



UNIVERSITAS INDONESIA

**ANALISIS KINERJA *NON-NEGATIVE MATRIX FACTORIZATION*
(NMF) DALAM SISTEM REKOMENDASI**

TESIS

SUPRIADI

1006786303

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
PROGRAM STUDI MAGISTER MATEMATIKA
DEPOK**



UNIVERSITAS INDONESIA

**Analisis Kinerja *Non-Negative Matrix Factorization* (NMF)
Dalam Sistem Rekomendasi**

TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar
Magister

SUPRIADI
1006786303

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
PROGRAM STUDI MAGISTER MATEMATIKA

DEPOK
JULI 2012

HALAMAN PERSETUJUAN

Judul : Analisis Kinerja *Non-Negative Matrix Factorization* (NMF) Dalam Sistem Rekomendasi
Nama : Supriadi
NPM : 1006786303

Laporan Tesis ini telah diperiksa dan disetujui.

Depok, 13 Juli 2012




Dr. rer.nat Hendri Murfi, M.Kom
Pembimbing Tesis

HALAMAN PERNYATAAN ORISINALITAS

Tesis ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.

Nama : Supriadi

NPM : 1006786303

Tanda Tangan : 

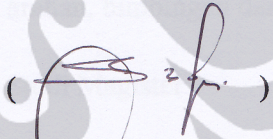
Tanggal : 13 Juli 2012


HALAMAN PENGESAHAN

Tesis ini diajukan oleh :
Nama : Supriadi
NPM : 1006786303
Program Studi : Magister Matematika
Judul Tesis : Analisis Kinerja *Non-Negative Matrix Factorization (NMF)* dalam Sistem Rekomendasi

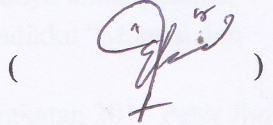
Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister pada Program Studi Magister Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing : Dr. rer nat Hendri Murfi, M.Kom ()

Penguji : Prof. Dr. Djati Kerami ()

Penguji : Dr. Hengki Tasman, M.Si ()

Penguji : Arie Wibowo S.Si, M.Si ()

Ditetapkan di : Depok
Tanggal : 13 Juli 2012

KATA PENGANTAR

Segala puji hanyalah milik Allah SWT yang telah memberikan segala nikmat bagi seluruh umat manusia. Shalawat teriring salam semoga selalu tercurahkan bagi teladan umat manusia, Nabi Muhammad SAW, kepada keluarga, kepada sahabat, dan kepada para pengikut beliau yang istiqomah hingga hari pembalasan nanti. Rasa syukur tiada terkira penulis sampaikan kehadiran Ilahi Rabbi sehingga penulisan tesis yang berjudul "**Analisis Kinerja *Non-Negative Matrix Factorization (NMF)* dalam Sistem Rekomendasi**" dapat terselesaikan pada waktunya.

Banyak pihak yang telah membantu dalam penulisan tesis ini, oleh sebab itu tidak berlebihan kiranya penulis mengucapkan terima kasih kepada pihak-pihak terkait :

1. Pemerintah Provinsi Jambi Melalui Dinas Pendidikan yang telah memberikan kesempatan kepada penulis untuk mendapatkan program beasiswa Pascasarjana
2. Bapak Dr. rer nat Hendri Murfi, M.Kom. atas segala arahan, bimbingan dan perhatian yang selama ini diberikan.
3. Bapak Prof. Dr. Djati Kerami selaku Ketua Program Studi Magister Matematika sekaligus penasehat akademik penulis.
4. Seluruh staf pengajar di Departemen Matematika FMIPA UI, terima kasih atas segala ilmu yang telah di berikan.
5. Seluruh karyawan di Departemen Matematika FMIPA UI, terima kasih atas semua bantuan yang telah di berikan kepada penulis.
6. Istri dan putra putriku terkasih atas segala do'a, perhatian, kesabaran serta dukungannya selama menempuh pendidikan ini.
7. Ibu dan bapak tercinta yang telah memberikan semuanya untuk penulis, serta kakak-ku Yanto beserta keluarganya dan adik-adikku "Maman dan Husen", terima kasih atas semua bantuannya .
8. Rekan-rekan mahasiswa pascasarjana matematika angkatan 2010 Peter Jhon, Martin, Subian, Dilla, Endaryono, Bob, Sigit, Tarhadi, Ilmiyati, Nurma, Rifkos, Uun, Dewi, Desti, Mia, Fenny, Sagita, Endang, Teteh, Agus Dahlia, Fathin, Shantika dan Debby semoga kebersamaan, kehangatan dan kekeluargaan yang telah tercipta abadi selamanya.
9. Khusus untuk Bapak Umar Ruswandi terima kasih atas semua yang telah diberikan, terutama bantuan programnya.

10. Rekan-rekan mahasiswa pascasarjana matematika angkatan 2010 asal Jambi M Haryono, Murtiningrum, Muzayyin Ahmad, Ridayanti, Rida Novrida, Meilisa, Tri Haryani.
11. Dan semua pihak yang telah membantu penulis dalam menyelesaikan tesis ini yang tidak dapat disebutkan satu persatu

Akhir kata, penulis mohon maaf jika terdapat kesalahan serta kekurangan dalam penulisan tesis ini, saya berharap semoga Allah SWT berkenan membalas atas segala kebaikan semua pihak yang telah membantu. Semoga tesis ini dapat bermanfaat bagi kita semua, Amin.

Depok, 13 Juli 2012

Supriadi

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini :

Nama : Supriadi
NPM : 1006786303
Program Studi : Magister Matematika
Fakultas : Matematika dan Ilmu Pengetahuan Alam
Jenis Karya : Tesis

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul :

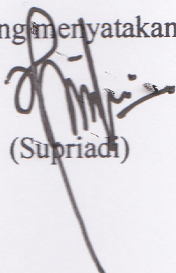
“Analisis Kinerja *Non-Negative Matrix Factorization* (NMF) dalam Sistem Rekomendasi”

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 13 Juli 2012

Yang menyatakan


(Supriadi)

ABSTRAK

Nama : Supriadi
Program Studi : Magister Matematika
Judul : Analisis Kinerja *Non-Negative Matrix Factorization* (NMF)
Dalam Sistem Rekomendasi

Pesatnya perkembangan teknologi informasi saat ini, telah berdampak terhadap transaksi perdagangan, dimana proses perdagangan tidak hanya menggunakan cara konvensional tetapi telah memanfaatkan teknologi informasi. Keberadaan suatu sistem yang dapat memberikan rekomendasi kepada pengguna atau lebih dikenal dengan istilah sistem rekomendasi mulai banyak diperhitungkan. Yang menjadi permasalahan adalah apakah rekomendasi yang diberikan telah sesuai dengan keinginan dan kebutuhan pengguna. Tesis ini menguraikan tentang analisis akurasi prediksi yang diperoleh dari sistem rekomendasi berdasarkan metode *collaborative filtering* dengan menggunakan teknik *Non-Negative Matrix Factorization* (NMF). Hasil pengujian menunjukkan bahwa akurasi prediksi yang dihasilkan telah relatif baik dengan nilai kesalahan sebesar 0,950491.

Kata Kunci : *sistem rekomendasi, collaborative filtering, Non-Negative Matrix Factorization (NMF)*

ABSTRACT

Name : Supriadi
Courses : Master of Mathematics
Title : Performance Analysis of Non-Negative Matrix Factorization (NMF) in Recommendation Systems

The rapid development of information technology today, has an impact on trade transactions, where the trade is not only using conventional means but it has been using information technology. The existence of a system that can provide recommendations to the user, or better known as a recommendation system began many counts. What matters is whether the recommendations are in accordance with the wishes and the needs of users. This thesis describes the analysis of the accuracy of predictions obtained from a recommendation system based on collaborative filtering method using the technique of Non-Negative Matrix Factorization (NMF). Test results show that the resulting prediction accuracy was relatively good with an error value of 0.950491

Keywords : system recommendations, collaborative filtering, Non-negative matrix factorization (NMF).

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	iii
HALAMAN PERNYATAAN ORISINALITAS	iv
HALAMAN PENGESAHAN	v
KATA PENGANTAR	vi
HALAMAN PERSETUJUAN PUBLIKASI KARYA ILMIAH	viii
ABSTRAK	ix
ABSTRACT	x
DAFTAR ISI	xi
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	4
1.3 Tujuan Penelitian	4
1.4 Metode Penelitian	4
2. LANDASAN TEORI	6
2.1 Sistem Rekomendasi	6
2.2 Collaborative Filtering	9
2.3 Machine Learning	18
2.4 Teori Matriks	20
3. NON-NEGATIVE MATRIX FACTORIZATION	26
3.1 Faktorisasi Matriks	26
3.2 Faktorisasi Matriks Dalam Sistem rekomendasi	27
3.3 Non-Negative Matrix Factorization (NMF)	30
3.4 Multiplicative Update Rule	32
3.5 Algoritma Multiplicative Update Rule	37
4. SIMULASI	40
4.1 Data	40
4.2 Spesifikasi Komputer	41
4.3 Skema Eksperimen	41
4.4 Hasil Penelitian	45
5. KESIMPULAN DAN SARAN	46
5.1 Kesimpulan	46
5.2 Saran	47
DAFTAR PUSTAKA	48

DAFTAR GAMBAR

Gambar 2.1	<i>Ilustrasi sistem rekomendasi</i>	8
Gambar 2.2	<i>Proses collaborative filtering</i>	12
Gambar 2.3	<i>Pembentukan user-neighborhood</i>	13
Gambar 2.4	<i>Contoh cosinus antara dua vektor</i>	14
Gambar 2.5	<i>Pembentukan item-neighborhood</i>	15
Gambar 2.6	<i>Ilustrasi proses menentukan nilai prediksi suatu produk</i>	16
Gambar 3.1	<i>Ilustrasi non-negative matrix factorization (NMF)</i>	30
Gambar 3.2	<i>Ilustrasi auxiliary function</i>	34
Gambar 3.3	<i>Algoritma multiplikative update untuk NMF</i>	38
Gambar 3.4	<i>Simulasi uji konvergensi algoritma multiplikative</i>	39
Gambar 4.1	<i>Contoh data training dan data testing yang saling bebas</i>	41
Gambar 4.2	<i>Skema percobaan model NMF</i>	42
Gambar 4.3	<i>Grafik perbandingan nilai root mean square error (RMSE) untuk berbagai nilai k</i>	44



DAFTAR TABEL

Tabel 2.1 <i>Contoh bentuk data triple</i>	10
Tabel 2.2 <i>Matriks yang dibentuk dari data rating suatu produk</i>	19
Tabel 4.1 <i>Indikator basis data movielens</i>	40
Tabel 4.4 <i>Perbandingan nilai RMSE hasil simulasi</i>	43



BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi internet yang semakin pesat berdampak terhadap semakin besar dan melimpahnya sumber informasi. Hal ini tentu saja mengakibatkan masyarakat sebagai pengguna (*user*) mengalami masalah dalam mendapatkan informasi yang tepat, cepat, dan relevan dengan kebutuhannya. Seiring dengan perkembangan teknologi internet tersebut, berkembang pula bidang-bidang ilmu pendukungnya. Salah satu bidang ilmu tersebut adalah *information retrieval*.

(Krzysztof, 2007) mendefinisikan *information retrieval* sebagai tindakan, metode dan prosedur untuk menemukan kembali data yang tersimpan, kemudian menyediakan informasi mengenai subyek yang dibutuhkan. Tindakan tersebut mencakup penyusunan indeks teks, analisis penyelidikan dan analisis relevansi. Data mencakup teks, tabel, gambar, ucapan, dan video. Informasi termasuk pengetahuan terkait yang dibutuhkan untuk mendukung penyelesaian masalah dan akuisisi pengetahuan. Informasi yang dicari dapat berupa berupa teks, gambar, audio, video dan lain-lain. Koleksi data teks yang dapat dijadikan sumber pencarian juga dapat berupa pesan teks, seperti *e-mail*, *fax*, dan dokumen berita, bahkan dokumen yang beredar di internet.

Dengan jumlah dokumen koleksi yang besar sebagai sumber pencarian, maka dibutuhkan suatu sistem yang dapat membantu pengguna menemukan dokumen yang relevan dalam waktu yang singkat dan tepat. Khusus pada industri film, setiap tahunnya terjadi penambahan tidak kurang dari 5.000 judul film dan terus bertambah. Hal ini tentu saja menciptakan suatu kondisi dimana pengguna

mengalami kesulitan dalam memperoleh informasi, memilih dan menentukan film yang sesuai dengan seleranya.

