Analisis dan Implementasi Pengukuran Semantic Relatedness Menggunakan Salient Semantic Analysis dengan Keyword Extraction Sebagai Preprocessing

Moch. Arif Bijaksana, Ir., M.Tech
Fakultas Teknik
Universitas Telkom

Bagus ArdiSaputra
S1 Teknik Informatika
Fakultas Teknik
Universitas Telkom
Bandung, Indonesia
aoinoyuki07@gmail.com

Siti Sa'adah, ST., MT.
Fakultas Teknik
Universitas Telkom

Abstrak

Pengukuran besar relasi antar kata atau teks mempunyai kendala pada pengimplementasian sistem yaitu memerlukan pengetahuan dalam jumlah banyak. Sistem yang ada juga harus dapat melakukan abstraksi dan generalisasi. Pengukuran ini yang disebut sebagai semantic relatedness. Semantic relatedness adalah salah satu pengukuran dalam text mining yang merepresentasikan hubungan antar kata atau teks. Pada tugas akhir ini digunakan metode salient semantic analysis dan keyword extraction untuk mendapatkan nilai semantic relatedness antar kata dan antar teks. Metode ini dipilih karena mampu menghitung nilai semantic relatedness antar kata dan teks dengan tingkat akurasi yang baik. Dari uji skenario yang dilakukan, didapatkan nilai korelasi tertinggi untuk antar kata sebesar 0.2137 dan akurasi tertinggi untuk antar teks sebesar 83.3%.

Kata kunci : semantic relatedness, salient semantic analysis, keyword extraction

I. Pendahuluan

Semantic relatedness merupakan salah satu pengukuran dalam text mining yang merepresentasikan hubungan antar kata atau teks. Semantic relatedness mempunyai tugas untuk mencari dan mengukur besarnya relasi semantic atau makna antar unit tekstual[8]. Hubungan antar kata dapat digambarkan dalam suatu pohon yang disebut pohon taksonomi. Besarnya relasi antar kata dapat diukur dengan menghitung besar relasi keduanya pada pohon taksonomi dengan menggunakan seluruh bentuk relasi[5].

Pertanyaan yang berkaitan dengan relatedness terkadang muncul dalam benak setiap orang seperti bagaimana hubungan antara "car" dan "train" atau "vacation" and "beach". Untuk menentukan hubungan secara semantic, manusia tidak dapat mengukurnya hanya dari kata saja[2] melainkan dengan menggunakan kumpulan pengetahuan dan pengalaman, pemikiran yang konseptual, serta abstraksi dan generalisasi[8]. Hal ini dapat menjadi masalah ketika diimplementasikan ke dalam sebuah sistem. Untuk membangun suatu sistem yang mampu mengukur semantic relatedness antar kata ataupun teks dengan tepat diperlukan pengetahuan yang besar dan menjangkau berbagai kategori[2]. Akan tetapi, hal itu tidaklah cukup untuk dapat menentukan nilai relatedness dengan tepat. Diperlukan suatu kemampuan untuk menggunakan pengetahuan yang ada dan dapat mengeneralisasikannya[8] sehingga sistem dapat menentukan nilai relatedness dengan tepat.

Wikipedia merupakan salah satu ensiklopedia yang dapat diakses secara online untuk mendapatkan berbagai informasi yang dibutuhkan. Wikipedia sebagai salah satu sumber pengetahuan terbesar dan saat ini terus berkembang, dapat digunakan oleh sistem untuk menentukan nilai semantic relatedness antar kata maupun antar teks. Wikipedia menyediakan berbagai artikel dalam jumlah banyak dan dalam berbagai kategori sehingga Wikipedia layak digunakan sebagai sumber pengetahuan bagi sistem yang akan dibangun.

Pada penelitian ini, metode salient semantic analysis diujikan bersama dengan keyword extraction yang digunakan sebagai preprocessing untuk mengetahui ketepatannya dalam mengukur semantic relatedness antar kata maupun antar teks berbahasa Inggris. SSA terbukti mampu memberikan akurasi yang baik dalam perhitungan semantic relatedness antar kata maupun teks[8]. Sedangkan metode keyword extraction yang digunakan terbukti mampu memberikan keyword berkualitas dan mampu menyaingi hasil yang diberikan oleh metode tfidf walaupun tanpa menggunakan corpus[10].

II. Dasar Teori

Semantic relatedness merupakan salah satu pengukuran dalam text mining yang merepresentasikan hubungan antar kata atau teks. Tujuan utama dari pengukuran semantic relatedness adalah untuk menemukan dan mengukur kuantitas keterhubungan antar kata atau antar teks secara semantic atau makna [8]. Berbeda dengan semantic similarity, semantic relatedness memiliki cakupan yang lebih luas. Maka, suatu kata yang secara semantic similarity tidak mempunyai hubungan, pada semantic relatedness dapat memiliki nilai relatedness yang besar.

