

Daftar Istilah

<i>Similarity</i>	Kemiripan antar pengguna terhadap pengguna lain atau kemiripan antar pekerjaan terhadap pekerjaan lain
<i>Social Aperture</i>	Social aperture pada kasus ini adalah nilai rekomendasi pada orang-orang berdasarkan pada preferensi pribadi dan kesmaan user terhadap user lainnya.
<i>Sparcity</i>	Data interaksi user ke pekerjaan yang kosong
<i>Hybrid</i>	Penggabungan antara 2 metode
<i>Ontologi</i>	Ontologi adalah deskripsi formal dari suatu struktur pada sistem berdasarkan entitas dan relasi yang telah di observasi

BAB 1. Pendahuluan

1.1. Latar Belakang

Mendapatkan informasi yang sesuai dengan minat pengguna menjadi suatu tantangan tersendiri pada era digital saat ini. Meningkatnya aliran data dan informasi menjadi suatu masalah saat saat user tidak dapat mencari produk yang pengguna butuhkan. *Recommender system* mengumpulkan informasi dari berbagai pilihan user kepada sekumpulan item [1].

Salah satu bukti nyata pengaruh *recommender system* yang bekerja pada saat ini adalah pada sistem iklan pekerjaan yang hanya mengandalkan browsing dan pencarian pada mesin pencari saat ini telah dikalahkan oleh sistem rekrutasi online seperti *careerBuilder.com* and *monster.com*. Kedua layanan online tersebut mencoba untuk memberi servis dalam jangka panjang untuk merekomendasikan pekerjaan kepada pengguna mereka dengan menyaring informasi berdasarkan profil yang disediakan oleh pengguna [3].

Pada tugas akhir ini, penulis akan menggunakan dataset dari website *xing* untuk merekomendasikan pekerjaan pada user. *Xing* adalah social network untuk bisnis yang digunakan oleh para pencari kerja dan perekrut yang menggunakan *Xing* untuk menemukan kandidat yang cocok pada pekerjaan yang ditawarkan [11].

1.2. Rumusan Masalah

Recommender system yang ada pada saat ini terdiri dari empat tipe yaitu *collaborative filtering*, *content based recommender system*, *knowledge based recommender system* dan *hybrid recommender system* [1].

Masing masing metode tersebut memiliki kelemahan dan kelebihan seperti *collaborative filtering* dapat merekomendasikan pekerjaan berdasarkan kumpulan user yang memiliki kesukaan yang sama dengan user tersebut. *Knowledge based recommender system* memodelkan user berdasarkan pada pengetahuan mengenai user tersebut dan merekomendasikan pekerjaan jika pekerjaan tersebut sudah memenuhi dengan kebutuhan user. Namun kedua metode tersebut memiliki kelemahan yaitu pada *knowledge based recommender system* masalah seperti ini juga masih dapat terjadi, meskipun pada dasarnya mengambil karakteristik utama

dari suatu pekerjaan tetapi jika user tidak memberi deskripsi yang jelas maka akan terjadi kegagalan saat rekomendasi [4]. Pada *collaborative filtering* masalah terjadi jika ada sparcity pada data sehingga jika suatu pekerjaan yang tidak pernah direkomendasikan oleh user lain, pekerjaan tersebut akan jarang muncul pada rekomendasi.

Pada tugas akhir ini penulis akan menggabungkan metode *knowledge based* dan *collaborative filtering* agar user dapat diberikan rekomendasi berdasarkan karakteristik pekerjaan dari preferensi user setelah itu diberikan rekomendasi dari social network menggunakan *collaborative filtering*. Rekomendasi hybrid akan menggabungkan dua metode ini dengan menggunakan *social aperture* sebagai parameter pengaruh prediksi pada *knowledge based recommender system* dan *collaborative filtering* [4].

Social aperture pada kasus ini adalah nilai rekomendasi pada orang-orang berdasarkan pada preferensi pribadi dan kesamaan user terhadap user lainnya. *Social aperture* akan dibagi menjadi 2 yaitu *moderate* dan *liberal*. *Social aperture moderate* mengartikan user lebih cenderung sebanyak 75% memilih rekomendasi dari berdasarkan preferensi pribadinya, namun menerima hasil rekomendasi dari user lain sebanyak 25%. *Social aperture liberal* menggabungkan hasil rekomendasi dengan nilai yang seimbang yaitu 50% berdasarkan preferensi pribadi dan 50% berdasarkan pada rekomendasi dari user lain [4].

