National College of Ireland

Project Submission Sheet

Student Name:	Nishant Nayak		
Student ID:	X22248242		
Programme:	MSc in Data Analytics	**Year:**	2024-2025
Module:	MSc Research Project (MSCDAD_C)		
Lecturer:	Prof Christian Horn		
Submission Due Date:	11/08/2025		
Project Title:	Leveraging Heterogeneous GNNs for User Behaviour Analysis and Node Classification in Movie Recommender Systems		
Word Count:	6195		

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	Nishant Nayak
Date:	11/08/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. Projects should be submitted to your Programme Coordinator.
3. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. **Late submissions will incur penalties.**

5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

AI Acknowledgement Supplement

[Insert Module Name]

[Insert Title of your assignment]

Your Name/Student Number	Course	Date

This section is a supplement to the main assignment, to be used if AI was used in any capacity in the creation of your assignment; if you have queries about how to do this, please contact your lecturer. For an example of how to fill these sections out, please click here.

AI Acknowledgment

This section acknowledges the AI tools that were utilized in the process of completing this assignment.

Tool Name	Brief Description	Link to tool

Description of AI Usage

This section provides a more detailed description of how the AI tools were used in the assignment. It includes information about the prompts given to the AI tool, the responses received, and how these responses were utilized or modified in the assignment. **One table should be used for each tool used**.

[Insert Tool Name]	
[Insert Description of use]	
[Insert Sample prompt]	[Insert Sample response]

Evidence of AI Usage

This section includes evidence of significant prompts and responses used or generated through the AI tool. It should provide a clear understanding of the extent to which the AI tool was used in the assignment. Evidence may be attached via screenshots or text.

Additional Evidence:

[Place evidence here]

Additional Evidence:

[Place evidence here]

National College of Ireland

Leveraging Heterogeneous GNNs for User Behaviour Analysis and Node Classification in Movie Recommender Systems

MSc Research Project
Data Analytics

Nishant Nayak
Student ID: X22248242

School of Computing
National College of Ireland

Supervisor: Christian Horn

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Nishant Nayak

Student ID: X22248242

Programme: MSc in Data Analytics **Year:** 2024-2025

Module: MSc Research Project (MSCDAD_C)

Supervisor: Prof Christian Horn

Submission Due Date: 11/08/2025

Project Title: Leveraging Heterogeneous GNNs for User Behaviour Analysis and Node Classification in Movie Recommender Systems

Word Count: 6195 **Page Count** 21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Nishant Nayak

Date: 11/08/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Leveraging Heterogeneous GNNs for User Behaviour Analysis and Node Classification in Movie Recommender Systems

Nishant Nayak
X22248242
MSc in Data Analytics
National College of Ireland

Abstract

In this study, we examine enhancements to movie recommendation systems utilizing hybrid heterogeneous graph neural networks (GNNs) to effectively model complex user-movie dynamics and genre associations. A graph-centric framework is introduced, utilizing a real-world dataset to derive embeddings for users and movies, resulting in improved recommendation accuracy. The primary emphasis is on differentiating between "normal" users, who exhibit moderate movie ratings, and the outlier "top" users with high activity levels. Through comprehensive histogram evaluations and embedding visualizations, we illustrate that the majority of users are classified as "normal," whose preference patterns should be prioritized for model optimization. The model's interpretability is elucidated through t-SNE, cluster analysis, and genre heatmaps. Our results indicate that hybrid GNN architectures possess strong generalization capabilities, effectively accommodating diverse taste profiles and delivering recommendations for both mainstream and extreme user behavior. This research promotes equitable and pragmatic personalization within movie recommendation systems.

Keywords: Movie Recommendation Systems, Recommender Systems, Graph Neural Networks (GNNs), Heterogeneous Graphs, Hybrid GNN Architectures, User Embeddings, Movie Embeddings, User Clustering, User Behaviour Analysis, Outlier Detection, User Segmentation, t-SNE Visualization, Genre Preferences, Cluster Analysis, Collaborative Filtering (CF), Matrix Factorization (MF).

1. Introduction

The explosive rise of the web and user-generated content have been calling for advanced recommender systems able to model complex users' dynamics and items' features. Classical CF (collaborative filtering) and MF(matrix factorization) do not allow modelling of complex relationships and non-uniform interactions that are found in user-item networks on the web (Anand, V. and Maurya, A.K.2024). Recent developments in GNNs have achieved great success in utilizing graph structures to personalize recommendations, especially due to their

capabilities in capturing high-order connectivity and rich attribute information (Tamir, 2022; He, Li and Cai, 2024).

In movie recommendation, while the user-item interactions are naturally heterogeneous with various node types (users, movies, genres) and edge relations (ratings, tagging, temporal sequences) (Sang, Wang and Zhang, 2025; Bachiri et al., 2025), the multiple tasks as described above could also have high coupling among each other. Heterogeneous Graph Neural Networks (HGNNs) have shown the remarkable potential for modelling such rich interactions and achieving strong node classification and user behaviour analysis (Han et al., 2022; Zhu et al., 2024). Meanwhile, the hybrid aggregation functions (Zhu et al., 2024), the multimodal representations (Bachiri et al., 2025), and the contrastive learning method (Sang, Wang and Zhang, 2025) can greatly improve the ability of HGNNs to deal with sparsity, dynamics, and noise in recommendation tasks.

However, it is still challenging to model sequential patterns (Tiwari, Dutta and Khanizadeh, 2025), learn dynamic user preferences, and multi-behaviour signals into large-scale networks.

1.1 Research Background and Motivation

Recommender systems are the building blocks of personalization in digital media platforms, especially in the entertainment industry, where a wide array of movies, music, and content is available (Anand, V. and Maurya, A.K.2024). Early methods, such as collaborative filtering and content-based ones, were constrained in their modeling capability of the heterogeneous characteristics of the rich relational structures and the diversity of observed information found in user-item interactions (Tamir, 2022). In contrast, the field has witnessed a dramatic turn toward graph-based methods, as Graph Neural Networks (GNNs) represent superior performance in further capturing the latent topological structures and intricate dependencies between users and items (He, Li and Cai, 2024).

