Jurnal SimanteC

Vol. 5, No. 2 Juni 2016 **ISSN 2088-2130;e-ISSN 2502-4884**

PEMBOBOTAN DINAMIS BERBASIS INFORMATION GAIN PADA TEMU KEMBALI INFORMASI

Hasan Dwi Cahyono[1)]

Prodi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Sebelas Maret
Surakarta, Indonesia
E-mail : [1)]hasandc@staff.uns.ac.id

ABSTRAK

Meningkatnya konten multimedia dan teks seiring berkembangnya internet mengakibatkan Temu Kembali Informasi (TKI) menjadi topik yang menarik dikembangkan. Tingkat heterogensi informasi yang tinggi serta distorsi informasi tekstual menjadi tantangan yang menarik untuk dipecahkan. TKI berbasis pencarian tekstual berbasis menggunakan *weighted tree simlarity* (W-Tree) terbukti dapat mengatasi perbedaan heterogensi informasi tekstual dengan memecah informasi kedalam cabang-cabang informasi. Namun, menentukan bobot setiap cabang dari tree menjadi sebuah kendala dimana setiap cabang belum tentu memberikan kontribusi informasi yang tepat. Hal ini dikarenakan cabang-cabang informasi tekstual tersebut justru memberikan distorsi, atau bahkan memberikan noise terhadap cabang-cabang tree lainnya. Oleh karena itu, dalam penelitian ini diusulkan metode dengan pembobotan *tree* dinamis menggunakan *Information Gain* (IG) dan *cosine similarity*. Pada tahap pertama, dilakukan proses pembentukan W-Tree dari database dengan W-Tree dari *query user* serta dilakukan pencocokan dengan cosine similarity dimana IG digunakan untuk memilah dan mengatur bobot informasi yang akan digunakan oleh W-Tree. Sistem akan menampilkan keluaran berupa daftar dokumen beserta nilai kemiripannya. Dari percobaan pada dataset ImageCLEF 2011 sebanyak 9516 dokumen, pencarian tekstual berbasis *cosine similarity* dan W-Tree dengan pembobotan dinamis berbasis IG mampu meningkatkan f-measure 73% dibanding pencarian tekstual tanpa mempertimbangkan nilai IG.

Kata kunci: Temu Kembali Teks, *Cosine similarity*, W-tree, *information gain.*

ABSTRACT

Rapid increasing of multimedia content over a massive development of internet makes information retrieval (IR) become an interesting topic to be investigated. The level of data heterogeneity and the distortion of textual information remain widely open to solve. IR with a textual search using Weighted Tree Similarity (W-Tree) proved able to overcome differences of textual information heterogeneity by breaking down such information into branches of information. However, determining the weight of each branch becomes an obstacle since they do not always give a proper contribution to the right information; meanwhile in a particular condition, some branches of textual information give distortion, or even provide noise to the other branches. Stated thus, the method of dynamic w-tree using Information Gain (IG) is proposed. For the first level, to form a process of W-Tree based on user's queries with W-Tree database and to use cosine similarity to conduct a document matching while the second level is employing IG to sort and arrange the weight information to be utilized by the W-Tree. The system will display a list of papers and their output value of similarity. From the experiments on as many as 9516 Image CLEF 2011 datasets, textual search based cosine similarity, and W-Tree with a dynamic weighting based IG are able to increase the f-measure of 73% compared to textual without considering their IG values.

Keywords: *Information retrieval, Cosine Similarity, W-tree, information gain.*

PENDAHULUAN

Perkembangan informasi menjadikan bervariasinya jenis dan kuantitas bertambah setiap harinya di dunia online. Sehingga, pencarian informasi yang lebih akurat menjadi dorongan agar dapat memberikan hasil yang lebih memuaskan.

Diantara metode pencarian informasi tersebut, salah satu yang dapat memberikan hasil yang memuaskan adalah pencarian tekstual. Berbagai macam mesin pencari popular telah menerapkan model pencarian ini dan pencarian tersebut sering kali memberikan hasil yang kurang memuaskan. Hal ini dikarenakan semua cabang informasi yang ada dianggap memiliki nilai yang sama. Padahal, beberapa cabang informasi tertentu menjadi inti dan bahkan memiliki kandungan informasi lebih besar dibanding cabang lainnya.

Salah satu cara untuk memilah pencarian berdasarkan cabang adalah *Weighted Tree Similarity*. Metode tersebut terbukti mampu membagi cabang informasi dengan bobot masing-masing cabangnya [1]. Namun, penentuan bobot masing-masing cabang tersebut menjadi sebuah permasalahan dikarenakan tinggi rendahnya nilai kemiripan sangat bergantung dengan bobot yang ditentukan dan sering kali harus ditentukan secara manual.

