imbuhan pada Bahasa Inggris. Imbuhan pada Bahasa Indonesia terdiri dari awalan *(prefixes)*, sisipan *(infixes)*, akhiran *(suffixes)*, bentuk perulangan *(repeated forms)* dan kombinasi awalan akhiran *(confixes)*. Imbuhan-imbuhan yang melekat pada suatu kata harus dihilangkan untuk mengubah bentuk kata tersebut menjadi bentuk kata dasarnya. *Steming* teks berbahasa indonesi memiliki beberapa masalah yang sangat khusus terhadap bahasa. Salah satu masalah

tersebut adalah perbedaan tipe dari imbuhan-imbuhan *(affixes)*, bahwa awalan *(prefixes)* dapat berubah tergantung dari huruf pertama pada kata dasar. Sebagai contoh "me-" dapat berubah "mem-" ketika huruf pertama dari kata dasar tersebut adalah "b", misalkan "membuat"(to make), tetapi "me-" juga dapat berubah menjadi "meny-" ketika huruf pertama dari kata dasar melekat adalah "s", misalkan "menyapu" *(to sweep)*.

3.2. *Stemmer*

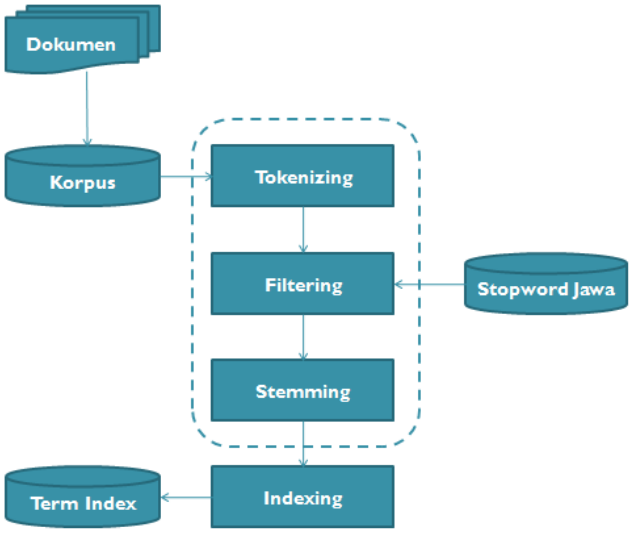
Stemming adalah proses untuk menemukan akar kata *(root)* atau kata dasar dengan memisahkan semua affix atau imbuhan yang melekat pada kata tersebut (indrajaya, 2003). *Affix* (imbuhan) bisa terdiri dari awalan *(prefix)*, akhiran *(suffix)*, sisipan *(infix)*, dan gabungan awalan-akhiran *(confix)*. Pada banyak bahasa, kata-kata biasanya dihasilkan dengan menambahkan imbuhan pada kata dasarnya *(root)*. Hasil dari stemming adalah stem (akar kata) yang merupakan bagian kata yang tersisa setelah dihilangkan imbuhannya.

Metode *Affix Removal* digunakan karena sifatnya yang fleksibel untuk digunakan sebagai *stemmer* berbagai macam bahasa dengan karakteristiknya yang lebih menekankan pada struktur morfologi suatu bahasa. Metode ini akan membuang Awalan *(prefix)*, *suffix* (akhiran) dan *infix* (sisipan) dari *term* menjadi suatu stem.

Stemmer diharapkan akan mampu mengurangi dimensi data sehingga nantinya akan meningkatkan performansi dari proses kategorisasi. Semakin sedikit dimensi data, maka akan semakin sedikit pula rule-rule untuk menentukan suatu dokumen terkategorisasi ke suatu kelas atau kategori tertentu sehingga bisa meningkatkan hasil kategorisasi. Pengurangan dimensi data dengan menggunakan stemming terjadi karena kata-kata yang memiliki kata dasar yang sama dikelompokkan menjadi satu atribut sehingga dimensi data bisa direduksi.

3.3. Arsitektur *Stemmer* Bahasa *Jawa Ngoko*

Arsitektur Informasi pada *Stemmer* Bahasa Jawa *Ngoko* menggunakan model Sistem temu kembali Informasi yang hanya diambil pada proses *Pre Processing*. Penekanan pada kajian stemmer sehingga yang dilakukan sistem hanya dimulai pada saat proses input dokumen kedalam korpus, proses tokenisasi, proses filtering dan Stemming. Gambar 3.1 menunjukkan arsitektur informasi *stemmer* bahasa *Jawa ngoko*.



Gambar 3.1 *Arsitektur Informasi Stemmer* Bahasa *Jawa Ngoko*

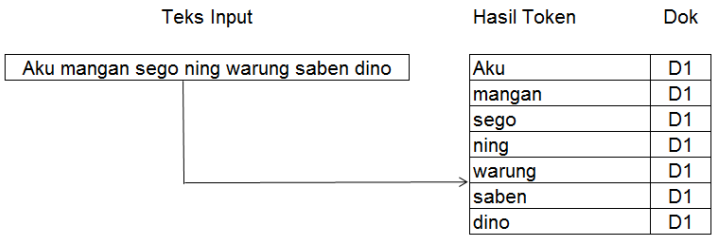
3.3.1. *Corpus (korpus)*

Corpus digunakan sebagai media untuk menempatkan data di database untuk dilakukan proses berikutnya dari sistem *stemmer* bahasa *jawa ngoko*. Penelitian dengan menggunakan database pada aplikasinya biasanya memakai korpus untuk proses pembuatan tabel pendukungnya. Penelitian empiris dapat dilakukan dengan menggunakan teks tertulis atau lisan, seperti teks-teks dasar dari berbagai jenis sastra dan analisis linguistik. Tapi gagasan tentang korpus sebagai dasar untuk sebuah bentuk linguistic empiris berbeda dalam beberapa cara mendasar dari teks-teks tertentu. Pada prinsipnya, setiap koleksi lebih dari satu teks dapat disebut dengan *korpus* (McEnery dan Wilson, 2001): istilah *korpus* dalam bahasa latin berarti *body*, maka *korpus* dapat didefinisikan sebagai isi setiap teks.

