SISTEM REKOMENDASI FILM MENGGUNAKAN DATA USER-END DAN KNOWLEDGE GRAPH CONVOLUTIONAL NETWORK

**PROPOSAL**

Diajuakan untuk Memenuhi Salah Satu Syarat

Matakuliah Tugas Akhir 1

Jenjang Strata 1 pada Program Studi Informatika

Universitas Jenderal Achmad Yani

Oleh:

Nama : Muhammad Rizki Yanuar

NIM : 3411211062

|  |  |
| --- | --- |
| Pembimbing 1 | Pembimbing 2 |
| Fajri Rakhmat Umbara, S.T., M.T  NID: 412185888 | Agus Komarudin, S.Kom, M.T.  NID: 412175878 |

Logo

Description automatically generated

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS SAINS DAN INFORMATIKA

UNIVERSITAS JENDERAL ACHMAD YANI

2025

ABSTRAK

Dalam era digital, sistem rekomendasi memainkan peran penting dalam membantu pengguna menemukan film sesuai preferensi mereka. Namun, metode tradisional seperti collaborative filtering dan content-based filtering sering menghadapi masalah sparsity dan cold start, terutama untuk pengguna baru. Penelitian ini mengusulkan implementasi Knowledge Graph Convolutional Network (KGCN) yang memanfaatkan data user-end untuk mengatasi keterbatasan tersebut. Knowledge Graph (KG) digunakan untuk mengintegrasikan atribut film dan informasi pengguna dalam struktur graf yang terarah, memungkinkan sistem untuk menghasilkan rekomendasi lebih relevan. Metode Graph Convolutional Network (GCN) diaplikasikan untuk mengagregasi informasi dari tetangga terdekat, memanfaatkan hubungan antar entitas dalam graf. Penelitian ini bertujuan mengembangkan sistem rekomendasi film yang lebih responsif terhadap preferensi pengguna, dengan mengevaluasi performa KGCN dibandingkan metode tradisional. Dataset MovieLens digunakan sebagai sumber data utama untuk menguji efektivitas pendekatan ini. Hasil penelitian diharapkan mampu meningkatkan akurasi rekomendasi melalui integrasi umpan balik pengguna secara eksplisit maupun implisit. Dengan pendekatan ini, sistem rekomendasi tidak hanya memanfaatkan atribut item, tetapi juga memperhitungkan preferensi dan interaksi pengguna secara komprehensif. Penelitian ini memberikan kontribusi signifikan terhadap pengembangan sistem rekomendasi berbasis KG, terutama dalam konteks permasalahan cold start dan sparsity.

Kata Kunci: Knowledge Graph, Graph Convolutional Network, Sistem Rekomendasi Film, Cold Start, Sparsity, Data User-End

1. Latar Belakang Masalah

Dalam industri film, sistem rekomendasi sangat penting untuk membantu pengguna menemukan film yang sesuai dengan preferensi mereka. Dengan rilisnya ribuan film setiap tahun, menimbulkan tantangan besar yaitu bagaimana menyaring pilihan sehingga pengguna dengan mudah menemukan film yang cocok bagi mereka. Meskipun telah digunakan secara luas, metode tradisional seperti *collaborative filtering* dan *content-based filtering* sering kali menghadapi masalah *sparsity* dan *cold start [1],* dikarenakan data interaksi pengguna dengan item terbatas atau tidak ada untuk pengguna baru bahkan sampai menyebabkan risiko overfitting pada model [2].

Untuk mengatasi masalah ini, penelitian telah beralih ke pendekatan *deep learning*, termasuk penggunaan *Knowledge Graph Convolutional Networks* (KGCN). *Knowledge Graph* (KG) merupakan sebuah grafik yang menyimpan informasi terarah dengan simpul yang mewakili entitas (seperti pengguna, item atau atribut) dan tepi yang mewakili hubungan antar entitas. Salah satu keunggulan KG adalah kemampuan dalam menghubungkan berbagai atribut dan informasi yang relevan untuk pengguna dan item. *Graph Convolutional Network* (GCN) merupakan metode yang dirancang untuk bekerja dengan data yang terstruktur dalam bentuk graf. Pada GCN terdapat fitur utama yaitu kemampuan dalam mengagregasi dan menggabungkan informasi dari tetangga terdekat, desain semacam ini memiliki 2 keuntungan: (1) Struktur kedekatan lokal secara efektif dicatat dan disimpan di setiap entitas melalui metode agregasi. (2) Tetangga diberi bobot berdasarkan skor yang bergantung pada pengguna tertentu dan hubungan antar entitas. Skor ini menggambarkan informasi konten dari KG serta minat relasional pengguna itu sendiri [3].

Namun penerapan KG pada sistem rekomendasi memiliki beberapa tantangan seperti masalah dimensi yang tinggi dan heterogenitas yang disebabkan adanya berbagai macam entitas dan hubungan yang terbentuk [3]. Selain itu pada GCN juga mengalami masalah jika berurusan dengan graph heterogenitas yang berskala besar, dimana model GCN umumnya mengalami kompleksitas waktu yang mahal dan ukuran memori yang besar dikarenakan jumlah nodes dan interaksi yang sangat besar, sehingga untuk mengurangi waktu komputasi dan penggunaan memori yang besar maka diterapkan metode sampling pada graph heterogenitas, dimana teknik sampling ini akan mengambil nodes yang lebih kecil pada *layer* tertentu tetapi representatif dari distribusi yang memberikan bobot lebih pada *layer* tertentu [2].

