

Perbandingan Penggunaan Arsitektur Graph GCN Dan LightGCN Pada Sistem Rekomendasi Hotel Berbasis Rating Pengguna Dengan Dataset Terbatas

Rhodil Fauzi^{1*}, Abu Tholib², Ahmad Hudawi AS³

^{1,2,3} Program Studi Teknik Informatika, Universitas Nurul Jadid

*rhodilyo@gmail.com

Abstrak

Sistem rekomendasi perhotelan sering gagal merekomendasikan hotel baru akibat masalah kelangkaan data ekstrem (item cold-start) dan rentan terhadap fenomena over-smoothing komputasi. Penelitian ini bertujuan mengevaluasi dan membandingkan secara komprehensif arsitektur pemeringkatan Graph Convolutional Network (GCN) dan LightGCN. Metode yang digunakan adalah eksperimen komputasional menggunakan pendekatan Ablation Study untuk membedah pengaruh kedalaman propagasi (1-Layer vs Multi-Layer) dan injeksi fitur eksternal. Pengujian dilakukan pada dataset berukuran kecil dari Mendeley Data yang berisi riwayat ulasan untuk 16 entitas hotel. Hasil utama menunjukkan bahwa LightGCN 1-Layer dengan fitur merupakan model paling superior untuk pengguna aktif (warm-start), dengan nilai NDCG@10 sebesar 0.8702. Namun pada skenario ekstrem hotel baru (0-shot cold-start), arsitektur dangkal tersebut gagal dan solusi terbaik justru dimenangkan oleh model Multi-Layer murni tanpa fitur (No-Feature) yang mencapai Hit Ratio (HR@10) 71.43%. Kesimpulannya, tidak ada model tunggal yang sempurna untuk semua kondisi; implementasi sistem disarankan menggunakan kerangka kerja ganda (dual-framework) yang memadukan kecepatan LightGCN 1-Layer dan ketangguhan propagasi Multi-Layer.

Kata kunci : Graph Neural Network, Item Cold-Start, LightGCN, Over-smoothing, Sistem Rekomendasi.

Abstract

Hotel recommendation systems often fail to recommend new hotels due to extreme data sparsity problems (item cold-start) and are vulnerable to the computational over-smoothing phenomenon. This study aims to comprehensively evaluate and compare the ranking architectures of Graph Convolutional Network (GCN) and LightGCN. The method used is a computational experiment using an Ablation Study approach to dissect the effect of propagation depth (1-Layer vs. Multi-Layer) and the injection of external features. The evaluation was conducted on a small-sized dataset from Mendeley Data containing review histories for 16 hotel entities. The main results show that the 1-Layer LightGCN with features is the most superior model for active users (warm-start), achieving an NDCG@10 score of 0.8702. However, in the extreme new hotel scenario (0-shot cold-start), this shallow architecture failed, and the best solution was actually won by the pure Multi-Layer model without features (No-Feature), which achieved a Hit Ratio (HR@10) of 71.43%. In conclusion, there is no single perfect model for all conditions; system implementation is recommended to adopt a dual-framework that integrates the speed of 1-Layer LightGCN and the propagation robustness of the Multi-Layer model.

Keywords : Graph Neural Network, Item Cold-Start, LightGCN, Over-smoothing, Recommendation Systems

1. Pendahuluan

Di era digitalisasi saat ini, pesatnya pertumbuhan industri pariwisata memunculkan tantangan nyata berupa melimpahnya data informasi (information overload) pada berbagai platform

daring. Banyaknya pilihan penginapan seringkali membuat wisatawan mengalami hambatan dan kebingungan dalam menentukan pilihan yang sesuai secara manual, seperti yang disoroti dalam suatu studi ^[1]. Melimpahnya data ini

menuntut adanya otomatisasi sistem penyaringan informasi yang efisien. Dari sudut pandang penyedia layanan pariwisata, tantangan utamanya adalah bagaimana menyajikan informasi yang tepat sasaran untuk meningkatkan kepuasan pengguna secara signifikan, sebagaimana dibuktikan dalam sejumlah riset terdahulu [2], [3].

Merespons isu krusial tersebut, pengembangan sistem cerdas menjadi prioritas utama para peneliti untuk membantu memproses keputusan pengguna secara cepat. Dalam beberapa tahun terakhir, sistem rekomendasi telah berkembang pesat dan diimplementasikan pada berbagai studi kasus di Indonesia, mulai dari sistem rekomendasi peminjaman perpustakaan institusi pendidikan [4] hingga prototipe penyaringan film nasional [5], serta berbagai aplikasi luas lainnya [6]. Dengan banyaknya pilihan yang tersedia, sistem rekomendasi bertindak sebagai alat penyaring informasi yang menawarkan konten relevan dan dipersonalisasi kepada pengguna [7]. Secara fundamental, pemodelan awal sistem ini mengandalkan interaksi historis pengguna untuk memprediksi preferensi yang akurat, sebagaimana dijelaskan dalam penelitian [8]. Lebih jauh lagi, pendapat pakar dalam sebuah literatur [9] menegaskan bahwa kemampuan personalisasi yang presisi tidak hanya mereduksi beban kognitif pengguna, tetapi juga berdampak langsung pada peningkatan loyalitas

