PENGUKURAN KEMIRIPAN DOKUMEN DENGAN MENGGUNAKAN *TOOLS GENSIM*

Kemal Ade Sekarwati[1*], Lintang Yuniar Banowosari[2], I Made Wiryana[2], Djati Kerami[1]
[1] Jurusan Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi,
Universitas Gunadarma
Jl. Margonda Raya 100, Pondok Cina, Depok 16424.
[2] Jurusan Manajemen Informatika, Direktorat Diploma Tiga Teknologi Informasi,
Universitas Gunadarma
Jl. Margonda Raya 100, Pondok Cina, Depok 16424.
[*]Email: ade@staff.gunadarma.ac.id

Abstrak

Gensim merupakan open-source model ruang vektor dan toolkit topic modeling, yang diimplementasikan dalam bahasa pemrograman Python. Untuk kinerja Gensim, digunakan NumPy, SciPy dan Cython (opsional). Gensim secara khusus ditujukan untuk menangani koleksi teks besar dengan menggunakan algoritma secara online. Gensim mengimplementasikan tf-idf, latent semantic analysis (LSA), Latent Dirichlet Analysis (LDA), dan lain-lain. Pada penelitian ini digunakan metode LSA yang terdapat pada Gensim dan beberapa rumus perhitungan kemiripan untuk mengukur kemiripan dokumen. Pengukuran kemiripan dokumen menggunakan rumus Cosine, Dice, dan Jaccard. Hasil perhitungan kemiripan berupa prosentase kemiripan. Dokumen yang digunakan adalah dokumen abstrak penulisan ilmiah berbahasa Indonesia. Pengujian dilakukan terhadap 30 pasang dokumen yang sama, 30 dokumen yang berbeda, 5 dokumen similar, dan 5 dokumen transposisi dua dan tiga kalimat. Hasil pengujian menunjukkan bahwa untuk dokumen yang sama terdapat kemiripan 100%, untuk dokumen yang berbeda dihasilkan prosentase kemiripan yang berbeda-beda, untuk pengujian dokumen similar menghasilkan kemiripan yang mendekati 100%, sedangkan untuk dokumen transposisi menunjukkan prosentase meningkat untuk transposisi dari dua kalimat ke tiga kalimat.

***Kata kunci**: gensim, lsa, pengukuran kemiripan*

1. PENDAHULUAN

Pendekatan pengukuran kemiripan dokumen dikategorikan menjadi tiga kategori (Stein, B., and Eissen, S., 2006) yaitu *string matching*, *keyword similarity* dan *fingerprint*. Pendekatan lain yang dapat digunakan untuk pengukuran kemiripan selain berbasiskan *string* yaitu pendekatan berbasiskan *corpus* dan pendekatan berbasiskan *knowledge* dan *corpus* (Gooma, Wael H., and Aly A. Fahmy, 2013). Salah satu pendekatan *corpus* yang merupakan teknik yang paling popular adalah pendekatan pengukuran dengan menggunakan teknik *Latent Semantic Analysis* (LSA). LSA merupakan teori dan metode untuk mengekstraksi dan merepresentasikan penggunaan makna kata yang kontekstual dengan perhitungan statistik yang diterapkan pada *corpus* teks yang besar (Landauer, T. K., Foltz, P. W., and Laham, D, 1998). Metode LSA menerima masukan berupa dokumen kemudian melakukan proses perbandingan terhadap kata-kata unik dari masukkan dokumen tersebut yang direpresentasikan dalam bentuk matriks. Nilai dari matriks merupakan banyaknya kemunculan sebuah kata pada setiap dokumen yang dibandingkan. Perhitungan kemiripan dokumen diambil dari nilai matriks yang dihasilkan.

Penelitian yang menggunakan *Latent Semantic Analysis* (LSA) di bidang pendidikan yaitu deteksi kemiripan dokumen berbahasa Inggris dengan menggunakan ruang semantik dilakukan oleh Khalid Shams [Shams, 2010]. Deteksi kemiripan yang dilakukan menggunakan kerangka sinonim dan antonim untuk memeriksa kemiripan teks yang berhubungan dengan kemiripan isi dokumen asli dengan dokumen yang diperiksa.