Semantic similarity merupakan bagian dari semantic relatedness yang mengukur relasi antar kata berdasarkan kemiripannya. Sebagai contoh kata "sea" dan "ocean", kedua kata tersebut merupakan hal yang sama sehingga memiliki semantic similarity dan semantic relatedness yang tinggi karena semantic similarity adalah bagian dari semantic relatedness.

III. Metode

Pada penelitian ini, beberapa metode telah digunakan antara lain keyword extraction menggunakan word co-occurrence dan salient semantic analysis. Penjelasan setiap metode adalah sebagai berikut:

A. Keyword Extraction

Keyword extraction merupakan suatu teknik yang digunakan untuk melakukan ekstrasi terhadap term pada suatu teks yang dapat mewakili apa yang ingin disampaikan dari teks tersebut, term ini yang disebut sebagai suatu keyword. Metode yang akan digunakan adalah keywords extraction dengan memanfaatkan informasi statistik dari word co-occurrence. Metode ini terdiri dari lima buah

tahapan yang pada akhirnya akan menghasilkan beberapa term yang merupakan keywords dari suatu teks. Tahapan dari metode keywords extraction using word co-occurrence statistical information adalah sebagai berikut[10]:

1. Tahap preprocessing

   Pada tahapan ini dilakukan stemming yang merupakan proses untuk mendapatkan kata dasar dari semua kata yang ada dalam teks dan dilanjutkan dengan menghilangkan semua kata

   yang merupakan stop word yang disebut dengan stopword removal sehingga didapatkan semua term yang membentuk teks tersebut.
2. Pemilihan top frequent term
   Pada tahapan ini dilakukan pemilihan term yang didapatkan dari tahapan preprocessing sebanyak 30% dari jumlah term yang ada yang mempunyai frekuensi kemunculan terbanyak.
3. Menghitung expected probability
   Dengan menghitung jumlah dari term co-occurrence dengan term dari top frequent term C dibagi dengan jumlah total term didapatkan nilai dari expected probability [10].

   = / (1)

   dengan,
   :
   : h –
   ∈
   : h

   Pada tahap ini, jumlah total term dan top frequent term telah didapatkan melalui tahap yang sebelumnya. Maka, yang perlu dilakukan pertama kali pada tahap ini adalah mencari jumlah term yang muncul bersama atau co-occurrence dengan term yang termasuk dalam top frequent term.
   Misal:
   Total term = 30
   Top frequent term = car
   Term "car" muncul di satu line atau baris kalimat yaitu:

   car vehicle ground wheel luxuri countri

   dari line kalimat tersebut sebanyak enam term co-occurring dengan term "car" termasuk term "car" itu sendiri. Sehingga dapat dilakukan perhitungan expected probability dengan menggunakan formula 2.1 sebagai berikut:

   = /
   = 6 / 30
   = 0.2

   Dari contoh ilustrasi didapatkan nilai expected probability untuk top frequent term "car" sebesar 0.2 .

4. Menghitung nilai dari setiap term
   Pada tahapan ini dilakukan perhitungan terhadap nilai dengan menggunakan formula sebagai berikut [10]:

   ( ) = ( ) – ∈ ((,)) (2)

   dengan,
   ( ):
   ( ) :
   ((,))
   ∈ :

   Dengan terlebih dahulu mencari nilai dari dengan menggunakan formula sebagai berikut[10]:

   ( ) = Σ∈ ((,)) (3)

   dengan,
   ( ) :
   (,):
   –
   :
   –

   Misal:
   Top frequent c = car, engine
   Term w = avanza

	car	engine
avanza	1	0.5

   Nilai dari term "avanza" dengan top frequent "car" didapatkan dengan menggunakan formula 2.3 tanpa penjumlahan seluruh top frequent c.
   freq (avanza,car) = 2
   = 20
   = 0.2

   avanza – car = ((,))
   = ( ) = 1

   Maka, didapatkan nilai ( ) = 1.5. Nilai ini kemudian digunakan dalam perhitungan dengan menggunakan formula 2.2.

   ( ) = 1.5 – 1 = 0.5

5. Pemilihan keywords
   Pada tahapan ini dilakukan pemilihan keywords dari term yang ada dengan nilai yang besar.

Term yang diambil adalah sebanyak 20 term dengan mempertimbangkan ukuran teks yang digunakan dan semantic profile yang dibangun.

B. Salient Semantic Analysis

Salient Semantic Analysis merupakan suatu novel unsupervised method untuk mengukur semantic relatedness. Secara umum salient semantic analysis mempunyai dua tahap utama untuk dapat mengukur semantic relatedness antar teks. Kedua tahap tersebut adalah membangun semantic profile dari artikel Wikipedia yang kemudian akan digunakan dalam perhitungan semantic relatedness antar kata dan perhitungan terhadap text to text relatedness yang akan memberikan nilai semantic relatedness antar teks.

