# BAB I

# PENDAHULUAN

## Latar Belakang

Saat ini sistem rekomendasi telah diterapkan di berbagai domain seperti musik, film, buku, dan produk (Yang, 2019). Penggunaan teknik rekomendasi yang akurat dan efisien sangat penting bagi suatu sistem untuk memberikan hasil yang baik dan bermanfaat kepada setiap penggunanya (Isinkaye, et al., 2015). Sistem rekomendasi *Collaborative Filtering* dapat merekomendasikan *item* berdasarkan kumpulan *user* dengan preferensi yang sama (Rahmawati, et al., 2018). Tetapi, sistem rekomendasi ini mengalami masalah *sparsity* pada matriks *rating*. Banyak *user* hanya menilai sejumlah kecil *item* sehingga rekomendasi yang dihasilkan memiliki akurasi yang sangat rendah (Saeed & MansooriI, 2017). Sementara dalam konteks perpustakaan digital, banyaknya koleksi dan kebutuhan informasi yang beragam membuat *rating* dari *user* sangat diperlukan. Tetapi, *rating* pada perpustakaan digital cenderung jarang diberikan oleh *user* (Wenige & Ruhland, 2018). Oleh karena itu, diperlukan suatu teknik yang dapat memprediksi *rating* *user* terhadap *item* (Bobadilla, et al., 2020).

*Deep Learning*, sebagai jenis pendekatan *machine learning* telah berhasil diterapkan di banyak domain penelitian berbeda, seperti *computer vision*, *speech recognition*, *natural language processing* dan lain sebagainya (Liu & Wu, 2017). Menggunakan *Deep Learning* pada sistem rekomendasi memungkinkan model secara otomatis mempelajari fitur yang digeneralisasi dengan baik bagi *user* dan *item* dari sumber daya yang berbeda. Dengan menggunakan *Deep Learning* dalam pemodelan berbagai tipe data, sistem rekomendasi akandapat lebih memahami apa yang dibutuhkan *user* dan hal tersebut akan meningkatkan hasil rekomendasi (Ikasari, et al., 2018).

Model *Deep Learning* untuk *Collaborative Filtering* merupakan penggabungan *Collaborative Filtering* berbasis *Matrix Factorization* dengan algoritma *Deep Learning*. *Deep Collaborative Filtering* yang merupakan model *hybrid*, menggunakan matriks *rating* dan *side information* serta menjembatani *Matrix Factorization* dan fitur *learning*. *Deep Collaborative Filtering* mengintegrasikan *Matrix Factorization* dan fitur *Deep Learning* dengan memodelkan pemetaan antara *latent factors* yang digunakan dalam *Collaborative Filtering* dan *latent layers* dalam *Deep models* (Li, et al., 2015).

Penerapan *Deep Collaborative Filtering* pada perpustakaan digital akan memberikan rekomendasi *item* lebih akurat daripada hanya menggunakan *Collaborative Filtering*. *User* pada perpustakaan digital jarang memberikan *rating* pada *item* (Wenige & Ruhland, 2018). Jika hanya menggunakan *Collaborative Filtering*, hasil rekomendasi tidak akan akurat karena terjadi *sparsity* matriks *rating*. Oleh karena itu, *Deep Collaborative Filtering* diharapkan dapat mengatasi masalah tersebut pada aplikasi perpustakaan digital.

Berdasarkan uraian di atas, maka dikembangkan sebuah aplikasi yang dapat memberikan rekomendasi kepada *user* tanpa mengalami *sparsity* pada matriks *rating* dan dituangkan dalam tugas akhir dengan judul “APLIKASI REKOMENDASI PERPUSTAKAN DIGITAL MENGGUNAKAN METODE DEEP COLLABORATIVE FILTERING BERBASIS MOBILE DAN WEB”.

