Analisis Efektifitas Pengukuran Keterkaitan Antar Teks Menggunakan Metode *Salient Semantic Analysis* Dengan *TextRank for Keyword Extraction* Sebagai *Preprocessing*

Bagus Widya Pratama S1
Teknik Informatika
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
widyabagus@gmail.com

Abstract--- Dengan meningkatnya kebutuhan masyarakat dalam mencari sebuah keterkaitan informasi berupa teks yang mereka baca dengan refrensi lain yang terkait seperti artikel berita dan sejenisnya, maka metode pengukuran *Semantic Relatedness* yang baik sangat dibutuhkan supaya hasil pencarian sebuah refrensi teks lain yang saling terkait dapat menghasilkan sebuah hasil yang tepat. Untuk itu pada tugas akhir kali ini akan dilakukan analisis terhadap efektifitas dari metode pengukuran *Semantic Relatedness* yaitu *Salient Semantic Analysis* dan menggunakan *TextRank for Keyword Extraction* sebagai *preprocessing* agar dapat diketahui bagaimana kerja metode tersebut dalam menentukan keterkaitan antar teks. *Salient Semantic Analysis* adalah metode untuk mengukur keterkaitan antar teks dengan memanfaatkan *Corpus* sebagai *Knowledge Based* yang mengambil sumber dari *Wikipedia*. Sedangkan *TextRank for Keyword Extraction* adalah metode untuk menentukan intisari/*keyword* dari sebuah kalimat.

Keywords : ***semantic relatedness, salient semantic analysis, teks, keyword, textrank for keyword extraction, preprocessing***

I. Pendahuluan

Dengan meningkatnya kebutuhan masyarakat dalam mencari keterkaitan informasi berupa teks yang mereka baca dengan refrensi teks lain yang ada seperti artikel berita dan sejenisnya, maka metode pengukuran *Semantic Relatedness* yang baik sangat dibutuhkan supaya hasil pencarian sebuah refrensi teks lain yang saling terkait dapat menghasilkan sebuah hasil yang tepat. Selama ini pencarian refrensi teks yang terkait kebanyakan memanfaatkan kesamaan kategori, padahal parameter kategori masih terlalu luas untuk menentukan keterkaitan antar teks. Misal sebuah teks/artikel berita dengan kategori teknologi, bisa saja artikel tersebut membahas tentang *smartphone, finger print,* dll. Perlu dilakukan pengukuran keterkaitan teks agar pencarian refrensi teks lain yang saling terkait bisa menghasilkan hasil yang tepat. Atas dasar itulah tugas akhir kali ini mengambil tema analisis efektifitas pengukuran keterkaitan antar teks menggunakan sebuah metode *Salient Semantic Analysis.* Sebelum melakukan pengukuran *Semantic Relatedness* tahap awal adalah melakukan *preprocessing* dengan *TextRank for Keyword Extraction* untuk mendapatkan *keyword* dari sebuah teks. Setelah dilakukan *preprocessing* tahap berikutnya adalah menjalankan metode *Salient Semantic Analysis.* Dalam metode *Salient Semantic Analysis* pengukuan *Semantic Relatedness* dilakukan dengan membangun *Semantic Profile* dari *Salient Encyclopedic Features.* Secara umum metode *Salient Semantic Analysis* terdiri dari 2 langkah. Langkah pertama kita memanfaatkan sebuah corpus sebagai *Knowledge Based* yang bersumber dari *Wikipedia*. Langkah kedua adalah mulai mengukur *Semantic Relatedness* dari sebuah kata/teks. Diharapkan dengan dilakukannya tugas akhir ini maka dapat diketahui performa *Salient Semantic Analysis* dalam mengukur nilai *Semantic Relatedness* dalam sebuah kata.