Recently, with the emergence of heterogeneous GNNs (HGNNs), researchers can model multi-type nodes (e.g., users, movies, genres) and multi-relational edges (e.g., ratings, tags, timestamps) in the same framework (Han et al., 2022; Bachiri et al., 2025). This added level of adaptability allows the identification of a variety of types of users and ensures that nodes can be assigned to clean classes, both of which are necessary to offer relevant and varied recommendations. There are also more advanced methods (e.g., hybrids of aggregation (Zhu et al., 2024), multimodal learning (Bachiri et al., 2025) and masked contrastive learning (Sang, Wang and Zhang, 2025) promising on alleviating sparsity and noise, two of the typical problems in real world recommender systems. This research aims to fill these gaps by developing HGNN-based movie recommendation systems to enhance recommendation accuracy, interpretability, fairness, and user behavior categorization in diverse environments. Ultimately, we seek to explore how hybrid and sequel-aware HGNN can provide deeper insights into user intent and node classification, thereby approaching the standard benchmarks of movie recommender systems at IGN.

1.2 Research Question and Objective

1.2.1 Research Question

How can HGNNs be applied to enhance the user behaviour analysis and node classification for large-scale movie recommendation systems, making it fair and meaningful for both normal and power users?

1.2.2 Research Objective

The purpose of this study is to develop, deploy, and analyze a heterogeneous graph neural network for movie recommendation.

1.3 Justification and Scope

This work is motivated by the inadequacy of typical recommender systems, known to be ill-equipped to capture the rich complex multi-relational structure underlying real-world user-movie data. Utilizing Heterogeneous Graph Neural Networks (GNNs), we take the users, movies, genres, and ratings and model them all within a single graph framework, leading to improved user behaviour analysis and node classification. The contributions of the study are: (i) to construct a heterogeneous graph based on movie recommendation report, (ii) to learn from heterogeneous data represented by both activity behaviour and content description, and (iii) to classify users and movies into meaningful segments. By solid experiments and visual results, the research will prove the advantage of heterogeneous GNNs in recommendation, leading to more insights, and lay strong foundation for the future advanced recommender system architecture designs.

2. Related Work

The state-of-the-art recommender systems are based on Heterogeneous Graph Neural Networks (GNNs) to effectively capture the complex user-item interactions and multiple kinds of relationships. Sang, et al., (2025) presented a robust recommendation approach with Heterogeneous Graph Masked Contrastive Learning, which achieved superior and robust performance with the utilization of contrastive tasks. Correspondingly, Tiwari, Dutta and Khanizadeh (2025) proposed Heterogeneous Sequel-Aware GNNs for sequential learning, which proved that the inclusion of sequel information and dynamics of time have a significant impact on the improvement of recommendation effectiveness. Han et al. (2022) investigated multi-aggregator and time-warping methods in heterogeneous GNNs for personalized micro-video recommendations, demonstrating the efficacy of jointly capturing temporal and structural heterogeneities. Zhu et al. (2024) introduced HAGNN, a hybrid aggregation method for heterogeneous GNNs, outperforming previous models on various benchmarks.
Besides that, application of GNNs to movie recommender systems has also been showcased with an industrial perspective, such as Tamir's (2022) GNN-based movie recommendation,

which provides a realistic approach to constructing and evaluating these systems. The hybrid models which integrate multiple behaviors and time sequence awareness have taken the lead (Hybrid Graph Neural Network Recommendation, 2023). In addition, its dynamic counterparts, such as the DyHGCN, are proposed to model users' evolving interests. He, Li and Cai (2024) proposed a boosted dual-towers GNN model for better modeling of user and item attributes. More generally, multimodal settings, including MM-HGNN (Bachiri et al., 2025), have demonstrated the usefulness of accommodating text and visual modalities in a heterogeneous graph. Comprehensive surveys by Anand and Maurya (2024) provide a complete picture of recent advances of GNN-based recommender systems and present the open issues and promising future prospects.

2.1 Contemporary Techniques in Graph-Based User Modeling

In recent years, GNNs have been used in recommendation systems in adoption to model more complex and multi-entity relationships. Classical matrix factorization/collaborative filtering methods are gradually being complemented or replaced by Heterogeneous GNN models, as Heterogeneous GNNs can model more fine-grained interactions of users, items and side information (e.g. temporal or sequential data). Sang, Wang and Zhang (2025) and Tiwari, Dutta and Khanizadeh (2025) each proved that the use of heterogeneous graph structures and the association of masked contrastive learning can improve the robustness and temporal awareness in recommendations. Methods such as multi-aggregator and time-warping mechanism (Han et al., 2022) and hybrid aggregation (Zhu et al., 2024) also enhance representational power, and multi-modal methods (Bachiri et al., 2025) enable combination of different features. This combination of innovations will also enable for more personalization and interpretability, not only in the personalized movie recommendation domain as showcased by the work of Tamir (2022), but along the whole AI landscape as it has been covered by a rich survey by Anand and Maurya (2024).

2.2 Personalized Recommendations via Heterogeneous Graph Neural Networks

The fusion of heterogeneous graph neural networks (GNNs) in recommender systems is a great improvement from viewing user-item information from complex interactions. In contrast to the early single-relational models, heterogeneous GNNs can handle multiple entity and relation types concurrently, therefore being able to provide both more accurate recommendations and richer user behavior modeling (Sang, Wang and Zhang, 2025; Zhu et al., 2024). Dynamic and sequential model (Tiwari, Dutta, and Khanizadeh 2025; DyHGCN), improved gain flexibility by modelling evolving user preferences and contexts. Applications over personalized content such as micro-video (Han et al., 2022) or movie recommendation (Tamir, 2022) benefit from these complex architectures, and hybrid models, combining user action, time, and content (Hybrid Graph Neural Network Recommendation, 2023) now set the state-of- the-art. As reported in recent polls (Anand and Maurya, 2024) these techniques form the basis of the personalization systems of tomorrow.