## Rumusan Masalah

Berikut rincian permasalahan yang telah dituliskan pada Latar Belakang :

1. Bagaimana penerapan dari sistem rekomendasi menggunakan metode *Deep Collaborative Filtering* pada perpustakaan digital?
2. Bagaimana metode *Deep Collaborative Filtering* mengatasi *sparsity* matriks *rating* pada perpustakaan digital?
3. Bagaimana *user* perpustakaan digital mendapatkan *item* yang sesuai dengan kriteria dan keinginannya?
   1. **Tujuan**

Tujuan tugas akhir ini yaitu sebagai berikut :

1. Menerapkan *Deep* *Collaborative Filtering*  pada perpustakaan digital.
2. Mengatasi *sparsity* matriks *rating* pada perpustakaan digital.
3. Mengembangkan aplikasi perpustakaan digital yang memberikan rekomendasi dalam menentukan *item* yang sesuai dengan kebutuhan dan keinginan *user*.
   1. **Manfaat**

Manfaat yang diharapkan dari tugas akhir ini adalah sebagai berikut :

1. Aplikasi perpustakaan digital yang bersifat *easy to use*, sehingga mempermudah dalam melakukan pencarian dan menampilkan hasil sesuai keinginan *user*.
2. Aplikasi ini dapat membantu masyarakat untuk terus membaca buku dimana saja dan kapan saja.
   1. **Batasan Masalah**

Batasan masalah pada penelitian implementasi *Deep Collaborative Filtering* untuk menyelesaikan rekomendasi pada perpustakaan digital dalam mendapatkan *item* sesuai dengan kebutuhan dan keinginan user sesuai dengan identifikasi dan rumusan masalah. Pembatasan masalah yang digunakan pada penelitian ini adalah:

1. *Rating user* digunakan sebagai parameter dalam sistem rekomendasi *Deep Collaborative Filtering* dengan menggunakan *history*  pencarian dan peminjaman yang dilakukan oleh pengguna dengan asumsi ketika *user* telah meminjam *item* perpustakaan berarti *user* telah memberikan *rating* pada *item* tersebut.
2. Hanya menggunakan teknik *Deep Collaborative Filtering* karena pada sistem rekomendasi ini *item* yang pernah dicari atau dipinjam *user* akan menjadi patokan untuk pemberian rekomendasi.
3. *Mobile application* sebagai produk dari Tugas Akhir ini hanya dapat dipasang pada *smartphone* dengan *operating system* Android versi Lollipop 5.0+ dan belum tersedia dalam sistem operasi IOS.
4. Jumlah aktor dalam sistem ini yaitu dari 3(tiga), terdiri dari admin, member dan *user* (aktor yang belum mendaftar ke dalam sistem) dengan masing-masing aktor yaitu sebagai berikut.

Admin :

1).Mengelola data item digital.

2). Mengelola peminjaman *item* digital.

4). Melakukan verifikasi member terdaftar.

5). Mengelola laporan.

Member :

1). Mendapat rekomendasi *item* digital dari algoritma *Deep Collaborative Filtering*.

2). Mencari, meminjam, membaca, mengembalikan dan memperpanjang *item* digital.

3). Memberi *rating* dan *review* *item* digital yang telah dipinjam.

*User* :

1). Mendapat rekomendasi *item* digital dari algoritma *Deep Collaborative Filtering*.

2). Mencari dan melihat rincian *item* digital.

3). Melakukan pendaftaran pada sistem.

* 1. **Metodologi Pengembangan Sistem**

Metodologi yang digunakan pada pengembangan sistem ini yaitu dengan model *waterfall*. Metode *waterfall* merupakan metodologi pengembangan sistem yang bersifat sistematis sehingga dapat menghasilkan software yang baik dan penataan perpustakaan yang terurut dengan rapi karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.

1. Analisa Proses

Pada tahapan ini perhitungan untuk algoritma *Deep Collaborative Filtering* yang akan digunakan dengan contoh kasus atau angka sederhana disajikan secara tahap demi tahap serta menggunakan *flowchart* untuk menganalisis proses algoritma yang digunakan*.*

1. Studi Literatur

Pada tahapan ini adalah mencari literatur yang berhubungan dengan topik yang diangkat. Tujuan dari tahapan ini adalah penulis memahami konsep, metode dan teknologi Native.