pelanggan serta nilai bisnis platform digital. Pemanfaatan teknologi graf, khususnya Graph Neural Networks (GNN), dalam sistem rekomendasi kemudian dieksplorasi secara mendalam guna mengatasi kelemahan model konvensional yang sering terjebak pada masalah kekerenggangan data (data sparsity), seperti yang dikemukakan dalam studi [10]. Kajian fundamental dari sebuah riset [11] mengusulkan Neural Graph Collaborative Filtering (NGCF) yang berhasil mengeksplorasi graf bipartit ke dalam proses embedding. Lebih lanjut, penelitian lain [12] membuktikan bahwa relasi high-order sangat esensial dalam meningkatkan akurasi rekomendasi. Pendekatan ini kemudian disempurnakan oleh studi lanjutan [12] melalui LightGCN, yang menyederhanakan arsitektur GCN dengan menghapus operasi transformasi fitur dan aktivasi non-linier, serta membuktikan efisiensinya untuk dataset besar. Di domain spesifik perhotelan, sebuah kajian [13] memperluas kemampuan GNN dengan melakukan feature engineering pada atribut hotel yang mampu meningkatkan performa model hingga 20%.

Namun, kelemahan spesifik dari literatur-literatur mutakhir tersebut adalah ketergantungan model yang sangat bias terhadap ekosistem dataset berskala raksasa (dense data) dengan jutaan interaksi, sebagaimana dievaluasi pada studi [14]. Pendekatan dari penelitian sebelumnya belum

ada yang menguji secara komprehensif mengenai bagaimana arsitektur graf ini bertahan jika dipaksa berjalan pada lingkungan yang memicu skenario extreme item cold-start (kondisi ketika hotel baru sama sekali belum memiliki riwayat ulasan). Penggunaan dataset berukuran kecil dalam penelitian ini menjadi sangat penting dan krusial karena merepresentasikan kondisi realistis dari banyak platform bisnis perhotelan rintisan atau aplikasi lokal yang belum memiliki basis data pengguna yang besar pada fase awal peluncurannya ^[15]. Berpijak pada celah penelitian (research gap) yang secara spesifik belum terpecahkan tersebut, kebaruan (novelty) dari riset ini adalah menguji dan memetakan daya tahan serta perilaku komputasional dari arsitektur GNN ketika dipaksa bekerja pada lingkungan data murni yang sangat terbatas. Secara teoritis, penerapan model graf yang berlapis-lapis pada dataset kecil memiliki risiko tinggi memicu anomali komputasi. Landasan ilmiah dari jurnal primer ^[16] menjelaskan tentang adanya fenomena over-smoothing, yaitu suatu kondisi di mana representasi node menjadi terlalu identik akibat komputasi perambatan lapisan yang berlebihan. Kekhawatiran mengenai risiko terjadinya overfitting pada model berarsitektur berat jika berjalan pada data terbatas juga diperkuat oleh studi ^[17]. Mengurai trade-off antara kecanggihan struktural graf dengan keterbatasan volume data inilah yang

menjadi sumbangsih kontribusi original dari penelitian ini.

Oleh karena itu, fokus penelitian ini diarahkan untuk mengeksplorasi dan membandingkan secara langsung tingkat efektivitas arsitektur Graph Convolutional Network (GCN) dan LightGCN secara head-to-head menggunakan dataset perhotelan berukuran kecil. Evaluasi perbandingan ini sangat krusial guna menemukan arsitektur algoritma yang paling optimal, hemat sumber daya komputasi, terhindar dari over-smoothing, namun tetap menghasilkan tingkat akurasi yang tinggi untuk menanggulangi masalah cold-start ekstrem di industri perhotelan.

2. Tinjauan Pustaka

2.1. Penelitian Terkait

Referensi dan perbandingan dari penelitian terdahulu yang digunakan dalam riset ini meliputi:

1. Penelitian mengusulkan arsitektur Light Graph Convolution Network (LightGCN) dengan menghapus operasi transformasi fitur dan aktivasi non-linier yang redundan pada GCN. Hasil uji coba pada dataset Yelp2018 dan Amazon-Book menunjukkan penghapusan ini meningkatkan efisiensi komputasi dan mencatatkan keunggulan metrik Recall@20 sebesar 16,01% dibandingkan NGCF ^[12].

2. Studi yang menerapkan algoritma Graph Neural Network (GNN) dengan rekayasa fitur geolokasi dan amenitas pada sistem rekomendasi

hotel di Indonesia. Hasil eksperimen membuktikan integrasi atribut eksternal ini mendongkrak akurasi prediksi hingga 20% pada kondisi pengguna aktif (warm-start) [13].