Dalam membangun semantic profile, sejumlah besar artikel dari Wikipedia dikumpulkan kemudian dibersihkan secara manual. Pembersihan secara manual dilakukan karena rekomendasi dari Wikipedia untuk tidak menggunakan regular expression yang disebabkan sulitnya melakukan maintenance. Kemudian dilanjutkan proses stopword removal, stemming, dan eliminasi link untuk mendapatkan salient concepts yang ada dari semua artikel yang ada. Semua salient concepts yang didapatkan kemudian difilter kembali melalui tiga tahapan utama yaitu sebagai berikut[8]:

1. Ekstraksi menggunakan manual links
   Pada tahap ini semua salient concepts diekstrak berdasarkan link yang diberikan oleh para contributor Wikipedia ataupun secara automatic annotation. Link ini dapat memberikan suatu sinyal bahwa concepts tersebut merupakan bagian yang penting dari artikel dan tidak ambigu.
2. Pencarian term menggunakan one sense per discourse heuristic[9]

   Pada tahap ini dilakukan pencarian terhadap terhadap link dan term dengan memperhatikan kemunculan term yang sama dalam satu konteks bahwa term tersebut mempunyai makna yang sama dan terhubung dengan artikel yang sama.
3. Eliminasi link menggunakan disambiguation method

   Pada tahap ini dilakukan eliminasi link dengan menghitung peluang saliency dengan membagi jumlah kemunculan kata atau frase yang ada dalam link dibagi dengan jumlah kemunculan kata atau frase tersebut dalam Wikipedia. Kata atau frase dengan nilai peluang kurang dari 0.5 tidak dimasukkan dalam daftar salient concept.

Tahap berikutnya dari salient semantic analysis adalah melakukan perhitungan nilai semantic relatedness antar teks. Tahap ini diawali dengan melakukan preprocessing terhadap teks yang digunakan sebagai input untuk mendapatkan semua term yang ada pada teks dilanjutkan dengan menghitung jumlah shared term yang digunakan oleh kedua teks. Kemudian dilakukan dengan perhitungan semantic relatedness dari semua kombinasi non-shared term kedua teks. Untuk mendapatkan nilai semantic relatedness dari kombinasi non-shared term, perlu dibangun sebuah semantic profile. Dalam membangun semantic profile, salient semantic analysis menggunakan metric yang telah dimodifikasi yaitu cosine similarity sebagai bahan evaluasi. Tahapan dalam membangun semantic profile adalah sebagai berikut[8]:

1. Membangun matriks E
   Pada tahap ini dilakukan pembangunan matriks E yang menggambarkan jumlah akumulasi dari frekuensi co-occurrence setiap term yang ada pada corpus [8].

   $$= (,) \quad (4)$$

   dengan,
   :
   : h –

   Misal:
   Term w = river
   Concept c = beach
   Artikel:
   beach landform coast ocean sea lakeriver consist loos particl compos rock sand gravel
   lake alongsid larg river beach refer small system rock materi move onshor offshor
   alongshor forc wave current geolog unit consider size describ detail larger

   Term "river" dan concept "beach" muncul bersama atau co-occurrence pada 2 baris kalimat sehingga nilai elemen pada matrik E yaitu
   , = 2 .

2. Membangun matriks P
   Pada tahap ini, matriks E digunakan untuk membangun matriks P dengan formula sebagai berikut[8]:

   $$= \quad ( ) \quad ( ) \quad (5)$$

   dengan,
   :
   :
   ( ):
   :

   , :

   Misal:
   Term w = river
   Concept c = beach
   (, h) = 2
   Token m = 2500
   ( ) = 2
   (h) = 4

   $$, = \frac{(, h)}{( ) \quad (h)}$$

   $$= \frac{\quad}{\quad}$$

   $$= 625 = 9.287$$

3. Perhitungan semantic relatedness
Pada tahap ini, perhitungan semantic relatedness antar kata dapat menggunakan formula cosine similarity yang telah dimodifikasi[8].

$$(,) = \frac{\Sigma \quad (\qquad)}{\Sigma \qquad \Sigma} \quad (6)$$

dengan,
(,)
:
, :
:

Misal:
= 0.5
Matrik P :

	beach	car
river	9.287	1.5
road	2.3	6.7

$$= \frac{(\quad , \quad)}{\Sigma \quad (\quad , \quad ,)}{\Sigma \quad , \quad \Sigma \quad ,}$$

$$= \frac{(\quad)(\quad)}{(\quad)(\quad)} = \frac{\quad}{\quad} = 0.3305$$

Setelah mendapatkan nilai semantic relatedness dari semua kombinasi pasangan non-shared term kedua teks, dilanjutkan dengan melakukan perhitungan semantic relatedness dari kedua teks dengan menggunakan formula yang telah dimodifikasi yaitu sebagai berikut[8]:

$$(,) = \frac{(\quad \Sigma^{1} \quad) \quad (\quad)}{} \quad (7)$$

dengan,

(,):
: h h
, :
: –

Misal:
Nilai w = 2
| |

Σ = 5.674
Size a = 166
Size b = 150

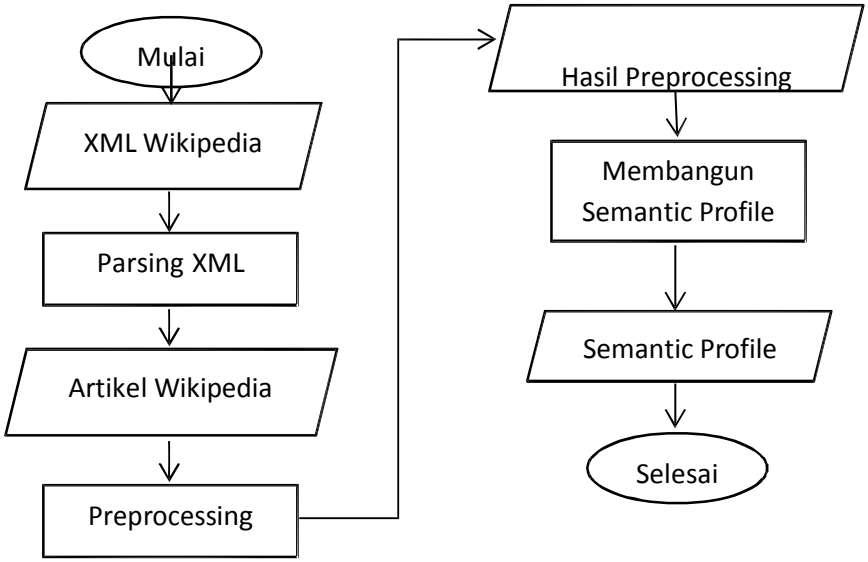
$$(,) = \frac{(2 + 5.674) \quad (2 \quad 166 \quad 150)}{166 + 150}$$

$$= 1209.38354$$

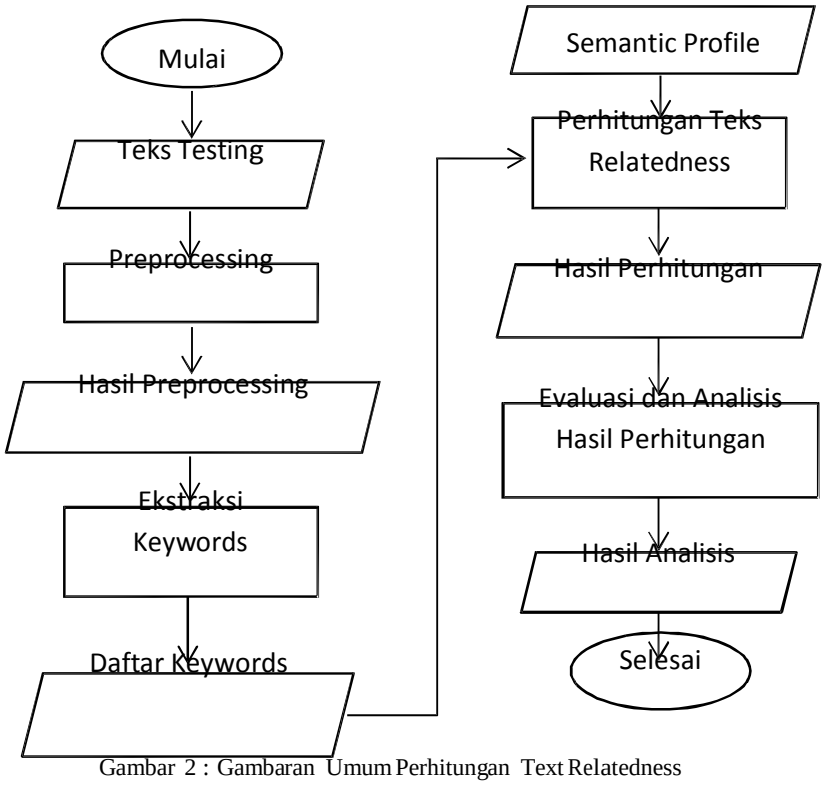
IV. Pembahasan

A. Perancangan Sistem

Dalam penelitian ini, dibangun sebuah sistem yang digunakan untuk melakukan perhitungan nilai relatedness antar kata dan teks secara semantic atau makna. Gambaran umum sistem dapat dilihat pada gambar 1 dan gambar 2.