1. Data *Processing*

Pada tahap ini akan dilakukan pengolahan data dengan baik, dimana akan melakukan transformasi data ke suatu format yang prosesnya lebih mudah dan efektif sebagai kebutuhan pengguna. Tahapan *processing* ini terdiri dari pemilihan atribut, penggabungan data, *data cleaning*, dan standarisasi data.

1. Perancangan

Pada tahapan ini adalah tahapan lanjutan dari tahapan analisis dengan melakukan perancangan tampilan (*interface*) menggunakan figma, pemodelan basis data menggunakan Microsoft Visio untuk perancangan ERD *(Entity Relationship Diagrams).*

1. Implementasi

Pada tahapan ini dilakukan penulisan kode pemrograman dengan bahasa pemrograman kotlin untuk aplikasi mobile dan PHP (*Hypertext Preprocessor)* dan *Javascript* untuk web. Tool dan library yang digunakan untuk membangun aplikasi adalah  :

* 1. Framework laravel versi 8
  2. Bootstrap CSS versi 5.0
  3. Xampp versi 8.0.1
  4. Android Studio versi 4.0
  5. Emulator Android Lollipop 5.0+
  6. Library Ereader Futurepress/Epub.js

1. Pengujian

Pengujian software diperlukan untuk memastikan aplikasi yang dibangun dapat berjalan sesuai fungsionalitasnya.

1. Pengujian sistem rekomendasi *Deep Collaborative Filtering* untuk mengukur keakuratan nilai rekomendasi dimana nilai dari hasil pengujian pada sistem rekomendasi akan menjadi nilai parameter akurat atau tidaknya rekomendasi yang diberikan kepada *user*.
2. Pengujian fungsionalitas dari aplikasi perpustakaan, menggunakan metode *Black Box Testing*  untuk menguji perangkat lunak apakah hasilnya sesuai dengan yang diharapkan.
3. Kesimpulan hasil pengujian

Penarikan Kesimpulan yang diambil berdasarkan hasil pengujian yang dilakukan dengan metode *Deep Collaborative Filtering.*

**BAB II**

**Tinjauan Pustaka**

**2.1 Perpustakaan Digital**

Perpustakaan Digital didefinisikan sebagai kumpulan informasi elektronik yang berisi repository objek digital yang besar dan beragam, dan dapat diakses oleh sejumlah besar pengguna yang tersebar secara geografis. Objek digital yang dimaksud seperti teks, gambar, peta, suara, video, katalog, dan jurnal ilmiah (Khiste, et al., 2018).

**2.2 Sistem Rekomendasi**

Sistem rekomendasi merupakan fitur yang berfungsi untuk membantu pengguna untuk memberi saran kepada *user* tentang *item* apa yang sebaiknya digunakan atau dipilih. Secara umum Sistem Rekomendasi ditujukan untuk individu yang kekurangan pengalaman atau kompetensi yang cukup untuk mengevaluasi banyaknya jumlah alternatif *item* yang ada pada suatu kasus tertentu (Elahi, 2016) terdapat beragam teori yang dapat digunakan untuk membuat sistem rekomendasi seperti *Bayesian Algorithm* (Bobadilla, 2017)*, Decision Tree Algoritm* (Guabassi, 2016) dan *Matrix factorization-based* (Aleksandrova, 2017) dan lain lain. Contoh *Matrix factorization-based* dapat dilihat pada gambar 2.1.



Gambar 2. 1 Contoh *Matrix factorization-based*

(Sumber : <https://developers.google.com/machine-learning/recommendation/collaborative/matrix>)

**2.3 Collaborative Filtering**

*Collaborative Filtering* merupakan pemfilteran kolaboratif yang relatif sederhana, efektif, dan telah digunakan secara luas oleh banyak situs web komersial. *Collaborative Filtering* memanfaatkan data *rating* *item* yang ada oleh *user* untuk membuat prediksi tentang preferensi *user*. Pendekatan *Collaborative Filtering* dapat dibagi menjadi dua kelompok: Pendekatan berbasis memori dan model. Pendekatan berbasis memori (berbasis *heuristik*) membuat prediksi berdasarkan kesamaan antara *user* dan *item*. Pendekatan berbasis model berusaha membuat model prediksi melalui *machine learning*. Secara khusus, model berbasis faktorisasi matriks telah memperoleh popularitas karena memiliki akurasi dan skalabilitas yang relatif tinggi. Meski demikian, Collaborative Filtering masih memiliki tantangan terhadap *sparsity* matriks *rating* (Yang, et al., 2016).