Berkaitan dengan hal itu, saat ini berkembang banyak sistem yang dirancang untuk mengatasi permasalahan tersebut. Salah satunya adalah suatu mekanisme yang dapat menyaring dan memberikan informasi atau rekomendasi yang sesuai dengan selera pengguna berdasarkan *rating* atau *review* dari pengguna sebelumnya yang memiliki keinginan yang sama (*user-neighbor*). Mekanisme ini dikenal sebagai suatu sistem rekomendasi (*recommender system*). Menurut (Sarwar, 2001), sistem rekomendasi adalah suatu sistem yang memberikan atau menyarankan informasi yang berguna atau memperkirakan apa yang akan dilakukan pengguna untuk mencapai keinginannya. (Karypis, 2001) berpendapat bahwa sistem rekomendasi adalah suatu mekanisme yang dapat menyaring dan memberikan suatu informasi atau rekomendasi berdasarkan nilai dari pengguna sebelumnya yang mempunyai selera yang sama (*user-neighbor*). Sedangkan menurut (McGinty, 2006), sistem rekomendasi diartikan sebagai sebuah model aplikasi dari sebuah observasi terhadap keadaan dan keinginan pelanggan, sistem rekomendasi memanfaatkan opini seseorang terhadap suatu barang dalam domain atau kategori tertentu untuk membantu seorang pelanggan dalam memilih dan menentukan suatu produk.

Terdapat banyak teknik dalam membangun sistem rekomendasi dan salah satu tehnik dalam sistem rekomendasi yang telah sukses dan banyak digunakan adalah *collaborative filtering*. Menurut (Sarwar, 2001), *collaborative filtering* adalah suatu teknik bagaimana membuat prediksi serta rekomendasi tentang minat seorang pengguna berdasarkan pengumpulan informasi atau opini dari banyak pengguna. Informasi yang diberikan pengguna tersebut merupakan tingkat kepuasan pengguna terhadap sebuah produk (*item*) tertentu dan dikenal sebagai *rating*. Dalam banyak kasus data informasi, sebagian besar pengguna tidak *merating* semua produk-produk yang ada, sehingga apabila data ini

direpresentasikan ke dalam bentuk matriks maka akan mengakibatkan banyaknya elemen (*entry*) dari matriks tersebut yang tidak memiliki nilai dan dapat dikatakan sebagai suatu matriks *sparse*. Untuk itulah di dalam *collaborative filtering* diperlukan metode yang mampu membangkitkan nilai prediksi dari elemen-elemen yang tidak *rating* oleh pengguna tersebut. Salah satu metode yang digunakan adalah *machine learning* dengan kategori pendekatan *unsupervised learning*, dimana menurut (Bishop, 2006), *machine learning* adalah suatu metode yang dapat belajar dari data historis sehingga menjadi cerdas, cerdas dalam artian memiliki kemampuan generalisasi terhadap data baru yang belum dipelajari sebelumnya, misalnya dalam memprediksi, mengklasifikasi, meranking, mengelompokkan, dan lain-lain. Dalam tataran metode/model, *learning* adalah proses penentuan parameter dari metode tersebut. Sedangkan pendekatan *unsupervised learning*, yaitu pendekatan pembelajaran dengan menggunakan data pelatihan (*training*) yang tidak dilengkapi dengan target pembelajaran. Tujuan pembelajarannya adalah menemukan komponen (variabel) tersembunyi pada data pelatihan, selanjutnya variabel-variabel tersembunyi tersebut digunakan untuk membangun model, disebut juga sebagai model laten variabel yang digunakan untuk memprediksi nilai *rating* pada suatu sistem rekomendasi.

Salah satu metode yang digunakan untuk membangun laten variabel model adalah faktorisasi matriks, di mana prinsip utama dari metode ini adalah jika diberikan suatu matriks $A_{m \times n}$ maka tentukan dua buah matriks $W_{m \times k}$ dan $H_{k \times n}$ sedemikian sehingga perkalian matriks W dan matriks H akan menghasilkan suatu matriks $\hat{A}_{m \times n}$ yang dapat mengaproksimasi matriks A , dapat juga dinyatakan sebagai $\hat{A} \approx A$. Dengan menggunakan metode faktorisasi ini kita dapat mereduksi dimensi matriks sehingga sistem menjadi lebih efisien.

Berbagai penelitian telah dilakukan dengan menerapkan teknik-teknik faktorisasi matriks. Penelitian-penelitian tersebut telah dipublikasikan secara luas, diantaranya adalah *Non-Negative Matrix Factorization (NMF)* (Lee & Seung,

2001), *Fast Maximum Margin Matrix Factorization (MMMF)* (Jason, 2005), *Probabilistic matrix factorization (PMF)* (Salakhutdinov, 2009) dan *Singular Value Decomposition (SVD)* (Karypis, 2009).

Dalam penelitian ini penulis melakukan analisis untuk mengetahui kinerja dari salah satu teknik faktorisasi matriks tersebut, yaitu *Non-Negative Matrix Factorization (NMF)* dengan penerapannya ke dalam sistem rekomendasi.

1.2. Perumusan Masalah

Berdasarkan latar belakang di atas, permasalahan dalam penelitian ini adalah :

- 1) Bagaimana menentukan parameter k dalam NMF sehingga menghasilkan matriks W dan matriks H yang optimal.
- 2) Sejauh mana kinerja NMF yang diterapkan dalam sistem rekomendasi.

1.3. Tujuan Penelitian

Sesuai dengan permasalahan, maka tujuan dari penelitian ini adalah sebagai berikut :

- 1) Menentukan parameter k dalam NMF sehingga menghasilkan matriks W dan H yang optimal.
- 2) Mengetahui sejauh mana kinerja NMF yang diterapkan dalam sistem rekomendasi.

1.4. Metode Penelitian

Penelitian ini dilakukan dalam empat tahapan yang berbeda, yaitu :

1. Studi literatur

Pada tahapan ini, proses yang dilakukan adalah pencarian informasi dan studi literatur yang diperlukan untuk pengumpulan data dan desain penelitian yang akan dibuat. Informasi diperoleh dengan melakukan studi

terhadap karya-karya ilmiah yang disajikan dalam bentuk buku, disertai ataupun artikel-artikel dan jurnal riset internasional serta beberapa referensi lain dari internet yang berisikan pembahasan mengenai sistem rekomendasi, *collaborative filtering*, *machine learning*, matriks, faktorisasi matriks, maupun literatur lain yang berhubungan dengan algoritma *Non-Negative Matrix Factorization (NMF)*.

2. Pemilihan data

Pada tahapan ini dilakukan pemilihan data yang akan digunakan dalam eksperimen, adapun data yang digunakan adalah basis data *rating* yang bersumber dari *MovieLens* (<http://www.grouplens.org/node/73>). Selanjutnya dilakukan analisis terhadap basis data *rating* tersebut guna merancang proses eksperimen yang akan dilakukan.

3. Implementasi dan pembuatan sistem

Pada tahap ini dilakukan implementasi metode *Non-Negative Matrix Factorization (NMF)* dalam melakukan reduksi dimensi pada basis data *rating* dengan menggunakan program matlab. Tahap ini terdiri atas beberapa sub tahap di antaranya yaitu pengolahan basis data *rating* untuk merepresentasikannya ke dalam bentuk matriks serta proses *multiplicative update* untuk dekomposisi matriks.

4. Uji coba (eksperimen), analisis dan evaluasi

Pada tahap ini dilakukan uji coba untuk menentukan nilai-nilai yang dipergunakan dalam perhitungan, antara lain adalah uji coba untuk menentukan parameter k dalam matriks W dan H sehingga menghasilkan matriks W dan H yang optimal, dimana matriks W dan H merupakan matrik non negatif sembarang, uji coba untuk menentukan kriteria penghentian iterasi, serta uji coba ringkasan akhir dengan menggunakan grafik.

5. Penyusunan laporan akhir tesis

Pada tahap ini dilakukan penyusunan tesis yang berisi dasar-dasar teori yang digunakan dan hasil-hasil yang diperoleh dalam melakukan eksperimen.

BAB II

LANDASAN TEORI

Bab ini menguraikan teori-teori yang merupakan landasan dalam *Non-Negative Matrix Factorization* (NMF) yang digunakan dalam penelitian. Penjelasan dibagi menjadi empat subbab. Pada subbab pertama membahas sistem rekomendasi, selanjutnya dalam subbab kedua membahas mengenai *collaborative filtering*, dan dalam subbab ketiga diuraikan mengenai *machine learning*, sedangkan pada subbab keempat diuraikan tentang teori matriks. Sementara pembahasan teori mengenai *Non-Negative Matrix Factorization* dibahas pada bab selanjutnya.

2.1 Sistem Rekomendasi

Dalam penguraian latar belakang telah dijelaskan beberapa definisi tentang sistem rekomendasi, di mana salah satunya adalah pendapat dari (Karypis, 2001) yang menyatakan bahwa sistem rekomendasi adalah suatu mekanisme yang dapat menyaring dan memberikan suatu informasi atau rekomendasi berdasarkan nilai (*rating*) atau *review* dari pengguna sebelumnya yang mempunyai keinginan yang sama (*user-neighbor*), sistem rekomendasi membandingkan profil pengguna terhadap beberapa referensi karakteristik yang telah dimiliki sistem, dan mencoba untuk memprediksi nilai suatu produk yang mungkin disukai dan belum dinilai oleh pengguna. Referensi karakteristik yang dimiliki sistem bisa berasal dari informasi produk (*content filtering*) atau kebiasaan setiap pengguna didalam sistem (*collaborative filtering*). Pada referensi karakteristik berbasis *content filtering*, profil dari pengguna dan produk harus didefinisikan terlebih dahulu. Sedangkan pada referensi berbasis *collaborative filtering* hanya bergantung dari

aktivitas pengguna sebelumnya tanpa harus mendefinisikan profil dari pengguna dan produknya terlebih dahulu.

Secara umum terdapat beberapa macam teknik dalam membangun suatu sistem rekomendasi (Sarwar, 2001), di antaranya adalah

1. *Knowledge Based Recommender System*

Sistem rekomendasi ini dibangun berdasarkan pengetahuan tentang pengguna atau produk untuk membuat rekomendasinya. Sistem ini tidak tergantung pada data statistical *rating* suatu produk yang diperoleh dari sekumpulan pengguna.

2. *Content Based Recommender System*

Pada sistem rekomendasi ini, rekomendasi suatu produk oleh seorang pengguna berdasarkan dari deskripsi produk tersebut serta profil dari keterkaitan seorang pengguna. Sistem rekomendasi *content based* menganalisa deskripsi dari setiap produk untuk mengidentifikasi produk mana yang mempunyai ketertarikan khusus terhadap seorang pengguna.

3. *Demographic Based Recommender System*

Rekomendasi *demographic* memanfaatkan fitur atau atribut pengguna. Teknik ini mencari sejumlah pengguna yang memiliki fitur yang mirip dan merekomendasikan produk yang telah disukai satu pengguna kepada pengguna yang lain yang fiturnya mirip tersebut.

4. *Collaborative Filtering*.

Pada *collaborative filtering*, rekomendasi berdasarkan pada korelasi yang didapat di antara pengguna yang telah *me-rating* atau bertransaksi dalam sistem. Teknik ini terbagi dua (Karypis, 2001), yaitu :

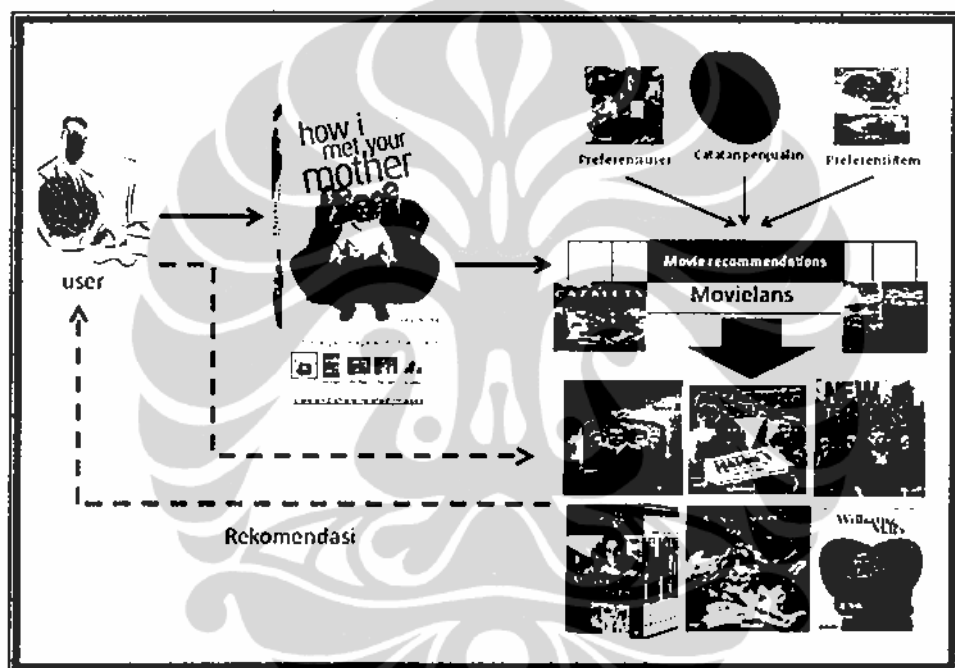
a. *User-based collaborative*

Sistem mencari sejumlah pengguna yang memiliki korelasi yang tinggi, kemudian sistem merekomendasikan sejumlah produk yang mungkin disukai oleh pengguna berdasarkan korelasi tersebut. Contoh apabila pengguna A menyukai produk 1, 2, 3 dan pengguna B menyukai produk 1,

2, 4, maka rekomendasi yang diberikan terhadap pengguna B adalah produk 3 dan rekomendasi terhadap pengguna A adalah produk 4.

b. Item-based collaborative

Kalau pada *user-based* yang dicari adalah korelasi antar pengguna, pada *item-based* korelasi yang dicari adalah antar produk yang disukai oleh pengguna kemudian produk yang berkorelasi tersebut direkomendasikan terhadap sejumlah pengguna lainnya.