3. Riset yang mengembangkan rekomendasi hotel hibrida berbasis kombinasi ulasan teks multi-kriteria dan teknik clustering. Algoritma ini meminimalkan kesalahan prediksi minat konsumen secara drastis dengan capaian skor Mean Absolute Error (MAE) sebesar 0,182 [3].

4. Penelitian dengan tema mengevaluasi preferensi wilayah untuk mengatasi luapan informasi pada platform pariwisata. Penerapan rekomendasi geospasial ini menghasilkan akurasi penawaran katalog akomodasi yang presisi dengan nilai Precision@10 sebesar 0,742 [1].

5. Evaluasi yang menyoroti tantangan kelangkaan interaksi ekstrem (cold-start) pada platform komersial digital menggunakan deep learning. Hasil uji menunjukkan efisiensi sistem tanpa manipulasi graf yang tepat anjlok hingga di bawah 35% saat matriks transaksi berada pada level sangat minim [15].

2.2. Landasan Teori

1. Sistem Rekomendasi

Sistem rekomendasi merupakan arsitektur perangkat lunak untuk memprediksi tingkat preferensi pengguna terhadap suatu item [18]. Secara teknis, sistem ini menyaring informasi secara terotomatisasi demi menyajikan saran

paling relevan bagi profil pengguna [19]. Guna menangkap interaksi data yang kompleks, tren pemodelan kini bergeser memanfaatkan arsitektur deep learning [20]. Sistem personalisasi yang akurat terbukti memberikan dampak ekonomi masif melalui peningkatan konversi dan retensi pengguna [21].

2. Masalah Cold-Start dan Data Sparsity

Cold-start merupakan kondisi kritis saat algoritma gagal memprediksi akibat ketiadaan riwayat interaksi awal pada pengguna atau item baru [22]. Masalah ini berkorelasi erat dengan data sparsity (kekerenggangan data), di mana volume interaksi historis jauh lebih sedikit dibandingkan kapasitas katalog item [23]. Evaluasi empiris [24] membuktikan kekerenggangan data ekstrem merusak kapasitas representasi model serta menurunkan akurasi prediksi konvensional.

3. Graph Neural Network (GNN)

Graph Neural Network (GNN) adalah kerangka deep learning untuk memproses data berstruktur non-Euclidean atau graf [25]. Pada sistem rekomendasi, interaksi timbal balik pengguna-item dimodelkan sebagai graf bipartit [26]. Pembaruan fitur representasi simpul (node embedding) dilakukan melalui teknik message passing dengan mengagregasi informasi dari simpul tetangga secara iteratif [27].

4. Arsitektur GCN dan LightGCN

Graph Convolutional Network (GCN) menggunakan transformasi matriks bobot dan

aktivasi non-linier untuk menyaring informasi graf. Namun, riset [28] menunjukkan operasi rumit tersebut redundan dan memperlambat komputasi rekomendasi. Sebagai solusinya, model LightGCN diperkenalkan [12] dengan mengeliminasi komponen non-linier tersebut agar pembelajaran murni berfokus pada agregasi lingkungan (neighborhood aggregation), sebuah konsep efisiensi yang diadopsi luas pada berbagai riset graf mutakhir [29].

5. Fenomena Over-Smoothing

Penggunaan lapisan graf (layer) yang terlalu dalam dapat memicu cacat komputasi berupa over-smoothing. Kondisi ini mengikis daya ekspresif fitur karena representasi setiap node terus dirata-ratakan secara berulang [30]. Akibatnya, nilai vektor pengguna dan item menjadi seragam (indistinguishable) sehingga model graf gagal menghasilkan daftar rekomendasi yang spesifik [31].

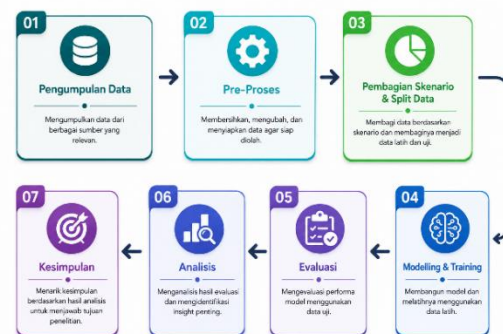
6. Metrik Evaluasi Kinerja (HR dan NDCG)

Kualitas akurasi daftar peringkat Top-N diukur menggunakan parameter statistik objektif. Metrik Hit Ratio (HR) menghitung persentase keberhasilan sistem menampilkan item target dalam daftar rekomendasi pengguna. Sementara itu, Normalized Discounted Cumulative Gain (NDCG), yang berakar dari riset temu kembali informasi [32], mengevaluasi kualitas urutan peringkat melalui pemberian penalti logaritmik posisi objek. Kombinasi kedua instrumen ini

terbukti sangat diskriminatif untuk memvalidasi performa rekomendasi [33]

3. Metode Penelitian

Penelitian ini menerapkan eksperimen komputasional (ablation study) menggunakan perangkat keras virtual Google Colaboratory. Data yang digunakan adalah data sekunder bersumber dari repositori publik Mendeley Data. Guna mengevaluasi ketahanan model, dataset perhotelan ini dipilih karena memiliki skala kecil dan tingkat kekerenggangan (sparsity) yang ekstrem, yang secara spesifik terdiri dari 5.390 user, 16 hotel, dan total 5.798 riwayat interaksi. Data ini juga dilengkapi dengan atribut hotel (geolokasi dan aspek penilaian tamu).