Berbagai penelitian telah dilakukan dalam pengembangan sistem rekomendasi yang memanfaatkan data item-end dan user-item dengan menggunakan berbagai metode. Pada penelitian [3] penggunaan KGCN dengan memanfaatkan data item-ends dan menggunakan *uniformly* *sampling fixed-size neighborhood* untuk mengatasi jumlah nodes tetangga yang sangat besar, yang menghasilkan rekomendasi berdasarkan atribut-atribut item dan mengurangi waktu komputasi karena menentukan jumlah nodes tetangga yang akan diambil. Penelitian ini menunjukkan bahwa KGCN dapat efektif dalam mengidentifikasi keterkaitan antar item dari atribut yang terdapat dalam *Knowledge Graph* dan *uniformly* *sampling fixed-size neighborhood* menunjukkan peningkatan rata-rata AUC 4,4% karena dapat menangkap informasi yang jauh lebih banyak. Meskipun penelitian ini memberikan hasil signifikan terhadap pengembangan sistem rekomendasi berbasis item, masih terdapat celah dalam pemanfaatan data user-end dan teknik sampling. Penelitian lebih lanjut diperlukan untuk mengeksplorasi bagaimana umpan balik pengguna, baik bersifat eksplisit maupun implisit dapat diintegrasikan ke dalam model KGCN dan bagaimana teknik *importance sampling* dapat meningkatkan akurasi model, dengan demikian sistem rekomendasi tidak hanya bergantung pada atribut item tetapi juga mempertimbangkan preferensi dan pengalaman pengguna serta menangkap informasi yang jauh lebih relevan ketika pemilihan nodes tetangga.

Pada penelitian [4] penggunaan sistem rekomendasi hybrid yang menggabungkan *Collaborative Filtering, Content-Based Filtering,* dan teknik *Self-Organizing* (SOM) dengan memanfaatkan data user-item sebagai dasar untuk mengembangkan model rekomendasi. Kemudian pendekatan hybrid ini memanfaatkan metode *weighted* dengan menggabungkan skor dari collaborative filtering dan content-based filtering untuk menghasilkan satu rekomendasi dan *Feature Augmentation* dengan memanfaatkan output dari hasil rekomendasi *collaborative dan content-based* kemudian menjadikan hasil rekomendasi tersebut sebagai informasi tambahan dalam menciptakan model rekomendasi. Penelitian ini menyarankan penerapan *Deep Learning* seperti *Graph Neural Network* (GNN) untuk menangkap pola interaksi yang lebih kompleks, pengembangan strategi untuk menangani masalah cold start bagi pengguna baru atau item baru, dan mengintegrasikan data user-end untuk mengeksplorasi bagaimana umpan balik eksplisit dan implisit dari pengguna dapat meningkatkan akurasi rekomendasi.

Penelitian ini berfokus pada mengatasi tantangan utama dalam sistem rekomendasi film, seperti masalah sparsity, cold-start, dan efisiensi komputasi. Dengan mengintegrasikan data user-end, termasuk umpan balik eksplisit dan implisit, ke dalam Knowledge Graph Convolutional Network (KGCN), sistem rekomendasi dapat menangkap preferensi pengguna secara lebih personalisasi, tidak hanya bergantung pada atribut item. Selain itu, teknik importance sampling diterapkan untuk mengoptimalkan pemilihan nodes tetangga dalam Knowledge Graph, memastikan hanya informasi yang relevan diambil, sehingga mengurangi beban komputasi tanpa mengorbankan akurasi rekomendasi. Performa model yang diusulkan akan dievaluasi dan dibandingkan dengan metode tradisional serta pendekatan deep learning lainnya untuk mengukur peningkatan dalam akurasi, relevansi rekomendasi, dan efisiensi waktu komputasi.

2. Rumusan Masalah

Berdasarkan latar belakang yang telah diidentifikasi, permasalahan utama dalam sistem rekomendasi ini meliputi sparsity dan cold-start, yang terjadi akibat interaksi yang jarang atau pada item serta pengguna baru, serta efisiensi komputasi yang rendah akibat waktu pembelajaran yang lama pada graf berukuran besar. Untuk mengatasi permasalahan tersebut, penelitian ini akan mengeksplorasi dua aspek utama, yaitu pemanfaatan data user-end dan penerapan teknik importance sampling dalam model Knowledge Graph Convolutional Network (KGCN).

3. Batasan Masalah

Penelitian ini hanya menggunakan dataset dari platform MovieLens tanpa menggabungkan data diluar platform tersebut.

4. Tujuan Penelitian

1. Menggunakan metode Knowledge Graph Convolutional Network pada data user-end untuk menangani masalah cold start, sparsity.
2. Penggunaan metode *importance sampling* pada Knowledge Graph Convolutional Network untuk mengurangi waktu komputasi.

5. Luaran dan Manfaat Penelitian

Luaran dari penelitian ini membuat sistem rekomendasi film menggunakan Knowledge Graph Convolutional Network (KGCN) dengan memanfaatkan data user-end dan importance sampling, sehingga dapat meningkatkan akurasi model dan memberikan rekomendasi yang relevan berdasarkan preferensi pengguna serta dapat mengurangi waktu komputasi model dalam pembelajaran. Manfaat yang dapat diambil dari penelitian ini adalah kemampuan untuk memberikan rekomendasi film yang lebih relevan dan personal kepada pengguna, dengan mempertimbangkan interaksi dan umpan balik dari pengguna.

