PEMBANGUNAN KAMUS BAHASA INDONESIA KATA TIDAK BAKU MENGGUNAKAN ALGORITMA LATENT SEMANTIC INDEXING DAN DAMERAU-LEVENSHTEIN DISTANCE

Jenal Abidin
Universitas Komputer Indonesia
Program Pascasarjana, Program Studi Magister Manajemen
Jl Dipati Ukur No 112-116, Bandung 40132
Jenalabidin2009@gmail.com

***Abstrak*—Analisa sentimen atau opinion mining merupakan proses memahami, mengekstrak dan mengolah data tekstual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam suatu kalimat opini. Permasalahan dalam analisa sentimen pada media sosial di Indonesia adalah ditemukan banyak kata tidak baku, singkatan, salah ketik dan penggantian huruf vokal menjadi angka. Untuk itu diperlukan sebuah algoritma yang dapat mengenali kata-kata tidak baku, menggabungkan dengan kata baku padanan pada kamus dan memasukan kata tidak baku tersebut ke dalam kamus agar dapat dikenali dengan cepat.**

Algoritma Damerau-Levenshtein Distance dapat melakukan pengoreksian otomatis dengan menghitung jarak dan memperhatikan pertukaran posisi kata sedangkan algoritma Latent Semantic Indexing (LSI) dapat mencari dan menemukan informasi berdasarkan makna keseluruhan (semantik) bukan hanya makna kata per kata. Di penelitian ini, penggabungan kedua algoritma tersebut terbukti menghasilkan sebuah modul *preprocessing teks* yang efektif dalam membangun kamus bahasa Indonesia kata tidak baku

Kata Kunci: Analisa sentimen, Semantik, preprocessing teks, media sosial, Damerau-Levenshtein Distance, Latent Semantic Indexing

I. PENDAHULUAN

Analisa sentimen atau opinion mining merupakan proses memahami, mengekstrak dan mengolah data teks tual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam suatu kalimat opini. Analisa sentimen dilakukan untuk melihat pendapat atau kecenderungan opini terhadap sebuah masalah atau objek oleh seseorang, apakah cenderung berpandangan atau beropini negatif atau positif. [1]

Media jejaring sosial seperti Twitter, Facebook, dan Youtube merupakan media sosial terpopuler di masyarakat saat ini. Media sosial merupakan sebuah media online yang memungkinkan para pengguna yang dapat berinteraksi satu sama lain. Salah satu media sosial adalah Twitter. Data yang terdapat pada *Twitter* dapat dimanfaatkan untuk memperoleh informasi tertentu, karena pengguna dapat mengungkapkan apa pun melalui Twitter, baik itu apa yang dia rasakan, apa yang sedang dialami ataupun kondisi yang terjadi di sekitarnya.

Permasalahan dalam analisa sentimen pada media sosial di Indonesia adalah ditemukan banyak kata tidak baku, singkatan dan penggantian huruf vokal menjadi angka. Sebagai contoh, sebuah kalimat "J4n64n 4d4 du5t4" membuat algoritma analisa sentimen menjadi tidak dapat memberikan hasil yang benar karena akan dianggap sebagai kata yang tidak memiliki makna pada tahap *preprocessing teks*. Untuk hal semacam ini diperlukan sebuah algoritma pencarian padanan kata tidak baku ke dalam kata baku yang sesuai dengan arti kata tersebut (semantik) dan sebuah kamus tidak baku.

Semantic search adalah metode pencarian untuk menampilkan hasil pencarian yang relevan dengan apa yang orang inginkan atau orang cari. *Semantic search* berusaha untuk meningkatkan akurasi pencarian dengan memahami maksud pencarian berdasarkan makna kontekstual sehingga menghasilkan hasil yang lebih relevan.

Ide utama dari pendekatan *semantic search* berasal dari pandangan kognitif terhadap dunia, yaitu bahwa arti dari suatu teks (kata) bergantung kepada relasi kontekstual terhadap obyek dalam dunia nyata dari pada relasi linguistik yang terdapat dalam teks atau kamus. Komponen penting dalam model ini adalah keberadaan struktur concept atau makna dari kalimat memetakan deskripsi objek informasi dengan concept yang terdapat dalam *query*. [2] Salah satu contoh sebuah *query* yang diketikan "Ford films" maka *semantic search* akan mencari juga pada artikel data yang mengandung kata *movie* karena films memiliki makna yang sama dengan *movie*.