Oleh karena itu, dalam penelitian ini diusulkan temu kembali teks dengan penerapan pengaturan bobot tree secara dinamis pada pencarian tekstual. Adapun model pembobotan dinamis yang akan digunakan adalah *Information Gain* (IG) yang telah terbukti mampu melakukan seleksi fitur dengan menilai kontribusi elemen terhadap sistem secara keseluruhan. Serta *Cosine similarity* yang terbukti mampu menilai kemiripan antara dua informasi dengan sumber berbeda.

METODE

Information Gain

Information Gain (IG) menggunakan disorder degree untuk mengukur entropi sebuah sistem [2]. Sehingga, menjadi salah satu algoritma seleksi fitur yang tepat untuk diuji coba. Salah satu penerapan IG terjadi pada seleksi label[3]. Rumus untuk menghitung IG ditunjukkan pada Persamaan 1.

$$IG(C \mid E) = H(C) - H(C \mid E), \qquad (1)$$

dimana IG(C|E) adalah information gain dari cabang atau atribut E, H(C) adalah sistem entropi dan H(C|E) adalah entropi relatif terhadap sistem ketika nilai cabang dari E diketahui.

Sistem entropi mengindikasikan disorder degree-nya dengan Persamaan 2.

$$H(C) = -\sum_{i=1}^{|C|} p(c_i)\log_2 p(c_i), \qquad (2)$$

dimana p(ci) adalah nilai probabilitas terhadap i. Berikut adalah persamaan 3 untuk entropi relatif.

$$H(C \mid E) = \sum_{i=1}^{|E|} p(e_j)\left(-\sum_{i=1}^{|C|} p(c_i \mid e_j)\log_2 p(c_i \mid e_j)\right), \qquad (3)$$

dimana $p(e_i)$ adalah nilai probabilitas i terhadap atribut e, dan $p(c_i|e_j)$ adalah probabilitas c_i terhadap e_j.

Cosine Similarity

Cosine similarity adalah sebuah pengukurang dua buah vektor yang mengukur sudut cosinus antara vektor-vektor tersebut [4]. Nilai *cosine* dihitung sebagai normalisasi *dot-product* dari dua vektor a dan b dimana nilai positifnya berada antara 0 hingga 1. Normalisasi tersebut biasanya menggunakan *Euclidean*. Nilai dot-product dihitung dengan Persamaan 4.

$$a.b = \sum_{i=0}^{n} a_i b_i, \qquad (4)$$

sedangkan Persamaan 5 utntuk normalisasinya :

$$\|x\| = \sqrt{x.x}, \quad (5)$$

dan selanjutnya persamaan 6 untuk *cosine similarity* didefinisikan sebagai:

$$cosine\ (a,b) = \frac{a.b}{\|a\| \times \|b\|}, \quad (6)$$

dimana berdasarkan persamaan (4) dan (5), persamaan 7 untuk *cosine similarity* menjadi:

$$cosine\ (a,b) = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}}. \quad (7)$$

Weighted Tree Similarity

Struktur *Weighted Tree* memperkenalkan konsep *node* berlabel, *arc* berlabel, dan *arc* berbobot yang merepresentasikan relasi *parent – child* dari suatu atribut produk / jasa. Informasi semantik dikandung tidak hanya pada label *node* tetapi juga pada label *arc.*

Sedangkan bobot *arc* merepresentasikan tingkat kepentingan *(importance)* dari suatu *arc* (atribut produk/jasa) [6]. Contoh penerapan sebagaimana pada

Gambar 1. Nilai kemiripan berada pada 0 hingga 1.

Gambaran Dataset

Pengujian

Dalam penelitian ini, data yang digunakan hanya pada informasi tekstual dari dataset CLEF (*Cross-Language Evaluation Forum*) yang berisi informasi teks dan citra pada tema khusus yaitu *Wikipedia Retrieval*.

Adapun bagian yang digunakan untuk pengujian adalah description, caption, dan juga comment yang berada pada anotasi bahasa Inggris (en), bahasa Jerman (de), dan bahasa Perancis (fr) saja. Untuk bagian yang lain tidak dimasukkan dalam pengujian.

Arsitektur Sistem

Pada sistem yang diajukan ada 3 tahapan utama yaitu preprocessing, perhitungan bobot, dan evaluasi hasil. Proses tersebut seperti pada Gambar 2.