II. Teori Terkait

Semantic Relatedness adalah sebuah metode untuk menentukan nilai keterkaitan diantara teks. Contoh ketika kita ingin menentukan keterkaitan kata antara *car* dan *automobile*, untuk membuat sebuah dugaan apakah ada keterkaitan antara kata tersebut kita menggunakan pengetahuan ataupun kemampuan berfikir secara konseptual. Corpus adalah kumpulan Teks dengan jumlah besar dan ter-struktur. Corpus biasanya digunakan untuk melakukan analisis statistik ataupun testing hipotesis. *Salient Semantic Analysis* adalah sebuah metode yang digunakan untuk mengukur *Semantic Relatedness* dengan membandingkan *Semantic Profile* yang dibangun dari *Salient Encyclopedic Features. TextRank for Keyword Extraction* adalah metode yang digunakan untuk menentukan sebuah *Keyword Extraction*. Hasil yang ingin didapat dari *TextRank for Keyword Extraction* adalah kumpulan kata/frase yang mewakili keseluruhan isi teks.

III. Metode

Dalam penelitian ini ada beberapa metode yang digunakan seperti saat preprocessing menggunakan *TextRank for Keyword Extraction,* dan proses utama menggunakan *Salient Semantic Analysis,* dst. Pada bagian ini akan diterangkan metode-metode tersebut, penjelasannya adalah sebagai berikut :

A. *TextRank for Keyword Extraction*

TextRank for Keyword Extraction adalah metode yang digunakan untuk menentukan sebuah *Keyword Extraction.* Hasil yang ingin didapat dari

TextRank for Keyword Extraction adalah kumpulan kata/frase yang mewakili keseluruhan isi teks. Dalam hal ini kata-kata yang menjadi kandidat sebagai *keyword* dalam sebuah dokumen akan diurutkan berdasarkan beberapa proses penilaian. Dalam *TextRank for Keyword Extraction* kumpulan *lexical* akan di *extract* dari sebuah teks dan kumpulan *lexical* tersebut akan dijadikan simpul dalam sebuah graph. Dan semua keterkaitan antara *lexical* tersebut akan saling dihubungkan.

Adapun urutan proses *TextRank for Keyword Extraction* adalah sebagai berikut :

a. Langkah pertama adalah melakukan proses *tokenisasi* terhadap teks.
b. Langkah kedua adalah melakukan Stopword(Syntatic Filter) terhadap teks yang sudah ditokenisasi agar hanya terpilih kata-kata penting saja.
c. Langkah ketiga adalah melakukan pembentukan graph dengan kata-kata yang lolos Syntatic Filter(Stopword) sebagai edges.
d. Langkah keempat adalah melakukan perhitungan untuk mendapatkan nilai pada tiap kata yang ada. Dengan menggunakan rumus :

$$S(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j)$$

Keterangan :

$S(V_i)$ = Nilai edges(kata) ke-i
d = Nilai yang di set antara 0 sampai 1
$In(V_i)$ = Predecessors dari edges V_i
$Out(V_j)$ = Successors dari edges V_j

B. *Salient Semantic Analysis*

Salient Semantic Analysis adalah sebuah metode yang digunakan untuk mengukur *Semantic Relatedness* dengan membandingkan *Semantic Profile* yang dibangun dari *Salient Encyclopedic Features.* Metode ini dibangun atas gagasan bahwa arti dari sebuah kata bisa di bangun karakteristiknya dengan *Salient Concept* yang ditemukan langsung dalam konteks itu. *Semantic Profile* ditentukan melalui *Profile Based* dalam *Wikipedia Corpus* dengan memanfaatkan *link* yang menghubungkan satu konsep dengan konsep lain yang sama.

