TINF-025
p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

STUDI AWAL PENGELOMPOKAN DATA TWITTER TOKOH POLITIK INDONESIA MENGGUNAKAN GRAPH CLUSTERING

Retnani Latifah
Teknik Informatika Fakultas Teknik, Universitas Muhammadiyah Jakarta, Jakarta
Jl Cempaka Putih Tengah 27, 10510 Jakarta Pusat
E-mail: retnani.latifah@ftumj.ac.id

ABSTRAK

Twitter sebagai sosial media yang populer, memiliki jumlah pengguna yang sangat besar. Pengelompokan pengguna Twitter menjadi penting untuk dilakukan. Salah satunya dapat menjadi strategi *marketing* suatu perusahaan dalam memasarkan produk yang digunakan. Pengelompokan dapat dilakukan dengan memanfaatkan fitur-fitur Twitter yang kemudian dimodelkan dalam bentuk *graph* sehingga dapat dilakukan *graph clustering*. Penelitian ini membandingkan tiga metode *graph clustering* yaitu *fastgreedy*, *walktrap* dan *leading eigenvector* dengan menggunakan 23000 *tweet* dari 96 akun politisi Indonesia. Dari hasil penelitian, nilai *purity* yang diperoleh adalah antara 0.7-0.8. Dengan nilai *purity* tertinggi diperoleh saat menggunakan algoritma *walktrap* dan *leading eigenvector* yaitu 0.833 dimana fitur Twitter yang digunakan adalah fitur *mention*s.

Kata kunci: Twitter, graph clustering, fastgreedy, walktrap, leading eigenvector, deteksi komunitas

ABSTRACT

As a popular media social, Twitter has huge amount of users. That's why it's important to cluster Twitter user. One of the reasons is for company to plan marketing strategy in advertising their product. Clustering in Twitter may use its feature which will be modeled into graph, and thus graph clustering can be conducted. This research compared three methods of graph clustering, fastgreedy, walktrap and leading eigenvector with 23000 tweets from 96 Indonesian politician accounts. The result showed that the purity of the clusters is around 0.7-0.8. Highest value was 0.833, performed by walktrap and leading eigenvector with mentions as feature.

Keywords : *Twitter, graph clustering, fastgreedy, walktrap, leading eigenvector, community detection*

PENDAHULUAN

Media sosial termasuk Twitter adalah suatu jaringan sosial sehingga Twitter dapat dikatakan sebagai struktur sosial yang terdiri dari simpul-simpul (*nodes*) ,yang berupa akun individu atau organisasi, dan sisi-sisi (*edges*) yang menghubungkan simpul-simpul dalam berbagai macam keterhubungan seperti pertemanan dan kekeluargaan (Tang & Liu, 2010).

Hubungan yang terjadi pada suatu jaringan, menyebabkan terbentuknya suatu kelompok atau komunitas. Komunitas dalam sudut pandang ilmu komputer adalah sekumpulan simpul yang memiliki lebih banyak sisi diantara mereka sendiri dibandingkan dengan jumlah sisi dengan simpul-simpul yang lain pada jaringan (Sadi, 2009). Deteksi komunitas merupakan pendekatan pokok dan penting untuk mendeteksi struktur komunitas yang ada di jaringan serta telah diterapkan pada berbagai domain penelitian seperti biologi, ilmu sosial, *online web*, jaringan sosial dan lain sebagainya (Zalmout & Ghanem, 2013).

Penemuan struktur komunitas bisa dianggap sebagai permasalahan *graph clustering* (Fortunato, 2010; Papadopoulos et al., 2011) walaupun Papadopoulos et al. (2011) menyebutkan bahwa terdapat perbedaan diantara keduanya yaitu perlunya mengetahui jumlah komunitas. Metode deteksi komunitas biasanya tidak perlu mengetahui jumlah komunitas dan menjadikan jumlah komunitas sebagai salah satu output yang dihasilkan. Meski demikian *graph clustering* masih dianggap sebagai istilah yang dapat

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

menggantikan deteksi komunitas karena banyak teknik dalam *graph clustering* yang digunakan untuk mendeteksi komunitas.