Gambar 2.1 *Ilustrasi Sistem Rekomendasi*

Gambar 2.1 di atas merupakan ilustrasi bagaimana suatu sistem rekomendasi bekerja. Pada gambar terlihat seorang pengguna sedang mencari dan memilih judul film yang sesuai dengan kebutuhannya pada salah satu alamat *web* (situs *online*) melalui media internet. Saat ini terdapat beberapa alamat situs *web* yang menjual maupun menyewakan film-film secara *online* antara lain yaitu *Movielans.org*, *Netflix.com*, dan *Amazon.com*. Ketika pengguna memilih film

yang berjudul “*how I met your mother*” sebagai film yang disukainya, maka sistem segera memproses pilihan pengguna tersebut dengan membandingkan pilihan pengguna tersebut terhadap data yang telah dimiliki oleh sistem. Pada Gambar 2.1 terlihat bahwa data yang digunakan sistem berasal dari preferensi pengguna, preferensi produk dan catatan penjualan. Berdasarkan hasil proses inilah sistem akan memprediksi dan merekomendasikan produk-produk (judul film) yang mungkin akan disukai oleh seorang pengguna.

2.2 Collaborative Filtering

Collaborative filtering adalah salah satu teknik bagaimana membuat prediksi atau rekomendasi tentang minat seorang pengguna dengan pengumpulan informasi atau opini dari banyak pengguna. Prinsip kerja dari *collaborative filtering* adalah memberikan prediksi atau rekomendasi produk terhadap pengguna berdasarkan pada gabungan berbagai macam opini pengguna-produk (*user-item*) yang telah diberikan yang memiliki kemiripan, opini diberikan secara *eksplisit* oleh pengguna berupa suatu nilai, biasanya dalam skala kuantitatif tertentu, atau bisa juga secara *implisit* yang diperoleh dari arsip transaksi. *Collaborative filtering* bertujuan untuk memprediksi keinginan pelanggan berdasarkan kegiatan yang dilakukan sekelompok pelanggan yakni pembelian atau pemberian nilai terhadap suatu produk (Sarwar, 2001).

Selanjutnya (Pilaszy, 2009), menyatakan bahwa dalam *collaborative filtering*, hubungan pengguna, produk dan *rating* dapat dinyatakan dalam *cartesian product* yang berbentuk tripel (U, I, R) di mana :

$U = \{1, 2, \dots, m\}$; U menyatakan indeks pengguna, m adalah jumlah pengguna.

$I = \{1, 2, \dots, n\}$; I menyatakan indeks produk, n adalah jumlah produk.

$R = \{1, 2, 3, 4, 5\}$; R adalah nilai produk.

Triple produk kartesian (U,I,R) dapat dinyatakan dalam bentuk (u, i, r) , yang artinya pengguna u merating film i dengan skor nilai r .

196	242	3
186	302	3
22	377	1
244	51	2
166	346	1
298	474	4
115	265	2
253	465	5
305	451	3
6	86	3
62	257	2
286	1014	5
200	222	5
210	40	3
224	29	3
303	785	3
122	387	5
194	274	2
291	1042	4

Tabel 2.1 Contoh bentuk data tripel

Dalam penelitian Graus (2011), dikatakan bahwa data awal *collaborative filtering* diperoleh dari data *rating* yang dikonversi ke dalam bentuk matriks, di mana baris-barisnya menyatakan indeks pengguna, sedangkan kolom-kolomnya menyatakan indeks produk, entri r_{ij} merupakan *rating* yang diberikan pengguna i terhadap produk j .

Saat ini teknik *collaborative filtering* telah banyak diaplikasikan dalam berbagai bidang ilmu pengetahuan, baik oleh kalangan akademisi maupun dalam dunia industri. Hal ini dikarenakan teknik ini merupakan teknik yang sederhana dan efektif dalam menganalisa proses yang terjadi dalam suatu sistem rekomendasi (Hao, 2009).

Terdapat dua tahapan proses yang dilakukan pada teknik *collaborative filtering* dalam sistem rekomendasi, yaitu :

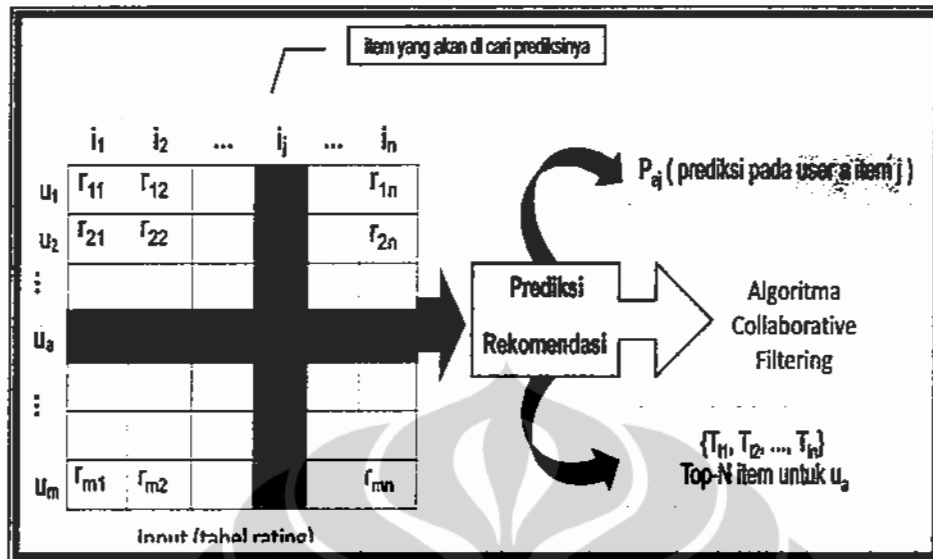
1. Prediksi

Prediksi opini akan diberikan oleh sistem dalam skala bilangan yang sama, di mana p_{aj} adalah prediksi *rating* produk j oleh pengguna a . Pada tahapan ini dilakukan pelatihan dengan menggunakan suatu metode terhadap data pelatihan (*training*), dengan tujuan untuk memperoleh suatu *model* yang paling optimal, optimal dalam artian memperoleh akurasi dengan nilai kesalahan (*error*) yang paling kecil.

2. Rekomendasi.

Memberikan rekomendasi berupa daftar produk dengan nilai prediksi tertinggi yang mungkin akan disukai oleh pengguna a . Hal ini sering di sebut juga *top-N recommendation*. Pada tahap ini setelah mendapatkan *model* yang paling optimal, model tersebut diujikan pada data pengujian, dengan tujuan untuk mendapatkan hasil peringkat sebagai *output* dari pemilihan data pengujian tersebut.

Gambar 2.2 menunjukkan diagram kerja dari proses sistem rekomendasi yang menggunakan pendekatan *collaborative filtering* dengan profile berbentuk matriks *rating* pengguna-produk berukuran $m \times n$, dimana m adalah jumlah pengguna dan n adalah jumlah produk sebagai input. Pengguna aktif (u_a) merupakan pengguna yang akan mencari produk yang mungkin disukainya dengan menggunakan algoritma *collaborative filtering*



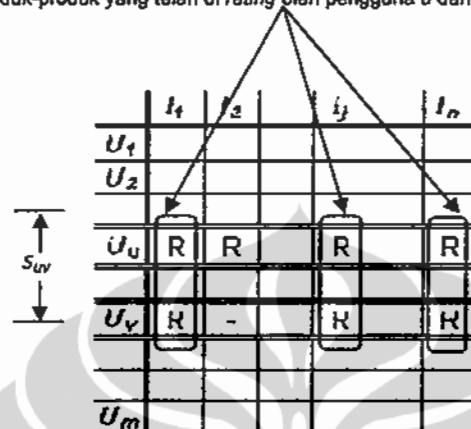
Gambar 2.2 Proses collaborative filtering

Pada prinsipnya terdapat dua pendekatan yang dapat diterapkan pada teknik *collaborative filtering* yaitu pendekatan yang berbasis memori (*memory-based collaborative filtering*) dan pendekatan yang berbasis model (*model-based collaborative filtering*). Dalam pendekatan yang berbasis memori hal utama yang dilakukan adalah menggunakan data *rating* untuk dihitung kemiripannya (*similarity*). Kemiripan ini mungkin terdapat pada pengguna ataupun pada produk. Berdasarkan nilai kemiripan yang diperoleh inilah kemudian yang digunakan untuk menentukan nilai prediksi atau rekomendasi. Pendekatan *memory-based collaborative filtering* terbagi ke dalam dua kategori yaitu *user-based collaborative filtering* dan *item-based collaborative filtering*. Sarwar (2001).

Dalam pendekatan *user-based collaborative filtering*, proses pembentukan *user-neighborhood* perhitungan prediksi dilakukan dengan menghitung kemiripan antara pengguna, kemudian untuk memprediksi produk-produk yang belum di nilai (*unknown rating*) dengan merata-ratakan nilai yang diketahui pada test produk melalui pengguna yang sama (*user based*). Perhitungan hanya dilakukan

pada himpunan produk-produk yang telah di *rating* oleh kedua pengguna yang dibandingkan, seperti yang ditunjukkan oleh Gambar 2.3

Kemiripan antara pengguna hanya dihitung dari produk-produk yang telah di *rating* oleh pengguna u dan pengguna v .



Gambar 2.3 Pembentukan *user-neighborhood*

Salah satu pengukuran kemiripan adalah dengan menghitung sudut antara pengguna. Misalkan pengguna yang dibandingkan adalah u dan v , dianggap sebagai vektor-vektor baris dengan anggotanya adalah nilai *rating* yang terdapat pada baris tersebut (Deshpande, 2004). S_{uv} menyatakan nilai kemiripan antara vektor u dan vektor v . Persamaan yang dapat digunakan antara lain adalah persamaan *adjusted-cosine similarity* seperti dalam persamaan (2.1).

$$\cos \alpha = \text{sim}(u,v) = \frac{\sum_{i \in I} (r_{i,u} - \bar{r}_u)(r_{i,v} - \bar{r}_v)}{\sqrt{\sum_{i \in I} (r_{i,u} - \bar{r}_u)^2} \sqrt{\sum_{i \in I} (r_{i,v} - \bar{r}_v)^2}} \quad (2.1)$$

Dimana

$\text{sim}(u,v)$: kemiripan antara pengguna u dan pengguna v ,

i : himpunan produk i yang di *rating* oleh pengguna u dan pengguna v ,

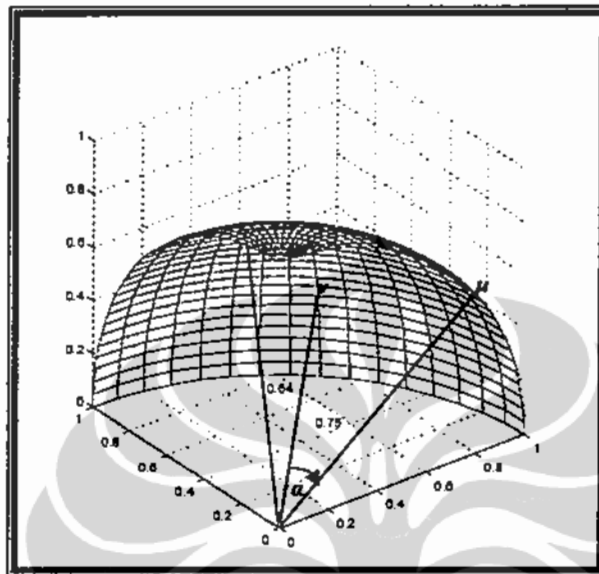
$r_{i,u}$: *rating* pengguna u pada produk i ,

$r_{i,v}$: *rating* pengguna v pada produk i

\bar{r}_u : rata-rata *rating* pengguna u ,

\bar{r}_v : rata-rata *rating* pengguna v .

