Peningkatan Sebuah PageRank Berbobot Untuk Menangani Persamaan Tautan Nol

Jeremy Andika

Fakultas Teknologi Informasi
Institut Teknologi Batam
Batam, Kepulauan Riau, Indonesia
2022002@student.iteba.ac.id

***Abstract*—Algoritma PageRank yang terkenal memanfaatkan struktur tautan untuk menghitung peringkat kualitas halaman. Pada dasarnya ia memberikan jumlah probabilitas yang sama untuk halaman tetangga dari sebuah halaman. Sebagai ekstensi, Algoritma PageRank berbobot yang diusulkan telah memberikan perbedaan bobot untuk link keluar dari halaman. Beberapa PageRank berbobot algoritma menggunakan persamaan antar halaman sebagai bobotnya. Dalam halaman web korea, kami menemukan bahwa terkadang kebetulan memiliki nilai nol untuk persamaan antar halaman dari halaman tetangga karena karakteristik bahasa tersebut. Proposal ini mengusulkan perbaikan algoritma PageRank berbobot yang dapat menangani hal seperti antar halaman bernilai nol. Metode yang diusulkan telah diterapkan menggunakan paradigma MapReduce untuk penanganan data yang besar, dan memiliki telah dievaluasi melalui halaman web Wikipedia Korea dan dibandingkan dengan dua metode lainnya.**

***Index Terms*—PageRank; PageRank Berbobot; Persamaan; MapReduce; TFIDF**

I. Pendahuluan

Belakangan ini, ketika orang ingin mengetahui sesuatu, kebanyakan dari mereka mencoba menemukannya di Internet. Mereka akan kewalahan jika terlalu banyak halaman yang diberikan seperti halaman yang relevan dalam pencarian di web. Di pengambilan informasi *(Information Retrival)*, peringkat telah menjadi salah satu masalah yang penting. Untuk memilah halaman berpengaruh dari yang dicari, berbagai peringkat algoritma telah diusulkan [1]–[11].

PageRank [1] merupakan salah satu algoritma peringkat terkenal yang menggunakan struktur link Web. Ini mengasumsikan bahwa seorang peselancar berjalan secara acak di atas halaman web dan mencoba untuk menentukan distribusi statis dari peselancar. Dengan penderitaan acak metafora, semakin banyak tautan yang dimiliki halaman, semakin tinggi peringkatnya. Di PageRank, yang menderita membuat jalan acak ke tetangga halaman dengan probabilitas yang sama. Kadang-kadang probabilitas yang sama ini tampaknya tidak masuk akal karena beberapa tautan terhubung ke halaman tetangga yang jauh lebih penting.

Untuk mengatasi situasi ini, algoritma PageRank berbobot [2], [3], [5] telah diusulkan. Mereka memperhitungkan baik distribusi jumlah in-link untuk node tetangga, jumlah kunjungan ke halaman tetangga, atau kesamaan antar halaman. Masing-masing memiliki pro dan kontra. Pembobotan berbasis kesamaan antar halaman terdengar baik untuk konten-peringkat berbasis.

Kami telah mencoba algoritma PageRank berbobot berbasis kesamaan antar halaman ke halaman Wikipedia bahasa Korea. Untuk menghitung kesamaan antar halaman, kami menggunakan model vektor [14]. Untuk mendapatkan representasi vektor untuk halaman, pertama-tama kami melakukan analisis morfologi untuk mengekstrak kata-kata. Berbeda dengan bahasa barat, kata benda dalam bahasa Korea menyampaikan informasi yang paling berarti. Karena karakteristik bahasa, kata benda diekstraksi untuk mengidentifikasi kata kunci. Kata kunci diidentifikasi menggunakan istilah frekuensi dan informasi frekuensi dokumen terbalik. Dengan kata kunci, representasi vektor untuk halaman diperoleh. Kemudian, kebetulan memiliki kesamaan nol ketika kesamaan antar halaman dihitung menggunakan produk dalam dari vektor tersebut. Sangat canggung untuk halaman tetangga yang tidak memiliki kesamaan. Makalah ini berkaitan dengan peningkatan PageRank berbobot berbasis kesamaan antar halaman untuk menangani kasus-kasus dengan tautan nol kesamaan.