Pada tahun 2002, Newman dan Girvan mempublikasikan sebuah studi yang menjadi awal berkembangnya metode–metode deteksi komunitas dengan pendekatan *graph clustering* yang paling terkenal adalah pendekatan yang memanfaatkan fungsi *modularity*, yaitu pembagian spesifik dari jaringan menjadi komunitas dan merupakan properti dari jaringan yang mengukur bagus tidaknya pembagian komunitas. Fungsi ini pertama kali dikenalkan pada tahun 2004 (Fortunato, 2010).

Twitter merupakan jaringan sosial yang menarik untuk diteliti dengan berbagai macam fitur-fitur unik yang dimiliki seperti *following*, *follower*, *lists*, *mentions*, *retweets*, *hashtags* dan *links*. Salah satu penelitian deteksi komunitas pada Twitter adalah penelitian yang menggunakan deteksi komunitas untuk meningkatkan performa analisa sentimen yang dilakukan (Deitrick *et al.*, 2013). Pada penelitian tersebut, peneliti menggunakan algoritma *walktrap* pada jaringan Twitter yang dibangun dari hubungan *following* dan *follower*.

Penelitian yang lain, ada yang menggunakan fitur *lists* untuk memperoleh kelompok topik yang ada di jaringan Twitter (Bhattacharya *et al.*, 2014) dan ada pula yang menggunakan fitur *mentions*, *retweets*, *hashtags* dan *links* sebagai matriks kemiripan untuk membangun graf yang kemudian dikelompokkan menggunakan *fastgreedy* (Zalmout & Ghanem, 2013).

Penelitian pada artkel ini merupakan studi awal dalam melakukan pengelompokan data Twitter tokoh politik Indonesia dimana fitur yang digunakan adalah fitur *mentions*, *hashtags* dan *links*, seperti yang dilakukan oleh Zalmout & Ghanem (2013). Pada artikel ini, metode *clustering* yang digunakan tidak hanya *fastgreedy* tapi membandingkan tiga metode *graph clustering* untuk mengetahui metode mana yang memiliki evaluasi terbaik untuk mendeteksi komunitas pada data akun tokoh politik Indonesia yang digunakan.

METODE

Data

Data yang digunakan pada penelitian ini adalah data akun tokoh politik Indonesia yang diambil dengan menggunakan library Twitter4J pada Mei 2015. Jumlah *tweet* yang diperoleh adalah 23.004 *tweet* dari 96 akun tokoh politik yang diperoleh. Data yang diambil dari setiap *tweet* adalah id akun, id *tweet*, daftar *mentions*, daftar *hashtags* dan daftar *links*.

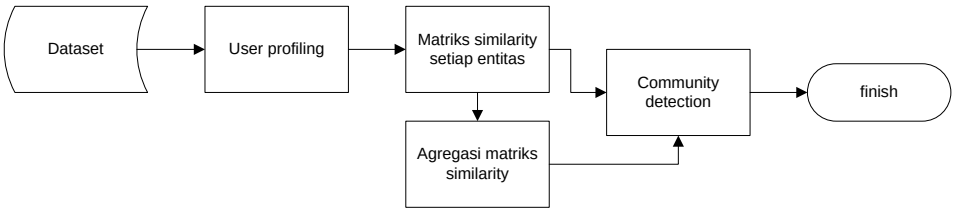
Sebagai evaluasi, data yang diperoleh dibagi menjadi dua yaitu tokoh politik dari partai Gerindra dan tokoh politik dari partai lain. Hal ini dikarenakan partai Gerindra sudah membuat *lists* akun-akun tokoh politik dari partainya sendiri sedangkan partai lain belum melakukan hal tersebut sehingga diambil beberapa tokoh dari masing-masing partai dan diasumsikan sebagai satu kelompok.