Berdasarkan latar belakang di atas rumusan masalah adalah berbagai berikut:

1. Bagaimana mengimplementasikan rekomendasi hybrid dengan metode *knowledge based recommender system* dan *collaborative filtering* dapat menyelesaikan masalah rekomendasi untuk merekomendasikan pekerjaan pada user berdasarkan pada website xing ?
2. Bagaimana pengaruh *social aperture* pada hasil prediksi interaksi dan rekomendasi antara pekerjaan dan user pada hybrid recommender system ?
3. Bagaimana hasil prediksi dan rekomendasi hybrid recommender system dibandingkan dengan metode lain ?

Batasan masalah pada tugas akhir ini adalah :

1. Pengerjaan tugas akhir ini hanya sampai prediksi interaksi user terhadap suatu pekerjaan tidak sampai pengujian pada online user
2. Hasil similarity yang diambil merupakan 20 similarity terbesar. [3]
3. Seorang user minimum melakukan 15 interaksi terhadap pekerjaan.

1.3. Tujuan

Tujuan dari pelaksanaan tugas akhir ini adalah sebagai berikut :

1. Mengimplementasikan dan menganalisa *hybrid recommender system* berdasarkan pada dataset pekerjaan dari website xing.
2. Mengetahui pengaruh *social aperture* pada *hybrid recommender system*.
3. Membandingkan hasil rekomendasi dari *hybrid recommender system* dengan metode lain seperti *collaborative filtering* dan *knowledge based recommender system*

1.4. Metodologi Penyelesaian Masalah

Metodologi yang digunakan pada penelitian ini terdapat tujuh tahapan, antara lain :

1. Studi Literatur

Pada bagian ini penulis mempelajari referensi mengenai recommender system dan mencari metode yang tepat untuk diimplementasikan.

2. Pendefinisian Masalah

Mendefinisikan masalah dan menggunakan metode-metode berdasarkan referensi yang didapat dari Studi Literatur untuk dijadikan bahan penelitian pada Tugas Akhir ini.

3. Analisis Literatur

Melakukan analisis terhadap jurnal dan paper yang telah dipilih untuk mengetahui metode relevan yang akan digunakan.

4. Perancangan system

Merancang sistem berdasarkan pre processing dataset, arsitektur sistem, pengujian.

5. Implementasi

Menerapkan rekomendasi hybrid dengan menggunakan knowledge based recommender system dan collaborative filtering.

6. Pengujian Sistem

Menguji dengan menggunakan nilai RMSE dan f1 score dan melihat tingkat akurasi.

7. Analisis Hasil Pengujian

Pada tahap ini dilakukan analisis dari hasil keluaran sistem untuk menyelesaikan permasalahan yang diangkat.

8. Pembuatan Laporan

Tahap ini merupakan tahap akhir dari penelitian dimana dilakukan penyusunan laporan, dokumentasi dan kesimpulan berdasarkan analisis hasil pengujian.

1.5. Sistematika Penulisan

Tugas akhir ini menggunakan sistematika penulisan sebagai berikut:

- Bab I Pendahuluan

Bab ini berisi tentang uraian tugas akhir secara umum yang meliputi latar belakang, rumusan masalah, batasan masalah, tujuan, metodologi penelitian, dan sistematika penulisan.

- Bab II Landasan Teori

Bab ini berisi teori-teori pendukung yang akan digunakan selama penelitian.

- Bab III Perancangan Sistem

Bab ini berisi perancangan sistem berdasarkan analisis, pada bab ini berisi pengolahan dataset, pemodelan sistem.

- Bab IV Pengujian dan analisis hasil

Bab ini berisi penjelasan hasil pengujian dari sistem dan hasil analisis dari sistem yang sudah dibuat.

- Bab V Kesimpulan dan saran

Bab ini berisi kesimpulan dari hasil penelitian dan saran bagi pembaca yang ingin melanjutkan penelitian yang sudah dibuat oleh penulis.

BAB 2. Studi Literatur

2.1. Recommender System

Recommender system adalah suatu software tools dan teknik yang menyediakan saran agar pekerjaan dapat digunakan oleh user. Saran yang diberikan recommender system berasal dari berbagai macam *decision-making process*, seperti pekerjaan yang harus dibeli, musik yang didengarkan atau berita yang dibaca [6].