Metode *Salient Semantic Analysis* secara umum terdiri dari 2 langkah. Yang pertama adalah membangun sebuah *corpus* yang bersumber dari *wikipedia.* Setelah itu membangun *concept-based word profiles* untuk mengukur *semantic relatedness* antar teks/kalimat.

a. Word Relatedness

Word Relatedness merupakan sebuah proses mencari nilai keterkaitan antar pasangan kata. Dalam hal ini *Semantic Profile* kata dibentuk dengan memanfaatkan *concept* yang paling terkait, dimana *concept* tersebut terdapat dalam artikel *Wikipedia.*

Dalam pembangkitan *World Relatedness* ini, kami memanfaatkan *Corpus c* dengan jumlah token *m.* Dimana jumlah *vocab* adalan N dan jumlah *concept* adalah W. Dan kami membangkitkan nilai keterkaitan pasangan kata dari seluruh kombinasi N x W. Adapun langkah-langkah nya sebagai berikut :

a. Langkah pertama dengan membangkitkan matriks E. Dalam tahap ini dilakukan pembangkitan matriks E dari *Corpus* yang ada. Dimana matriks E dibangun dengan mendeteksi jumlah kemunculan *term* ω(term yang ada di corpus) dengan *concept* c(link yang ada di corpus) secara bersamaan sepanjang *k* didalam *Corpus*. Adapun rumus untuk membangkitkan Matriks E adalah :

$$E_{ij} = f(\omega_i, c_j)$$

Keterangan :

E_{ij} = Matriks E
$f(\omega_i, c_j)$ = Frekuensi kemunculan *term* ω dan *concept* c secara bersamaan dalam token *k.*

b. Langkah kedua adalah dengan membangkitkan Matriks P. Matriks P dibangkitkan dari *Corpus* yang ada dan Matriks E yang sudah terbentuk. Adapun rumus yang digunakan adala sebagai berikut :

$$P_{ij} = \frac{f(\omega_i, c_j) \times m}{f(\omega_i) \times f(c_j)}$$

Keterangan :

P_{ij} = Matriks P
$f(\omega_i, c_j)$ = Frekuensi kemunculan *term* ω dan *concept* c secara bersamaan sepanjang token k.
m = Jumlah token dalam *Corpus*
$f(\omega_i)$ = Jumlah frekuensi *term* ω dalam *Corpus*
$f(c_j)$ = Jumlan frekuensi *concept c* dalam *Corpus*

c. Langkah berikutnya adalah dengan menghitung nilai keterkaitan antar kata dengan memasangkan kombinasi seluruh vocab yang ada dengan seluruh link yang sudah dibangkitkan. Dan seluruh hasil perhitungan tersebut akan disimpan dalam sebuah kamus kata. Adapun rumus yang digunakan adalah sebagai berikut :

$$= \frac{\sum_{y=1}^{N} S_{rel}(A,B) (P_{iy} * P_{jy})^{\gamma}}{\sqrt{\sum_{y=1}^{N} P_{iy}^{2\gamma}} * \sqrt{\sum_{y=1}^{N} P_{jy}^{2\gamma}}}$$

Keterangan :

$S_{rel}(A,B)$ = Nilai keterkaitan antar kata A dan B

P = Matriks P(term i dan concept y)

γ = Nilai parameter

C. Corpus

Corpus adalah kumpulan Teks dengan jumlah besar dan ter-struktur. Corpus biasanya digunakan untuk melakukan analisis statistik ataupun testing hipotesis.

Terdapat 3 langkah dalam pembentukan Corpus yang bersumber dari *Wikipedia.* Langkah-langkah tersebut antara lain :

a. Menggunakan link-link yang tersedia di *Wikipedia* dimana link tersebut bersumber dari para pengguna *Wikipedia*.
b. Langkah kedua adalah dengan menggunakan konsep *one sense per discourse heuristic.* Dimana jika ada kata yang bukan merupakan link tetapi sama dengan link yang ada, maka kata tersebut dianggap link.
c. Langkah terakhir adalah dengan menggunakan menghitung nilai *disambiguation* yaitu dengan cara membagi jumlah kata yang berupa link dengan jumlah seluruh kata baik link atau bukan. Jika nilai *disambiguation* lebih besar atau sama dengan 0.5 maka kata tersebut adalah link.