6. Tinjauan Pustaka

1. Matrix Interaksi User-Item

Pada bentuk sistem rekomendasi, terdapat interaksi yang secara *implicit* diberikan oleh pengguna mengenai preferensi yang dipilih (seperti, melihat, klik, membeli, menilai). Terdapat sekumpulan pengguna di definisikan sebagai dan sekumpulan item di definisikan sebagai . Pada persamaan (1) Matrix interaksi user-item didefinisikan sebagai:

|  |  |
| --- | --- |
|  | (1) |

Penjelasan:

Matrix interaksi antara pengguna dan item

Himpunan bilangan real dalam matrix

Jumlah baris yang mewakili jumlah pengguna dalam matrix

Jumlah kolom item yang mewakili jumlah item dalam matrix

Matrix pengguna-item tersebut disesuaikan dengan umpan balik *implicit* pengguna, di mana 1 yang mengindikasikan bahwa user tersebut berinteraksi dengan item , dan sebaliknya 0 [3].

1. Knowledge Graph

Knowledge Graph (KG) adalah struktur data yang menyimpan informasi dalam bentuk entitas dan hubungan antar entitas. Setiap node dalam KG merujuk pada entitas atau konsep tertentu, sedangkan edges merujuk pada jenis hubungan antara entitas tersebut. KG digunakan untuk memperkaya model preferensi pengguna dengan informasi tambahan yang tidak hanya bergantung pada interaksi pengguna-item sehingga rekomendasi menjadi lebih relevan dan akurat [1].

Struktur Knowledge Graph:

* Nodes

Dalam konteks sistem rekomendasi film, nodes dapat mencakup berbagai entitas seperti pengguna, film, genre, sutradara dan aktor. Setiap node mewakili sebuah item atau entitas yang relevan dengan domain rekomendasi.

* Edges

Edges merupakan penghubung antar node yang menampilkan koneksi di antara entitas. Misalnya “hasGenre” (memiliki genre), “hasDirector” (memiliki sutradara), atau “rates” [5]

*Knowledge graph* menyimpan struktur informasi dalam bentuk grafik heterogen, pada knowledge graph didefinisikan menjadi sekumpulan entitas dan menjadi sekumpulan relasi yang mencakup hubungan dari 2 arah, baik hubungan biasa ataupun sebaliknya, seperti relasi ‘director’ dan ‘directed-by’. Pada persamaan (2) knowledge Graph dibuat sebuah triplet didefinisikan sebagai:

|  |  |
| --- | --- |
|  | (2) |

Penjelasan:

Sekumpulan triple yang menyimpan informasi hubungan antar entitas

kepala, relasi dan ekor dari sebuah triple pengetahuan

Himpunan entitas dalam knowledge graph

Himpunan relasi dalam knowledge graph [6]

*Knowledge graph* dapat diperkaya dengan informasi tentang preferensi pengguna dalam sistem rekomendasi, Sistem rekomendasi bertujuan untuk mempelajari fungsi dalam memprediksi probabilitas interaksi penggguna dengan item. Matriks interaksi pengguna-item serta *Knowledge Graph ,* pada persamaan (3)digunakan untuk memprediksi probabilitas interaksi yang dilakukan oleh pengguna dengan item[7]:

|  |  |
| --- | --- |
|  | (3) |

Penjelasan:

Nilai prediksi probabilitas pengguna akan berinteraksi dengan item

Fungsi untuk memprediksi nilai

Input :

Parameter model mengoptimalkan fungsi prediksi

Matriks interaksi pengguna-item

*Knowledge Graph* memeberikan konteks tambahan tentang hubungan antar entitas

1. Graph Convolutional Network

Graph Convolutional Network (GCN) adalah jenis jaringan saraf yang dirancang untuk dirancang untuk bekerja dengan data yang terstruktur dalam bentuk graf. Berdasarkan metode ekstraksi fitur GCN terbagi 2, GCN berdasarkan domain spektrum dan domain spasial. GCN menggunakan domain spektrum berfokus pada analisis data dalam domain frekuensi, dimana metode ini melibatkan transformasi sinyal kedalam domain spektrum, sedangkan GCN berdasarkan domain spatial mendefinisikan konvolusi grafik dengan memulai dari simpul pusat yang mengagregasi informasi dari simpul tetangganya untuk menangkap hubungan lokal antar simpul. Konsep utama GCN adalah membuat embedding node dengan memodelkan pengiriman pesan dalam struktur graf, dengan meng-*aggregasi* data dari tetangga, sehingga setiap node memperoleh embeddingnya sendiri [8].

Agar GCN dapat menangkap lebih banyak informasi dari tetangga user dan item, dibuat sebuah layer. Layer ini akan mempertimbangkan pasangan kandidat user dan item , dengan menunjukkan jumlah tetangga yang terhubung dengan item . GCN menggunakan fungsi (misal, produk dalam) yang digunakan pada persamaan (4) untuk menghitung skor user dan relasi dilakukan:

|  |  |
| --- | --- |
|  | (4) |

Penjelasan:

Menggambarkan pentingnya hubungan dengan pengguna

melakukan embedding user dan hubungan berdasarkan dimensinya masing-masing. Pada persamaan (5) fungsi adalah skor hubungan-pengguna yang telah dinormalisasikan:

|  |  |
| --- | --- |
|  | (5) |

Skor hubungan-pengguna ini akan bertindak sebagai filter personalisasi saat menghitung representasi lingkungan entitas.

