KRESNA: Jurnal Riset dan Pengabdian Masyarakat
Volume 1, Nomor 1, November, 2021, Hal: 51-59 ISSN: 2809-6509 (Online)

Survei Terhadap Pengukuran Kesamaan Teks

Survey of Text Similarity Measurement

Krisna Adiyarta[1*], Suwasti Broto[2]

[1]Fakultas Teknologi Informasi
[2]Fakultas Teknik
[1,2]Universitas Budi Luhur
E-mail: krisna.adiyarta@budiluhur.ac.id[1]*, suwasti.broto@budiluhur.ac.id[2]
(* corresponding author)

Abstract

Measuring the similarity between words, sentences, paragraphs and documents is an important research and discussion space in various discussions such as information search, document grouping, word-sense disambiguation, automatic essay scoring, short answer assessment, machine translation and text summarization. This article discus a survey which discussing methods for measuring the similarity of text or strings. This article is structred into three approaches; String-based, corpus-based, and knowledge-based similarity measures and presents the combinations of these similarities measures.

Keywords **:** *text similarity, semantic similarity, string-based similarity, corpus-based similarity, knowledge-based similarity.*

Abstrak

Mengukur kesamaan antara kata, kalimat, paragraf dan dokumen merupakan ruang peneitian dan bahasan yang penting dalam berbagai diskusi seperti pencarian informasi, pengelompokan dokumen, disambiguasi kata-indra, penilaian esai otomatis, penilaian jawaban singkat, terjemahan mesin dan peringkasan teks. Artikel ini melakukan survei ini terhadap diskusi-diskusi yang membahas tentang metode dalam mengukur kesamaan teks atau string. Arikel ini membaginya menjadi tiga pendekatan; Kesamaan berbasis string, berbasis korpus, dan berbasis pengetahuan serta menyajikan kombinasi antara persamaan-persamaan tersebut.

Kata kunci **:** kesamaan teks, kesamaan semantik, kesamaan berbasis string, kesamaan berbasis korpus, kesamaan pengetahuan

1. PENDAHULUAN

Pengukuran terhadap kesamaan teks memainkan peran yang semakin penting dalam penelitian dan aplikasi yang terkait dengan teks pada fungsi-fungsi seperti pencarian informasi, klasifikasi teks, pengelompokan dokumen, pendeteksian topik, pelacakan topik, pembuatan pertanyaan, mejawab pertanyaan, penilaian terhadap esai, penilaian jawaban singkat, penterjemahan mesin, peringkasan teks dan lain-lain. Menentukan kesamaan atau kemiripan antar kata merupakan prosedur utama pada fungsi untuk mengukur kesamaan teks yang kemudian digunakan sebagai bagian utama dalam mengukur persamaan kalimat, paragraf dan dokumen. Pengukuran atas kesamaan kata dapat ditinjau dalam dua cara yaitu peninjauan secara leksikal dan semantik. Kata-kata yang dinyatakan serupa secara leksikal apabila pada kata-kata tersebut memiliki bentuk urutan penggunaan karakter yang mirip. Kata-kata akan dinyatakan serupa secara semantik apabila pada kata-kata tersebut memiliki makna yang sama,

digunakan dengan cara yang sama, digunakan dalam konteks yang sama. Dalam maknakel ini, kesamaan leksikal didiskusikan dalam implementasi algoritma pencocokan yang berbasis kepada bentukan string sementara kesamaan semantik didiskusikan dalam bentuk implementasi algoritma yang berbasis korpus dan berbasis pengetahuan. Langkah-langkah pada algoritma yang berbasis string berfokus kepada operasi-operasi yang memperhatikan urutan karakter pada string dan komposisi karakter. Metrik string adalah ukuran yang digunakan untuk mengukur tingkat kesamaan atau perbedaan (ketidaksamaan) antara dua string teks dalam pencocokan atau perbandingan string. Kesamaan berbasis korpus adalah ukuran kesamaan semantik yang akan mengidikasikan tingkat kesamaan diantara kata dan kata dengan memperhatikan informasi yang diperoleh dari sebuah korpora yang besar. Kesamaan yang berbasis kepada pengetahuan adalah pengukuran kesamaan semantik dari sepasang kata dengan menggunakan informasi yang berasal dari jaringan semantic (*Semantic Network*).