Penelitian LSA untuk penilaian esai secara otomatis dilakukan oleh Heninggar Saptiantri pada tahun 2009 [Heninggar Saptiantri, 2009]. Penelitian ini membandingkan LSA dan *Vector Space Model* (VSM) untuk menilai jawaban berbentuk esai, serta meneliti pengaruh pemotongan imbuhan dan perluasan kata kunci terhadap efektifias sistem. Penilaian esai otomatis yang dilakukan berdasarkan pencocokan kata kunci yaitu mencocokan kata-kata yang muncul pada kunci jawaban sebagai vektor kueri dan kumpulan jawaban siswa sebagai vektor dukumen.

ISBN 978-602-99334-4-4

Kedekatan antara dua vektor dihitung dengan *cosine similarity*. Proses *stemming* pada penilaian esai secara otomatis dilakukan di luar sistem untuk memproses dokumen.

Penelitian lain yang menggunakan LSA untuk pemodelan bahasa *cross-lingual* [Kim, Woosung., and Sanjeev Khudanpur., 2004] yang tidak membutuhkan *corpus* kalimat. LSA dari kumpulan teks dokumen dua bahasa menyediakan representasi kata dalam kedua bahasa dengan *Euclidean space* berdimensi rendah. Dengan kumpulan teks ini berarti menggunakan sumber bahasa yang kaya untuk meningkatkan pemodelan bahasa pada sumber bahasa yang tidak sempurna.

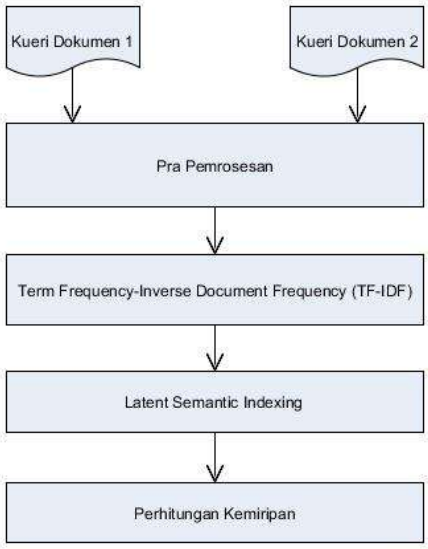
Josef Steinberger dan Karel Ježek [Steinberger, Josef., and Karel Ježek, 2004] menggunakan LSA untuk *text summarization*. Metode *text summarization* secara umum menggunakan teknik LSA untuk mengidentifikasi kalimat-kalimat penting secara semantik. Metode evaluasi yang digunakan untuk mengukur kemiripan isi antara dokumen asli dan ringkasannya. Pada bagian evaluasi, dibandingkan tujuh metode ringkasan dengan *content-based evaluator* klasik dan dengan dua *evaluator* LSA. Dokumen yang digunakan adalah kumpulan berita Reuters. Hasil eksperimen menunjukkan bahwa LSA masih perlu dikembangkan untuk proses lematisasi bahasa Inggris dan Ceko.

Dari penelitian-penelitian sebelumnya, belum terdapat penelitian tentang pengukuran kemiripan dokumen ilmiah berbahasa Indonesia dengan metode LSA dan menggunakan *tools* Gensim. Penelitian ini difokuskan pada pengukuran kemiripan dokumen berbahasa Indonesia dengan menggunakan metode LSA. Sistem yang dikembangkan adalah untuk mengukur kemiripan antar dua dokumen yang diinput ke dalam sistem. Dokumen yang dibandingkan berupa file teks yang terdiri dari huruf dan angka, tidak mencakup gambar, tabel, dan rumus. Proses perbandingan dilakukan dengan menggunakan *term frequency-inverse document frequency* (*tf-idf*) yaitu menghitung bobot dari sebuah frekuensi kemunculan kata pada dokumen. Hasil perhitungan tf-idf berupa nilai matrik. Nilai dari matriks merupakan banyaknya kemunculan sebuah kata pada setiap dokumen yang dibandingkan. Perhitungan kemiripan dokumen diambil dari nilai matriks yang dihasilkan.

Tujuan dari penelitian ini adalah membangun sistem pengukuran kemiripan dokumen berbahasa Indonesia dengan menggunakan metode *Latent Semantic Analysis* (LSA) sedangkan *tools* yang digunakan untuk penelitian ini adalah Gensim. Gensim merupakan *open-source* model ruang vektor dan *toolkit topic modeling*, yang diimplementasikan dalam bahasa pemrograman Python. Untuk kinerja Gensim, digunakan NumPy, SciPy dan Cython (opsional).