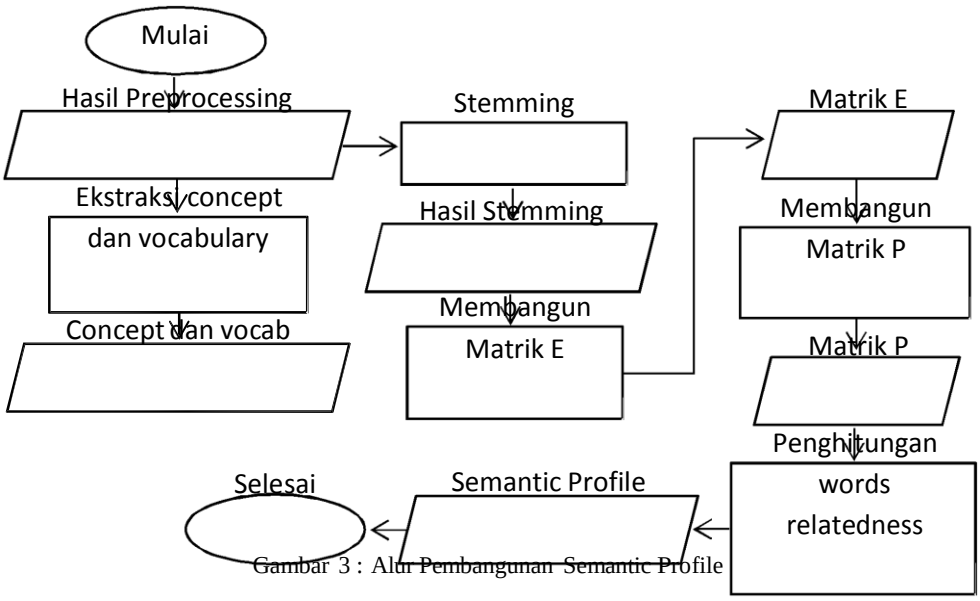
Gambar 1 : Gambaran Umum Pembangunan Semantic Profile



Gambar 2 : Gambaran Umum Perhitungan Text Relatedness

Sistem yang dibangun mempunyai dua bagian yang berbeda yaitu proses pembangunan semantic profile yang merupakan kamus kata yang terdiri dari pasangan kata beserta nilai semantic relatedness-nya yang dapat dilihat pada gambar 1 dan ekstraksi keyword dan perhitungan text relatedness yang dapat dilihat pada gambar 2.

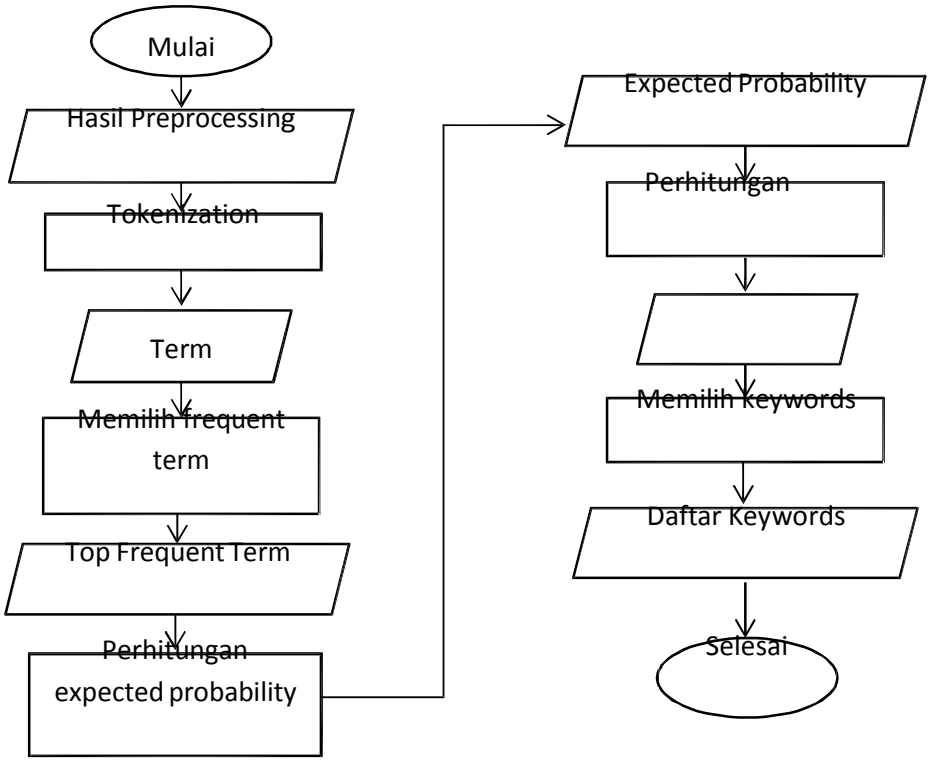
Pembangunan semantic profile dibangun dengan menggunakan artikel Wikipedia yang telah diekstrak concept beserta vocabulary-nya yang kemudian dibangun matrik E dan matrik P yang akan menjadi komponen dalam pembangunan semantic profile seperti pada gambar 3.



Gambar 3 : Alur Pembangunan Semantic Profile

Matrik E merupakan jumlah kemunculan bersama antara vocab dan concept pada k-window yang telah ditentukan. Sedangkan matrik P merupakan nilai logaritma dari elemen matrik E dikalikan dengan jumlah token dari keseluruhan corpus dan dibagi dengan perkalian dari jumlah dokumen kemunculan vocab dan concept masing – masing dalam seluruh dokumen corpus.