**2.3 Matrix Factorization**

*Matrix Factorization* adalah pendekatan *Collaborative Filtering* yang paling efektif. Ini memungkinkan kita untuk menemukan *latent factor* interaksi *user-item* dengan memfaktorkan matriks interaksi ke dalam *latent space* fitur *user-item*. Salah satu metode *Matrix Factorization* klasik adalah *Probabilistic Matrix Factorization* (PMF). Banyak algoritma telah dikembangkan untuk meningkatkan kinerja PMF, dengan memasukkan *side information* seperti hubungan sosial. Tetapi metode *Matrix Factorization* mengalami masalah *cold-start*, yaitu rekomendasi apa yang harus dibuat ketika *user* / *item* baru tiba di sistem. Masalah lain yang sering muncul di banyak aplikasi dunia nyata adalah ketersebaran data atau cakupan yang berkurang. Memasukkan *side information* telah menunjukkan kinerja yang menjanjikan dalam *Collaborative Filtering*. Tetapi akan bermasalah jika *side information* tidak lengkap. Oleh karena itu, fitur pembelajaran untuk *Matrix Factorization* sangat penting untuk diterapkan.

Sistem mempelajari *latent factors* dengan meminimalkan *objective function* sebagai berikut :



Gambar 2. 2 Sistem mempelajari *latent factors*

Di mana *l (R, U, V)* adalah fungsi untuk memprediksi *rating* menggunakan *latent factor* U dan V dan dua istilah terakhir adalah regularisasi yang digunakan untuk menghindari *overfitting*. *||· ||*F menunjukkan *Frobenius norm*. (Li, et al., 2015)

# DAFTAR PUSTAKA

Bobadilla, J., Alonso, S. & Hernando, A., 2020. Deep Learning Architecture for Collaborative Filtering Recommender Systems. p. 1.

Chen, J. et al., 2020. Deep attention user-based collaborative filtering for recommendation. p. 2.

Elahi, M., 2016. A survey of active learning in collaborative filteringrecommender systems. *C O M P U T E R S C I E N C E R E V I E W2 0 ( 2 0 1 6 ).*

Ikasari, D., Suhendra, A. & Farida, N., 2018. Metode Deep Learning Pada Sistem Rekomendasi : Review Paper. Volume 2, p. 47.

Isinkaye, F., Folajimi, Y. & Ojokoh, B., 2015. Recommendation systems: Principles, methods and. p. 4.

Li, S., Kawale, J. & Fu, Y., 2015. Deep Collaborative Filtering via Marginalized Denoising Auto-encoder. pp. 812, 814.

Liu, J. & Wu, C., 2017. Deep Learning Based Recommendation: A Survey. Volume 424, p. 451.

Rahmawati, S., Nurjanah, D. & Rismala, R., 2018. Analisis dan Implementasi Pendekatan Hybrid untuk Sistem Rekomendasi dengan Metode Knowledge Based Recommender System dan Collaborative Filtering. *Ind. Journal on Computing,* 3(2), p. 12.

Saeed, M. & MansooriI, E. G., 2017. A NOVEL FUZZY-BASED SIMILARITY MEASURE FOR COLLABORATIVE FILTERING TO ALLEVIATE THE SPARSITY PROBLEM. *Journal of Fuzzy Systems,* Volume 14, p. 2.

Wenige, L. & Ruhland, J., 2018. Retrieval by recommendation: using LOD technologies to improve. *digital library search,* p. 2.

Zhang, M. & Yang, Z., 2019. GACOforRec: Session-Based Graph Convolutional Neural Networks Recommendation Model. p. 114077.

ZHANG, S., 2019. *Deep Learning based Recommender System: A Survey and New Perspectives.*