Preprocessing

Untuk dapat menganalisis hasil, akan dilakukan tiga percobaan pembobotan dinamis *W-Tree*. Pertama pembobotan berdasarkan cabang dengan nilai IG sebagai C1. Kedua, pembobotan dengan menggabungkan 2 cabang dengan nilai IG tertinggi sebagai C2. Serta terakhir, penggabungan seluruh cabang sebagai C3.

Pencarian Dokumen Tekstual

Permasalahan yang muncul ketika dilakukan penggabungan bahasa adalah memilih nilai kemiripan yang tepat jika ada lebih dari 1 nilai. Untuk memilih nilai kemiripan digunakan nilai maksimal seperti pada Persamaan 8 berikut:

$$s = \max_{i} \left(\sum_{j} doc\_score_{ij} \right), \quad (8)$$

dimana i adalah cabang (*description*, *comment*, *caption*) dan j adalah kombinasi bahasa (en, de, fr, en+de, en+fr, en+de+fr). Hal ini dilakukan karena setiap cabang bahasa tersebut merepresentasikan arti yang serupa. Sehingga diambil nilai tertinggi yang dapat merepresentasikan kemiripan tertinggi.

Setelah nilai kemiripan didapatkan dengan menggunakan *cosine similarity*, maka dilakukan pembobotan cabang berdasarkan IG untuk dilakukan pengecekan kemiripan dengan menggunakan algoritma *W-Tree*.

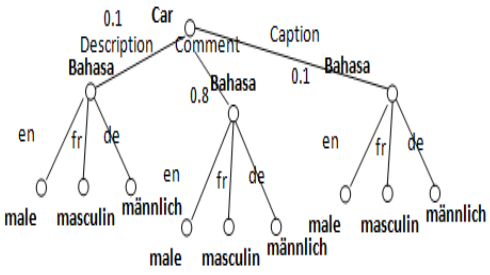
Pengukuran Efektifitas Sistem

Untuk mengukur seberapa efektif sistem yang digagas, digunakan *f-measure*. Model pengukuran tersebut telah terbukti

mampu melakukan evaluasi performa sistem[7].

HASIL DAN PEMBAHASAN

Tujuan dilakukan penelitian ini adalah membandingkan performa IG dalam penentuan bobot pada cabang *W-Tree*. Selain itu juga, untuk mengetahui hasil temu kembali informasi tersebut. Dataset yang digunakan tersaji pada **Error! Reference source not found.**. Adapun proses pencarian yang dilakukan terbagi menjadi pencarian tekstual, dan penggabungan cabang-cabang dengan nilai IG tertinggi.



Gambar 1. Tree dengan label dan bobot.

Pembobotan Berbasis Information Gain

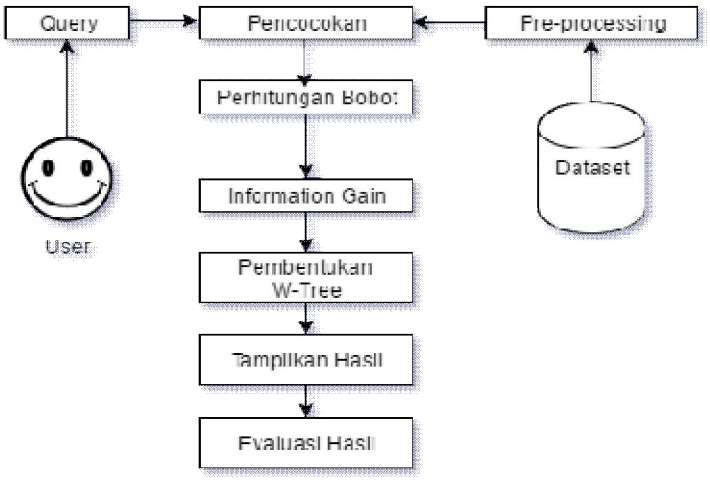
Tabel 2 menunjukkan nilai IG antara masing-masing cabang. Q1, Q2, dan Q5 hanya memiliki satu cabang yang memiliki nilai IG. Sedangkan pada cabang yang tidak memiliki nilai IG, dikarenakan tidak ditemukan kecocokan dengan *input user*.

Pada Q3 dan Q4 masing-masing cabang memiliki nilai IG dengan nilai berbeda. Seperti yang terlihat pada Tabel 1, dapat dilihat bahwa nilai *f-measure* tertinggi terjadi pada C1 (satu cabang dengan nilai IG tertinggi). Q1, Q2, dan Q5 tidak terjadi perbedaan hasil antara C1, C2, dan C3. Hal ini dikarenakan hanya satu cabang saja yang memiliki nilai IG.