Pada tahap keyword extraction, semua teks yang akan digunakan dalam text relatedness akan dilakukan preprocessing terlebih dahulu dan dilakukan tokenization. Tahapan dari keyword extraction dapat dilihat pada gambar 4. Keyword extraction dimulai dengan melakukan tokenization untuk mendapatkan semua term unik yang disebut sebagai running term beserta dengan frekuensi kemunculannya. Sebanyak 30% dari total running term dipilih sebagai top frequent term. Setiap term yang ada dalam daftar top frequent term dicari nilai expected probability dengan membagi jumlah term yang muncul bersama dengan term pada top frequent dalam satu baris kalimat dengan total running term sehingga didapatkan expected probability masing – masing term pada top frequent. Dengan menggunakan formula 2 dicari nilai x^'2 dari setiap term di running term menggunakan seluruh term yang ada pada top frequent beserta nilai expected probability-nya. Setelah mendapatkan nilai x^'2 dari masing-masing term, dilakukan sorting berdasarkan nilai dari x^'2 sehingga didapatkan daftar term yang telah terurut. Term dengan nilai x^'2 tertinggi adalah keyword dari teks yang digunakan, jumlah term yang digunakan sebagai keyword bergantung dari kebutuhan masing – masing tugas yang dilakukan. Dalam penelitian ini digunakan 20 term yang diambil sebagai keyword berdasarkan pertimbangan ukuran teks dan semantic profile.



Gambar 4 : Alur Ekstraksi keyword dari Teks

Pada tahap text relatedness, daftar keyword dan semantic profile digunakan sebagai komponen dalam perhitungan. Tahap ini dimulai dari menghitung jumlah shared keyword dari kedua teks yang diujikan. Shared keyword ini adalah keyword yang ada pada teks pertama dan ada juga pada teks kedua, keyword ini kemudian dikeluarkan dari daftar keyword setelah seluruh jumlahnya didapatkan. Selanjutnya, keyword yang tersisa dikombinasikan membentuk pasangan keyword yang unik. Pasangan ini yang disebut dengan pasangan non-shared keyword. Setiap pasangan ini dicari nilai relatedness-nya dalam semantic profile yang telah dibangun. Setelah didapatkan seluruh nilai relatedness pasangan keyword, nilai tersebut dijumlahkan. Setelah mendapatkan seluruh komponen yang dibutuhkan, dilakukan perhitungan dengan menggunakan formula 7 dengan a dan b adalah ukuran dari masing- masing teks. Ukuran yang dimaksud adalah jumlah token dalam teks setelah proses preprocessing. Secara lebih jelas proses text relatedness dapat dilihat pada gambar 5.

Setelah melakukan seluruh proses atau tahapan, selanjutnya dilakukan pengujian untuk word relatedness dan text relatedness.

B. Hasil Pengujian

Pada penelitian ini, tujuan dari pengujian yang dilakukan adalah sebagai berikut:

1. Menganalisis pengaruh nilai k-window terhadap semantic profile yang terbentuk.
2. Menganalisis pengaruh nilai gamma terhadap semantic profile yang terbentuk.
3. Menganalisis hasil perhitungan semantic relatedness antar teks.

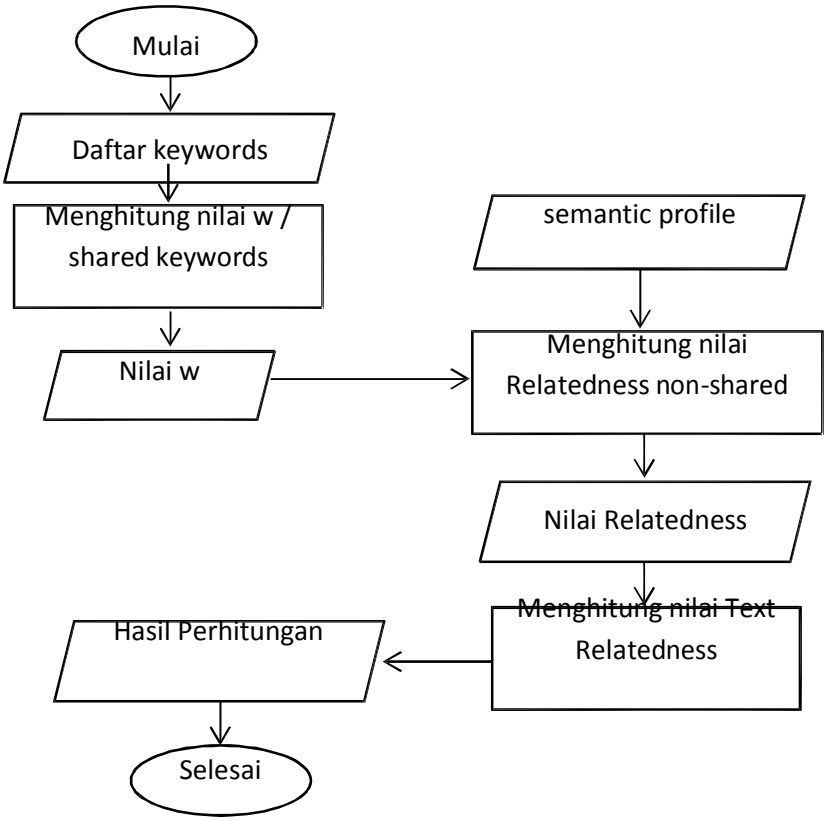
Dari hasil skenario pengujian yang telah dilakukan, ditemukan bahwa semakin besar nilai k-window atau k-words yang digunakan semakin besar pula nilai korelasi yang diberikan. Selain itu, dengan semakin besar nilai k semakin banyak pasangan kata yang terbentuk dalam semantic profile sehingga dapat memberikan nilai relatedness terhadap lebih banyak pasangan kata. Hal ini terjadi karena peluang kemunculan bersama antar vocab dan concept semakin besar sehingga pasangan kata yang terbentuk dalam semantic profile semakin banyak dan semakin besar nilai relatedness-nya. Pada pengujian pengaruh nilai k didapatkan nilai korelasi tertinggi sebesar 0.186 dan 0.213 untuk gamma masing – masing adalah 0.05 dan 1 dengan nilai k sebesar 24. Hasil pengujian terhadap pengaruh nilai k dapat dilihat pada tabel 1 dan 2.

