

Sistem Rekomendasi Jual Beli Barang Dengan Memanfaatkan Metode *Collaborative Filtering* dan Basis Data Graf. Studi Kasus: Bukalapak.com

David Eko Wibowo

School of Electrical Engineering and Informatics, Institute
Technology of Bandung,
Bandung, Indonesia.
13510006@std.stei.itb.ac.id

Rinaldi Munir

School of Electrical Engineering and Informatics, Institute
Technology of Bandung,
Bandung, Indonesia.
rinaldi@informatika.org

Abstrak- Sistem rekomendasi merupakan hal yang penting dalam sebuah situs jual beli barang, contoh situs jual beli barang saat ini yaitu bukalapak.com. Pada situs tersebut rekomendasi diberikan berdasarkan barang yang paling banyak dilihat dan kemiripan sebuah nama barang saja. Rekomendasi yang digunakan pada bukalapak.com kurang optimal karena rekomendasi yang diberikan tidak personal untuk setiap pengguna. Oleh karena itu diperlukan sebuah sistem rekomendasi yang memanfaatkan perilaku pengguna (*collaborative filtering*) agar rekomendasi yang diberikan bisa lebih optimal. Model yang cocok digunakan untuk merepresentasikan perilaku pengguna dalam metode *collaborative filtering* adalah graf, akan tetapi muncul masalah ketika harus memodelkan graf dalam sebuah basis data relasional terkait waktu pemrosesan, maka dari itu digunakan basis data graf untuk mengatasi masalah tersebut.

Model data dibangun menggunakan data interaksi yang dilakukan oleh pengguna dan untuk membangun data tersebut dilakukan implementasi berdasarkan analisis yang telah dilakukan. Implementasi dibagi ke dalam dua bagian yakni UDP Server untuk pengumpulan data, dan API rekomendasi untuk menguji hasil rekomendasi. Dari hasil pengujian yang didapat dengan menggunakan data yang telah dikumpulkan dari situs bukalapak.com diketahui bahwa nilai presisi dan recall untuk algoritma Grafil lebih besar dibanding dengan algoritma KNN, maka dari itu dapat disimpulkan bahwa algoritma Grafil lebih cocok untuk diimplementasikan di bukalapak.com. Sedangkan pemodelan graf berdasarkan interaksi yang dilakukan pengguna berpengaruh kepada tipe algoritma yang digunakan. Pada hasil yang didapat terlihat bahwa algoritma KNN belum menunjukkan hasil yang optimal karena penilaian implisit yang diberikan belum bisa menggantikan penilaian eksplisit yang biasa diberikan oleh pengguna langsung. Sedangkan penggunaan basis data graf menunjukkan kecocokan karena waktu pemrosesan yang terhitung singkat serta kueri yang lebih sederhana dalam menerapkan metode *collaborative filtering* dan penggunaan setiap algoritma.

Kata kunci : sistem, rekomendasi, bukalapak, *collaborative filtering*, knn, grafil, graf, basis data

I. LATAR BELAKANG

Rekomendasi adalah saran yang bersifat menganjurkan menurut Kamus Besar Bahasa Indonesia (KBBI). Penggunaan rekomendasi saat ini dapat ditemui pada sebuah situs jual beli barang. Rekomendasi pada situs ini digunakan untuk membantu pengguna dalam menentukan pilihan barang yang akan dibelinya. Salah satu situs jual beli barang adalah bukalapak.com. Situs ini merupakan salah satu situs jual beli terbesar di Indonesia. Pada sistem bukalapak.com, rekomendasi yang diberikan ke pengguna ada dua jenis, yang pertama rekomendasi berdasarkan jumlah *view* pada barang tersebut, semakin banyak *view* yang didapat maka barang ini akan semakin direkomendasikan, yang kedua rekomendasi berdasarkan kemiripan nama dan deskripsi sebuah barang, ketika pengguna melihat sebuah barang, maka akan direkomendasikan barang sejenis dengan kemiripan nama dan deskripsi.

Rekomendasi yang terdapat pada bukalapak.com kurang optimal, karena rekomendasi yang diberikan tidak personal. Maka dari itu perlu dibangun sebuah sistem rekomendasi yang bisa menghasilkan rekomendasi secara personal sehingga rekomendasi menjadi lebih optimal. Terdapat tiga jenis sistem rekomendasi berdasarkan metode yang digunakannya yakni, *collaborative filtering*, *content-based filtering*, dan *hybrid*. Metode *collaborative filtering* merupakan metode yang memanfaatkan perilaku pengguna untuk menghasilkan rekomendasi, metode *content-based filtering* merupakan metode yang menggunakan *data-mining* ataupun pencarian berdasarkan nama atau deskripsi sebuah barang, dan metode *hybrid* yang merupakan kombinasi *collaborative filtering* dan *content-based filtering*.

Dalam studi kasus bukalapak.com, rekomendasi personal bisa dihasilkan dengan menggunakan metode *collaborative filtering*. Rekomendasi ini akan dihasilkan berdasarkan perilaku pengguna yang dimodelkan dengan relasi pengguna dengan barang. Relasi pengguna dengan barang ini merupakan bentuk dari model yang memanfaatkan interaksi yang dilakukan pengguna terhadap sebuah barang. Representasi

yang paling sesuai untuk memodelkan relasi ini adalah sebuah graf. Akan tetapi jika implementasi model graf dilakukan pada basis data relasional maka terdapat masalah terkait waktu pemrosesan yang besar, ditambah lagi dengan jumlah data yang banyak pada sebuah situs jual beli barang [1]. Saat ini telah dikembangkan basis data graf untuk mengatasi masalah yang muncul pada basis data relasional terkait model graf. Basis data graf adalah sebuah basis data yang telah dioptimasi sedemikian rupa sehingga memiliki performansi yang lebih tinggi [2].

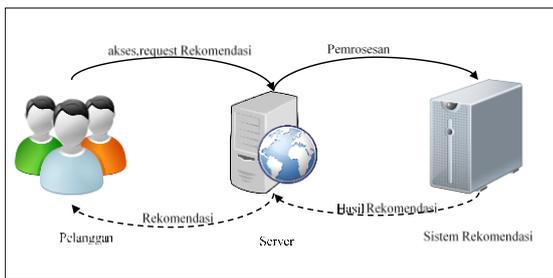
Dalam sistem rekomendasi, hasil rekomendasi bukan hanya terpengaruh oleh model saja, melainkan juga algoritma yang digunakan. Pada penelitian yang telah dilakukan, algoritma *K-Nearest Neighbors*(KNN) menunjukkan waktu komputasi yang rendah serta kecocokan penggunaan *cosine similarity* untuk barang yang memiliki penilaian yang jarang sehingga performansinya memuaskan [3]. Algoritma yang akan dibandingkan dengan algoritma KNN adalah algoritma *Graph Similarity Filtering* (Grafil) yang telah diteliti dan menghasilkan performansi yang baik pada studi kasusnya yakni topologi ikatan kimia yang modelnya adalah sebuah Graf [4].

II. DASAR TEORI

Dalam dasar teori akan dibahas beberapa teori yang dipakai dalam makalah ini, yaitu tentang sistem rekomendasi, teori graf, algoritma KNN, algoritma Grafil, basis data graf, decision support metrics, dan penelitian terkait.

A. Sistem Rekomendasi

Sistem rekomendasi merupakan sistem yang mengaplikasikan berbagai macam teknik pencarian pengetahuan untuk mengatasi masalah pembuatan rekomendasi secara personal [5]. Rekomendasi umumnya diberikan berdasarkan preferensi pemberi rekomendasi, pencari rekomendasi (pelanggan sistem rekomendasi), atau kelompok yang dianggap dapat merepresentasikan selera pasar.