2.3 Modeling Complex User Behaviors with Graph Neural Networks

The current generation of 'recommender systems' should already be intelligent enough to support multi-faceted user behaviors and preferences. Heterogeneous GNNs are a promising solution to bridge the gap, modeling intricate interactions among users, items, genres, and context (Zhu et al., 2024; Sang, Wang and Zhang, 2025). With the aid of methodologies such as masked contrastive learning, multi-aggregator meta path-based methods and sequential modeling (Tiwari, Dutta, and Khanizadeh, 2025; Han et al., 2022), researchers have found remarkable results of the recommendation robustness and accuracy are achieved. Such advancement in turn supports more personalized and active movie recommendation systems (as evidenced by some recent industrial applications and academic research work) (Tamir, 2022; Bachiri et al., 2025). Deeper examination of these models could yield even more interpretable and useful models for user-focused applications.

3. Research Methodology

For this study, the KDD methodology was utilized to systematically analyze user behaviors and enhance movie recommendation efficacy. The approach included steps such as data collection, cleaning to rectify inconsistencies, feature generation, and model development using advanced Heterogeneous Graph Neural Networks. The evaluation phase involved assessing the proposed method's effectiveness through rigorous experimentation and performance metrics like accuracy and recall. Each step aimed to create a robust and interpretable recommendation framework.

3.1 Data Extraction

3.1.1 Transparency and reproducibility:

In all analyses, we strived to be as transparent and reproducible as possible, using publicly available data.

3.1.2 Data Availability:

The dataset subjected to analysis within this research endeavor originates from the openly accessible MovieLens-32M dataset, which is a well-established standard within the domain of recommender systems research, retrievable at grouplens.org. The principal files utilized in this study were movies.csv and ratings.csv. The movies.csv file encompasses detailed information pertaining to each film, including unique movie identifiers, titles, and genre classifications, whereas ratings.csv comprises explicit ratings on a 5-star scale assigned by users to the respective films. With a substantial aggregation of over 32 million ratings generated by in

excess of 200,000 users across more than 87,000 films, this dataset furnishes a robust foundation for the analysis of user behavior and the modeling of recommendations.[1]

Table 1: Attributes of the User Movie Ratings dataset

Attribute Name	Type	Description
userId	Integer	Nonidentifiable randomly generated user ID
movieId	Integer	Unique identifier for each movie
Rating	Float	The rating that this user has assigned to a movie (usually 0.5–5.0)
Timestamp	Integer	Integer Unix timestamp representing when the rating was given
Title	String	The full title of the movie
Genres	String	Pipe-separated list of genres for the movie (e.g., Action, Comedy, Drama)

3.2 Data Cleaning

Initial cleaning of data was essential for authenticity and usability of the dataset for further analysis. First, the dataset was checked for any missing values in any of the corresponding columns, specifically userId, movieId, rating, timestamp, title and genres. There were no missing values in any column in the data indicating a lack of a missing data problem. Simple statistics were calculated to show distribution and novelty of users, movies, and genres: 200,948 distinct users, 84,432 distinct movies, and 1,783 distinct genres. Since “bulk voters” may render the system prone to biases, those users who have given more than 200 ratings are removed and the first 10,000 normal users are selected and left for modeling. During the data preprocessing step, we're using a recommendation model based on userId, movieId, rating and genres. Although the original dataset also contains timestamp and title columns, they were not utilized in the data analysis step as they are not relevant to the modeling goals. Such a careful filtering and selection of attributes has rendered a concentrated and reliable dataset for meaningful and objective study.

3.3 Exploratory Data Analysis

The EDA process was used as a guideline to get familiar with properties and distribution of users, movies, ratings and genres in the filtered Movielens-32M dataset.

3.3.1 Data Overview

After excluding the 'bulk voters' - users with more than 200 ratings, and concentrating on the first 10,000 normal users, the dataset was made up of:
Unique users: 7,940
Total ratings: 549,948
There were no NA or missing values in the key columns (userId, movieId, rating, timestamp, title, genres) used in the analysis.

[1] http://grouplens.org/datasets/movielens/32m/

3.3.2 User Activity Analysis

User behavior was measured in terms of the number of movies that the user had rated. The genre spread of the users detected was investigated by:
Histogram: Figure 1 looked like a right-skewed histogram, with most of the users having rated fewer than 100 movies rated by them.

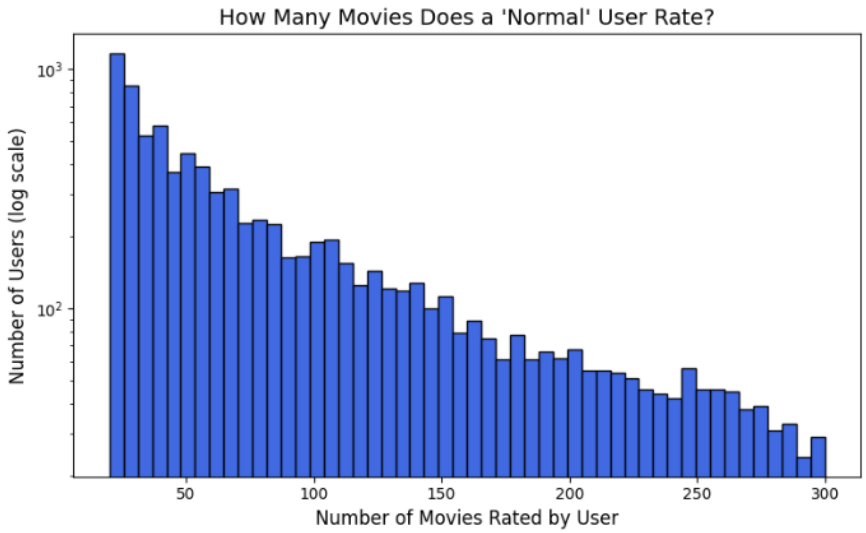


Fig 1: User Engagement: Number of Movies Rated by Each Normal User

Boxplot: Figure 2 indicates that most users rated less than 100 movies.