Gambar 1. Alur Program

3.1. Pra-pemrosesan Data & Pemisahan Data

Tahap awal mengubah data interaksi menjadi struktur graf bipartit guna memfasilitasi model dalam menangkap sinyal kolaboratif secara alamiah. Atribut pelengkap hotel diekstraksi menjadi vektor fitur berdimensi 21. Data kemudian diekstraksi menjadi himpunan latih

(train), validasi (validation), dan uji (test) dengan dua skenario: (a) Skenario umum (warm-start), dan (b) Skenario item baru (cold-start) melalui simulasi batasan riwayat (few-shot $K=0, 1, 3$, dan 5).

3.1. Pemodelan Arsitektur

Terdapat delapan varian arsitektur yang dikomparasikan (kombinasi GCN vs. LightGCN, injeksi fitur vs. tanpa fitur, dan kedalaman 1-Layer vs. 3-Layer). Seluruh model dioptimasi menggunakan *Bayesian Personalized Ranking* (BPR Loss) dan *negative sampling* untuk pemeringkatan *item* implisit. Pada GCN, perambatan pesan (*message passing*) untuk pembaruan *node* antarlapisan dihitung menggunakan Persamaan (1).

$$GCN \quad h_i^{(l+1)} = \sigma \left(\sum_{j \in \mathcal{N}(i)} \frac{1}{\sqrt{|\mathcal{N}(i)|} \sqrt{|\mathcal{N}(j)|}} h_j^{(l)} W^{(l)} \right) \quad (1)$$

Sebaliknya, arsitektur LightGCN menyederhanakan komputasi dengan menghilangkan transformasi matriks bobot dan aktivasi non-linier. Pembaruan representasi *user* dan *item* pada LightGCN dirumuskan pada Persamaan (2).

$$LightGCN \quad \begin{aligned} e_u^{(k+1)} &= \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|} \sqrt{|\mathcal{N}_i|}} e_i^{(k)} \\ e_i^{(k+1)} &= \sum_{u \in \mathcal{N}_i} \frac{1}{\sqrt{|\mathcal{N}_i|} \sqrt{|\mathcal{N}_u|}} e_u^{(k)} \end{aligned} \quad (2)$$

3.2. Evaluasi & Analisis

Kinerja prediksi model dievaluasi dalam menghasilkan daftar rekomendasi Top-K ($K=1, 3, 5, 10$) menggunakan tiga metrik fundamental: HR@k (Hit Ratio), Recall@k, dan NDCG@k. Analisis komparatif kemudian difokuskan pada evaluasi daya tahan model (trade-off) dalam menanggulangi cold-start ekstrem di dataset bervolume kecil, serta menginvestigasi dampak fenomena komputasional bawaan seperti feature noise, over-smoothing, dan overfitting untuk menyimpulkan arsitektur yang paling optimal

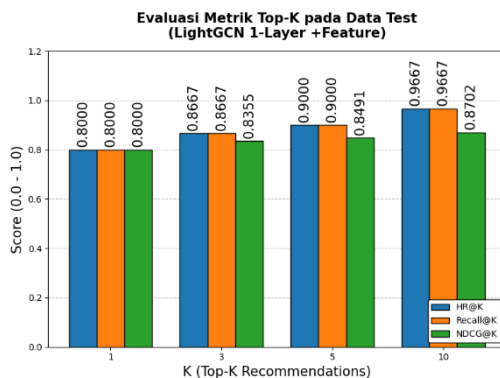
4. Hasil dan Pembahasan

4.1. Hasil Penelitian

Pengujian kinerja model dievaluasi pada dua skenario ketersediaan data, yakni warm-start (pengguna aktif) dan cold-start ekstrem (hotel baru / few-shot = 0). Evaluasi diukur menggunakan parameter Hit Ratio (HR), Recall, dan Normalized Discounted Cumulative Gain (NDCG).

1. Hasil Skenario Warm-Start

Rincian metrik pada skenario warm-start disajikan pada Tabel 1, dan pergerakan model terbaik divisualisasikan pada Gambar 2.