Gambar 1 Sistem Rekomendasi [6]

Sistem rekomendasi pada Gambar 1 bekerja ketika seorang pelanggan mengakses situs. *Server* akan meminta rekomendasi untuk menghasilkan saran (rekomendasi) untuk pelanggan. Sistem rekomendasi memroses permintaan *server* dengan melakukan komputasi berdasarkan metode yang dimilikinya. Hasil dari sistem rekomendasi dapat berupa saran mengenai produk yang paling sesuai untuk pelanggan, informasi yang disajikan sesuai dengan keinginan pelanggan,

atau produk yang paling populer berdasarkan pendapat komunitas pelanggan [6]. Terdapat tiga jenis sistem rekomendasi berdasarkan metodenya yaitu *collaborative filtering*, *content-based filtering*, dan *hybrid*.

Collaborative Filtering

Merupakan sebuah proses *filtering* atau evaluasi sebuah produk menggunakan penilaian pengguna maupun kebiasaan pengguna. Jika dibandingkan dalam kehidupan sebenarnya bisa diibaratkan seperti ketika seseorang memiliki kumpulan rekomendasi film untuk temannya, dan temannya juga memiliki kumpulan rekomendasi tersendiri untuk dibagikan kepada temannya. Setiap orang yang memiliki rekomendasi pastilah memiliki penilaian terhadap apa yang mereka ingin rekomendasikan, dan pada teknik *collaborative filtering* hal tersebut dimodelkan [7].

Bentuk penilaian di dalam metode *collaborative filtering* dibedakan menjadi tiga yakni *scalar ratings*, *binary ratings*, dan *unary ratings*. *Scalar ratings* adalah penilaian yang diberikan dengan nilai eksak, misalnya satu sampai lima ataupun nilai eksak dengan rentang tertentu. *Binary ratings* adalah penilaian yang dilakukan dengan memberikan pertanyaan setuju atau tidak setuju dan baik atau buruk. *Unary ratings* adalah penilaian yang diindikasikan dari kegiatan pengguna seperti melihat ataupun membeli sebuah barang.

Dalam menghasilkan rekomendasi, metode *collaborative filtering* menggunakan perhitungan kemiripan dengan beberapa cara seperti *cosine-based similarity*, *correlation-based similarity*, dan *adjusted cosine similarity*. *Cosine-based similarity* mengibaratkan dua barang sebagai dua buah vektor i dan j pada suatu n -ruang pengguna. Kemiripan lalu dihitung dengan kosinus sudut $sim(i,j)$ antara dua vektor tersebut. Persamaan yang digunakan dalam *cosine-based similarity* dapat dilihat pada Persamaan 2.1.

$$sim(i,j) = \cos(i,j) = \frac{i \cdot j}{||i|| ||j||} \tag{2.1}$$

Perhitungan kemiripan dengan menggunakan *correlation-based similarity* dilakukan dengan mengisolasi dua buah barang yang diberikan penilaian oleh pengguna yang sama. Jika kumpulan pengguna yang memberikan penilaian kepada i dan j dinotasikan dengan U maka korelasi kemiripannya dapat dinyatakan seperti pada Persamaan 2.2.

$$sim(i,j) = \frac{\sum_{u \in U} (R_{u,i} - R_i)(R_{u,j} - R_j)}{\sqrt{\sum_{u \in U} (R_{u,i} - R_i)^2} \sqrt{\sum_{u \in U} (R_{u,j} - R_j)^2}} \tag{2.2}$$

Pada persamaan 2.2, $R_{u,i}$ menotasikan penilaian pengguna u terhadap barang i , $R_{u,j}$ menotasikan penilaian pengguna u terhadap barang j , R_i menotasikan rata-rata penilaian barang ke- i . Perhitungan dengan menggunakan *adjusted cosine similarity* mengimbangi kekurangan perbedaan rentang dalam penilaian pengguna yang tidak sesuai dengan cara melakukan substraksi rata-rata pengguna dari setiap nilai pasangan yang

telah diberi nilai. Secara formal persamaannya dapat dilihat pada Persamaan 2.3.

$$sim(i, j) = \frac{\sum_{u \in U} (R_{u,i} - R_u)(R_{u,j} - R_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - R_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - R_u)^2}} \quad (2.3)$$

Pada persamaan 2.3, R_u adalah rata-rata penilaian pengguna ke- u , $R_{u,i}$ menotasikan penilaian pengguna u terhadap barang i , $R_{u,j}$ menotasikan penilaian pengguna u terhadap barang j .

Content-based Filtering

Sistem yang menggunakan metode *content-based filtering* menganalisis deskripsi ataupun karakteristik dari suatu barang agar bisa dicocokkan dengan keinginan dari pengguna. Sistem rekomendasi dengan metode ini akan berhadapan dengan kumpulan besar *data training* dan melakukan klasifikasi untuk barang yang disukai pengguna ataupun berguna untuk pengguna [8].

Ada tiga tahap yang dilakukan di dalam sistem rekomendasi yang menggunakan teknik *content-based filtering* yang dikerjakan dalam komponen yang berbeda [9]. Komponen pertama yaitu *content analyzer*, ketika informasi yang didapat belum terstruktur, sebuah *pre-processing* haruslah dilakukan terlebih dahulu di dalam komponen ini agar terekstraksi menjadi informasi yang terstruktur, tanggung jawab utama dari komponen ini adalah mengubah bentuk informasi yang datang menjadi bentuk informasi yang bisa diterima oleh proses selanjutnya yaitu *profile learner* dan *filtering component*. Komponen yang kedua adalah *profile learner*, komponen ini mengumpulkan data representatif dari output komponen sebelumnya dan melakukan generalisasi data untuk membangun profil pengguna. Biasanya strategi di dalam melakukan generalisasi data menggunakan bantuan algoritma *machine learning* yang akan menjadikan model barang yang pengguna sukai dan tidak sukai. Komponen yang ketiga yaitu *filtering component*, bertugas untuk melakukan penggunaan profil pengguna yang telah dihasilkan untuk mendapatkan rekomendasi barang yang relevan dan sesuai dengan representasi profil yang didapat.

Hybrid System Recommendation

Sistem rekomendasi yang menggunakan teknik ini merupakan gabungan dari sistem rekomendasi dengan teknik *collaborative filtering* dan *content-based filtering*. Dalam meningkatkan efisiensi dari pemberian rekomendasi pada suatu pengguna, *content-based filtering* digunakan untuk mendukung *collaborative filtering*. Knowledge-based machine yang digunakan dalam proses *filtering* sangatlah efektif untuk membantu *collaborative filtering* [10].

Metode hybrid ini sebenarnya dilakukan guna membantu dalam mengatasi setiap batasan pada metode *collaborative filtering* dan *content-based filtering*. Pendekatan untuk membuat sistem rekomendasi dengan metode hybrid bisa dilakukan dengan berbagai cara yang berbeda dan bisa diklasifikasikan seperti melakukan implementasi *collaborative filtering* dan *content-based filtering* dan menggabungkan hasil

prediksinya, menggabungkan beberapa karakteristik dari *content-based filtering* ke dalam *collaborative filtering*, menggabungkan beberapa karakteristik dari *collaborative filtering* ke dalam *content-based filtering*, dan membangun model secara general yang saling menggabungkan karakteristik dari *collaborative filtering* dan *content-based filtering*.