Dalam penelitian ini dibutuhkan algoritma yang dapat mengenali suatu istilah yang berupa sinonim atau istilah yang memiliki banyak makna (polisemi). Algoritma LSI memiliki kemampuan mengenali sinonim dan polisemi. Algoritma LSI melihat istilah berdasarkan nilai semantik bukan berdasarkan sintaksis. Penggunaan teknik matematika yakni Singular Value Decomposition (SVD) menjadikan dimensi yang lebih kecil menampakkan hubungan yang mendasari istilah dan dokumen dalam semua kombinasi yang memungkinkan dan membuang ganguan yang ada pada ruang vektor [3]. Kekurangan algoritma LSI adalah tidak dapat mengenali salah pengetikan dan pengoreksian kata.

Algoritma Damerau-Levenshtein distance (DLD) memiliki kemampuan memperbaiki kata berdasarkan empat macam kesalahan pengetikan yaitu, penyisipan sebuah huruf, penghapusan huruf, penggantian huruf dengan huruf lain, dan penukaran urutan huruf [4]. Kekurangan algoritma DLD adalah hanya berfokus pada pengoreksian kata secara sintaksis, tidak memperhatikan keterkaitan makna (semantik). Penggabungan algoritma LSI dan DLD diharapkan dapat menghasilkan algoritma baru yang lebih baik.

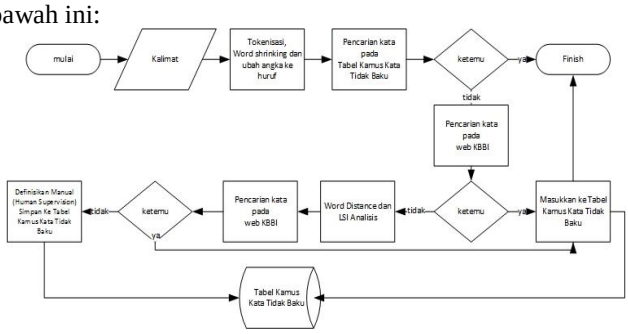
Agar kata-kata tidak baku yang telah berhasil dikenali oleh algoritma LSI-DLD maupun kata yang tidak dapat dikenali

otomatis tetapi dapat dikenali secara manual tidak perlu diproses kembali di masa depan, maka kata yang sudah dikenali disimpan ke dalam kamus bahasa indonesia kata tidak resmi..

II. METODE

A. Pembuatan Kamus Bahasa Indonesia Kata Tidak Baku (KBIKTB)

Pembangunan KBIKTB dapat dilihat dari diagram alur dibawah ini:



Gambar 1 Diagram Alur

Di **Gambar 1** menjelaskan proses penerjemahan kalimat *tweet* menjadi kalimat baku meliputi beberapa tahap yakni:

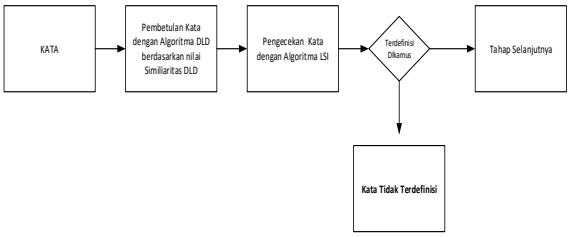
1. Crawling *tweet* berdasarkan kata kunci. Pada tahap ini modul *crawler* melakukan pengambilan data secara streaming ke *twitter* berdasarkan kata kunci yang sudah di definisikan.

2. Proses tokenisasi kalimat. Pada tahap ini dilakukan proses pemecahan kalimat menjadi kata sesuai dengan aturan bahasa Indonesia. Contoh : kalimat "Besok off kira2 Kemana tuh bocah 😷" menjadi ['besok','off','kira2','kemana','tuh','bocah','😷']

3. Word shrinking dan perubahan huruf ke angka. Pada tahap ini dilakukan penyusutan kata dan perubahan huruf ke angka. Contoh : kalimat "@kthyxg gAa 4LΔY aQ tUuh cM $4LaAaaH 94uUL 4jhAaaA h3h3" akan diubah menjadi ['@kthyxg','gaa','alay','aq','tuuh','cm','salaah','gaul','ajhaa','hehe'] dan ['@kthyxg','ga','alay','aq','tuh','cm','salah','gaul','ajha','hehe']

4. Pencarian kata pada KBBI. Pada tahap ini dilakukan Pencarian kata dalam list yang dihasilkan pada tahap sebelumnya pada database web KBBI pada alamat http://kbbi.kemdikbud.go.id/ jika ditemukan definisi maka kata tersebut adalah kata baku dan tidak ada perubahan kata ketika dimasukan ke hasil kalimat akhir pada proses 7. Jika tidak ketemu definisinya makan akan masuk tahap 5.