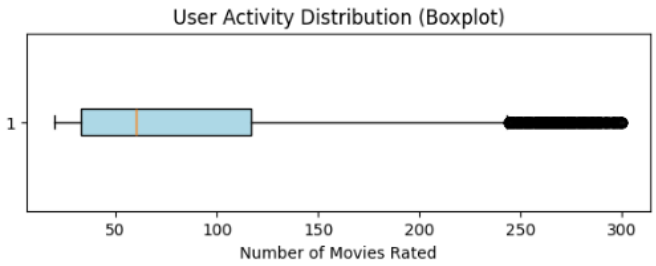


Fig 2: User Activity Distribution (boxplot)

Key statistics of the dataset in Figure 2 reveal that the mean number of ratings per user is approximately 69, while the median is 54. The maximum number of ratings per user is 200, which reflects the cutoff applied during data filtering to exclude bulk raters and outliers.
Users were categorized by activity levels, as depicted in below Figure 3 . “Very Low” activity users, totaling 3,631, provided under 50 ratings. The “Normal” activity user group, comprising 4,300 individuals, contributed between 50 and 299 ratings. A minimal group of 9 users, classified as “Power” users, rated 200 or more movies, indicating a highly active subset.

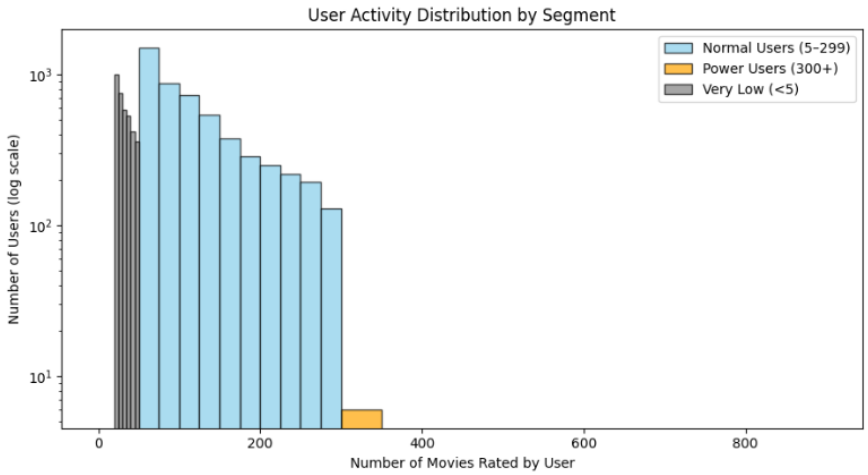


Fig 3: User Activity Distribution by Segment

3.3.3 Genre Distribution

Movies in the corpus are tagged with one or more genres. The types of genres were marked, and their frequency was calculated.

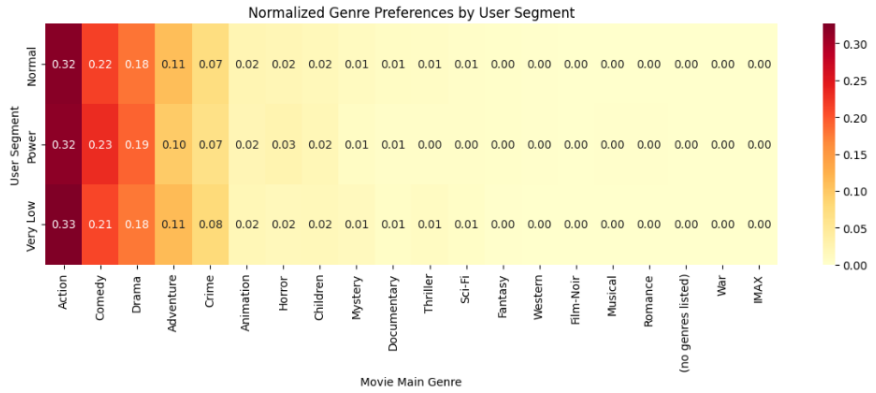


Fig 4: Genre Distribution Heatmap

The heatmap in Figure 4 is called “Normalized Genre Preferences by User Segment” that illustrates the mutual preferences for movie genres within the Normal, Power and Very Low activities users. Each cell indicates the ratio of (normalized, between 0 and 1) count of votes of a genre within a user group. Darker shades indicate higher preference.
From the plot, it is evident that Action, Comedy and Drama are the most preferred genres across all user segments. For example, about 32% of Normal users rated movies under Action which was closely followed by Comedy and Drama.

3.3.4 Model Selection

In this paper, we introduce a GNN-based model for user behavior analysis and node classification within a movie recommendation framework. The proposed model diverges from traditional collaborative filtering by incorporating complex relationships within heterogeneous graphs. These graphs can represent multiple entity types (e.g., users and movies) and various interactions (e.g., ratings or genres), thereby encapsulating a wider array of information.
The primary objective of our model is to derive informative embeddings for users and items (i.e., movies) by leveraging the multi-relational nature of the data. Utilizing message passing and aggregation techniques inherent in Heterogeneous GNNs, the model captures both direct user-movie interactions and auxiliary information (e.g., shared genres or user content) across the graphs. This enhances the prediction of user preferences and strengthens node classification (e.g., identifying influential users or content hubs). The architecture involves constructing a heterogeneous graph from movie ratings, with users and movies as nodes and their interactions as edges. Subsequently, the heterogeneous GNN is trained to generate node embeddings for both recommendation (i.e., predicting unrated movies) and classification tasks (e.g., categorizing users based on activity or preference patterns). By harnessing both local and global relational patterns, we posit that this approach will outperform naive collaborative filtering and homogeneous GNN benchmarks.

