JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

Sistem Rekomendasi Film Menggunakan Data User-End dan Knowledge Graph Convolutional Network pada Dataset MovieLens 1M

Muhammad Rizki Yanuar [*], Fajri Rakhmat Umbara, Agus Komarudin

Sains dan Informatika, Informatika, Universitas Jenderal Achmad Yani, Cimahi, Indonesia
Email: [1*] mrizkiyanuar21@if.unjani.ac.id, [2]fajri.rakhmat@lecture.unjani.ac.id, [3]agus.komarudin@lecture.unjani.ac.id
Email Penulis Korespondensi: mrizkiyanuar21@if.unjani.ac.id
Submitted **18-06-2025;** Accepted **01-08-2025;** Published **30-08-2025**

Abstrak

Sistem rekomendasi tradisional Collaborative Filtering dan Content-Based Filtering seringkali gagal memberikan rekomendasi yang relevan karena memiliki keterbatasan dalam menangani masalah sparsity dan cold-start.. Penelitian ini mengusulkan model Knowledge Graph Convolutional Network (KGCN) yang diperkaya dengan data demografis pengguna dari dataset MovieLens 1M untuk mengatasi masalah tersebut. Fokus utama penelitian adalah membuktikan bahwa teknik Importance Sampling secara signifikan lebih unggul dibandingkan Uniform Sampling dalam melatih model secara efektif. Setelah proses hyperparameter tuning, konfigurasi model terbaik berhasil mencapai performa puncak dengan skor AUC sebesar 0.8798 dan NDCG@10 sebesar 0.9719. Hasil ini membuktikan bahwa pendekatan yang diusulkan efektif dalam membangun sistem rekomendasi yang akurat, personal, dan mampu menangani masalah sparsity serta cold-start.

Kata Kunci: Knowledge Graph; Graph Convolutional Network; Sistem Rekomendasi Film; Importance Sampling; Sparsity;

Abstract

Traditional recommendation systems such as Collaborative Filtering and Content-Based Filtering often fail to provide relevant recommendations due to their limitations in handling sparsity and cold-start problems. This study proposes a Knowledge Graph Convolutional Network (KGCN) model enriched with user demographic data from the MovieLens 1M dataset to address these issues. The primary focus of the research is to demonstrate that the Importance Sampling technique is significantly superior to Uniform Sampling in effectively training the model. After hyperparameter tuning, the optimal model configuration achieved peak performance with an AUC score of 0.8798 and NDCG@10 of 0.9719. These results demonstrate that the proposed approach is effective in building an accurate, personalised recommendation system capable of addressing sparsity and cold-start issues.

Keywords: Knowledge Graph; Graph Convolutional Network; Sistem Rekomendasi Film; Importance Sampling; Sparsity;

1. PENDAHULUAN

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].

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.

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

Pada penelitian [4], 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.

Pada penelitian [5], diperkenalkan metode importance sampling sebagai solusi untuk mengurangi kompleksitas komputasi dalam training GCN. Berdasarkan eksperimen pada dataset seperti Pubmed dan Cora, importance sampling menunjukkan hasil yang lebih baik dibandingkan sampling uniform, baik dari segi akurasi maupun waktu pelatihan. Sebagai contoh, pada Pubmed, F1-score meningkat dari 0.721 menjadi 0.776, dan pada Cora dari 0.723 menjadi 0.818. Evaluasi menunjukkan efektivitas metode ini dalam meningkatkan kinerja model GCN.

Penelitian oleh [6] memperkenalkan model Knowledge Graph-based Intent Network (KGIN) untuk sistem rekomendasi berbasis knowledge graph. Model ini mengatasi keterbatasan metode GNN sebelumnya yang hanya memodelkan relasi pengguna-item secara kasar dan belum mampu menangkap intent pengguna secara detail maupun memanfaatkan dependensi relasi dalam knowledge graph. KGIN terdiri dari dua komponen utama: pemodelan intent pengguna, yang merepresentasikan setiap intent sebagai kombinasi relasi pada knowledge graph dengan mekanisme attention dan regularisasi agar setiap intent saling independen; serta agregasi informasi berbasis jalur relasi (relational path-aware aggregation), yang memungkinkan model menangkap semantik dan dependensi relasi jarak jauh antar entitas.

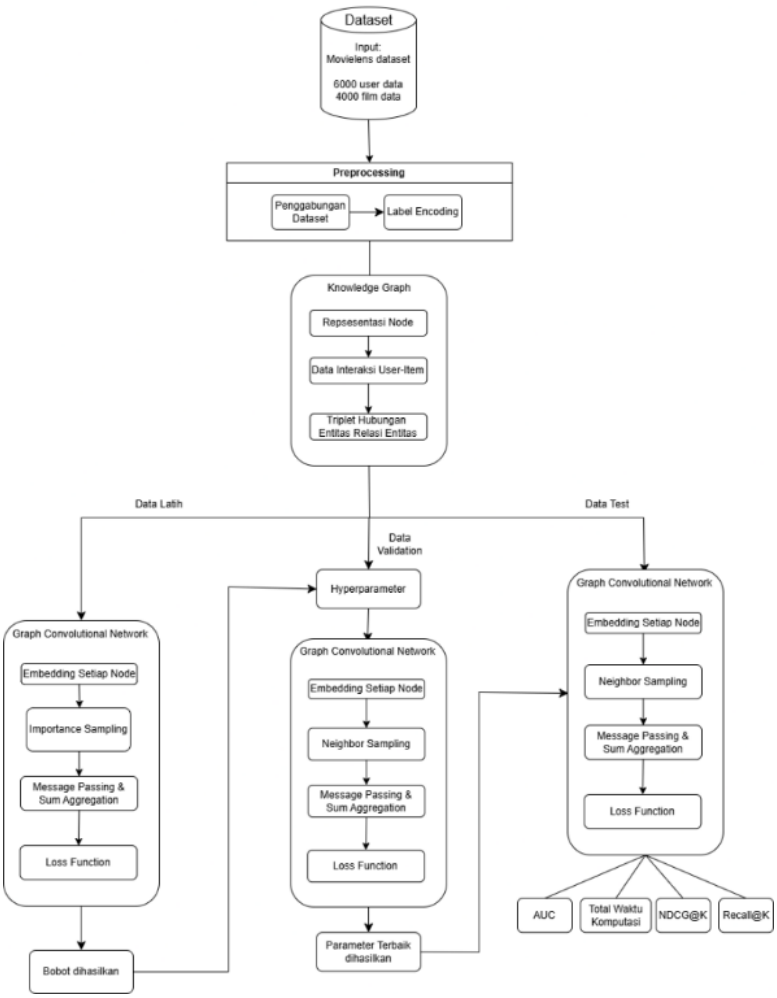
Penelitian oleh [7] menunjukkan fleksibilitas pendekatan KGCN dengan menerapkannya pada domain unik, yaitu rekomendasi pola ekologi pedesaan. Dengan membangun knowledge graph berbasis fitur geografis, mereka berhasil mengatasi masalah sparsity dan cold-start, yang membuktikan bahwa arsitektur KGCN efektif untuk menangkap hubungan kompleks bahkan di luar domain rekomendasi item tradisional.