Sisa proposal ini disusun sebagai berikut: Bagian II menyajikan PageRank dan variannya secara lebih rinci. Bagian III memperkenalkan peningkatan pada PageRank berbobot, dan Bagian IV menunjukkan beberapa hasil eksperimen untuk metode yang diusulkan. Kami menarik kenotasian di Bagian V.

II. Karya yang terkait

A. Algoritma PageRank

PageRank [1] adalah algoritma perbatasan yang memberi peringkat halaman dengan mengacu pada struktur tautan Web. PageRank memperlakukan halaman sebagai node dan hyperlink sebagai tepi grafik. Setiap node memiliki nilai Rank sendiri dan mendistribusikannya secara merata ke tetangganya. Distribusi berulang tanpa batas sampai semua nilai peringkat konvergen. Distribusi stasioner dari Peringkat dianggap sebagai skor Peringkat akhir halaman. Untuk mencegah peningkatan nilai Rank yang tidak terbatas, jumlah semua Rank dibatasi menjadi 1, dan juga untuk setiap nilai Rank tidak

boleh lebih besar dari 1. Nilai Rank r_j node j dihitung sebagai berikut :

$$r_j = \sum_{i \overrightarrow{out} j} \frac{r_i}{L_{out}(i)} \quad (1)$$

$$r_j = \sum r_i = 1 \quad (2)$$

Dimana $L_{out}(i)$ adalah jumlah out-link dari node i. Setiap notasi i mendistribusikan skor peringkatnya r_i secara merata notasi tetangganya j. Sebuah node j mengumpulkan semua nilai Rank yang dikirim dari node tetangga dan mengambil jumlah mereka sebagai nilai Rank baru r_j. Proses distribusi rank ini dapat dinyatakan dengan matriks ketetanggaan stokastik M dan vektor Rank r. Matriks M adalah matriks yang berdekatan untuk web yang mengkodekan hubungan ketetanggaan antara halaman dan distribusi stasioner dari nilai Peringkat. Nilai Rank baru $r^{(t+1)}$ dihitung sebagai berikut:

$$r^{(t+1)} = \mathrm{M}r^{(t)} \quad (3)$$

di mana $t + 1$ adalah langkah selanjutnya dari t . Ketika semua nilai peringkat konvergen, $r^{(t+1)}$ sangat mirip dengan $r^{(t)}$. Di sini, konvergen r sesuai dengan vektor eigen dengan nilai eigen 1 untuk matriks M.

B. Algoritma PageRank Berbobot

Penjelajah web sebenarnya tidak melakukan asal jalan seperti di PageRank. Untuk mengakomodasi karakteristik perilaku seperti itu, pembobotan Algoritma PageRank telah diusulkan yang memungkinkan penderitaan untuk membuat transisi probabilistik yang tidak merata ke tetangga halaman [2], [3], [5].

1) PageRank berbobot berdasarkan jumlah in-link dari halaman tetangga: Xing dan Ghorbani [2] mengusulkan algoritma PageRank berbobot yang memberikan lebih banyak porsi Peringkat ke halaman tetangga dengan lebih banyak tautan. Hal ini memang tidak cukup mencerminkan perilaku peselancar yang sebenarnya, karena hanya informasi struktur topologi yang digunakan.

2) PageRank Tertimbang berdasarkan Ukuran Kesamaan: Qiaoet al. [3] mengusulkan varian algoritma PageRank berbobot yang disebut SimRank. Mendistribusikan ke antarhalaman yang sama, untuk menerapkan metode ini dibutuhkan infrastruktur komputasi paralel terdistribusi seperti Hadoop MapReduce.