Artikel ini disusun sebagai berikut: pada bagian kedua akan menyajikan diskusi survey terhadap algoritma yang berbasis string yang dibagi menjadi dua jenis yaitu ukuran berbasis karakter dan ukuran yang berbasis istilah. Pada bagian ketiga dan keempat di maknakel ini menguraikan survey terhadap algoritma yang berbasis kepada korpus dan berbasis kepada pengetahuan. Uraian pada bagian ke-enam menguraikan survey terhadap algoritma yang mengkombinasikan beberapa pendekan. Diakhir maknakel ini, bagian enam menyajikan kesimpulan survei

2. KESAMAAN BERBASIS STRING

Langkah-langkah dalam mengukur tingkat kesamaan string dioperasikan pada urutan atau bentukan string dan komposisi karakter. Metrik string adalah metrik yang digukanan untuk mengukur tingkat kesamaan atau perbedaan (ketidaksamaan) antara dua string teks dalam fungsi untuk pencocokan atau perbandingan dua string. Maknakel ini merujuk kepada ukuran kesamaan string yang diimplementasikan dalam SimMetrics [1]. SimMetrics mendifinisikam empat belas algoritma yang diperkenalkan secara singkat. Tujuh di antaranya berbasis karakter sementara yang lain mengukur perbedaan (ketidaksamaan) yang didasarkan kepada penggunaan istilah (term).

2.1. Ukuran Kesamaan Berbasis Karakter

Jaro, pengukuran ini didasarkan pada urutan karakter umum dan jumlah di antara dua string dan dengan mempertimbangkan ejaan dalam bentukan yang khusus dan yang umum digunakan [2, 3]. *Jaro–Winkler* merupakan pengembangan dari pendekatan *Jaro*; pedekatan ini menggunakan skala imbuhan awalan yang memberikan peringkat yang lebih baik untuk string yang cocok dari sekumpulan panjang awalan [4]. *Longest Common SubString* (*LCS*), ukuran ini menganggap bahwa kesamaan antara dua string didasarkan kepada panjang rantai karakter yang berdekatan yang ada pada kedua string yang dibandingkan. *Damerau-Levenshtein* mendefinisikan jarak dari dua string dengan menghitung jumlah minimum operasi yang diperlukan untuk mengubah satu string ke string yang lain, di mana operasi eperasi yang didefinisikan adalah pada operasi penghapusan, penyisipan, atau transposisi dua karakter yang berdekatan, atau penggantian karakter tunggal [5, 6]. *Needleman-Wunsch* algorithm merupakan salah satu bentuk pemrograman dinamis. Pada awalnya algoritma ini diterapkan pada aplikasi pemrograman dinamis untuk membandingkan perbandingan urutan untuk domain biologis. Algoritma ini melakukan penyelarasan secara global untuk menemukan penyelarasan terbaik dari dua urutan. Sangat cocok pada dua urutan memiliki panjang yang sama, dengan tingkat kesamaan yang signifikan [7]. *N-gram* adalah sub-urutan dari *n* item dari urutan sebuah string/teks. Dalam algoritma kesemaan, *n-gram* dari dua buah string akan dibandingkan. Jarak

dihitung dengan membagi jumlah *n-gram* yang serupa dengan jumlah maksimum *n-gram* [8]. *Smith-Waterman* juga merupakan algoritma dalam bentuk pemrograman dinamis. Algoritma ini melakukan penyelarasan lokal sebagai cara untuk menemukan penyelarasan terbaik atas domain yang dilestarikan dari dua urutan. Hal ini berguna untuk urutan-urutan berbeda yang diduga mengandung daerah kesamaan atau motif berurutan yang serupa dalam konteks urutan yang lebih besar [9].