D. Wikipedia

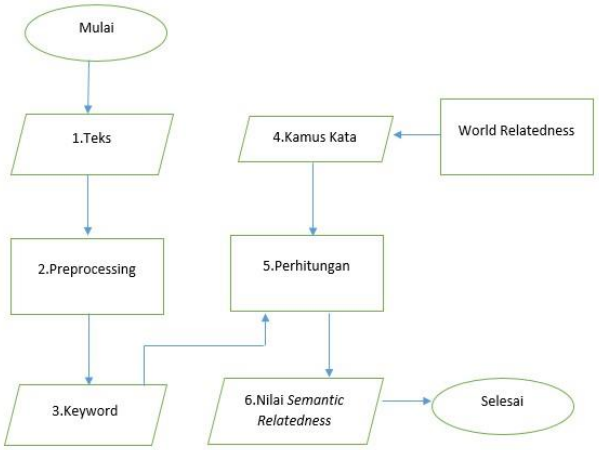
Wikipedia memberikan sebuah pengetahuan untuk perhitungan keterkaitan kata dalam konsep yang lebih terstruktur dari pada beberapa *search engine.* Jika dibandingkan dengan beberapa acuan dataset seperti *WordNet*, *Wikipedia* memiliki *performance* lebih baik terutama ketika mengaplikasikan sebuah dataset yang besar. Kategori ruang pencarian pada *Wikipedia* berupa graph yang memiliki kedalaman yang sangat besar pada *term*, faktor percabangan, dan turunan keterkaitan yang sangat luas. Dan relasi kategori pada *Wikipedia* tidak dapat diinterpretasikan sebagai hubungan *Taxonomy* sejak *Wikipedia* menunjukan hubungan *meronymic*.

IV. Perancangan Sistem

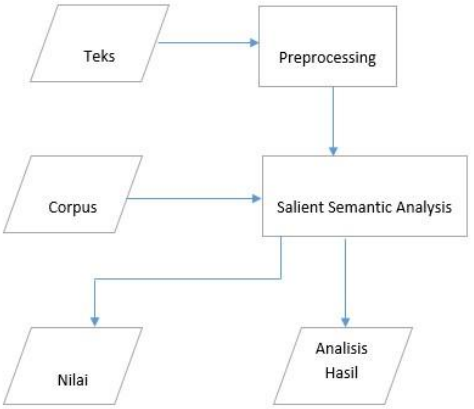
Pada penelitian dilakukan analisis perhitungan *Semantic Realtedness* pada sebuah teks menggunakan metode *Salient Semantic Analysis.* Dengan *TextRank for Keyword Extraction* sebagai *preprocessing* untuk mencari *keyword* dari sebuah teks. Dan memanfaatkan sebuah *Corpus* sebagai *Knowledge Based* dengan mengambil sumber dari *Wikipedia.* Input dari proses pada sistem ini adalah sebuah teks yang telah dilakukan *preprocessing* untuk menentukan *keyword* pada teks tersebut. Dan output dari proses ini adalah sebuah nilai *Semantic Relatedness* antar teks yang dibandingkan dan hasil analisa terhadap efektifitas dari metode yang digunakan. Proses

A. Gambaran Umum Sistem

awal dalam sistem ini adalah *preprocessing* dengan menggunakan *TextRank for Keyword Extraction* untuk mencari *keyword* dari teks. Setelah itu masuk ke proses penghitungan *Semantic Relatedness* menggunakan metode *Salient Semantic Analysis* dengan langkah pengerjaan yang pertama memanfaatkan *Corpus* sebagai *Knowledge Based* yang mengambil sumber dari *Wikipedia*. Setelah itu mulai melakukan perhitungan *semantic relatedness.* Gambaran umum sistem jika dituliskan dalam *flowchart* adalah sebagai berikut :