Sedangkan pada Q3 dan Q4 didapatkan hasil yang bervariasi antar nilai C1, C2, dan C3. Hal ini dikarenakan masing-masing cabang memiliki nilai IG yang berbeda. Sedangkan penggabungan cabang (C2 dan C3) tidak memberikan nilai *f-measure* yang lebih tinggi dibanding hanya 1 cabang tertinggi (C1). Hasil ini terjadi karena penggabungan cabang justru membuat, nilai FP dan FN semakin besar. Sehinggga menurunkan nilai *f-measure*. Peningkatan maksimal *f-measure* dengan menggunakan IG tertinggi (C1) sebesar 73% dibanding menggunakan seluruh cabang (C3).

Tabel 1. *Query* yang digunakan.

indeks		Isi teks	Dokumen relevan	Total dokumen
	en	drawings of skeletons		
Q1	de	Zeichnungen von Skeletten	3	
	fr	dessins de squelettes		
	en	white ballet dress		
Q2	de	weisses Balletkleid	5	
	fr	robe de ballet blanche		
	en	chinese characters		
Q3	de	chinesische Schriftzeichen	22	9516
	fr	caractères chinois		
	en	male color portrait		
Q4	de	männliches Farbporträt	10	
	fr	portrait masculin en couleur		
	en	yellow flames		
Q5	de	gelbe Flammen	8	
	fr	flamme jaune		



Gambar 2. Arsitektur sistem.

Tabel 2. Nilai information gain setiap cabang.

Indeks	Jumlah Dokumen Relevan	*Information Gain*		
		Description	***Caption***	***Comment***
Q1	3	0	0.66702	0
Q2	5	0.90083	0	0
Q3	22	0.56929	0.66702	0.42821
Q4	10	0.56929	0.90083	0.4282
Q5	8	0	0.45019	0

Tabel 3. Perbandingan penggabungan nilai kemiripan.

Indeks	Penggabungan	TP	FN	FP	TN	F-Measure (%) x 10^{-2}
Q1	C1	1	2	944	8570	21.10
	C2	1	2	944	8570	21.10
	C3	1	2	944	8570	21.10
Q2	C1	1	4	1129	8383	17.62
	C2	1	4	1129	8383	17.62
	C3	1	4	1129	8383	17.62
Q3	C1	1	21	944	8551	20.68
	C2	1	21	1547	7948	12.74
	C3	1	21	1646	7849	11.98
Q4	C1	1	9	1129	8378	17.54
	C2	1	9	1710	7797	11.62
	C3	1	9	1807	7700	11.00
Q5	C1	1	7	627	8882	31.45
	C2	1	7	627	8882	31.45
	C3	1	7	627	8882	31.45

SIMPULAN

Pada penelitian ini, pencarian dokumen dengan dataset yang terdiri dari 3 bahasa memiliki tingkat heterogensi yang cukup tinggi. Adapun hasil pecobaan membuktikan pencarian tekstual dengan mengggunakan cabang tertinggi saja dengan penggunaan seluruh cabang, memberikan hasil berbeda.

Hal ini dapat dilihat dari nilai IG yang didapat pada percobaan. Hanya dengan menggunakan nilai IG tertinggi, ternyata mampu meningkatkan nilai *f-measure* yang cukup signifikan. Dimasa mendatang perlu diselidiki performa algoritma pencocokan selain *cosine similarity* agar mampu memberikan hasil yang lebih memuaskan.

DAFTAR PUSTAKA

[1] R. Sarno and F. Rahutomo, "Penerapan Algoritma Weighted Tree Similarity," *J. Teknol. Inf.*, no. January, pp. 39–46, 2008.

[2] C. E. Shannon, "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 1948.

[3] M. T. Martin-Valdivia and M. C. . Diaz-Galiano, "Using information gain to improve multi-modal information retrieval systems," vol. 44, pp. 1146–1158, 2008.

[4] G. Sidorov, A. Gelbukh, H. Gómez-Adorno, and D. Pinto, "Soft similarity and soft cosine measure: Similarity of features in vector space model," *Comput. y Sist.*, vol. 18, no. 3, pp. 491–504, 2014.

[5] S. Niwattanakul, J. Singthongchai, E. Naenudorn, and S. Wanapu, "Using of Jaccard Coefficient for Keywords Similarity," *Int. MultiConference Eng. Comput. Sci.*, vol. I, pp. 380–384, 2013.

[6] V. Bhavsar, H. Boley, and L. Yang, "A weighted-tree similarity algorithm for multi-agent systems in e-business environments," *Comput. Intell.*, vol. 20, no. 4, pp. 584–602, 2004.

[7] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation," no. December, p. 24, 2007.