Implementasi dan Analisis Pengukuran *Semantic Relatedness* menggunakan *Random Walks* berbasis WordNet

Semantic Relatedness Measurement Implementation and Analysis using WordNet-based Random Walks

[1]Muhammad Kenzi, [2]Moch. Arif Bijaksana, Ph.D., [3]Dr. Adiwijaya

[1,2,3]Program Studi Sarjana Teknik Informatika, Fakultas Informatika, Universitas Telkom

[1]kenzimuhammad@gmail.com, [2]arifbijaksana@telkomuniversity.ac.id, [3]kangadiwijaya@gmail.com

Abstrak

Pengukuran *semantic relatedness* merupakan suatu pekerjaan untuk memperkirakan kedekatan arti dari pasangan kata. Pekerjaan ini dengan mudah bisa dilakukan oleh manusia berdasarkan pengalaman dan wawasannya, namun komputer hanya bisa melakukannya dengan mengukur nilai keterkaitannya dengan bantuan pemahaman arti kata seperti kamus, tesaurus, dan ensiklopedia. Salah satu contoh bantuannya yang sering dipakai pada pekerjaan ini yaitu WordNet yang akan digunakan pada pengerjaan tugas akhir ini. Pekerjaan ini akan mempermudah pekerjaan *Natural Language Processing,* seperti pendeteksi plagiarisme antara dua data teks yang berbeda. Pada tugas akhir ini pengukuran *semantic relatedness* dilakukan pada sebuah model *graph* yang dibangun dengan bantuan WordNet. Setiap kata dan arti-artinya di dalam WordNet akan dijadikan sekumpulan *node* yang saling terhubung melalui beberapa tipe *edge* berdasarkan suatu relasi yang menghubungkannya. Tiap tipe *edge* tersebut memiliki cara pembobotan yang berbeda-beda. Pembobotan tersebut akan mempengaruhi hasil dari pengukuran *semantic relatedness.Graph* yang telah dibangun tersebut adalah untuk menjalankan metode pengukuran *semantic relatedness* yang dinamakan *random walks. Graph* akan menjadi tempat berjalannya partikel yang berasal dari *random walks*. Partikel tersebut akan berjalan dari *node* ke *node* lainnya yang terhubung berdasarkan probabilitas berpindahnya. Probabilitas kepindahannya tersebut hanya dipengaruhi oleh *node* yang sedang disinggahi partikel, hal ini lah yang membuat *random walk* ini disebut sebagai *Markov chain*. Hasil dari *semantic relatedness* melalui sistem yang dibuat dengan menggunakan *dataset* Rubenstein dan Goodenough mendapatkan hasil evaluasi menggunakan *Pearson correlation coefficient* dengan nilai tertinggi 0.477.

Kata kunci: *semantic relatedness, WordNet, Natural Language Processing, random walks, graph, Markov chain, Pearson correlation coefficient*

