ANALISIS DAN IMPLEMENTASI COLLABORATIVE FILTERING MENGGUNAKAN STRATEGI SMOOTHING DAN FUSING PADA RECOMMENDER SYSTEM

ANALYSIS AND IMPLEMENTATION COLLABORATIVE FILTERING USING SMOOTHING AND FUSING STRATEGY ON RECOMMENDER SYSTEM

Wahyu Rismawan[1], Agung Toto Wibowo, ST., MT.[2], Mahmud Dwi Sulistyo, ST., MT.[3]

[1,2,3]Prodi S1 Teknik Informatika, Fakultas Teknik, Universitas Telkom

[1]**wahyurismawan@gmail.com**, [2] **atwbox@gmail.com**, **mahmuddwis@gmail.com**

Abstrak

Collaborative Filtering (CF) adalah salah satu pendekatan yang populer untuk membangun *Recommender System* dengan memanfaatkan informasi dan preferensi dari *user* lain untuk memberikan rekomendasi *item*. Salah satu permasalahan mendasar dalam CF adalah data rating yang sangat sedikit (*data sparsity)* yang mampu mempengaruhi hasil rekomendasi. Secara umum terdapat dua tipe algoritma pada CF, yaitu *memory-based* dan *model-based* yang memiliki kelebihan dan kekurangan masing-masing. Dalam tugas akhir ini, digunakan strategi *smoothing* dan *fusing* yang merupakan pendekatan hybrid dari *memory-based* dan *model-based* untuk menangani permasalahan *data sparsity*.

Berdasarkan hasil pengujian, strategi *smoothing* dan *fusing* mampu menurunkan error sistem yang diukur menggunakan MAE dari 2,277 menjadi 0,746 atau menurun sebesar 50.62% dibandingkan tanpa menggunakan strategi *smoothing* dan *fusing*. Selain itu, akurasi sistem juga dipengaruhi oleh level *sparsity* dari data rating. Semakin *sparse* data rating yang dimiliki, maka akurasi yang dihasilkan semakin buruk.

Kata kunci : Collaborative Filtering, Recommender System, Smoothing and Fusing, Data Spasity

Abstract

Collaborative Filtering (CF) is one of a popular approach for building Recommender System using information and preference of another users in order to give an items recommendation. One of basic problem in CF is data sparsity, which affect recommendation result. In general, there are two types of algorithm in CF: memory-based and model-based, each methods has a plus and minus for giving recommendation. This research use smoothing and fusing strategy. That's method is an hybrid approach of memory-based and model-based to solve data sparsity problem.

Based on testing result, smoothing strategy can reduce error system which measured by MAE from 2,277 to 0,746 or reducing until 50.62% compared with did not use smoothing and fusing strategy. Furthermore, the sparsity level can affect accuracy. The more sparse data rating owned, then the result accuracy is getting worse.

Kata kunci : Collaborative Filtering, Recommender System, Smoothing and Fusing, Data Spasity

1. Pendahuluan

Teknologi Informasi dan Komunikasi yang sangat cepat berkembang beberapa tahun ini mengakibatkan aktivitas di dunia maya seperti pada *e-commerce*, media sosial, pendistribusian konten (film; buku; musik; berita) juga ikut meningkat. Di banyak kasus, user dihadapkan pada sangat banyak pilihan *item* yang secara langsung atau tidak langsung dipaksa untuk memilih *item* tersebut.

Untuk membantu user dalam membuat keputusan, maka dibutuhkan sebuah *Recommender System* (RS) yang dapat memberikan rekomendasi pilihan yang mungkin disukai oleh user. Pendekatan *Collaborative Filtering* (CF) adalah salah satu teknik untuk membangun RS dengan memanfaatkan informasi dari interaksi user terhadap sebuah item seperti *purchase* pada *e-commerce* atau pemberian *rating* pada aplikasi pendistribusian konten. Namun CF memiliki salah satu permasalahan mendasar yaitu *data sparsity* yang akan mempengaruhi kualitas rekomendasi yang dihasilkan.

