

Collaborative Filtering dan Aplikasinya

Eka Angga Laksana

Teknik Informatika, Universitas Widyatama

Bandung, Indonesia

Eka.angga@widyatama.ac.id

Abstract—Collaborative filtering merupakan salah satu dari teknik di dalam Recommender System yang paling sering digunakan saat ini karena kehandalannya. Recommender system banyak dipakai di dalam dunia e-Commerce untuk membuat personalisasi di dalam sebuah website. Customer website e-Commerce datang untuk mencari produk tertentu, di sinilah peran recommender system dalam menampilkan produk yang berkaitan yang disukai oleh customer. Biasanya customer memberikan rating kepada produk yang mereka sukai dalam skala tertentu. Ada berbagai macam algoritma yang bisa dipakai dalam mengitung similarity antar item maupun user. Algoritma ini sangat berhubungan dengan teknik yang dipakai di dalam machine learning dan information retrieval. Begitu banyak riset di bidang recommender ini membuat aplikasi rekomen berkembang menjadi semakin luas demi tujuan riset maupun bisnis. Penelitian ini memberikan pemahaman singkat mengenai Collaborative filtering dan aplikasinya yang bersifat open source sehingga memberikan kesempatan untuk para peneliti agar mengembangkan hasil penelitiannya di atas salah satu platform aplikasi ini. Salah satu komponen penting di dalam aplikasi adalah validasi, dan inilah yang membuat aplikasi-aplikasi ini menjadi begitu diandalkan dalam Recommender System.

Keywords—Collaborative filtering; Recommender System; e-Commerce; Machine Learning; Informatin Retrieval; similarity

I. PENDAHULUAN

Recommendation system banyak digunakan untuk membuat prediksi seperti buku, music dalam film. Sasarannya adalah pengunjung toko online, di mana system ini memberikan prediksi dan penawaran terbaik produk apa yang disukai. Data referensi produk dari pengunjung disimpan untuk kemudian diolah menggunakan teknik statistik sehingga menghasilkan rekommendasi barang terkait[1]. Teknologi ini banyak diadopsi oleh perusahaan besar di dunia e-commerce dalam seperti Amazon, Netflix dan Pandora. Sedemikian pentingnya recomender system ini bagi ecommerce karena kemampuannya untuk memberikan referensi yang disukai oleh customer, membuat strategi marketing, hingga meningkatkan loyalitas pelanggan. Bahkan kompetisi pun telah diadakan oleh Netflix untuk menguji hasil rekomendasi dengan nilai yang cukup besar. Riset di recommender system menjadi begitu penting karena perannya di bidang e-commerce telah terbukti banyak memberikan terobosan baru di dunia bisnis dan ilmu pengetahuan.

Recommendation system dibagi menjadi dua yaitu Content Based Approach dan Collaborative Filtering[2]. Penelitian ini bertujuan untuk memberikan gambaran mengenai recommender system beserta aplikasinya yang bersifat opensource yang masih dikembangkan hingga saat ini sehingga memudahkan para peneliti untuk melakukan penelitian di bidang ini. Hal tersebut dapat dicapai dengan menggali aspek dan tujuan dari tiap aplikasi termasuk algoritma yang digunakan. Sebagian besar aplikasi berupa kumpulan library yang dibuat berdasarkan algoritma recommender tertentu. Aplikasi ini menggunakan bahasa pemrograman yang beragam meliputi java, phyton dan php yang tentunya telah familiar di mata para peneliti sehingga membuat library ini sangat flexible untuk tujuan riset maupun bisnis.

II. COLLABORATIVE FILTERING

Collaborative filtering merupakan salah satu algoritma yang digunakan untuk menyusun recommender system dan telah terbukti memberikan hasil yang sangat baik. Rating produk merupakan elemen terpenting dari algoritma ini, rating diperoleh dari sebagian besar customer di mana customer secara explicit memberikan penilaiannya terhadap produk. Kesimpulannya ialah system memberikan imbal balik kepada customer dengan mengolah data-data tersebut, sebagai gambaran dari skala nol sampai 5 yang mengindikasikan penilaian yang paling tidak disukai hingga paling disukai menurut sudut pandang customer, data ini memungkinkan untuk dilakukannya perhitungan statistik yang hasilnya menunjukkan produk mana yang diberikan rating tinggi oleh customer.