Pada Knowledge graph, jumlah nodes dapat bervariasi secara signifikan, sehingga untuk menjaga komputasi lebih efisien dan membantu model dalam meningkatkan akurasi maka dilakukan pengambilan sample tetangga menggunakan *importance sampling* seperti pada persamaan (6). *Importance sampling* (IS) merupakan teknik pengambilan sampel node yang memiliki bobot lebih, pada metode ini dilakukan sampling terhadap nodes yang dianggap penting berdasarkan distribusi sehingga informasi yang terkumpul dapat memenuhi ekspektasi Dimana nodes yang memiliki informasi penting diambil berdasarkan edges dan informasi yang dipropagasi . Sehingga estimasi dinyatakan sebagai [2]:

|  |  |
| --- | --- |
|  | (6) |

Penjelasan:

= Jumlah tetangga yang diambil sampelnya

= bobot dari node

= informasi yang terkait dengan node

= node tetangga yang diambil dari distribusi

Pada persamaan (7) langkah terakhir pada GCN layer adalah melakukan agregasi representasi entitas dan representasi tetangganya menjadi satu vektor [9]

|  |  |
| --- | --- |
|  | (7) |

Penjelasan:

Transformasi bobot

Transformasi bias

Nonliniear function

Tujuan fungsi *agregasi* ini dirancang untuk mengumpulkan dan menggabungkan informasi dari simpul tetangga sehingga memungkinkan model untuk memahami pola dan ketergantungan lokal dalam graf [10].

1. Hyperparameter Tuning

Hyperparameter tuning adalah proses dimana parameter model dilakukan penyesuaian atau disetel sedemikian rupa untuk meningkatkan kinerja model. Pada KGCN beberapa parameter yang disesuaikan diantaranya:

Tabel Hyperparameter Tuning

|  |  |
| --- | --- |
| *learning rate* | Mengontrol seberapa besar model menyesuaikan bobot selama proses pelatihan |
| *batch size* | Jumlah sampel yang diproses dalam satu iterasi pelatihan |
| *number of layer* | Jumlah lapisan konvolusional dalam model |
| *optimizer* | Algoritma yang digunakan untuk memperbarui bobot model |
| *Neighbor Aggregation Size* | Menentukan jumlah node tetangga yang di aggregasi disetiap layer |
| *Embedding Dimension* | Ukuran dimensi embedding dalam merepresentasikan isi node |
| *Iteration number* | Jumlah perulangan yang dilakukan model |

Metode hyperparameter yang akan digunakan adalah *GridSearchCV* dimana parameter akan ditentukan secara manual berdasarkan metrik yang sudah dipilih[11][12].

1. Output

Diberikan pasangan user dan film () dari hasil prediksi persamaan (3) yang menghasilkan skor probabilitas interaksi yang dilakukan oleh user dengan item.

1. Mean Absolute Error

Mean absolute error (MAE) merupakan standard metrics yang digunakan untuk evaluasi model, digambarkan pada permasaan (8):

|  |  |
| --- | --- |
|  | (8) |

Penjelasan:

jumlah n sample yang di terambil

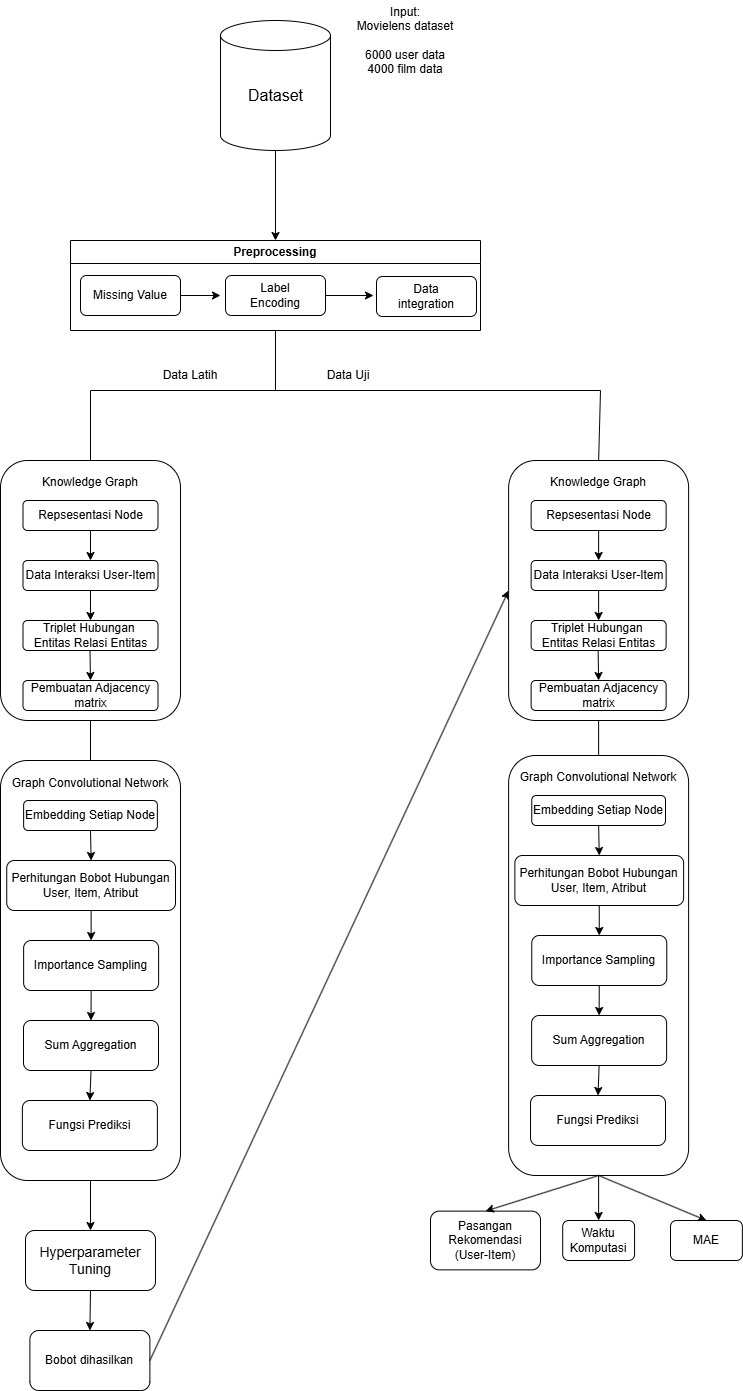
Jumlah n yang diprediksi sesuai[13]