Secara umum pendekatan CF dapat dikategorikan menjadi dua kategori, yaitu Memory-based dan Model-based [1]. *Memory-based* mampu menghasilkan akurasi yang tinggi dengan mengeksploitasi fungsi *similarity* dari *item* dan *user*, tetapi seiring bertambahnya data *user* atau *item*, performansi sistem dengan *memory-based* menurun karena proses komputasi untuk menghasilkan rekomendasi melibatkan seluruh matriks *user-item*. Beberapa metode Memory-based antara lain: *Pearson-Correlation* [2] The Vector Similarity [3] dan Generalized Vector Space Model [4]. Kedua adalah *Model-based yang* memiliki kemampuan dalam menangani permasalahan skalabilitas sistem. Clustering [5] dan Bayesian Network [3]. Kekurangan dari *Model-based* adalah akurasi yang dihasilkan sangat bergantung pada banyaknya data rating yang tersedia. Pada kenyataannya, data *rating* yang tersedia pada sistem

rekomendasi sangat jarang, yaitu kurang dari 1% [6]. Dengan kondisi demikian, *Model-based* akan memberikan kualitas rekomendasi yang kurang baik [6].

Pada penelitian ini digunakan metode Collaborative Filtering dengan strategi Smoothing dan Fusing (CFSF). CFSF adalah salah satu metode pada CF dengan pendekatan gabungan antara *Memory-based* dengan *Model-based*. CFSF menggunakan teknik *clustering* dan *data smoothing* yang terbukti dapat menangani masalah *data sparsity*.

2. Landasan Teori

2.1 Collaborative Filtering

Collaborative Filtering adalah salah satu teknik yang dapat digunakan untuk pemberian rekomendasi *item* dengan mempertimbangkan persamaan preferensi user lain [8] Preferensi tersebut didapat dari *rating* yang diberikan oleh user secara eksplisit atau rekaman aktivitas interaksi user dengan item [1]. Pekerjaan dasar CF pada Sistem Rekomendasi adalah memprediksi *rating* pada item dari user [7].

2.2 User-based Collaborative Filtering

User-based CF memiliki konsep yang mirip dengan item-based CF, hanya saja sudut pandang untuk menghitung nilai *similarity* dilihat dari sisi *user*. Nilai *similarity* antar *user* bisa didapat menggunakan persamaan 2.

$$\text{(1)}$$

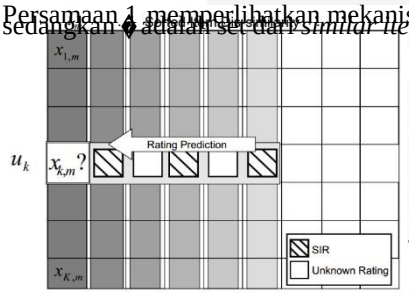
Persamaan 2 memperlihatkan mekanisme user-based CF dimana adalah nilai similarity antar user, sedangkan SU adalah set dari *like-minded user*. Ilustrasi proses prediksi rating pada User-Based CF dapat dilihat pada Gambar 2. Nilai prediksi rating yang belum diketahui, didapat dengan melibatkan item-item yang diketahui dari *like-minded* user (*SUR*).

2.3 Item-based Collaborative Filtering

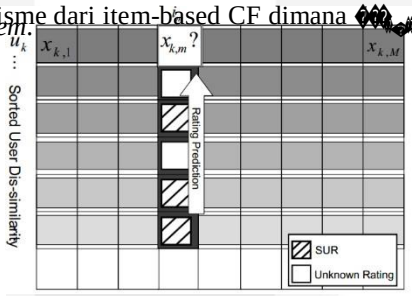
Item-based CF adalah salah satu algoritma model-based untuk membuat sebuah rekomendasi. Pada pendekatan ini, kemiripan antar item diukur menggunakan sebuah *similarity measure*, salah satunya menggunakan Pearson's Correlation seperti pada persamaan 2.

$$\text{(2)}$$

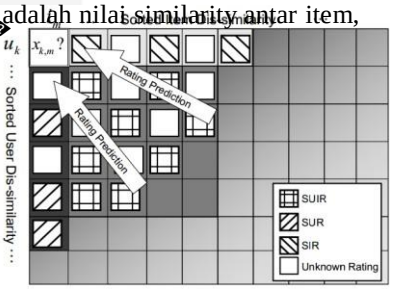
Persamaan 1 memperlihatkan mekanisme dari item-based CF dimana adalah nilai similarity antar item, sedangkan adalah set dari *similar item*.



Gambar 1 Item-based CF



Gambar 2 User-based CF



Gambar 3 User-Item-based CF

Seperti yang diilustrasikan pada Gambar 1, prediksi rating yang nilainya belum diketahui di dapat dengan cara mencari nilai rata-rata dari rating *similar item* (*SIR*).