Beberapa penelitian sebelumnya telah menerapkan KGCN untuk sistem rekomendasi, namun penggunaan data *user-end* dan teknik *importance sampling* belum banyak diterapkan pada sistem rekomendasi. Mayoritas penelitian yang menggunakan KGCN cenderung menggunakan data interaksi antara pengguna-item, dan hanya menerapkan teknik *uniform sampling*. 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 teknik *uniform sampling* mengukur peningkatan dalam akurasi, relevansi rekomendasi, dan efisiensi waktu komputasi.

2. METODOLOGI PENELITIAN

Penelitian dilakukan dengan pendekatan kuantitatif, yaitu pendekatan dalam menguji teknik *importance sampling* berbasis bobot *edge* pada model *Graph Convolutional Network* (GCN). *Importance sampling* dilakukan dengan memilih *edge* yang memiliki bobot (edge_weight) lebih besar dari threshold tertentu, kemudian sampling edge dilakukan secara probabilistik proporsional terhadap bobot tersebut. Teknik ini bertujuan untuk memfokuskan pelatihan pada hubungan yang lebih penting antara pengguna dan item, sehingga meningkatkan efisiensi dan akurasi model. Evaluasi model dilakukan menggunakan metrik evaluasi seperti Precision@K, Recall@K, NDCG@K, dan AUC untuk mengukur kualitas rekomendasi dan mengurangi waktu komputasi selama training. Untuk memberikan menyeluruh, keseluruhan proses divisualisasikan dalam diagram alur pada Gambar 1.

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom



Gambar 1. Metode Penelitian

2.1 Tahapan Penelitian

Pada gambar 1 dijelaskan tahap-tahap penelitian ini menggunakan kajian yang mengadopsi metode penelitian kuantitatif. Penelitian kuantitatif adalah penelitian yang menggunakan angka-angka mulai dari pengumpulan data, *preprocessing data* hingga hasil atau penarikan kesimpulan. Penelitian ini dilakukan melalui beberapa tahapan yang terstruktur dan sistematis untuk memastikan setiap langkah yang dilakukan akan menghasilkan data dan hasil yang akurat dan dapat digunakan. Adapun penjelasan pada Gambar 1 mengenai metode penelitian yang digunakan sebagai berikut:

a. Dataset

Pengambilan data diambil dari website *grouplens*, dimana data yang diambil memiliki 100.000 data rating. Dataset terbagi menjadi tiga bagian:

1. Data pengguna
2. Data film
3. Data rating

b. Preprocessing Data

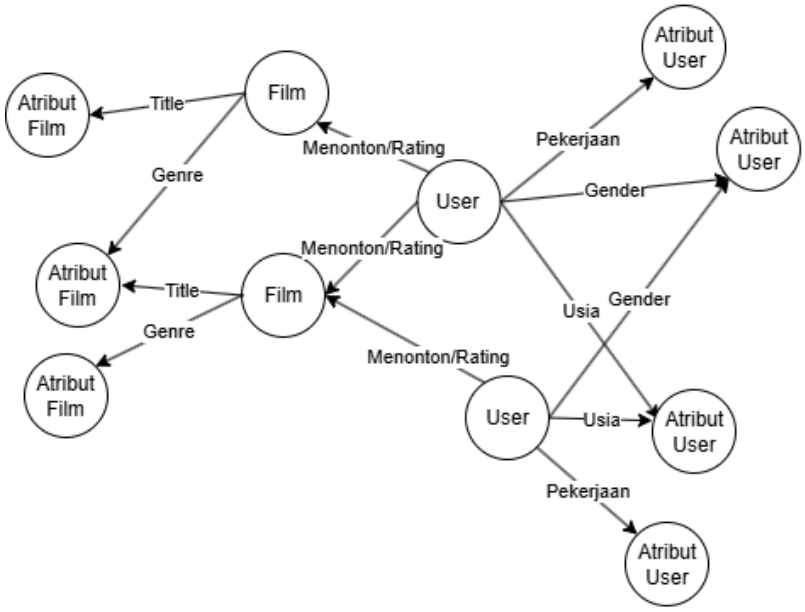
Preprocessing data meruapakan tahap membersihkan data agar data tidak mengurangi akurasi model sehingga data mentah bisa dipakai saat dianalisis, tahapan preprocessing yang dilakukan sebagai berikut:

1. Penggabungan Dataset
   Pada tahap ini, beberapa dataset digabungkan menjadi satu kesatuan berdasarkan kesamaan MovieID dan UserID. Proses ini bertujuan untuk memperoleh informasi yang lebih lengkap mengenai interaksi pengguna terhadap film, seperti rating yang diberikan serta genre film yang disukai. Dengan melakukan penggabungan ini, analisis dapat mengungkap preferensi pengguna berdasarkan atribut-atribut tertentu (seperti gender, usia, atau pekerjaan), sehingga dapat diketahui pola kecenderungan pengguna dengan karakteristik tertentu terhadap genre film tertentu. Hasil penggabungan ini menjadi fondasi penting dalam membangun sistem rekomendasi yang personal dan relevan.
2. Label Encoding
   Tahap ini dilakukan proses transformasi terhadap data kategorik menjadi data numerik unik yang memiliki makna tanpa memperhatikan urutan atau tingkatan [8] [9]. Proses ini penting karena sebagian model algoritma machine learning dan deep learning tidak dapat langsung menerima input dalam bentuk teks atau kategori.