B. Teori Graf

Graf digunakan untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Dalam teorinya graf $G = (V, E)$ adalah

V = himpunan tidak kosong dari simpul simpul (*vertices*) = $\{v_1, v_2, \dots, v_n\}$

E = himpunan sisi (*edges*) yang menghubungkan semacam simpul = $\{e_1, e_2, \dots, e_n\}$

Jenis-jenis graf berdasarkan ada tidaknya gelang ataupun sisi ganda pada sebuah graf dibedakan menjadi dua yaitu graf sederhana dan graf tidak sederhana. Sedangkan jika dilihat jenisnya berdasarkan orientasi arah pada dua sisi, maka dapat dibedakan menjadi dua pula yaitu graf tak berarah dan graf berarah [11].

C. Algoritma K-Nearest Neighbor (KNN)

Algoritma k-nearest neighbour (KNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Tujuan dari algoritma ini adalah mengklasifikasikan obyek baru berdasarkan atribut dan *training sample*. *Classifier* tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan memori. Diberikan titik *query* akan ditemukan sejumlah objek atau (titik *training*) yang paling dekat dengan titik *query*. Secara umum algoritma KNN terdapat pada algoritma 1.

```

Input :  $S$  : Simpul set,  $k$  : nilai penentu simpul yang diambil
        $maxS$  : Ukuran maksimum simpul set
        $c$  : Simpul yang ingin diklasifikasikan
Output : Item set  $I$ 
-----
For setiap simpul  $S_i, i < maxS$  do
   $H \leftarrow$  hitung_jarak_euclidean( $c, S_i$ )
  Set atribut_relasi ( $c, S_i$ ) to  $H$ 
EndFor
 $I \leftarrow$  Get  $k$  simpul dengan jarak terdekat
Kembalikan  $I$ 

```

Algoritma 1 Algoritma KNN

Algoritma ini bekerja berdasarkan jarak terpendek dari *query* ke *training sample* untuk menentukan KNN-nya. *Training sample* diproyeksikan ke ruang berdimensi banyak, dimana masing-masing dimensi merepresentasikan fitur dari data. Ruang ini dibagi menjadi bagian-bagian berdasarkan klasifikasi *training sample*. Sebuah titik pada ruang ini ditandai kelas c jika kelas c merupakan klasifikasi yang paling banyak ditemui pada k buah tetangga terdekat dari titik tersebut. Dekat atau jauhnya tetangga biasanya dihitung

berdasarkan *Euclidean distance* atau *cosine-based similarity*. Ketepatan algoritma KNN dipengaruhi oleh ada atau tidaknya fitur-fitur yang relevan atau jika bobot fitur tersebut tidak setara dengan relevansinya terhadap klasifikasi.

D. Graph Algorithm Filtering (Grafil)

Grafil merupakan algoritma untuk memberikan rekomendasi dengan pencocokan sub-graf. Grafil terdiri dari dua komponen yakni komponen dasar dan komponen *clustering*. Masing-masing dari komponen tersebut memanfaatkan strategi komposisi *multifilter*. Komponen dasar menghasilkan fitur dengan mengelompokkan dengan ukuran yang sama dan menggunakannya untuk melakukan penyaringan graf berdasarkan batas atas yang diizinkan. Untuk komponen *clustering* menggabungkan fitur dari setiap ukuran yang berbeda dan dikelompokkan sesuai nilai yang diinginkan [12]. Secara umum algoritma Grafil dapat dilihat pada Algoritma 2.

```

Input :  $D$  : Basis data graf,  $F$  : Fitur set
       $maxL$  : Ukuran maksimum fitur
       $Q$  : Query
Output : Hasil kandidat  $C_Q$ 

-----
 $C_Q = D$ 
For setiap fitur set  $F_i, i < maxL$  do
     $d_{max} \leftarrow$  Hitung_maximum_feature_misses( $F_i$ )
     $C_Q = \{ G \mid d(G, Q) \leq d_{max}, G \in C_Q \}$ 
EndFor
For setiap fitur set  $F_i \cup F_{i+1}, i < maxL$  do
    Hitung selektivitas berdasarkan  $C_Q$ 
    Lakukan clustering secara hirarkis kepada fitur  $F_i \cup F_{i+1}$ 
    Kelompokkan fitur kepada tiga kelompok,  $X_1, X_2, X_3$ 
    For setiap cluster  $X$  do
        Hitung maximum feature misses  $d_{max}$ 
         $C_Q = \{ G \mid d(G, Q) \leq d_{max}, G \in C_Q \}$ 
    EndFor
EndFor
Kembalikan  $C_Q$ 

```

Algoritma 2 Algoritma Grafil [12]

Pada algoritma tersebut variabel D merupakan kumpulan simpul pengguna di dalam basis data graf, F adalah kumpulan fitur dari *query* graf dimana fitur ini didapat dari *index* basis data graf. C_Q adalah hasil pemrosesan *query*, $maxL$ merupakan nilai yang didapat dari hasil *indexing* basis data graf. Sedangkan untuk d_{max} adalah variable yang didapatkan dari perhitungan fitur di dalam *query* graf, dan $d(G, Q)$ adalah perhitungan selektivitas yang dilakukan dengan menjumlahkan rata-rata perbedaan frekuensi dari setiap fitur dari graf target dan *query* graf.

E. Basis Data Graf

Basis Data Graf (*Graph database*) merupakan suatu *online database management system* dengan metode CRUD (*Create, Read, Update, Delete*) yang telah ada untuk membentuk model data graf. Secara umum *graph database* dibangun untuk digunakan dengan sistem transaksional (OLTP). Ada dua property dari basis data graf yang perlu dipertimbangkan

dalam mempelajari teknologi basis data graf ini yaitu dasar penyimpanan dan mesin pemroses.

Dasar penyimpanan beberapa basis data graf menggunakan penyimpanan graf secara asli yang dioptimalkan dan didesain untuk menyimpan dan mengatur graf. Tidak semua penyimpanan basis data graf menggunakan penyimpanan asli, beberapa menyambung-nyambungkan melalui basis data relasional, basis data berbasis objek, atau basis data yang lainnya untuk kebutuhan penyimpanan tertentu.

Pada mesin pemroses beberapa definisi menyebutkan bahwa basis data graf menggunakan indeks bebas yang berdekatan, sehingga setiap simpul yang terhubung secara fisik mengarah satu sama lain di dalam basis data.

Hubungan antar entitas adalah sebuah hal yang paling diutamakan dalam model data graf, tidak seperti sistem manajemen basis data lain yang membuat koneksi antara entitasnya dengan properti lain seperti *foreign key*. Dengan menyusun abstraksi sederhana dari setiap *nodes* dan *relationships* menjadi struktur yang saling berhubungan, *graph database* bisa memodelkan suatu bentuk yang memetakan dengan dekat terhadap domain masalah yang ingin diselesaikan [13].

Beberapa varian basis data graf yang ada saat ini yaitu Neo4j, OrientDB, Titan, dan DEX. Setiap basis data graf tersebut memiliki kelebihan dan kekurangan masing-masing yang disesuaikan dengan tujuan penggunaan.

F. Decision Support Metrics

Performansi dari pemberi rekomendasi, secara khusus pada domain *unary* seperti riwayat pembelian bisa di evaluasi dari *precision-recall framework* dari penerimaan informasi yang diturunkan dari teori keputusan statistik. *Framework* ini memeriksa kapasitas dari sistem untuk mengidentifikasi secara akurat sumber yang tersedia untuk *query*, mengukur secara terpisah kapasitas untuk mencari barang yang relevan serta menghindari barang yang tidak relevan.