1. Pendahuluan

Mengetahui seberapa terkaitnya suatu kata dengan kata lainnya merupakan sebuah pekerjaan yang mudah bagi manusia berdasarkan pengalaman dan wawasannya. Namun untuk komputer diperlukan suatu pengukuran yang mengandalkan angka untuk menilai seberapa terkaitnya dua pasang kata yang akan dibandingkan. Contohnya adalah kata rumah dengan kata hotel, yang keduanya memiliki kesamaan yaitu suatu tempat akomodasi dan contoh lainnya adalah kucing dan kelelawar yang dua-duanya merupakan mamalia. Nilai keterkaitan kedua pasang kata ini dikenal sebagai *semantic relatedness* [1]. Pengukuran *semantic relatedness* bisa dilakukan dengan bantuan referensi leksikal seperti kamus, tesaurus, dan ensiklopedi. Salah satu bantuan yang sering digunakan dalam menghitung *semantic relatedness* adalah WordNet, sebuah kamus elektronik di mana tiap arti katanya terhubung melalui relasi semantic [2]. Dalam penerapan berbagai cara pengukuran *semantic relatedness*, masalah yang terjadi adalah nilai *semantic relatedness* antarkata yang seharusnya memiliki keterkaitan satu sama lain, namun memiliki sedikit relasi secara langsung. Relasi yang menghubungkan arti kata "apartemen" dengan "rumah" yang keduanya terhubung dengan arti "tempat tinggal". WordNet yang memiliki relasi semantik yang detil sangat membantu pengukuran *semantic relatedness* antarkata, beberapa relasinya adalah sinonim, antonim, meronim, holonim, hipernim dan hiponim [3]. Salah satu metode yang pernah dicoba dalam WordNet ini adalah *shortest path*, dengan harapan dapat menemukan relasi antarkata karena kedekatannya [4]. Namun, tidak semua pasangan kata yang berhubungan pada WordNet memiliki jarak yang dekat saja [5]. Dua kata yang berhubungan bisa memiliki jarak berjauhan pada WordNet dan juga bisa memiliki lebih dari satu relasi berdasarkan jalurnya. Metode *random walks* ini merupakan salah satu cara untuk menemukan relasi antara dua kata tanpa harus mencari yang terdekat saja. Karena *random walks* memiliki partikel yang berpindah-pindah dari suatu *node* ke *node* lain berdasarkan peluang perpindahannya. Partikel tersebut akan berpindah ke banyak *node* dan memungkinkan untuk menemukan banyak jalur untuk bisa mengaitkan *node* yang pertama disinggahinya ke suatu *node* lain [6]. Setiap *node* yang dikunjunginya akan dihitung *stationary distribution*-nya, di mana *stationary distribution* ini dapat digunakan untuk

mengukur *semantic relatedness* antara dua kata yang akan dibandingkan [6]. Untuk itu, pengukuran *semantic relatedness* antara dua kata bisa dikatakan lebih baik hasilnya jika menggunakan *random walks.*Pada tugas akhir ini pengujian *semantic relatedness* menggunakan *cosine similarity* untuk mengukur semua *stationary distributions* yang akan dihitung saat *random walks* berjalan. Data yang diuji adalah *dataset* Rubenstein dan Goodenough [7]. *Dataset* ini merupakan kumpulan data dengan 65 pasang kata yang sudah diberi nilai keterkaitan berdasarkan *human judgement* untuk setiap pasangan katanya.

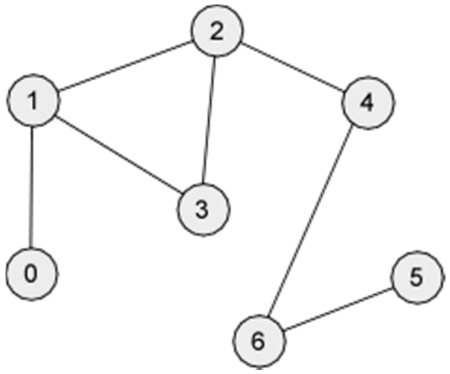
2. Dasar Teori

2.1. WordNet

WordNet adalah suatu kamus elektrik yang tersedia untuk bahasa Inggris. WordNet mengelompokkan setiap kata, baik kata benda, kata kerja, kata sifat maupun kata keterangan dalam suatu kumpulan arti kata yang sinonim dengan lainnya [3]. Kumpulan arti kata yang sinonim ini lebih dikenal dalam istilah *synonym set (synset),* di mana setiap *synset* memiliki definisi singkat, cara penggunaan, dan relasi dengan *synset* lainnya. Pada WordNet juga terdapat *gloss* yaitu penjelasan singkat yang terdapat pada *synset.* Tidak hanya arti saja yang dijelaskan, namun pada beberapa *synset* dijelaskan cara pemakaian kata yang terhubung dengan *synset* [2].

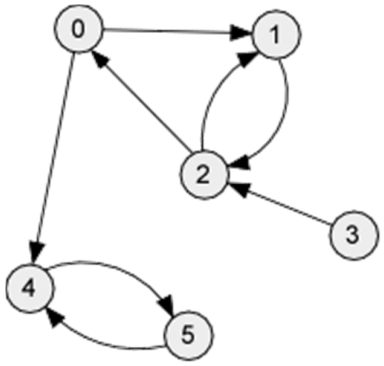
2.2. Markov Chain

Secara konsep, *graph* merupakan sekumpulan *node* dan *edge* di mana tiap *node*-nya memiliki keterhubungan yang digambarkan oleh *edge.* Secara formal, *graph* merupakan satu pasang set *node* dan *edge*, di mana *edge* merupakan suatu elemen yang bersifat *multiset,* yaitu dapat muncul tidak hanya sekali. Suatu *node* mempunyai kemungkinan untuk memiliki lebih dari satu relasi dengan *node* lainnya.