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

c. Knowledge Graph

Setelah melalui tahap preprocessing, langkah selanjutnya adalah membangun Knowledge Graph (KG) yang digunakan untuk merepresentasikan hubungan kompleks antara user dan item, serta atribut-atribut tambahan dari masing-masing entitas seperti pada Gambar III.2.



Gambar 2. Knowledge Graph

Knowledge Graph (KG) meruapakan struktur data yang menyimpan informasi dalam bentuk entitas serta hubungan antar entitas [10]. Setiap node dalam KG merepresentasikan entitas, sedangkan edge menggambarkan jenis hubungan antara dua entitas. Dalam sistem rekomendasi, KG digunakan untuk mengumpulkan informasi tambahan dan konektivitas menjadi pengetahuan yang tidak hanya bergantung pada interaksi pengguna-item sehingga informasi menjadi lebih mudah dipahami sehingga membentuk informasi yang penuh makna [1]. Pada dasarnya struktur KG membentuk struktur jaringan heterogen yang terdiri dari atas *node* dan *edge,* dalam konteks sistem rekomendasi *node* mencakup entitas seperti pengguna, film, *genre*, *age*, *occupation*, dan *gender*. Sementara *edge* menggambarkan hubungan antar entitas, seperti *hasGenre* (memiliki genre), *hasOccupation* (memiliki pekerjaan) atau *rates* (memberi rating), dan sebuah *knowledge graph* umumnya direpresentasikan dalam bentuk *(h, r, t)* yang terdiri dari *head, relation, tail* [11]. Beberapa langkah dalam membuat struktur KG:

1. Representasi Node
   Representasi node dilakukan untuk mendapatkan informasi penting seperti atribut pengguna dan atribut film, untuk mendapatkan representasi yang lebih baik suatu individu [12].
2. Data Interaksi Pengguna-Item
   Interaksi eksplisit antara user dan film direpresentasikan dalam bentuk matrix interaksi seperti persamaan (1) interaksi user-item didefinisikan sebagai:

$$Y \in \mathbb{R}^{MxN} \quad (1)$$

   Penjelasan:
   $\boldsymbol{Y}$: Interaksi antara pengguna dan item
   $\mathbb{R}$: Himpunan bilangan real dalam matrix $\boldsymbol{Y}$
   $\boldsymbol{M}$: Jumlah baris yang mewakili jumlah pengguna dalam matrix
   N : Jumlah kolom item yang mewakili jumlah item dalam matrix
3. Pembentukan Triplet
   Hubungan antar entitas dalam KG disusun dalam bentuk triplet menggunakan persamaan (2), secara sistematis triplet dinyatakan sebagai berikut[13]:

$$\mathcal{G} = \{(h, r, t) \mid h \in e, t \in e, r \in R \quad (2)$$

   Penjelasan:
   $\mathcal{G}$: Sekumpulan triple yang menyimpan informasi hubungan antar entitas
   $(\boldsymbol{h}, \boldsymbol{r}, \boldsymbol{t})$: kepala, relasi dan ekor dari sebuah tripl7 pengetahuan
   e : Himpunan entitas dalam knowledge graph
   R : Himpunan relasi dalam knowledge graph

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

Triplet-triplet ini mencerminkan hubungan semantik dan struktural yang akan digunakan sebagai dasar dalam pembuatan edge pada graph.

d. Graph Convolutional Network

Graph Convolutional Network (GCN) adalah jenis jaringan saraf yang dirancang untuk bekerja dengan data yang terstruktur dalam bentuk graf, memungkinkan menangkap informasi dari *node* tetangga melalui mekanisme propagasi informasi[14]. GCN digunakan untuk melakukan propagasi informasi antar *node* dan menghasilkan representasi (embedding) yang merepresentasikan relasi kompleks antara pengguna, item, dan atribut. Tahapan dalam GCN meliputi:

1. Embedding awal
   Pada tahap awal proses pelatihan, setiap node dalam graph, seperti pengguna, *gender*, *age*, *occupation,* film, genre, diberikan vektor embedding secara acak. Pemberian embedding awal secara acak ini bertujuan sebagai titik awal bagi model untuk mempelajari informasi yang bermakna.
2. Importance Sampling
   Kemudian dilakukan sampling menggunakan *importance sampling* untuk setiap node pengguna dengan memilih tetangga paling relevan berdasarkan bobot relasinya melalui persamaan (3):

$$\hat{\mu}_q = \frac{1}{n}\sum_j \frac{p(\hat{v}_j)}{q(\hat{v}_j)} f(\hat{v}_j) \quad (3)$$

   Penjelasan:
   n : Jumlah tetangga yang diambil sampelnya
   $p(\hat{v}_j)$: bobot dari *node* atau *edge* $\hat{v}_j$
   $f(\hat{v}_j)$: informasi node $\hat{v}_j$ yang akan dipropagasi
   $\hat{v}_j$ = node tetangga yang diambil dari distribusi q
   Sampling dilakukan pada *2-layer* dimana masing-masing *layer* diambil 5 nodes tetangga berdasarkan skor hubungan yang dihitung menggunakan *dot product* antar embedding dan bobot menggunakan persamaan (4):

$$\pi_r^u = g(u, r) \quad (4)$$

   Penjelasan:
   π_r^u : Menggambarkan pentingnya hubungan r dengan pengguna u
   $g(u,r)$: Fungsi interaksi antara embedding pengguna dan embedding relasi
   Setelah didapat bobot hubungan, selanjutnya dilakukan normalisasi pada bobot hubungan menggunakan persamaan (5):

$$\tilde{\pi}_{r_{v,e}}^u = \frac{\exp(\pi_{r_{v,e}}^u)}{\sum e \in N(v)^{\exp(\pi_{r_{v,e}}^u)}} \quad (5)$$