Tabel 1 Retrieval Confusion Matriks

	<i>Relevant</i>	<i>Irrelevant</i>
<i>Retrieved</i>	TP	FP
<i>Not Retrieved</i>	FN	TN

Tabel 2 Perhitungan Precision dan Recall

	<i>Relevant</i>	<i>Irrelevant</i>
<i>Retrieved</i>	TP	FP
<i>Not Retrieved</i>	FN	TN

TP = *True Positive*, menyatakan secara benar diidentifikasi
 TN = *True Negative*, menyatakan secara tidak benar diidentifikasi

FP = *False Positive*, menyatakan secara benar ditolak

FN = *False Negative*, menyatakan secara tidak benar ditolak

Nilai precision dihitung dengan memanfaatkan persamaan 2.4, sedangkan nilai recall dihitung dengan memanfaatkan persamaan 2.5.

$$Precision = \frac{TP}{TP + FP} \quad (2.4)$$

$$Recall = \frac{TP}{TP + FN} \quad (2.5)$$

Pada persamaan 2.4 dan 2.5 *precision* menyatakan barang yang diberi oleh sistem rekomendasi dan dibeli, sedangkan *recall* adalah barang yang dikembalikan dari sistem rekomendasi.

G. Penelitian Terkait

Terdapat penelitian terkait yang dilakukan oleh Zan Huang (2005). Penelitian ini melakukan investigasi terhadap *graph-based algorithms* untuk mengatasi masalah yang sering ada pada sistem rekomendasi yakni kejarangan data, dan juga mengatasi kekurangan sebuah *unified framework* agar bisa menerima berbagai macam input data. Pada penelitian ini, investigasi lebih difokuskan kepada *consumer-product graphs* yang menggambarkan setiap transaksi antara simpul *consumer* dan simpul *product*.

Untuk mengatasi masalah kejarangan data, maka dilakukan investigasi untuk algoritma *network spreading activation* dan algoritma baru yang diusulkan yaitu algoritma *link analysis* yang memiliki ide dari *Web graph analysis techniques*. Sedangkan untuk *unified framework* yang ingin dibangun didasarkan pada *probabilistic relational models* (PRMs) dan di buat dalam skema relational.

Algoritma *network spreading activation* merupakan pendekatan yang dilakukan untuk mengatasi kejarangan data dalam *collaborative filtering* pada disertasi ini. Algoritma ini telah lama digunakan untuk *associative retrieval* masing-masing untuk manusia dan pemrosesan model informasi, dan sebagai mekanisme komputasional untuk mempercepat proses eksplorasi dari jaringan asosiasi. *Spreading activation* disini akan digunakan untuk metode komputasi untuk menelusuri asosiasi transitif secara efisien antara konsumen dan produk dalam *collaborative filtering*. Di dalam *spreading activation* terdapat beberapa algoritma yang merepresentasikannya yaitu *Leaky Capacitor Model* (LCM), *branch-bound serial, symbolic search algorithm* (BNB), dan *Hopfield net parallel relaxation search algorithm* (Hopfield). Ketika membandingkan dengan algoritma *collaborative filtering* lain, algoritma yang dipilih untuk merepresentasikan *spreading activation* adalah Hopfield karena sifat konsistennya yang telah diuji coba pada aplikasi lain. Hasil yang didapat pada pengujian hipotesis yang ada yakni algoritma *spreading activation collaborative filtering* bisa menghasilkan rekomendasi berkualitas tinggi jika dibandingkan algoritma lain menunjukkan bahwa algoritma ini mendapatkan nilai yang tertinggi dibandingkan dengan algoritma lainnya yaitu 3-hop, *user-based corellation*, *user-based vector similarity*, dan *item based*.

Untuk algoritma *link analysis* yang diusulkan, pengujian dilakukan dengan menggunakan data dari *online bookstore* hasil yang didapat menunjukkan hasil yang lebih baik dalam menentukan *neighborhoods* tanpa mementingkan struktur

globalnya. Sedangkan untuk *unified framework* yang menggunakan *probabilistic relational models* (PRMs) sebagai pendekatan utama *relational learning*. *Framework* ini digunakan pada basis data dengan skema relational dengan mendeskripsikan beberapa kumpulan kelas yang di dalam basis data diwujudkan sebagai tabel. Pada *framework* ini juga digunakan multi-set operator untuk mengeksplorasi pola data yang penting dan relevan untuk sistem rekomendasi. Hasil pengujian *framework* ini menunjukkan bahwa sistem yang dibuat belum bisa diintegrasikan secara sistematis dengan berbagai pendekatan rekomendasi yang ada, untuk menginterasikan serta mendapatkan hasil yang diinginkan maka diperlukan spesialisasi khusus untuk setiap pendekatan rekomendasi.

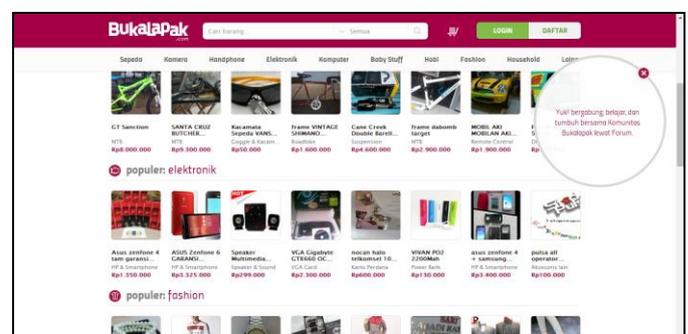
III. ANALISIS

Pada bagian ini akan dibahas mengenai perancangan solusi yang diajukan, mulai dari pembangunan model, dan penerapan algoritma.

A. Analisis Situs Bukalapak.com

Bukalapak.com adalah salah satu sistem jual beli barang secara *online* terbesar di Indonesia. Pada situs ini layaknya sebuah sistem jual beli barang terdapat elemen-elemen penting yang mencirikan sebuah tempat penjualan barang secara *online*. Elemen-elemen tersebut adalah adanya barang-barang yang dijual, informasi penjual barang tersebut, mekanisme pembelian, mekanisme penjualan, akun penggunaannya itu sendiri dan tentunya sebuah rekomendasi yang diberikan kepada setiap pengguna.

Setiap barang yang dijual di dalam bukalapak.com menggunakan mekanisme yang berbeda jika dibandingkan situ jual beli lainnya. Di dalam melakukan transaksi pembelian bukalapak.com menjadi perantara di antara penjual dan pembeli untuk menjamin ketersediaan uang dan juga barang sehingga penjual dan pembeli dapat dipastikan mendapatkan haknya masing-masing. Setiap transaksi yang dilakukan oleh pengguna baik penjual maupun pembeli dicatat sehingga pengguna bisa memastikan setiap transaksi yang dilakukannya.



Gambar 2 Tampilan Situs Bukalapak.com

Berbagai kategori barang terdapat di dalam bukalapak.com, dan di dalam setiap kategori tersebut terdapat penjual-penjual yang sudah memasang barang yang akan

dijualnya sehingga siap ditawarkan kepada pembeli. Setiap penjual di dalam bukalapak.com bisa menerima sebuah pencapaian ataupun penghargaan ketika telah melakukan transaksi yang banyak serta mendapat masukan positif dari pengguna. Penjual dengan *feedback* positif dari pengguna disebut juga sebagai *Top Seller* di dalam bukalapak.com. Dengan adanya predikat ini tingkat kepercayaan pengguna pun meningkat sehingga transaksi penjualan bisa ditingkatkan.

Selain mekanisme penjual dan pembeli yang ada di bukalapak.com terdapat pula sistem rekomendasi yang digunakan untuk membantu pengguna dalam menentukan barang yang akan dibelinya. Pada saat ini rekomendasi yang diberikan oleh bukalapak.com menggunakan sistem *Apache Solr Lucene* yakni mesin untuk melakukan pencarian sebuah kata kunci terhadap dokumen data yang ada. Melalui sistem ini pengguna yang mencari sebuah barang maka akan diberikan rekomendasi barang yang sejenis. Barang sejenis yang direkomendasikan didapatkan dari pencocokan berdasarkan kategori dan juga nama barang. Selain itu untuk mendukung hasil pencarian digunakan juga algoritma "*more like this*" yang sudah terdapat pula dalam *Apache Lucene* untuk menentukan kemiripan hasil pencarian antara satu barang dengan barang lainnya.