Tahapan penelitian

Dari kumpulan *tweet* tersebut, hal pertama yang dilakukan adalah melakukan *user profiling* yaitu mengelompokkan *tweet* beserta entitas-entitasnya berdasarkan *user* yang melakukan *tweet*. Setelah *user profiling* selesai dilakukan maka tahap selanjutnya adalah mengambil entitas *mentions*, *hashtags* dan *links*. Entitas *hashtags* dan *links* digunakan untuk membangun matriks kemiripan dari segi topik. Sedangkan *mentions* digunakan untuk membangun matriks yang berisi kemiripan interaksi dengan *user* yang lain. Matriks *mentions* ini tidak hanya dari *mentions* dan *replies* tapi juga dari *retweets*. Skema implementasi penelitian ini ditunjukkan pada gambar 1.

Matriks masing-masing entitias dibangun dengan cara melakukan penambahan nilai matriks setiap kali dua *user* yang berbeda memiliki entitas yang sama. Matriks dari masing – masing entitas kemudian diagregasi sehingga terbentuk matriks baru yang merepresentasikan keterhubungan dua *user* dari segi topik dan interaksi. Implementasi matriks ini dibangun dengan menggunakan bahasa pemrograman java yang *output* matriksnya disimpan dalam bentuk *csv* untuk kemudian diproses di tahap kedua yaitu deteksi komunitas.

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek



Gambar 1 Skema Implementasi

Deteksi komunitas dilakukan dengan menggunakan *igraph tool* yang dijalankan di R *software*. Matriks akan dijalankan adalah algoritma *fastgreedy*, *walktrap* dan *leading eigenvector*. Hasil deteksi komunitas dari ketiga metode ini akan dibandingkan dan dianalisis. Pemilihan ketiga metode ini adalah karena ketiganya merupakan algoritma *graph clustering* dengan teknik pengklusteran yang berbeda satu sama lain.

Algoritma *fastgreedy* dan *leading eigenvector* adalah algoritma yang menggunakan fungsi *modularity* dimana *modularity* didefinisikan sebagai :

$$Q = \frac{1}{2m}\sum_{l=1}^{o}\sum_{i\epsilon C_l, j\epsilon C_l} A_{ij} - \frac{k_i k_j}{2m} \qquad (1)$$

Dimana $\frac{1}{2m}$ digunakan untuk normalisasi nilai *modularity* agar berada diantara -1 dan 1, m adalah jumlah sisi pada graf. Sedangkan $\sum_{i\epsilon C_l, j\epsilon C_l} A_{ij} - \frac{k_i k_j}{2m}$ menunjukkan kekuatan dari komunitas dengan k_i adalah banyaknya sisi yang terhubung dengan simpul i, A_{ij} adalah banyaknya sisi antara simpul i dan j, dan $\frac{k_i k_j}{2m}$ adalah jumlah sisi yang diharapkan dari semua pasangan simpul.

Algoritma *fastgreedy* mempercepat proses komputasi dan menghemat memori karena algoritma ini melakukan *update* hanya pada sel matriks yang memiliki nilai perubahan *modularity* diantara dua simpul. Hal ini didasarkan pada asumsi bahwa dua komunitas tanpa sisi tidak akan pernah bergabung (Clauset *et al.*, 2004).

Algoritma *leading eigenvector* adalah algoritma yang mengoptimasi *modularity*. Algoritma ini memiliki matriks khusus yang disebut sebagai matriks *modularity*. Matriks *modularity* ini memiliki komponen yang berupa peluang adanya sisi diantara dua simpul. Algoritma ini menghitung *eigenvector* dari matriks *modularity* dengan nilai *eigen* positif terbesar dan kemudian memisahkan simpul menjadi dua komunitas yang berbeda berdasarkan nilai positif-negatif dari elemen *eigenvector* yang terkait. Jika tidak ada nilai eigen positif maka jaringan tidak memiliki struktur komunitas (Newman, 2006).

Algoritma *walktrap* menggunakan sebuah perhitungan jarak antara dua simpul, r, yang dihitung menggunakan probabilitas *random walk* bergerak dari satu simpul ke simpul lain setelah t langkah (P_{ij}^t). Nilai P_{ij}^t tinggi jika v_i dan v_j berada pada satu komunitas, namun bukan berarti dengan P_{ij}^t yang tinggi maka v_i dan v_j berada pada satu komunitas. Probabilitas P_{ij}^t dipengaruhi oleh *degree* d_j sehingga *walker* cenderung lebih memilih simpul dengan *degree* yang tinggi (Pons & Latapy, 2005).

