Analisis dan Implementasi Graph Clustering pada Berita Digital Menggunakan Algoritma Star Clustering

Aufa Bil Ahdi P[1] , Kemas Rahmat Saleh W, S.T., M.Eng [2], Anisa Herdiani, S.T., M.T [3]
[1.2.3] *Teknik Informatika, Fakultas Teknik Informatika, Telkom University*
Jalan Telekomunikasi No.1, Dayeuh Kolot, Bandung 40257
aufa_ba@yahoo.co.id[1], bagidok3m45@gmail.com[2] , anisaherdiani@gmail.com[3]

Abstrak

Berkembangnya media Internet saat ini mempengaruhi penyebaran informasi melalui Internet menggunakan berbagai macam media. Salah satu bentuk pengembangan media informasi saat ini adalah banyaknya artikel berita *digital* yang tersebar secara *online*. Oleh karena banyaknya penyebaran berita *digital*, diperlukan pengelompokkan berita berdasarkan topik dan keterkaitan tertentu dengan menerapkan model *graph* untuk memetakan hubungan berita.

Model *graph* dipilih karena dapat memodelkan hubungan antar objek dan memberikan visualisasi yang mudah dipahami. Berita dapat direpresentasikan sebagai *node* dan dapat dihubungkan dengan *node* lain yang memiliki hubungan menggunakan *edge*. *Node* yang terbentuk akan di kelompokkan ke dalam sejumlah *cluster* menggunakan algoritma *star clustering*.

Algoritma *Star Clustering* merupakan salah satu algoritma pengelempokkan *graph* menjadi *subgraph/cluster* dengan keterkaitan tertentu. Algoritma *star clustering* dikenal sebagai algoritma yang mudah digunakan, dan memiliki tingkat akurasi yang cukup baik. Dalam tugas akhir ini didapatkan hasil pengujian penerapan algoritma *star clustering* pada berita digital dengan tingkat akurasi 80.98% untuk perbandingan dengan *clustering expert* dan menghasilkan 62.87129% cluster yang baik yaitu *cluster* yang memiliki nilai *intracluster* lebih besar daripada *intercluster*-nya.

Kata kunci : *graph, graph clustering, star clustering, subgraph*

1. Pendahuluan

Semenjak perkembangan teknologi Internet semakin besar di zaman modern ini. Berita digital sudah merupakan konsumsi umum bagi para pengguna Internet saat ini, karena merupakan sarana penyebaran informasi.

Berita digital yang tersebar pada Internet tidak semuanya dilengkapi dengan kategori oleh penulisnya sehingga mempersulit pencarian informasi mengenai suatu *event* dan kejadian tertentu. Untuk mengatasi permasalahan tersebut diperlukan sebuah sistem pengelompokan berita digital berdasarkan karakteristik tertentu dan konten berita secara otomatis.

Penerapan *graph* sangat cocok untuk kasus *clustering* berita digital karena pada model *graph* dapat menggambarkan sebuah berita digital menjadi sebuah *node* dan hubungan antara berita-berita yang saling terkait nantinya mudah digambarkan dengan hubungan *edge* antar *node* pada model *graph.*

Pengelompokkan atau *clustering* adalah suatu proses dimana objek-objek digolongkan ke dalam kelompok-kelompok yang disebut klaster [1]. Tujuan dari *clustering* yang baik adalah untuk membagi data menjadi klaster-klaster tertentu dimana masing-masing klaster memiliki kemiripan atau hubungan tertentu [2]

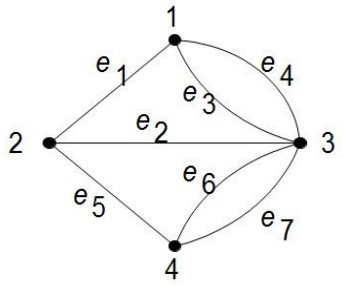
Dari sekian banyak metode *graph clustering* seperti MST *Clustering*, *Chameleon*, *Makarov Clustering*, dan *Star Clustering*, *Star Clustering* lah yang dianggap paling tepat, karena lebih mudah dimengerti, akurat, efisien dan menjamin kualitas dari *cluster* yang dihasilkan[3].