2.2. Ukuran Kesamaan Berbasis Istilah

Jarak Blok (*Block Distance*) atau dikenal sebagai jarak *Manhattan* (*Manhattan Distance*). Pendekatan ini melakukan penghitungan dengan memperhatikan jarak untuk dari satu titik data ke yang lain. Jarak antara dua item adalah jumlah dari perbedaan komponen/elemen dalam string yang sesuai [10]. Kesamaan kosinus (*Cosine similarity*) adalah ukuran kesamaan antara dua vektor ruang yang didapatkan dari hasil kali dalam yang mengukur kosinus sudut di antara keduanya. Jarak *Euclidean* (*Euclidean distance*) atau jarak *L2* adalah akar kuadrat dari jumlah selisih kuadrat antara elemen-elemen yang bersesuaian dari dua vektor. Kesamaan *Jaccard* (*Jaccard similarity*) ukuran yang menghitung jumlah istilah/term bersama terhadap jumlah semua istilah/term yang unik dari kedua string [11]. Koefisien *Dice's* (*Dice's coefficient*) merupakan ukuran yang didefinisikan sebagai dua kali jumlah suku umum pada string yang dibandingkan dibagi dengan jumlah total elemen pada kedua string [12]. Koefisien Pencocokan (*Matching Coefficient*) adalah pendekatan berbasis vektor yang sangat sederhana yang hanya menghitung jumlah elemen yang mirip pada kedua vektor yang bukan nol. Koefisien tumpang tindih (*Overlap coefficient*) mirip dengan koefisien *Dice's* akan tetapi pengukuran ini menganggap dua string cocok sepenuhnya jika salah satu adalah merupakan bagian dari yang lain.

3. KESAMAAN BERBASIS KORPUS. KESAMAAN BERBASIS STRING

Kesamaan berbasis korpus adalah pendekatan dalam mengukur kesamaan semantik dengan memanfaatkan informasi yang diperoleh dari korpora besar. Korpus adalah merupakan kumpulan besar teks tertulis atau lisan yang biasa digunakan dalam penelitian bahasa. Beberapa pendekatan dalam katagori ini diantaranya adalah *Latent Semantic Analysis* (*LSA*), *Hyperspace Analogue to Language* (*HAL*), *Generalized Latent Semantic Analysis* (*GLSA*), *Explicit Semantic Analysis* (*ESA*).

Latent Semantic Analysis (*LSA*) [13] adalah teknik paling populer dari sekian banyak pendekatan untuk mengukur kesamaan yang berbasis pada korpus. *LSA* mengasumsikan bahwa kata-kata yang memiliki makna yang dekat akan muncul dalam potongan-potongan teks yang serupa. Sebuah matriks yang berisi jumlah kata per paragraf (baris yang mewakili kata-kata unik dan kolom mewakili setiap paragraf) dibangun dari potongan besar teks dan disertai teknik matematika yang disebut singular value decomposition (*SVD*) digunakan untuk mengurangi jumlah kolom sambil mempertahankan kesamaan struktur antar baris. Kata-kata tersebut kemudian dibandingkan dengan mengambil kosinus sudut antara dua vektor yang dibentuk oleh dua baris.

Hyperspace Analogue to Language (*HAL*) [14,15] menciptakan ruang semantik dari kemunculan kata bersama. Matriks kata demi kata dibentuk dengan setiap elemen matriks yang memperlihatkan kekuatan asosiasi antara kata yang diwakili oleh baris dan kata yang diwakili oleh kolom. Pengguna algoritma kemudian memiliki pilihan untuk menghapus kolom entropi yang rendah dari matriks. Saat teks dianalisis, kata yang menjadi fokus ditempatkan di awal jendela dengan sepuluh kata yang merupakan kata-kata tetangga/terdekat yang akan turut

dihitung bersama. Nilai matriks diakumulasikan dengan pembobotan bersama yang berbanding terbalik dengan jarak dari kata yang fokuskan; kata-kata tetangga yang lebih dekat dianggap mencerminkan lebih banyak semantik kata yang difokuskan dan karenanya berbobot lebih tinggi. *HAL* juga mencatat informasi pengurutan kata dengan memperlakukan keberadaannya bersama secara berbeda berdasarkan apakah kata tetangga muncul sebelum atau sesudah kata yang difokuskan.