Evaluasi komunitas yang terdeteksi dilakukan dengan menggunakan *purity* yang merupakan metode evaluasi yang bagus untuk mendeteksi dan menurunkan elemen – elemen *noise* didalam kluster. Rumus evaluasinya adalah sebagai berikut (Khorasgani *et al.*, 2010):

$$\text{purity}\,(\mathrm{R},\mathrm{G}) = \frac{1}{\mathrm{n}} \times \sum_{\mathrm{j}} \max_{\mathrm{i}} \left|\mathrm{R_j} \cap \mathrm{G_i}\right| \qquad (2)$$

Dimana R adalah himpunan *node* pada komunitas hasil deteksi komunitas, G adalah himpunan *node* pada komunitas yang sebenarnya, n adalah jumlah keseluruhan *node*.

Evaluasi yang dilakukan adalah membandingkan nilai *purity* dari komunitas yang terbentuk dari empat matriks yaitu matriks *mentions*, matriks *hashtags*, matriks *links* dan matriks agregasi dari ketiga fitur.

HASIL DAN PEMBAHASAN

Jumlah Komunitas

Dari hasil percobaan diketahui bahwa matriks *mentions* mengelompokkan data menjadi tepat dua komunitas. Sedangkan hasil yang berbeda diperoleh dengan menggunakan matriks *hashtags* dan *links*, dimana semua

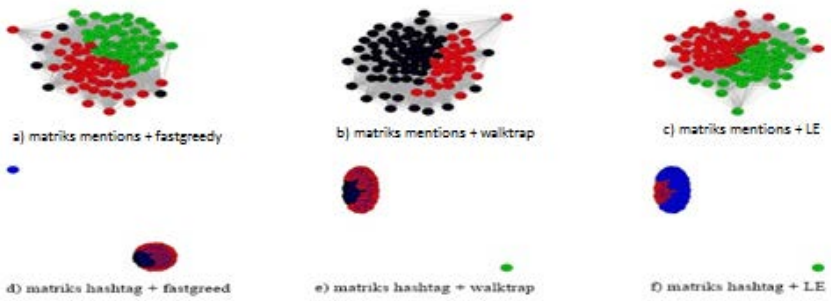
p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

algoritma mendeteksi lebih dari dua komunitas. Meski jumlah komunitas yang terdeteksi adalah lebih dari dua, akan tetapi hanya dua komunitas yang memiliki jumlah anggota dalam jumlah yang banyak. Komunitas lain hanya memiliki satu atau dua anggpta yang menunjukkan adanya pencilan. Hal ini terjadi karena ada tokoh politik yang melakukan *tweet* dengan meggunakan *hashtags* dan *links* yang tidak berhubungan dengan kegiatan politik atau tokoh politik lain tidak memiliki ketertarikan dengan topik *hashtags* dan *links* tersebut. Hasil jumlah komunitas yang terdeteksi dapat dilihat pada tabel 1 sedangkan ilustrasi persebaran anggota di masing-masing komunitas dapat dilihat pada gambar 2 dan 3 .

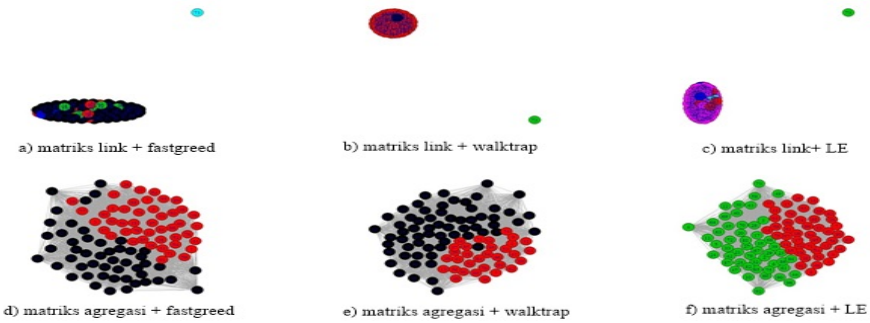
Tabel 1. Jumlah Komunitas yang Terdeteksi

Nama matriks	Jumlah komunitas yang terbentuk		
	fastgreedy	walktrap	eigenvector
mentions	2	2	2
hashtags	3	4	3
links	3	5	5
agregasi	2	2	2

Perlu diperhatikan bahwa matriks agregasi menyebabkan jumlah komunitas yang terdeteksi menjadi dua. Hal ini berarti tokoh politik yang awalnya merupakan pencilan sudah tidak menjadi pencilan lagi karena memiliki kemiripan dengan tokoh yang lain dari fitur *mentions*.