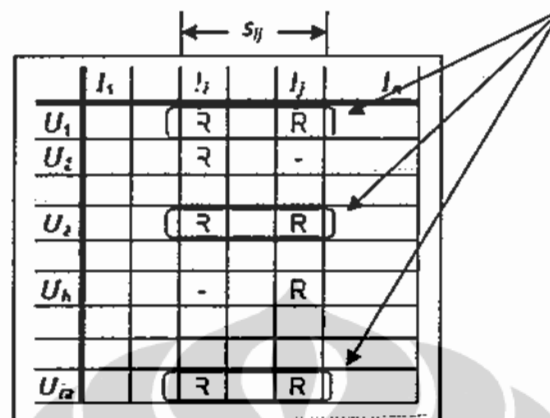
Antara vektor u dan vektor v membentuk sudut α dimana nilai cosinus dari sudut tersebut menunjukkan kemiripan dari kedua vektor tersebut.



Gambar 2.4 Contoh cosinus antar vektor

Sedangkan pendekatan *item based collaborative filtering* menjadikan produk-produk yang telah *dirating* oleh pengguna menjadi dasar perhitungan. Algoritma ini melakukan perhitungan bagaimana kemiripan produk-produk yang telah *dirating* dengan produk-produk lain dan selanjutnya dipilih sekelompok produk yang mempunyai kemiripan dengan produk yang telah *dirating*. Nilai kemiripan tersebut dicatat untuk dijadikan nilai bobot untuk memprediksi nilai *rating* pada produk target. Dasar perhitungan kemiripan antara dua produk i dan j adalah terlebih dahulu mengisolasi pengguna-pengguna yang telah *me-rating* keduanya kemudian teknik *item similarity* diterapkan untuk memperoleh nilai kemiripan seperti ditunjukkan Gambar 2.5

Kemiripan antara produk dihitung hanya pada produk-produk yang kedua-duanya dirating



Gambar 2.5 Pembentukan *item-neighborhood*

Nilai kemiripan produk-produk dapat dihitung dengan menggunakan persamaan *adjusted-cosine similarity* seperti halnya menghitung kemiripan antara pengguna-pengguna, yang berbeda adalah orientasinya, yaitu membandingkan antara dua produk, misalnya produk i dan produk j . Sehingga produk-produk tersebut dianggap sebagai vektor kolom dengan anggota nilai rating pada kolom tersebut (Deshpande, 2004). Persamaan (2.2) menunjukkan hal tersebut.

$$\cos \alpha = \text{sim}(i,j) = \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_i)(r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U} (r_{u,i} - \bar{r}_i)^2} \sqrt{\sum_{u \in U} (r_{u,j} - \bar{r}_j)^2}} \quad (2.2)$$

Di mana :

- $\text{sim}(i,j)$: kemiripan antara produk i dan produk j
- u : himpunan pengguna u yang merating produk i dan produk j
- $r_{u,i}$: nilai pengguna u pada produk i
- $r_{u,j}$: nilai pengguna u pada produk j
- \bar{r}_i : rata-rata nilai pada produk i
- \bar{r}_j : rata-rata nilai pada produk j

Untuk lebih memperjelas proses yang terjadi dalam *collaborative filtering*, diberikan ilustrasi sederhana seperti Gambar 2.5. Diasumsikan ada lima pengguna yaitu u_1, u_2, \dots, u_5 dan lima produk i_1, i_2, \dots, i_5 . Yang ingin dilakukan adalah memprediksi nilai *rating* yang diberikan pengguna u_1 terhadap produk i_5 . Akan di cari beberapa pengguna yang sudah memberi nilai *rating* untuk i_5 dan paling mirip dengan u_1 . Dalam kasus ini ditemukan yang paling dekat dengan u_1 adalah u_2 dan u_3 . Karena kedua pengguna sudah memberikan nilai *rating* 7 pada i_5 , maka sistem akan memberikan nilai prediksi u_1 untuk i_5 sebesar 7 juga.

	i_1	i_2	i_3	i_4	i_5
u_1	4	6	4	6	?
u_2	4	6	4	6	7
u_3	4	6	4	6	7
u_4	5	5	5	5	4
u_5	5	5	5	5	4

Gambar 2.6. Ilustrasi proses menentukan nilai prediksi suatu produk

Setelah nilai kemiripan antar produk atau pengguna diketahui maka langkah selanjutnya adalah menentukan nilai prediksi untuk produk yang belum di nilai oleh pengguna. Persamaan yang digunakan adalah persamaan *weighted sum* (penjumlahan terbobot), yaitu :

$$P_{(u,i)} = \frac{\sum_{i \in M} (r_{u,i} \cdot s_{i,j})}{\sum_{i \in M} |s_{i,j}|} \quad , \quad (2.3)$$

di mana :

- $P_{(u,i)}$: prediksi untuk pengguna u pada produk j ,
- M : himpunan produk yang mirip dengan produk j ,
- r_{ui} : nilai *rating* pengguna u pada produk i ,
- s_{ij} : nilai kemiripan antara produk i dan produk j .

Setelah model diperoleh dilakukan pengujian keakurasiannya dengan menggunakan data penguji sehingga didapat nilai keakurasiannya yang paling optimum. Adapun persamaan yang digunakan untuk mengetahui tingkat keakurasiannya adalah dengan menggunakan persamaan *root mean square error (RMSE)*.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (p_{u,i} - r_{u,i})^2} \quad , \quad (2.4)$$

di mana :

- n : banyaknya *entri* yang *dirating* oleh pengguna
- p_{ui} : *rating* prediksi dari pengguna u terhadap produk i pada data penguji
- r_{ui} : *rating* sebenarnya yang diberikan pengguna u pada produk i

Hal ini sesuai dengan pendapat (*Khoshgoftaar, 2009*) yang menyatakan bahwa *RMSE* adalah satuan ukur yang digunakan untuk menilai keakuratan kinerja sistem rekomendasi. Selain itu (*Asela, 2009*) juga berpendapat bahwa sebagai satuan ukur yang digunakan dalam sistem rekomendasi adalah dengan *RMSE*, yaitu metode yang mengukur perbedaan di antara preferensi prediksi dengan preferensi aktual dari produk-produk, dimana semakin rendah nilai *RMSE* yang didapat, maka sistem semakin akurat dalam memprediksi *rating*.

Berbeda dengan pendekatan yang berbasis memori (*memory-based collaborative filtering*), dalam pendekatan yang berbasis model (*model-based collaborative filtering*) diperlukan contoh data pelatihan (*training*) untuk memprediksi produk-produk yang terdapat pada data penguji (*testing*). Pendekatan yang berbasis model mampu memprediksi nilai untuk produk-produk yang tidak pernah *dirating* oleh pengguna pada data pelatihan, model akan diperoleh dengan menerapkan metode ini pada data pelatihan, selanjutnya model digunakan untuk memprediksi produk-produk yang terdapat pada data penguji.

(Khoshgoftaar, 2009) menyatakan bahwa kemampuan pendekatan yang berbasis model dalam membuat pemodelan memiliki beberapa kelebihan, di antaranya adalah kemampuannya dalam menyelesaikan masalah data *sparse* yang jauh lebih baik dibandingkan bila menggunakan pendekatan yang berbasis memori. Selain itu teknik pendekatan yang berbasis model juga memberikan kemudahan dalam memproses data dalam ukuran dimensi yang jauh lebih besar. (SongJie Gong, 2009) dalam penelitiannya juga menyatakan bahwa kemampuan *collaborative filtering* berbasis model dalam menangani *sparsity* dan *scalability* data lebih baik bila dibandingkan dengan metode *collaborative filtering* yang berbasis memori.

Salah satu metode yang digunakan dalam *collaborative filtering* dengan teknik pendekatan yang berbasis model (*model-based collaborative filtering*) adalah *machine learning*.

2.3 Machine Learning

Machine learning adalah suatu metode yang dapat belajar dari data historis sehingga menjadi cerdas, cerdas dalam artian memiliki kemampuan generalisasi terhadap data baru yang belum dipelajari sebelumnya, misalnya dalam memprediksi, mengklasifikasi, meranking, mengelompokan, dan lain-lain. Dalam tataran metode/model, *learning* adalah proses penentuan parameter dari metode tersebut (Bishop, 2006).

Dalam *machine learning* secara garis besar metode pembelajaran dibagi ke dalam dua pendekatan, yaitu :

1. Pembelajaran dengan pengawasan (*supervised learning*)

Pada pendekatan *supervised learning* pembelajaran menggunakan data pelatihan yang dilengkapi dengan target pembelajaran. Tujuan dari pendekatan *supervised learning* adalah membangun model yang dapat menghasilkan *output* yang benar jika diberikan *input* data baru.

2. Pembelajaran tanpa pengawasan (*unsupervised learning*).

Pada pendekatan *unsupervised learning* data pelatihan yang digunakan tidak dilengkapi dengan target pembelajaran. Tujuan pembelajaran adalah membangun model yang dapat menemukan variabel atau komponen tersembunyi pada data pelatihan sehingga model yang terbentuk dapat digunakan untuk membuat sistem rekomendasi.

Dalam penelitian ini metode yang digunakan untuk menemukan komponen (variabel) tersembunyi adalah *machine learning* dengan kategori pendekatan *unsupervised learning* yang dikenal sebagai *latent variable models*. Pada metode ini, biasanya data direpresentasikan dalam bentuk matriks, di mana baris-baris pada matriks tersebut menyatakan pengguna, sementara kolom-kolomnya menyatakan produk. Secara umum matriks tersebut dapat ditulis dalam bentuk $R_{m \times n}$ di mana r_{ij} adalah *rating* yang diberikan oleh pengguna i pada produk j . Tabel 2.2 berikut merupakan contoh bentuk matriks yang dibentuk dari data *rating* suatu produk :

	Produk 1	Produk 2	Produk 3	...	Produk n
Pengguna 1	r_{11}	r_{12}	r_{13}	...	r_{1n}
Pengguna 2	r_{21}	r_{22}	r_{23}	...	r_{2n}
Pengguna 3	r_{31}	r_{32}	r_{33}	...	r_{3n}
.
Pengguna m	r_{m1}	r_{m2}	r_{m3}	...	r_{mn}

Tabel 2.2 Matriks yang dibentuk dari data rating suatu produk

Sebagian besar data *rating* yang direpresentasikan ke dalam bentuk matriks menghasilkan suatu bentuk matriks yang *sparse*, hal ini dikarenakan pada umumnya pengguna tidak memberikan penilaian terhadap semua produk yang ada. (Sarwar, 2001) menyatakan bahwa untuk mengekstrak variabel tersembunyi seperti pada *latent semantic analysis* (LSA) hanya dapat dilakukan dengan sejumlah trik, misalnya mengganti variabel yang tidak diketahui dengan nol

(*imputation*), akan tetapi hal ini dapat menyebabkan peningkatan jumlah data yang tidak dilakukan secara benar sehingga akan merusak data. Untuk itu diperlukan suatu metode yang mampu merumuskan pemodelan yang dapat diperoleh berdasarkan bentuk matriks *sparse* tersebut. Beberapa metode yang populer dalam *latent variable models* yang dapat digunakan adalah berdasarkan faktorisasi matriks. Metode ini menjadi populer karena memberikan skalabilitas yang baik dengan akurasi yang prediktif.

Beberapa metode faktorisasi matriks yang sering digunakan antara lain adalah :

1. *Probability Matrix Factorization* (PMF)
2. *Regularized Incremental Simultaneous Matrix Factorization* (RISMF)
3. *Maximum-Margin Matrix Factorization* (MIMMF)
4. *Non-Negative Matrix Factorization* (NMF)

Dalam pembahasan berikutnya tesis ini secara spesifik akan menerapkan salah satu algoritma *collaborative-filtering* yang berbasis model, yaitu metode *Non-Negative Matrix Factorization* (NMF), di mana pembahasan dan uraian mengenai metode ini akan dibahas lebih lengkap pada bab tiga.

2.4 Teori Matriks

Dalam penulisan tesis ini teori matriks merupakan landasan teori yang banyak diaplikasikan, definisi dan teorema yang digunakan dalam tesis ini di ambil dari berbagai referensi, pembuktian definisi dan teorema tidak diberikan dalam tesis ini, tetapi dapat di lihat pada referensi yang digunakan..