Generalized Latent Semantic Analysis (*GLSA*) [16] merupakan bentuk kerangka kerja yang menghitung istilah pada vektor dokumen yang ditujukan untuk mendapatkan semantic dokumen. Ini merupakan pengembangan pendekatan *LSA* yang berfokus pada vektor dari istilah daripada representasi istilah dokumen ganda. *GLSA* membutuhkan ukuran yang mengasosiasikan semantik antara istilah dan metode pengurangan dimensi. Pendekatan *GLSA* dapat menggabungkan segala jenis ukuran kesamaan pada ruang suku dengan metode pengurangan dimensi yang sesuai. Matriks dokumen istilah tradisional digunakan pada langkah terakhir untuk memberikan bobot dalam kombinasi linier dari vektor istilah.

Explicit Semantic Analysis (*ESA*) [17] adalah ukuran yang digunakan untuk menghitung keterkaitan semantik antara dua teks arbitrer. Teknik berbasis Wikipedia merepresentasikan istilah (atau teks) sebagai vektor berdimensi tinggi, setiap entri vektor menyajikan bobot TF-IDF antara istilah dan satu maknakel Wikipedia. Keterkaitan semantik antara dua istilah (atau teks) dinyatakan dengan ukuran kosinus antara vektor yang sesuai.

Pointwise Mutual Information - Information Retrieval (*PMI-IR*) [18] adalah metode untuk menghitung kesamaan antara pasangan kata, menggunakan sintaks query / Pencarian AltaVista untuk menghitung probabilitas. Semakin sering dua kata muncul bersamaan di halaman web, semakin tinggi skor kesamaan *PMI-IR* mereka.

The cross-language explicit semantic analysis (*CL-ESA*) [19] adalah generalisasi multibahasa dari *ESA*. *CL-ESA* mengeksploitasi koleksi referensi multibahasa yang selaras dengan dokumen seperti Wikipedia untuk mewakili dokumen sebagai vektor konsep yang tidak bergantung pada bahasa. Keterkaitan dua dokumen dalam bahasa yang berbeda dinilai oleh kesamaan kosinus antara representasi vektor yang sesuai.

Second-order co-occurrence pointwise mutual information (*SCO-PMI*) [20,21] adalah ukuran kesamaan semantik menggunakan informasi timbal balik pointwise untuk mengurutkan daftar kata tetangga penting dari dua kata target dari korpus besar. Keuntungan menggunakan *SOC-PMI* adalah dapat menghitung kemiripan antara dua kata yang tidak sering muncul bersamaan, karena keduanya muncul bersama dengan kata tetangga yang sama.

Normalized Google Distance (*NGD*) [22] adalah ukuran kesamaan semantik yang diturunkan dari jumlah klik (pilihan pengguna) yang dikembalikan oleh mesin pencari *Google* untuk sekumpulan kata kunci tertentu. Kata kunci dengan makna yang sama atau mirip dalam bahasa alami cenderung berjarak "dekat" pada satuan jarak *Google*, sedangkan kata-kata dengan makna yang berbeda cenderung berjarak lebih jauh. Secara khusus, jarak *Google* yang dinormalisasi di antara dua istilah pencarian *x* dan *y* adalah

$$NGD(x,y) = \frac{max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log M - \{\log f(x), \log f(y)\}}$$

di mana *M* adalah jumlah total halaman *web* yang dicari oleh *Google*; *f*(*x*) dan *f*(*y*) masing-masing adalah jumlah klik untuk penelusuran istilah *x* dan *y* dan *f*(*x*, *y*) adalah jumlah halaman web dimana *x* dan *y* muncul. Jika dua istilah dalam penelusuran *x* dan *y* tidak pernah muncul secara bersamaan pada halaman web yang sama, tetapi terjadi secara terpisah, maka jarak Google yang dinormalisasi di antara keduanya tidak terbatas. Jika kedua suku selalu muncul bersamaan, *NGD*-nya adalah nol, atau setara dengan koefisien antara *x* kuadrat dan *y* kuadrat.