Penelitian terdahulu [3] berfokus pada pengembangan metode sistem rekomendasi yang memanfaatkan *Knowledge Graph* (KG) melalui pendekatan yang disebut *Knowledge Graph Convolutional Networks* (KGCN). Pada penelitian ini mengalami permasalahan seperti dimensionalitas tinggi dan heterogenitas KG yang membuatnya sulit untuk diolah secara langsung dalam model pembelajaran mesin, untuk mengatasi masalah tersebut digunakan metode agregasi penjumlahan informasi dari tetangga dengan cara menjumlahkan informasi dari 2 entitas yang terhubung dan menggabungkannya dengan bias yang relevan sehingga menangkap baik informasi struktural maupun semantik KG. Metode ini menghasilkan AUC 0.978 pada dataset MovieLens.

Pada penelitian [14], pengaruh knowledge graph memberikan hasil AUC 0.982 dijelaskan melalui penerapan metode CUIKG (Combining User-end and Item-end Knowledge Graph learning). Metode ini secara bersamaan mempelajari embedding pengguna dan item dari knowledge graph di kedua sisi, yang memungkinkan model untuk menangkap relevansi antara pengguna dan item dengan lebih baik. Dengan memanfaatkan informasi semantik yang kaya dari knowledge graph, model ini dapat mengatasi masalah sparsity data dengan memperkenalkan lebih banyak hubungan semantik yang mendukung akurasi rekomendasi. Selain itu, model ini memperhitungkan atribut pengguna dan bagaimana atribut tersebut mempengaruhi preferensi mereka terhadap item, sehingga menciptakan representasi yang lebih akurat. Pengelolaan informasi tetangga juga dilakukan untuk menghindari agregasi informasi yang tidak perlu, yang dapat mengurangi noise dan menjaga kualitas representasi pengguna dan item.

Pada penelitian [15] berfokus pada penerapan *importance sampling* sebagai teknik sampling yang diimplementasikan pada model GCN. Metode ini membantu model FastGCN dalam mempercepat proses pelatihan dan juga meningkatkan akurasi dalam prediksi, dalam hasil perbandingan waktu pelatihan antara GCN menunjukkan bahwa FastGCN pada dataset Cora menghasilkan 0.0084 detik, Pudmed menghasilkan 0.0047 detik, dan reddit 0.0129 detik, sedangkan pada model GCN biasa yang menggunakan dataset cora menghasilkan 0.0166 detik. Pudmed 0.0815 detik, dan pada dataset reddit menghasilkan 2.1731 detik. Dari hasil tersebut menunjukkan bahwa model FastGCN lebih cepat dalam waktu komputasi ketika masa pelatihan dibandingkan dengan model GCN tradisional.

7. Metode Penelitian



Gambar Metode Penelitian

7.1. Dataset

Pada penelitian ini, terdapat 3 file dataset diperoleh dari website grouplens yang berisi rating film yang telah diberikan oleh user. Dataset ini terdiri dari 6000 users dan 4000 film, dimana pada file rating.dat berisikan UserID, MovieID, Rating, Timestamp kemudian file users.dat berisikan UserID, Gender, Age, Occupation, Zip-Code, sedangkan file movies.dat berisikan MovieID, Title, Genres.

Tabel Film Atribut

|  |  |  |
| --- | --- | --- |
| No | Atribut | Deskripsi |
| 1 | UserID | ID yang mewakili user |
| 2 | Gender | Merupakan atribut user yang menjelaskan jenis kelamin user |
| 3 | Age | Merupakan atribut user yang berisikan kategori umur user |
| 4 | Occupation | Merupakan atribut user yang berisikan kategori pekerjaan user |
| 5 | Zip-Code | Merupakan atribut user yang berisikan lokasi user menonton |

Tabel Atribut Film

|  |  |  |
| --- | --- | --- |
| No | Atribut | Deskripsi |
| 1 | MovieID | ID yang mewakili film tertentu |
| 2 | Title | Merupakan Atribut judul film |
| 3 | Genres | Merupakan atribut film yang mengkategorikan genre film |

Tabel Rating Film

|  |  |  |
| --- | --- | --- |
| No | Atribut | Deskripsi |
| 1 | UserID | ID yang mewakili user tertentu |
| 2 | MovieID | ID yang mewakili film tertentu |
| 3 | Ratings | Penilaian yang diberikan user terhadap film (1-5 bintang) |
| 4 | Timestamp | Tanggal user melakukan rating |

7.2. **Preprocessing Data**

Preprocessing data merupakan tahap untuk membersihkan data agar data tidak mengurangi akurasi model sehingga data mentah bisa dipakai saat dianalisis, dilakukan beberapa tahap preprocessing:

7.2.1. Missing Values

Dataset akan dilakukan pengecheckan missing values terlebih dahulu di semua atribut, karena jika ditemukan missing values akan mempengaruhi kemampuan model dalam memahami struktur graph.