Berikut adalah beberapa definisi dan teorema yang akan digunakan pada bab selanjutnya

Definisi 2.4.1 (*Hadarmard product*)

Jika A adalah sebuah matriks $m \times n$ dan B adalah sebuah matriks $m \times n$, maka hasil kali $A \odot B$ (*Hadarmard product*) adalah matriks C yang berukuran $m \times n$ yang entri-entrinya diperoleh dengan mengalikan entri-entri yang bersesuaian pada matriks A dan matriks B

Dalam notasi matriks, jika $A = [a_{ij}]$ adalah sembarang matriks berukuran $m \times n$ dan $B = [b_{ij}]$ adalah sembarang matriks berukuran $m \times n$,

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \quad B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{bmatrix}$$

maka entri-entri $(A \odot B)_{ij}$ pada baris ke- i dan kolom ke- j dari $A \odot B$ diberikan oleh $C_{ij} = a_{ij} \times b_{ij}$

$$\begin{aligned} A \odot B &= \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \odot \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{bmatrix} \\ &= \begin{bmatrix} a_{11} \times b_{11} & a_{12} \times b_{12} & \cdots & a_{1n} \times b_{1n} \\ a_{21} \times b_{21} & a_{22} \times b_{22} & \cdots & a_{2n} \times b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} \times b_{m1} & a_{m2} \times b_{m2} & \cdots & a_{mn} \times b_{mn} \end{bmatrix} \end{aligned}$$

(Anton, 2000)

Definisi 2.4.2 (basis)

Jika V adalah sembarang ruang vektor dan $S = \{v_1, v_2, \dots, v_n\}$ merupakan himpunan berhingga dari vektor-vektor pada V , maka S dinyatakan sebagai basis untuk V jika :

- a) S adalah bebas linear
- b) S span di V

(Anton, 2000)

Definisi 2.4.3 (Matriks non negatif)

Suatu matriks $A = [a_{ij}]$, disebut matriks non negative jika untuk setiap $a_{ij} \geq 0$

(Anton, 2000)

Definisi 2.4.4 (Matriks simetrik)

Suatu matriks $A = [a_{ij}]$, disebut simetrik jika $A^T = A$

(Anton, 2000)

Definisi 2.4.5 (Bentuk kuadrat)

Suatu persamaan kuadrat dengan dua variable x dan y adalah suatu persamaan berbentuk :

$$ax^2 + 2bxy + cy^2 + dx + ey + f = 0 \quad , \quad (1)$$

bentuk persamaan (1) dapat dinyatakan dalam bentuk

$$\begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} d & e \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + f = 0$$

misalkan $x = \begin{pmatrix} x \\ y \end{pmatrix}$ dan $A = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$ maka bentuk

$$x^T Ax = ax^2 + 2bxy + cy^2$$

(Anton, 2000)

Definisi 2.4.6 (Matriks semidefinit positif)

Jika A adalah matriks simetri, maka matriks A disebut matriks semidefinit positif jika bentuk kuadrat $x^T Ax \geq 0$ untuk semua vektor x

Contoh :

Matriks $A = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix}$ merupakan matriks semidefinit positif karena bentuk kuadrat

$$\begin{aligned} x^T Ax &= (x \ y) \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ &= x^2 + 4xy + 4y^2 \\ &= (x + 2y)^2 \geq 0 \end{aligned}$$

(Anton, 2000)

Definisi 2.4.7 (Ruang hasil kali dalam)

Sebuah ruang hasil kali dalam pada ruang vektor riil V adalah fungsi yang memetakan bilangan riil (u, v) dengan masing-masing pasangan vektor u dan v pada V sedemikian hingga aksioma-aksioma berikut dipenuhi untuk semua vektor u, v dan w di V dan juga semua skalar k .

- a) $(u, v) = (v, u)$ aksioma simetris
- b) $(u + v, w) = (u, w) + (v, w)$ aksioma penambahan

- c) $(k u, v) = k(u, v)$ aksioma kehomogenan
 d) $(v, v) \geq 0, (v, v) = 0 \leftrightarrow v = 0$ aksioma kepositifan

Definisi 2.4.6

Jika V adalah sebuah ruang hasil kali, maka *norm* vektor u yang dinyatakan oleh $\|u\|$, dapat didefinisikan sebagai berikut :

$$\|u\| = \langle u, u \rangle^{1/2}$$

Definisi 2.4.7 (Norm)

Norm adalah suatu fungsi $\|\cdot\|: \mathbb{R}^n \rightarrow \mathbb{R}$ yaitu fungsi yang memetakan suatu vektor ke panjangnya yang bernilai riil yang memenuhi sifat berikut, misal x dan y adalah vektor dan α adalah skalar, maka

1. $\|x\| \geq 0$ dan $\|x\| = 0$ jika dan hanya jika $x = 0$
2. $\|x + y\| \leq \|x\| + \|y\|$
3. $\|\alpha x\| \leq |\alpha| \cdot \|x\|$

Definisi 2.4.8 (Norm matriks)

Norm dari suatu matriks merupakan *norm* vektor yang diterapkan dalam ruang vektor berdimensi mn untuk matriks berukuran $m \times n$, misal A dan B adalah matriks, maka :

1. $\|A\| \geq 0$ dan $\|A\| = 0$ jika dan hanya jika $A = 0$
2. $\|A + B\| \leq \|A\| + \|B\|$
3. $\|\alpha A\| \leq |\alpha| \cdot \|A\|$

1. $\|A\| \geq 0$ dan $\|A\| = 0$ jika dan hanya jika $A = 0$
2. $\|A + B\| \leq \|A\| + \|B\|$
3. $\|\alpha A\| \leq |\alpha| \|A\|$

Dalam penulisan tesis ini norm yang digunakan adalah Norm Frobenius sebagaimana yang terdapat dalam (Roman, 2008)

Definisi 2.4.9 (Norm Frobenius)

Jika $A = [a_{ij}]$ adalah matrik ukuran $m \times n$, maka Norm Frobenius dari matrik A adalah :

$$\|A\|_F = \left(\sum_{i,j} a_{i,j}^2 \right)^{\frac{1}{2}}$$

Contoh :

Misalkan $A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$ maka :

$$\|A\|_F = \left(a_{11}^2 + a_{12}^2 + a_{13}^2 + a_{21}^2 + a_{22}^2 + a_{23}^2 + a_{31}^2 + a_{32}^2 + a_{33}^2 \right)^{\frac{1}{2}}$$

BAB III

NON-NEGATIVE MATRIX FACTORIZATION

Bab ini terbagi atas lima subbab. Subbab pertama menguraikan tentang faktorisasi matriks, selanjutnya dalam subbab kedua dibahas mengenai faktorisasi matriks dalam sistem rekomendasi dan subbab ketiga berisikan pembahasan tentang *Non-Negative Matrix Factorization (NMF)*. Penjelasan tentang *multiplicative update rule* terdapat dalam subbab keempat. Sedangkan pada subbab kelima terdapat uraian mengenai algoritma *multiplicative update*.

3.1 Faktorisasi Matriks

Faktorisasi matriks adalah suatu proses memecahkan atau menguraikan suatu matriks menjadi beberapa buah matriks. Tujuan dari faktorisasi ini adalah memperoleh matriks-matriks yang memiliki struktur tertentu sehingga beberapa operasi akan menjadi lebih sederhana (efisiensi dari segi waktu komputasi) atau jumlah variabel yang jauh lebih kecil (efisiensi dari segi ruang memori). Secara umum metode faktorisasi matriks terbagi menjadi dua kelompok, yaitu :

1. Metode langsung

Beberapa faktorisasi matriks dengan menggunakan metode langsung yang banyak digunakan antara lain adalah :

a) Faktorisasi LU

Faktorisasi yang menguraikan suatu matriks $A_{m \times n}$ menjadi dua buah matriks, yaitu matriks segitiga bawah ($L_{m \times m}$) dan matriks segitiga atas ($U_{m \times n}$)

b) Faktorisasi Cholesky

Faktorisasi yang menguraikan suatu matriks definit positif $A_{m \times n}$ menjadi dua buah matriks, yaitu matriks segitiga bawah ($G_{m \times m}$) dan matriks segitiga atas ($G_{m \times n}^T$)

c) Faktorisasi QR

Faktorisasi yang menguraikan suatu matriks $A_{m \times n}$ menjadi dua buah matriks, yaitu matriks orthogonal ($Q_{m \times m}$) dan matriks segitiga atas ($R_{m \times n}$)

2. Metode aproksimasi

Beberapa faktorisasi matriks dengan menggunakan metode aproksimasi yang banyak digunakan adalah :

a) *Singular Value Decomposition* (SVD)

Faktorisasi yang menguraikan suatu matriks $A_{m \times n}$ menjadi tiga buah matriks, yaitu matriks orthogonal ($S_{m \times m}$), matriks diagonal ($U_{n \times n}$) dan matriks orthogonal ($V_{n \times m}$)

b) *Non-Negative Matrix Factorization* (NMF)

Faktorisasi yang menguraikan suatu matriks $A_{m \times n}$ menjadi dua buah matriks, yaitu matriks $W_{m \times k}$ dan matriks $H_{k \times n}$.

3.2 Faktorisasi Matriks dalam Sistem Rekomendasi

Saat ini teknik faktorisasi matriks merupakan metode yang sangat populer dalam sistem rekomendasi. Dalam penelitian ini teknik faktorisasi matriks yang digunakan adalah metode aproksimasi, dimana prinsip utama dari metode ini adalah jika diberikan suatu matriks $A_{m \times n}$ maka terlebih dahulu ditentukan dua buah matriks $W_{m \times k}$ dan $H_{k \times n}$ sedemikian sehingga perkalian matriks W dan matriks H akan menghasilkan suatu matriks $\hat{A}_{m \times n}$ yang dapat mengaproksimasi matriks A , dapat juga dinyatakan sebagai $\hat{A} \approx A$.

(Jason, 2005) Menyatakan bahwa, jika A adalah suatu matriks target berukuran $m \times n$, maka matriks A dapat diaproksimasi dengan suatu matriks \hat{A} , di mana matriks \hat{A} merupakan hasil perkalian dua buah matriks W dan matriks H atau ditulis $\hat{A} = WH^T$ dengan $W \in R_{m \times k}$ dan $H \in R_{k \times n}$.

Misalkan $A_{m \times n}$ adalah sembarang matriks,

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

maka dalam *approximation factorization matrix* terlebih dahulu kita menentukan dua buah matriks sembarang misalnya $W_{m \times k}$ dan $H_{k \times n}$

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1k} \\ w_{21} & w_{22} & \cdots & w_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mk} \end{bmatrix} \quad H = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & h_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ h_{k1} & h_{k2} & \cdots & h_{kn} \end{bmatrix}$$

sedemikian sehingga $WH = \hat{A}$

$$\hat{A} = \begin{bmatrix} \hat{a}_{11} & \hat{a}_{12} & \cdots & \hat{a}_{1n} \\ \hat{a}_{21} & \hat{a}_{22} & \cdots & \hat{a}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{a}_{m1} & \hat{a}_{m2} & \cdots & \hat{a}_{mn} \end{bmatrix}$$

Maka, matriks \hat{A} inilah yang akan digunakan untuk mengaproksimasi matriks A , kita tulis sebagai $\hat{A} \approx A$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \approx \begin{bmatrix} \hat{a}_{11} & \hat{a}_{12} & \cdots & \hat{a}_{1n} \\ \hat{a}_{21} & \hat{a}_{22} & \cdots & \hat{a}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{a}_{m1} & \hat{a}_{m2} & \cdots & \hat{a}_{mn} \end{bmatrix}$$

Dalam penelitian ini matriks $W_{m \times k}$ merupakan matriks pengguna fitur dan matriks $H_{k \times n}$ adalah matriks fitur produk sedangkan k adalah banyaknya fitur yang dipilih dalam faktorisasi matriks.

Misalkan dibentuk dua buah vektor w_i dan h_j , di mana w_i adalah vektor yang entri-entrinya diambil dari baris ke- i pada matriks W ,

$$w_i = \begin{bmatrix} w_{i,1} \\ w_{i,2} \\ \vdots \\ w_{i,k} \end{bmatrix}$$

sedangkan h_j adalah vektor yang entri-entrinya diambil dari kolom ke- j pada matriks H ,

$$h_j = \begin{bmatrix} h_{1,j} \\ h_{2,j} \\ \vdots \\ h_{k,j} \end{bmatrix}$$

karena $WH = \hat{A}$, maka $w_i' h_j$ adalah entri matriks \hat{A} pada baris ke- i kolom ke- j atau ditulis $\hat{A} = \hat{a}_{ij} = w_i' h_j$ di mana \hat{a}_{ij} adalah prediksi *rating* untuk entri a_{ij} pada matriks A .

$$\hat{A} = \begin{bmatrix} \hat{a}_{11} & \hat{a}_{12} & \cdots & \hat{a}_{1n} \\ \hat{a}_{21} & \hat{a}_{22} & \cdots & \hat{a}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{a}_{m1} & \hat{a}_{m2} & \cdots & \hat{a}_{mn} \end{bmatrix}$$

3.3 Non-Negative Matrix Factorization (NMF)

Non-Negative Matrix Factorization merupakan metode faktorisasi matriks non negatif V yang berukuran $m \times n$ menjadi matriks W dan H yang bernilai non negatif dan masing-masing berukuran $m \times k$ dan $k \times n$. (Lee & Seung, 2001) menyatakan bahwa metode faktorisasi matriks non negatif didefinisikan sebagai, “Jika diberikan matriks non-negatif A , tentukan matriks non-negatif W dan H sedemikian hingga”

$$A \approx W.H \quad (3.1)$$

Misalkan terdapat matriks non negatif $A_{m \times n}$ yang merupakan kumpulan himpunan vektor berdimensi m , dan n adalah banyak vektor. Matriks $A_{m \times n}$ ini akan difaktorisasi menjadi matriks $W_{m \times k}$ dan matriks $H_{k \times n}$ dengan k bilangan bulat positif di mana k jauh lebih kecil dari nilai m dan n ($k \ll m$ dan n), hal ini tentu saja mengakibatkan ukuran matriks W dan matriks H akan lebih kecil dari ukuran matriks A .