3) PageRank berbobot berdasarkan kunjungan tautan: Kumaret al. [5] memperkenalkan algoritma PageRank berbobot di mana node mendistribusikan lebih banyak nilai Peringkat ke tautan keluar yang lebih sering dikunjungi oleh pengguna. Algoritme ini memerlukan data klik tautan di seluruh Web. Oleh karena itu, ini sangat ideal, tetapi tidak mudah untuk menerapkannya pada skala Web.

C. Algoritma Hub dan Otoritas

Najork [7] mengambil pendekatan yang agak berbeda dari PageRank dan mengusulkan algoritma peringkat yang disebut HITS (pencarian topik yang diinduksi hyperlink). Sementara PageRank mengasumsikan gagasan penting satu dimensi untuk halaman, HITS memandang halaman penting memiliki dua rasa kepentingan. [12] Oleh karena itu, algoritme menetapkan dua skor untuk setiap halaman. Halaman tertentu berharga karena memberikan informasi tentang suatu topik. Halaman-halaman ini disebut otoritas. halaman lainnya adalah

berharga karena mereka memberi tahu Anda ke mana harus pergi untuk mencari tahu tentang topik itu. Halaman-halaman ini disebut hub. Algoritma mendefinisikan dua konsep dengan cara yang saling rekursif. Sebuah halaman dianggap sebagai hub yang baik jika terhubung ke otoritas yang baik. Sebuah halaman dianggap sebagai otoritas yang baik jika ditautkan oleh hub yang baik. [12]

D. Komputasi Terdistribusi dan Paralel

Pengambilan informasi dari penyimpanan data yang sangat besar, seperti Web dan penyimpanan data besar memerlukan infrastruktur komputasi yang menyimpan dan memproses data tersebut. Kami dapat menggunakan sistem superkomputer atau sistem komputasi terdistribusi dan paralel.

Hadoop [13] adalah infrastruktur komputasi yang baik yang dapat dibangun dengan biaya ekonomis. Ini adalah proyek Apache untuk platform komputasi terdistribusi yang menyediakan sistem file terdistribusi yang disebut HDFS (Hadoop Distributed File System) dan kerangka kerja komputasi paralel terdistribusi yang disebut MapReduce. Kerangka kerja MapReduce mengatur pekerjaan menjadi tugas Peta dan Mengurangi tugas. Data input dipartisi dan diproses oleh proses Map, dan hasil pemrosesannya dibentuk menjadi pasangan nilai kunci. Hasil tugas peta diacak menjadi Kurangi tugas sesuai dengan kuncinya. Reduce proses menggabungkan nilai dengan kunci yang sama, untuk mendapatkan hasil akhir. Kerangka kerja komputasi ini memungkinkan kami untuk menangani komputasi berat seperti komputasi kesamaan halaman berpasangan.

III. ALGORITMA YANG DIUSULKAN

PageRank berbobot berbasis kesamaan antar halaman berdasarkan kesamaan tidak dapat menangani situasi di mana kesamaan antar halaman adalah 0. Untuk mengatasi situasi ini, kami mengusulkan metode untuk menangani nol kesamaan antar halaman dan menyesuaikan bobot untuk distribusi nilai peringkat.

Kata benda- ekstraksi kata kunci berbasis seperti di halaman Korea terkadang menemukan kata kunci umum di antara halaman yang ditautkan. Meskipun gagasan yang melekat kesamaan antar halaman untuk memperkirakan frekuensi traversal link, situasi nol-kemiripan menghambat penerapan algoritma PageRank tertimbang.

Untuk meningkatkan penerapan PageRank berbobot, kami mengusulkan metode untuk menjamin beberapa bobot minimum dan menyesuaikan bobot. PageRank berbobot yang diusulkan bekerja sebagai berikut, yang pada dasarnya berperilaku dengan cara yang sama seperti algoritma Qiao et al. [3].