Extracting DIStributionally similar words using CO-occurrences (*DISCO*) [23, 24], Kesamaan distribusi antara kata-kata mengasumsikan bahwa kata-kata dengan makna yang sama terjadi dalam konteks yang sama. Koleksi teks besar dianalisis secara statistik untuk mendapatkan kesamaan distribusi. *DISCO* adalah metode yang menghitung kesamaan distribusi antar kata dengan menggunakan jendela konteks sederhana berukuran ±3 kata dalam menghitung kemunculan bersama. Ketika dua kata dikenakan kesamaan persis, *DISCO* hanya mengambil vektor kata mereka dari data yang diindeks, dan menghitung kesamaan menurut ukuran Lin [25]. Jika kata yang paling mirip secara distribusi diperlukan, *DISCO* mengembalikan vektor kata urutan kedua untuk kata yang diberikan. *DISCO* memiliki dua ukuran kesamaan utama *DISCO1* dan *DISCO2* dimana *DISCO1* menghitung kesamaan urutan pertama antara dua kata masukan berdasarkan kumpulan lokasi mereka. *DISCO2* menghitung kesamaan orde kedua antara dua kata masukan berdasarkan kumpulan kata yang serupa secara distribusi.

4. KESAMAAN BERBASIS PENGETAHUAN

Ukuran yang mendasarkan pada pengidentifikasian derajat kemiripan antar kata dengan menggunakan informasi yang berasal dari jaringan semantik [26]. *WordNe*t [27] adalah jaringan semantik paling populer di bidang pengukuran kesamaan berbasis pengetahuan antara kata-kata; *WordNet* adalah database leksikal bahasa Inggris yang besar. Kata benda, kata kerja, kata sifat dan kata keterangan dikelompokkan ke dalam set sinonim kognitif (*synsets*), masing-masing mengekspresikan konsep yang berbeda. Synsets saling terkait melalui hubungan konseptual-semantik dan leksikal.

Kesamaan berbasis pengetahuan dapat dibagi secara kasar menjadi dua kelompok: ukuran kesamaan semantik dan ukuran keterkaitan semantik. Konsep serupa secara semantik dianggap terkait berdasarkan kemiripannya. Keterkaitan semantik, di sisi lain, adalah gagasan keterkaitan yang lebih umum, tidak secara khusus terikat pada bentuk atau bentuk konsep. Dengan kata lain, kesamaan semantik adalah sejenis keterkaitan antara dua kata, ini mencakup hubungan yang lebih luas antara konsep yang mencakup hubungan kesamaan ekstra seperti *is-a-specific-example-of, is-a-kind-of,* is-*a-specific-example-of, is-a-part-of, is-a-kind-of, is-oposite-of* [28].

Ada enam bentuk ukuran untuk kesamaan semantik; tiga ukuran didasarkan kepada konten atau isi informasi yaitu: Lin (*lin*) [25], Resnik (*res*) [29] dan Jiang & Conrath (*jcn*) [30]. Sementara, tiga ukuran lainnya didasarkan kepada seberapa panjang jalur yang menghubungkan kedua istilah atau kata yang diukur yaitu: Leacock & Chodorow (*lch*) [31], Wu & Palmer (*wup*) [32] dan Panjang Jalur (*path*).

Nilai terkait dalam ukuran res sama dengan kandungan informasi (*IC*) dari *Least Common Subsumer*. Ini bermakna bahwa nilainya akan selalu lebih besar dari atau sama dengan nol. Batas atas pada nilai umumnya cukup besar dan bervariasi tergantung pada ukuran korpus yang digunakan untuk menentukan nilai konten informasi. Pengukuran *lin* dan *jcn* menambah isi informasi dari *Least Common Subsumer* dengan jumlah isi informasi dari konsep A dan B itu sendiri. Ukuran *lin* menskalakan isi informasi dari Subsumer yang Paling Kurang Persekutuan

dengan jumlah ini, sedangkan *jcn* mengambil selisih dari jumlah ini dan isi informasi dari *Subsumer* pada persekutuan terkecil.

Ukuran *lch* mengembalikan skor yang menunjukkan seberapa mirip dua indera kata, berdasarkan jalur terpendek yang menghubungkan indra dan kedalaman maksimum taksonomi di mana indra terjadi. Pengukuran *wup* mengembalikan skor yang menunjukkan seberapa mirip dua indera kata, berdasarkan kedalaman dua indera dalam taksonomi dan *Least Common Subsumer* mereka. ukuran jalur mengembalikan skor yang menunjukkan seberapa mirip dua indera kata, berdasarkan jalur terpendek yang menghubungkan indra dalam taksonomi *is-a* (hipernim/hiponim).