2. METODE PENELIATIAN

Tahapan pengukuran kemiripan dokumen dapat dilihat pada gambar 1 berikut ini.



Gambar 1. Arsitektur Deteksi Kemiripan Dokumen

Berikut ini penjelasan tentang arsitektur deteksi kemiripan dokumen.

1. Kueri Dokumen
   Kueri dokumen ini bertujuan untuk mengambil dokumen yang akan diperiksa, apakah dokumen tersebut mempunyai kemiripan dengan dokumen lain. Kedua dokumen berada pada media penyimpanan lokal.
2. Pra Pemrosesan.
   Pada tahapan ini yang dilakukan adalah melakukan penghapusan simbol-simbol, tanda baca selain titik, kata sambung dan preposisi yang terdapat pada isi dokumen yang sedang dideteksi.
3. *Term Frequency-Inverse Document Frequency* (TF-IDF)
   Jika suatu kata atau *term* muncul di dokumen, maka nilai vektornya adalah bukan-nol (*non-zero*). Beberapa cara berbeda untuk menghitung nilai tersebut, dikenal sebagai bobot *term* (*term weight*). Salah satu formula yang terkenal yaitu pembobotan *term frequency-inverse document frequency* (tf-idf).
4. *Latent Semantic Indexing* (LSI)
   Pada tahapan ini dilakukan proses membuat ruang semantik. Ruang semantik adalah sebuah matriks kata-dokumen dibuat dengan menggunakan SVD untuk mengurangi dimensionalitas matriks kata-dokumen asli yang dibuat dari *corpus*. SVD mengurai asli matriks X, ke dalam produk tiga matriks baru, W, S, dan P. S adalah matriks diagonal yang berisi nilai-nilai singular. Nilai singular X adalah akar kuadrat *eigenvalues* X^TX yang disusun dalam urutan yang ukurannya menurun (Lay, D. C., 1996).
5. Perhitungan kemiripan.
   Pada tahapan ini dilakukan proses membandingkan kalimat pada dokumen yang diperiksa dengan kalimat yang terdapat pada dokumen lain. Perbandingan dilakukan terhadap kata-kata yang terdapat pada kalimat dalam dokumen 1 atau yang diperiksa dibandingkan dengan kata-kata yang terdapat pada kalimat dalam dokumen 2. Hasil dari perhitungan kemiripan berupa nilai kemiripan dokumen. Pada peneltian ini dilakukan uji pengukuran kemiripan dokumen dengan menggunakan rumus *Cosine Similarity*, *Dice's Similarity*, dan *Jaccard Similarity*. Ketiga pengukuran ini merupakan pengukuran yang terbaik dari beberapa pengukuran yang ada (Zhang, J., Yunchuan Sun, Huilin Wang, and Yanqing He., 2011).
   Berikut ini variabel-variabel yang digunakan dalam perhitungan kemiripan dokumen dengan *Dice's Similarity* dan *Jaccard Similarity* [Zhang, J., et al, 2011] :
   s_a = kalimat dengan panjang m (m ≥ 2)
   s_b = kalimat dengan panjang n (n ≥ 2)

$$s_a = w_{a1}w_{a2}w_{a3} \dots w_{am}((m \geq 2) \quad \text{.....} \quad [2.1]$$

$$s_b = w_{b1}w_{b2}w_{b3} \dots w_{bn}((n \geq 2) \quad \text{.....} \quad [2.2]$$

   Keterangan :
   $w_{ai}(i \in [1,m])$ dan $w_{bj}(j \in [1,n])$ = kata atau pemisah pada s_a dan s_b.
   $w(s_a)$ = kumpulan kata yang berisikan seluruh kata $w_{ai}(i \in [1,m])$.
   $w(s_b)$ = kumpulan kata yang berisikan seluruh kata $w_{bj}(j \in [1,n])$.