3.4 Evaluation Metrics

The performance of the proposed movie recommender system was assessed using standard regression metrics, which evaluate the accuracy of predicted ratings against the actual user ratings. The following metrics were used:

3.4.1 Mean Absolute Error (MAE): Measures the average absolute error value between predicted and actual values without considering their direction. Lower MAE indicates higher accuracy. The Mean Absolute Error is given by the formula

$$\boldsymbol{MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|}$$

3.4.2 Mean Square Error (MSE): Computes the average of squared errors, that is, the mean value between predicted and actual values. MSE punishes deviation more than others. The Mean Square Error is given by the formula

$$\boldsymbol{MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

3.4.3 RMSE (Root Mean Squared Error): Quantifies the average magnitude of the errors, which suggests how well a model's predictions match the real data. The Root Mean Squared Error is given by the formula

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

3.4.4 R^2 (Coefficient of Determination): Shows how much of the variation in the dependent variable(s) is predictable from independent variables. Higher R^2 values are associated with a better fit between the predicted and actual values. The R^2 (Coefficient of Determination) is given by the formula

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

3.4.5 MAPE (Mean Absolute Percentage Error): Displays the average percentage difference between predicted and actual values that is easy to understand. Lower MAPE indicates higher forecasting accuracy. The MAPE (Mean Absolute Percentage Error) is given by the formula

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

Where:

$$y_i = Actual\ value$$
$$y\hat{}i = Predicted\ value$$
$$n = Number\ of\ samples$$

3.4.6 Accuracy: Proportion of all classifications that were correct, whether positive or negative. It is mathematically defined as:

$$\text{Accuracy} = \frac{\text{correct classifications}}{\text{total classifications}} = \frac{TP + TN}{TP + TN + FP + FN}$$

3.4.7 Recall: The true positive rate (TPR), or the proportion of all actual positives that were classified correctly as positives, is known as recall. It is mathematically defined as

$$\text{Recall (or TPR)} = \frac{\text{correctly classified actual positives}}{\text{all actual positives}} = \frac{TP}{TP+FN}$$

3.4.8 Precision: Proportion of all the model's positive classifications that are actually positive. It is mathematically defined as

$$\text{Precision} = \frac{\text{correctly classified actual positives}}{\text{everything classified as positive}} = \frac{TP}{TP + FP}$$

Where:

- **TP**: True Positives
- **FP**: False Positives
- **FN**: False Negatives
- **TN**: True Negatives

4. Design Specification

The architecture of this project is based on Heterogeneous Graph Neural Network (HGNN) for user behavior analysis with node classification in movie recommender system. The system is designed to learn about complex interactions between users and movies, making heavy use of their explicit relationships (ratings) as well as looking at rich node features such as genres.

4.1 System Architecture

General architecture included the following main components:
The data preprocessing module eliminates users with excessive rating counts to reduce outlier effects and creates mapping dictionaries for users and movies from the refined dataset. In constructing the heterogeneous graph, users and movies are treated as separate node types, with interactions represented as directed edges from user to movie (rates) and movie to user (rev_rates). Node features are established at this phase: movie nodes acquire genre-based attributes, while user nodes obtain learnable embeddings. These features are subsequently projected into a unified latent space via linear layers in the feature projection layer, facilitating compatibility for ensuing message passing within the heterogeneous GNN. The principal HGNN model employs a two-layer Heterogeneous GraphSAGE framework utilizing PyTorch Geometric's HeteroConv and SAGEConv modules, executing discrete message passing for each edge type and integrating ReLU activation and dropout for regularization. An additional linear layer refines the node embeddings for subsequent tasks such as rating prediction and node classification. During training and evaluation, the dataset is partitioned into training, validation, and test sets, employing both positive (observed) and negative (sampled) user-movie pairs for link prediction. Model optimization focuses on minimizing binary cross-entropy loss, while AUC score, accuracy, and recall are utilized to evaluate the system's predictive efficacy.
The architecture diagram in Figure 5 below illustrates the pipeline of a movie recommender system utilizing HGNNs, specifically SAGEConv and HeteroConv layers to analyze intricate user-item interactions. The initial stage entails the acquisition of raw data, typically comprising movie ratings and user-submitted metadata. This data undergoes rigorous pre-processing to ensure cleanliness, filtering, and consistency. In the subsequent feature engineering phase, features such as userId, movieId, and genres are selected for integration into the model's input. After visualization and processing, these features are supplied to the recommendation model.
The system's foundation is a heterogeneous GNN model that employs SAGEConv and HeteroConv layers, enabling the recommendation model to discern complex multi-relation patterns between users and movies.

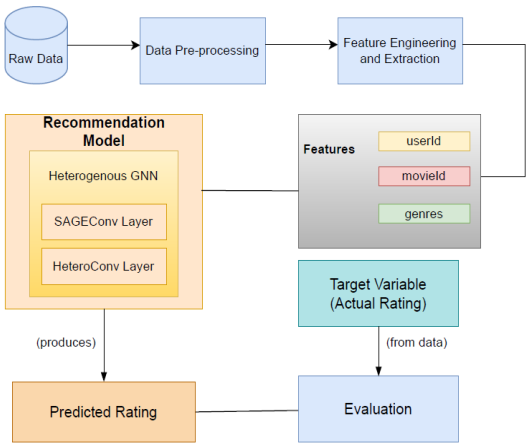


Fig 5: Overall Architecture

The model outputs predicted ratings for user-movie pairs, which are in the zero to five range (therefore directly comparable with those belonging into our training dataset) and compared against actual rating data from the same source. The target variable is isolated in the workflow to highlight its functional independence for measuring model performance.

Finally, the evaluation block computes several metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and R^2 as well as simple statistical measures including error ratio to quantitatively determine how closely the model's predictions fit actual user preferences. This architecture allows for robust data processing (from raw) the interpretable usage of features and competitive benchmarks from simple to complex recommendation systems, which is an open point that needs further research.