2. Landasan Teori

A. Teori Graph

Secara sistematis *graph* G didefinisikan sebagai pasangan himpunan (V,E) dimana V adalah himpunan *node* tidak kosong $\{v_1, v_2, v_{3,\ldots,}v_n\}$. Sedangkan E adalah himpunan sisi yang menghubungkan antara dua *node* $\{e_1,e_2, e_{3,\ldots,}e_n\}$ Sehingga dapat ditulis G = (V,E). Derajat(*degree*)

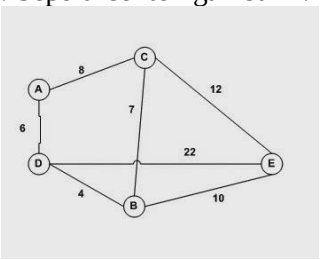
suatu simpul adalah jumlah sisi yang bersisian dengan simpul tersebut dilambangkan dengan *d*(v). Contoh pemodelan *graph* dapat dilihat pada gambar 2.1.



2.1 Graph G

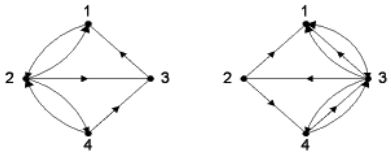
Graph dibagi menjadi beberapa jenis yaitu :

1. *graph* berbobot dimana graph memiliki nilai *edge* sebagai penghubung antar *node*. Seperti contoh gambar 2.2.



Gambar 2.2 Graph berbobot

2. *graph* berarah dimana *edge* sebuah *node* memiliki arah terhadap *node* lain seperti pada gambar 2.3.



Gambar 2.3 Graph berarah

3. *graph* tidak-berarah dimana *edge graph* tidak memiliki orientasi arah kepada node lain seperti contoh pada gambar 2.1.

B. Graph Clustering

Graph Clustering adalah metode pengelompokan sebuah graph menjadi beberapa kelompok/*cluster* berdasarkan keterkaitan tertentu. Pembuatan klaster dilakukan dengan cara memotong atau membuang *edge* yang tidak berguna berdasarkan ukuran tertentu[3]. Berdasarkan beberapa literatur, klaster pada *graph* disebut *community*[2].

Penerapan *clustering* pada *graph* tidak seperti *clustering* biasa dan memiliki metode tersendiri, contoh metode *clustering graph* adalah MST *Clustering*[3], *Chameleon*[3], *Makarov Clustering*[3], dan *Star Clustering*[3]

C. Star Clustering

Algoritma *star clustering* banyak digunakan karena cukup mudah dimengerti, sederhana, dan dapat membuat klaster yang tepat dengan lebih efisien. Sebuah *cluster* pada *star clustering* terdiri dari 1 *node star centre* dan *node*-node lain yang terhubung dengan *star* centre disebut dengan *satellite*[4]. Setiap *satellite node* boleh terhubung dengan lebih dari 1 *star centre*. Pada algoritma *star clustering* digunakan *threshold* yang ditentukan user untuk pembentukan *cluster* dimana *edge* yang bernilai dibawah *threshold* akan dihapus dari *graph*. Langkah-langkah dalam algoritma *star clustering* adalah sebagai berikut :

1. Setiap node dihitung jumlah degree masing-masing dan dilakukan pengurutan jumlah degree dari yang terbanyak ke yang paling sedikit.
2. Selanjutnya dilakukan penunjukan *star centre* dengan melakukan pemeriksaan terhadap setiap *node* yang memiliki jumlah *degree* paling banyak dan tidak terhubung dengan *star centre* yang lain.
3. Lakukan pengulangan proses 2 hingga semua *star centre* dalam *graph* tersebut didapatkan.
4. Selanjutnya dilakukan pembentukan *cluster* pada *graph*. Sebuah *cluster* pada algoritma *star clustering* adalah kumpulan dari sebuah *star centre* dan *node satellite*-nya, apabila sebuah *node satellite* terhubung lebih dari satu *star centre* maka *node* tersebut akan tergabung dengan *cluster star centre* yang memiliki bobot paling besar dengan *node satellite* tersebut.