Gambar 2. Ilustrasi hasil komunitas yang terdeteksi dari matriks mentions dan matriks hashtags dengan metode fastgreedy, walktrap dan leading eigenvector



Gambar 3. Ilustrasi hasil komunitas yang terdeteksi dari matriks mentions dan matriks hashtags dengan metode fastgreedy, walktrap dan leading eigenvector

Evaluasi Komunitas

Dari hasil percobaan ditemukan bahwa matriks *mentions*, memiliki nilai *purity* yang lebih baik dibandingkan dengan menggunakan matriks *hashtags*, matriks *links* ataupun agregasinya. Perbedaan nilai *purity* yang paling sedikit adalah 2%, yaitu antara matriks *mentions* dan matriks agregasi dengan menggunakan *leading eigenvector*. Sedangkan selisih paling besar adalah 39% antara matriks *mentions* dan matriks *links* dengan menggunakan *leading eigenvector*. Tabel 2 menunjukkan nilai *purity* hasil deteksi komunitas setiap matriks dengan masing-masing metode.

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

Tabel 2. Nilai *Purity* Komunitas yang Terdeteksi

matriks	fastgreedy	waltrap	eigenvector
mentions	0.7396	**0.8333**	**0.8333**
hashtags	0.7188	0.6875	0.7188
links	0.4583	0.5417	0.4479
agregasi	0.8229	0.7188	0.8125

Dari keempat matriks terlihat bahwa matriks *mentions* dan matriks agregasi menghasilkan komunitas yang memiliki nilai *purity* yang memuaskan yaitu diatas 0.7 dengan menggunakan ketiga algoritma. Hal ini menunjukkan bahwa fitur interaksi dengan *user* yang lain memiliki pengaruh yang lebih besar dalam menentukan komunitas dari data tokoh politik yang digunakan dibandingkan dengan menggunakan fitur kemiripan topik. Hal ini mungkin terjadi karena tokoh politik dapat melakukan *tweet* yang tidak berhubungan dengan dunia politik yang mana tokoh politik yang lain tidak memiliki ketertarikan. Berbeda dengan interaksi, dimana tokoh politik cenderung berinteraksi dengan orang-orang yang berada di komunitas yang sama.

Fitur *links* memiliki *purity* yang paling rendah diantara ketiga matriks karena jarang ada tokoh politik yang menggunakan *links* dalam *tweet*-nya sehingga kemiripan *link*s juga sulit untuk dibangun.

Agregasi semua fitur menyebabkan peningkatkan nilai *purity* jika dibandingkan dengan hanya menggunakan *hashtags* atau *links* saja. Dengan menggunakan algoritma *fastgreedy*, nilai *purity*-nya juga naik sebesar 8.3% dibanding dengan matriks *mentions*. Hal ini menunjukkan bahwa menggunakan semua fitur dapat memberikan lebih banyak kemiripan sehingga meningkatkan evaluasi. Meskipun saat digunakan algoritma *walktrap* dan *leading eigenvector*, nilai *purity* justru mengalami penurunan sebesar 11.5% dan 2.1%. Hal ini dikarenakan saat dilakukan agregasi, tokoh politik yang awalnya sudah ditempatkan di komunitas dengan member yang sesuai mengalami perpindahan komunitas karena pengaruh kemiripan topik dari sisi *hashtags* dan *links*. Hal ini juga menunjukkan bahwa agregasi matriks juga dapat menyebabkan penurunan kemiripan, sesuai dengan teknik *clustering* yang digunakan.