5. Word distance dan LSI analisis. Pada tahap ini dilakukan pembenaran kata berdasarkan *database* Kamus baru dengan metode DLD dan LSI seperti pada **Gambar 2**



Gambar 2 Modul Word Distance dan LSI

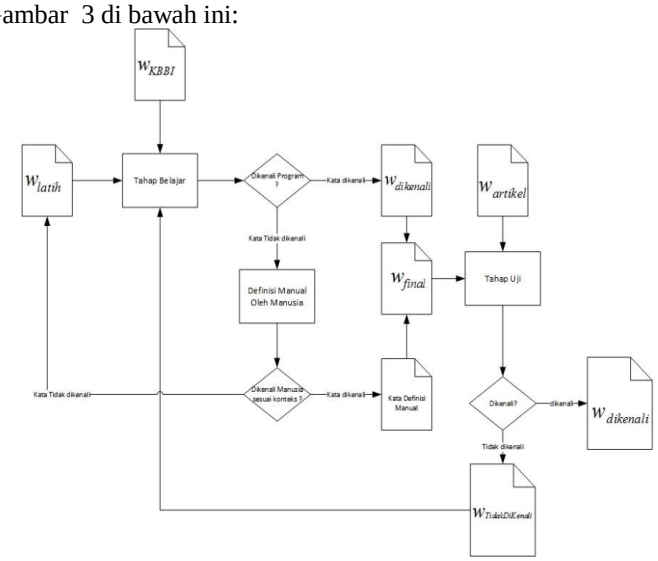
Kata akan dilakukan pembenaran dengan algoritma DLD dan Algoritma LSI, jika kata terdefinisi maka kata akan dimasukkan ke kamus baru dan kata awal diubah menjadi kata baru untuk proses 7. Jika tidak masuk ke tahap selanjutnya

6. Tahap definisi manual merupakan tahapan pendefinisian kata-kata yang tidak dapat didefinisi oleh program oleh manusia dan memasukkannya ke database Kamus baru untuk digunakan sebagai definisi jika program menemukan kata tersebut pada proses penerjemahan selanjutnya.

7. Tahap kalimat lengkap adalah proses penyatuan kata baku dan kata baru sebagai pengganti kata tidak baku pada tahap-tahap sebelumnya menjadi kalimat baku.

B. Tahapan Pembuatan Modul preprosessing Teks

Tahapan pembuatan modul preprosessing teks seperti pada Gambar 3 di bawah ini:



Gambar 3 Tahapan Pembangunan Modul Preprosessing Teks

Gambar 3 menjelaskan proses yang dilakukan dalam pengujian pada tahap belajar dan tahap uji :

1. Tahap belajar, pada tahap ini program dilakukan pengujian dalam mengenali kata tidak baku (w_{latih}) data latih dengan menggunakan kata dari KBBI

(w_{KBBI}). Kata yang tidak dikenali akan dilakukan definisi manual oleh manusia, kata yang dikenali oleh manusia akan disatukan dengan kata yang dikenali oleh program menjadi kata yang dikenali akhir (w_{final}) dan disimpan ke dalam kamus bahasa Indonesia Kata Tidak Baku. Kata yang tidak dikenali oleh manusia akan masuk kedalam $w_{latih.}$

2. Tahap uji, pada tahap ini kata hasil tahap belajar (w_{final}) digunakan program dalam pengujian kata tidak baku pada data uji ($w_{artikel}$). Kata tidak baku yang dikenali ($w_{dikenali}$) akan dirubah menjadi kata baku, sedangkan kata yang tidak dikenali ($w_{tidakdikenali}$) akan kembali ke tahap belajar.