3.3.2. *Tokenizing*

Sistem temu kembali Informasi dimulai dengan proses memisahkan kata yang ada pada dokumen berdasarkan spasi kemudian memproses kata yang telah dipisahkan tersebutk kedalam sebuah tabel untuk dilakukan proses berikutnya. *Tokenizing* merupakan proses pemisahan suatu rangkaian karakter berdasarkan karakter spasi, dan mungkin pada waktu yang bersamaan dilakukan juga proses penghapusan karakter tertentu, seperti tanda baca. Sebagai contoh, kata-kata "*computer*", "*computing*", dan "*compute*" semua berasal dari term yang sama yaitu "*comput*" , tanpa pengetahuan sebelumnya dari morfologi bahasa Inggris. Token seringkali disebut sebagai istilah (*term*) atau kata, sebagai contoh sebuah token merupakan suatu urutan karakter dari dokumen tertentu yang dikelompokkan sebagai unit semantik yang berguna untuk diproses (Salton, 1989). contoh tokenisasi bisa dilihat pada gambar 3.2. Input : Aku mangan sega ning warung saben dino (amin, 2015)

Output :



Gambar 3.2 Contoh hasil proses tokenisasi

3.3.3. *Filtering*

Pada riset *stemmer jawa ngoko*, proses filtering tidak dilakukan karena yang menjadi konsentrasi adalah proses *stemming*. *Filtering* akan memproses kata hasil proses tokenisasi menjadi lebih sedikit dengan cara mengurangi kata tersebut dengan kata yang termasuk dalam *stopwords*. Eliminasi *stopwords* memiliki banyak keuntungan, yaitu akan mengurangi *space* pada tabel *term index* hingga 40% atau lebih (Baeza, 1999,h.167). Proses *stopword removal* merupakan proses penghapusan *term* yang tidak memiliki arti atau tidak relevan. Proses ini dilakukan pada saat proses tokenisasi. Proses tokenisasi menghasilkan sebuah

term, dan term tersebut selanjutnya di periksa dalam daftar *stopword*. Apabila *term* tersebut terdapat dalam daftar *stopword* maka *term* tersebut tidak akan dimasukkan dalam tabel *term*. Sebaliknya *term* hasil tokenisasi apabila diperiksa ke dalam daftar *stopword* dan hasilnya nihil maka *term* tersebut akan dimasukkan ke dalam tabel *term*(Baeza, 1999,h.167).

beberapa cara dalam proses *stopword removal*, antara lain meletakkan proses stopword sebelum term hasil tokenisasi dimasukkan ke dalam tabel *term*, cara yang kedua menempatkan proses *stopword* setelah term hasil tokenisasi masuk kedalam tabel (amin, 2015). Gambar 3.3 menunjukkan prosec filtering.

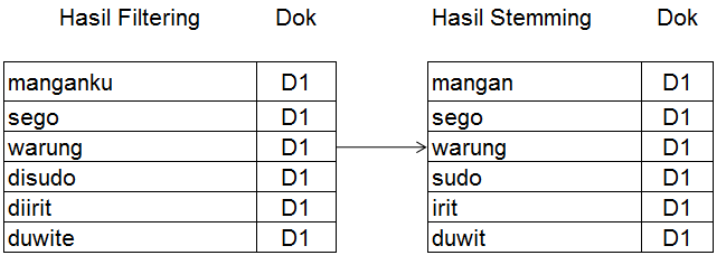
Gambar 3.3 Hasil proses Filtering

3.3.4. *Stemming*

Stemmer bahasa *Jawa ngoko* dibuat menggunakan pendekatan morfologi bahasa jawa ngoko. Proses *Stemming* digunakan untuk mengubah *term* yang masih melekat dalam term tersebut awalan, sisipan, dan akhiran. Selanjutnya *term* tersebut diproses untuk dihilangkan awalan, sisipan dan akhiran sehingga menjadi term kata dasar. Proses membuat *term* dasar ini mengacu kepada bahasa jawa ngoko yang benar (amin, 2015).

Proses stemming dilakukan dengan cara menghilangkan semua imbuhan (*affixes*) baik yang terdiri dari awalan *(prefixes*), sisipan *(infixes)*, akhiran *(suffixes)* dan *confixes* (kombinasi dari awalan dan akhiran) pada kata turunan. *Stemming* digunakan untuk mengganti bentuk dari suatu kata menjadi kata dasar dari kata tersebut yang sesuai dengan struktur morfologi bahasa jawa yang benar.

Imbuhan *(affixes)* pada Bahasa *Jawa ngoko* lebih kompleks jika dibandingkan dengan imbuhan pada Bahasa Inggris. Imbuhan pada Bahasa Jawa ngoko terdiri dari awalan *(prefixes)*, sisipan *(infixes)*, akhiran *(suffixes)*, bentuk perulangan *(repeated forms)* dan kombinasi awalan akhiran *(confixes)*. Imbuhan-imbuhan yang melekat pada suatu kata harus dihilangkan untuk mengubah bentuk kata tersebut menjadi bentuk kata dasarnya. *Steming* teks berbahasa Jawa ngoko memiliki beberapa masalah yang sangat khusus terhadap bahasa. Salah satu masalah tersebut adalah perbedaan tipe dari imbuhan-imbuhan *(affixes)*, bahwa awalan *(prefixes)* dapat berubah tergantung dari huruf pertama pada kata dasar. Sebagai contoh "ng-" dapat berubah "k-" ketika huruf pertama dari kata dasar tersebut adalah "ng", misalkan "ngetok"(kata dasar kethok), tetapi dapat berubah menjadi "ng-" ketika huruf pertama dari kata dasar melekat adalah "k", misalkan "ngethok" *(*kata dasar kethok*)*. Contoh proses *Stemming* bisa dilihat pada gambar 3.4. (amin, 2015)



Hasil Filtering	Dok
manganku	D1
sego	D1
warung	D1
disudo	D1
diirit	D1
duwite	D1

Hasil Stemming	Dok
mangan	D1
sego	D1
warung	D1
sudo	D1
irit	D1
duwit	D1

Gambar 3.4 Hasil proses Stemming

Penelitian terhadap *stemming* untuk *retrieval*, *machine translation*, *document summarization* dan *text classification* sudah pernah dilakukan sebelumnya. Stemming yang dilakukan pada *text retrieval*, stemming ini meningkatkan kesensitifan *retrieval* dengan meningkatkan kemampuan untuk menemukan dokumen yang relevan, tetapi hal itu terkait dengan pengurangan pada pemilihan dimana pengelompokan menjadi kata dasar menyebabkan penghilangan makna kata (amin, 2015).

3.3.4.1. *Stemmer* Bahasa *Jawa ngoko*

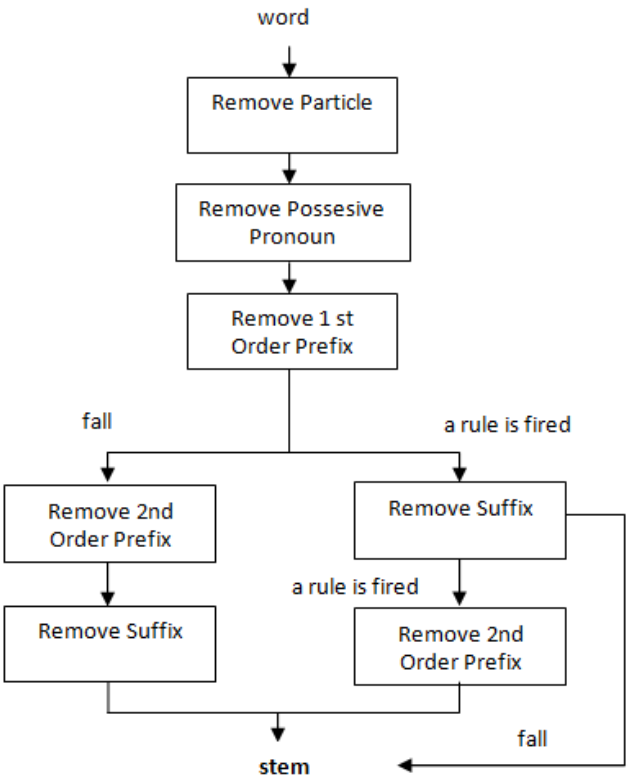
Pada penelitian *Stemmer* Bahasa *Jawa Ngoko* di adopsi dari algoritma *porter stemmer*. Metode *Stemmer* menggunakan *rule base* analisis untuk mencari *root* sebuah kata. *Stemmer* ini sama sekali tidak menggunakan kamus sebagai acuan. Struktur pembentukan kata dalam bahasa Jawa Ngoko adalah sebagai berikut:

[awalan]+[sisipan]+[dasar]+[akhiran]

Masing-masing bagian digabungkan dengan kata dasar membentuk kata berimbuhan. Algoritma Bahasa *Jawa Ngoko* menggunakan algoritma *rule based stemming* seperti halnya dengan algoritma *porter* pada *stemming* bahasa inggris.
Pada *stemmer* Bahasa Jawa ngoko Terdapat 5 langkah utama dengan 2 langkah awal dan 2 langkah pilihan, langkah-langkah tersebut adalah sebagai berikut:

a. Menghilangkan awalan (awalan-4, awalan-3, awalan-2 dan awalan-1)
b. Jika suatu aturan terpenuhi jalankan sbb :
   - Hilangkan akhiran (akhiran-3, akhiran-2, akhiran-1)
   - Jika suatu aturan terpenuhi, hilangkan awalan , jika tidak proses *stemming* selesai.
c. Jika tidak ada aturan yang terpenuhi jalankan sbb :
   a. Hilangkan awalan
   b. Hilangkan akhiran
   c. Proses *stemming* selesai.

Pada penelitian stemmer bahasa Jawa ngoko digunakan arsitektur *stemmer* bahasa Indonesia Tala. Hal ini dilakukan karena Tala melakukan penelitian tentang *stemmer* bahasa Indonesia, dan bahasa jawa adalah ibu dari bahasa Indonesia. Gambar 3.5 menunjukkan *stemmer* bahasa Indonesia.



Gambar 3.5. *The basic design of a Porter stemmer for* Bahasa Indonesia (Tala, 2003)

3.3.4.1.1. Proses menghilangkan Awalan *(ater-ater)*

Proses menghilangkan awalan dilakukan untuk mencari kata dasar dengan cara memisahkan awalan dengan kata dasarnya. Dalam bahasa Jawa, jumlah dan jenis prefiks *(ater-ater)* adalah sebagai berikut.

Jenis prefiks selanjutnya adalah sebagai berikut.

a. Awalan yang terdiri dari 4 huruf (4 digit): Kuma, Kapi

Kuma	+	Wani	=	Kumawani
Kapi	+	Lare	=	Kapilare
Kapi	+	Andreng	=	Kapiandreng

b. Awalan yang terdiri dari 3 huruf (3 digit) yaitu: Dak, Kok, Pan, Pra, Tar, Tak, Tok.

Dak	+	Gawa	=	Dakgawa
Dak	+	Tulis	=	Daktulis
Kok	+	Jupuk	=	Kokjupuk
Kok	+	Pangan	=	Kokpangan
Pan	+	gayuh	=	Panggayuh

Pra + karsa = Prakarsa
Pra + lambang = Pralambang
Tar + Buka = Tarbuka
Tak + Antem = Takantem
Tok + Simpen = Toksimpen

c. Awalan yang terdiri dari 2 huruf (2 digit) yaitu:Di, Ka, Ke, Ma, Pa, Pi, Sa.
Di + Pendhem = Dipendhem
DI + Pala = Dipala
Ka + Boyong = Kaboyong
Ke + Thutuk = Kethutuk
Ke + Banting = Kebanting
Ma + Guru = Maguru
Ma + Gawe = Magawe
Pa + warta = Pawarta
Pa + mirsa = Pamirsa
Pi + wulang = Piwulang
Pi + takon = Pitakon
Sa + wiji = sawiji
Sa + tunggal = satunggal
Sa + omah = saomah

d. Awalan yang terdiri dari 1 huruf (1 digit) yaitu: A
A + Gawe = Agawe
A + Wujud = Awujud

3.3.4.1.2. Menghilangkan akhiran *(penambang)*

Proses menghilangkan akhiran dilakukan untuk mencari kata dasar dengan cara memisahkan akhiran dengan kata dasarnya. Wujud sufiks dalam bahasa Jawa beserta contoh pemakaiannya tampak dalam deret di bawah ini.

a. Akhiran yang terdiri dari 3 huruf (3 digit) yaitu: Ana, Ane.
Silih + ana = Silihana
Jupuk + ana = Jupukana

b. Akhiran yang terdiri dari 2 huruf (2 digit) yaitu: An, Na, Ne.
Tulis + an = tulisan
Suntik + an = Suntikan
Gambar + na = Gambarna
Tulis + na = Tulisna
Rayi + ne = Rayine
Sega + ne = Segane

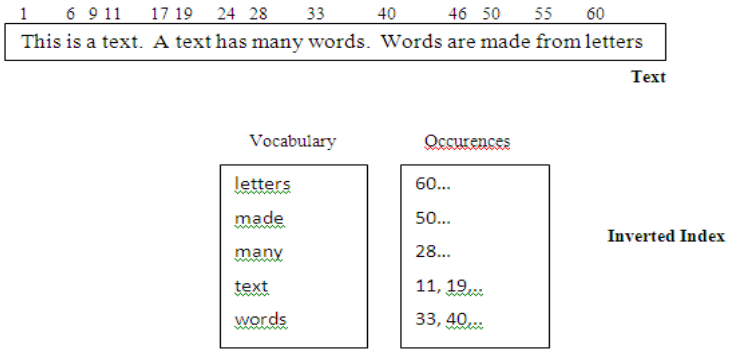
c. Akhiran yang terdiri dari 1 huruf (1 digit) yaitu: A, e, i.
Tuku + a = Tukua

Lunga	+	a	=	Lungaa
Mangan	+	a	=	mangana
Omahe	+	e	=	Omahe
Pitik	+	e	=	pitike
Golek	+	I	=	Goleki

3.3.5. *Inverted Index*

Pada prinsipnya proses menemukan *records* adalah menjawab dari permintaan *(request)* informasi didasarkan pada kemiripan diantara *query* dan kumpulan *term* pada sistem (Salton, 1989). *Inverted file* atau *inverted index* merupakan mekanisme untuk pengindeksan kata dari koleksi teks yang digunakan untuk mempercepat proses pencarian. Elemen penting dalam struktur *inverted file* ada dua, yaitu: kata *(vocabulary)* dan kemunculan *(occurrences)*. Kata-kata tersebut adalah himpunan dari kata-kata yang ada pada teks, atau merupakan ekstraksi dari kumpulan teks yang ada.

Cara kerja *index inverted*, *term-term* dikonversikan menjadi karakter huruf kecil *(lower-case)*. Kolom *vocabulary* adalah kata-kata yang telah diekstraksi dari koleksi teks, sedangkan *occurrences* adalah posisi kemunculan pada teks (gambar 3.6).