Collaborative filtering menggunakan database yang diperoleh dari user. Ada dua komponen utama dalam data ini agar dapat membuat prediksi bagi recommender system yaitu user dan item. Keduanya membentuk rating matrix berupa m user $\{u_1, u_2, u_3, \dots, u_m\}$ dan daftar n item $\{i_1, i_2, i_3, \dots, i_n\}$. Di mana setiap user memberikan penilaiannya pada item berupa rating dalam skala 1 sampai 5. Rating ini dilambangkan dengan I_{u_i} . Tidak semua user memberikan rating ke setiap produk karena berbagai macam faktor, hal ini menyebabkan banyaknya missing value yang mengakibatkan sparsity pada data. User item rating matrix dapat digambarkan dengan table di bawah.

Tabel 1 Rating Matrix

| | i_1 | i_2 | i_3 | i_4 | i_m |
|-------|-------|-------|-------|-------|-------|
| u_1 | 1 | ... | 3 | ... | |
| u_2 | 5 | 4 | ... | ... | |
| u_3 | ... | 5 | 3 | ... | |
| u_4 | ... | 4 | ... | ... | |
| u_m | | | | | |

Collaborative filtering dibagi menjadi dua kategori[3] yaitu memory based, model based dan gabungan keduanya menjadi hybrid recommendation system[4] bertujuan untuk mengatasi kelemahan yang muncul pada kedua kategori sebelumnya. Memory based recommendation menggunakan user rating sebagai bahan untuk menemukan similarity atau derajat kesamaan antar user. Di domain bisnis algoritma ini telah diterapkan pada situs Amazon, keunggulannya adalah kemudahan dalam implementasi dan sangat efektif. Sedangkan untuk model based recommendation tidak jauh berbeda dengan algoritma sebelumnya, yaitu menggunakan rating sebagai sumber data. Namun algoritma ini menggunakan teknik-teknik di data mining atau machine learning seperti Bayesian dan clustering. Gabungan dari model dan memory based membentuk hybrid recommender system. Algoritma ini di ciptakan untuk mengatasi kelemahan yang terdapat pada dua algoritma sebelumnya seperti sparsity. Berikut adalah perbandingan antara memory based dan model based recommendation system[5]:

A. Memory based

Teknik yang dipakai adalah: Neighbor-based CF (item based/user-based CF. Algorithms with Pearson/vector cosine correlation) Item-based/user-based top-N Recommendations. Keunggulannya berupa implementasi mudah, mudah menambahkan data-data baru tidak perlu mempertimbangkan content item yang direkomendasikan, skala yang baik dengan co-rated item. Sedangkan kekurangan dari algoritma ini adalah bergantung pada rating dari user, menurunnya performa jika data jarang, skalabilitas yang terbatas pada dataset yang besar

B. Model Based

Teknik yang dipakai adalah: Bayesian belief nets CF, clustering CF, MDP-based CF, latent semantic CF, sparse factor analysis, CF using dimensionality, reduction techniques seperti *SVD* dan *PC*. Kelebihan: dapat mengatasi masalah data yang jarang, scalability dan masalah lainnya, akurasi meningkat, memberikan intuitive rational untuk rekomendasi. Kekurangannya adalah diperlukannya sumber daya yang besar untuk proses komputasi.

C. Hybrid Recommendation System

Teknik yang digunakan berupa content based CF (fab), content boosted CF, hybrid CF kombinasi memory based dan model based (personality diagnosis). Teknik ini sebagai solusi atas kelemahan yang terdapat pada memory dan model based CF seperti sparsity dan grayship, sehingga meningkatkan akurasi prediksi. Beberapa kelemahan juga muncul seperti meningkatnya kompleksitas program hingga perlunya data external yang terkadang tidak tersedia.

III. EVALUASI RECOMMENDER SYSTEM

Efektifitas dan performa suatu algoritma rekomendasi dapat dilihat dari hasil evaluasi. Tipe evaluasi yang dipakai sangat dipengaruhi oleh jenis recommender system yang dipakai. Evaluasi untuk recommender system tidak jauh berbeda dengan teknik yang digunakan dalam machine learning dan information retrieval. Pengujian ini dapat diklasifikasikan menjadi tiga jenis[6] yaitu predictive accuracy metric seperti Mean Absolute Error(MAE) dan variasinya, classification accuracy metrics seperti precision, recall, F1-measure dan ROC sensitivity; dan tipe klasifikasi yang ketiga adalah rank accuracy metric seperti Pearson's product-moment correlation, Kendall's Tau, Mean Average Precision(MAP), half-life utility dan normalized distance performance Metric(NDPM). Tidak semua jenis evaluasi ini diterapkan oleh aplikasi karena karakteristik dan tujuan tiap aplikasi berbeda. Untuk jenis evaluasi apa saja yang diuji cobakan oleh tiap aplikasi akan dibahas di bab berikutnya.