Berdasarkan pemaparan **Error! Reference source not found.** di atas pengujian program akan mengukur efektivitas program dalam mengenali kata tidak baku. Adapun rumus perhitungan persentase efektivitas adalah :

$$\text{efektivitas} = \frac{w_{final}}{w_{latih}} \text{ x } 100\% \quad \textbf{(1)}$$

Keterangan :
efektivitas = Jumlah persentase efektivitas
w_{final} = Jumlah kata yang dikenali final
w_{latih} = Jumlah kata latih

Dari rumus di atas semakin besar w_{final} semakin tinggi efektivitas program dalam mengenali kata tidak baku.

$$\Sigma\text{perubahan Huruf} = \left\lfloor \frac{persentase\ distorsi * panjang\ kata}{100} \right\rfloor \quad \textbf{(2)}$$

Teknik pengujian penelitian pada tahap belajar dilakukan dengan cara memasukan kalimat baku yang dibuat menjadi kalimat tidak baku dengan cara merubah kata baku menjadi kata tidak baku dengan melakukan metode sebagai berikut :

1. Melakukan penghilangan huruf vokal seperti kata **'yang'** menjadi **'yng'**.
2. Melakukan penggeseran posisi huruf pada kata seperti **'yang'** menjadi **'ynag'** atau **'yagn'**.
3. Melakukan perubahan kata menjadi kata singkatan yang sudah umum seperti **'yang'** menjadi **'yg'**.
4. Mengganti huruf menjadi angka yang umum digunakan sebagai mengganti seperti huruf 'a' dengan angka 4 huruf 'g' dengan angka 9 sehingga **'yang'** menjadi **'y4n9'**.

Rumus perhitungan banyak huruf kata tidak baku dengan melakukan gangguan (distorsi) menggunakan 4 langkah di atas adalah:
Contoh kata 'Namun' dengan distorsi 30% makan menghasilkan sebagai berikut:

$$\Sigma\text{perubahan Huruf} = \left\lfloor \frac{30*5}{100} \right\rfloor = \lfloor 1.5 \rfloor = 1$$

Hasil perubahan kalimat tidak baku akan dilakukan pengujian pada program. Pengujian dilakukan dengan data latih menggunakan data dari kata baku dari artikel *website* tempo yang didapat dari internet. Pengujian dilakukan dengan kondisi awal kamus tidak baku kosong dan hanya menggunakan kamus KBBI *online* dalam melakukan proses eksekusi program.
Hasil pengujian akan dimuat dalam grafik pengujian kemampuan program dalam mengenali dan mengoreksi kata tidak baku.

III. HASIL DAN PEMBAHASAN

A. Sumber Data

Sumber data pengujian pada penelitian ini adalah data kalimat baku yang berupa artikel berita website tempo. Data tempo diambil dari alamat http://ilps.science.uva.nl/ilps/wp-content/uploads/sites/6/files/bahasaindonesia/tempo.zip.
Kalimat baku ini akan dilakukan perubahan menjadi kata tidak baku untuk pengujian.

B. Proses Distorsi

Perubahan kata baku menjadi kata tidak baku pada penelitian ini dilakukan dengan melakukan gangguan (distorsi) sebesar 10% - 50%. Distorsi dilakukan dengan cara sebagai berikut:

1. Melakukan penghilangan huruf vokal.
2. Melakukan penggeseran posisi huruf.
3. Melakukan perubahan kata menjadi kata singkatan yang sudah umum.
4. Mengganti huruf menjadi angka.

Cara di atas dilakukan secara acak pada kalimat sesuai dengan nilai distorsi yang diberikan. Hasil perubahan kata baku menjadi kata tidak baku bisa dilihat pada tabel dibawah ini:

Tabel 1 Perubahan Kata

Kata Asli	Perubahan Kata Dengan Distorsi				
	10%	20%	30%	40%	50%
Namun hingga berita ini diturunkan acara perganti an mulur.	Namun hingga berita ini diturunkna acara p3rganti an mulur.	N4mun ihngga beriat ini d1trunkan 4cara 3prganti an mlur.	Nmun hngga brita ini ditruunakn 4cara p3ragnti an mluur.	N4mn h1gnga 8rita 1ni dtrunk 4n cara pgranti na mlru	Nmn nhgga e8rita 1ni dtrunk 4n c4ara pgrntia n mulur.