$$\begin{array}{c}
 \left[\begin{array}{cccc} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{m1} & v_{m2} & \cdots & v_{mn} \end{array} \right] \approx \left[\begin{array}{cccc} w_{11} & w_{12} & \cdots & w_{1k} \\ w_{21} & w_{22} & \cdots & w_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mk} \end{array} \right] \left[\begin{array}{cccc} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & h_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ h_{k1} & h_{k2} & \cdots & h_{kn} \end{array} \right] \\
 A \qquad \qquad \qquad W \qquad \qquad \qquad H
 \end{array}$$

Gambar 3.1 Ilustrasi *Non-Negative Matrix Factorization (NMF)*

Berdasarkan uraian di atas dapat dikatakan bahwa NMF merupakan teknik reduksi dimensi di mana untuk merepresentasikan data yang besar yaitu sebanyak m hanya digunakan basis yang sedikit yaitu k dengan k jauh lebih kecil dari m dan n . Bentuk $A \approx WH$ pada Persamaan (3.1) dapat dijelaskan sebagai bentuk kesamaan antara matriks A dengan hasil perkalian dari matriks W dan H . Untuk mencapai

kondisi kesamaan antara A dengan WH tersebut, diperlukan suatu kriteria yang dapat dikatakan sebagai *Cost Function*. Beberapa model *Cost Function* dapat dibangun dengan pengukuran jarak antara dua matriks non-negatif A dan B , seperti dijelaskan dalam Persamaan (3.2) (Lee and Seung, 2001).

$$\|A - B\|^2 = \sum_y (A_y - B_y)^2 \quad (3.2)$$

Persamaan (3.2) menunjukkan aturan *Cost Function* untuk jarak antara A dan B yang memiliki batas bawah nol, dan kondisi batas bawah nol terpenuhi jika dan hanya jika $A = B$.

Dalam penelitian ini, proses pengukuran menuju kondisi $A \approx WH$ menggunakan aturan *Norm Frobenius* (Lee, Park, Ahn & Kim, 2009) yang dibuat berdasarkan aturan *Cost Function* seperti dijelaskan pada persamaan (3.2). Dengan aturan ini, terdapat dua buah matriks yang akan dihitung jarak keduanya, yaitu matriks A dengan perkalian antara matriks W dan H . Aturan *Norm Frobenius* yang digunakan dalam metode dekomposisi NMF dijelaskan pada persamaan (3.3) berikut.

$$f(W, H) = \|A - WH\|_F^2 = \sum_{i=1}^m \sum_{j=1}^n \left(A_{ij} - \sum_{l=1}^k W_{il} H_{lj} \right)^2 \quad (3.3)$$

Adapun bentuk umum dari *Norm Frobenius* adalah $\|A\|_F^2 = \sum_y |a_y|^2$. Nilai variabel A

menunjukkan suatu matriks, sehingga nilai *Norm Frobenius* dari A atau $\|A\|_F^2$ dinyatakan dalam bentuk jumlah kuadrat dari masing-masing elemen matriks penyusun matriks A .

Dengan demikian bentuk *Norm Frobenius* yang digunakan dalam metode NMF seperti pada persamaan (3.3) menunjukkan hasil perhitungan *Norm Frobenius* untuk selisih masing-masing elemen dari matriks A dengan matriks hasil perkalian

matriks W dan H . Selanjutnya permasalahan NMF dinyatakan dalam bentuk persamaan :

$$\min_{\substack{W, H \\ s.t. W \geq 0, H \geq 0, \forall i, j}} f(W, H) = \frac{1}{2} \|A - WH\|_F^2 \quad (3.4)$$

Matriks W dan H yang digunakan dalam meminimumkan fungsi di atas diperoleh dengan cara iteratif hingga fungsi tersebut mencapai suatu kondisi yang konvergen, dimana bentuk awal dari matriks W dan H ini adalah matriks sembarang non negatif.

3.4 Multiplicative Update Rule

Multiplicative Update berfungsi untuk meng-*update* nilai matriks W dan H untuk mencapai kondisi $A \approx WH$. Oleh karena itu terdapat suatu persoalan dimana nilai nilai $\|A - WH\|_F^2$ harus kecil. Terdapat beberapa algoritma yang dapat digunakan dalam menyelesaikan permasalahan NMF. Pendekatan yang populer dan banyak digunakan untuk menyelesaikan permasalahan ini adalah dengan menggunakan metode multiplikatif (*multiplicative method*), dimana dengan metode ini diharapkan nilai dari fungsi pada persamaan (3.4) tidak akan naik (*non-increasing*) (Lee and Seung, 2001).

Adapun bentuk persamaan *update rule* yang digunakan dinyatakan dalam persamaan (3.5) dan persamaan (3.6).

$$H_{ij} \leftarrow H_{ij} \odot \frac{(W^T A)_{ij}}{(W^T W H)_{ij}} \quad (3.5)$$

$$W_{ik} \leftarrow W_{ik} \odot \frac{(A H^T)_{ik}}{(W H H^T)_{ik}} \quad (3.6)$$

Proses perkalian dan pembagian dalam *update rule* dalam persamaan (3.5) dan persamaan (3.6) merupakan proses *hadarmard produc*, yaitu perhitungan perkalian dan pembagian matriksnya dilakukan dengan perkalian dan pembagian antar elemen matriks, seperti yang ditunjukkan oleh indeks ke-*ik* dan indeks ke-*kj* pada matriks *W* dan matriks *H*.

Teorema 3.1 Jarak *euclidean* $\|A - WH\|$ merupakan fungsi tidak naik (*nonincreasing*) dengan *update rule* :

$$H \leftarrow H \frac{(W^T A)}{(W^T W H)} \quad (3.7)$$

$$W \leftarrow W \frac{(A H^T)}{(W H H^T)} \quad (3.8)$$

Bukti, (Lee and Seung, 2009)

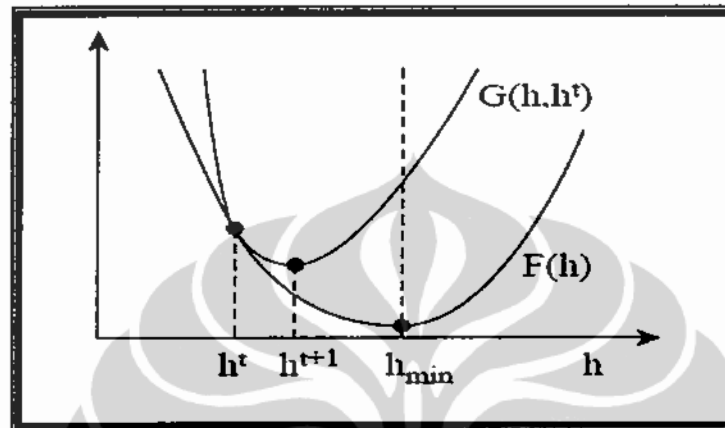
Untuk membuktikan teorema di atas akan diberikan definisi *auxiliary function*, *lemma 3.1* dan *lemma 3.2* sebagai berikut :

Definisi 3.1

$G(h, h')$ adalah *auxiliary function* untuk fungsi F jika memenuhi kondisi berikut :

$$G(h, h') \geq F(h), \quad G(h, h') = F(h) \quad (3.9)$$

Auxiliary function digunakan pada *lemma 3.1* dan *lemma 3.2* sebagaimana yang di ilustrasikan oleh gambar 3.3



Gambar 3.2 Ilustrasi dari *auxiliary function*

Lemma 3.1

Jika G adalah *auxiliary function* maka F *nonincreasing* terhadap *update* :

$$F^{t+1} = \arg \min_h G(h, h^t) \quad (3.10)$$

Bukti

$$F(h^{t+1}) \leq G(h^{t+1}, h^t) \leq G(h^t, h^t) = F(h^t)$$

Perhatikan bahwa $F(h^{t+1}) = F(h^t)$ jika h^t adalah local minimum dari $G(h, h^t)$. Jika turunan dari F ada dan kontinu pada *neighborhood* dari h^t , hal ini juga menyiratkan bahwa turunan dari $\nabla F(h^t) = 0$. Jadi, dengan iterasi *update* pada Persamaan (3.10) diperoleh barisan estimasi yang konvergen untuk minimum lokal $h_{min} = \arg \min_h F(h)$ dari fungsi objektif :

$$F(h_{min}) \leq \dots F(h^{t+1}) \leq F(h^t) \dots \leq F(h_2) \leq F(h_1) \leq F(h_0) \quad (3.11)$$

Selanjutnya akan ditunjukkan dengan mendefinisikan $G(h, h')$ adalah *auxiliary function* untuk $\|V - WH\|$, akan lebih sederhana untuk membuktikan teorema 3.1 dengan Persamaan (3.10)

Lemma 3.2

Jika $K(h')$ adalah matriks diagonal

$$K_{ab}(h') = \delta_{ab} (W'WH')_a / h'_a \quad (3.12)$$

Maka

$$G(h, h') = F(h') + (h - h')^T \nabla F(h') + \frac{1}{2} (h - h')^T K(h') (h - h') \quad (3.13)$$

adalah *auxiliary function* untuk

$$F(h) = \frac{1}{2} \sum_i \left(v_i - \sum_a w_{ia} h_a \right)^2 \quad (3.14)$$

Bukti:

Karena $G(h, h) = F(h)$, maka hanya akan ditunjukkan bahwa $G(h, h') \geq F(h)$, hal ini dilakukan dengan membandingkan persamaan

$$F(h) = F(h') + (h - h')^T \nabla F(h') + \frac{1}{2} (h - h')^T (W'W)(h - h') \quad (3.15)$$

terhadap persamaan (3.13) sehingga di dapatkan bentuk $G(h, h') \geq F(h)$ yang ekuivalen dengan bentuk persamaan

$$0 \leq (h - h')^T [K(h') - W'W] (h - h') \quad (3.16)$$

Untuk membuktikan persamaan (3.16) semidefinit positif, anggap matriks :

$$M_{ab}(h') = h'_a(K(h') - W'W)_{ab}h'_b \quad (3.17)$$

yang mengubah entri-entri dari $K - W^T W$, kemudian $K - W^T W$ merupakan semidefinit positif jika dan hanya jika M :

$$\begin{aligned} v' M v &= \sum_{ab} v_a M_{ab} v_b \\ &= \sum_{ab} h'_a (W'W)_{ab} h'_b v_a^2 - v_a h'_a (W^T W)_{ab} h'_b v_b \\ &= \sum_{ab} (W'W)_{ab} h'_a h'_b \left[\frac{1}{2} v_a^2 + \frac{1}{2} v_b^2 - v_a v_b \right] \\ &= \frac{1}{2} \sum_{ab} (W'W)_{ab} h'_a h'_b (v_a - v_b) \\ &\geq 0 \end{aligned}$$

Selanjutnya dapat ditunjukkan konvergensi dari teorema 3.1 dengan mengganti bentuk $G(h, h')$ pada persamaan (3.10) dengan bentuk $G(h, h')$ pada persamaan (3.13) sehingga diperoleh bentuk persamaan

$$h'^{+1} = h' - K(h')^{-1} \nabla F(h') \quad (3.18)$$

yang merupakan persamaan *update rule*.

Karena persamaan (3.13) adalah *auxiliary function* maka F tidak naik terhadap *update rule* di atas, hal ini sesuai dengan lemma 3.1. Dengan menuliskan entri-entri pada persamaan (3.18) secara eksplisit diperoleh bentuk persamaan

$$h_a^{t+1} = h_a^t \frac{(W^T x)_a}{(W^T W h^t)_a} \quad (3.19)$$

Dengan menggunakan aturan W dan H pada lemma 3.1 dan lemma 3.2, maka untuk W akan terbukti bahwa F juga tidak naik terhadap *update rule* tersebut.