Gambar 2. Evaluasi Metrik LightGCN 1 Layer + Featur pada skenario warm start

Tabel 1. Komparasi Hasil HR@10, Recall@10 & NDCG@10 semua model di skenario warm start

Model	HR@10	Recall@10	NDCG@10
GCN 1 Layer No Feat	0.9167	0.9167	0.5316
GCN 1 Layer + Feat	0.9500	0.9500	0.5566
GCN 3 Layer No Feat	0.9333	0.9333	0.5401
GCN 3 Layer + Feat	0.9333	0.9333	0.5288
LightGCN 1 Layer No Feat	0.9667	0.9667	0.8390
LightGCN 1 Layer + Feat	0.9667	0.9667	0.8702
Light GCN 3 Layer No Feat	0.9000	0.9000	0.4927
LightGCN 3 Layer + Feat	0.9167	0.9167	0.7925

Berdasarkan Tabel 1, LightGCN 1-Layer (+Feature) terbukti menjadi model paling superior dengan capaian nilai HR@10 sebesar 0.9667 (96,67%) dan NDCG@10 tertinggi sebesar 0.8702. Capaian ini secara mutlak mengungguli arsitektur GCN konvensional (GCN 1-Layer +Feat) yang hanya mampu mencetak NDCG sebesar 0.5566. Tingginya nilai NDCG pada LightGCN 1-Layer ini mengindikasikan

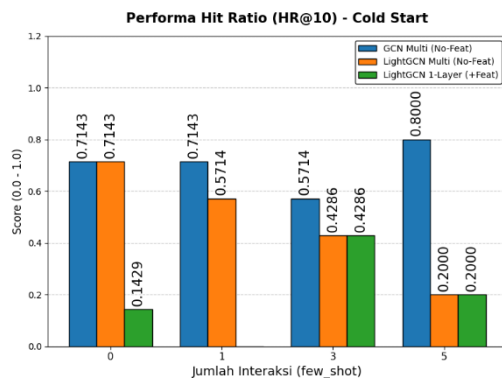
bahwa model tidak hanya berhasil menebak hotel yang relevan, tetapi juga sukses menempatkannya pada urutan teratas daftar rekomendasi. Secara praktis, akurasi pemeringkatan ini akan sangat mempermudah wisatawan dalam mengambil keputusan tanpa perlu menggulir layar perangkat mereka terlalu jauh.

2. Hasil Skenario Cold-Start

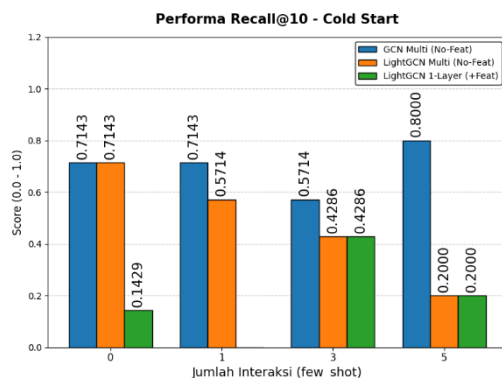
Rincian metrik pada skenario cold-start disajikan pada Tabel 2, dan pergerakan model terbaik divisualisasikan pada Gambar 3, 4, dan 5.

Tabel 2. Komparasi Hasil HR@10, Recall@10 & NDCG@10 semua model di skenario cold start dengan few-shot 0

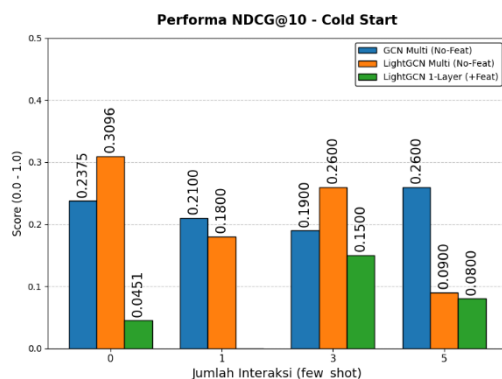
Model	Few shot	HR @10	Recall @10	NDCG @10
GCN 1 Layer No Feat	0	0.1429	0.1429	0.0430
GCN 1 Layer + Feat	0	0.2857	0.2857	0.1905
GCN 3 Layer No Feat	0	0.7143	0.7143	0.2375
GCN 3 Layer + Feat	0	0.4286	0.4286	0.1390
LightGCN 1 Layer No Feat	0	0.4286	0.4286	0.1732
LightGCN 1 Layer + Feat	0	0.1429	0.1429	0.0451
Light GCN 3 Layer No Feat	0	0.7143	0.7143	0.3096
LightGCN 3 Layer + Feat	0	0.1429	0.1429	0.0430



Gambar 3. Evaluasi Metrik HR@10 GCN



Gambar 4. Evaluasi Metrik Recall@10 GCN



Gambar 5. Evaluasi Metrik NDCG@10 GCN

Dinamika performa bergeser sangat kontras. Dominasi LightGCN 1-Layer runtuh dan hanya mampu mencetak HR di bawah 0.28. Skenario ekstrem ini justru dimenangkan oleh model berarsitektur dalam tanpa fitur, yakni LightGCN Multi-Layer (No-Feature) dan GCN Multi-Layer

(No-Feature), yang secara impresif mendominasi dengan capaian HR@10 sebesar 0.7143 (71,43%). Meskipun nilai NDCG pada fase ini secara alamiah mengalami penurunan dibandingkan skenario warm-start akibat terbatasnya sinyal preferensi, pencapaian Hit Ratio yang tinggi membuktikan bahwa model Multi-Layer tetap sanggup memunculkan hotel baru tersebut ke dalam radar pencarian pengguna, sekaligus mengamankannya dari risiko isolasi permanen di dalam platform.