Selain berdasarkan kemiripan nama sebuah barang, pada awal membuka situs bukalapak.com maka pengguna juga akan mendapatkan rekomendasi berdasarkan sejarah pengguna di dalam situs tersebut. Jika pengunjung tersebut pertama kali melakukan akses ke bukalapak maka rekomendasi yang diberikan adalah barang-barang yang memiliki tingkat pembelian paling baik tingkat pembelannya (*Top Product*) dari kategori yang populer di dalam situs bukalapak.com.

Sistem rekomendasi yang digunakan dalam bukalapak.com terhubung dengan situs bukalapak.com karena metode pencarian rekomendasi dilakukan secara *online* melalui *wrapper apache Solr lucene* yang ada di dalam situs bukalapak.com.

Interaksi Pengguna

Pengguna yang dalam hal ini seorang pembeli memiliki dua jenis interaksi terhadap sebuah barang. Interaksi yang pertama adalah melihat sebuah barang, pengguna dikatakan melihat sebuah barang ketika pengguna tersebut melihat detail dari barang tersebut, perpindahan dari satu barang ke barang lain pun dipertimbangkan waktunya untuk meminimalisir kesalahan pengguna dalam memilih sebuah barang. Interaksi kedua yang dilakukan pengguna adalah membeli sebuah barang. Ketika seorang pengguna membeli sebuah barang maka dapat dipastikan bahwa ketertarikan pengguna tersebut kepada barang yang dibelinya lebih besar daripada yang dilihatnya. Interaksi melihat dan membeli ini akan menjadi indikator penting dalam memodelkan interaksi pengguna di dalam basis data graf.

Perbedaan minat yang muncul secara implisit di dalam interaksi yang berbeda yakni melihat dan membeli juga dapat menjadi alasan adanya pemberian nilai terhadap masing-masing tipe interaksi. Rentang nilai yang ditentukan juga haruslah sesuai agar hasil rekomendasi yang didapat

maksimal. Selama ini penilaian biasanya berasal dari pengguna ketika mereka telah membeli sebuah barang, dan jika kita lihat di dalam sistem bukalapak.com mekanisme tersebut belum ada. Pembeli hanya akan bisa memberikan respon kepuasan atau ketidakpuasan atas barang yang dibelinya. Maka dari itu penilaian interaksi yang dilakukan sebelum membeli sebuah barang merupakan hal cukup penting untuk diperhitungkan dalam menghasilkan rekomendasi sesuai dengan perilaku atau minat pengguna tersebut.

Rekomendasi Barang Untuk Pengguna

Saat ini jika kita mengunjungi halaman utama dari bukalapak.com maka dapat kita temui bagian rekomendasi. Rekomendasi yang diberikan pada halaman utama ini masih sama untuk setiap pengguna, karena rekomendasi yang diberikan adalah barang-barang yang memiliki jumlah lihat terbanyak atau paling populer.

Rekomendasi berdasarkan algoritma *More Like This (MLT)* baru akan digunakan ketika pengguna melihat sebuah barang. Berdasarkan data barang yang dilihat tersebut akan dicari barang yang memiliki kemiripan berdasarkan nama barang maupun data deskripsi yang ada di dalamnya.

Penerapan Metode Collaborative Filtering

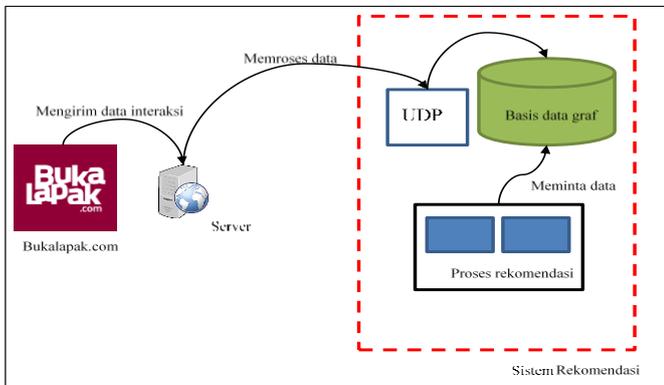
Pemodelan yang ingin dilakukan akan menggunakan perilaku dari pengguna dan juga data dari barang yang dijual pada situs bukalapak.com. Metode ini dipilih karena paling sesuai dengan kebutuhan pemodelan yang menggambarkan perilaku pengguna di dalam memodelkan hubungan antara pengguna dan barang. Selain itu penggunaan metode ini karena kekurangan dari teknik *content-based filtering* pada *Apache Solr Lucene* mengenai referensi yang ditujukan personal untuk masing-masing pengguna. Referensi yang dibutuhkan pada teknik ini dapat diperoleh dari teknik *collaborative filtering*, sehingga rekomendasi yang dihasilkan bisa lebih tepat dan optimal untuk setiap pengguna.

Pengumpulan data yang akan dilakukan untuk pemodelan dilakukan secara *online*, sedangkan pengujian algoritma nantinya di dalam *collaborative filtering* dilakukan secara *offline*. Untuk membantu dalam pengumpulan data, maka rekomendasi yang telah digunakan di dalam situs bukalapak.com akan sangat membantu dalam membangun model hubungan antara seorang pengguna dengan sebuah barang. Teknik *content-based filtering* merupakan metode yang digunakan untuk menganalisis data barang dan data pengguna. Metode yang akan digunakan akan menghasilkan barang-barang berdasarkan kemiripan nama saja dengan menggunakan *framework* yang sudah ada yakni *Apache Solr Lucene*. Melalui *framework* ini kita hanya dapat menghasilkan barang yang serupa berdasarkan kata kunci dalam nama yang digunakan oleh pengguna.

Penggunaan metode *collaborative filtering* diusulkan karena adanya kebutuhan analisis perilaku pengguna dan juga kebutuhan analisis relasi pengguna dengan barang. Ketika memodelkan hubungan antara pengguna nanti, sebagai pembeli dibutuhkan perilakunya dalam berinteraksi dengan setiap barang yang ada, dalam merekomendasikan barang

nanti unsur utama yang digunakan adalah tetap interaksi dari pengguna yang lebih menonjol.

Gambar 3 menunjukkan arsitektur lebih lanjut pengujian prototipe sistem rekomendasi yang akan dibangun, pada sistem rekomendasi yang akan dibangun berada terpisah dari server bukalapak.com. Sedangkan komponen di dalam sistem rekomendasi itu sendiri ada dua yakni basis data graf dan sistem pemroses rekomendasi. Basis data graf akan di akses dan dikonfigurasi agar dapat menerima *input* sehingga pemodelan dapat dilakukan ketika interaksi telah dilakukan. Sedangkan sistem pemroses rekomendasi mengambil data dari basis data graf melakukan komputasi terhadap algoritma yang digunakan yakni KNN dengan *cosine similarity* dan *Grafil*.



Gambar 3 Arsitektur Pengujian Prototipe Sistem Rekomendasi

Di dalam sistem pemroses rekomendasi terdapat dua bagian yang dilibatkan dalam proses penghasilan rekomendasi. Bagian pertama adalah penghitung kemiripan antara satu simpul pengguna dengan pengguna yang lain. Lalu bagian kedua adalah bagian yang mengambil hasil rekomendasi untuk diberikan kepada pengguna. Pada bagian penghitung kemiripan akan dilakukan pemrosesan menggunakan algoritma KNN dan *Grafil*. Hasil dari setiap algoritma akan dibuat relasi baru antar setiap pengguna yang kemiripannya sesuai.