3.5 Algoritma Multiplicative Update NMF

Berdasarkan *update rule multiplikatif* dalam persamaan (3.7) dan (3.8) di bentuk algoritma untuk menyelesaikan masalah pada NMF sebagai berikut

Multiplicative Update Algorithm For NMF

$V = \{(x, (i, j, r_{ij})) \mid u \in X, i \in Y, j \in I, r_{ij} \in T\}$; (file u.data)

Dimana :

V : Data awal
 x : pengamatan ke- x
 i : pengguna ke- i
 j : produk ke- j
 k : jumlah fitur
 r_{ij} : rating yang diberikan pengguna ke- i pada produk ke- j
 Tr = Data pelatihan
 Ts = Data Testing
 $m = 943$ % jumlah pengguna
 $n = 1682$ % jumlah produk
 $S = 100.000$ % jumlah data rating
 $X = \{1, 2, \dots, S\}$ %himpunan indeks pengamatan
 $Y = \{1, 2, \dots, m\}$ %himpunan indeks pengguna
 $I = \{1, 2, \dots, n\}$ %himpunan indeks produk
 $T = \{1, 2, 3, 4, 5\}$ %himpunan skala likert
 $Tr \subset V$ %(file ul.base)
 $Ts \subset V$ %(file ul.test)
 $Tr \cap Ts = \emptyset$ (disjoint)
 $Tr \cup Ts = V$


```

Multiplicative Update Algorithm For NMF (lanjutan)

V = load -ascii u.data %membaca semua data rating (V)
Tr = load -ascii ul.base %membaca semua data pelatihan
Ts = load -ascii ul.test %%membaca semua data penguji
[m n]= size(V);
[m1,n1]= size(Tr);
[m2,n2]= size(Ts);
For k = 1
  W = rand(m,k) %matriks non-negatif sembarang m x k
  H = rand(k,n) %matriks non-negatif sembarang k x n
  For i = 1 : 100

    
$$H \leftarrow H \odot \frac{(W^T A)}{(W^T W H) + eps}$$


    
$$W \leftarrow W \odot \frac{(A H^T)}{(W H H^T) + eps}$$


    
$$\hat{V} = W.H$$


    RMSE k =  $\sqrt{\frac{1}{n} \sum_{i=1}^n (p_{u,i} - r_{u,i})^2}$ 

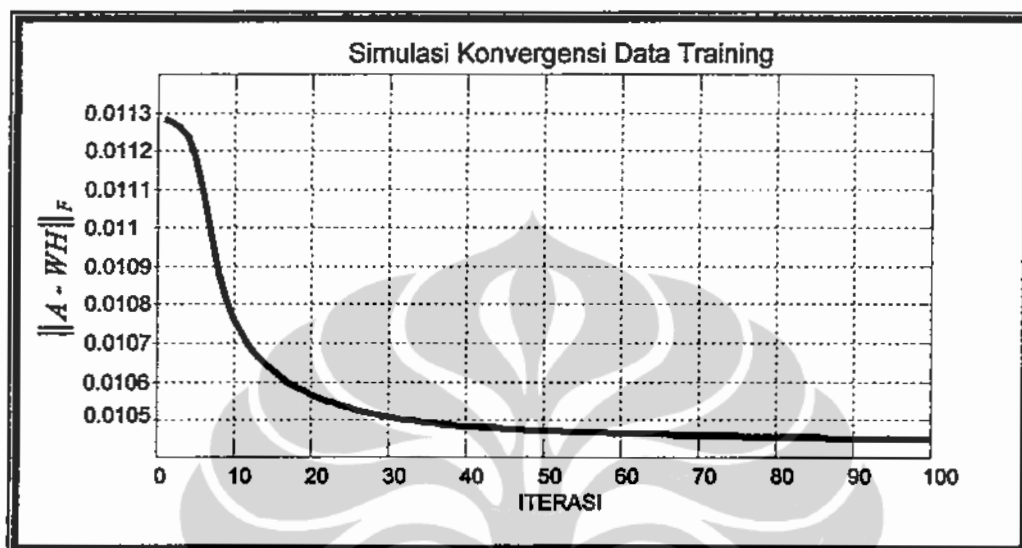
  end
end

```

Gambar 3.3. Algoritma Multiplicative Update untuk NMF

Untuk menghindari munculnya kasus pembagian sama dengan nol, maka dalam aplikasinya tiap-tiap penyebut dari persamaan multiplikatif tersebut ditambahkan dengan suatu parameter (konstanta). *Lee and Seung (2001)* telah membuktikan bahwa penggunaan algoritma multiplikatif tersebut tidak mengakibatkan fungsi menjadi naik (*non-increasing*), selain itu *konvergensi* juga dijamin yaitu kondisi dimana matriks W dan H pada iterasi ke- k akan memiliki perbedaan yang sangat kecil sehingga dapat dianggap sama dengan matriks W dan H pada iterasi ke $(k-1)$ sehingga iterasi dapat dihentikan (*konvergen*). Untuk memperkuat klaim dari *Lee and Seung* tersebut penulis melakukan simulasi dengan menggunakan data pelatihan (*training*) yang terdapat dalam *database rating*

MovieLens yang digunakan dalam penelitian ini. Adapun hasil yang diperoleh dari simulasi dapat di lihat pada gambar (3.4) berikut :



Gambar 3.4 Simulasi uji konvergensi algoritma multiplikatif

Berdasarkan gambar 3.4 terlihat bahwa jarak $\|A - WH\|_F$ hasil uji algoritma multiplikatif dengan menggunakan data pelatihan tidak pernah naik (*nonincreasing*), dan setelah iterasi 70-an selisih antara jarak $\|A - WH\|_F$ relatif sangat kecil sekali, dengan demikian dapat dikatakan bahwa hasil uji ini memperkuat pendapat dari *Lee and Seung (2001)*.

BAB IV

SIMULASI

Bab ini menjelaskan hasil simulasi penggunaan NMF pada data riil, di mana pembahasan di bagi menjadi empat subbab. Subbab pertama berisi tentang deskripsi data. Subbab kedua menguraikan tentang spesifikasi komputer yang digunakan dalam penelitian, sedangkan pada subbab ketiga dijelaskan mengenai desain eksperimen, dan subbab keempat berisikan hasil penelitian.

4.1 Data

Data yang digunakan dalam penelitian ini adalah basis data yang bersumber dari *MovieLens* yang merupakan basis data *rating* tentang peminjaman film yang diambil dari (<http://www.grouplens.org/node/73>). Data ini merupakan data yang terbuka untuk umum (*open source*) yang diorganisasi oleh *University of Minneosta* melalui *GroupLens Research Project*. Secara umum data terdiri atas 100.000 *rating* dengan skala 1 – 5 yang merupakan hasil *rating* yang diberikan oleh 943 pengguna terhadap 1682 judul film. Adapun indikator dari data *MovieLens* adalah sebagai berikut :

Tabel 4.1. Indikator basis data *MovieLens*

No	Indikator	Nilai
1	User (N)	943
2	Item (M)	1682
3	Data Rating	100.000
4	Data Training	80.000
5	Data Testing	20.000
6	Ukuran Matrik	943 x 1682
7	Jumlah entri matrik	1.586.126
8	Prosentasi Data Rating	$100.000/1.586.126 = 6.3 \%$
9	Prosentasi Data Training	$80.000/1.586.126 = 0.0503 \%$
10	Prosentasi Data testing	$20.000/1.586.126 = 0,0126 \%$
11	Sparsity Data Rating	93.7 %
12	Sparsity Data Training	99,9597 %
13	Sparsity Data Testing	99,9074 %

Data dengan indikator sebagaimana yang terdapat pada tabel 4.1 terbagi menjadi dua kelompok yang masing-masing merupakan data pelatihan dan data pengujian dimana kedua kelompok data tersebut saling bebas yaitu seorang pengguna tidak merating produk yang sama pada kedua data.

1	1	5	1	6	5
1	2	3	1	10	3
1	3	4	1	12	5
1	4	3	1	14	5
1	5	3	1	17	3
1	7	4	1	20	4
1	8	1	1	23	4
1	9	5	1	24	3
1	11	2	1	27	2
1	13	5	1	31	3
1	15	5	1	33	4
1	16	5	1	36	2
1	18	4	1	39	4
1	19	5	1	44	5
1	21	1	1	47	4
1	22	4	1	49	3
1	25	4	1	51	4
1	26	3	1	53	3
1	28	4	1	54	3
1	29	1	1	56	4
1	30	3	1	60	5

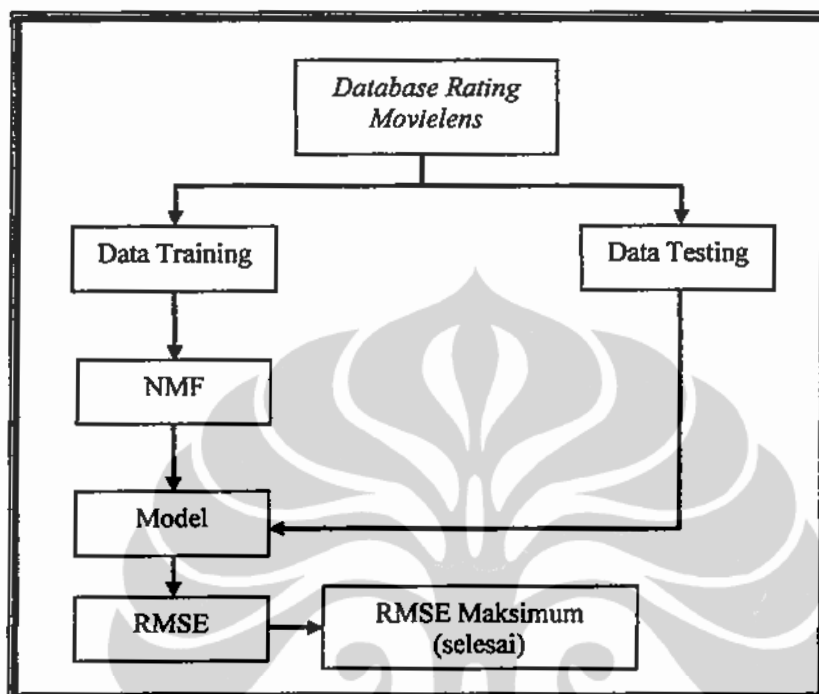
Gambar 4.1 Contoh data *training* dan data *testing* yang saling bebas

4.2 Spesifikasi Komputer

Proses simulasi uji kinerja dari faktorisasi matriks non negatif (NMF) dilakukan dengan menggunakan program *Matlab*. Adapun spesifikasi komputer yang digunakan dalam percobaan ini adalah *Acer Nplify TM802.11b/g/n (ASPIRE 4740)* dengan *Processor 330M Intel CoreTMi3 (2.13 GHz 1066 MHz FSB)*, 320 GB HDD, memory 1 GB dan *OS Windows 7*.

4.3 Skema Eksperimen

Untuk menentukan tahapan-tahapan dalam penelitian ini, berikut digambarkan skema eksperimen yang dilakukan :



Gambar 4.2 Skema percobaan model NMF

Berdasarkan gambar 4.2 terlihat bahwa proses simulasi ini dilakukan dalam beberapa tahapan, yaitu

- 1) Langkah awal dari penelitian ini adalah memilih data pelatihan dan data pengujian yang terdapat dalam basis data *rating* di mana setiap data direpresentasikan ke dalam bentuk vektor
- 2) Dengan menggunakan metode NMF terhadap data pelatihan diperoleh bentuk *latent variable model*. Sebagai ilustrasi, misalkan terdapat matriks non negatif $V_{m \times n}$ yang merupakan representatif dari data pelatihan. Kemudian dilakukan penentuan inisialisasi matriks pengguna-*fitur* $W \in R_{m \times k}$ dan matriks *fitur*-*produk* $H \in R_{k \times n}$ dimana m adalah banyaknya pengguna dan n adalah banyaknya produk (judul film), dengan W dan H matriks non negatif. Matriks $V_{m \times n}$ ini akan diaproksimasi oleh hasil kali dari matriks $W_{m \times k}$ dan matriks $H_{k \times n}$ dengan k adalah bilangan bulat positif.

- 3) Untuk setiap parameter nilai k yang diperoleh dari hasil NMF pada langkah kedua, dilakukan pengujian dengan menggunakan data pelatihan hingga diketahui nilai formula dari masing-masing model.

Dalam penelitian ini penulis melakukan simulasi dengan menggunakan algoritma *multiplikatif* terhadap berbagai nilai k ($2 \leq k \leq 20$), dimana untuk tiap-tiap nilai k dilakukan 100 kali iterasi. Hal ini berdasarkan hasil uji konvergensi algoritma multiplikatif (Gambar 3.4), dimana diketahui bahwa setelah iterasi ke 100 selisih antara jarak $\|A - WH\|_F$ relatif sangat kecil sekali.