3. Analisis Waktu Komputasi (Efisiensi Arsitektur)

Selain metrik akurasi, efisiensi arsitektur diukur berdasarkan metrik waktu komputasi pelatihan (training computational time). Berdasarkan log komputasi sistem, arsitektur LightGCN terbukti mengonsumsi waktu eksekusi per epoch yang jauh lebih cepat dibandingkan GCN konvensional. Penghapusan operasi aktivasi non-linier dan matriks bobot, khususnya pada varian 1-Layer, berhasil memangkas beban alokasi memori (RAM/GPU) secara drastis, menjadikannya model yang paling ringan dan efisien secara operasional untuk diimplementasikan pada ekosistem web sesungguhnya. Keunggulan komputasional ini memberikan implikasi positif terhadap skalabilitas sistem, di mana penyedia layanan dapat menghemat biaya server cloud computing sekaligus mempercepat waktu respon (inference

time) saat menyajikan rekomendasi secara real-time kepada ribuan pengguna yang mengakses platform secara bersamaan

4.2. Pembahasan

Bagian ini menginterpretasikan fenomena anomali dari hasil pengujian yang memunculkan dikotomi arsitektur yang kontras antara kondisi warm-start dan cold-start.

Pertama, keunggulan telak LightGCN 1-Layer pada skenario warm-start mengonfirmasi bahwa penghapusan transformasi non-linear sangat efektif untuk memodelkan implicit feedback. Penggunaan arsitektur dangkal (1-Layer) terbukti krusial untuk dataset berskala mikro ini. Penambahan lapisan (Multi-Layer) pada fase ini justru memicu fenomena over-smoothing, di mana vektor representasi (embedding) pengguna dan hotel menjadi terlalu homogen. Hal ini dibuktikan dengan penurunan drastis kualitas pemeringkatan (NDCG LightGCN +Feat merosot dari 0.8702 di 1-Layer menjadi 0.7925 di Multi-Layer). Dalam konteks dataset mikro yang struktur grafnya sempit, perambatan pesan yang terlalu dalam menyebabkan model gagal mengenali preferensi unik dari masing-masing individu karena seluruh node pengguna diasumsikan menyukai tipe hotel yang sama persis.

Kedua, kejatuhan model 1-Layer pada skenario cold-start membuktikan bahwa satu lapisan

konvolusi tidak memiliki jangkauan arsitektural untuk menemukan asosiasi node yang terisolasi. Arsitektur Multi-Layer mengatasi kebuntuan ini melalui mekanisme multi-hop propagation. Secara arsitektural, lapisan pertama hingga lapisan ketiga pada model ini bekerja sebagai jembatan tidak langsung (indirect bridge). Sinyal preferensi disebarkan secara berantai dari tetangga ke tetangga lainnya, sehingga sistem mampu menebak probabilitas ketertarikan meskipun hotel tersebut belum pernah mendapatkan ulasan secara langsung.

Ketiga, terjadi anomali feature noise dan overfitting. Secara intuitif, penambahan fitur eksternal (+Feature) seperti geolokasi dan amenitas diharapkan menambal kekosongan riwayat. Namun, varian murni (No-Feature) justru konsisten memenangkan skenario cold-start. Pada ukuran matriks referensi (support set) yang sangat minim, model berfitur sangat rentan terdistraksi oleh varians atribut yang tidak merepresentasikan pola umum. Sebagai contoh, pada kondisi kelangkaan data ekstrem, model berfitur dapat dengan mudah mengambil kesimpulan keliru bahwa pengguna menyukai seluruh hotel di area geografis tertentu hanya karena satu-satunya klik yang tersedia berada di area tersebut. Sebaliknya, model murni yang hanya bertumpu pada topologi graf struktural (ikatan murni bipartit) dipaksa untuk lebih objektif dalam mencari pola ketertarikan tersembunyi

(latent interest), yang terbukti jauh lebih kokoh (robust) dalam menembus anomali informasi ekstrem

5. Kesimpulan

Penelitian ini menyimpulkan bahwa performa arsitektur Graph Neural Network (GNN) sangat bergantung pada ketersediaan riwayat data. Pada skenario pengguna aktif (warm-start), LightGCN 1-Layer dengan fitur terbukti paling superior (NDCG@10 sebesar 0.8702) dan sukses memitigasi over-smoothing. Sebaliknya, pada skenario ekstrem hotel baru (0-shot item cold-start), arsitektur tersebut gagal dan secara mutlak diungguli oleh model Multi-Layer tanpa fitur (No-Feature). Model mendalam ini tangguh melakukan propagasi multi-hop pada node terisolasi dan kebal terhadap feature noise hingga mencetak Hit Ratio (HR@10) 71.43%. Oleh karena itu, implementasi sistem rekomendasi perhotelan disarankan mengadopsi kerangka kerja ganda (dual-framework) yang memadukan kecepatan LightGCN 1-Layer dan ketangguhan propagasi Multi-Layer.