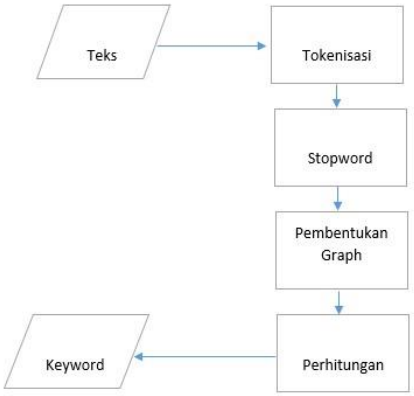


B. Deskripsi Proses Secara Umum



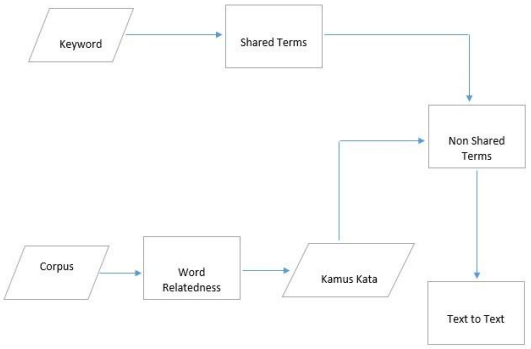
- **Teks**
  Dalam hal ini data teks yang dijadikan percobaan adalah sebuah teks artikel berbahasa Inggris.
- **Preprocessing**
  Pada tahap ini dilakukan proses preprocessing pada teks yang akan digunakan sebagai data uji dengan menggunakan metode **TextRank for Keyword Extraction** hingga menghasilkan sebuah keyword. Berikut contoh input dan hasil output dari proses ini.
- **Corpus**
  Untuk menghitung keterkaitan antar teks dengan metode Salient Semantic Analysis, dalam hal ini menggunakan Corpus untuk membangkitkan *Semantic Profile* dengan memanfaatkan database wikipedia
- **Salient Semantic Analysis**
  Pada tahap ini sudah dimulai menghitung keterkaitan antar teks dengan metode *Salient Semantic Analysis.* Dengan memanfaatkan wikipedia sebagai corpus dan *keyword* dari teks yang sudah dilakukan proses *preprocessing* sebagai data uji.
- **Nilai**
  Nilai disini adalah output dari sebuah proses, yaitu nilai keterkaitan antar teks yang diuji.
- **Analisis Hasil**
  Dalam tahap ini adalah menganalisa hasil dari proses tersebut. Yaitu hasil pengukuran dari metode yang digunakan.

C. Deskripsi Proses Preprocessing dengan TextRank for Keyword Extraction



- **Teks**
  Sebuah teks/artikel ber-bahasa Inggris sebagai data uji
- **Tokenisasi**
  Pada tahap ini, dilakukan tokenisasi terhadap teks yang ada. Tokenisasi adalah proses memisahkan tiap kata yang terkandung di dalam teks dengan acuan adalah spasi(" ").
- **Stopword**
  Pada tahap ini dilakukan Stopword(Syntatic Filter) terhadap teks yang sudah ditokenisasi agar hanya terpilih kata-kata penting saja.
- **Pembentukan Graph**
  Pada tahap ini, dilakukan pembentukan graph dengan kata-kata yang lolos Syntatic Filter(Stopword) sebagai edges. Dan setiap edges akan saling terhubung satu sama lain dengan *vertex*. Dimana keterhubungan tiap edges ditentukan dengan kemunculan secara bersama didalam teks sepanjang windows N.
- **Perhitungan**
  Pada tahap ini dilakukan perhitungan untuk mendapatkan nilai pada tiap kata yang ada.
- **Keyword**
  Pada tahap ini didapatkan kumpulan keyword terpilih dari hasil pemilihan keyword dengan nilai tertinggi.