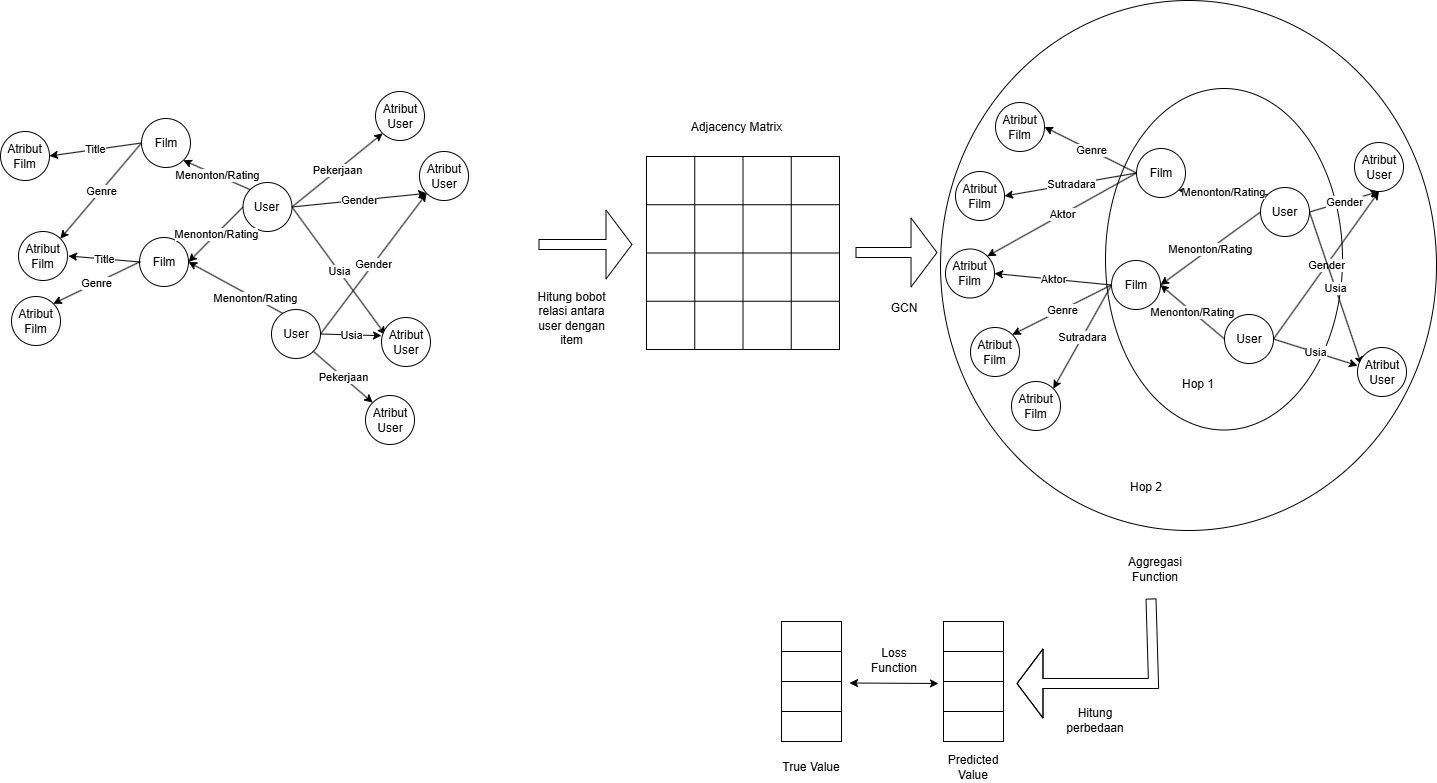
7.2.2. Label Encoding

Dataset juga akan dilakukan transformasi pada atribut berbentuk kategorikal menggunakan label encoding agar meningkatkan kinerja model dalam belajar pola

7.2.3. Data Integration

Dataset akan digabung menjadi satu berdasarkan nomor MovieID, dan userID agar didapatkan informasi yang lebih jelas mengenai bagaimana user berinteraksi dengan film tersebut.

7.3. Knowledge Graph



Gambar Arsitektur Knowledge Graph

Dari dataset yang digabung dimiliki kumpulan pengguna dan kumpulan film , kemudian dibuat representasi nodes yang menggambarkan entitas apa saja dan interaksi yang terjadi, seperti:

Entitas:

* UserID
* Gender
* Age
* Occupation
* Zip-Code
* MovieID
* Title
* Genre
* Rating
* Timestamp

Interaksi:

* memberikan rating
* memiliki umur
* bekerja sebagai
* memiliki genre
* memiliki title