   Penjelasan:
   $\pi_{r_{v,e}}^u$: Skor atau bobot hubungan awal antara *node* v dan tetangga e berdasarkan relasi dan konteks user
   $N(v)$: Himpunan tetangga dari *node* v
   $e \in N(v)$: Setiap edge (hubungan) antara *node* pusat dan tetangga
   exp : Fungsi eksponensial yang digunakan untuk memastikan semua nilai menjadi positif dan memperkuat perbedaan relatif antar skor
3. Message Passing & Sum Aggregation
   Embedding dari node tetangga yang telah disampling dipropagasikan ke node pusat dengan bobot hasil normalisasi, sehingga akan menghasilkan embedding baru untuk node user yang dihasilkan dari penjumlahan informasi yang terhubung ke user langsung, melalui persamaan (6).

$$agg_{sum} = \sigma(W.(v + v_{S(v)}^u) + b \quad (6)$$

   Penjelasan:
   W : Transformasi bobot
   v : Embedding *node* pusat
   b : Transformasi bias
   σ : Nonliniear function
   Proses ini dikenal sebagai message passing dengan agregasi sum (penjumlahan), di mana setiap node mengumpulkan informasi dari tetangganya dengan menjumlahkan embedding tetangga yang sudah ditimbang.
4. Prediksi
   Kemudian dilakukan skor prediksi model memanfaatkan interaksi pengguna-item yang direpresentasikan dalam bentuk matriks Y serta struktur KG $\mathcal{G}$. Prediksi probabilitas interaksi yang dilakukan oleh pengguna u dengan item v dihitung pada persamaan (7)[15]:

$$\hat{y}_{uv} = \mathcal{F}(u, v|\Theta, Y, \mathcal{G}) \quad (7)$$

   Penjelasan:
   $\hat{y}_{uv}$: Nilai prediksi probabilitas pengguna u akan berinteraksi dengan item v
   $\mathcal{F}$: Fungsi untuk memprediksi nilai $\hat{y}_{uv}$

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

Θ : Parameter model mengoptimalkan fungsi prediksi
Y : Matriks interaksi pengguna-item
$\mathcal{G}$: *Knowledge Graph* memeberikan konteks tambahan tentang hubungan antar entitas

e. Metrik Evaluasi

Metrik evaluasi adalah alat untuk mengukur kinerja model pada berbagai tugas seperti klasifikasi biner, multi-kelas, regresi, segmentasi gambar[16]. Berikut metrik evaluasi yang digunakan pada sistem rekomendasi:

1. Recall@K
Merupakan sebuah evaluasi untuk rekomendasi yang diberikan model. Pada persamaan (8) Recall@K merepresentasikan berapa banyak item di top_K yang termasuk kedalam *ground_truth_item* [11].

$$Rec@K = \frac{ground\_truth\_items \cap top\_k}{ground\_truth\_items} \quad (8)$$

2. Precision@K
Menunjukkan jumlah item K di *ground_truth_item* yang termasuk kedalam top_K dan relevan untuk direkomendasikan kepada pengguna [4], seperti pada persamaan (9).

$$Pre@K = \frac{ground\_truth\_items \cap top\_k}{top\_k} \quad (9)$$

3. Normalized Discounted Cumulative Gain (NDCG@K)
Digunakan untuk merepresentasikan kualitas peringkat rekomendasi yang dihasilkan, semakin mendekati nilai 1 mengindikasikan kualitas peringkat rekomendasi yang sangat baik [10], seperti pada persamaan (10).

$$NDCG(u) = \frac{DCG@K}{IDCG@K} \quad (10)$$

Dimana didapat dari persamaan (11) untuk menghitung *Discounted Cumulative Gain* (DCG) dalam memperhitungkan posisi item, jika nilai DCG tinggi menunjukkan lebih banyak item dengan peringkat tinggi [17]dan *Ideal Discounted Cumulative Gain* (IDCG) yang dijadikan sebagai perbandingan[18], seperti pada persamaan (12).

$$DCG(u) = \sum_{j=1}^{n} \frac{2^{rel_i} - 1}{log_2(i+1)} \quad (11)$$

$$IDCG(u) = \sum_{i=1}^{|rel_k|} \frac{2^{rel_i - 1}}{log_2(i+1)} \quad (12)$$

Penjelasan:
K = Peringkat maksimum
i = posisi item dalam peringkat
rel_i = nilai relevansi item pada posisi i

4. AUC (Area Under Curve)
Metric yang digunakan untuk mengevaluasi kemampuan model dalam melakukan klasifikasi, nilai yang lebih tinggi menunjukkan kinerja model yang lebih baik [19], seperti pada persamaan (13)

$$AUC = \frac{1}{|R|N|} \sum_{i \in R} \sum_{j \in N} \delta(s_i < s_j) \quad (13)$$

Penjelasan:
R = Himpunan item relevan rating ≥ 4
N = Himpunan item tidak relevan
s_i = Skor prediksi model untuk item i
s_j = Skor prediksi model untuk item j

3. HASIL DAN PEMBAHASAN

3.1 Dataset

Pada penelitian ini, data bersumber dari dataset publik MovieLens 1M yang terdiri dari 6040 pengguna unik, 3883 film, dan 1.000.209 data rating, dataset ini juga pernah digunakan pada penelitian [20]. Untuk memahami karakteristik pengguna, penelitian ini memanfaatkan data demografis yang rinciannya dapat dilihat pada Tabel 1.