Gambar 3.6 *A sample text and an inverted index built on it (Baeza*, 1999,h.193)

Nilai kemunculan dari kata-kata memerlukan ruangan *(space)* yang tidak sedikit, karena tiap kata muncul pada teks sekali pada struktur *occurrences*, sehingga ada ruangan extra atau dilambangkan dengan *O(n)*. Tidak semua kata diindekskan karena ada kata-kata *stopword* yang dibuang, overhead yang muncul

akibat penambahan indeks ini mencapai 30% sampai dengan 40% dari ukuran besar koleksi teks (Baeza, 1999,h.193).

3.4. Uji *Hasil Stemmer* Bahasa *Jawa Ngoko*

Hasil *stemmer* bahasa *Jawa Ngoko* selanjutnya dilakukan uji hasil untuk menentukan tingkat kefektifan atau akurasi dari *stemmer* yang dibuat. Uji stemmer dilakukan dengan membandingkan hasil kerja program stemmer dengan kata dasar yang benar. Sebuah kata jadian (tembung andhahan) dilakukan pencarian kata dasar dengan menghilangkan awalan dan akhiran, selanjutnya hasilnya berupa kata dasar. Kata dasar hasil proses *stemmer* selanjutnya di cek kebenarannya menggunakan kata dasar bahasa jawa ngoko yang benar.

BAB IV
METODE PENELITIAN

4.1. METODE PENELITIAN

Metodologi yang digunakan pada penelitian ini adalah sebagai berikut:

4.1.1. Obyek Penelitian

Obyek penelitian dari penelitian ini adalah *Stemmer* Bahasa *Jawa Ngoko*.

4.1.2. Teknik Pengumpulan Data

Pengumpulan data dimaksudkan agar mendapatkan bahan-bahan yang relevan, akurat dan *reliable.* Maka teknik pengumpulan data yang dilakukan dalam penelitian ini adalah sebagai berikut :

a. Observasi

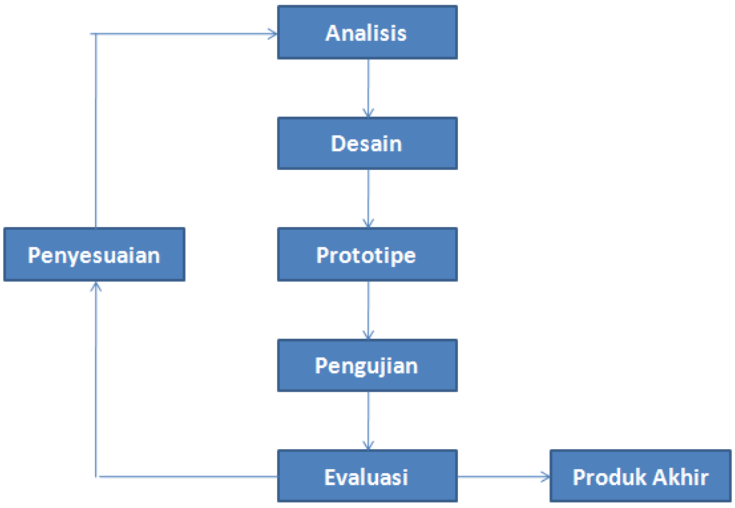
   Melakukan pengamatan dan pencatatan secara sistematis tentang hal-hal yang berhubungan dengan basis data dokumen teks Bahasa *Jawa Ngoko*.

b. Studi Pustaka

   Pengumpulan data dari bahan-bahan referensi, arsip, dan dokumen yang berhubungan dengan permasalan dalam penelitian ini.

4.1.3. Metode Pengembangan

Penelitian ini menggunakan model *prototype.* Di dalam model ini sistem dirancang dan dibangun secara bertahap dan untuk setiap tahap pengembangan dilakukan percobaan-percobaan untuk melihat apakah sistem sudah bekerja sesuai dengan yang diinginkan. Sistematika model *prototype* terdapat pada Gambar 4.1 memperlihatkan tahapan pada *prototype.*



Gambar 4.1. Tahapan *Prototype (Pressman, 2001)*

Berikut adalah tahapan yang dilakukan pada penelitian ini dengan metode pengembangan *prototype*

a. **Analisis**

   Pada tahap ini dilakukan analisa tentang masalah morfologi bahasa *jawa ngoko* dan menentukan pemecahan masalah yang tepat untuk menyelesaikannya.

b. **Desain**

   Pada tahap ini dibangun rancangan sistem dengan beberapa diagram bantu seperti Arsitektur Informasi dari program dan *Flowchart.* Pembuatan *Flowcart* dilakukan untuk membuat proses menjadi runtut dan membantu proses desain.

c. ***Prototype***

   Pada tahap ini dibangun aplikasi *Stemmer* Bahasa *Jawa Ngoko dengan Metode Rule Based Analysis* yang sesuai dengan desain dan kebutuhan sistem.

d. **Pengujian**

   Pada tahap ini dilakukan pengujian *software Stemmer* dengan uji Kamus Bahasa *Jawa Ngoko*.

e. **Evaluasi**

Pada tahap ini dilakukan evaluasi apakah performa aplikasi sudah sesuai dengan yang diharapkan, apabila belum maka dilakukan penyesuaian-penyesuaian secukupnya.

f. **Penyesuaian**

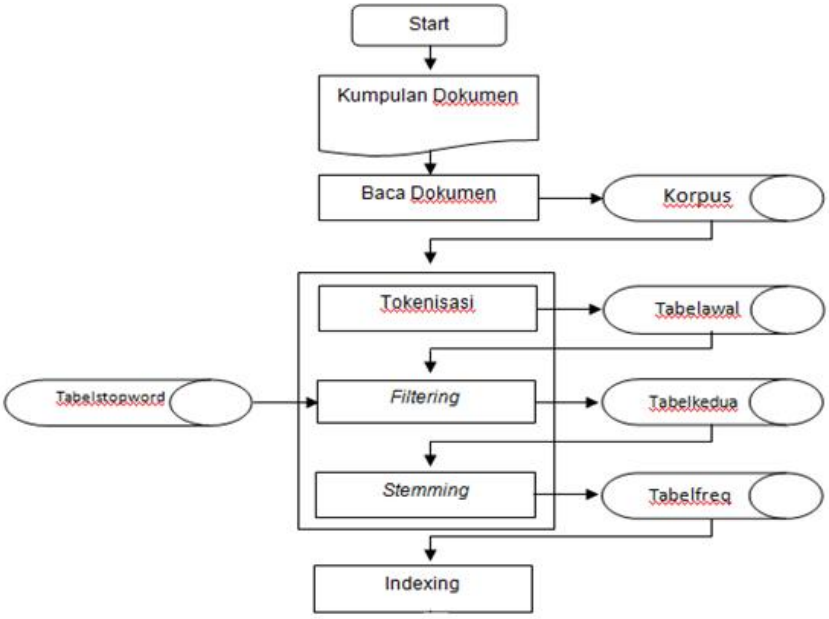
Tahap ini dilakukan apabila pada evaluasi performa aplikasi kurang memadai dan dibutuhkan perbaikan, tahap ini melakukan penyesuaian dan perbaikan pada aplikasi sesuia dengan kebutuhan