Keterbatasan penelitian ini terletak pada penggunaan dataset berskala mikro dengan tingkat kekerenggangan (sparsity) ekstrem, sehingga generalisasi performa arsitektur pada ekosistem data raksasa belum teruji. Selain itu, rekomendasi dual-framework belum

mengkalkulasi beban komputasi tambahan (overhead) akibat beroperasinya dua model secara bersamaan. Penelitian lanjutan disarankan untuk merancang sebuah algoritma adaptif tunggal yang mampu menggabungkan keunggulan kedua arsitektur graf tersebut secara dinamis dan efisien

6. Daftar Pustaka

- [1] A. Solano-Barliza *et al.*, "Recommender systems applied to the tourism industry: a literature review," *Cogent Business & Management*, vol. 11, no. 1, p. 2367088, 2024, doi: <https://doi.org/10.1080/23311975.2024.2367088>.
- [2] J. Borràs, A. Moreno, and A. Valls, "Intelligent tourism recommender systems: A survey," *Expert Syst. Appl.*, vol. 41, no. 16, pp. 7370–7389, 2014, doi: <https://doi.org/10.1016/j.eswa.2014.06.007>.
- [3] M. Nilashi *et al.*, "Recommendation agents and information sharing through social media for coronavirus outbreak," *Telematics and Informatics*, vol. 61, p. 101597, 2021, doi: <https://doi.org/10.1016/j.tele.2021.101597>.
- [4] R. B. Perdana, D. Hartanti, and H. Hasanah, "Sistem Rekomendasi Peminjaman Buku Menggunakan Metode Vector Space Model Berbasis Pembobotan TF-IDF dan FastText (Studi Kasus Perpustakaan SMP Negeri 1 Kartasura)," *Infotek : Jurnal Informatika dan Teknologi*, vol. Vol. 8 No. 2 (2025), Jul. 2025, doi: <https://doi.org/10.29408/jit.v8i2.30540>.
- [5] D. T. Santoso, V. Atina, and D. Hartanti, "Prototipe Sistem Rekomendasi Film Indonesia Menggunakan Pendekatan Content Based Filtering dan Metode Vector Space Model," *Infotek : Jurnal Informatika dan Teknologi*, vol. Vol. 7 No. 2 (2024), Jul.

- 2024, doi: <https://doi.org/10.29408/jit.v7i2.26083>.
- [6] A. Tholib, T. Widiyaningtyas, and D. D. Prasetya, "An intelligent recommendation system utilizing a hybrid deep learning method," *Engineering, Technology & Applied Science Research*, vol. 15, no. 4, pp. 25971–25977, 2025, doi: <https://doi.org/10.48084/etasr.12230>.
- [7] A. Tholib, *Menguasai Sistem Rekomendasi di Machine Learning: Teori dan Praktik dengan Bahasa Pemrograman Python*. Kaizen Media Publishing, 2025.
- [8] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer (Long. Beach. Calif.)*, vol. 42, no. 8, pp. 30–37, 2009, doi: <https://doi.org/10.1109/MC.2009.263>.
- [9] D. Jannach and M. Jugovac, "Measuring the business value of recommender systems," *ACM Transactions on Management Information Systems (TMIS)*, vol. 10, no. 4, pp. 1–23, 2019, doi: <https://doi.org/10.1145/3370082>.
- [10] G. Adomavicius and A. Tuzhilin, "Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," 2005.
- [11] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, 2019, pp. 165–174. doi: <https://doi.org/10.1145/3331184.3331267>.
- [12] X. He, K. Deng, X. Wang, Y. Li, Y. D. Zhang, and M. Wang, "LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation," in *SIGIR 2020 - Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Association for Computing Machinery, Inc, Jul. 2020, pp. 639–648. doi: [10.1145/3397271.3401063](https://doi.org/10.1145/3397271.3401063).
- [13] F. A. Panarto, T. M. Phanghegar, V. Gunardi, and T. W. Cenggoro, "Leveraging user-item interactions and hotel characteristics for hotel recommendations in Indonesia with graph neural networks," *Procedia Comput. Sci.*, vol. 245, pp. 710–719, 2024, doi: <https://doi.org/10.1016/j.procs.2024.10.297>.
- [14] Z. Sun *et al.*, "Are we evaluating rigorously? benchmarking recommendation for reproducible evaluation and fair comparison," in *Proceedings of the 14th ACM Conference on Recommender Systems*, 2020, pp. 23–32. doi: <https://doi.org/10.1145/3383313.3412489>.
- [15] S. Kumar, V. K. Chauhan, D. Upadhyay, S. Singh, and P. Tripathi, "Enhancing recommender systems to alleviate data sparsity and the cold start problem," in *2023 12th International Conference on System Modeling & Advancement in Research Trends (SMART)*, IEEE, 2023, pp. 486–491. doi: <https://doi.org/10.1109/SMART59791.2023.10428531>.
- [16] Q. Li, Z. Han, and X.-M. Wu, "Deeper insights into graph convolutional networks for semi-supervised learning," in *Proceedings of the AAAI conference on artificial intelligence*, 2018. doi: <https://doi.org/10.1609/aaai.v32i1.11604>.
- [17] L. Chen, L. Wu, R. Hong, K. Zhang, and M. Wang, "Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach," in *Proceedings of the AAAI conference on artificial intelligence*, 2020, pp. 27–34. doi: <https://doi.org/10.1609/aaai.v34i01.5330>.
- [18] C. C. Aggarwal, *Recommender systems*, vol. 1, no. 1. Springer, 2016. doi: <https://doi.org/10.1007/978-3-319-2969-3>.
- [19] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian informatics journal*, vol. 16, no. 3,