a. Dataset Pengguna:

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

Tabel 1 Data Pengguna

UserID	Gender	Age	Occupation
1	M	1	3
2	F	25	2
3	M	2	2

Selain data pengguna, penelitian ini juga menggunakan data film yang mencakup atribut seperti ID unik, judul, dan genre untuk membangun *knowledge graph*. Atribut-atribut ini disajikan secara rinci pada Tabel 2.

b. Dataset film:

Tabel 2. Data Film

MovieID	Title	Genres
1	Toy Story (1995)	Animation|Children's|Comedy
2	Jumanji (1995)	Adventure|Childen's|Fantasy
3	Grumpier Old Men (1995)	Comedy|Romance

Kemudian digunakan juga data rating yang menggambarkan preferensi pengguna terhadap film, interaksi digambarkan dengan rating yang diberikan oleh pengguna terhadap film yang telah ditonton. Struktur dari data rating disajikan pada Tabel 3.

c. Dataset rating:

Tabel 3. Data Rating

UserID	MovieID	Ratings
1	1	4
2	2	3
3	3	1

3.2 Preprocessing Data

a. Penggabungan Dataset

Pada tahap ini, beberapa dataset digabungkan menjadi satu kesatuan berdasarkan kesamaan MovieID dan UserID. Proses ini bertujuan untuk memperoleh informasi yang lebih lengkap mengenai interaksi pengguna terhadap film, seperti rating yang diberikan serta genre film yang disukai. Dengan melakukan penggabungan ini, analisis dapat mengungkap preferensi pengguna berdasarkan atribut-atribut tertentu (seperti gender, usia, atau pekerjaan), sehingga dapat diketahui pola kecenderungan pengguna dengan karakteristik tertentu terhadap genre film tertentu. Berikut adalah data setelah dilakukan penggabungan dataset seperti pada Gambar 3.

UserID	Gender	Age	Occupation	MovieID	Rating	Title	Genres
1	F	1	10	1193	5	One Flew Over the Cuckoo's Nest (1975)	Drama
1	F	1	10	661	3	James and the Giant Peach (1996)	Animation|Children's|Musical
1	F	1	10	914	3	My Fair Lady (1964)	Musical|Romance
1	F	1	10	3408	4	Erin Brockovich (2000)	Drama
1	F	1	10	2355	5	Bug's Life, A (1998)	Animation|Children's|Comedy

Gambar 3. Hasil Penggabungan Dataset

Dari penggabungan dataset tersebut didapatkan informasi lebih jelas mengenai atribut user dan film dengan genre apa yang disukai oleh user tersebut berdasarkan nilai rating yang diberikan.

b. Label Encoding

Label encoding untuk mengubah setiap data kategorik menjadi data numerik unik yang memiliki makna tanpa memperhatikan urutan atau tingkatan [8][9], misalnya pada kolom gender terdapat 2 kategori yaitu, M dan F, melalui label encoding, kategori tersebut akan di-*transform* menjadi 0 dan 1. Berikut hasil setelah dilakukan label encoding seperti pada Gambar 4.

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

UserID	Gender	Age	Occupation	MovieID	Rating	Title	Genres
1	0	1	10	1193	5	One Flew Over the Cuckoo's Nest (1975)	Drama
1	0	1	10	661	3	James and the Giant Peach (1996)	Animation|Children's|Musical
1	0	1	10	914	3	My Fair Lady (1964)	Musical|Romance
1	0	1	10	3408	4	Erin Brockovich (2000)	Drama
1	0	1	10	2355	5	Bug's Life, A (1998)	Animation|Children's|Comedy
...	...	...	...	...	...	...	...
6040	1	25	6	1091	1	Weekend at Bernie's (1989)	Comedy
6040	1	25	6	1094	5	Crying Game, The (1992)	Drama|Romance|War
6040	1	25	6	562	5	Welcome to the Dollhouse (1995)	Comedy|Drama
6040	1	25	6	1096	4	Sophie's Choice (1982)	Drama
6040	1	25	6	1097	4	E.T. the Extra-Terrestrial (1982)	Children's|Drama|Fantasy|Sci-Fi

Gambar 4. Hasil Label Encoding

3.3 Knowledge Graph

Setelah data berhasil melalui tahap preprocessing, langkah selanjutnya adalah membangun Knowledge Graph (KG) digunakan untuk merepresentasikan hubungan kompleks antara user dan item, serta atribut-atribut tambahan dari masing-masing entitas. Berikut langkah dalam membuat KG:

a. Representasi Node

Membuat representasi node untuk setiap entitas yang terlibat. Entitas ini berasal dari atribut-atribut pengguna dan film, yang mencakup UserID, MovieID, *Genre*, *Age*, *Occupation* dan *Gender*, dimana setiap entitas akan di-*mapping* ke dalam ID numerik unik untuk menghindari duplikasi dan memastikan konsistensi saat membangun graph. Representasi node ini bertujuan untuk menciptakan struktur graph heterogen di mana setiap jenis entitas dapat dikenali secara eksplisit.

b. Data Interaksi User-Item

Interaksi eksplisit antara user dan film direpresentasikan dalam bentuk matrix interaksi seperti persamaan (1), dimana nilai *Y* mereprestasikan rating yang diberikan user ke-i terhadap film ke-j. Nilai rating tersebut akan digunakan untuk bobot relasi antar node user dan film.

c. Pembentukan Triplet

Hubungan antar entitas dalam Knowledge Graph disusun dalam bentuk triplet menggunakan persamaan (2), Triplet-triplet ini mencerminkan hubungan semantik dan struktural yang akan digunakan sebagai dasar dalam pembuatan edge pada graph. Berikut hasil pembentukan triplet pada tabel 4.