Berdasarkan ukuran kemiripan, bobot w_{ij} pada simpul i sampai j dihitung sebagai berikut:

$$\mathrm{w}_{ij} = \frac{s_{ij}}{\sum_{k \in L_{out}(i)} s_k} \quad (4)$$

di mana s_{ij} adalah kesamaan antara halaman i dan j, dan $L_{out}(i)$ menunjukkan halaman yang ditunjuk oleh halaman i.

Nilai peringkat r_j halaman j diperbarui, hingga peringkat semua nilai konvergen, sebagai berikut:

$$r_j = \sum i \in L_{in}(j)\beta w_{ij} r_i + (1-\beta)\frac{1}{N} \tag{5}$$

dimana β menunjukkan tingkat teleportasi seperti di PageRank, $L_{in}(j)$ adalah halaman yang mengarah ke halaman j, dan N adalah jumlah total halaman.

Untuk mengukur kemiripan antar halaman, kita menggunakan jarak kosinus antara vektor kata kunci yang elemennya merupakan nilai TFIDF lemma. Lemma diekstraksi dari halaman Korea menggunakan penganalisis morfologi Korea. TF (Term Frequency) adalah frekuensi lemma dalam satu halaman, dan IDF (Inverse document frequency) adalah frekuensi halaman yang mengandung lemma. [14] Untuk kata kunci di halaman, TFIDF-nya dihitung sebagai berikut:

$$\text{TFIDF} = \frac{TF}{\log\left(\frac{DF}{N}\right)} \tag{6}$$

Selanjutnya, kemiripan s_{ij} antara halaman i dan j dihitung dengan jarak cosinus antara vektor kata kunci dari TFIDF vlueus:

$$S_{ij} = \frac{K_i \cdot K_j}{|K_i|\,|K_j|} \tag{7}$$

di mana K_i adalah vektor kata kunci halaman i.

Menggunakan kesamaan dari Persamaan (7), bobot dihitung sebagai Persamaan (4). Namun, itu tidak mempertimbangkan situasi bahwa kesamaan interpage adalah nol. Oleh karena itu, kami mengusulkan sebuah algoritme, yang menanganinya dengan mengalokasikan kesamaan minimum ke tautan ke halaman dengan kesamaan nol.

$$\rho\frac{\min\left(s_{ij}\right)}{\sum_{L_{in}(i)} s_{ik}} = \alpha(1-\rho)ZR \tag{8}$$

di mana ρ adalah parameter yang disediakan pengguna untuk kesamaan nol, dan ZR adalah jumlah tautan kesamaan bukan nol.

persamaan (8) dikembangkan untuk membuat bobot kemiripan yang disesuaikan dengan nol kesamaan lebih kecil dari nilai-nilai kesamaan bukan nol. α ditentukan menurut Persamaan. (8), di mana kesamaan minimum dikendalikan oleh α. Seiring dengan kesamaan baru yang disesuaikan, bobot hanya perlu dihitung ulang seperti biasa. Pada akhirnya, nilai peringkat ditentukan seperti pada algoritma PageRank berbobot dengan Persamaan (4).

IV. Percobaan

Untuk mengevaluasi kinerjanya, kami menerapkan metode yang diusulkan untuk menentukan peringkat halaman web di Wikipedia bahasa Korea. Kami telah mengumpulkan sekitar 300.000 halaman dari Wikipedia bahasa Korea. Gambar 1 menunjukkan arsitektur sistem percobaan.

Seluruh halaman dari ko.wikipedia.org dirayapi dan disimpan ke dalam Hadoop HDFS. Halaman-halaman diurai menggunakan penganalisis morfologi Korea untuk mengekstrak lemma. Program Hadoop MapReduce dikembangkan untuk menghitung kemunculan kata-kata di halaman. Berdasarkan data jumlah kata, TFIDFs untuk lemma setiap halaman dihitung untuk menentukan kata kunci, dan vektor kata kunci dibangun untuk setiap halaman. Vektor kata kunci dinyatakan dengan nilai TFIDF yang dihitung. Kesamaan antara halaman tetangga dihitung menggunakan jarak cosinus. Akhirnya, nilai peringkat ditentukan oleh Persamaan (4) menggunakan bobot yang dihitung.