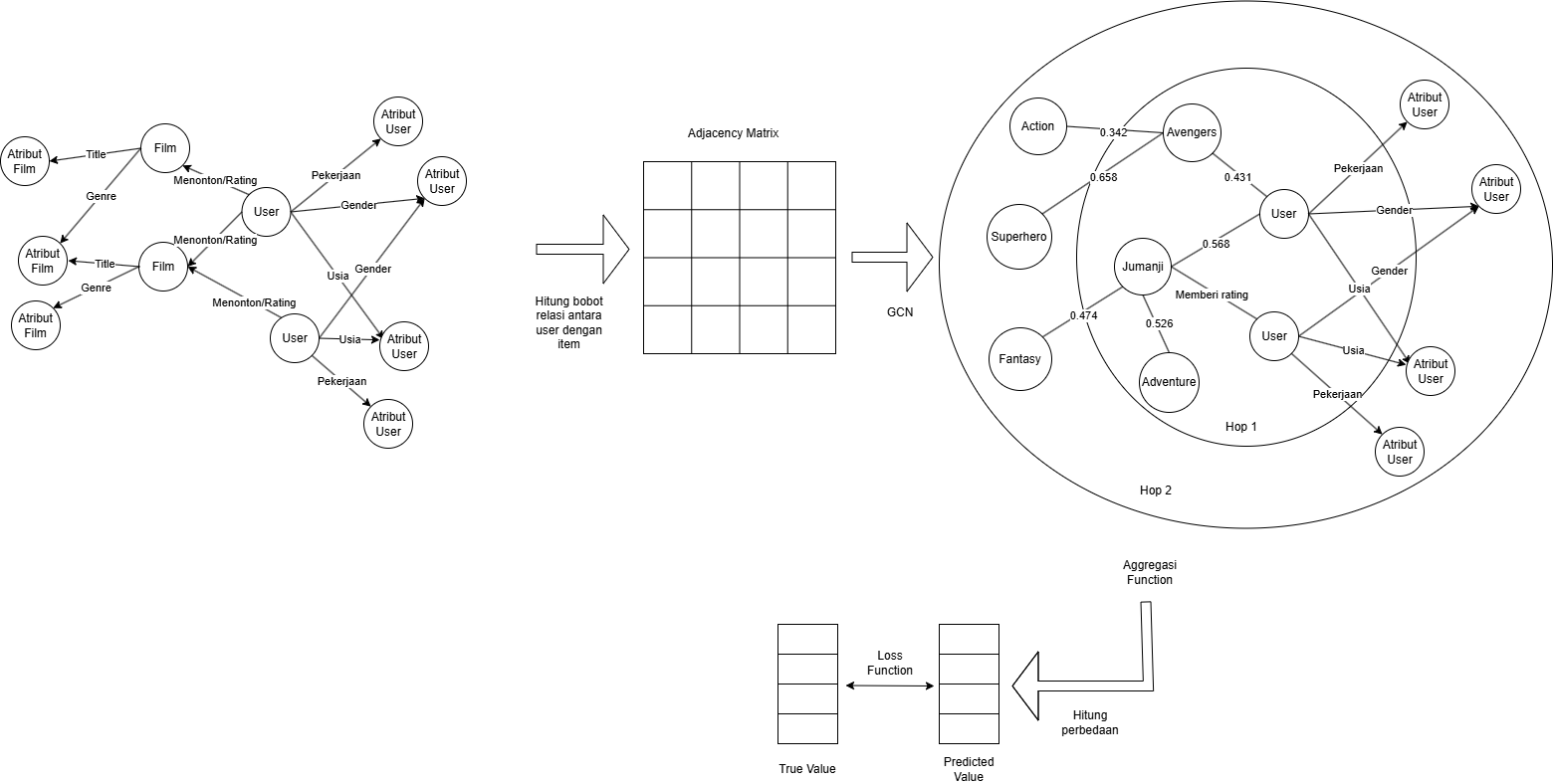
Setelah representasi nodes telah dibuat selanjutnya dibuat matrix hubungan user-item menggunakan persamaan (1) yang berarti:

dimana matrix hubungan user-item jika user telah memberikan rating film dan jika user belum pernah memberikan rating film tersebut. Kemudian dibuat knowledge graph untuk menambahkan informasi dari atribut user dan atribut item dengan aturan triple hubungan entitas-relasi-entitas menggunakan persamaan (2) dimana h, r, t menunjukkan kepala, relasi, dan ekor seperti:

Tabel Triplet Hubungan

|  |  |  |
| --- | --- | --- |
| Head (h) | Relation (r) | Tail (t) |
| UserID | memberi rating | MovieID |
| UserID | memiliki umur | 45 |
| UserID | bekerja sebagai | writer |
| MovieID | memiliki genre | Fantasy |
| MovieID | memiliki genre | Adventure |
| MovieID | memiliki judul | Jumanji |

7.4. Graph Convolutional Network



Gambar Arsitektur Graph Convolutional Network

Setelah terbentuk sebuah triplet hubungan, selanjutnya GCN menerima input berupa *adjacency matrix* untuk menunjukkan hubungan antar user, item dan atribut yang dimiliki, kemudian diberi bobot hubungan antara user-item dan relasinya menggunakan persamaan (4), misalkan, “user memberi rating 5 pada film berjudul Jumanji dengan genre fantasy”, maka diberikan embedding awal secara acak:

User1 = [0.3, 0.5, 0.7],

Jumanji = [0.4, 0.6, 0.8]

Fantasy = [0.3, 0.7, 0.5]

Adventure = [0.5, 0.2, 0.9]

Avengers = [0.5, 0.2, 0.9]

Superhero = [0.6, 0.4, 0.3]

Action = [0.2, 0.8, 0.5]

Menggunakan dot produk untuk menentukan bobot hubungan user – film:

Setelah didapat bobot hubungan, selanjutnya normalisasi pada bobot hubungan menggunakan persamaan (5) maka:

Film Jumanji memberikan kontribusi tinggi pada embedding user dibandingkan dengan avengers, tetapi pada genre yang memiliki bobot tertinggi yang akan lebih berkontribusi pada embedding user. Kemudian implementasi teknik *importance sampling* digunakan dengan menentukan informasi yang diambil berdasarkan bobot relasi tertinggi, dimisalkan bahwa *importance sampling* pada persamaan (6) hanya mengambil 2 node dengan bobot tertinggi, terdapat pada node superhero (0.658) dan adventure (0.526) maka 2 node ini akan dilakukan persamaan (7), dimana:

Sehingga menghasilkan embedding baru untuk user berdasarkan genre yang relevan dengan user. Kemudian dilakukan skor prediksi menggunakan persamaan (3) untuk film baru, misalkan terdapat film *Toy Story* yang memiliki genre *Cartoon*, *Adventure* sehingga:

Misalkan ToyStory memiliki embedding

ToyStory = [0.10, 0.6, 0.25]

Karena nilai < 0.8 maka film Toy Story tidak akan direkomendasikan kepada user

8. Jadwal Penelitian

Tabel Jadwal Penelitian

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Jadwal Kegiatan | Bulan 1 | | | | Bulan 2 | | | | Bulan 3 | | | | Bulan 4 | | | |
|  | Minggu ke- | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1. | Pengambilan Dataset |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. | Preprocessing Data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. | Pembangunan dan pelatihan model KGCN |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | Pengujian Model KGCN |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5. | Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6. | Pembuatan Laporan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

9. Referensi

[1] Q. Guo *et al.*, “A Survey on Knowledge Graph-Based Recommender Systems,” *IEEE Trans Knowl Data Eng*, vol. 34, no. 8, pp. 3549–3568, Aug. 2022, doi: 10.1109/TKDE.2020.3028705.

[2] Y. Ji *et al.*, “Accelerating Large-Scale Heterogeneous Interaction Graph Embedding Learning via Importance Sampling,” *ACM Trans Knowl Discov Data*, vol. 15, no. 1, Jan. 2021, doi: 10.1145/3418684.

[3] H. Wang, M. Zhao, X. Xie, W. Li, and M. Guo, “Knowledge graph convolutional networks for recommender systems,” in *The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW 2019*, Association for Computing Machinery, Inc, May 2019, pp. 3307–3313. doi: 10.1145/3308558.3313417.

[4] Y. Afoudi, M. Lazaar, and M. Al Achhab, “Hybrid recommendation system combined content-based filtering and collaborative prediction using artificial neural network,” *Simul Model Pract Theory*, vol. 113, Dec. 2021, doi: 10.1016/j.simpat.2021.102375.

[5] J. A. P. Sacenti, R. Fileto, and R. Willrich, “Knowledge graph summarization impacts on movie recommendations,” *J Intell Inf Syst*, vol. 58, no. 1, pp. 43–66, Feb. 2022, doi: 10.1007/s10844-021-00650-z.

[6] X. Wang *et al.*, “Learning intents behind interactions with knowledge graph for recommendation,” in *The Web Conference 2021 - Proceedings of the World Wide Web Conference, WWW 2021*, Association for Computing Machinery, Inc, Apr. 2021, pp. 878–887. doi: 10.1145/3442381.3450133.

[7] C. Li, Y. Cao, Y. Zhu, D. Cheng, C. Li, and Y. Morimoto, “Ripple Knowledge Graph Convolutional Networks for Recommendation Systems,” *Machine Intelligence Research*, vol. 21, no. 3, pp. 481–494, Jun. 2024, doi: 10.1007/s11633-023-1440-x.

[8] H. Xia, K. Huang, and Y. Liu, “Unexpected interest recommender system with graph neural network,” *Complex and Intelligent Systems*, vol. 9, no. 4, pp. 3819–3833, Aug. 2023, doi: 10.1007/s40747-022-00849-9.

[9] Y. Wang, X. F. Ma, and M. Zhu, “A knowledge graph algorithm enabled deep recommendation system,” *PeerJ Comput Sci*, vol. 10, 2024, doi: 10.7717/PEERJ-CS.2010.

[10] U. A. Bhatti, H. Tang, G. Wu, S. Marjan, and A. Hussain, “Deep Learning with Graph Convolutional Networks: An Overview and Latest Applications in Computational Intelligence,” 2023, *Wiley-Hindawi*. doi: 10.1155/2023/8342104.

[11] S. Nur Himawa and R. Sohiburoyyan, “Hyperparameter Tuning on Graph Neural Network for the Classification of SARS-CoV-2 Inhibitors,” 2023. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC

[12] X. Zeng, S. Wang, Y. Zhu, M. Xu, and Z. Zou, “A Knowledge Graph Convolutional Networks Method for Countryside Ecological Patterns Recommendation by Mining Geographical Features,” *ISPRS Int J Geoinf*, vol. 11, no. 12, Dec. 2022, doi: 10.3390/ijgi11120625.

[13] T. O. Hodson, “Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not,” Jul. 19, 2022, *Copernicus GmbH*. doi: 10.5194/gmd-15-5481-2022.

[14] T. Gu, H. Liang, C. Bin, and L. Chang, “Combining user-end and item-end knowledge graph learning for personalized recommendation,” *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 5, pp. 9213–9225, 2021, doi: 10.3233/JIFS-201635.

[15] J. Chen, T. Ma, and C. Xiao, “FastGCN: Fast Learning with Graph Convolutional Networks via Importance Sampling,” Jan. 2018, [Online]. Available: http://arxiv.org/abs/1801.10247