Evaluasi dilakukan dengan cara memotong dataset menjadi dua: train dan evaluasi. Skema pengujian ini umum dilakukan dalam pengujian machine learning. Pada umumnya peneliti menggunakan real dataset seperti movielens, jester dan Netflix. Aplikasi open source recommender system selain melakukan testing juga telah disertakan mekanisme evaluasinya. Hal ini memudahkan pengembangan algoritma baru di atas platform aplikasi tersebut untuk diuji coba seberapa besar performanya.

IV. KENDALA DALAM RECOMMENDER SYSTEM

A. Sparsity

Sparsity diidentikan dengan data yang jarang atau belum terisi penuh. Hal ini diakibatkan oleh adanya item baru yang muncul, Item baru yang muncul bisa jadi kurang sesuai dengan keinginan user karena kurangnya referensi data. User rating merupakan bahan mentah dalam menghasilkan rekomendasi. Item yang tidak mendapatkan rating akan sulit untuk ditemukan kaitannya dengan item yang lain. Penyebab lain adalah user baru biasanya tidak memberikan ratingnya kepada suatu produk. Bisa jadi karena user tidak mengetahui barang yang dia suka atau membiarkan begitu saja produk yang ditampilkan tanpa memberikan rating. Masih banyak lagi factor yang menyebabkan data sparsity ini sehingga menjadi tantangan terbesar dalam collaborative filtering dan paling sulit untuk diatasi[7]. Beberapa riset

telah dilakukan untuk mengatasi masalah ini. Berikut adalah tabel riset sebelumnya untuk mengatasi data sparsity.

Tabel 2 Riset pada data Sparsity

| No. | Penelitian | Teknik |
|-----|------------|---|
| 1 | [8] | collaborative filtering using cluster-based |
| 2 | [9] | Item Classification |
| 3 | [10] | Case-based Reasoning and Item-based Collaborative Filtering |
| 4 | [11] | BP Neural network |

B. Scalability

Collaborative filtering memerlukan training data yang semakin besar agar produk yang direkomendasikan akurat. Amazon, sebagai salah satu situs e-commerce terbesar, semakin hari data yang tersimpan akan semakin besar seiring pertumbuhan jumlah pengunjung. Hal ini tentu saja memakan banyak sumber daya dan komputasi. Komputasi untuk data dalam skala besar menjadi syarat penting untuk rekomen system dalam dunia bisnis dan riset. Oleh karena itu selain spesifikasi hardware, algoritma yang dipakai juga turut berperan performa rekomen system. Tabel berikut adalah riset dalam collaborative filtering untuk mengatasi masalah di skalabilitas ini

Tabel 3 Riset pada Scalability

| No. | Penelitian | Teknik |
|-----|------------|---------|
| 1 | [12] | Cluster |
| 2 | [13] | Hybrid |

V. OPEN SOURCE RECOMMENDER SYSTEM PROJECT

Penelitian yang di bidang recommender system telah berkembang cukup pesat di berbagai belahan dunia. Perkembangan aplikasi open source pun telah membantu para peneliti untuk memberikan kontribusinya. Berikut pembahasan sekilas mengenai aplikasi open source pada recommender system.

A. APACHE MAHOUT

Apache mahout[14] menerapkan algoritma-algoritma di dalam machine learning di dalam environment yang memerlukan skalabilitas besar. Aplikasi ini dapat dijalankan dalam baik system terdistribusi maupun system tunggal. System terdistribusi tepat digunakan dalam skala besar misalkan untuk tujuan bisnis. Sebaliknya bagi peneliti yang mempunyai sumber daya terbatas, apache mahout pun dapat digunakan dalam system tunggal. Ada tiga algoritma yang menjadi fokus utama dalam pengembangan aplikasi ini yaitu recommender mining, Clustering, dan classification.