Gambar 2-1: Contoh undirected graph

Graph memiliki beberapa jenis yang sering dijumpai, yaitu *undirected graph* dan *directed graph*. *Undirected graph* merupakan *graph* yang tidak memiliki *edge* berarah, contohnya ada pada Gambar 2-1. *Graph* ini mengartikan bahwa tiap *node* yang terhubung dengan *edge* tidak berarah ini dapat saling mengunjungi satu sama lain tanpa restriksi apapun.



Gambar 2 2: Contoh directed graph

Directed graph merupakan jenis *graph* yang memiliki *edge* berarah untuk menggambarkan relasi antara *node*. Contohnya ada pada Gambar 2 2, di mana tiap *node*-nya memiliki *edge* berarah yang menunjuk ke *node* lain. *Node* yang menunjuk hanya dapat mengunjungi *node* yang ditunjuk tanpa bisa dikunjungi *node* yang ditunjuk tersebut, namun jika *node* yang ditunjuk memiliki *edge* berarah ke arah *node* yang menunjuk maka *node* yang menunjuk dapat dikunjungi lewat *edge* lain.

2.2.1. Model Graph pada Penelitian Hughes dan Ramage

Pada penelitian oleh Hughes dan Ramage, terdapat tiga model *graph* yang dibangun dengan tiga jenis *node* dan beberapa tipe *edge*. Ketiga *graph* ini merepresentasikan relasi antara satu kata dengan kata lainnya, di mana tiap kata direpresentasikan oleh *node* pada *graph*-*graph* tersebut. Relasi antara kedua *node* dalam *graph* berdasarkan data yang diambil dari WordNet versi 2.1. *Node*-*node* tersebut merepresentasikan beberapa jenis *node,* yaitu *Synset, Token,* dan *TokenPOS.* Berikut ini merupakan penjelasan terhadap jenis-jenis *node* tersebut:

1. *Synset* yang merupakan jenis node yang dinamakan dari singkatan dari *synonym set* merupakan arti kata secara luas dari beberapa kata yang dianggap memiliki *semantic relatedness*. Jenis *node* ini merupakan representasi dari arti kumpulan kata pada WordNet. Contohnya adalah "wizard#n#2" dan "sorcerer#n#1" memiliki *synonym set.*

2. *TokenPOS* adalah jenis *node* yang merepresentasikan kata yang sudah dipasangkan dengan jenis kata yang ada pada WordNet. Jenis kata yang ada pada WordNet antara lain *noun* (kata benda), *verb* (kata kerja), *adjective* (kata sifat), dan *adverb* (kata keterangan). *TokenPOS* bisa memiliki relasi dengan lebih dari satu *node Synset*, contohnya "wizard#n" berelasi dengan "wizard#n#1" yang berarti orang yang ahli dalam berbagai bidang, dan juga berelasi dengan "wizard#n#2" yang artinya praktisi sihir.

3. *Token* merepresentasikan kata secara keseluruhan tanpa adanya jenis kata yang dipasangkan terhadap kata tersebut. *Token* berelasi langsung dengan *TokenPOS* dan bisa memiliki minimal satu *TokenPOS* dan maksimal empat *TokenPOS.* Contoh penamaan *node Token* pada *graph* adalah "wizard" yang memiliki relasi dengan dua *TokenPOS* yaitu "wizard#n" dan "wizard#a".

Selain itu, ketiga model *graph* ini juga memiliki beberapa jenis *edge* berdasarkan relasi antara jenis *node*-nya. *Edge*-*edge* pada *graph* ini dibedakan berdasarkan cara pembobotannya. Berikut ini merupakan jenis-jenis *edge* yang menghubungkan *node* pada ketiga model *graph* tersebut:

1. Tipe *edge* yang pertama merupakan *edge* berarah dari *TokenPOS* ke *Synset. Edge* ini memiliki bobot berdasarkan *SemCor frequency counts,* jumlah frekuensi SemCor yang sudah termasuk di dalam WordNet. Karena tidak semua kata memiliki jumlah frekuensi SemCor yang mengakibatkan bobotnya menjadi 0 maka diperlukan bobot baru untuk membedakan relasi antara dua jenis *node* tersebut. Hal ini dibutuhkan karena bobot 0 mengartikan suatu *node* tidak memiliki relasi apapun dengan *node* tetangganya, dapat dikatakan tidak memiliki *edge* yang menghubungkan kedua *node* tersebut. Bobot *edge* ini jika mengikuti penelitian oleh Hughes dan Ramage adalah semacam *pseudo-counts* yang nilainya 0.1 untuk setiap jenis *edge* ini [6].