BAB V
HASIL DAN PEMBAHASAN

5.1. Rancang Bangun *Stemmer* Bahasa *Jawa Ngoko*

5.1.1. Flowchart Stemmer

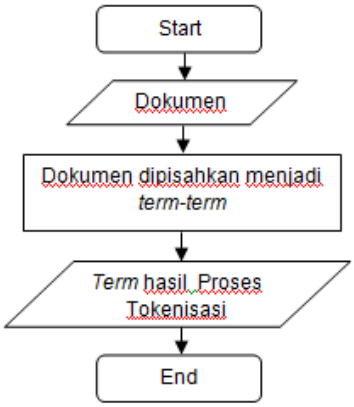
Stemmer dirancang bisa menghasilkan kata dasar bahasa jawa ngoko dengan benar. *Stemmer* Bahasa *Jawa Ngoko* dirancang melalui beberapa tahap, yaitu: Tokenisasi, *filtering* dan *Stemming*. Gambar 5.1 menunjukkan proses *Stemmer* Bahasa *Jawa Ngoko* dimulai dengan proses pengumpulan dokumen yang diinputk kedalam Korpus (tabel). Data berupa dokumen kemudian di proses tokenisasi yang menghasilkan *term-term* yang terpisah didasarkan pada spasi saat pemrosesannya. *Term* hasil proses tokenisasi selanjutnya dilakukan proses *filtering*. Term hasil proses *filtering* kemudian dilakukan *stemming* dan menghasilkan kata dasar Bahasa *Jawa Ngoko*.



Gambar 5.1. *flowchart Stemmer* Bahasa *Jawa Ngoko*

5.1.1.1.Flowchart Tokenisasi

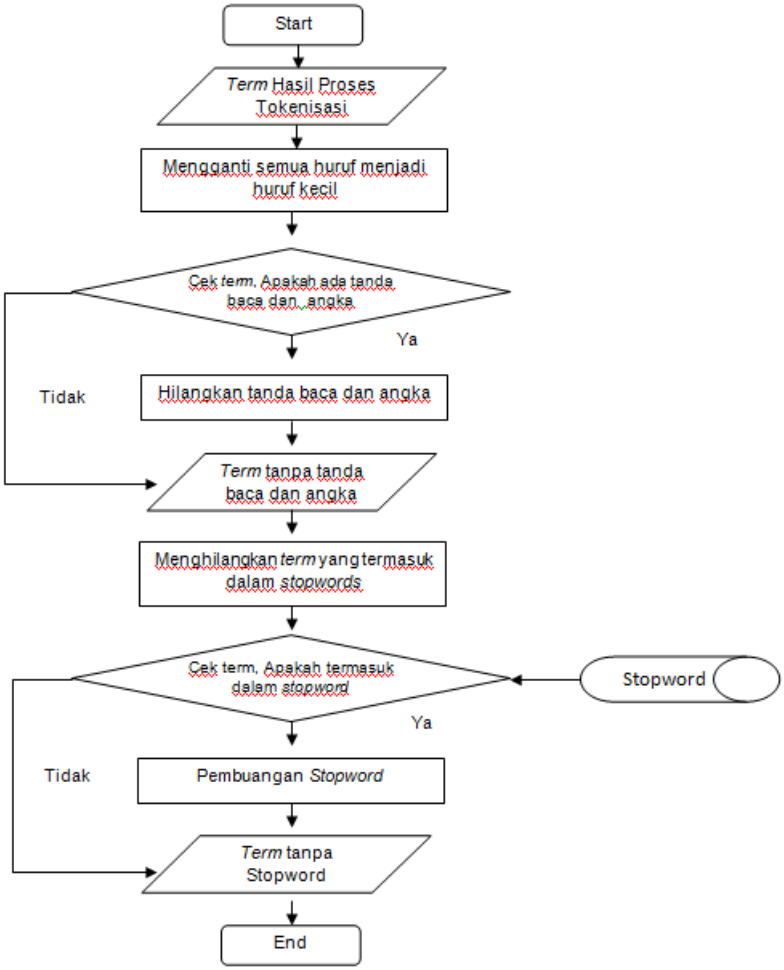
Proses Tokenisasi dirancang untuk dapat memisahkan dokumen menjadi *term-term* yang akan diproses pada tahap *filtering*. Proses tokenisasi diawali dengan *scanner* dokumen yang ada pada korpus kemudian diproses menjadi *term*. *Flowchart* tokenisasi bisa dilihat pada gambar 5.2.



Gambar 5.2 *Flowchart* Proses Tokenisasi

5.1.1.2. Flowchart Filtering

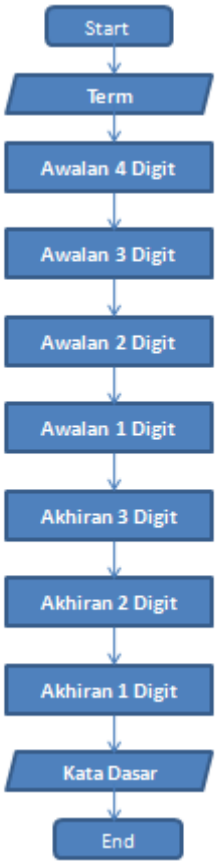
Proses *Filtering* dirancang untuk menghasilkan *term* tanpa *stopwords*. *Flowchart filtering* dimulai dengan mengganti huruf kapital menjadi huruf kecil, menghilangkan tanda baca dan angka, dan menghilangkan *term* yang termasuk dalam *stopwords*. Gambar 5.3. menunjukkan *flowchart* proses *filtering*.



Gambar 5.3. *Flowchart* Proses *Filtering*

5.1.1.3. Flowchart stemming

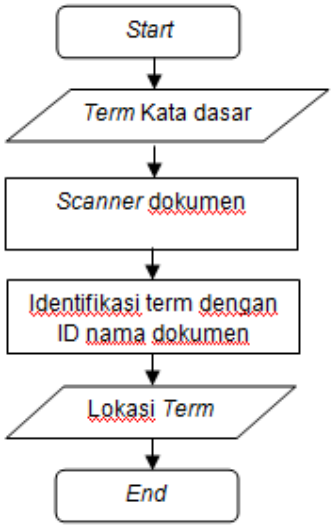
Proses *stemming* dirancang agar *term* hasil *filtering* diubah menjadi *term* kata dasar. Proses *stemming* dimulai dengan menghilangkan awalan dan akhiran. Proses ini juga dirancang dapat melakukan *replace* ketika awalan dihilangkan dan menggantinya dengan huruf yang sesuai. Proses menghilangkan awalan, akhiran, dan *replace* sisipan dilakukan dalam satu tahap proses. Gambar 5.4 menunjukkan *flowchart stemming*.



Gambar 5.4. *Flowchart* Proses *Stemming (stemmer Bahasa Jawa Ngoko)*

5.1.1.4. Flowchart Indexing

Term kata dasar hasil proses *stemming* selanjutnya dimasukkan dalam tabel untuk diproses pada perhitungan. Proses *indexing* menggunakan metode *inverted indexing*, yaitu dengan membedakan letak tiap term dalam dokumen. Gambar 5.5. menunjukkan *flowchart indexing*.



