CLUSTERING SEARCH ENGINE MENGGUNAKAN DYNAMIC SINGULAR VALUE DECOMPOSITION

Nurismy Aulia Aprina[1], Yanuar Firdaus A.w.[2], Shaufiah[3]

[1]Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom

Abstrak

Seiring dengan perkembangan teknologi saat ini, penggunaan internet sudah menjadi hal yang umum. Salah satu penggunaan internet adalah untuk mencari informasi melalui search engine. Jika pencarian dilakukan dengan hanya memasukkan satu term saja, maka akan menimbulkan kebingungan pada pengguna internet apakah informasi yang ditampilkan sudah sesuai.
Salah satu cara untuk mengurangi kebingungan pengguna dalam memilih informasi yang ditampilkan adalah dengan mengelompokkan hasil pencarian. Dengan pengelompokkan tersebut, hasil pencarian yang ada akan dikelompokkan berdasarkan label tertentu dimana label yang dihasilkan berdasarkan term frequency yang terbanyak dari setiap cluster yang dihasilkan. Pada Tugas Akhir ini akan dilakukan pengelompokkan hasil pencarian search engine menggunakan algoritma Dynamic Singular Value Decomposition berbasis Latent Semantic Indexing. Sebelum melakukan proses pengelompokkan di perlukan preprocessing yang terdiri dari pembuangan stoplist, dan stemming dan diikuti proses term weighting.
Hasil pengujian menunjukkan bahwa algoritma dynamic SVD clustering ini tidak cocok untuk dokumen snippets, sedangkan untuk pembobotan didapatkan metode pembobotan menggunakan TF-IDF menghasilkan nilai precision dan recall yang lebih baik. Metode pelabelan berdasarkan kemunculan term terbanyak dalam suatu cluster kadang kala menghasilkan label yang tidak sesuai dengan isi cluster yang dihasilkan.

Kata Kunci : hasil pencarian, dynamic singular value decomposition, preprocessing, clustering, latent semantic indexing, label

Abstract

The fast growing of technology makes internet utilization has become very common. One of the internet utilization is searching information using search engine. The problem is, when using search engine to search for information using one term, the result is still questionable, whether the information met his/her needs or not.
It is possible to help user choose the information that he/she need by clustering search result. by doing such, search result will be clustered based on certain labels. The labels itself obtained based on the most frequent term occured in the cluster. This final project will implement search engine's search result clustering using Dynamic Singular Value Composition which based on Latent Semantic Indexing. Prepocessing is required before clustering. The preprocessing itself will be consist of stoplist removal, stemming and then followed by term weighting.
Testing result showed that the Dynamic SVD Clustering Algorithm did not match for snippet documents, whereas for weighting scheme between TF and TF-IDF it showed that TF-IDF weighting schemes are having better value of precision and recall than TF weighting scheme. For labeling method by the most frequently term in one cluster, sometime are not compatible between label and cluster documents.

Keywords : search result, dynamic singular value decomposition, preprocessing, clustering, latent semantic indexing, label

1. Pendahuluan

1.1 Latar belakang

Search Engine seperti *google* saat ini sudah menjadi sarana umum bagi pengguna Internet karena kemudahaan dalam penggunaannya, yaitu user dapat memasukkan query sesuai dengan keinginan user dan kepentingan user, kemudian *search engine* akan melakukan pencarian dengan boolean search dan menampilkan hasil pencarian tersebut dengan disertai *snippets* atau deskripsi singkat dari hasil pencarian. Deskripsi singkat ini yang kemudian akan menjadi acuan user apakah informasi yang dicari sudah sesuai atau belum. Namun dibalik kemudahannya dalam melakukan pencarian hanya berdasarkan satu kata kunci yang umum, hal ini bisa menyebabkan pengguna awam kebingungan, karena hasil pencarian yang ditampilkan pastinya akan menjadi banyak sekali dan belum tentu informasi yang dicari oleh pengguna berada pada halaman awal (*top post)* sebagai contoh seorang pengguna ingin mencari suatu informasi dimana pengguna hanya memiliki petunjuk kata kunci 'super' atau 'job', dengan demikian *search engine* akan menampilkan berbagai informasi yang mengandung *term* 'super' atau 'job' dengan jumlah yang banyak. Informasi yang ditampilkan berupa *snippets* dengan jumlah halaman yang banyak.

Kondisi tersebut akan membuat pengguna kebingungan, karena pengguna tidak tahu dokumen yang mana yang dihasilkan dari pencarian tersebut yang sesuai, satu – satunya cara untuk mengetahui apakah dokumen yang dihasilkan adalah dokumen yang sesuai, maka pengguna harus melakukan pengecekan satu – persatu terhadap dokumen hasil pencarian. Jika dokumen yang dicari berada pada halaman awal (*top post*) maka hal ini tidak akan berpengaruh bagi pengguna, namun jika dokumen yang dicari tidak berada pada halaman – halaman awal maka hal ini dapat membuat pengguna kebingungan bahkan frustasi karena dokumen yang dicari belum juga ditemukan.