Ada tiga ukuran lain yang keterkaitan semantik: Lesk (*lesk*) [33], St. Onge (*hso*) [34] dan pasangan vektor (vektor) [35]. Pengukura *hso* bekerja dengan menemukan rantai leksikal yang menghubungkan dua indera kata. Ada tiga kelas hubungan yang dipertimbangkan: ekstra kuat, kuat, dan kuat sedang. Skor keterkaitan maksimum adalah 16. ukuran *lesk* bekerja dengan cara menemukan *synset* yang tumpang tindih dalam *glosses* dari dua *synsets*. Skor keterkaitan adalah jumlah kuadrat dari panjang tumpang tindih. ukuran vektor membuat matriks kejadian tambahan untuk setiap kata yang digunakan dalam *WordNet glos* dari korpus tertentu, dan kemudian mewakili setiap konsep dengan vektor yang merupakan rata-rata dari vektor yang muncul (terjadi). Paket paling populer yang mencakup ukuran kesamaan berbasis pengetahuan adalah *WordNet: Similarity* dan *Natural Language Toolkit* (*NLTK*).

5. UKURAN KESAMAAN HIBRIDA

Metode hibrida menggunakan beberapa ukuran kesamaan; banyak penelitian sudah dilakukan pada bidang ini. Delapan ukuran kesamaan semantik diuji di [26]. Dua dari penelitian ini adalah penelitian berbasis korpus dan enam lainnya berbasis pengetahuan. Pertama, kedelapan algoritma ini dievaluasi secara terpisah, kemudian digabungkan bersama. Performa terbaik dicapai dengan menggunakan metode yang menggabungkan beberapa metrik kesamaan menjadi satu.

Sebuah metode untuk mengukur kesamaan semantik antara kalimat atau teks yang sangat pendek, berdasarkan informasi semantik dan urutan kata didiskusiskan dalam [36]. Pertama, kesamaan semantik diturunkan dari basis pengetahuan leksikal dan korpus. Kedua, metode yang diusulkan mempertimbangkan dampak urutan kata pada makna kalimat. Kesamaan urutan kata turunan mengukur jumlah kata yang berbeda serta jumlah pasangan kata dalam urutan yang berbeda.

Penulis [37] menyajikan pendekatan yang menggabungkan ukuran keterkaitan semantik berbasis korpus di seluruh kalimat bersama dengan skor kesamaan semantik berbasis pengetahuan yang diperoleh untuk kata-kata yang berada di bawah peran sintaksis yang sama di kedua kalimat. Semua skor sebagai fitur diumpankan ke model pembelajaran mesin, seperti regresi linier, dan model bagging untuk mendapatkan skor tunggal yang memberikan tingkat kesamaan antar kalimat. Pendekatan ini menunjukkan peningkatan yang signifikan dalam menghitung kesamaan semantik antara kalimat dengan menyisir ukuran kesamaan berbasis pengetahuan dan ukuran keterkaitan berbasis korpus terhadap ukuran berbasis korpus yang diambil sendiri.

Penulis [38] mempresentasikan sebuah metode dan menamakannya *Semantic Text Similarity* (*STS*). Metode ini menentukan kesamaan dua teks dari kombinasi antara informasi semantik dan sintaksis. Mereka mempertimbangkan dua fungsi wajib (kesamaan string dan

kesamaan kata semantik) dan fungsi opsional (kesamaan urutan kata umum). Metode *STS* mencapai koefisien korelasi Pearson yang sangat baik untuk 30 pasangan kalimat dari kumpulan data dan mengungguli hasil yang diperoleh pada [36].

Korelasi yang menjanjikan antara hasil kesamaan manual dan otomatis dicapai pada [39] dengan menggabungkan dua modul. Modul pertama menghitung kemiripan antar kalimat menggunakan kemiripan berbasis N-gram, dan modul kedua menghitung kemiripan antar konsep dalam dua kalimat menggunakan ukuran kemiripan konsep dan *WordNet*.