- pp. 261–273, 2015, doi: <https://doi.org/10.1016/j.eij.2015.06.005>.
- [20] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: a comprehensive review," *Comput. Soc. Netw.*, vol. 6, no. 1, pp. 1–23, 2019.
- [21] C. A. Gomez-Urbe and N. Hunt, "The netflix recommender system: Algorithms, business value, and innovation," *ACM Transactions on Management Information Systems (TMIS)*, vol. 6, no. 4, pp. 1–19, 2015, doi: <https://doi.org/10.1145/2843948>.
- [22] B. Lika, K. Kolomvatsos, and S. Hadjiefthymiades, "Facing the cold start problem in recommender systems," *Expert Syst. Appl.*, vol. 41, no. 4, pp. 2065–2073, 2014, doi: <https://doi.org/10.1016/j.eswa.2013.09.05>.
- [23] B. Smith and G. Linden, "Two decades of recommender systems at Amazon. com," *IEEE Internet Comput.*, vol. 21, no. 3, pp. 12–18, 2017, doi: <https://doi.org/10.1109/MIC.2017.72>.
- [24] M. Ghanem, W. Guezguez, and R. Ayachi, "A Multimodal DeBERTa-Based Recommender System for Low-Resource and Sparse Data Environments," doi: <https://doi.org/10.5220/001464630004052>
- [25] J. Zhou *et al.*, "Graph neural networks: A review of methods and applications," *AI open*, vol. 1, pp. 57–81, 2020, doi: <https://doi.org/10.1016/j.aiopen.2021.01.1>
- [26] S. Wu, F. Sun, W. Zhang, X. Xie, and B. Cui, "Graph neural networks in recommender systems: a survey," *ACM Comput. Surv.*, vol. 55, no. 5, pp. 1–37, 2022, doi: <https://doi.org/10.1145/3535101>.
- [27] Z. Zhang, P. Cui, and W. Zhu, "Deep learning on graphs: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 1, pp. 249–270, 2020, doi: <https://doi.org/10.1109/TKDE.2020.2981333>.
- [28] K. Mao, J. Zhu, X. Xiao, B. Lu, Z. Wang, and X. He, "UltraGCN: ultra simplification of graph convolutional networks for recommendation," in *Proceedings of the 30th ACM international conference on information & knowledge management*, 2021, pp. 1253–1262. doi: <https://doi.org/10.1145/3459637.3482291>.
- [29] J. Wu *et al.*, "Self-supervised graph learning for recommendation," in *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 2021, pp. 726–735. doi: <https://doi.org/10.1145/3404835.3462862>.
- [30] K. Oono and T. Suzuki, "Graph neural networks exponentially lose expressive power for node classification," *arXiv preprint arXiv:1905.10947*, 2019, doi: <https://doi.org/10.48550/arXiv.1905.10947>
- [31] L. Zhao and L. Akoglu, "Pairnorm: Tackling oversmoothing in gnns," *arXiv preprint arXiv:1909.12223*, 2019, doi: <https://doi.org/10.48550/arXiv.1909.12223>
- [32] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Transactions on Information Systems (TOIS)*, vol. 20, no. 4, pp. 422–446, 2002, doi: <https://doi.org/10.1145/582415.582418>.
- [33] D. Valcarce, A. Bellogin, J. Parapar, and P. Castells, "On the robustness and discriminative power of information retrieval metrics for top-N recommendation," in *Proceedings of the 12th ACM conference on recommender systems*, 2018, pp. 260–268. doi: <https://doi.org/10.1145/3240323.3240347>.