Aplikasi ini ditulis dengan menggunakan bahasa pemrograman java. Apache sedang mengembangkan riset di bidang big data, dengan implementasinya yang terkenal yaitu apache Maven dan Hadoop. Apache mahout kerja di atas platform ini. Maka tak heran jika penerapan riset recommender system dalam skala yang besar menjadi keunggulannya. Algoritma Collaborative Filtering telah disiapkan sekaligus dengan evaluasinya. User similarity, item similarity, user neighbourhood dan recommender merupakan algoritma yang menarik untuk dicoba. Keunggulan lainnya adalah dukungan komunitas telah tersedia di situs resminya, memungkinkan peneliti untuk berinteraksi dengan peneliti lain di berbagai belahan dunia.

Walaupun aplikasi ini sudah begitu lengkap namun celah untuk pengembangan tetap ada. Ada tiga hal[15] yang masih perlu untuk dikembangkan yaitu adalah ketika proses evaluasi masih kurang untuk dapat menentukan jumlah item terdekat, regulator constant, dan jumlah feature untuk latent factor model. Kedua adalah kemampuan untuk integrasi dengan variable external seperti content attribute, temporal data dan social link. Terakhir bisa dilihat dari sisi teknikal yaitu support dalam mengolah data yang sudah kehabisan memory namun masih dapat dioperasikan di single system dan perlunya support external hingga user dan item baru.

B. Crab

Collaborative filtering menjadi fokus utama dalam pengembangan aplikasi ini. Dengan sifatnya yang fleksibel, akan mudah bagi para peneliti untuk menerapkan algoritma hasil riset mereka. Project ini diawal pada tahun 2010 oleh Marcel Caraciolo setelah itu semakin banyak peneliti yang berkontribusi. Belum ada situs resminya menjadi satu kekurangan tersendiri. Namun bagi para peneliti bisa mengunduhnya di Github. Sedikit sekali informasi yang diperoleh dari situs aplikasi ini, terakhir kali diupdate oleh penulis pada february 2012.

C. Easy Rec

Easy Rec[16] memudahkan recommender system untuk dipadukan dengan website. Website menggunakan beragam bahasa pemrograman seperti php, jsp dan asp namun Easy Rec telah menyediakan fasilitas web service untuk diintegrasikan dengan web. Dukungan API pun membuat proses integrasi menjadi lebih mudah. Algoritma diimplementasi ke dalam plugin, sehingga untuk melakukan testing algoritma hanya memodifikasi plugin tanpa perlu mengubah core program.

Aplikasi ini dibuat dengan bahasa pemrograman java. Sesuai dengan namanya, yaitu bertujuan untuk memudahkan dalam membuat recommender system. Aplikasi dapat dikembangkan sesuai kebutuhan user dan tepat untuk dikembangkan ke dalam halaman web. Algoritma yang dipakai memang khusus untuk recommender system seperti Collaborative filtering dan similarities between item. Untuk aplikasi bisnis yang memerlukan aplikasi instant yang tidak memerlukan custom code, easy rec lebih cocok digunakan.

Namun tidak menutup kemungkinan juga jika digunakan untuk tujuan riset. Dokumentasi dan tutorial cukup lengkap, sehingga tidak sulit untuk diuji cobakan.

D. LENSKIT

Lenskit[17] menjadi salah satu aplikasi terkemuka yang khusus diciptakan untuk penelitian pada recommender system. Dokumentasi penggunaan aplikasi ini begitu lengkap, ditambah lagi dengan adanya beberapa algoritma yang siap untuk digunakan seperti item based CF, user based CF, dan matrix factorization. Aplikasi ini melakukan pengolahan data secara offline dan evaluasi pun telah tersedia untuk menguji hasil rekomendasi ini.

Cukup banyak publikasi penelitian yang berbasis Lenskit. Begitu banyak riset dan aplikasi dengan berbagai metodologi memunculkan adanya perbedaan yang signifikan di bidang recommender system. Ditambah lagi dengan metode evaluasi yang berbeda dan terkadang peneliti kurang memperhatikan hal-hal penting dalam prosesnya. Karena alasan inilah dibuatnya Lenskit, sebagai aplikasi yang menciptakan standar bagi recommender system beserta aplikasinya sehingga peneliti dapat fokus untuk peningkatan algoritma recommender yang sudah ada[18]. Algoritma yang dipakai adalah Item-based CF, User-based CF, Matrix factorization dan slope-one.