   1. Dice's Similarity

$$Dice(s_a, s_b) = \frac{2|w(s_a) \cap w(s_b)|}{|w(s_a)| + |w(s_b)|} \quad \text{.....} \quad [2.3]$$

      Keterangan rumus :
      s_a = kalimat dengan panjang m (m ≥ 2)
      s_b = kalimat dengan panjang n (n ≥ 2)
      $w(s_a)$ = kumpulan kata yang berisikan seluruh kata $w_{ai}(i \in [1,m])$
      $w(s_b)$ = kumpulan kata yang berisikan seluruh kata $w_{bi}(j \in [1,n])$

   2. *Jaccard Similarity*

ISBN 978-602-99334-4-4

$$Jaccard\ (s_a, s_b) = \frac{|w(s_a) \cap w(s_b)|}{|w(s_a) \cup w(s_b)|} \quad \text{..... [2.4]}$$

Keterangan rumus :

s_a = kalimat dengan panjang m (m ≥ 2)

s_b = kalimat dengan panjang n (n ≥ 2)

$w(s_a)$ = kumpulan kata yang berisikan seluruh kata $w_{ai} (i \in [1, m])$

$w(s_b)$ = kumpulan kata yang berisikan seluruh kata $w_{bi} (j \in [1, n])$

3. Cosine Similarity

Untuk menghitung kemiripan kalimat berbasiskan vektor kata, vektor kata dari kalimat dibangun terlebih dahulu. Jika kata pada $w(s_a)$ dan $w(s_b)$ sebagai bobot, s_a dan s_b dapat direpresentasikan sebagai *bags of words*. Vektor dari dua kalimat sebagai berikut :

$$v(s_a) = \{(w_1, w_{a1}), (w_2, w_{a2}), \dots, (w_{i+j}, w_{a(i+j)})\}$$

$$v(s_b) = \{(w_1, w_{b1}), (w_2, w_{b2}), \dots, (w_{i+j}, w_{b(i+j)})\}$$

$$Cosine\ (s_a, s_b) = \frac{\sum_{k=1}^{i+j} w_{ak} w_{ab}}{\sqrt{\sum_{k=1}^{i+j} w_{ak}^2} \sqrt{\sum_{k=1}^{i+j} w_{bk}^2}} \quad \text{..... [2.5]}$$

Cosine dari dua vektor dapat diperoleh dengan menggunakan formula *Euclidean dot product :*

$$a \,.\, b = \|a\| \, \|b\| \cos\theta \quad \text{.....[2.6]}$$

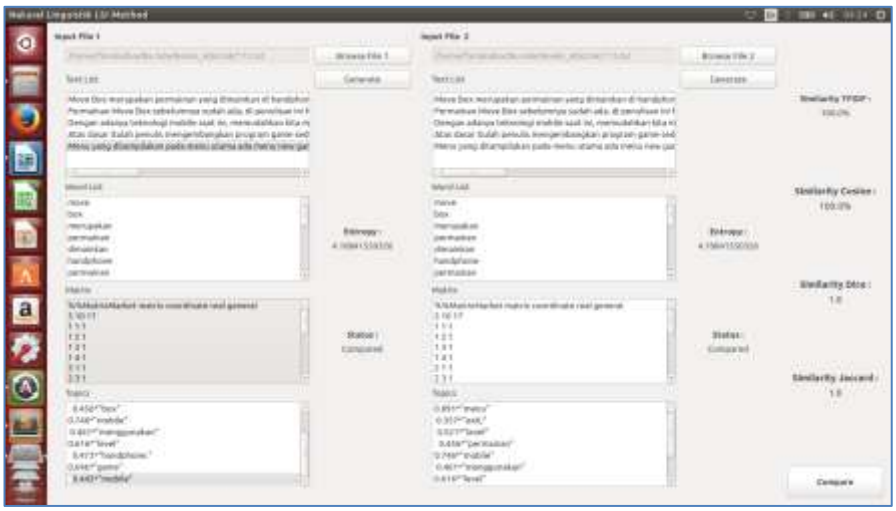
Terdapat dua atribut vektor yaitu A dan B, *cosine similarity* cos(0) direpresentasikan dengan menggunakan *dot product* dan *magnitute* :

$$similarity = \cos\theta = \frac{A \,.\, B}{\|A\| \|B\|} = \frac{\sum_{i=1}^{n} A_i \, x \, B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \, x \, \sqrt{\sum_{i=1}^{n} (B_i)^2}} \quad \text{.....[2.7]}$$

3. HASIL DAN PEMBAHASAN

Pengujian dokumen dilakukan terhadap dokumen yang diduga mempunyai kemiripan dengan dokumen lain. Dokumen 1 merupakan dokumen yang diuji, sedangkan dokumen 2 merupakan dokumen asli. Uji coba dilakukan terhadap Dokumen yang sama, dokumen yang berbeda, dokumen yang dimodifikasi yaitu dengan memindahkan dua posisi kalimat dan tiga posisi kalimat, dan dengan menggantikan beberapa kata dengan sinonimnya.