Secara khusus setiap teknik pada *recommender system* memiliki background data (informasi yang system miliki sebelum dimulai proses rekomendasi), input (informasi yang akan dimasukan pada sistem supaya dapat menghasilkan rekomendasi) dan algoritma yang menggabungkan antara *background data* dan input supaya dapat menyarankan suatu pekerjaan pada user. Pada dasarnya recommender system memiliki tiga teknik rekomendasi dasar seperti yang digambarkan pada table 1. Asumsikan bahwa I adalah kumpulan pekerjaan yang mungkin di rekomendasikan, U adalah kumpulan *user* yang preferensinya sudah diketahui, u adalah user yang rekomendasinya akan digenerate dan i adalah pekerjaan yang ingin diprediksi berdasarkan preferensi dari user u [6].

Tabel 1 Teknik Rekomendasi

Teknik	Background data	Input	Algoritma
<i>Collaborative filtering</i>	Rating dari U untuk pekerjaan I	Rating dari u pada pekerjaan I	Identifikasi user di U yang sama dengan user u dan prediksi rating mereka untuk pekerjaan I
<i>Content based recommender System</i>	Fitur dari pekerjaan dalam I	Rating dari user u untuk pekerjaan I	<i>Generate classifier</i> yang cocok dengan aturan rating u yang lalu gunakan pada i
<i>Knowledge based recommender system</i>	Fitur dari pekerjaan I yang diketahui dari domain knowledge tentang bagaimana pekerjaan dapat	Deskripsi mengenai kebutuhan dan minat dari user u .	Merepresentasikan pekerjaan berdasarkan pengetahuan yang didapat dari u

	memenuhi kebutuhan user.		
--	--------------------------	--	--

2.2. Knowledge Based Reommender System

Knowledge based system merekomendasikan pekerjaan berdasarkan *domain knowledge* tentang pengaruh fitur dari pekerjaan terhadap kebutuhan dan preferensi user [1]. Pada kasus ini *similarity score* pada *knowledge based system* merepresentasikan kesamaan fitur antara pekerjaan yang berbeda [4]. Nilai *similarity* antar individu bisa didapatkan dengan rumus 1 yang diimplementasikan pada algoritma 1 dan 2. *Similarity* akan direpresentasikan dengan *reference matrix*.

$$S_{ab} = \sum_{i=1}^{\#p} \left(\frac{\text{common}(a, b, P[i])}{\max(\text{deg}(a, P[i]), \text{deg}(b, P[i]))} \right) * \text{Weight}(P[i]) \quad (1)$$

Hasil dari persamaan diatas berupa *reference matrix* seperti dibawah ini:

$$\text{RefSim} = \begin{bmatrix} 1 & S_{12} & S_{13} & \dots & S_{1n} \\ S_{21} & 1 & S_{23} & \dots & S_{2n} \\ S_{31} & S_{32} & 1 & \dots & S_{3n} \end{bmatrix} \quad (2)$$

Keterangan :

- P (*properties*) adalah vector yang mengandung sejumlah atribut dan hubungan antar atribut pada *reference class* dalam formula tersebut digunakan untuk menghitung *similarity*, contohnya:
P=[career_level, discipline_id, country], #P=3
- Deg(a,p) merepresentasikan jumlah *instance* yang terhubung dengan individual a melalui relasi p, Contoh:
a.Hasdiscipline_id=[1]
- Common(a,b,p) merepresentasikan jumlah *instance* sama yang terhubung dengan adan p melalui relasi dari *property* p, contoh :
a.Hasdiscipline_id =[1]
b.Hasdiscipline_id [1]
common(a,b,p)=1

- Weight(p) menandakan pentingnya suatu property pada relasi p dalam formula, dalam kasus ini weight didapat dari $1/\#P$. Contoh jika $\#P=5$, maka masing masing weight pada pekerjaan adalah 0.2

<p>Algoritma 1: fungsi Similarity</p> <p>Input : A dan B merepresentasikan sekumpulan instansi dimana $\{\{E_1, E_2, E_3, \dots, E_n\} E_x \text{ adalah sekumpulan elemen bertipe } x\}$; Set Weight = $\{(r_1, k_1), (r_2, k_2), (r_3, k_3), \dots, (r_m, k_m) r_x \text{ adalah property/relationship dan } k_x \text{ adalah weight pada persamaan}\}$.</p> <p>Output : similarity rate antara A dan B dinormalisasikan ke dalam $[0,1]$</p> <pre> subTotal ← 0 totalWeight ← 0 foreach (r_i, k_i) ∈ Weight do elementsA ← E_{r_i} ∈ A elementsB ← E_{r_i} ∈ B totalWeight ← totalWeight + k_i common ← elementsA ∩ elementsB denominator = max(#elementsA, #elementsB) if denominator ≠ 0 then subTotal ← subTotal + $\frac{\text{common} * k_i}{\text{denominator}}$ end end return (subTotal/totalWeight) </pre>
--