E. MyMediaLite

MyMediaLite[19] dibuat dengan tujuan untuk menghasilkan recommendation system yang ringan. Beberapa algoritma juga telah tersedia untuk diuji. Pengujian dapat dilakukan setelah algoritma diterapkan. MAE, RMSE dan AUC merupakan teknik pengujian yang telah tersedia. Tak heran rasanya bila aplikasi ini patut dicoba untuk tujuan riset, ditambah lagi untuk peneliti yang memiliki sumber daya terbatas.

Aplikasi ini dibuat dengan bahasa pemrograman C#, namun untuk platform Linux telah tersedia versi mono. Tujuannya adalah sebagai riset framework untuk dipakai oleh para peneliti. Semenjak dirilis sebagai project open source pada tahun 2010 aplikasi ini telah didownload lebih dari 2.300 kali dan telah digunakan oleh berbagai penelitian[20].

F. SVD feature

Aplikasi berupa toolkit yang menerapkan machine learning untuk feature based Collaborative Filtering. Keandalannya tidak diragukan lagi karena telah teruji dengan menjuarai KDD CUP selama dua tahun[21]. Feature based matrix factorization menjadi masalah utama yang diangkat di dalam aplikasi ini[22]. Kemampuannya dalam mengolah data yang sangat membuat SVD feature menjadi sangat efisien.

Fitur yang dimiliki berupa dapat menghandle big data dalam single machine, membangun platform state of the art di dalam Collaborative Filtering, dan performa yang tinggi. Aplikasi ini dibuat dengan bahasa pemrograman C++.

Dokumentasi cukup lengkap, dapat dilihat dari situs resminya dan juga ada user baru telah disediakan demo aplikasi.

G. Duine Framework

Duine framework merupakan kumpulan library yang dapat digunakan oleh peneliti untuk membuat algoritma recommender hasil karya mereka sendiri. Aplikasi ini membuat prediksi atas informasi yang diperlukan oleh user, baik yang mereka sukai maupun tidak. Jika digabungkan ke web maka informasi yang disajikan dapat dipersonalisasi sesuai karakter user. Kelebihan yang dimiliki adalah diantaranya kemampuan untuk memberikan prediksi yang akurat dan dapat mengatasi kendala cold-start. Cold-start adalah kondisi dimana informasi yang tersimpan sangat sedikit, akibatnya adalah rekomendasi tidak dapat menurunnya prediksi yang dihasilkan oleh recommender.

Aplikasi ini dibuat dengan bahasa pemrograman java. Dikutip dari situs resminya[23] ada tiga hal yang ditawarkan untuk menambah kontribusi di pengembangan recommender system. Pertama kemampuannya untuk menerapkan algoritma baru ada collaborative filtering. Selain itu telah dilengkapi dengan kumpulan evaluation tools untuk menguji akurasi prediksi. Dan terakhir adalah Explanation API yang itu membuat hasil rekomendasi lebih mudah dipahami dengan tampilan grafik yang menarik.

H. Recommenderlab

Recommenderlab[24] merupakan extension dari R statistical tool. Aplikasi ini diciptakan untuk membuat dan melakukan testing pada algoritma collaborative filtering. Aplikasi ini menyediakan riset infrastruktur tidak hanya untuk rating matrix namun juga untuk binary 0-1 data. Fokus aplikasi ini adalah untuk data handling yang efisien, konsistensi dan uji coba algoritma terbaru. Berbagai teknik pengujian juga telah tersedia di Recommenderlab.

Tujuan dibuat aplikasi ini adalah sebagai general research infrastruktur di dalam collaborative filtering. Algoritma yang telah tersedia diantaranya Top N, IBCF, UBCF lengkap dengan mekanisme evaluasinya. Hasil pengujian dapat diubah menjadi tampilan grafik. Recommenderlab menjadi sangat handal karena dibentuk di atas platform R, sehingga kemampuannya untuk teknik-teknik statistik. Walau demikian masih mungkin untuk dikembangkan algoritma baru di Recommenderlab, sesuai dengan tujuannya yaitu untuk menguji rekomendasi yang dihasilkan oleh collaborative filtering.