Tabel 1: Perbandingan korelasi gamma 0.05

Nilai k	Correlation
12	0.169
24	0.186

Tabel 2: Perbandingan korelasi gamma 1

Nilai k	Correlation
12	0.210
24	0.213



Gambar 5 : Alur Perhitungan Nilai Text Relatedness

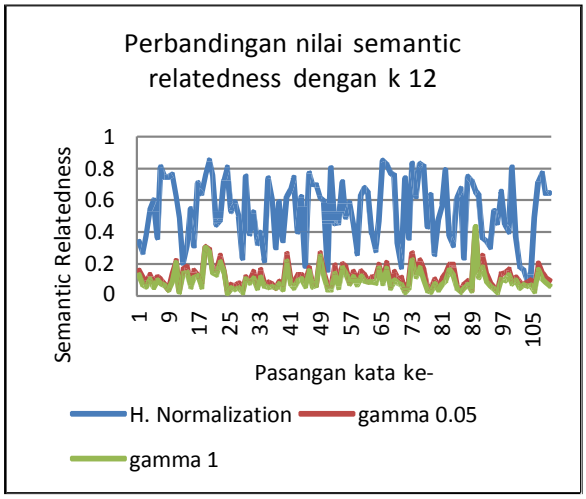
Dari hasil skenario pengujian yang telah dilakukan dapat dilihat dari hasil yang didapatkan bahwa semakin besar nilai gamma semakin besar pula nilai korelasi yang didapatkan walaupun nilai semantic relatedness yang diberikan dapat menjadi lebih rendah. Hal ini terjadi karena nilai gamma berperan sebagai pengontrol bias dari nilai semantic relatedness sehingga mengakibatkan nilai semantic relatedness yang diberikan dapat menjadi lebih rendah tapi nilai korelasi meningkat. Dari pengujian di atas didapatkan nilai korelasi tertinggi yaitu 0.210 dan 0.213 dengan nilai gamma sebesar 1. Hasil ini dapat dilihat pada tabel 3 dan 4.

Setelah melakukan analisis terhadap pengaruh k-window dan pengaruh nilai gamma, selanjutnya dilakukan analisis terhadap hasil perhitungan semantic relatedness

antar teks dengan cara membandingkan hasil perhitungan sistem dengan perkiraan secara manual. Ringkasan hasil dari semua pengujian di atas dapat dilihat pada tabel 5.

Tabel 3 : Perbandingan Nilai Korelasi k 12

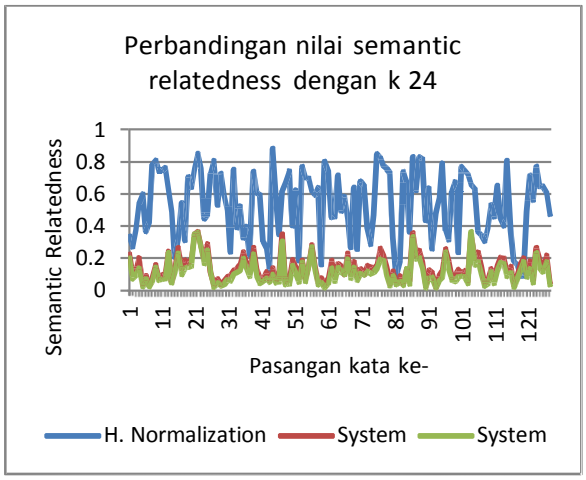
Nilai gamma	Correlation
0.05	0.169
1	0.210



Gambar 6 : Grafik Perbandingan Nilai Semantic Relatedness dengan k 12

Tabel 4 : Perbandingan Nilai Korelasi k 24

Nilai gamma	Correlation
0.05	0.186
1	0.213



Gambar 7 : Grafik Perbandingan Nilai Semantic Relatedness dengan k 24

Dari tabel 5 dapat dilihat hasil pengujian dengan ukuran teks yang seimbang memberikan hasil akurasi yang baik yang pada contoh di atas ditunjukkan oleh ukuran long dan short dengan akurasi masing – masing sebesar 75% dan 83.3%. Sedangkan, hasil yang kurang memuaskan ditunjukkan oleh kelompok campuran dengan akurasi yang berbeda untuk tiap k-window. Secara keseluruhan pengujian, perbedaan k-window tidak banyak berpengaruh pada nilai akhir semantic relatedness hanya saja mempengaruhi tingkat cakupan pasangan kata dan peningkatan nilai semantic relatedness antar kata. Dengan peningkatan cakupan pasangan kata, nilai pasangan non-shared juga mengalami peningkatan yang dapat memberikan peningkatan pada nilai semantic relatedness.