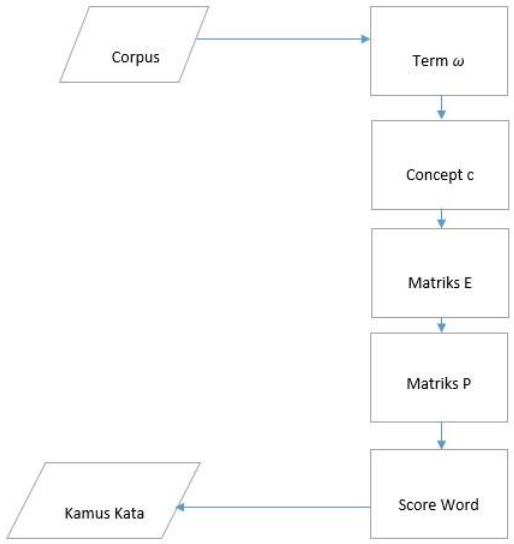
D. Deskripsi Proses Perhitungan Semantic Relatedness Denga Metode Salient Semantic Analysis



- **Keyword**
  Keyword terpilih saat proses *preprocessing.*
- **Shared Term**
  Pada tahap ini dilakukan perhitungan terhadap jumlah term/keyword yang sama dari 2 teks yang akan dibandingkan. Jadi, jika ada keyword yang sama antar teks yang dibandingkan maka jumlah keyword yang sama tersebut dijumlahkan untuk mendapatkan nilai *shared term.*
- **Corpus**
  Pada tahap ini akan dilakukan pembangkitan *Semantic Profile* dengan memanfaatkan *Corpus* yang bersumber dari *Wikipedia.*
- **Word Relatedness**
  Pada proses ini dilakukan proses untuk membangkitkan *Semantic Profile* dari sumber *Corpus* yang bersumber dari *Wikipedia.*
- **Kamus Kata**
  Pada tahap ini diperoleh kamus kata yang berasal dari proses *word relatedness*, kamus kata berisi nilai antar pasangan kata.

- **Non Shared Term**
  Pada tahap ini dilakukan perhitungan antar kombinasi pasangan kata yang berbeda dari 2 teks yang akan dibandingkan. Dimana nilai keterkaitan antar 2 kata tersebut didapat dari kamus kata yang dibangkitkan dalam proses *word relatedness*.
- **Text to Text**
  Pada tahap ini dilakukan perhitungan keterkaitan antar 2 teks yang akan dibandingkan.

E. Deskripsi Proses Word Relatedness



- **Corpus**
  Corpus yang digunakan diambil dari *wikipedia* dengan m token.
- **Term ω**
  Dalam tahap ini dilakukan proses pembentukan *term* yang berupa *vocab* yang diambil dari *Corpus* yang ada. Dimana *Corpus* tersebut mengambil sumber dari *Wikipedia*.
- **Concept c**
  Dalam tahap ini dilakukan proses pembentukan concept yang berupa sebuah *link* yang diambil dari *Corpus* yang ada. Dimana *Corpus* tersebut mengambil sumber dari *Wikipedia*.
- **Matriks E**
  Dalam tahap ini dilakukan pembangkitan matriks E dari *Corpus* yang ada. Dimana matriks E dibangun dengan mendeteksi jumlah kemunculan *term* (term yang ada di corpus) dengan *concept* c(link yang ada di corpus) secara bersamaan sepanjang *k* didalam *Corpus*.
- **Matriks P**
  Dalam tahap ini dilakukan pembangkitan matrix P yaitu dengan memasangkan seluruh kombinasi *term* (term yang ada di corpus) dengan *concept* c(link yang ada di corpus) dari *Corpus* yang ada dan memanfaatkan Matriks E yang sudah terbentuk.
- **Score Word**
  Dalam tahap ini dilakukan perhitungan nilai keterkaitan antar pasangan kata(vocab) yang ada di dalam *Corpus.* Dan seluruh hasil perhitungan tersebut akan disimpan dalam sebuah kamus kata.
- **Kamus Kata**
  Dalam tahap ini, seluruh pasangan kata yang sudah dihitung nilai keterkaitannya disimpan dalam sebuah kamus kata.