D. Cosine Similarity

Cosine Similarity adalah sebuah metode perhitungan untuk menentukan bobot/nilai keterkaitan antara dua objek berdasarkan sudut kosinus antara dua objek tersebut[5]. Metode ini sering digunakan pada proses *text mining*. Rumus *cosine similarity* adalah :

Cosine Similarity (d1,d2) = $\frac{\vec{d1} \cdot \vec{d2}}{||\vec{d1}|| \times ||\vec{d2}||}$

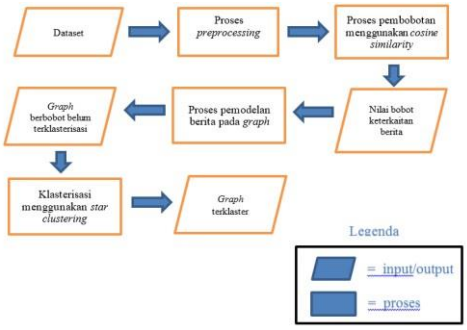
Apabila dua buah *vector* memiliki kemiripan mendekati nilai 1, maka dapat dikatakan terdapat kemiripan antara dua *vector* tersebut. Untuk menghitung nilai *cosine similarity*, terlebih dahulu harus mencari nilai TF, IDF, dan TFxIDF.

Gambar 2.4 Proses sebelum penghitungan *Cosine Similarity*

3. Gambaran Umum Sitem dan Skenario Pengujian

A. Gambaran Umum Sistem

Pada tugas akhir ini akan dibangun sistem *clustering graph* dengan memanfaatkan algoritma *Star Clustering*, dimana setiap *node* dalam *graph* tersebut merupakan permodelan dari sebuah berita digital berbahasa Indonesia. Gambar 3.1 adalah gambaran umum dari sistem yang dibentuk :



Gambar 3.1 Gambaran umum sistem

a. Dataset berupa file json berita digital berbahasa Indonesia yang diproses melalui tahap *preprocessing* sehingga menjadi kata-kata berupa token yang dapat dihitung keterkaitannya menggunakan *cosine similarity.*
b. Data hasil *preprocessing* berupa kumpulan token kata selanjutnya diproses menggunakan rumus *cosine similarity* sehingga menghasilkan nilai bobot antar dokumen berdasarkan keterkaitan antar berita.
c. Kemudian data berita dimodelkan kedalam *graph* yang belum ter-*cluster* dimana satu dokumen berita menjadi satu *node* dan bobot hasil *cosine similarity* menjadi nilai *edge* antar *node.*
d. *Graph* di-*clustering* menggunakan algoritma *star clustering* sehingga terbentuk kumpulan *cluster* berita berdasarkan keterkaitan antar berita.

B. Skenario Pengujian

a. Akan dilakukan pengamatan untuk mengetahui pengaruh jumlah berita dan penentuan *threshold* terhadap nilai *intracluster* dan *intercluster*, pada skenario ini dilakukan beberapa percobaan pengujian akurasi *intracluster*. Percobaan dilakukan dengan 4 jumlah berita yaitu 223, 167, 112, 56, dan 3 nilai *threshold* yaitu 0.05, 0.1, dan 0.15. Ketika proses pengujian berlangsung,
b. Akan dibandingkan hasil *clustering* dari sistem dengan hasil *clustering* yang dilakukan oleh *expert*. *Expert* yang melakukan *clustering* pada pengujian ini adalah seoran jurnalis/reporter media swasta yang merupakan lulusan program studi Ilmu Komunikasi Universitas Indonesia. Perhitungan akurasi kecocokan berita digunakan perhitungan nilai *F-measure*, *precision*, dan *recall* dari masing-masing *cluster* yang terbentuk.

4. Hasil dan Analisis Pengujian

A. Skenario 1

Berdasarkan hasil percobaan didapatkan persentase 62,87% untuk data uji utama yaitu 202 node dan *threshold* 0.1. Berdasarkan data pada tabel 4.1, didapatkan :

Table 4.1 Hasil uji skenario 1

Total Node	*Threshold*	Total *Cluster*	*Node* dgn nilai *intracluster* lebih baik	*Node* dgn nilai *intercluster* lebih baik	Persentase Cluster yang baik
54	0.05	10	43	12	79.62%
102	0.05	14	67	35	65.68%
150	0.05	11	72	78	48%
202	0.05	19	72	130	35.64%
54	0.1	16	42	12	77.778%
102	0.1	17	71	31	69.608%
150	0.1	25	107	43	71.333%
202	0.1	33	127	75	62.87129%
54	0.15	23	36	18	66.67%
102	0.15	28	76	26	74.5%
150	0.15	41	114	36	76%
202	0.15	54	153	49	75.742%
252	0.15	75	182	70	72.22%
297	0.15	84	215	82	72.39%