Saat ini untuk mengatasi kondisi tersebut adalah dengan menggunakan *clustering engines* yaitu mengelompokkan hasil pencarian berupa *snippets* (deskripsi singkat hasil pencarian) kedalam beberapa kelompok untuk memudahkan user dalam melakukan pencarian. Ide ini bukanlah satu ide baru, namun sudah cukup lama di kembangkan didalam *Information Retrieval*. Karena dalam proses pengelompokkannya tehnik *clustering engine* meng-analisa *snippet*, dimana *snippets* ini mengandung 0 – 40 kata sehingga tidak membutuhkan waktu yang lama dalam proses pengelompokkannya, namun bagaimanapun *snippets* seringkali tidak merepresentasikan isi/inti keseluruhan suatu dokumen, dan hal ini bisa menyebabkan terjadinya penurunan kualitas dari *cluster*.

Untuk mengatasi penurunan kualitas dari *cluster* tersebut, maka akan dilakukan pembuatan satu perangkat lunak *clustering engine* dengan menganalisa *snippet*. Algoritma yang digunakan dalam tehnik *clustering engine* yang digunakan yaitu *Dynamic SVD Clustering (DSC)* berbasiskan *Latent Semantic Indexing(LSI). Dynamic SVD Clustering* (DSC) merupakan algoritma *clustering* baru yang diyakini mempunyai ke akurasian tinggi, oleh karena itu algoritma DSC cocok untuk diterapkan dalam *clustering search result*. Algoritma yang digunakan berbasis LSI karena Latent Semantic Indexing (LSI) merupakan salah satu tehnik untuk memproyeksikan dokumen kedalam bentuk matrik.

1.2 Perumusan masalah

Rumusan masalah yang akan dikaji dalam tugas akhir ini adalah :

a. Bagaimana menerapkan algoritma *Dynamic SVD Clustering* berbasis *Latent Semantic Indexing* untuk mengelompokkan hasil pencarian.
b. Bagaimana menganalisis hasil keluaran dari *clustering* sebagai pengujian terhadap kinerja sistem.

Adapun beberapa batasan-batasan dalam tugas akhir ini yaitu :

a. Perangkat lunak yang dibuat hanya menyediakan dokumen koleksi berbahasa inggris. Dokumen berupa *snippets*.
b. *Query* yang dipakai hanya *query* dalam bahasa inggris.
c. Aplikasi berjalan secara *offline*.
d. Parameter yang digunakan untuk menguji kinerja sistem diantaranya adalah *precision* dan *recall*.

1.3 Tujuan

Tujuan yang ingin dicapai dalam pelaksanaan Tugas Akhir ini adalah :

a. Menerapkan algoritma *Dynamic SVD Clustering* dengan inputan dokumen *snippets*. Menerapkan algoritma *Dynamic SVD Clustering* untuk mengelompokkan hasil pencarian *search engine*.
b. Menganalisis hasil keluaran dari *clustering* berupa penghitungan *precision* dan *recall* untuk dua jenis pembobotan yang berbeda, yaitu pembobotan menggunakan TF, dan pembobotan menggunakan TF-IDF.
c. Menilai relevansi antara label yang dihasilkan dengan dokumen yang dikelompokkan berdasarkan perbandingan kategori jumlah dokumen relevan terbanyak dalam suatu *cluster* dengan label yang dihasilkan.

1.4 Metodologi penyelesaian masalah

Metodologi yang digunakan dalam pelaksanaan tugas akhir ini, yaitu:

a. Mengumpulkan bahan-bahan referensi seperti *paper,* jurnal ilmiah, atau buku yang berkaitan dengan *clustering* dengan menggunakan *Dynamic SVD Clustering.*

b. Melakukan analisis mengenai permasalahan yang ada dalam membuat perangkat lunak yang memiliki kemampuan untuk mengelompokkan hasil *search engine*.

   Hipotesa awal terhadap perangkat lunak ini yaitu dengan menggunakan algoritma *Dynamic Singular Value Decomposition* berbasis *Latent Semantic Indexing* untuk *clustering*/pengelompokkan dapat mengelompokkan hasil pencarian *search engine* untuk mempermudah pengguna dalam mencari informasi.

c. Melakukan penyediaan data berupa *snippets* hasil pencarian *search engine.*
d. Membuat analisis pengetesan *cluster* dengan menggunakan *precision* dan *recall*.
e. Melakukan analisis dan desain perangkat lunak.
f. Pembangunan model.
g. Melakukan implementasi dari hasil analisis dan perancangan perangkat lunak.
h. Analisis hasil implementasi.
i. Pembuatan laporan terhadap penelitian yang dilakukan.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil pengujian dengan data set berupa dokumen *snippet* yang terdiri dari DMOZsamples3, DMOZsamples4B, DMOZsamples5A, DMOZsamples6, dan DMOZsamples8, maka dapat ditarik kesimpulan sebagai berikut :