Gambar 5.5. *Flowchart* Proses *Indexing*

*5.1.1.5.*Rancangan Tabel

Pada *Stemmer* Bahasa *Jawa Ngoko* ini menggunakan beberapa tabel untuk tempat meletakkan kumpulan data pada korpus, *term-term* hasil proses Tokenisasi, *Filtering* dan *Stemming*. Berikut ini Rancangan tabel yang akan digunakan dalam software *Stemmer* Bahasa *Jawa Ngoko* pada penelitian ini;

5.1.1.5.1. Rancangan Tabel Korpus

Tabel korpus digunakan untuk meletakkan dokumen-dokumen dengan *field-field* id, judul, isi dan dokumen (tabel 5.1).

Tabel 5.1 Rancangan Tabel Korpus

Field	Type	Collation	Attributs	Null	Default	Extra
Id	int(11)	utf8_general_ci		No		auto_increment
judul	varchar(500)	utf8_general_ci		No		
isi	varchar(3000)	utf8_general_ci		No		
dokumen	varchar(100)	utf8_general_ci		No		

5.1.1.5.2. Rancangan Tabel Tabelawal

Tabelawal digunakan untuk meletakkan term hasil tokenisasi dengan *field-field* judul, *term* dan dokumen (tabel 5.2).

Tabel 5.2 Rancangan Tabel Tabelawal

Field	Type	Collation	Attributs	Null	Default	Extra
judul	varchar(250)	utf8_general_ci		No		
term	varchar(500)	utf8_general_ci		No		
dokumen	varchar(100)	utf8_general_ci		No		

5.1.1.5.3. Rancangan Tabel TabelKedua

Tabelkedua digunakan untuk meletakkan term hasil proses *filtering* dengan *field-field* judul, term dan dokumen (tabel 5.3).

Tabel 5.3 Rancangan Tabel Tabelkedua

Field	Type	Collation	Attributs	Null	Default	Extra
judul	varchar(250)	utf8_general_ci		No		
term	varchar(250)	utf8_general_ci		No		
dokumen	varchar(100)	utf8_general_ci		No		

5.1.1.5.4. Rancangan Tabel Tabelfreq

Tabelfreq digunakan untuk meletakkan term hasil proses *stemming* dengan *field-field* judul, *term, freq* dan freqpangkat (tabel 5.4).

Tabel 5.4 Rancangan Tabel Tabelfreq

Field	Type	Collation	Attributs	Null	Default	Extra
judul	varchar(250)	utf8_general_ci		No		
Term	varchar(250)	utf8_general_ci		No		
Freq	int(11)	utf8_general_ci		No		
frekpangkat	int(11)	utf8_general_ci		No		

5.1.1.6. Rancangan *Interface*

5.1.1.6.1. Rancangan *Interface Stemmer* Bahasa *Jawa Ngoko*

Rancangan interface *Stemmer* Bahasa *Jawa Ngoko* ditampilkan dalam bentuk dan susunan field-field sebagai berikut: Judul, Term, Frekuensi, Awalan, Kata Dasar dan Akhiran. Gambar 5.6 menunjukkan rancangan *Stemmer* Bahasa *Jawa Ngoko* .

Gambar 5.6. Rancangan *Interface Stemmer* Bahasa *Jawa Ngoko*

5.1.1.6.2. Rancangan Hasil *Stemmer*

Rancangan menu Hasil Stemmer Bahasa Jawa Ngoko akan ditampilkan per kata. Setiap kata jadian (tembung andhahan) akan dilakukan proses pemisahan kata dengan awalan dan akhiran. Aplikasi hasil stemmer bisa dilihat pada gambar 5.7

Gambar 5.7. Rancangan *Interface* Hasil *Stemmer* Bahasa *Jawa Ngoko*

5.2. Aplikasi *Stemmer* Bahasa *Jawa Ngoko*

5.2.1. Memasukkan Kata Jadian *(Tembung Andhahan)* kedalam tabel Korpus

Kata Jadian *(tembung andhahan)* di *input* secara manual dengan format dokumen teks kedalam tabel korpus. Proses ini dilakukan dengan cara memasukkan *tembung andhahan* bahan kajian penelitian kedalam tabel korpus. Sebelum dimasukkan kedalam tabel, dibuat satu tabel dengan nama tabel korpus yang digunakan sebagai tempat data. Tabel korpus ini memiliki *fiel-field* id, judul, isi dan dokumen. *Field* id berisi urutan data penelitian didalam korpus yang tersusun sesuai dengan urutan input data. Proses memasukkan dokumen ke dalam tabel korpus ini memerlukan waktu relative lama bergantung pada jumlah data yang akan di *input* kedalam tabel korpus. Bentuk tabel korpus seperti terlihat pada tabel 5.5.

Tabel 5.5. Tabel Korpus

id	judul	isi	dokumen
1	Ater-ater Hanuswara	Mbathik mupus ndudut nulis nggawa ngethok nyuwil nyikut	A1
2	Ater-ater Tripurasa	Dakpangan dakdhudhuk kojupuk kogoreng diambung dibatin	A2
3	Ater-ater Liyane	Alungguh malumpat kalimpe kesandhung sagegem palilah pitutur prawira kumawani kamituwo kapilare tarwaca	A3
4	Seselan	Gumuyu kemayu sinerat ginawa delewer kelumpruk ceruwil kerelip	S1
5	Penambang	Turuti jupukake tekane bapake jalukane kethoke turua wenehana gawakna bukuku omahmu lepehen	P1

5.2.2. Proses Tokenisasi

Proses tokenisasi dilakukan untuk mendapatkan term berdasarkan spasi. Proses *scanner* dokumen korpus menggunakan format teks dilakukan dengan cara masuk kedalam dokumen korpus melalui perantara program php ke dalam database mysql. Proses *scanner* data dilakukan dengan cara *scanner* baris per baris, untuk tiap-tiap file naskah yang ada di dokumen. Tokenisasi dimulai dengan memisahkan *term-term* yang ada pada dokumen korpus menjadi kumpulan term melalui proses *scanner* dengan dasar spasi. Selanjutnya term hasil proses tokenisasi di masukkan kedalam tabelawal dengan menyertakan *field-field* judul, *term* dan dokumen. Tabel 5.6 menunjukkan hasi tokenisasi.

Tabel 5.6. Aplikasi Proses Tokenisasi pada Tabel Awal

judul	term	dokumen
Ater-ater Hanuswara	mbathik	A1
Ater-ater Hanuswara	mupus	A1
Ater-ater Hanuswara	ndudut	A1
Ater-ater Hanuswara	nulis	A1
Ater-ater Hanuswara	nggawa	A1
Ater-ater Hanuswara	ngethok	A1
Ater-ater Hanuswara	nyuwil	A1
Ater-ater Hanuswara	nyikut	A1
Ater-ater Tripurasa	dakpangan	A2
Ater-ater Tripurasa	dakdhudhuk	A2
Ater-ater Tripurasa	kojupuk	A2
Ater-ater Tripurasa	kogoreng	A2
Ater-ater Tripurasa	diambung	A2
Ater-ater Tripurasa	dibatin	A2

5.2.3. Proses *Filtering*

Proses *Filtering* dibuat tapi tidak digunakan dalam penelitian ini, hal ini karena yang menjadi fokus adalah *stemming*. Proses selanjutnya setelah proses tokenisasi adalah proses *filtering*. Proses filtering adalah proses baca tabel kedua untuk diperiksa apakah semua term memiliki term-term yang termasuk dalam stopword list jawa. Jika dalam tabel kedua terdapat *term-term* yang termasuk dalam *stopword*, maka akan dilakukan penghilangan *term-term* tersebut. Hasil proses filtering selanjutnya dimasukkan dalam tabel freq (tabel 5.7).