V. Hasil Pengujian

Dalam penelitian kali ini dilakukan 3 skenario pengujian, adapun skenario pengujian dan hasil sebagai berikut :

A. Pengujian Pengaruh Nilai Windows k
Tabel Hasil Pengujian :

Tabel 4.1 hasil pengujian pengaruh windows k dengan γ = 4

No	Nilai γ	Nilai windows k	Nilai Korelasi
1.	4	20	0,359156564
2.	4	30	0,299635837

Tabel 4.2 hasil pengujian pengaruh windowz k dengan γ = 2(1)

No	Nilai γ	Nilai windows k	Nilai Korelasi
1.	2	20	0,369254781
2.	2	30	0,306404185

Tabel 4.3 hasil pengujian pengaruh windowz k dengan γ = 2(2)

No	Nilai γ	Nilai windows k	Nilai Korelasi
1.	2	30	0,306404185
2.	2	40	0,295594247

Tabel 4.4 hasil pengujian pengaruh windowz k dengan γ = 2(2)

No	Nilai γ	Nilai windows k	Nilai Korelasi
1.	2	20	0,369254781
2.	2	30	0,306404185
3	2	40	0,295594247

B. Pengujian Pengaruh Nilai Gama()
Tabel Hasil Pengujian :

Tabel 4.5 hasil pengujian pengaruh nilai gama(γ) dengan k = 20

No	Nilai windows k	Nilai γ	Nilai Korelasi
1.	20	2	0,369254781
2.	20	4	0,359156564

Tabel 4.6 hasil pengujian pengaruh nilai gama(γ) dengan k = 30

No	Nilai windows k	Nilai γ	Nilai Korelasi
1.	30	2	0,306404185
2.	30	4	0,299635837

Tabel 4.7 hasil pengujian pengaruh nilai gama(γ) dengan k = 30

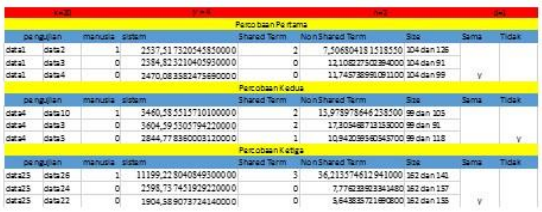
No	Nilai windows k	Nilai γ	Nilai Korelasi
1.	30	4	0,299635837
2.	30	6	0,295484973

Tabel 4.8 hasil pengujian pengaruh nilai gama(γ) dengan k = 30

No	Nilai windows k	Nilai γ	Nilai Korelasi
1.	30	2	0,306404185
2.	30	4	0,299635837
3.	30	6	0,295484973

C. Pengujian Hasil Perhitungan Nilai Keterkaitan antar Teks

Tabel Hasil Pengujian :



Percobaan Pertama								
pengujian		manusia	sistem	Shared Term	Non Shared Term	Size	Sama	Tidak
data1	data2	1	2537,51732054585000000	2	7,506804181518550	104 dan 126		
data1	data3	0	2384,82321040593000000	0	12,10822750284000	104 dan 91		
data1	data4	0	2470,08338824736900000	0	11,74573899091100	104 dan 99	v	
Percobaan Kedua								
pengujian		manusia	sistem	Shared Term	Non Shared Term	Size	Sama	Tidak
data4	data10	1	3460,58551571010000000	2	13,97897864623850	99 dan 103		
data4	data3	0	3604,59130579422000000	2	17,80394871813530	99 dan 91		
data4	data5	0	2844,77839000831200000	1	10,94203936043700	99 dan 118		v
Percobaan Ketiga								
pengujian		manusia	sistem	Shared Term	Non Shared Term	Size	Sama	Tidak
data23	data26	1	11199,22 80408493000000	3	36,21357461294100	162 dan 141		
data23	data24	0	2598,73 74519292100000	0	7,77623285334480	162 dan 157		
data23	data22	0	1904,58 90737241400000	0	5,64383572189800	162 dan 135	v	

VI. Dsikusi

A. Pengaruh Nilai Windows k :

- Tidak ada hubungan langsung antara besar windows k dengan besar nilai korelasi.
- Besar Nilai windows k berpengaruh pada besar matriks E, Matriks P dan nilai keterkaitan antar kata yang terbentuk.