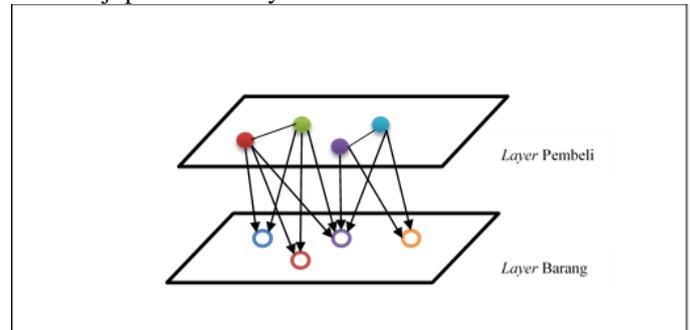
Untuk pengguna yang belum memiliki graf di dalam basis data, maka rekomendasi akan dimunculkan dahulu berdasarkan *content-based filtering*, setelah itu ketika pengguna telah melihat atau membeli sebuah barang maka graf di dalam basis data akan mulai terbentuk lalu rekomendasi untuk pengguna tersebut mulai bisa dihasilkan. Selain itu kata kunci yang diketikkan pengguna di dalam kolom pencarian di dalam basis data juga akan dicatat sebagai tambahan referensi untuk sistem dalam memberikan rekomendasi.

Pembangunan Model Di Dalam Basis Data Graf

Berdasarkan penelitian yang telah dilakukan sebelumnya oleh Zan Huang pada tahun 2005, ketika ia mencoba memodelkan hubungan antara pengguna dan barang dengan metode *hybrid*, model yang digunakan adalah *two-layer graph model*. Hal ini digunakan karena pada setiap *layer* tersebut dimodelkan graf dimana jika yang digunakan oleh Zan Huang, satu *layer* untuk hubungan antar pengguna, dan

satu *layer* untuk hubungan antar barang. Sedangkan untuk hubungan antara pengguna dan barang dilakukan dengan cara menghubungkan kedua *layer* tersebut.

Metode yang digunakan pada tugas akhir ini adalah *two-layer graph model* dengan adanya modifikasi model pada *layer* barang yang kemiripannya tidak akan dihitung terlebih dahulu sehingga fokus kepada kemiripan pengguna saja. Seperti yang telah dijelaskan sebelumnya kemiripan antar pengguna dihasilkan dengan menggunakan algoritma yang akan diuji performansinya.



Gambar 4 Skema *Two-Layer Graph Model* [14]

Pada Gambar 4 digambarkan skema pemodelan yang diusulkan, modelnya digambarkan serupa dengan yang sudah diteliti oleh Zan Huang yakni dengan dua lapisan. Perbedaannya terdapat pada letak *layer* kemiripan barang yang tidak dibuat relasinya satu sama lain. Sehingga dari model ini dimanfaatkan fleksibilitasnya untuk mengatur metode mana yang sesuai untuk setiap tipe pengguna yang berbeda.

B. Penerapan Algoritma

Algoritma yang akan digunakan yakni Algoritma KNN dan Algoritma *Grafil*. Algoritma KNN dengan *cosine similarity* dipilih untuk diterapkan karena melalui teknik ini dapat dihitung kemiripan seorang pengguna dengan pengguna lainnya dengan memberikan penilaian secara implisit pada setiap interaksi. Sedangkan algoritma *Graph Algorithm Filtering* (*Grafil*) dipilih karena melalui metode ini dapat dihasilkan rekomendasi berdasarkan kemiripan bentuk interaksi antara satu pengguna dengan pengguna yang lain.

Penerapan Algoritma KNN dengan Cosine Similarity

Pada sistem yang akan diimplementasikan perhitungan *cosine similarity* dilakukan dengan terlebih dahulu melakukan kueri terhadap dua orang pengguna i dan j untuk mengambil barang yang sama-sama dilihat atau dibeli. Akan tetapi karena sistem jual barang yang dijadikan studi kasus tidak memiliki mekanisme penilaian dari pengguna, maka setiap interaksi yang dilakukan oleh pengguna akan diberi pembobotan.

Untuk setiap interaksi “melihat” yang dilakukan oleh pengguna maka di dalam relasi/sisi yang akan dibuat diberi bobot 1, sedangkan untuk setiap interaksi “beli” maka relasi/sisi akan diberi penilaian 2. Setelah penilaian ini maka prediksi bisa dilakukan dengan menghitung kemiripan antara dua pengguna dengan menggunakan barang yang sama-sama dilihat oleh kedua pengguna tersebut.

Kemiripan yang dihitung menggunakan *cosine similarity* ditambahkan ke dalam relasi yang telah dibuat. Nilai kemiripan yang dihasilkan jika mendekati nilai 1 berarti kemiripannya bagus sedangkan jika mendekati 0 berarti kemiripannya mendekati buruk. Setelah kemiripan ini ditambahkan maka rekomendasi pun bisa dihasilkan. Rekomendasi yang dihasilkan diambil dengan algoritma KNN dari kemiripan yang telah ada pada relasi, berdasarkan kemiripan tersebut dicari tetangga yang paling dekat dari simpul pengguna yang ingin dicari rekomendasinya.

Penerapan Algoritma Grafil

Pada penerapan algoritma grafil, pencocokan antara sub-graf dengan graf tidak menggunakan perhitungan kemiripan berdasarkan penilaian implisit seperti di *similarity matrix*s. Perhitungan kemiripan dilakukan berdasarkan cocok atau tidaknya sebuah fitur terhadap graf yang dibandingkannya. Hal ini dilakukan karena algoritma ini bekerja dengan mencocokkan bentuk, tipe serta atribut busur yang dimiliki pengguna yang ingin dicari rekomendasinya.

Pada penerapan algoritma ini pengguna akan dibandingkan dengan pengguna lain yang sudah mengunjungi barang yang dilihat pengguna yang ingin diberikan rekomendasi, setelah itu berdasarkan penilaian kecocokannya dilakukan pengelompokan sesuai hasil penilaian tersebut dan hasil tersebut kemudian diberikan kepada pengguna. Adanya *pre-processing* yakni perhitungan kemiripan dalam algoritma ini memungkinkan dibutuhkannya waktu yang lama dalam memroses data perbandingan dan menjadikannya sebuah rekomendasi, penggunaan basis data graf dapat membantu proses ini sehingga waktu yang dibutuhkan tidak terlalu lama.

IV. IMPLEMENTASI

Implementasi dibagi ke dalam dua bagian yakni UDPServer dan API. UDPServer digunakan untuk mengumpulkan data, sedangkan API digunakan untuk menghasilkan rekomendasi. Bahasa pemrograman yang digunakan adalah *Ruby*, bahasa pemrograman ini dipilih karena menyesuaikan dengan bahasa pemrograman yang digunakan dalam situs bukalapak.com serta menyesuaikan pula dengan basis data graf yang dipilih yakni Neo4j. Untuk terhubung dengan basis data graf kedua bagian implementasi tersebut menggunakan *wrapper* yakni *Neography*.

A. UDPServer

Untuk pengambilan data digunakan mekanisme *User Datagram Protocol (UDP)* karena menyesuaikan dari interaksi yang dilakukan oleh pengguna dilakukan di situs bukalapak.com yang bersifat terus menerus. Program *server* UDP dibangun menggunakan bahasa pemrograman yang sama yakni *Ruby*, dengan *gem* atau *library* yang digunakan yakni *Eventmachine* dan *Neography*.