2.4 User-Item-based Collaborative Filtering

User-based CF memiliki konsep yang mirip dengan item-based CF, hanya saja sudut pandang untuk menghitung nilai *similarity* dilihat dari sisi *user*. Nilai *similarity* antar *user* bisa didapat menggunakan persamaan 2.

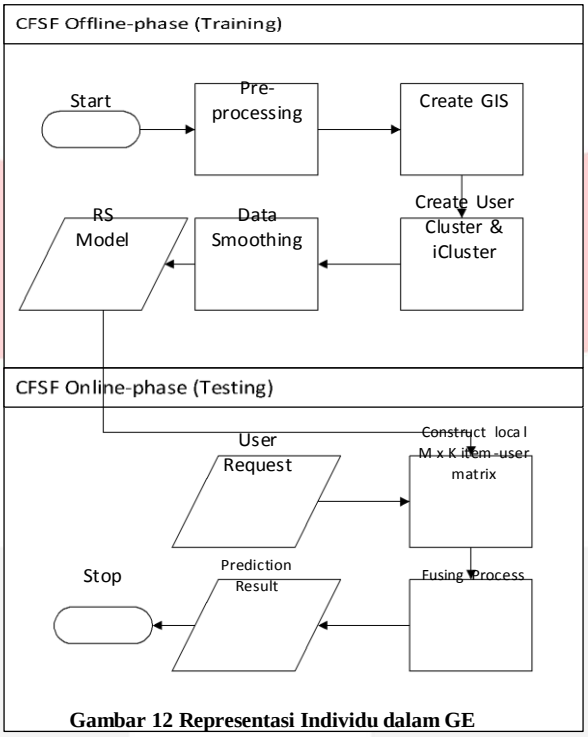
$$\text{(3)}$$

Persamaan 2 memperlihatkan mekanisme user-based CF dimana adalah nilai similarity antar user, sedangkan SU adalah set dari *like-minded user*. User-based dan Item-based masih memiliki kelemahan ketika kondisi item banyak yang tidak memiliki rating sangat banyak, sehingga ada memungkinkan prediksi yang

dihasilkan didapat dari rata-rata user atau item yang 'tidak similar'. Oleh karena itu, User-Item based melibatkan item yang dirating oleh *like-minded user (SUR)*, rating dari *similar item* (*SIR)* dan rating dari *similar item* yang berasal dari *like-minded user (SUIR)* untuk perhitungan prediksi [1]

2.4 Perancangan Sistem

Pada penelitian tugas akhir ini, sistem dibuat bertujuan untuk memberikan rekomendasi item dalam bentuk prediksi rating item. Dataset yang digunakan adalah dataset rating film berasal dari MovieLens. Sistem menerima request berupa user-id dan item-id yang disebut dengan user-aktif dan item-aktif dan mengeluarkan keluaran berupa prediksi nilai rating dari item tersebut.



Gambar 12 Representasi Individu dalam GE

Terdapat dua proses utama dalam sistem, yaitu proses learning dan proses testing. Proses learning adalah pembuatan model yang akan menjadi basis rekomendasi pada saat proses testing. Langkah pertama yang dilakukan pada saat learning adalah membuat cluster user menggunakan algoritma K-Means. Proses clustering ini bertujuan untuk mengelompokkan user yang memiliki kemiripan yang dilihat dari rating yang dimiliki. Informasi pada cluster user digunakan untuk melakukan data smoothing. Data smoothing bertujuan untuk memberikan nilai rating sementara pada item yang belum diberikan nilai rating oleh user. Langkah selanjutnya adalah membangun Global Item Similarity (GIS) yang berisi nilai similarity antar item. Perhitungan nilai similarity antar item menggunakan persamaan Pearson's Correlation. Langkah terakhir di proses learning adalah membuat iCluster yang berisi nilai similarity antara user dengan cluster yang di urutkan secara *descending*. iCluster bertujuan untuk mempermudah pencarian user yang mirip pada saat proses testing. Proses learning akan menghasilkan cluster user, iCluster dan GIS sebagai model untuk perhitungan prediksi di proses testing.