1. Algoritma Dynamic SVD Clustering tidak cocok digunakan pada dokumen data set.
2. Pembobotan TF-IDF lebih berhasil mengelompokkan dokumen – dokumen dengan kategori yang sama dalam satu *cluster*, karena TF-IDF memperhatikan kemunculan *term* pada kumpulan dokumen, sehingga dapat menghilangkan *noise* dari kumpulan dokumen dan terlihat dari hasil nilai rata – rata pengujian *precision* dan *recall* untuk pembobotan TF-IDF menghasilkan nilai yang lebih baik dari pembobotan TF.
3. Pembobotan TF seringkali menghasilkan *cluster* yang memiliki kumpulan dokumen relevan terbanyak yang sama antara satu *cluster* dengan *cluster* lainnya, hal ini dikarenakan lebih banyak *term* yang diproses dalam SVD sehinga *noise* yang dihasilkan lebih banyak.
4. Metoda pelabelan berdasarkan *term* dengan frekuensi kemunculan tertinggi kadangkala tidak menggambarkan isi *cluster* karena inputan berupa *snippet* saja dimana *snippet* ini hanya memberikan informasi yang sedikit dari dokumen bisa sangat bervariasi dengan *term* yang bervariasi juga untuk setiap kategori dokumennya, hal ini bisa menyebabkan bukan *term* yang berasal dari kumpulan dokumen relevan terbanyak dalam *cluster* yang menjadi label suatu *cluster*.

5.2 Saran

1. Sebaiknya digunakan juga inputan berupa dokumen utuh, untuk mengetahui kemampuan algoritma *dynamic SVD* pada dokumen utuh.
2. Untuk pelabelan sebaiknya dilakukan percobaan dengan metode pelabelan lainnya untuk menghasilkan label yang lebih merepresentasikan isi *cluster*nya.
3. Sebaiknya dokumen data masukan tidak hanya terbatas untuk dokumen berbahasa inggris saja, dan aplikasi di lengkapi dengan deteksi bahasa untuk memilih dokumen yang akan dikelompokkan.

6. Daftar Pustaka

1	A. K. Jain, M. N. Murty, P. J. Flynn. Data Clustering: A Review. ACM Computing Surveys (1999) 31(3):265-323.
2	Andrianto, Tomy. Latent Semantic Indexing untuk Information Retrieval. 2005. http://www.hansmichael.com/default.asp?cat=exta199113754
3	Ayuningtias, Vidya. *Pengkategorian Hasil Pencarian Dokumen dengan Clustering*. 2007. ITTelkom, Bandung. Indonesia.
4	Eckel, Bruce. Thinking in Java, 4th Edition. 2007. Prentice Hall
5	Firdaus, Yanuar. *Diktat Kuliah Information Retrieval*. 2007. ITTelkom, Bandung.
6	Garcia, Edel. Singular Value Decomposition (SVD), A Fast Track Tutorial. 2006. http://www.miislita.com/information-retrieval-tutorial/
7	Goodrich, Micheal, T. Tamassia, Roberto. Data Structures & Algorithms in Java. 2001. John Wiley & Sons, Inc.
8	http://ienx.files.wordpress.com/2007/09/bab-ii-landasan-teori.pdf
9	http://id.wikipedia.org/wiki/Algoritma_Prim
10	Larman, Craig. Applying UML and Patterns, An Introduction to Object-Oriented Analysis and Design and the Unified Process. 1999. Prentice Hall
11	http://en.wikipedia.org/wiki/Structured_document
12	Manning, Christoper D. Raghavan, Prabakhar. Schütze, Hinrich, Introduction to Information Retrieval. 2008. Cambridge University Press
13	Rumbaugh, James. Jacobson, Ivar. Booch, Grady. The Unified Modeling Language User Guide. 1999. Addison Wesley
14	M. W. Berry. Large-Scale Sparse Singular Value Computations. The International Journal of Supercomputer Applications (1992) 6(1):13—49
15	Munir, Rinaldi. 2003. Matematika Diskrit. Bandung. Informatika

16	Mecca, Giansalvatore. Raunich, Salvatore. Pappalardo, Alessandro. A New Algorithm For Clustering Search Result. 2007.
17	R. C. Prim. Shortest Connection Networks and Some Generalisations. *Bell ystems Technical Journal* (1957) 36:1389-1401.
18	Tan, Pang-Ning. and Kumar, Vipin. Introduction to Data Mining. Pearson Education, Inc., Boston, 2006.
19	The Noodles Project Web Site. http://www.db.unibas.it/projects/noodles.
20	The Open Directory Project (DMOZ). http://www.dmoz.org.
21	C. T. Zahn. Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters. *IEEE Transactions on Computers* (1971) C-20:68-86.