6. KESIMPULAN

Dalam survei ini dibahas tiga pendekatan kesamaan teks; Kesamaan berbasis string, berbasis Corpus, dan berbasis Pengetahuan. Langkah-langkah Berbasis String beroperasi pada urutan string dan komposisi karakter. Empat belas algoritma diperkenalkan; Tujuh di antaranya berbasis karakter sementara yang lain mengukur jarak berdasarkan istilah. Kesamaan berbasis korpus adalah ukuran kesamaan semantik yang menentukan kesamaan antar kata menurut informasi yang diperoleh dari korpora besar. Sembilan algoritma dijelaskan; *HAL*, *LSA*, *GLSA*, *ESA*, *CL-ESA*, *PMI-IR*, *SCO-PMI*, *NGD* dan *DISCO*. *Knowledge-Based* similarity merupakan salah satu ukuran kesamaan semantik yang didasarkan pada identifikasi derajat kemiripan antar kata dengan menggunakan informasi yang berasal dari jaringan semantik. Sembilan algoritma diperkenalkan; Enam di antaranya didasarkan pada kesamaan semantik *-res*, *lin*, *jcn*, *lch*, *wup* dan *path*- sedangkan tiga lainnya didasarkan pada keterkaitan semantik *-hso, lesk* dan *vektor-*. Beberapa dari algoritma ini digabungkan bersama dalam banyak penelitian. Akhirnya paket kesamaan yang berguna disebutkan seperti *SimMetrics, WordNet*: *Similarity* dan *NLTK*.

DAFTAR PUSTAKA

[1] Chapman, S. (2006). SimMetrics: a java & c# .net library of similarity metrics, http://sourceforge.net/projects/simmetrics/.
[2] Jaro, M. A. (1989). Advances in record linkage methodology as applied to the 1985 census of Tampa Florida, *Journal of the American Statistical Society*, vol. 84, 406, pp 414-420.
[3] Jaro, M. A. (1995). Probabilistic linkage of large public health data file, *Statistics in Medicine* 14 (5-7), 491-8.
[4] Winkler W. E. (1990). String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage, *Proceedings of the Section on Survey Research Methods*, American Statistical Association, 354–359.
[5] Hall, P. A. V. & Dowling, G. R. (1980) Approximate string matching, Comput. Surveys, 12:381-402.
[6] Peterson, J. L. (1980). Computer programs for detecting and correcting spelling errors, *Comm. Assoc. Comput. Mach.*, 23:676-687.
[7] Needleman, B. S. & Wunsch, D. C. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins", *Journal of Molecular Biology* 48(3): 443–53.
[8] Alberto, B. Paolo, R., Eneko A. & Gorka L. (2010). Plagiarism Detection across Distant Language Pairs, *In Proceedings of the 23rd International Conference on Computational Linguistics*, pages 37–45.
[9] Smith, F. T. & Waterman, S. M. (1981). Identification of Common Molecular Subsequences, *Journal of Molecular Biology* 147: 195–197.
[10] Eugene FK. (1987). *Taxicab Geometry*, Dover. ISBN 0-486-25202-7.