2. *Edge* berarah yang menghubungkan *Token* ke *TokenPOS* merupakan *edge* tipe kedua. *Edge* ini diberikan bobot dari jumlah semua bobot *edge* jenis pertama, yaitu *edge* yang menghubungkan *node TokenPOS* ke *Synset.*

3. Tipe *edge* ketiga adalah *edge* dua arah yang menghubungkan *Synset* dengan *Synset* lainnya. *Edge* ini dibobotkan berdasarkan jumlah semua *TokenPOS* yang saling terhubung dengan kedua *Synset* tersebut.

4. Tipe *edge* yang keempat merupakan *edge* dua arah yang mirip dengan tipe *edge* ketiga, yaitu menghubungkan *Synset* dengan *Synset* lainnya. Namun, *edge* ini dihubungkan berdasarkan jenis relasi antar kata yang ada pada WordNet, seperti sinonim, antonim, meronim/holonim dan hipernim/holonim. *Edge* ini pada penelitian Hughes dan Ramage tidak diberikan bobot jadi ada asumsi di mana yang dituliskan pada Hughes dan Ramage jika tidak diberikan bobot maka bobotnya dianggap 1.0 [6]. Namun pada penelitian Yanbo Xu diberi bobot 10.0 karena memiliki relasi yang jelas di dalam WordNet [5]. Pada tugas akhir ini, bobot pada tipe *edge* ini akan menjadi salah satu parameter untuk melihat perbedaan antara kedua pembobotan tersebut.

5. Tipe edge kelima adalah *edge* berarah yang menghubungkan *node Synset* ke *TokenPOS.* Pembobotannya berdasarkan penjelasan arti kata pada WordNet yang merupakan suatu kalimat panjang yang memiliki banyak kata dan dilakukan menggunakan *Non-Monotonic Document Frequency* (NMDF), suatu pembobotan kata yang mirip dengan TF-IDF.

Semua model *graph* menggunakan semua jenis *node,* namun tidak semua jenis *edge* digunakan dalam tiap *model* graph. Berikut ini merupakan tiga jenis model *graph* tersebut dan jenis *edge* apa saja yang ada di dalamnya:

1. MarkovLink

   Model *graph* ini menggunakan semua tipe *edge,* kecuali tipe *edge* berarah yang menghubungkan *Synset* ke *TokenPOS.* Model *graph* ini yang akan digunakan pada penelitian tugas akhir ini.

2. MarkovGloss

   Model *graph* ini menggunakan semua tipe *edge,* kecuali dua tipe *edge* dua arah yang menghubungkan *Synset* dengan *Synset* lainnya.

3. MarkovJoined

   Model *graph* ini menggunakan semua tipe *edge,* dan tidak ada pengecualian. Model *graph* ini bisa dikatakan gabungan dari kedua model *graph* di atas.

Ketiga model *graph* tersebut memiliki *node* berjumlah 422.831 *nodes.* Hal ini yang akan digunakan untuk validasi *graph* yang akan dibangun pada penelitian tugas akhir ini.