Tabel 5.7 Hasil Proses Filtering pada Tabel Frekuensi

judul	term	dokumen
Ater-ater Hanuswara	mbathik	A1
Ater-ater Hanuswara	mupus	A1
Ater-ater Hanuswara	ndudut	A1
Ater-ater Hanuswara	nulis	A1
Ater-ater Hanuswara	nggawa	A1
Ater-ater Hanuswara	ngethok	A1
Ater-ater Hanuswara	nyuwil	A1
Ater-ater Hanuswara	nyikut	A1
Ater-ater Tripurasa	dakpangan	A2
Ater-ater Tripurasa	dakdhudhuk	A2
Ater-ater Tripurasa	kojupuk	A2
Ater-ater Tripurasa	kogoreng	A2
Ater-ater Tripurasa	diambung	A2
Ater-ater Tripurasa	dibatin	A2

5.2.4. Proses *Stemming*

Proses *stemming* yang digunakan adalah proses *stemmer* menggunakan *stemmer* untuk bahasa Jawa ngoko berdasarkan stemmer bahasa Indonesia yang dibuat Tala. Proses stemming dengan menggunakan *stemmer* jawa ngoko melalui beberapa tahapan seperti terlihat pada gambar 5.4 dan untuk mendukung proses ini juga digunakan *stopword list jawa ngoko*. Hasil akhir dari proses stemming adalah kumpulan *term* yang sudah menjadi kata dasar yang diinput dalam tabel *freq*.

Proses *stemming* menghasilkan kumpulan term berupa kata dasar hasil scanner *term* pada tabel kedua. Proses *stemming* didukung stopword jawa ngoko yang digunakan untuk mengurangi term yang ada pada tabel kedua. Selanjutnya *term* hasil *stemming* di letakkan pada tabel freq (tabel 5.8).

Tabel 5.8 Tabel Frekuensi

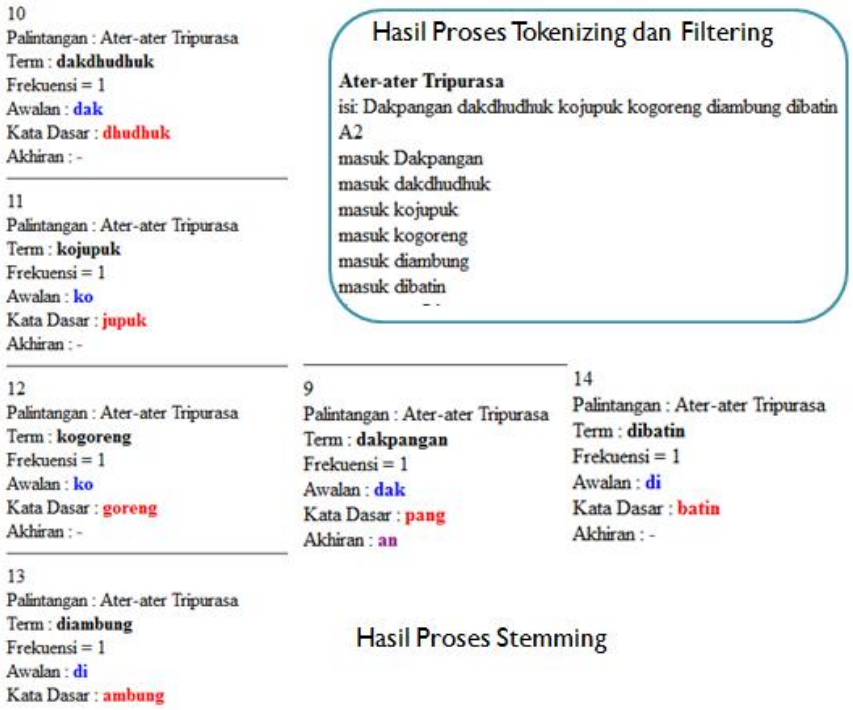
Penambang	turut
Penambang	jupuk
Penambang	tek
Penambang	bap
Penambang	jaluk
Penambang	ketho
Penambang	turua
Penambang	weneh
Penambang	gawak
Penambang	buku
Penambang	omah
Penambang	lepeh

5.2.5. Proses *Indexing*

Proses *indexing* dilakukan untuk mengambil atau meretrieve *term-term* yang ada pada tabelfreq untuk selanjutnya diproses pada saat proses pemisahan kata jadian menjadi bentuk kata dasar.

5.3. Prosedur

Stemmer Bahasa *Jawa Ngoko* ini dirancang agar didapatkan bentuk kata dasar bahasa jawa ngoko yang benar. Tampilan *interface* dirancang dalam bentuk terpisah mulai dari term kata jadian, awalan, akhiran dan kata dasar hasil proses *stemming*. (gambar 5.8). Prosedur menggunakan *Stemmer* bahasa *Jawa Ngoko* ini sangat mudah, yaitu dengan menampilkan softwarenya.



Gambar 5.8. *Interface Stemmer* Bahasa *Jawa ngoko*

5.4. Studi Kasus *Stemmer* Bahasa Jawa Ngoko

Studi kasus pada aplikasi *Stemmer* bahasa *Jawa Ngoko* ini menggunakan kumpulan tembung andhahan (kata jadian). Sebagai contoh digunakan tembung andhahan yang telah diberi ater-ater tripurasa. Term yang akan diproses adalah: *"dakpangan dakdhudhuk kojupuk kogoreng diambung dibatin"* Gambar 5.8 menunjukkan implementasi proses tersebut. Proses stemmer menghasilkan program bisa memisahkan awalan dengan kata dengan benar, meskipun ada satu kata yang terambil tidak benar yaitu kata *"dakpangan"*. Beberapa kata yang terabil dan diproses dengan benar diantaranya:

"dakdhudhuk" = dak + duduk
"kojupuk" = ko + jupuk
"kogoreng" = ko + goreng
"diambung" = di + ambung
" dibatin" = di + batin

5.5. Pengujian Hasil *Stemmer* Bahasa *Jawa Ngoko*

Pengujian program dilakukan dengan cara membandingkan hasil proses stemming dengan kamus bahasa jawa ngoko. Pelaksanaan pengujian hasil dilakukan pada setiap awalan *(ater-ater)*, sisipan *(seselan)* dan akhiran *(penambang).*

5.5.1. Uji Hasil *Ater-ater Hanuswara*

Ater-ater Hanuswara adalah awalan dalam bahasa Jawa ngoko. Adapun yang termasuk dalam ater-ater hanuswara antara lain; m, n, ng, ny (tabel 5.9)

Tabel 5.9 Tabel Ater-ater Hanuswara

m	bathik	mbathik
m	pupus	mupus
n	dudut	ndudut
n	tulis	nulis
ng	gawa	nggawa
ng	kethok	ngethok
ny	cuwil	nyuwil
ny	sikut	nyikut

Hasil dari proses pengujian hasil stemmer Bahasa Jawa ngoko adalah ater-ater Hanuswara: m-, n-, ng-, ny-, belum bisa distemmer dengan hasil yang benar.