[11] Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin de la Société Vaudoise des Sciences Naturelles* 37, 547-579.
[12] Dice, L. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3).
[13] Landauer, T.K. & Dumais, S.T. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge", *Psychological Review*, 104.
[14] Lund, K., Burgess, C. & Atchley, R. A. (1995). Semantic and associative priming in a high-dimensional semantic space. *Cognitive Science Proceedings* (LEA), 660-665.
[15] Lund, K. & Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments & Computers*, 28(2),203-208.
[16] Matveeva, I., Levow, G., Farahat, A. & Royer, C. (2005). Generalized latent semantic analysis for term representation. In *Proc. of RANLP*.
[17] Gabrilovich E. & Markovitch, S. (2007). Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis, *Proceedings of the 20th International Joint Conference on Maknaficial Intelligence,* pages 6–12.
[18] Turney, P. (2001). Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the Twelfth European Conference on Machine Learning* (*ECML*).
[19] Mmaknan, P., Benno, S. & Maik, A. (2008). A Wikipedia-based multilingual retrieval model. *Proceedings of the 30th European Conference on IR Research* (*ECIR*), pp. 522-530.
[20] Islam, A. and Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transaction Knowledge Discovery*. *Dat ACM Transactions on Knowledge Discovery from Data* 2 (Jul. 2008), 1–25.
[21] Islam, A. and Inkpen, D. (2006). Second Order Co-occurrence PMI for Determining the Semantic Similarity of Words, in *Proceedings of the International Conference on Language Resources and Evaluation* (*LREC 2006*), Genoa, Italy, pp. 1033–1038.
[22] Cilibrasi, R.L. & Vitanyi, P.M.B. (2007). The Google Similarity Distance, *IEEE Trans. Knowledge and Data Engineering*, 19:3, 370-383.
[23] Peter, K. (2009). Experiments on the difference between semantic similarity and relatedness. In *Proceedings of the 17th Nordic Conference on Computational Linguistics - NODALIDA '09*, Odense, Denmark.
[24] Peter, K. (2009). Experiments on the difference between semantic similarity and relatedness. In *Proceedings of the 17th Nordic Conference on Computational Linguistics - NODALIDA '09*, Odense, Denmark.
[25] Lin, D. (1998b). Extracting Collocations from Text Corpora. In *Workshop on Computational Terminology*, Montreal, Kanada, 57–63.
[26] Mihalcea, R., Corley, C. & Strapparava, C. (2006). Corpus based and knowledge-based measures of text semantic similarity. In *Proceedings of the American Association for Maknaficial Intelligence*. (Boston, MA).
[27] Miller, G.A., Beckwith, R., Fellbaum, C.D., Gross, D. & Miller, K. (1990). WordNet: An online lexical database. *Int. J. Lexicograph.* 3, 4, pp. 235–244.
[28] Patwardhan, S., Banerjee, S. & Pedersen, T. (2003). Using measures of semantic relatedness for word sense disambiguation. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, pp. 241–257.
[29] Resnik, R. (1995). Using information content to evaluate semantic similarity. In *Proceedings of the 14th International Joint Conference on Maknaficial Intelligence*, Montreal, Canada.

[30] Jiang, J. & Conrath, D. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the International Conference on Research in Computational Linguistics*, Taiwan.
[31] Leacock, C. & Chodorow, M. (1998). Combining local context and WordNet sense similarity for word sense identification. In WordNet, An Electronic Lexical Database. The MIT Press.
[32] Wu, Z.& Palmer, M. (1994). Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, Las Cruces, New Mexico.
[33] Banerjee, S. & Pedersen, T. (2002). An adapted Lesk algorithm for word sense disambiguation using WordNet. In *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, pp 136–145.
[34] Hirst, G. & St-Onge, D. (1998). Lexical chains as representations of context for the detection and correction of malapropisms. In *C*. Fellbaum, editor, WordNet: An electronic lexical database, pp 305–332. MIT Press.
[35] Patwardhan, V. (2003). Incorporating dictionary and corpus information into a context vector measure of semantic relatedness. Master's thesis, University of Minnesota, Duluth.
[36] Li, Y., McLean, D., Bandar, Z., O'Shea, J., & Crockett, K. (2006). Sentence similarity based on semantic nets and corpus statistics. *IEEE Transactions on Knowledge and Data Engineering*, 18(8), 1138–1149.
[37] Nitish, A., Kmaknak, A. & Paul, B. (2012). DERI&UPM: Pushing Corpus Based Relatedness to Similarity: Shared Task System Description. *First Joint Conference on Lexical and Computational Semantics* (*SEM*), pages 643–647, Montreal, Canada, June 7-8, 2012 Association for Computational Linguistics.
[38] Islam, A., & Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data*, 2(2), 1–25.
[39] Davide, B., Ronan, T., Nathalie A., & Josiane, M. (2012), IRIT: Textual Similarity Combining Conceptual with an N-Gram Comparison Method. First Joint Conference on Lexical and ComputationalSemantics (*SEM), pages 552–556, Montreal, Canada, June 7-8, 2012 Association for Computational Linguistics.