Berikut ini merupakan tampilan aplikasi program yang menggunakan tools Gensim sebagai aplikasi yang membantu pengukuran kemiripan dokumen.



Gambar 2. Aplikasi Pengukuran Kemiripan

Berikut ini merupakan hasil pengujian kemiripan dokumen :

Tabel 1. Hasil Pengujian Dokumen yang Sama

NO	DOK1	DOK2	TF-IDF (%)	NILAI SIMILARITAS (%)		
				COSINE	DICE	JACCARD
1	101	101	100	100	100	100
2	102	102	100	100	100	100
3	103	103	100	100	100	100
:						
:						
28	128	128	100	100	100	100
29	129	129	100	100	100	100
30	130	130	100	100	100	100

Hasil pengujian yang telah dilakukan terhadap 30 dokumen sebagai berikut : berdasarkan kolom prosentase kemiripan menunjukkan bahwa prosentase kemiripan 100% untuk semua dokumen yang dibandingkan dan untuk semua pengukuran kemiripan karena dokumen 1 dan dokumen 2 yang diperiksa adalah sama.

Tabel 2. Hasil Pengujian Dokumen yang Tidak Sama

NO	DOK1	DOK2	TF-IDF (%)	NILAI SIMILARITAS		
				COSINE	DICE	JACCARD
1	101	102	56,92	59,54	31,18	18,99
2	102	103	63,54	89,61	23,43	11,71
3	103	104	55,48	88,33	25,86	12,93
:						
:						
28	128	129	63,86	29,82	29,02	17,43
29	129	130	73,13	81,40	29,05	14,52
30	130	131	85,28	79,27	19,53	9,77

Hasil pengujian yang telah dilakukan terhadap 30 dokumen sebagai berikut : berdasarkan kolom prosentase kemiripan menunjukkan bahwa prosentase kemiripan yang dihasilkan sangat bervariasi karena dokkumen 1 dan dokumen 2 yang diperiksa adalah dokumen yang berbeda.

Tabel 3. Hasil Pengujian Dokumen Transposisi

NO	NAMA DOK	TRANSPOSISI 2 KALIMAT				TRANSPOSISI 3 KALIMAT			
		TFIDF	COS	DICE	JACCARD	TFIDF	COS	DICE	JACCARD
1	105	78,20	91,51	51,76	45,67	76,71	92,70	64,29	58,33
2	108	83,98	78,97	22,27	11,21	81,41	79,97	22,43	11,34
3	111	78,24	85,02	65,32	56,00	71,92	86,11	69,52	56,57
4	112	40,41	35,92	29,41	18,43	75,34	68,04	35,48	27,42
5	115	96,58	80,20	60,00	46,67	88,25	86,54	65,00	56,19

Hasil pengujian yang telah dilakukan terhadap 5 dokumen sebagai berikut : berdasarkan kolom prosentase kemiripan menunjukkan bahwa perhitungan kemiripan dengan *Cosinus, Dice* dan *Jaccard* untuk transposisi dua kalimat dan tiga kalimat mempunyai prosentase yang meningkat karena jumlah transposisi kalimat bertambah dari dua kalimat ke tiga kalimat.