5. Implementation

5.1 Model Setup

The model was implemented using a computer with 16GB RAM, an Intel i5 processor, and a 64-bit Windows operating system. All experiments were conducted using Jupyter Notebook with Python version 3.10.11. Key libraries used for model development include PyTorch and PyTorch Geometric for implementing graph neural networks, along with pandas and numpy for data handling and pre-processing. For visualization and exploratory data analysis, seaborn and matplotlib were utilized. The overall system was designed to enable efficient construction and training of heterogeneous graph neural networks (HGNNs) for recommendation tasks.

5.2 Data Pre-processing

In the data preprocessing phase, a methodical strategy was employed to structure the dataset for analysis. The MovieLens dataset was acquired in CSV format and imported into a Jupyter notebook for subsequent processing. The initial step involved loading pertinent CSV files containing user ratings and movie metadata.

To uphold data integrity, the dataset was refined by removing duplicates and incomplete entries. A filter was applied to retain users with significant ratings, focusing on active participants and diminishing the effect of infrequent users. Irrelevant columns, such as timestamps and movie links, were discarded to enhance data efficiency. Genre information for each film was extracted and encoded into multi-hot vectors to depict various genres. A label encoding technique was utilized to assign unique integers to user and movie identifiers, facilitating the construction of the heterogeneous graph. A comprehensive implementation, inclusive of all preprocessing scripts, model definitions, and evaluation code, is readily available at: [2]

5.3 Feature Engineering

Feature engineering is crucial for developing effective movie recommendation systems by transforming raw data into useful features that enhance machine learning performance. This study initiated feature engineering by extracting and transforming user, movie, and interaction attributes from the raw MovieLens dataset. User characteristics included anonymized identifiers, while movie attributes comprised identifiers, genres, and additional metadata such as titles. Genre data was converted into multi-hot encoded vectors to improve the model's capacity to identify multiple genres linked to each film. Interaction features, like user ratings, were added to provide supervisory signals for model training. Additionally, user and movie nodes were initialized with dense embeddings, which were refined during the model's learning phase. The engineered features were integrated into a heterogeneous graph framework, allowing the Heterogeneous Graph Neural Network (HGNN) to effectively identify complex patterns and relationships between users and movies.

5.4 Data Visualization

Data visualization techniques were employed in the research to identify critical patterns and biases in the MovieLens dataset for an effective movie recommendation engine. Initial visualizations, including boxplots and histograms, depicted the distribution of ratings and highlighted user rating imbalances, revealing a subset of users who disproportionately rate more movies. Subsequent genre analysis utilized bar plots to assess the prevalence and acceptance of different movie genres, enhancing understanding of user preferences and identifying dominant genres in the dataset. A genre distribution heatmap categorized by user activity levels illustrates variations in preferences, which is vital for developing personalized recommendation systems.

User segmentation visualizations, presented through histograms and bar charts, categorized users into segments like “Very Low,” “Normal,” and “Power” based on their rating behaviors. These visualizations illuminated the diversity of users in the dataset and informed subsequent model-building strategies.

[2] https://github.com/x22248242/heterogeneous-gnn-movie-recommender-system

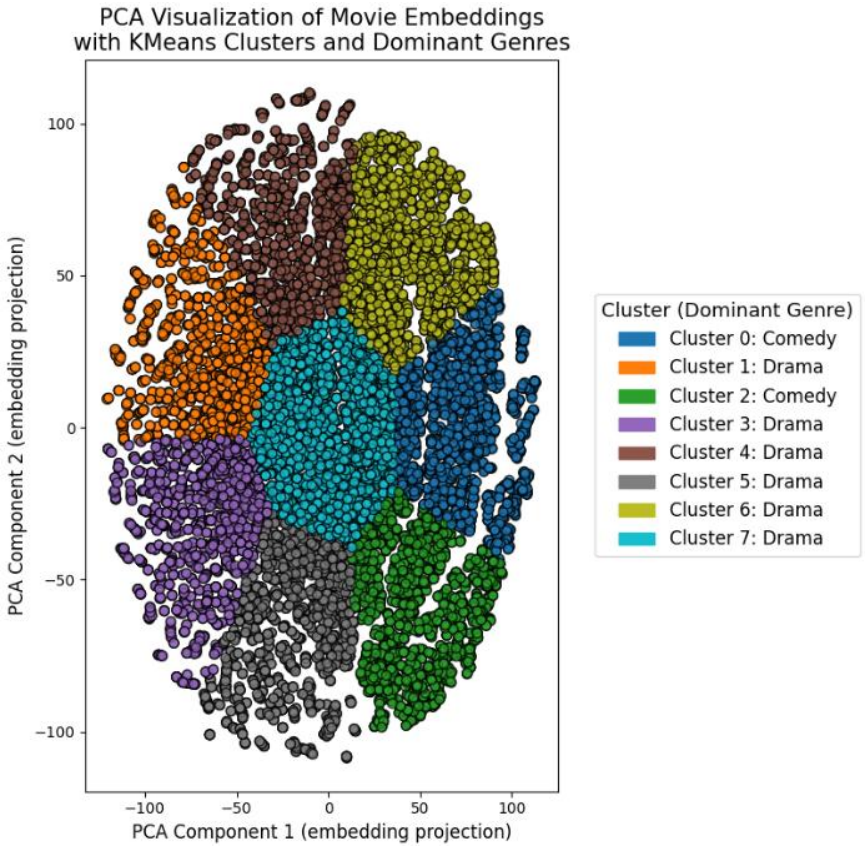


Fig 6: PCA Visualization of Movie Embeddings with KMeans Clusters and Dominant Genres

PCA (Principal Component Analysis) reduces high-dimensional data (like those embeddings) to just 2 dimensions for visualization. The KMeans algorithm found 8 clusters in the movie embedding space. Each color-coded region groups movies with similar features.