5.5.2. Uji Hasil *Ater-ater Tripurasa*

Ater-ater Tripurasa adalah awalan dalam bahasa Jawa ngoko. Adapun yang termasuk dalam ater-ater Tripurasa antara lain; dak, ko, di (tabel 5.10)

Tabel 5.10 Tabel Ater-ater Tripurasa

dak	pangan	dakpangan
dak	dhudhuk	dakdhudhuk
ko	jupuk	kojupuk
ko	goreng	kogoreng
di	ambung	diambung
di	batin	dibatin

Hasil dari proses pengujian hasil *stemmer* Bahasa *Jawa ngoko* adalah *ater-ater Tripurasa*: Dak-, belum bisa distemmer dengan hasil yang benar. Sedangkan ater-ater Ko- dan di- berhasil mendapatkan term kata dasar benar.

5.5.3. Uji Hasil *Ater-ater Liyane*

Ater-ater Liyane adalah awalan dalam bahasa Jawa ngoko. Adapun yang termasuk dalam *ater-ater Liyane* antara lain; a, ma, ka, ke, sa, pa, pi, pra, kuma, kami, kapi, tar (tabel 5.11)

Tabel 5.11 Tabel Ater-ater Liyane

a	lungguh	alungguh
ma	lumpat	malumpat
ka	limpe	kalimpe
ke	sandhung	kesandhung
sa	gegem	sagegem
pa	lilah	palilah
pi	tutur	pitutur
pra	wira	prawira
kuma	wani	kumawani
kami	tuwo	kamituwo
kapi	lare	kapilare
tar	waca	tarwaca

Hasil dari proses pengujian hasil *stemmer* Bahasa *Jawa ngoko* adalah ater-ater liyane: a-, belum bisa distemmer dengan hasil yang benar. Sedangkan ater-ater ma, ka, ke, sa, pa, pi, pra, kuma, kami, kapi, dan tar berhasil mendapatkan term kata dasar benar.

5.5.4. Uji Hasil *Seselan*

Seselan adalah sisipan dalam bahasa Jawa ngoko. Adapun yang termasuk dalam *Seselan* antara lain; -um, -in, -el lan -er (tabel 5.12)

Tabel 5.12 Tabel Seselan

-um	guyu	gumuyu
-um	ayu	kemayu
-in	serat	sinerat

-in	gawat	ginawa
-el	dewer	delewer
-el	kumpruk	kelumpruk
-er	cuwil	ceruwil
-er	kelip	kerelip

Hasil dari proses pengujian hasil *stemmer* Bahasa *Jawa ngoko* adalah seselan: -um, -in, -el lan -er belum bisa distemmer dengan hasil yang benar.

5.5.5. Uji Hasil *Penambang*

Penambang adalah akhiran dalam bahasa Jawa ngoko. Adapun yang termasuk dalam *penambang* antara lain; -I, -ake, -e, -ane, -ke, -a, -ana, -na, -ku, -mu, ,-en (tabel 5.13)

Tabel 5.13 Tabel Penambang

turut	-i	turuti
jupuk	-ake	jupukake
teka	-ne	tekane
bapak	-e	bapake
jaluk	-ane	jalukane
kethok	-ke	kethoke
turu	-a	turua
weneh	-ana	wenehana
gawa	-na	gawakna
buku	-ku	bukuku
omah	-mu	omahmu
lepeh	-en	lepehen

Hasil dari proses pengujian hasil *stemmer* Bahasa *Jawa ngoko* adalah penambang: -e, -ke, dan -a belum bisa distemmer dengan hasil yang benar. Sedangkan penambang -I, -ake, -ane, -, -ana, -na, -ku, -mu, ,-en berhasil mendapatkan term kata dasar benar.

BAB VI
SIMPULAN DAN SARAN

6.1. Kesimpulan

a. *Stemmer* Bahasa *Jawa ngoko* mampu membuat kata dasar *jawa ngoko* dengan benar 62 % atau 21 dari 34 (*ater-ater*/awalan, *seselan*/sisipan dan *penambang*/akhiran)
b. *Stemmer* Bahasa *Jawa ngoko* mampu melakukan proses *stemming* dengan hasil yang mudah di pahami karena dibuat terpisah antara awalan, kata dasar dan akhiran.

6.2. Saran

a. Perlunya Perbaikan kode untuk: *ater-ater Hanuswara* (m-, n-, ng-, ny-), ater-*ater tripurasa* (dak-), *ater-ater liyane* (a-), *Seselan* (-um, -in, -el, -er) dan *Penambang* (-e, -ke, -a) sehingga akan menghasilkan *stemmer* Bahasa *Jawa Ngoko* dengan 100% benar

b. Penggunaan kamus akan membantu proses *Stemming*

DAFTAR PUSTAKA

Budi, I., Aji, R.F., 2006. Efektifitas Seleksi Fitur dalam Sistem Temu Kembali Informasi. Seminar Nasional Aplikasi Teknologi Informasi (SNATI), ISSN : 1907-5022.

Bum, K.Y., 2010. *An autonomous assessment system based on combined latent semantic kernels. Expert Systems with Applications: An International Journal , Volume 37 Issue 4.*

Deepika Sharma (2012) Stemming Algorithms: A Comparative Study and their Analysis, International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868 Foundation of Computer Science FCS, Volume 4– No.3, September 2012, New York, USA

Kadir, A., 2001. Dasar Pemrograman Web Dinamis menggunakan PHP. Penerbit Andi. Yogyakarta.

May Y. Al-Nashashibi, D.(2010), Stemming Techniques for Arabic Words: A Comparative Study, 2010 2nd International Conerence on Computer Technolog and Development (CCTD 2010)

Manning, C., Raghavan, P., 2007. An Introduction to Information Retrieval, Stanford. USA.

Meadow, C.T., 1997. *Text Information Retrieval Systems*. Academic Press.New York.

Tala, F.Z., 2003, *A Study of Stemming Effects on Information Retrieval in bahasa Indonesia*. Institut for logic, Language and Computation Universiteit van Amsterdam The Netherlands.

Porter, M. F. (1980). An algorithm for suffix stripping'. Program, 14, 1304137.

Salton, G., 1989, *Automatic Text Processing, The Transformation, Analysis, and Retrieval of information by computer*. Addison – Wesly Publishing Company, Inc. USA.

Sandeep R. Sirsat (2013), Strength and Accuracy Analysis of Affix Removal Stemming Algorithms, Internetional Journal of Computer Science and Information Technology, Vol 4 (2), IJCSIT

Yates, R.B, 1999. *Modern Information Retrieval*, *Addison Wesley*-Pearson international edition, Boston. USA.