<p>Algoritma 2: Algoritma yang menghasilkan Similarity Reference Matrix RefSim</p> <p>Input : Sekumpulan instansi $P = (P_1, P_2, P_3, \dots, P_n)$; Set Weight berisi seluruh property/relationship dan weight mereka pada persamaan</p> <p>Output : Similarity Reference Matrix</p> <pre> for i ← 1 to #P do for j ← 1 to #P do if i < j then Refsim[i][j] ← Similarity(P_i, P_j, Weight) end end end return RefSim </pre>

Kemudian setelah didapatkan *reference matrix* maka akan dilakukan proses inferensi user untuk mendapatkan hubungan dari suatu pekerjaan terhadap preferensi dari user yang ditujunya sesuai dengan relasi pada objek dan user.

2.3. Ontologi

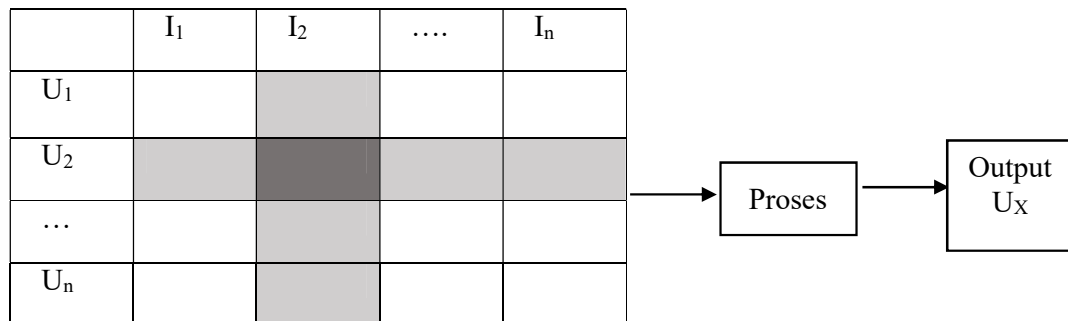
Untuk memodelkan *domain knowledge* pada *knowledge based recommender system* maka dibutuhkan ontologi. Ontologi adalah model formal dari suatu struktur

sistem yang dapat dimengerti manusia dan dibaca mesin. Ontologi terdiri dari atribut, relasi, entitas dan axiom. Ontologi dapat menyediakan konseptualisasi yang beraneka ragam untuk menggambarkan domain pekerjaan dari suatu organisasi yang merepresentasikan konsep dan relasi dari aktivitas kerja. Ontologi dapat membantu *recommender system* bekerja pada lingkungan yang memiliki banyak kelas yang memungkinkan pendekatan *knowledge based* bekerja dengan algoritma machine learning [7].

2.4. Collaborative Filtering

Collaborative filtering merupakan metode yang paling lama dan banyak digunakan pada suatu recommender system. Pada *collaborative filtering* active user akan memberikan rekomendasi kepada user lain yang memiliki preferensi yang sama berdasarkan similarity dari tiap tiap user. Proses *collaborative filtering* dilakukan dengan melihat persamaan rating pada pekerjaan dengan metode user to user. Similarity antara user dapat dihitung berdasarkan dari rating yang diberikan pada oleh user. Metode *collaborative filtering* juga sering disebut sebagai “*people-to-people correlation*” [1].

Proses cf ini akan mencari nilai prediksi yang berupa prediksi interaksi terhadap suatu pekerjaan. Proses pada user based *collaborative filtering* dibagi menjadi 3 tahap yaitu masukan(*input*), proses, dan keluaran. Inputan berupa irisan dari interaksi terhadap pekerjaan dan user.