- Pada umumnya apabila semakin besar jumlah berita maka akan semakin kecil akurasi perbandingan nilai *intracluster* dan sebaliknya, tapi terdapat perbedaan pada *threshold* 0.15 yang menghasilkan akurasi semakin besar apabila jumlah berita semakin besar.
- Jumlah *node* dan jumlah *cluster* sangat mempengaruhi hasil akurasi. Apabila jumlah *node* semakin besar dan jumlah *cluster* semakin besar, maka nilai akurasi semakin besar dan apabila jumlah *nod* semakin besar dan jumlah *cluster* kecil, maka nilai akurasi akan semakin kecil.
- Pada *threshold* 0.15 terjadi hasil yang berbeda karena apabila jumlah berita bertambah banyak, *cluster* yang terbentuk sangat banyak dan memiliki banyak *cluster-cluster* yang hanya memiliki sedikit *node.*
- Pada *threshold* 0.15 didapatkan hasil *intracluster* yang baik dan konsisten yaitu diatas 70 % untuk berita diatas 112 *node.*
- Nilai *threshold* berpengaruh pada akurasi *intracluster* dimana semakin besar nilai *threshold* maka nilai akurasi semakin besar.

B. Skenario 2

Pada hasil pengujian skenario 2 dengan membandingkan kecocokan *cluster* hasil sistem dengan *cluster* hasil pengelompokkan dari *expert*, didapatkan akurasi dengan nilai 80.98%. Seperti yang dapat dilihat pada tabel 3.2.

Table 4.2 Hasil uji skenario 2

Nama *Cluster*	Node Expert	Node Sistem	Node Relevant	Recall	Precision	F Measure
Cluster 1	3	6	3	100%	50%	67%
Cluster 2	3	3	3	100%	100%	100%
Cluster 3	8	10	7	87.50%	70%	78%
Cluster 4	2	2	2	100%	100%	100%
Cluster 5	2	2	2	100%	100%	100%
Cluster 6	18	17	16	88.89%	94.12%	91%
Cluster 7	1	1	1	100%	100%	100%
Cluster 8	1	1	1	100%	100%	100%
Cluster 9	3	7	3	100%	42.86%	60%
Cluster 10	2	2	2	100%	100%	100%
Cluster 11	21	20	20	95.22%	100%	98%
Cluster 12	3	1	1	33.33%	100%	50%
Cluster 13	4	3	3	75%	100%	86%
Cluster 14	18	15	13	72.22%	86.67%	79%
Cluster 15	32	32	31	96.88%	96.88%	97%
Cluster 16	15	9	9	60%	100%	75%
Cluster 17	1	4	1	100%	25%	40%
Cluster 18	10	10	10	100%	100%	100%
Cluster 19	3	3	3	100.00%	100%	100%
Cluster 20	5	0	0	0%	0%	0%
Cluster 21	1	2	1	100%	50%	67%
Cluster 22	10	6	5	50%	83.33%	62%
Cluster 23	8	7	7	87.50%	100%	93%
Cluster 24	2	2	2	100%	100%	100%
Cluster 25	9	9	7	77.78%	77.78%	78%
Cluster 26	1	3	1	100%	33.33%	50%
Cluster 27	2	2	2	100%	100%	100%
Cluster 28	2	2	2	100%	100%	100%
Cluster 29	5	4	4	80%	100%	89%
Cluster 30	2	3	2	100%	66.67%	80%
Cluster 31	2	2	2	100%	100%	100%
Cluster 32	2	1	1	50%	100%	67%
Cluster 33	1	2	1	100%	50%	67%

Berdasarkan jumlah *cluster* yang dihasilkan, percobaan *expert* menghasilkan 33 *cluster* dan percobaan sistem menghasilkan 37 *cluster*, namun memiliki isi yang berbeda-beda. Perbedaan hasil *cluster* yang terjadi disebabkan karena ada *node* yang terkait berdasarkan kesamaan kata namun topik utama dari berita tersebut berbeda, seperti contohnya pada jenis berita "kriminal" dimana terdapat banyak jenis *cluster* berbeda yang berbeda-beda, namun ada beberapa *node* yang tergabung pada *cluster* yang salah karena lebih banyak menggandung kata yang sama dengan *star centre* pada *cluster* lain dibandingkan *star centre* yang seharusnya. Kesalahan ini banyak terjadi karena berita tersebut lebih banyak mengandung kata "korban" dibandingkan dengan topik khusus-nya sendiri seperti "mesum", "jambret", "curanmor", dll.