Dari sisi algoritma, dapat dilihat bahwa perbedaan nilai *purity* ketiga algoritma tidak terlalu jauh untuk setiap matriks. Dapat dilihat bahwa secara umum, *fastgreedy* dan *leading eigenvector* memiliki kemiripan nilai purity untuk semua matriks. Hanya matriks *mentions* yang memberikan selisih yang cukup signifikan, yaitu 9.37%. Hal ini menunjukkan bahwa penggunaan algoritma dengan optimasi *modularity* memiliki hasil yang sedikit lebih baik dibandingkan dengan algoritma dengan *random walk* karena *modularity* menunjukkan nilai seberapa baik suatu komunitas pada jaringan.

SIMPULAN DAN SARAN

Untuk melakukan deteksi komunitas pada data tokoh politik Indonesia yang digunakan pada penelitian ini, fitur yang paling baik adalah menggunakan fitur *mentions* atau interaksi antara *user* yang satu dengan user yang lain. Hal ini ditunjukkan dengan nilai *purity* tertinggi yang diperoleh adalah dengan menggunakan matriks *mentions* pada algoritma *walktrap* dan *leading eigenvector* yaitu sebesar 0.8333.

Secara umum algoritma *fastgreedy* dan *leading eigenvector* memiliki kinerja yang cukup baik dengan selisih keduanya tidak terlalu jauh. Kedua algoritma adalah merupakan algoritma optimasi *modularity* sehingga dapat dikatakan pada penelitian ini algoritma dengan mengoptimasi *modularity* memiliki kinerja yang sedikit lebih baik dibandingkan dengan algoritma *walktrap* yang menggunakan *random walk*.

Karena penelitian ini masih dalam tahap studi awal, masih banyak hal yang perlu ditambahkan seperti jumlah data, keseimbangan data, jumlah kelompok dari *ground truth*, penggunaan fitur-fitur lain untuk pembangunan matriks serta penggunaan metode-metode deteksi komunitas lain untuk perbandingan.

DAFTAR PUSTAKA

Bhattacharya, P., Zafar, M.B., Gummadi, K.P., Ghosh, S., Kulshrestha, J., Mondal, M. & Ganguly, N. 2014. Deep Twitter Diving□: Exploring Topical Groups in Microblogs at Scale. *CSCW '14 Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, (February): 197–210.

Clauset, A., Newman, M.E.J. & Moore, C. 2004. Finding community structure in very large networks. *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics*, 70(6 Pt

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

2): 066111. Tersedia di http://www.ncbi.nlm.nih.gov/pubmed/15697438.

Deitrick, W., Valyou, B., Jones, W., Timian, J. & Hu, W. 2013. Enhancing Sentiment Analysis on Twitter Using Community Detection. *Communications and Network*, (August): 192–197.

Fortunato, S. 2010. Community detection in graphs. *Physics Reporrt*, 75–174.

Khorasgani, R.R., Chen, J. & Zaïane, O.R. 2010. Top Leaders Community Detection Approach in Information Networks. *4th SNA-KDDWorkshop on Social Network Mining and Analysis, Washington D.C*, (July).

Newman, M.E.J. 2006. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 74(3).

Papadopoulos, S., Kompatsiaris, Y., Vakali, A. & Spyridonos, P. 2011. Community detection in Social Media. *Data Mining and Knowledge Discovery*, 24(3): 515–554. Tersedia di http://link.springer.com/10.1007/s10618-011-0224-z [Accessed 2 October 2014].

Pons, P. & Latapy, M. 2005. Computing communities in large networks using random walks. 20. Tersedia di http://arxiv.org/abs/physics/0512106.

Sadi, S. 2009. Community Detection Using Ant Colony Optimization Techniques. *15th International Conference on Soft Computing, MENDEL 2009*, 206–213.

Tang, L. & Liu, H. 2010. *Community Detection and Mining in Social Media*. *Synthesis Lectures on Data Mining and Knowledge Discovery*, Tersedia di http://www.morganclaypool.com/doi/abs/10.2200/S00298ED1V01Y201009DMK003.

Zalmout, N. & Ghanem, M. 2013. Multidimensional community detection in Twitter. *8th International Conference for Internet Technology and Secured Transactions (ICITST-2013)*, 83–88. Tersedia di http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6750167.