Pada proses testing, sistem menerima *request* atau masukan berupa informasi user-id dan item-id untuk di prediksi nilai ratingnya. Dari *request* tersebut dibuat sebuah *local item-user matrix* dengan cara memilih Top-K user yang memiliki kemiripan dengan user aktif dan Top-M item yang memiliki kemiripan dengan item aktif. Dari *local item-user matrix* dilakukan prediksi menggunakan pendekatan memory-based dengan proses *fusing*.

3. Hasil Pengujian dan Kesimpulan

Parameter ω berperan sebagai pembeda rating asli dengan rating hasil smoothing. Parameter ω memberikan porsi seberapa besar dilibatkannya rating asli dan rating hasil smoothing pada saat perhitungan rekomendasi. Pada percobaan kali ini, nilai MAE terkecil dicapai ketika ω bernilai 0.5 dengan data Given20. Hasil yang sama juga terlihat pada data Given5 dan Given10, dimana MAE yang terkecil dicapai ketika ω bernilai 0.5.

ω	MAE			
	Given5	***Given10***	***Given20***	***Rata-rata***
0	0.769169	0.759022	0.749463	0.7592182
0.1	0.765032	0.753804	0.744481	0.7544391
0.2	0.762392	0.750988	0.741803	0.7517277
0.3	0.762398	0.749201	0.741176	0.7509248
0.4	0.768029	0.755424	0.741069	0.7548406
0.5	0.75383	0.748331	0.737344	0.7465016
0.6	0.765029	0.778725	0.812848	0.7855342
0.7	1.013978	1.065269	0.918824	0.9993571
0.8	0.892372	0.864161	0.868027	0.8748536
0.9	1.291068	1.226908	1.378394	1.2987901
1	2.563313	2.39405	1.875359	2.2775741
Perbaikan MAE	***1.809482***	***1.645719***	***1.138016***	***1.531073***

Tabel 1 Hasil Pengujian Parameter Smoothing

Dari hasil pengujian didapat bahwa terjadi penurunan MAE atau peningkatan akurasi ketika strategi smoothing digunakan. Ketika ω bernilai 0,5 yang bearti bobot rating asli dan rating smoothing seimbang, rata-rata nilai MAE berubah dari 2,227 menjadi 0.746 atau turun sebesar 50.624%. Hal ini membuktikan bahwa strategi smoothing dengan nilai ω tepat dapat meningkatkan akurasi sistem jika dibandingkan dengan hanya menggunakan rating asli saja.

Daftar Pustaka:

[1]. J. Wang, A. P. De Vries and M. J. Reinders, "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," in *In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, Seattle, 2006.

[2]. P.e and a. Resnick, "GroupLens: an open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 1994.

[3]. J. S. Breese, D. Heckerman and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, 1998.

[4]. I. Soboroff and C. Nicholas, "Collaborative filtering and the generalized vector space model," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, 2000.

[5]. A. K. B. Merialdo, "Clustering for collaborative filtering applications," in *Intelligent Image Processing, Data Analysis & Information Retrieval*, 1999.

[6]. D. Zhang, J. Cao, J. Zhou, M. Guo and V. Raychoudhury, "An efficient collaborative filtering approach using smoothing and fusing," in *Parallel Processing, 2009. ICPP'09. International Conference on*, 2009.

[7]. P. B. Kantor, L. Rokach, F. Ricci and B. Shapira, Recommender systems handbook, Springer, 2011.

[8]. J. McCrae, A. Pieatek and A. Langley, "Collaborative Filtering," 2004. [Online]. Available: http://www.imperialviolet.org. [Accessed 2015].

[9]. A. S. Das, M. Datar, A. Garg and S. Rajaram, "Google news personalization: scalable online collaborative filtering," in *Proceedings of the 16th international conference on World Wide Web. ACM*, 2007.

[10]. B. Sarwar, G. Karypis, J. Konstan and J. Riedl, "Item-based collaborative filtering recommendation algorithm," in *ACM*, 2001.

[11]. Adomavicius, Gedimina and A. Tuzhilin, "Adomavicius, Gediminas, and Alexander Tuzhilin. "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," in *Knowledge and Data Engineering, IEEE Transactions*, 2005.

[12]. R. Bell, Y. Koren and C. Volinsky, "Modeling relationships at multiple scales to improve accuracy of large recommender systems," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM*, 2007.

[13]. J. Kleinberg and M. Sandler, "Using mixture models for collaborative filtering," in *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing. ACM*, 2004.