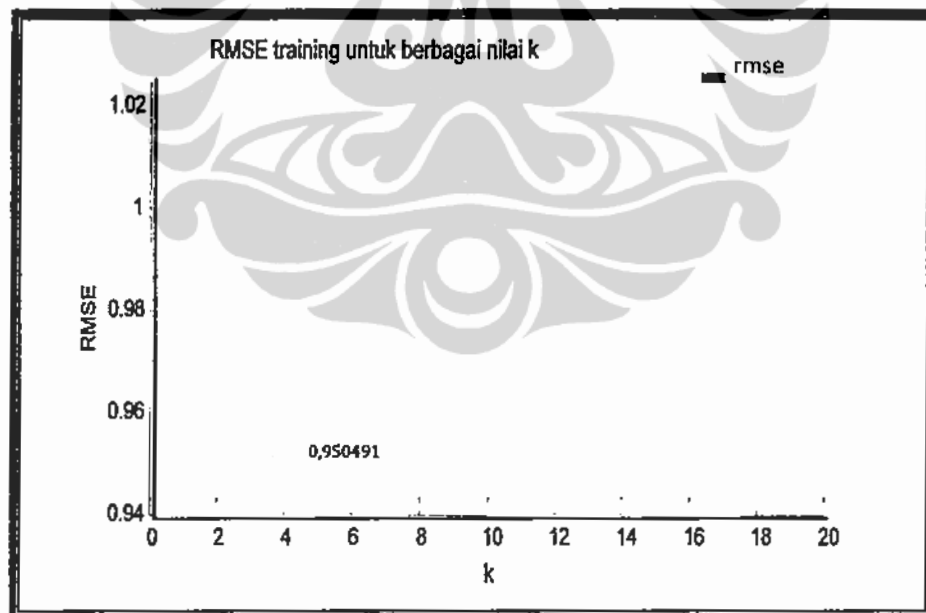
Hasil simulasi yang dilakukan dapat dilihat pada tabel 4.2 berikut.

Tabel 4.2 Perbandingan nilai RMSE hasil simulasi

Perbandingan RMSE Berbagai nilai k dengan 100 Iterasi									
k	2	3	4	5	6	7	8	9	10
Iterasi									
1	1.153807	1.153778	1.153914	1.153832	1.154009	1.153574	1.153717	1.153921	1.15402
2	1.153816	1.153772	1.153912	1.153832	1.153994	1.153575	1.153696	1.15392	1.154005
3	1.153829	1.153761	1.15391	1.153834	1.153975	1.153575	1.153666	1.153919	1.153986
4	1.153845	1.153743	1.153907	1.153838	1.153959	1.153572	1.153621	1.153914	1.153962
5	1.153863	1.153712	1.153902	1.153844	1.153927	1.153563	1.153549	1.153898	1.153928
6	1.153883	1.153661	1.153894	1.153846	1.153892	1.153542	1.153426	1.153861	1.153876
7	1.153903	1.153577	1.153879	1.15384	1.153843	1.153499	1.153218	1.153785	1.153789
8	1.153925	1.153441	1.15385	1.153815	1.15377	1.153421	1.152875	1.153645	1.153646
9	1.153946	1.153225	1.153801	1.153759	1.153658	1.15329	1.152323	1.153406	1.153411
10	1.153966	1.152883	1.153717	1.153651	1.153487	1.15308	1.151456	1.153016	1.153035
11	1.153985	1.15235	1.153579	1.153463	1.153226	1.15275	1.150124	1.152399	1.152443
12	1.153999	1.151527	1.153358	1.153149	1.152834	1.152245	1.148117	1.151442	1.151525
13	1.154007	1.150272	1.153009	1.152644	1.15225	1.151483	1.145154	1.149989	1.150125
14	1.154004	1.148389	1.152465	1.151849	1.151385	1.150352	1.140889	1.147819	1.148024
15	1.153984	1.14562	1.15163	1.150624	1.150118	1.1487	1.134949	1.144648	1.144931
f	f	f	f	f	f	f	f	f	f
100	1.008855	0.979258	0.950491	0.963952	0.980601	0.992729	1.001642	1.058963	1.051358
min	1.008855	0.979258	0.950491	0.963952	0.980601	0.992729	1.001642	1.058963	1.051358

Perbandingan RMSE Berbagai nilai k dengan 100 iterasi (lanjutan)									
11	12	13	14	15	16	17	18	19	20
1.154	1.15429	1.1541	1.15396	1.15455	1.15448	1.15472	1.1546	1.1548	1.15436
1.154	1.15429	1.1541	1.15395	1.15454	1.15446	1.15469	1.15458	1.15475	1.15434
1.154	1.15428	1.15409	1.15393	1.15452	1.15443	1.15466	1.15455	1.15469	1.15432
1.154	1.15427	1.15407	1.15391	1.1545	1.15439	1.15462	1.15453	1.15463	1.15429
1.154	1.15425	1.15405	1.15387	1.15446	1.15434	1.15458	1.1545	1.15456	1.15423
1.154	1.15421	1.15399	1.1538	1.15438	1.15424	1.15451	1.15444	1.15446	1.15413
1.154	1.15413	1.15387	1.15368	1.15426	1.15407	1.15439	1.15432	1.15431	1.15396
1.154	1.15398	1.15367	1.15347	1.15405	1.15378	1.15419	1.1541	1.15406	1.15366
1.153	1.15371	1.15332	1.15311	1.15371	1.15329	1.15384	1.1537	1.15365	1.15316
1.153	1.15327	1.15276	1.15253	1.15315	1.15248	1.15328	1.15304	1.15299	1.15239
1.152	1.15257	1.15187	1.1516	1.15229	1.15121	1.15238	1.15199	1.15197	1.15119
1.15	1.15148	1.15051	1.15017	1.15098	1.14928	1.15099	1.15035	1.15039	1.14938
1.147	1.14983	1.14845	1.148	1.14901	1.14638	1.14889	1.14787	1.14802	1.14672
1.144	1.14737	1.14543	1.1448	1.1461	1.14219	1.14579	1.14424	1.14455	1.14288
1.138	1.14383	1.1411	1.14026	1.14192	1.13637	1.14137	1.13909	1.13962	1.13751
	i	i	i	i	i	i	i	i	i
1.028	1.03994	1.04101	1.04933	1.04461	1.06297	1.0659	1.0774	1.08057	1.08293
1.028	1.03994	1.04101	1.04933	1.04461	1.06297	1.0659	1.0774	1.08057	1.08293

- 4) Dengan mengambil nilai RMSE yang minimum untuk tiap-tiap simulasi pada Tabel 4.2 diperoleh grafik hasil sebagaimana yang terlihat pada Gambar 4.3 berikut.



Gambar 4.3 Grafik perbandingan nilai *RMSE* untuk berbagai nilai *k*

4.4 Hasil Penelitian

Berdasarkan eksperimen yang telah dilakukan pada subbab 4.3 sebagaimana yang terdapat pada Tabel 4.2 dan Gambar 4.3 diketahui bahwa nilai RMSE yang paling minimum yaitu sebesar 0,950491 diperoleh pada saat $k = 4$. Hasil ini didapat setelah dilakukan simulasi terhadap berbagai nilai k dengan menggunakan algoritma multiplikatif *update* sebagaimana yang terdapat dalam pembahasan subbab 3.5.



BAB V

KESIMPULAN DAN SARAN

Bagian ini berisikan kesimpulan dari proses penelitian yang telah penulis lakukan. Selanjutnya terdapat saran untuk perbaikan-perbaikan pada penelitian lanjutan. Proses penelitian dalam tesis ini dilakukan dalam beberapa tahapan, yang terdiri dari tinjauan pustaka terhadap teori-teori yang digunakan dalam penelitian, diantaranya adalah sistem rekomendasi, *collaborative filtering*, *machine learning*, faktorisasi matriks, *non-negative matrix factorization* (NMF) dan *multiplicative update*. Kemudian memilih data yang akan digunakan, dalam penelitian ini data yang digunakan adalah basis data *rating* yang bersumber dari *MovieLens* (<http://www.grouplens.org/node/73>) yang merupakan data peminjaman film. Selanjutnya dengan menggunakan algoritma multiplikatif *update* dilakukan eksperimen untuk mengetahui kinerja dari NMF.

5.1 Kesimpulan

Berdasarkan hasil uji eksperimen terhadap sistem rekomendasi dengan metode NMF pada data *MovieLens*, diperoleh hasil sebagai berikut :

1. Metode NMF dapat digunakan dalam sistem rekomendasi
2. Metode ini optimal dengan nilai akurasi RMSE sebesar 0,950491 di dapat pada saat nilai parameter $k = 4$, $\text{eps} = 10^{-9}$ dan iterasi sebanyak 100 kali
3. Eksperimen juga semakin memperkuat klaim bahwa algoritma multiplikatif *update rule* yang digunakan dalam penelitian ini tidak pernah naik (*non increasing*)

5.2 Saran

Setelah melakukan penelitian ini, penulis merasa terdapat beberapa hal yang perlu untuk dilakukan perbaikan pada penelitian lanjutan, di antaranya adalah

1. Eksperimen dengan NMF dilakukan terhadap semua kumpulan data yang terdapat dalam basis data *rating MovieLens*
2. Eksperimen dengan NMF dalam sistem rekomendasi dilakukan dengan menggunakan basis data *rating* lain
3. Melakukan analisis lanjutan terhadap metode NMF dengan memperhatikan parameter-parameter lain
4. Penggunaan metode faktorisasi matriks yang berbeda dalam sistem rekomendasi
5. Penggunaan algoritma lain dalam NMF



DAFTAR PUSTAKA

- Anton, H. (1988). *Aljabar Linier Elementer Edisi Kelima*. Erlangga. Jakarta
- Anton, H. (2000). *Elementary Linear Algebra : Eight Edition*, John Willey and Sons, Inc., New York.
- Asela, G. dan Guy, S. (2009). *A Survey of Accuracy Evaluation Metrics of Recommendation Tasks*, *Journal of Machine Learning Research*. 10 2935-2962.
- Berry, M. W., Browne, M., Langville, A.N., and Plemmons, R. J. (2007). *Algorithm and applications for approximate non negative matrix factorization*. Computational Statistics and Data Analysis.
- Bishop, C.H. (2006). *Pattern Recognition and Machine Learning*, Springer.
- Deshpande, M. dan Karypis, G. (2004). *Produk-Based Top-N Recommendation Algorithms*. ACM Transactions on Information Systems, ACM New York USA.
- Weisstein, E. W. (2004) "Frobenius Norm", *Mathword, A Wolfram Web Resource, Wolfram Research Inc.*, <http://www.mathword.wolfram.com/Frobenius/Norm.html>.
- Goldberg, K., Roeder, T., Gupta, D., dan Perkins, C. (2001), *Eigentaste : A Constant Time Collaborative Filtering Algorithms*, Information Retrieval Journal
- http://en.wikipedia.org/wiki/Information_retrieval

- Lee, D.D. dan Seung, H.S. (2001). *Algorithms for Non-Negative Matrix factorization*. Advances in Neural Information Processing Systems
- Lee, J. H. Park, S, Ahn, C. M and Kim, D. (2009). *Automatic generic document summarization based on for Non-Negative Matrix factorization*. *Information Processing and Management*.
- Istvan, P. (2009). *Factorization-Based large Scale Recommendation Algorithms*, Ph.D. Thesis, Budapest University of Technology and Economics Hungary.
- Rennie, J. D. M dan Srebro, N. (2005). *Fast Maximum Margin Matrix Factorization for Collaborative Prediction*. In Luc De Raedt and Stefan Wrobel, editors, *Machine Learning, Proceedings of the Twenty Second International Conference (ICML 2005)*, Bonn, Germany.
- Karypis, G. (2000). *Evaluation of Item-Based Top-N Recommendation Algorithms* <http://glaros.dtc.umn.edu/gkhome/node/124>.
- Krzysztof, J. C. Etc. (2007) *Data Mining A Knowledge Discovery Approach*, Springer.
- Hao, M. A. (2009). *Learning to Recommend*, Ph.D. Thesis, The Chinese University of Hongkong.
- Graus, M. (2011). *Understanding the latent Features of Matrix factorization Algorithms in Movie Recommender Systems*. TUE: School of Innovation Sciences. Series master Theses Human Technology Interaction (HTI).
- McGinty, L. dan Smyth, B. (2006) *Adaptive Selection : analysis of critiquing and preference based feed back in conversation on recommender*

systems", International Journal Electronic Commerce.

Salakhutdinov, R and Mnih, A. (2009). *Probabilistic matrix factorization*. In J.C Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*. MIT Press, Cambridge, Massachusett, USA.

Roman, S. (2008). *Advanced Linear Algebra, Graduate Texts in Mathematics (3rd ed)*, Springer.

Sarwar, B.M. Karypis, G. Konstan, J.A. dan Riedl, T.J. (2000), *Application of Dimensionality Reduction in Recommender System: A Case Studi*, Proceeding KDD Workshop on web mining for e-commerce, ACM Press:285-295.

Sarwar, B.M. Karypis, G. Konstan, J.A. dan Riedl, T.J. (2001). *Item-Based Collaborative Filtering recommendation algorithms*, Proceeding of 10th International World Wide Web Conference.

Schafer, J.B., Frankowski, D., Herlocker, J. Dan Sen, S. (2007), *Collaborative Filtering Redommender System*. Springer-Verlag, Berlin, Heidelberg, <http://www.inf.unibbz.it/~ricci/ATIS/index.html>

Gong, S. (2009). *Combining Memory-Based and Model-Based Collaborative Filtering in Recommender System*. Pacific-Asia Conference on Circuit, Communication and System.

Khoshgoftaar. (2009). Review Article *A survey of Collaborative Filtering Techniques*. *Advanced in Artificial Intelligence*, Hindawi Publishing Corporation.