Gambar 1 Proses Collaborative Filtering

Proses pada *collaborative filtering* dimulai dengan menghitung interaksi pengguna yang memiliki ketertarikan yang sama didapatkan dari rumus similarity *user based collaborative filtering* seperti pada persamaan 3 [1]

$$S_{uj} = \frac{\sum_{x \in I_{ju}} (U_x - \bar{U})(J_x - \bar{J})}{\sqrt{\sum_{x \in I_{ju}} (U_x - \bar{U})^2 \sum_{x \in I_{ju}} (J_x - \bar{J})^2}} \quad (3)$$

Untuk menghitung interaksi pengguna yang memiliki ketertarikan yang sama didapatkan dari rumus *user based collaborative filtering* seperti pada persamaan 4.

$$U_x = \bar{U} + \frac{\sum_{J \in \text{Raters of } x} (J_x - \bar{J}) S_{UJ}}{\sum_{J \in \text{Raters of } x} |S_{UJ}|} \quad (4)$$

Keterangan :

- U_x adalah interaksi yang akan di prediksi dari predicted user U pada pekerjaan x
- \bar{U} adalah rata rata interaksi dari seluruh user pada U
- J_x adalah rating dari user u pada pekerjaan x
- S_{uj} adalah similarity antara user U dan user J

2.5. Hybrid Recommender System

Hybrid recommender system adalah suatu teknik yang menggabungkan 2 teknik dasar seperti yang disebutkan diatas. *Hybrid recommender system* memiliki beberapa teknik hybrid seperti yang dijelaskan pada tabel 2 [4]

Table 2 Teknik Rekomendasi hybrid

Metode hybrid	Deskripsi
<i>Weighted</i>	Nilai Pada rekomendasi antara beberapa teknik di gabungkan berdasarkan parameter tertentu
<i>Switching</i>	Sistem menggabungkan 2 buah teknik secara bergantian
<i>Mixed</i>	Rekomendasi dari beberapa recommender disatukan
<i>Cascade</i>	Suatu rekomendasi memperbaiki hasil dari rekomendasi lain

<i>Feature Augmentation</i>	Output dari suatu teknik digunakan untuk input dari teknik lainnya
<i>Meta-level</i>	Model yang dipelajari dari suatu recommender digunakan untuk input pada recommender lain

Pada tugas akhir ini penulis akan fokus menggunakan metode hybrid dengan teknik *weighted* berdasarkan social aperture yang diberikan.

2.6. K Fold Cross Validation

K-fold cross validation adalah salah satu cara untuk meningkatkan validasi pada holdout methode. Kumpulan data dibagi menjadi subset k, dan holdout methode diulang sebanyak k kali. Setiap kali, salah satu subset k digunakan sebagai test set dan lainnya subset k-1 disatukan untuk membentuk satu set training. Maka rata-rata error di semua percobaan k dihitung. Keuntungan dari metode ini adalah kurang penting bagaimana data akan dibagi [8].

2.7. Top N Recommendation

Amazon.com's Book Matcher, Levi's Style Finder and My CDNOW memanfaatkan rekomendasi dengan menggunakan *top N list*. Setiap situs akan mempelajari atribut yang disenangi dan tidak disenangi oleh user. Sehingga setiap situs bias mendapatkan daftar preferensi personal dari user berdasarkan pada pekerjaan yang disenangi dan tidak disenangi user, dari data yang telah disediakan maka bisa didapatkan daftar N pekerjaan yang belum pernah dirating oleh user namun berpotensi untuk di sukai oleh user. *Top N recommendation* dapat membantu user dalam mengambil keputusan pada suatu situs untuk dapat menentukan pekerjaan yang mungkin akan dipilihnya [9].

2.8. Root Mean Square Error (RMSE)

Root mean square error merupakan suatu ukuran kesalahan yang mengukur perbedaan antara hasil prediksi dan hasil real dari suatu model. Dibandingkan dengan MAE, RMSE lebih sensitif terhadap nilai kesalahan yang besar.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (5)$$

dimana \hat{y}_i merupakan nilai yang diamati untuk observasi (nilai aktual) ke i dan y_i adalah nilai prediksi [2].

2.9. F1 Measure

F1-measure digunakan untuk melakukan pengukuran pada masalah seperti *binary classification*, *multi-label classification*, dan output hasil prediksi yang terstruktur. Persamaan yang digunakan untuk melakukan *F1 measure* membutuhkan nilai dari *Precision* dan *Recall* [4]. Perhitungan *F1 score* dapat dilakukan dengan persamaan dibawah ini :

$$\text{precision} = \frac{|\text{Rekomendasi pekerjaan yang relevan}|}{|\text{Jumlah seluruh pekerjaan yang direkomendasikan}|} \quad (6)$$

$$\text{recall} = \frac{|\text{Rekomendasi pekerjaan yang relevan}|}{|\text{Jumlah seluruh pekerjaan relevan}|} \quad (7)$$

$$\text{F1 - Measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (8)$$