2.3. Random Walks

Random walks merupakan sebuah metode matematika yang membuat sebuah jalur dari sejumlah langkah yang diambil secara acak. Pada *graph,* partikel *random walk* berpindah dari satu *node* ke *node* lainnya pada tiap langkah melalui bobot *edge* yang sudah dijadikan *transition probabilities* [9]. Setiap *node* yang dilalui akan membuat sebuah rantai *node.* Rantai ini dikenal sebagai *Markov chain,* karena tiap perpindahan ke *node* selanjutnya hanya bergantung pada *node* yang sekarang sedang disinggahi partikel *random walk.*

Dalam pengerjaan tugas akhir ini digunakan metode *random walk* yang menggunakan persamaan (2-1). Persamaan ini diambil dari algoritma Pagerank [9]. Persamaan (2-1) menjelaskan bahwa peluang sebuah partikel untuk mencapai *node i* pada waktu *t* dari *node j* yang ada pada waktu sebelumnya adalah dengan mengalikan jumlah semua peluang yang dari *j* ke *node* lainnya dikalikan dengan peluang partikel berpindah dari *j* ke *i.*

$$P\{X_t = i\} = \sum_{n_j \in V} P\{X_{t-1} = j\}\, P\{X_t = i | P\{X_{t-1} = j\} \qquad (2\text{-}1)$$

2.4. Stationary Distribution

Stationary distribution yang juga dikenal sebagai *limiting probability* pada *Markov chain,* merupakan suatu perhitungan distribusi probabilitas yang tidak akan berubah seiring waktu berjalan jika sudah mencapai batasnya. Batasan ini disebut *convergence*.

$$v_t = \beta v_0 + (1 - \beta) N v_{t-1} \tag{2-2}$$

Dalam menghitung *stationary distribution,* sebelumnya harus diketahui *initial distribution*-nya. Pada penelitian ini, nilai *initial distribution* nya adalah 1 untuk *node* yang sedang disinggahi dan 0 untuk semua *node* lainnya [5]. Setelah diketahui *initial distribution*-nya maka langkah selanjutnya adalah mengukur *stationary distribution* sampai nilai *stationary distribution* sebelum dikurangi *stationary distribution* sesudah adalah kurang dari nilai *convergence.*

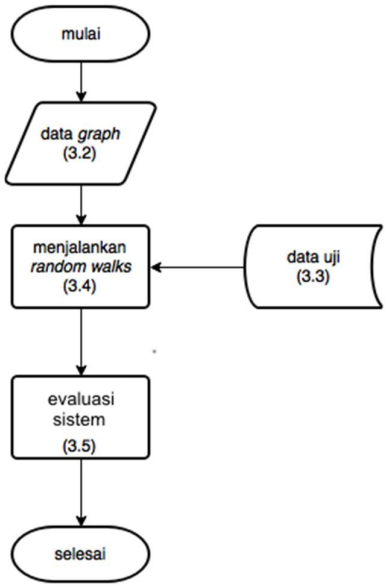
$$\|v_t - v_{t-1}\|_1 < conv \tag{2-3}$$

Cosine similarity merupakan salah satu cara yang sering digunakan untuk mengukur kedekatan antara satu vektor dengan vektor lainnya. Suatu vektor akan dihitung jaraknya berdasarkan kosinus jarak antara kedua vektor tersebut. Perhitungan ini mendekati perhitungan *eucledian distance,* sebuah perhitungan jarak antara suatu titik dengan titik lainnya.

$$sim_{cos}(p, q) = \frac{\sum_{i \in V} p_i q_i}{\|p\| \, \|q\|} \tag{2-4}$$

3. Perancangan Sistem Keseluruhan

Berikut merupakan gambaran sistem untuk setiap tahapan proses yang terdapat di dalamnya yang akan dibangun pada tugas akhir ini:



Gambar 3-1: Rancangan sistem secara keseluruhan

Sistem yang dibangun memiliki dua input yaitu data *graph* dan *dataset.* Kemudian dari *graph* tersebut akan dijalankan *random walks* dengan dua partikel untuk masing-masing kata pada pasangan kata yang ada. Langkah terkahir adalah mencari *similarity* dengan menggunakan *cosine similarity*.

4. Hasil Pengujian

Tabel di bawah ini adalah tabel hasil pengujian berdasarkan parameter yang dilakukan:

Tabel 1: Hasil pengujian berdasarkan bobot

Parameter	Akurasi
Default	73,292
Bobot 1,0	72,267
Bobot 20,0	71,341

Tabel 2: Hasil pengujian berdasarkan langkah

Parameter	Akurasi
Default	73,292
500 langkah	69,443
2000 langkah-	74,579

Tabel di bawah ini adalah tabel hasil pengujian evaluasi sistem:

Tabel 3: Hasil evaluasi sistem

Parameter		Jumlah Sampel	Hasil Evaluasi	Rata-rata Evaluasi
Bobot	Langkah			
10.0	1000	5	0.439	0,450
10.0	1000	5	0.461	
10.0	1000	10	0.393	0,351
10.0	1000	10	0,309	
1.0	1000	5	0,359	0,322
1.0	1000	5	0,285	
20.0	1000	5	0,340	0,329
20.0	1000	5	0,318	
10.0	500	5	0,394	0,399
10.0	500	5	0,404	
10.0	2000	5	0,477	0,438
10.0	2000	5	0,398	

Nilai hasil evaluasi terbesar terdapat pada percobaan dengan bobot 10.0 dan jumlah langkah 2000, yaitu 0.477. Menurut hasil evaluasi **Error! Reference source not found.**, jumlah sampel juga memengaruhi hasil evaluasi, di mana percobaan dengan jumlah sampel 10 mendapatkan hasil yang kurang dibandingkan dengan percobaan dengan parameter yang sama dengan jumlah sampel 5. Namun, rata-rata hasil evaluasi tertinggi tetap pada parameter *default,* yaitu dengan bobot 10.0 dan jumlah langkah 1000.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Dari analisis pengujian yang telah dilakukan di bagian sebelumnya dapat ditarik beberapa kesimpulan. Berikut ini merupakan kesimpulan tersebut:

1. Nilai bobot hanya berpengaruh sedikit terhadap nilai akurasi.
2. Jumlah langkah yang diambil berpengaruh besar terhadap nilai akurasi. Semakin besar maka semakin baik akurasinya.
3. Jumlah langkah yang diambil partikel *random walk* juga memengaruhi lamanya kerja sistem.
4. Hasil evaluasi sistem yang terbesar, 0,477 didapatkan oleh pengujian dengan menggunakan jumlah langkah 2000 pada percobaan pertama.

5.2 Saran

Berdasarkan kesimpulan dan hasil analisis tugas akhir ini, maka dapat disalurkan beberapa saran bagi yang tertarik untuk meneliti tugas akhir ini kedepannya, antara lain:

1. Untuk penelitian selanjutnya dapat dicoba untuk membuat model *graph* lainnya yang ada pada jurnal oleh Hughes dan Ramage [6] Karena pada penelitian ini hanya menggunakan satu model *graph* dari tiga yang tersedia.
2. Untuk penelitian selanjutnya dapat mencoba untuk mengganti nilai *convergence* untuk *stationary distribution*.
3. Menerapkan *random walks* secara penuh, terutama pada perhitungan *stationary distributionnya.*
4. Diharapkan pada penelitian selanjutnya menggunakan WordNet versi yang terbaru..

DAFTAR PUSTAKA

[4] A. T. Madhuri, M. M. Raghuwanshi dan L. Malik, "WordNet Based Method for Determining Semantic Sentence Similarity through Various Word Senses," Raisoni College of Engineering, Nagpur.

[2] C. Fellbaum, "WordNet: An Electronic Lexical Database," MIT Press, Cambridge, 1998.

[8] C. M. Grinstead dan J. L. Snell, "Markov Chains," dalam *Introduction to Probability: Second Revised Edition*, Hanover, American Mathematical Society, 2003, pp. 405-413.

[3] G. A. Miller, "WordNet: A Lexical Database for English," *Communications of the ACM,* vol. 38, no. 11, pp. 39-41, 1995.

[7] H. Rubenstein dan J. B. Goodenough, "Contextual Correlates of Synonymy," *Communications of the ACM,* vol. 8, no. 10, pp. 627-6333, 1965.

[9] K. Costello, "Random Walks on Directed Graphs," 2005.

[10] P. Berkhin, "A Survey on Pagerank Computing," *Internet Mathematics,* vol. 2, no. 1, pp. 73-120, 2005.

[6] T. Hughes dan D. Ramage, "Lexical Semantic Relatedness with Random Graph Walks," dalam *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, 2007.

[5] Y. Xu, "Random Walk on WordNet to Measure Lexical Semantic Relatedness," University of Minnesota, Duluth, 2011.

[1] Z. Zhang, A. L. Gentile dan F. Ciravegna, "Recent Advances in Methods of Lexical Semantic Relatedness," Cambridge University, Cambridge, 2012.