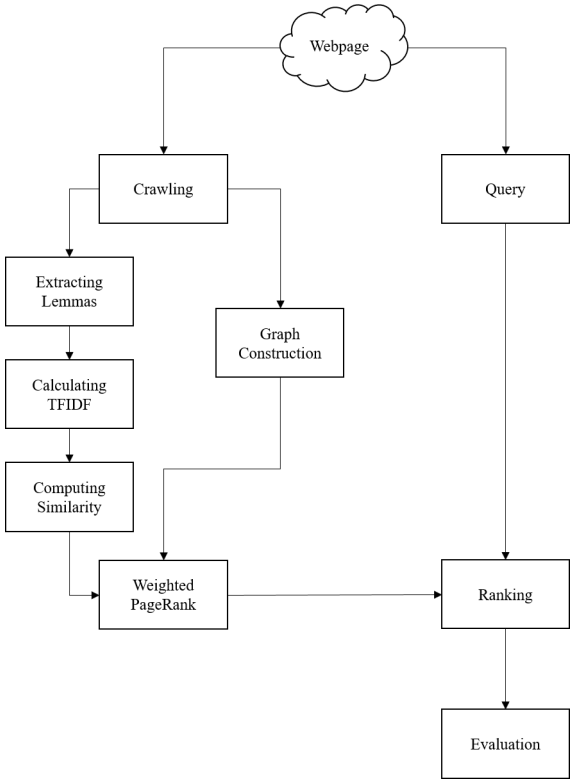

Fig. 1. Arsitektur Sistem Eksperimen

Dalam percobaan, kami menggunakan cluster Hadoop dari 5 node untuk menangani volume data yang besar. Semua tugas dari metode yang diusulkan telah diimplementasikan dalam program MapReduce. Kami melakukan percobaan 10 kali untuk 10 kata kunci yang dipilih secara acak dari Wikipedia dan menemukan halaman yang berisi kata kueri dan mengurutkannya menurut urutan penurunan nilai peringkatnya. Kemudian kami memilih 20 halaman teratas dan mengevaluasi relevansinya dengan 5 skala level. Kami menghitung Normalized Discounted Cumulative Gain(NDCG) [14] untuk 20 halaman teratas untuk setiap kueri. NDCG adalah metrik evaluasi yang digunakan untuk mengevaluasi kinerja mesin pencari web. Ini memberikan nilai dari 0,0 hingga 1,0, dan nilai 1,0 adalah peringkat ideal entitas. Halaman kebenaran dasar untuk kueri ditentukan dengan memilih halaman tautan keluar dari halaman kueri dalam urutan penurunan kesamaan.

Gambar 2 menunjukkan hasil eksperimen untuk PageRank

asli, Qias et al. [3] metode dan metode yang diusulkan dalam hal NDCG. Diamati bahwa metode yang diusulkan telah memberikan peningkatan sekitar 2 rata-rata di atas PageRank asli. Selama Qias et al. [3], metode yang diusulkan meningkat pada NDCG sekitar 1,3.

Dari percobaan, kami mengamati bahwa metode yang diusulkan menghasilkan hasil yang sedikit lebih baik secara rata-rata dibandingkan dengan metode lainnya.