VI. KESIMPULAN

Collaborative filtering menjadi topik riset yang populer saat ini. Hal ini dibuktikan dengan banyaknya aplikasi open source yang khusus diciptakan untuk penelitian ini. Framework yang dibuat tak hanya untuk tujuan riset, namun juga dalam hal bisnis. Selain itu, tiap tahun juga diadakan kompetisi yang bernama KDD CUP yang diselenggarakan oleh ACM Special Interest Group on

Knowledge Discovery and Data Mining dan ada aplikasi recommender yang telah dibahas sebelumnya, telah menjuarai kompetisi tersebut. Bagi penelitian, hal ini merupakan sebuah tonggak penting untuk pengembangan algoritma yang lebih efisien. Kontribusi peneliti akan lebih banyak lagi jika dilakukan di atas frame work ini. Waktu penelitian juga dapat lebih singkat karena peneliti dapat fokus untuk peningkatan algoritma, ditambah lagi evaluasi atau pengujian juga telah tersedia di dalam tiap aplikasi. Namun review tiap aplikasi di paper ini masih sangat sedikit sehingga untuk penelitian berikutnya perlu digali lebih dalam lagi pembahasan untuk tiap aplikasi. Uji coba algoritma juga dapat dilakukan di bidang yang lain tidak hanya e-commerce namun juga di bidang yang lain.

VII. DAFTAR PUSTAKA

- [1] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Analysis of Recommendation Algorithms for E-Commerce," *Discovery*.
- [2] A. Ansari, S. Essegai, and R. Kohli, "Internet Recommendation System.pdf." Proceedings of the Fourteenth Conference, pp. 363–375, 2000.
- [3] "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," pp. 43–52.
- [4] P. Melville, R. J. Mooney, and R. Nagarajan, "Content-Boosted Collaborative Filtering for Improved Recommendations," pp. 187–192.
- [5] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," *Adv. Artif. Intell.*, vol. 2009, no. Section 3, pp. 1–19, 2009.
- [6] J. L. Herlocker, J. a. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, Jan. 2004.
- [7] L. Li, R. Hsu, and F. Lee, "Review of Recommender Systems and Their Applications."
- [8] H. Mase and H. Ohwada, "A Collaborative Filtering Incorporating Hybrid-Clustering Technology i I , I," no. Icsai, pp. 2342–2346, 2012.
- [9] H. Tan, "A Collaborative Filtering Recommendation Algorithm Based On Item Classification," 2009.
- [10] R. Systems and S. Gong, "Joining Case-based Reasoning and Item-based Collaborative Filtering in," pp. 40–42, 2009.
- [11] D. Chen, "Networks," vol. 1, pp. 2–4, 2009.
- [12] X. Li, "Collaborative Filtering Recommendation Algorithm Based on Cluster," pp. 2682–2685, 2011.
- [13] S. Gong and H. Tan, "Combining Memory-Based and Model-Based Collaborative Filtering in Recommender System," pp. 14–17, 2009.
- [14] "Apache Mahout: Scalable machine learning and data mining." [Online]. Available: <http://mahout.apache.org/>. [Accessed: 08-Dec-2014].
- [15] S. Schelter and S. Owen, "Collaborative Filtering with Apache Mahout Categories and Subject Descriptors," vol. i, 2012.
- [16] "easyrec :: open source recommendation engine." [Online]. Available: <http://easyrec.org/>. [Accessed: 08-Dec-2014].
- [17] "LensKit Recommender Toolkit." [Online]. Available: <http://lenskit.org/>. [Accessed: 08-Dec-2014].
- [18] M. D. Ekstrand, M. Ludwig, J. A. Konstan, and J. T. Riedl, "Rethinking the Recommender Research Ecosystem : Categories and Subject Descriptors."
- [19] "mloss | Project details:MyMediaLite." [Online]. Available: <http://mloss.org/software/view/282/>. [Accessed: 08-Dec-2014].
- [20] L. Schmidt-thieme and S. Rendle, "MyMediaLite : A Free Recommender System Library," pp. 305–308, 2011.
- [21] "SVDFeature - SVDFeature." [Online]. Available: http://svdfeature.apexlab.org/wiki/Main_Page. [Accessed: 08-Dec-2014].
- [22] T. Chen and Q. Lu, "SVDFeature : A Toolkit for Feature-based Collaborative Filtering," vol. 13, pp. 3619–3622, 2012.
- [23] "Duine Framework - Recommender Software Toolkit." [Online]. Available: <http://www.duineframework.org/>. [Accessed: 08-Dec-2014].
- [24] M. Hahsler, "recommenderlab : A Framework for Developing and Testing Recommendation Algorithms."