Tabel 4. Triplet Hubungan Antar Node

head	*relation*	*tail*
UserID	*rates*	MovieID
UserID	*has_gender*	*Gender*
UserID	*has_age*	*Age*
UserID	*has_occupation*	*Occupation*
MovieID	*has_genre*	*Genre*

3.4 Splitting Dataset

Dataset dibagi menjadi tiga bagian, yaitu *train*, *validation* dan *test* dengan proporsi 80:10:10. Pada data *train* digunakan untuk melatih model, data *validation* digunakan untuk melatih model melalui hyperparameter yang sudah ditetapkan, dan data *test* digunakan untuk mengevaluasi model.

3.5 Graph Convolutional Network

a. Embedding Awal

Pada tahap awal pelatihan, setiap node dalam knowledge graph akan direpresentasikan dalam bentuk vektor embedding berdimensi 64. Embedding ini pada awalnya diinisialisasi secara acak dan akan diperbarui selama pelatihan. Contoh Embedding awal berdimensi 4, maka

Tabel 5. Embedding Awal Setiap Node

Node	Embedding
U1	[0.1, 0.2, 0.0, 0.4]
M1	[0.4, 0.1, 0.1, 0.3]

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

M2	[0.2, 0.3, 0.2, 0.1]
Action	[0.1, 0.0, 0.2, 0.2]
Thriller	[0.3, 0.2, 0.0, 0.2]
Comedy	[0.3, 0.1, 0.2, 0.1]
Romance	[0.4, 0.1, 0.2, 0.0]
M	[0.0, 0.2, 0.1, 0.2]
25	[0.2, 0.2, 0.2, 0.2]
Engineer	[0.1, 0.1, 0.3, 0.3]

Embedding awal akan diperbarui melalui proses propagasi informasi antar node graph.

b. Importance Sampling

Karena graph besar memiliki banyak tetangga, maka teknik importance sampling digunakan untuk memilih tetangga yang paling relevan. Pendekatan umum dari teknik ini dijelaskan dalam persamaan (3). Proses *importance sampling* terdiri dari tiga tahapan utama:

1. Perhitungan skor relevansi menggunakan persamaan (4) antara *embedding* pengguna u dan *embedding* relasi r, kemudian dikalikan dengan bobot rating.
2. Skor relevansi kemudian diubah menjadi nilai probabilitas menggunakan fungsi softmax pada persamaan (5)
3. Dari distribusi probabilitas yang didapat, dipilih top-5 tetangga dengan skor tertinggi untuk setiap jenis relasi.

c. Message Passing

Setelah tetangga paling relevan dipilih melalui *importance sampling*, dilakukan agregasi informasi melalui proses *message passing*. Cara pengerjaannya adalah:

1. Model mengumpulkan vektor *embedding* dari semua *node* tetangga yang telah terpilih pada langkah sebelumnya
2. Informasi yang terkumpul kemudian diagregasi menjadi satu vektor baru. Pada penelitian ini menggunakan *sum aggregation*, di mana *embedding* dari para tetangga yang telah diberi bobot dijumlahkan, sesuai persamaan (6).
3. Vektor agregat tersebut digunakan untuk memperbarui *embedding node* pusat. Sehingga menghasilkan sebuah *embedding* baru yang telah diperkaya dengan informasi dari lingkungannya, sehingga model dapat memahami pola dan ketergantungan lokal.

d. Prediksi

Setelah seluruh *embedding node* diperbarui melalui beberapa lapisan *message passing*, langkah terakhir adalah melakukan perhitungan prediksi melalui persamaan (7). Proses ini bertujuan untuk menghasilkan skor yang merepresentasikan probabilitas interaksi antara seorang pengguna u dan sebuah film v.

3.6 Pelatihan Model

Model awal dilatih terlebih dahulu menggunakan *data training* menggunakan parameter pada tabel 6.

Tabel 6. Parameter Awal Model

Parameter	Nilai
Optimizer	Adam
Batch Size	128
Learning Rate	0.005
Weight Decay	0.0001
Embedding Size	64
Model Layers	2

Model dilatih selama 10 epoch dengan menghasilkan sebanyak k = 10 item rekomendasi pada setiap iterasi pelatihan. Hasil penelitian menunjukkan bahwa penerapan teknik importance sampling mampu meningkatkan performa model secara signifikan dibandingkan dengan uniform sampling, khususnya pada metrik evaluasi Precision@K, Recall@K, NDCG@K, dan AUC. Hasil lebih lengkap terdapat pada tabel 7 dibawah ini.

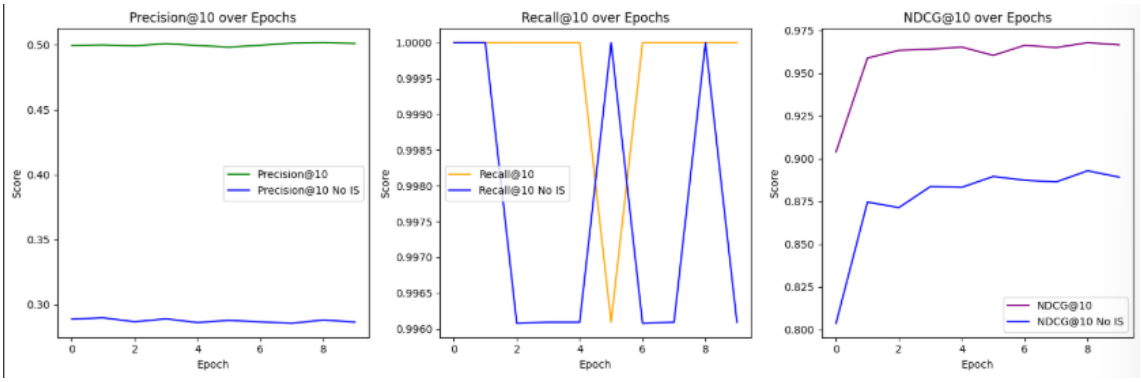
Tabel 7. Perbandingan Waktu Komputasi Training