Eventmachine adalah *library* dalam program *Ruby* yang diperuntukkan agar sistem menjadi scalable dalam menangani data dengan konkurensi tanpa mengganggu performansi dari sistem. Pola dari *eventmachine* yang seperti reaktor sehingga respon disampaikan bersamaan ketika data yang diterima satu

atau lebih. Penggunaan library *Eventmachine* digunakan karena beban data yang dikirim akan cukup besar sehingga dibutuhkan pola yang digunakan dalam *eventmachine* agar data yang dikirim tidak terganggu sehingga tidak mengganggu situs bukalapak.com dalam pengiriman datanya. Setiap data yang diterima dikirim dalam format : tipe_interaksi<spasi>userid,itemid ; contohnya jika *user* a melihat barang b maka format data yang dikirim adalah LOOK a,b, sedangkan jika user a membeli barang b maka format data yang dikirim adalah PURCHASE a,b.

Setiap data yang diterima akan langsung diproses ke dalam basis data graf dengan bantuan *wrapper Neography* sehingga terhubung ke basis data graf. Library ini akan menghubungkan program dengan basis data graf sehingga setiap data yang diterima langsung dibuat di dalam basis data graf Neo4j.

B. Application Programming Interface (API)

Pada sistem ini terdapat tiga buah kelas *controller* yakni *ClustersController*, *RecommendationsController*, dan *PrecisionsController*. Masing-masing *controller* memiliki tanggung jawabnya masing-masing dalam menangani setiap *request* yang datang dan memberikan respon sesuai dengan tipe *request* yang diterima.

V. PENGUJIAN

Pengujian performansi dilakukan secara *offline*. Lingkungan yang digunakan untuk pengujian adalah sebagai berikut :

Lingkungan perangkat keras (*hardware*) yang digunakan adalah :

1. *Processor* Intel Core i5-2450M CPU @ 2.50 GHz
2. 500 GB HDD
3. *Memory* 4 GB RAM

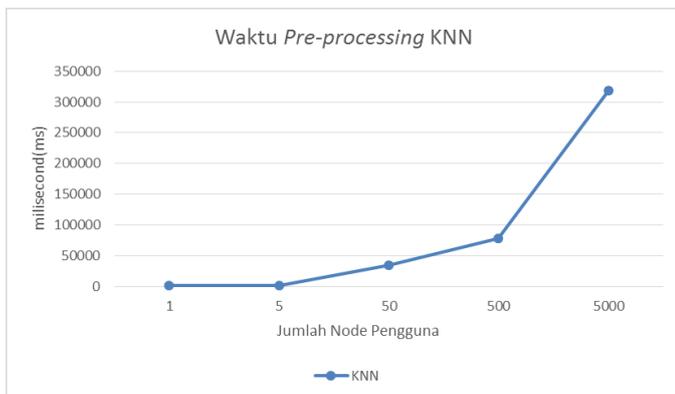
Lingkungan perangkat lunak (*software*) yang digunakan adalah :

1. Sistem operasi Ubuntu 14.04 Trusty Tahr 64-bit
2. Bahasa pemrograman Ruby v1.9.3 dengan Rails v3.2.13
3. Basis data graf Neo4j *Graph Database* v2.1.2
4. Pengelolaan teks Sublime Text 2
5. Alat penguji API POSTMAN
6. *Library* "Neography" dan *library* "Eventmachine"

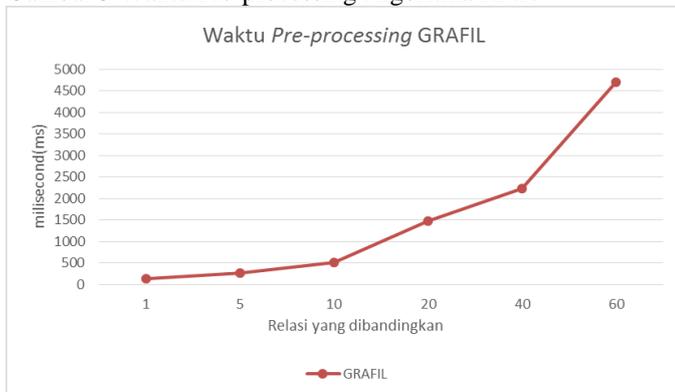
Pengumpulan data menggunakan UDPServer dilakukan selama tujuh hari dengan menanam program di *server* bukalapak.com. Setelah data berhasil dikumpulkan data tersebut dibagi menjadi data training dan data test dengan perbandingan 3:1 berdasarkan penelitian yang dilakukan [15].

Pengujian dibagi menjadi menjadi dua tipe yakni pengujian waktu dalam menghasilkan rekomendasi dan pengujian keefektifan rekomendasi dengan menghitung *precision* dan *recall*.

Hasil pengujian waktu dibagi menjadi *pre-processing* yakni waktu yang digunakan untuk melakukan *clustering* dan waktu dalam menghasilkan rekomendasi. Hasil waktu *pre-processing* Algoritma KNN dapat dilihat pada Gambar 5, sedangkan waktu *pre-processing* Algoritma Grafil dapat dilihat pada Gambar 6.

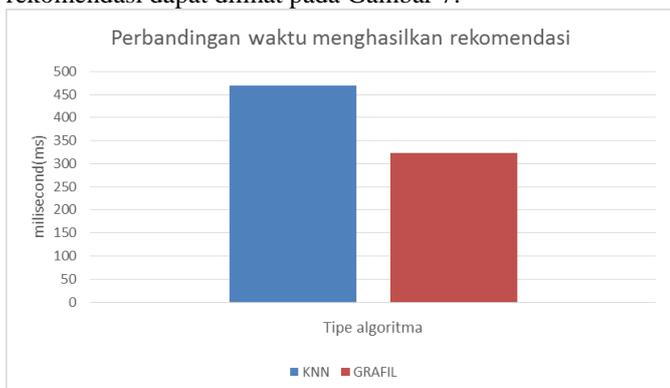


Gambar 5 Waktu Pre-processing Algoritma KNN



Gambar 6 Waktu Pre-processing Algoritma Grafil

Sedangkan perbandingan waktu dalam menghasilkan rekomendasi dapat dilihat pada Gambar 7.



Gambar 7 Perbandingan Waktu dalam menghasilkan rekomendasi.

Untuk hasil perhitungan nilai presisi dan *recall* dapat dilihat pada Tabel 3.

HASIL PERHITUNGAN				
Jumlah Rekomendasi	KNN dengan <i>cosine similarity</i>		Grafil	
	Presisi	<i>Recall</i>	Presisi	<i>Recall</i>
5	0,1784	0,2706	0,3364	0,51031
10	0,1249	0,378944	0,1876	0,51031
20	0,06245	0,378944	0,0841	0,51031

Pada Tabel 3 terlihat jumlah rekomendasi yang dibandingkan, hal ini dilakukan untuk menentukan pada jumlah

rekomendasi berapa masing-masing algoritma mendapat nilai optimal.

VI. EVALUASI

Berdasarkan hasil pengujian yang didapat, diketahui bahwa proses *pre-processing* untuk setiap algoritma cukup memakan waktu yang lama. Pada KNN lamanya waktu ditentukan oleh banyaknya tetangga dari simpul yang ingin dikelompokkan, semakin banyak tetangganya maka akan semakin lama waktunya karena pada saat melakukan pengelompokan dilakukan juga pembuatan relasi baru yakni relasi kemiripan (*similarity*) dengan atribut nilai perhitungan dengan *cosine similarity*. Sedangkan pada algoritma Grafil lamanya proses pengelompokan dipengaruhi oleh banyaknya simpul pengguna lain yang didapat dari hasil traversal simpul yang ingin dikelompokkan dan juga banyaknya relasi yang dimiliki oleh simpul pengguna yang ingin dibandingkan. Berdasarkan hasil yang didapat, algoritma grafil memiliki waktu yang cenderung lebih lama dalam melakukan pengelompokan yang disebabkan banyaknya simpul pengguna dan relasi ke simpul barang yang harus dicocokkan dengan relasi ke simpul barang yang dimiliki oleh simpul pengguna yang ingin dikelompokkan. Pencocokan ini memakan waktu yang lebih lama dibandingkan perhitungan kemiripan yang dilakukan dalam algoritma KNN.