B. Pengaruh Nilai Gama(□) :

- Besar nilai gama(□) tidak terlalu berpengaruh pada nilai keterkaitan antar kata maupun nilai korelasi, karena nilai gama(□) digunakan untuk melakukan normalisasi agar nilai keterkaitan antar kata yang terbentuk tidak terlalu bias bukan untuk merubah nilai secara drastis**.**

C. Perhitungan Keterkaitan Antar Kata :

- Size(ukuran) pada teks berpengaruh pada besar nilai keterkaitan yang terbentuk.
- Jumlah kamus kata yang dibangkitkan berpengaruh pada hasil nilai keterkaitan yang dihitung.
- metode pengukuran keterkaitan antar teks menggunakan *Salient Semantic Analysis* dengan *TextRank for Keyword Exctraction* sebagai *preprocessing* tepat dan efektif untuk digunakan.

VII. Kesimpulan dan Saran

A. Kesimpulan

Berdasarkan analisis terhadap hasil pengujian yang sudah dilakukan, maka dapat ditarik kesimpulan sebagai berikut :

1. Metode pengukuran keterkaitan antar teks menggunakan *Salient Semantic Analysis* dengan *TextRank for Keyword Exctraction* sebagai preprocessing tepat dan efektif untuk digunakan.
2. Semakin besar windows k, maka peluang kata untuk bisa ter-*cover* dalam pembentukan kamus kata semakin besar.
3. Besar nilai gama(□) tidak terlalu berpengaruh pada besar nilai keterkaitan antar kata yang terbentuk.
4. Jumlah artikel yang digunakan saat proses pembangunan *Corpus* berpengaruh pada hasil nilai keterkaitan antar teks yang dilakukan.

B. Saran

Adapun saran yang diperlukan pada tugas akhir ini :

1. Dalam melakukan pengujian, data yang digunakan harus sesuai dengan tema artikel yang digunakan untuk membangkitkan *Corpus* agar pasangan kata bisa ter-cover dengan baik.
2. Diperlukan sistem terdistribusi dalam hal pengecekan pasangan nilai antar kata dalam kamus agar proses perhitungan bisa berjalan lebih cepat.

VIII. Referensi

[1] H. Samer dan M. Rada, "Semantic Relatedness Using Salient Semantic Analysis," 2011.

[2] A. I. Md dan I. Diana, "Second Order Co-occurance PMI or Determining the Semantic Similarity of Words," 2006.

[3] T. George, V. Iraklis dan V. Michalis, "Text Relatedness Based om a Word Thesaurus," 2010.

[4] M. Rada, C. Courtney dan S. Carlo, "Corpus-based and Knowledge-based Measures of Text Semantic Similarity," 2006.

[5] M. Rada dan T. Paul, "TextRank : Bringing Order into Texts," 2004.

[6] A. Rashmi dan B. Mridula, "A Detailed Study on Text Mining Techniques," 2013.

[7] J. M. Raymond dan Y. N. Un, "Text Mining With Information Extraction," 2002.

[8] R. Milos dan I. Mirjana, "Text Mining Approaches and Applications," 2008.

[9] S. Michael dan P. P. Simone, "Wikielate! Computing Semantic Relatedness Using Wikipedia," 2006.

[10] A. G. William, W. C. Kenneth dan Y. David, "One Sense Per Discourse," 1992.