Epoch	Importance Sampling		Uniform Sampling	
	Waktu (s)	AUC	Waktu (s)	AUC
1	219.55s	0.6843	199.98s	0.6465
2	223.82s	0.8393	197.19s	0.7766
3	441.95s	0.8485	478.49s	0.7866
4	536.63s	0.8521	513.62s	0.7926
5	617.53s	0.8554	592.25s	0.7953
6	663.63s	0.8586	626.83s	0.7969
7	699.07s	0.8616	664.86s	0.7986
8	735.14s	0.8620	690.11s	0.7988

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

9	761.97s	0.8628	694.50s	0.7994
10	768.45s	0.8636	722.15s	0.8004

Penerapan importance sampling terbukti memberikan nilai AUC yang lebih tinggi dibandingkan uniform sampling, yang menunjukkan kemampuan model dalam membedakan item relevan dan tidak relevan secara lebih efektif. Meskipun demikian, teknik ini memerlukan waktu komputasi yang lebih lama pada setiap epoch, menandakan adanya trade-off antara akurasi dan efisiensi waktu dalam proses pelatihan. Hal ini disebabkan oleh adanya proses tambahan untuk menghitung distribusi bobot penting dari edges sebelum dilakukan proses sampling. Meskipun memerlukan waktu komputasi yang lebih tinggi, peningkatan performa evaluasi yang signifikan menjadikan keuntungan, karena akurasi dan relevansi hasil rekomendasi yang dihasilkan oleh model menjadi lebih optimal. Berikut perbandingan metrik Precision@K, Recall@K, NDCG@K selama 10 epoch pada gambar 5.



Gambar 5. Evaluasi Importance Sampling vs Uniform Sampling

Hasil menunjukkan perbandingan kinerja model dengan menerapkan *importance sampling* dan *uniform sampling* selama pelatihan. Pada metrik Precision@10, model yang menerapkan *importance sampling* secara konsisten memperoleh nilai 0.50 lebih tinggi dibandingkan model yang menerapkan *uniform sampling* yang hanya mencapai 0.28, dari metrik ini menunjukkan bahwa penerapan *importance sampling* dapat membantu model merekomendasikan film yang benar-benar relevan untuk user pada posisi 10 teratas. Pada metrik Recall@10, model yang menerapkan *importance sampling* memiliki performa yang cukup stabil sebesar 1.000 walaupun sempat drop ke 0.9963, sedangkan pada model yang menerapkan *uniform sampling* mendapatkan nilai yang sedikit lebih rendah sebesar 0.9963. Hal ini menunjukkan bahwa *importance sampling* membantu model dalam menemukan hampir seluruh film relevan dalam top rekomendasi. Pada metrik NDCG@10 menunjukkan perbedaan yang cukup signifikan, model yang menerapkan *importance sampling* mampu mendapatkan skor 0.96, sedangkan model yang menerapkan *uniform sampling* hanya mendapatkan skor 0.88. Ini menunjukkan bahwa model yang menerapkan *importance sampling* dapat menempatkan posisi item yang sangat relevan di posisi tertinggi dalam daftar rekomendasi.

3.7 Hyperparameter

Kemudian dilakukan percobaan hyperparameter dengan memanfaatkan data validasi dengan beberapa parameter pada tabel 8.

Tabel 8. Hyperparameter Model

Parameter	Nilai
Learning_Rate	0.001, 0.01, 0.05
Embedding Size	32
Epoch	20

Berdasarkan hasil percobaan hyperparameter dengan optimizer Adam, kombinasi parameter terbaik diperoleh pada learning rate sebesar 0.001. Pada data validasi untuk *importance sampling*, konfigurasi ini menghasilkan Precision@10 sebesar 0.2904, Recall@10 sebesar 1.0000, NDCG@10 sebesar 0.8513, dan AUC sebesar 0.7459. Sementara itu, pada learning rate 0.01 hanya diperoleh AUC sebesar 0.7167, dan pada learning rate 0.05 AUC yang didapatkan turun signifikan menjadi 0.5002. Kemudian pada *uniform sampling*, kombinasi parameter terbaik diperoleh pada learning rate yang sama, konfigurasi ini menghasilkan Precision@10 sebesar 0.2871, Recall@10 sebesar 0.9961, NDCG@10 sebesar 0.8519, AUC sebesar 0.7624. Sementara itu, pada learning rate 0.01 hanya diperoleh Precision@10 sebesar 0.2876, Recall@10 sebesar 1.0000, NDCG@10 sebesar 8404, AUC sebesar 0.7221, dan pada learning rate 0.05 AUC yang didapatkan turun sangat signifikan menjadi 0.4997. Sehingga learning 0.001 dipilih sebagai konfigurasi terbaik karena memberikan performa evaluasi tertinggi pada data validasi. Hasil tersebut akan dijadikan acuan untuk melanjutkan ke tahap pelatihan ulang model sepenuhnya pada data latih, agar model yang dihasilkan bisa belajar secara maksimal dari seluruh data yang tersedia.

3.8 Pelatihan Ulang

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 4, Agustus 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i4.8772
Hal 610-621
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

Setelah menemukan kombinasi *hyperparamater* terbaik, kedua model (dengan *importance sampling* dan *uniform sampling*) dilatih ulang menggunakan seluruh data latih untuk membangun model final yang optimal. Pelatihan ulang dilakukan dengan kombinasi terbaik yang ditemukan yaitu, *learning rate* sebesar 0.001 dan *embedding size* 32. Hasil dari proses ini menunjukkan dampak positif yang signifikan. Model yang menerapkan *importance sampling* mengalami peningkatan performa sekitar 1% pada skor NDCG@10 dan AUC. Lebih lanjut, efisiensi waktu pelatihan meningkat drastis, dimana untuk model *importance sampling* mengalami penurunan dari rata-rata 566 detik menjadi 106 detik, dengan tren penurunan serupa terlihat pada model *uniform sampling*. Hal ini menunjukkan bahwa *embedding size* 32 merupakan konfigurasi yang sangat efektif, karena mampu menurunkan waktu pelatihan secara signifikan tanpa mengorbankan akurasi model.