Evaluasi selanjutnya adalah tentang nilai presisi dan *recall* yang didapat, dari nilai presisi kedua algoritma menunjukkan nilai presisi yang cukup rendah yakni 17,84% untuk KNN dan 33,36% untuk Grafil. Berdasarkan perhitungan dan pencocokan data memang ditemukan banyaknya rekomendasi yang tidak berada di dalam *test set*, sedikitnya data interaksi per pengguna juga menjadi salah satu penyebab cukup rendahnya nilai presisi pada kedua algoritma. Akan tetapi jika kita melihat pada nilai *Recall* yang didapat oleh algoritma Grafil, nilai tersebut mengindikasikan nilai yang baik untuk sebuah rekomendasi karena setengah dari *test set* berada di dalam rekomendasi yang dihasilkan.

Jumlah rekomendasi yang dihasilkan juga berpengaruh terhadap hasil perhitungan nilai presisi, dalam tabel tersebut algoritma KNN mendapat penurunan nilai presisi tidak terlalu signifikan karena ditemukan kembali hasil rekomendasi yang ada di dalam *test set*. Jika dibandingkan dengan algoritma Grafil, nilai presisi yang didapat algoritma KNN masih lebih rendah hal ini disebabkan karena algoritma Grafil menemukan lebih banyak rekomendasi yang cocok pada saat menghasilkan jumlah rekomendasi 5, sedangkan algoritma KNN memerlukan jumlah rekomendasi 10 agar kecocokannya bertambah.

Berdasarkan hasil pengujian ini juga, model di dalam basis data graf tentang interaksi pengguna dan barang dalam menghasilkan rekomendasi sangat berpengaruh. Pada tempat studi kasus yakni bukalapak.com, algoritma menunjukkan ketidakcocokan penggunaan dengan didapatnya nilai presisi dan *recall* yang lebih kecil. Penilaian secara implisit yang diberikan kepada setiap interaksi belum bisa dengan optimal menggantikan penilaian pengguna secara eksplisit. Hal ini ditunjukkan dengan hasil penilaian untuk sebuah rekomendasi

yang dihasilkan algoritma KNN yang belum menunjukkan kecocokan.

VII. KESIMPULAN

Berikut merupakan kesimpulan yang didapat setelah pengujian dilakukan :

1. Berdasarkan data *training* dan data *test* yang digunakan dalam pengujian maka hasil yang didapat yakni 17,84% presisi maksimal dan 37,89% *recall* maksimal untuk algoritma KNN dengan *cosine similarity*, lalu presisi maksimal sebesar 33,64% dan *recall* maksimal sebesar 51,03%. Dari data nilai *presisi* dan *recall* tersebut diketahui algoritma *Graph Similarity Filtering* (Grafil) memiliki nilai *presisi* dan *recall* yang lebih tinggi sehingga algoritma ini lebih cocok diimplementasikan di bukalapak.com.
2. Pemodelan graf berdasarkan interaksi pengguna yang diterapkan pada basis data graf berpengaruh kepada tipe algoritma yang digunakan. Pada tempat studi kasus yakni bukalapak.com. Algoritma KNN mendapatkan nilai *presisi* dan *recall* yang cukup rendah dan menunjukkan kurangnya kecocokan terhadap penggunaannya di dalam sebuah penilaian implisit.
3. Waktu yang dibutuhkan algoritma Grafil dalam menghasilkan rekomendasi tanpa memperhitungkan pre-proses adalah 324 ms, dan 469 ms untuk algoritma KNN dengan *cosine similarity*. Perbedaan yang terdapat pada waktu menghasilkan rekomendasi sangat kecil, akan tetapi hal ini bergantung juga kepada simpul pengguna yang dicari rekomendasinya.
4. Penggunaan basis data graf pada sistem rekomendasi yang menggunakan metode *collaborative filtering* cocok digunakan karena waktu yang digunakan untuk pemrosesan algoritma yang cukup kompleks seperti KNN dan Grafil terhitung singkat, serta kueri yang digunakan juga lebih sederhana dibandingkan dengan kueri yang dilakukan pada basis data relasional.

VIII. SARAN

Untuk penelitian dan pengembangan selanjutnya hal-hal yang dapat dipertimbangkan antara lain :

1. Pemodelan graf yang dilakukan dapat dimaksimalkan dengan melihat urutan interaksi.

2. Pada penggunaan sistem rekomendasi ini, graf yang dibangun sebaiknya memiliki batasan sisi yang dimiliki oleh seorang pengguna karena jika dibiarkan terus berkembang maka waktu pemrosesan akan menjadi lebih lama serta hasil rekomendasi mungkin menjadi tidak valid.

REFERENSI

- [1] J. Herlocker, J. Konstan, A. Borchers, J.Riedl. (1999). "An Algorithmic Framework for performing Collaborative Filtering". Proceedings of the 1999 Conference on Research and Development in Information Retrieval, Aug, 1999
- [2] Robinson, Ian & Webber, Jim & Eifrem, Emil. (2013). *Graph Databases*. O'Reilly Media : United States of America
- [3] Burke, Robin. (2005). "Hybrid Recommender Systems : Survey and Experiments." . California State University, Fullerton. <http://josquin.cs.depaul.edu/~rburke/pubs/>. Diakses pada : 9 November 2013.
- [4] Fouss, Piroette, Renders, Saerens. (2006). "Random-walk computation of similarities between nodes of a graph, with application to collaborative recommendation". *Universit'e catholique de Louvain, Place des Doyens 1, B-1348 Louvain-la-Neuve, Belgium*
- [5] Gediminas Adomavicius dan Alexander Tuzhilin. (2005). "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions". IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING.
- [6] Greg Linden, Brent Smith, and Jeremy York. (2003). "Amazon.com Recommendations Item-to-Item Collaborative Filtering". IEEE INTERNET COMPUTING.
- [7] Huang, Zan. (2005). "Graph-based Analysis on Recommendation System". *University of Arizona*.
- [8] J.B. Schafer, J. Konstan, J. Riedl. (2001). "E-Commerce Recommendation Application". GroupLens Research Project, Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455, USA.
- [9] Munir, Rinaldi. (2013). *Graf*. Institut Teknologi Bandung : Bandung.
- [10] O'Donovan, John & Smyth, Barry. (2005). "Trust in Recommender Systems". Department of Computer Science, University College Dublin, Belfield, Dublin 4, Ireland.
- [11] Paramita, Astrid. (2002). "Sistem rekomendasi produk dalam personalisasi situs e-commerce". Institut Teknologi Bandung.
- [12] Pasquale Lops, Marco de Gemmis and Giovanni Semeraro . (2011). "Recommender Systems Handbook". Department of Computer Science, University of Bari "Aldo Moro", Via E. Orabona, 4, Bari (Italy).
- [13] Robin van Meteren dan Maarten van Someren. (2000). "Using Content-Based Filtering for Recommendation". NetlinQ Group, Gerard Brandtstraat 26-28, 1054 JK, Amsterdam.
- [14] Steck, Harald. (2010). Training and Testing of Recommender Systems on Data Missing Not at Random. Bell Labs, Alcatel-Lucent. Murray Hill, NJ 07974, USA