Faktor lain yang mempengaruhi nilai akurasi adalah proses *preprocessing* yang tidak sempurna menyebabkan ada beberapa data uji yang tidak sempurna diproses pada saat dilakukan *stemming* seperti yang sudah dijelaskan pada bab 3, sehingga luarannya tidak akurat 100% karena berpengaruh pada saat pembobotan antar

dokumen dan menyebabkan nilai bobot jadi sedikit berbeda dengan yang seharusnya.

5. Kesimpulan

1. Pemodelan berita digital dapat dimodelkan menggunakan model *graph* agar mudah dimengerti untuk melihat keterkaitan antar berita dengan cara menjadikan *edge* antar node untuk menggambarkan keterkaitan berita dan diberi nilai bobot masing-masing dan setiap node menggambarkan sebuah berita yang menyimpan konten berita, judul berita, pengarang, dll.
2. Penerapan algoritma *star clustering* dapat diterapkan dalam pengelompokkan berita berdasarkan keterkaitan isi beritanya dengan cara penunjukkan *star centre – star centre* yang merupakan *node* berita dengan *degree* terbesar dan kemudian mengelompokkan *cluster* berdasarkan keterkaitan dengan *star centre*-nya.
3. Algoritma *star clustering* dapat melakukan pengelompokkan berita digital berdasarkan keterkaitan isi beritanya dengan mudah dipahami, cepat, efektif, dan memiliki tingkat akurasi yang cukup baik.
4. Pengelompokkan berita menggunakan *graph clustering* mudah dimengerti karena menghasilkan visualisasi *cluster* yang baik, sehingga pengamat dapat melihat keterkaitan *cluster* lebih mudah berdasarkan pewarnaan antar *node* yang memiliki *cluster* sama dan juga melihat keterkaitan antar *cluster* dengan *edge* yang ada.
5. Hasil *clustering* pada algoritma *star clustering* sangat bergantung pada *star centre* yang dibentuk, karna acuan *cluster* yang dibentuk akan mengacu pada isi berita dari *star centre*, sehingga dalam sebuah *cluster* isi berita dari *satellite* mengacu pada *star centre*, sementara antar *star centre* memiliki isi berita yang sangat berbeda karena antar *star centre* tidak boleh saling berhubungan.
6. Keterkaitan antar berita pada penelitian ini didapatkan berdasarkan kemiripan kata-kata penyusun antara masing- masing berita, sehingga *cluster* yang dibentuk bukan berdasarkan topik yang umumnya didapatkan pada pengelompokkan berita pada media berita digital, melainkan berdasarkan kata-kata yang sering muncul dari berita tersebut.
7. Jumlah berita dan nilai *threshold* mempengaruhi akurasi perbandingan nilai *intracluster* dan *intercluster*, Namun pada percobaan kali ini didapatkan nilai *intracluster* yang tidak terlalu baik karena jumlah berita yang banyak.

6. Daftar Pusaka

[1] Feldman, Ronen and James Sanger. 2007. The Text Mining Handbook "*Advanced Approaches in Analyzing Unstructed Data".* University Press, Cambridge

[2] Schaeffer, Satu Elisa.2007. "*Graph Clustering".* Laboratory for Theoretical Computer Science, Helsinki University of Technology, Finland

[3] Wijaya, Derry Tanti. 2008. Graph, Clustering and Applications. Depatment of Computer Science National University of Singapore, Singapore.

[4] Aslam, J.A., Pelekhov, K., and Rus, D. 2004. The Star Clustering Algorithm for Information Organization.

[5] Sree, K. S. (2012). Clustering Based on Cosine Similarity Measure.