Tabel 4. Hasil Pengujian Dokumen Similar

NO	NAMA DOK	5 KATA SINONIM				10 KATA SINONIM				15 KATA SINONIM			
		1	2	3	4	1	2	3	4	1	2	3	4
1	101	87,35	93,60	56,23	48,70	81,56	88,67	48,71	39,34	83,04	92,70	75,40	65,08
2	102	96,92	96,66	60,09	45,57	96,89	96,19	37,46	24,40	93,89	95,25	41,61	28,52

ISBN 978-602-99334-4-4

NO	NAMA DOK	5 KATA SINONIM				10 KATA SINONIM				15 KATA SINONIM			
		1	2	3	4	1	2	3	4	1	2	3	4
3	103	57,50	97,66	18,00	11,11	52,87	96,83	23,43	14,67	69,51	97,04	26,67	16,22
4	104	74,70	99,51	41,18	28,36	69,40	98,45	39,25	26,87	73,43	98,78	40,46	27,77
5	105	90,27	98,33	51,92	40,12	82,52	97,17	20,05	11,32	84,92	99,90	55,36	42,63

Keterangan nomor kolom : 1. Perhitungan *tf-idf*, 2. Perhitungan *Cosinus similarity*, 3. Perhitungan Dice *similary*, dan 4. Perhitungan Jaccard *similarity*.

Hasil pengujian yang telah dilakukan terhadap 3 dokumen sebagai berikut : berdasarkan kolom prosentase kemiripan menunjukkan bahwa perhitungan kemiripan dengan *Cosinus* untuk sinonim 5 kata, 10 kata, dan 15 kata menghasilkan nilai prosentase mendekati 100%.

4. KESIMPULAN

Hasil pengujian terhadap dokumen yang sama menghasilkan prosentase kemiripan 100% baik pengkuran similaritas *Cosine*, *Dice*, maupun *Jaccard*. Hasil pengujian kelompok dokumen kedua yaitu dokumen yang tidak sama, menghasilkan prosentase yang bervariasi baik pengkuran similaritas *Cosine, Dice,* maupun *Jaccard*. Hasil pengujian kelompok dokumen transposisi menghasilkan prosentase kemiripan dengan *Cosinus, Dice, dan Jaccard* untuk transposisi dua kalimat dan tiga kalimat mempunyai prosentase yang meningkat. Hasil pengujian kelompok dokumen similar menghasilkan prosentase kemiripan dengan *Cosinus* yang hampir semuanya mendekati 100% kecuali hasil pengukuran kemiripan *Dice* dan *Jaccard*. Kinerja dari Dice dan Jaccard mendekati fungsi similaritas Cosine yang merupakan pengukuran berbasiskan vektor kata. Dari hasil tersebut dapat disimpulkan bahwa menunjukkan bahwa *tools* Gensim yang dipakai pada penelitian ini dapat memeriksa kemiripan dokumen. Nilai prosentase kemiripan yang dihasilkan dapat digunakan untuk mengetahui apakah dokumen yang diuji merupakan dokumen hasil plagiarisme atau bukan.

DAFTAR PUSTAKA

http://radimrehurek.com/gensim, tanggal akses : 14 Februari 2015.

Gooma, Wael H., and Aly A. Fahmy, (2013), *A Survey of Text Similarity Approaches*, International Journal of Computer Applications (0975 – 8887), Volume 68, No. 13.

Heninggar Saptiantri, 2009, *Perbandingan Metode Latent Semantic Analysis dan Vector Space Model Untuk Sistem Penilai Jawaban Esai Otomatis Bahasa Indonesia*, Fakultas Ilmu Komputer, Universitas Indonesia.

Kim, Woosung., and Sanjeev Khudanpur., 2004, *Cross-Lingual Latent Semantic Analysis For Language Modeling*, In Proceeding of ICASSP, pages 257–260.

Landauer, T. K., Foltz, P. W., and Laham, D.,(1998), *Introduction to Latent Semantic Analysis*. Discourse Processes, 25, 259-284.

Lay, D. C,. (1996), *Linear Algebra And Its Applications*, Second Edition, Addison Wesley Longman.

Shams, Khalid., 2010, *Plagiarism Detection Using Semantic Analysis, School of Engineering and Computer Science*, Thesis Report, BRAC University, Dhaka, Bangladesh.

Stein, B., and Eissen, S., (2006), *Near Similarity Search and Plagiarism Analysis*, Conference of German Classification Sociaty Magdeburg, ISBN 1431-8814, pp. 430-437.

Steinberger, Josef., and Karel Ježek, 2004, *Using Latent Semantic Analysis in Text Summarization and Summary Evaluation,* In Proceedings ISIM, pages 93 – 100.

Zhang, J., Yunchuan Sun, Huilin Wang, and Yanqing He., (2011), *Calculating Statistical Similarity between Sentences*, Journal of Convergence Information Technology, Volume 6, Number 2.