3.9 Test Model

Kemudian dilakukan pengujian model pada data tes, yang merupakan data yang belum pernah dilihat oleh model. Pengujian ini dilakukan pada kedua model (*importance sampling* dan *uniform sampling)* yang telah dilatih ulang menggunakan hyperparameter terbaik. Hasil pengujian akhir dari kedua model dapat dilihat pada Tabel 9 berikut.

Tabel 9. Hasil Evaluasi Data Test

Metrik	Importance Sampling	Uniform Sampling
AUC	0.8798	0.8147
NDCG@10	0.9719	0.8983
Precision@10	0.4988	0.2845
Recall@10	1.0000	0.9961

Berdasarkan tabel perbandingan di atas, terlihat jelas bahwa model yang menggunakan *importance sampling* secara konsisten mengungguli model dengan *uniform sampling* di semua metrik evaluasi. Dari metrik Precision@10 teknik *importance sampling* menunjukkan kemampuan yang signifikan dalam merekomendasikan item yang relevan kepada user. Hal ini dibuktikan dengan skor yang mencapai 0.4988 pada *importance sampling* dibandingkan dengan *uniform sampling*. Jika dibandingkan dengan penelitian [3], yang menggunakan pendekatan *uniform sampling*. Penelitian ini mengambil langkah lebih jauh dengan membuktikan bahwa teknik *importance sampling* mampu memberikan peningkatan akurasi yang signifikan. Selain inovasi pada teknik sampling, penelitian ini membuktikan efektivitas integrasi data *user-end.* Penelitian [3] lebih berfokus pada pemanfaatan data *item-end.* Dengan menambahkan data demografis pengguna, model KGCN mampu menghasilkan representasi pengguna yang lebih kaya dan kontekstual. Ini memungkinkan personalisasi yang lebih mendalam dibandingkan model yang hanya mengandalkan data interaksi saja.

4. KESIMPULAN

Berdasarkan hasil penelitian dan analisis, penelitian ini membuktikan bahwa penerapan teknik *Importance Sampling* (IS) pada model Knowledge Graph Convolutional Network (KGCN) secara signifikan lebih unggul dibandingkan *Uniform Sampling* (US) pada seluruh metrik evaluasi. Keunggulan ini dicapai melalui pemanfaatan Knowledge Graph yang mengintegrasikan data demografis pengguna, yang memungkinkan IS memilih tetangga paling relevan secara lebih efektif selama proses pelatihan. Selain itu, proses hyperparameter tuning menemukan bahwa arsitektur model yang lebih ramping (embedding size 32) mampu meningkatkan akurasi sekaligus efisiensi waktu komputasi secara signifikan. Secara keseluruhan, penelitian ini mengonfirmasi bahwa integrasi KGCN dengan Importance Sampling dan data user-end merupakan pendekatan yang robust, efisien, dan akurat untuk membangun sistem rekomendasi film dengan kemampuan generalisasi yang baik. Saran untuk pengembangan lebih lanjut dari penelitian ini dapat diarahkan pada penggunaan arsitektur jaringan graf alternatif seperti *Relational Graph Convolutinal Network* (R-GCN) atau *Graph Attention Network* (GAT) dapat menjadi pilihan untuk menangkap relasi multi-tipe antar node dengan lebih akurat, dan menambahkan atribut seperti waktu, lokasi dan perangkat yang digunakan pengguna untuk menghasilkan sistem rekomendasi yang lebih personal dan kontekstual.

REFERENCES

[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] 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.

[5] 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

[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] 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.
[8] T. Al-shehari and R. A. Alsowail, "An insider data leakage detection using one-hot encoding, synthetic minority oversampling and machine learning techniques," Entropy, vol. 23, no. 10, Oct. 2021, doi: 10.3390/e23101258.
[9] Intan Permata and Esther Sorta Mauli Nababan, "Application Of Game Theory In Determining Optimum Marketing Strategy In Marketplace," JURNAL RISET RUMPUN MATEMATIKA DAN ILMU PENGETAHUAN ALAM, vol. 2, no. 2, pp. 65–71, Jul. 2023, doi: 10.55606/jurrimipa.v2i2.1336.
[10] L. Wang and Y. Zhang, "Digital Dissemination of Information Based on Knowledge Graph Recommendation Algorithm," Informatica (Slovenia), vol. 48, no. 19, pp. 31–44, 2024, doi: 10.31449/inf.v48i19.6561.
[11] 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.
[12] Y. Wang, J. Zeng, Y. Wang, C. Wang, Y. Liu, and X. Wu, "Network Representation Learning: From Preprocessing, Feature Extraction to Node Embedding," ACM Computing Surveys (CSUR), vol. 55, no. 11, pp. 1–36, 2023, doi: 10.1145/3491206.
[13] 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.
[14] 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.
[15] 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.
[16] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," Sci Rep, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-56706-x.
[17] S. Siet, S. Peng, S. Ilkhomjon, M. Kang, and D. S. Park, "Enhancing Sequence Movie Recommendation System Using Deep Learning and KMeans," Applied Sciences (Switzerland), vol. 14, no. 6, Mar. 2024, doi: 10.3390/app14062505.
[18] S. Subhan, D. L. Syarif, E. Widhihastuti, S. K. Rakainsa, M. Sam'an, and Y. N. Ifriza, "Improved recommender system using Neural Network Collaborative Filtering (NNCF) for E-commerce cosmetic product," Sinergi (Indonesia), vol. 29, no. 1, pp. 155–162, 2025, doi: 10.22441/sinergi.2025.1.014.
[19] H. Yang, J. Li, and S. Jahng, "The Application of Knowledge Graph Convolutional Network-based Film and Television Interaction under Artificial Intelligence," IEEE Access, 2024, doi: 10.1109/ACCESS.2024.3459448.
[20] 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.