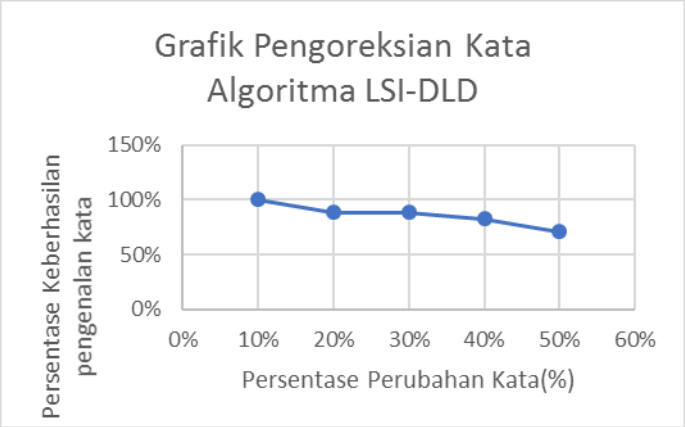
C. Pengujian

Pengujian ini meliputi pembahasan pengujian dalam pembangunan kamus bahasa Indonesia tidak resmi dengan melakukan pengujian kemampuan program dalam melakukan pengoreksian kata tidak baku menjadi kata baku pada saat pembuatan Kamus bahasa Indonesia tidak resmi seperti tabel di bawah ini:

Tabel 2 Tabel Pengujian

Persentase Perubahan Kata	Persentase Keberhasilan
10%	100%
20%	88%
30%	88%
40%	82%
50%	71%

Dalam bentuk grafik dapat dilihat pada di bawah ini :



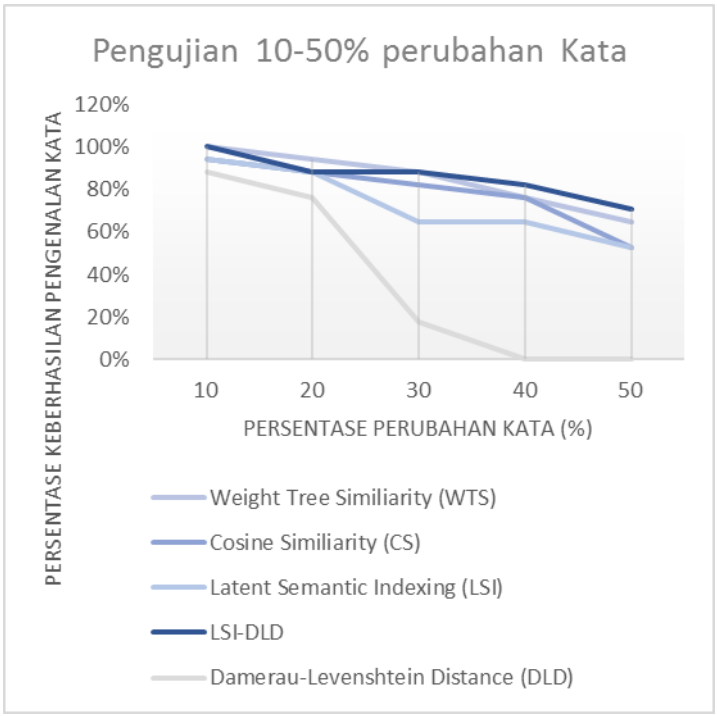
Gambar 4 Grafik Pengoreksian Kata

Di **Gambar 4** dapat diketahui algoritma LSI-DLD mengenali kata sangat baik pada distorsi kata sebagai berikut:

1. Pada distorsi 10% memiliki tingkat keberhasilan sebesar 100%.
2. Pada distorsi 20% memiliki tingkat keberhasilan sebesar 88%.
3. Pada distorsi 30% memiliki tingkat keberhasilan sebesar 88%.
4. Pada distorsi 40% memiliki tingkat keberhasilan sebesar 82%.
5. Pada distorsi 50% memiliki tingkat keberhasilan sebesar 71%.

Hal ini menunjukkan bahwa tingkat keberhasilan algoritma LSI-DLD sangat bagus dengan persentase keberhasilan di atas 70% dengan level distorsi 50%.

Sedangkan perbandingan dengan algoritma penelitian sebelumnya dapat dilihat pada Gambar 5



Gambar 5 Perbadingan Algoritma

Di **Gambar 5** di atas dapat diketahui algoritma LSI-DLD mengenali kata sangat baik pada gangguan (distorsi) kata sebagai berikut:

1. Pada distorsi 10% algoritma Weight Tree Similiarity (WTS) dan algoritma LSI-DLD memiliki tingkat keberhasilan sebesar 100%, algoritma Cosine Similiarity (CS) dan LSI memiliki tingkat keberhasilan sebesar 94%, dan algoritma DLD memiliki tingkat keberhasilan sebesar 88% .
2. Pada distorsi 20% algoritma WTS memimpin dengan tingkat keberhasilan sebesar 94% dikuti oleh algoritma LSI-DLD, CS, dan LSI dengan tingkat keberhasilan sebesar 88%, sedangkan algoritma DLD berada diposisi akhir dengan tingkat keberhasilan sebesar 76%.
3. Pada distorsi 30% algoritma LSI-DLD dan WTS memimpin dengan tingkat keberhasilan sebesar 88% dikuti oleh algoritma CS dengan tingkat keberhasilan sebesar 82%, diikuti oleh algoritma LSI dengan tingkat keberhasilan sebesar 65%, sedangkan algoritma DLD berada diposisi akhir dengan tingkat keberhasilan sebesar 18%.
4. Pada distorsi 40% algoritma LSI-DLD memimpin dengan tingkat keberhasilan sebesar 82% dikuti oleh algoritma WTS dan CS dengan tingkat keberhasilan sebesar 76%, diikuti oleh algoritma LSI dengan tingkat keberhasilan sebesar 65%, sedangkan algoritma DLD berada diposisi akhir dengan tingkat keberhasilan sebesar 0%.
5. Pada distorsi 50% algoritma LSI-DLD memimpin dengan tingkat keberhasilan sebesar 71% dikuti oleh algoritma WTS dengan tingkat keberhasilan sebesar

65%, CS dan LSI dengan tingkat keberhasilan sebesar 53%, sedangkan algoritma DLD berada diposisi akhir dengan tingkat keberhasilan sebesar 0%.

Dari pemaparan di atas dapat diketahui algoritma LSI-DLD menunjukkan kinerja lebih baik pada saat ada perubahan kata dari 30% ke atas. Hal ini menunjukkan penggunaan algoritma LSI-DLD dalam pembangunan kamus bahasa Indonesia kata tidak baku sangat efektif.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Penggunaan LSI - DLD memiliki kinerja yang cukup baik mengenali kata tidak baku dan membuat relasinya dengan kata baku dengan nilai persentase yang stabil setelah mengalami gangguan perubahan kata sebesar 30% hal ini menunjukkan bahwa algoritma LSI - DLD efektif.

B. Saran

Saran pengembangan penelitian ini adalah sebagai berikut:

1. Pada penelitian ini pembangunan index LSI membutuhkan proses yang sangat lama ketika data kamus sudah banyak, sehingga perlu diselidiki lebih lanjut mengenai metode optimasi pembangunan index algoritma LSI.
2. Penggunaan algoritma LSI-DLD masih mempunyai kelemahan pada suatu penentuan makna jika kalimat tidak baku ada 2 kata yang berbeda makna tetapi kalimatnya tidak utuh seperti "jk menyuruh telkomsel perbaiki layanan jk ingin terus dpke". Algoritma merubah kata 'jk' menjadi 'jika' padahal seharusnya kata 'jk' pertama adalah Jusuf Kalla.

V. REFERENSI

[1] "Implementasi Opinion Mining (Analisis Sentimen) untuk Ekstraksi Data Opini Publik pada Perguruan Tinggi," *Jurnal EECCIS Vol. 6, No. 1, Juni 2012,* vol. 6, no. 1, p. 37, 2012.

[2] H. Sembiring, "Universitas Sumatra Utara," 2011. [Online]. Available: http://repository.usu.ac.id/bitstream/123456789/29092/4/Chapter%20II.pdf. [Diakses 01 07 2016].

[3] E. Ferdian, R. Hadisaputra dan N. Madjid, "Penerapan Metode Latent Semantic Indexing Pada Search Engine," 2005.

[4] A. Adnani, DETEKSI KATA GAUL BAHASA INDONESIA DAN PEMANFAATANNYA PADA TEXT PREPROCESSING DALAM TEXT MINING, Bandung: ITB, 2016.

[5] A. Chaer, Pengantar Semantik Bahasa Indonesia, Jakarta : Rineka Cipta, 2009.

[6] R. Sarno dan F. Rahutomo, "PENERAPAN ALGORITMA WEIGHTED TREE SIMILARITY UNTUK PENCARIAN SEMANTIK," 2008.

[7] P. Resnik, "Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language," *Journal of Artificial In telligence Research 11,* 1999.

[8] H. Bunyamin, Information Retrieval System dengan metode Latent Semantic Indexing, Bandung: ITB, 2005.

[9] Y. Rochmawati dan R. Kusumaningrum, "Studi Perbandingan Algoritma Pencarian String dalam Metode Approximate String Matching untuk Identifikasi Kesalahan Pengetikan Teks," *Jurnal Buana Informatika,* vol. 7, no. 2, pp. 125-134, 2016.