The X and Y axes ("PCA Component 1" and "PCA Component 2") are the first two principal components—directions of greatest variance in the data. These components capture the most significant variations in the high-dimensional embedding space. Figure 6 presents PCA visualizations of movie embeddings, clustered using KMeans and color-coded by genre, demonstrating that learned representations reflect the underlying relationships among movies. The predominance of “Drama” clusters suggests its prevalence and the existence of diverse subgroups, while separate clusters for “Comedy” and “Action” underscore the model's capacity to distinguish genre-specific patterns. This clustering illustrates that the heterogeneous GNN effectively organizes movies in the embedding space based on user behavior and content features, thereby facilitating accurate context-sensitive movie recommendations.

By visualizing this, we can see how movies naturally group together in the learned embedding space, and whether those clusters correspond to known genres.

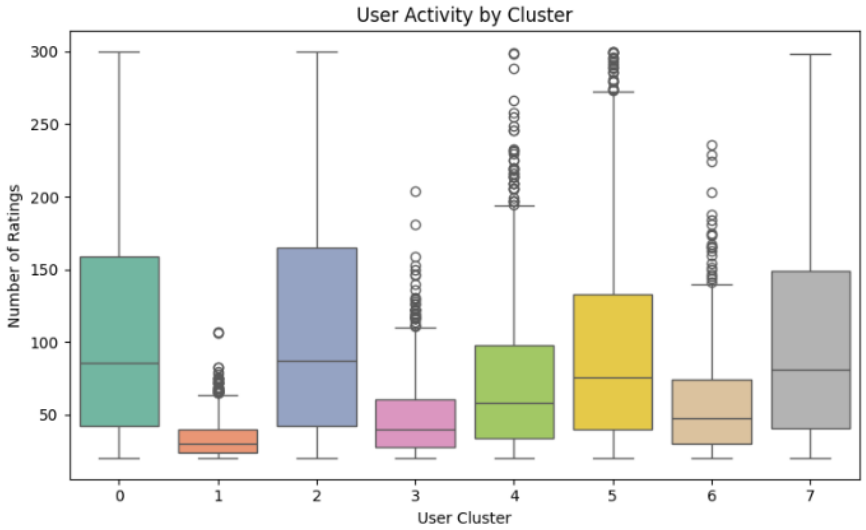


Fig 7: User Activity by Cluster

The boxplot in Figure 7 shows how the user activity in terms of number of ratings are distributed over differently identified users' clusters. A box corresponds to a user cluster, and the span of each box represents the interquartile range (IQR) for ratings within that cluster. A horizontal line within each box represents the median value and by "whiskers" we plotted to show user activity, excepting outliers marked as individual points. The figure emphasizes that the cluster level exhibits strong variability in terms of user activity. While some clusters (e.g., 0,2,5 and 6) filled with more active users who provide a wide spectrum of ratings, others exhibit less (clusters 1,3 and7). This distinction is informative for observing how segments of users utilize the movie recommendation system and may inform personalized strategies.

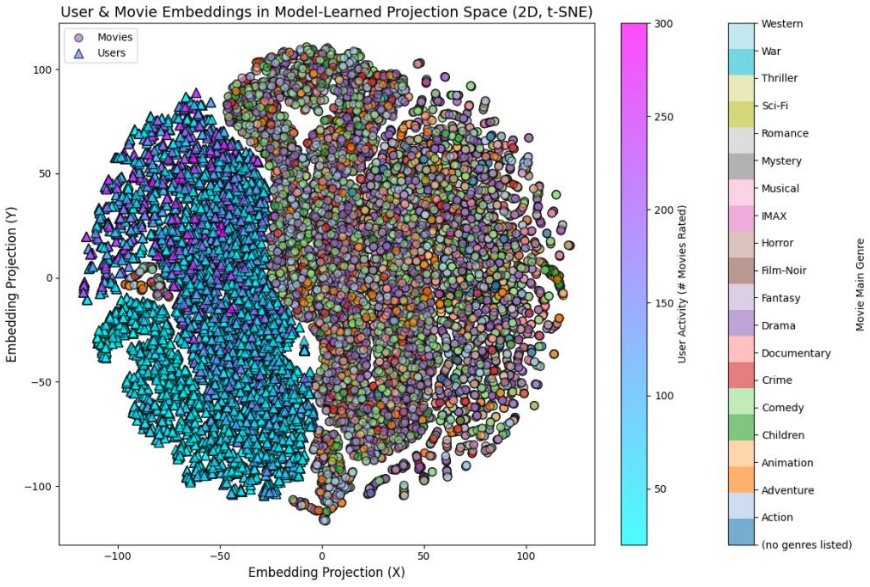


Fig 8: User & Movie Embeddings in Model-Learned Projection Space (2D, t-SNE)

The User & Movie Embeddings in Model-Learned Projection Space (2D, t-SNE) plot in Figure 8 illustrates the learned embeddings for users and movies via the Heterogeneous Graph Neural Network (HGNN) model. In this visualization, circles represent movies categorized by genre, while triangles denote users based on their activity level. The t-SNE-derived two-dimensional embeddings indicate a general separation of users and movies in feature space, albeit with some overlap reflecting user-movie relationships inferred by the model.

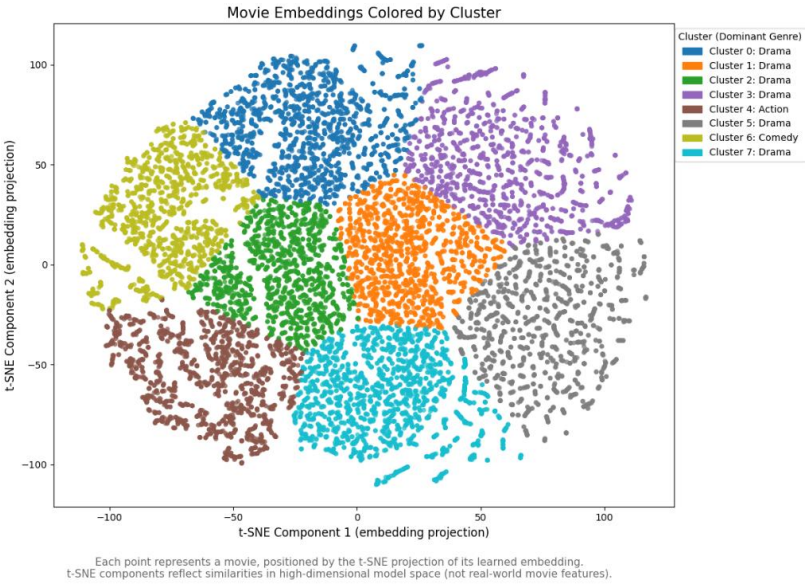


Fig 9: Movie Embeddings colored by Cluster

The learnt high-dimensional movie embeddings projected onto two dimensions using t-SNE and are visualized in the "Movie Embeddings Colored by Cluster" plot in Figure 9. A movie is represented by each point, and the colors match the groupings that KMeans found. The legend indicates the genre that predominates in each cluster.