Tabel 5 : Ringkasan hasil semua pengujian text relatedness

Kelompok percobaan			Hasil			Akurasi
K	Gamma	Size	Tepat	Tidak	Total	
12	1	Long	9	3	12	75
24	1	Long	9	3	12	75
12	1	Short	10	2	12	83.3
24	1	Short	10	2	12	83.3
12	1	Mix	9	6	15	60
24	1	Mix	7	8	15	46.6

Dari tabel di atas dapat dilihat hasil pengujian dengan ukuran teks yang seimbang memberikan hasil akurasi yang baik yang pada contoh di atas ditunjukkan oleh ukuran long dan short dengan akurasi masing – masing sebesar 75% dan 83.3%. Sedangkan, hasil yang kurang memuaskan ditunjukkan oleh kelompok campuran dengan akurasi yang berbeda untuk tiap k-window. Secara keseluruhan pengujian, perbedaan k-window tidak banyak berpengaruh pada nilai akhir semantic relatedness hanya saja mempengaruhi tingkat cakupan pasangan kata dan peningkatan nilai semantic relatedness antar kata. Dengan peningkatan cakupan pasangan kata, nilai pasangan non-shared juga mengalami peningkatan yang dapat memberikan peningkatan pada nilai semantic relatedness.

Dari hasil analisis yang telah dilakukan ditemukan bahwa hasil yang diberikan oleh sistem cukup memuaskan walaupun pada kondisi tertentu memberikan hasil yang kurang bagus dan faktor – faktor seperti jumlah nilai pasangan non-shared, ukuran teks yang diuji, dan jumlah shared keyword berpengaruh terhadap hasil semantic relatedness yang diberikan serta nilai k dan gamma berpengaruh pada kualitas dari semantic profile yang dibangun.

V. Kesimpulan dan Saran

Berdasarkan hasil analisis terhadap hasil pengujian yang telah dilakukan pada penelitian ini, dapat disimpulkan bahwa:

1. Metode Salient Semantic Analysis dapat digunakan untuk melakukan perhitungan nilai semantic relatedness antar kata dan antar teks.

2. Metode Word Co-occurrence Keyword Extraction dapat digunakan sebagai preprocessing dari metode Salient Semantic Analysis.
3. Kualitas semantic profile dipengaruhi oleh beberapa faktor di antaranya besarnya nilai k dan gamma.
4. Semakin besar nilai k dan gamma serta semakin kecil perbedaan ukuran teks maka nilai semantic relatedness antar teks akan semakin bagus karena semakin banyak pasangan non-shared yang mempunyai nilai semantic relatedness.

Saran yang diperlukan dari penelitian ini untuk pengembangan sistem yang lebih lanjut adalah sebagai berikut:

1. Memperbanyak artikel Wikipedia yang digunakan yang mencakup banyak kategori sehingga pasangan kata yang dibentuk semakin banyak dan nilai semantic relatedness juga meningkat.
2. Menambahkan phrase pada term yang digunakan dalam perhitungan tanpa mengurangi akurasi dari sistem.
3. Menambahkan filtering dan normalisasi untuk meningkatkan kualitas dari semantic profile.

VI. Referensi

[1] A. G. Jivani, "A Comparative Study of Stemming Algorithms," 2011.

[2] E. Gabrilovich and S. Markovitch, "Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis," 2007.

[3] M. A. Islam and D. Inkpen, "Second order Co-occurrence PMI for Determining the Semantic Similarity of Words," 2006.

[4] M. F. Porter, "An Algorithm for Suffix Stripping," 1980.

[5] M. Strube and S. P. Ponzetto, "WikiRelate! Computing Semantic Relatedness Using Wikipedia," 2006.

[6] R. Agrawal and M. Brata, "A Detailed Study on Text Mining Techniques," International Journal of Soft Computing and Engineering (IJSCE), vol. 2, no. 6, 2013.

[7] R. Mihalcea and A. Csomai, "Wikify! Linking Documents to Encyclopedic Knowledge," 2007.

[8] S. Hassan and R. Mihalcea, "Semantic Relatedness Using Salient Semantic Analysis," 2011.

[9] W. A. Gale, K. W. Church and D. Yarowsky, "One Sense Per Discourse," 1992.

[10] Y. Matsuo and M. Ishizuka, "Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information," 2003.

[11] Z. Zhang, A. L. Gentile and F. Ciravegna, "Recent Advances in Methods of Lexical Semantic Relatedness - a survey," Natural Language Engineering, vol. 19, no. 4, pp. 411-479, 2013.