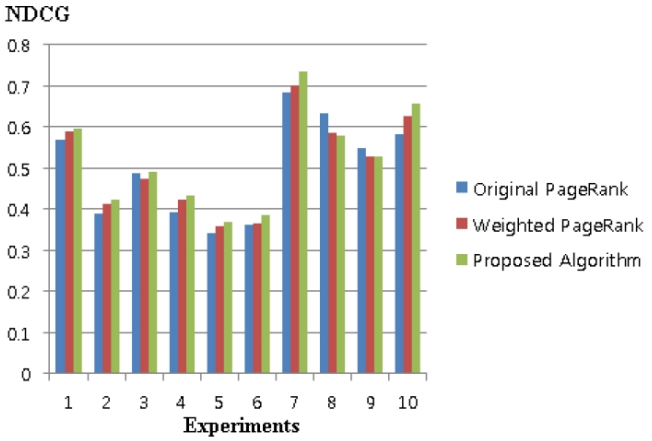

Fig. 2. Perbandingan PageRank asli dan Algoritma yang diusulkan

V. Kesimpulan

Dalam studi ini, kami menganalisis perilaku algoritma PageRank berbobot dan mengidentifikasi bahwa algoritma PageRank berbobot berbasis antar-kemiripan tidak dapat bekerja dengan baik dalam beberapa situasi, terutama, ketika kata kunci kata benda diekstraksi dari halaman Korea untuk perhitungan kesamaan. Varian baru dari algoritma PageRank berbobot diusulkan untuk menangani nol kesamaan antar halaman. Untuk pemrosesan data volume besar, algoritma yang diusulkan diimplementasikan dalam program MapReduce dan kumpulan data eksperimental diproses pada cluster Hadoop dari 5 node. Algoritma yang diusulkan telah diterapkan ke Wikipedia Korea untuk evaluasi kinerja. Dalam percobaan, kami menerapkan tiga algoritme: PageRank asli, PageRank berbobot Qias dkk. [3], dan metode yang diusulkan. Dari percobaan kami telah mengamati metode yang diusulkan mencapai beberapa perbaikan dalam hal NDCG dibandingkan metode yang dibandingkan.

References

[1] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, vol. 30, no. 1-7, pp. 107–117, 1998.

[2] W. Xing and A. Ghorbani, "Weighted pagerank algorithm," in *Proceedings. Second Annual Conference on Communication Networks and Services Research, 2004.* IEEE, 2004, pp. 305–314.

[3] S. Qiao, T. Li, H. Li, Y. Zhu, J. Peng, and J. Qiu, "Simrank: A page rank approach based on similarity measure," in *2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering*. IEEE, 2010, pp. 390–395.

[4] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Tech. Rep., 1999.

[5] K. Kumar and F. D. M. Abhaya, "Pagerank algorithm and its variations: A survey report," *IOSR Journal of Computer Engineering (IOSR-JCE)*, vol. 14, no. 1, pp. 38–45, 2013.

[6] N. Duhan, A. Sharma, and K. K. Bhatia, "Page ranking algorithms: a survey," in *2009 IEEE International Advance Computing Conference*. IEEE, 2009, pp. 1530–1537.

[7] M. A. Najork, "Comparing the effectiveness of hits and salsa," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, 2007, pp. 157–164.

[8] G. Kumar, N. Duhan, and A. Sharma, "Page ranking based on number of visits of links of web page," in *2011 2nd International Conference on Computer and Communication Technology (ICCCT-2011)*. IEEE, 2011, pp. 11–14.

[9] D. Nemirovsky and K. Avrachenkov, "Weighted pagerank: cluster-related weights," SAINT PETERSBURG STATE UNIV (RUSSIA), Tech. Rep., 2008.

[10] N. Tyagi and S. Sharma, "Weighted page rank algorithm based on number of visits of links of web page," *International Journal of Soft Computing and Engineering (IJSCE) ISSN*, pp. 2231–2307, 2012.

[11] T. H. Haveliwala, "Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search," *IEEE transactions on knowledge and data engineering*, vol. 15, no. 4, pp. 784–796, 2003.

[12] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of massive data sets*. Cambridge university press, 2020.

[13] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[14] P. Resnik, "Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language," *Journal of artificial intelligence research*, vol. 11, pp. 95–130, 1999.