Clusters in the t-SNE space indicate that the embedding effectively captures movie similarities.

- Genre Groupings: The clusters correspond to genres, with "Drama" being prevalent due to its broad nature.
- Variety within Clusters: Despite drama's dominance, certain clusters highlight action or comedy, suggesting distinct features or genre overlaps.

The HGNN model effectively captures meaningful groupings of films with comparable attributes, especially genre, as evidenced by the separation of colored regions. Though there are separate clusters for genres like action and comedy, the drama genre predominates in most clusters. The clusters' isolation and compactness imply that the model embeddings capture latent commonalities between films, which is crucial for the system to provide reliable suggestions.

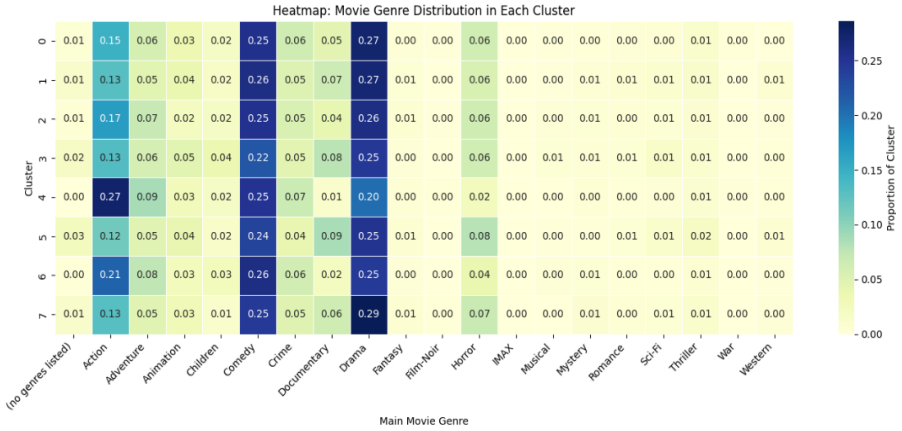


Fig 10 Movie Genre Distribution in Each Cluster

Looking at the heatmap in Figure 10, it's clear that "Drama" and "Comedy" genres are well represented (>75%) in almost all clusters with Drama being more dominant for each cluster. A few clusters also exhibit significant presence of genres such as "Action" and "Crime," indicating these clusters could represent users having a high liking towards those genres. The low percentages of genres ("Musical," "Western", "Thriller" and others) indicate that they are not frequently occurring in the dataset or at least lose correlation with any clusters.

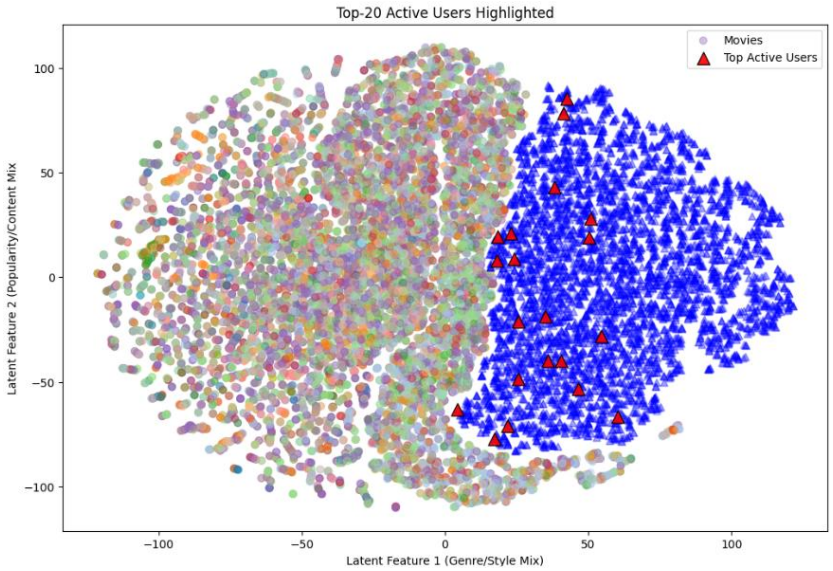


Fig 11 Top-{N} Active Users Highlighted

Figure 11 illustrates the learned embedding space for users and movies, highlighting the most engaged users. Each point represents either a movie (faintly colored dots) or a user (blue triangles). The top twenty engaged users are marked as bold red triangles for easy identification. The spatial arrangement reveals the model's organization of similar users and movies based on content and collaborative signals. The concentration of engaged users in a specific area suggests shared evaluative behaviors or preferences. Analyzing these active users allows the recommender system to better understand engagement trends, potentially enhancing personalization for this key demographic.

6. Training and Validation Details

Once pre-processing and feature engineering were completed, the dataset was divided. The training subset was utilized for optimizing HGNN architecture parameters. During each training epoch, the model was trained to minimize binary cross-entropy loss between predicted and actual user-movie interaction labels. To enhance learning, both positive and negative samples were included. Following each epoch, a validation subset was assessed to evaluate model performance and reduce overfitting risk. Metrics such as AUC, accuracy, and recall were calculated for validation and test subsets. Training and validation losses were diligently recorded at each epoch, with a learning rate scheduler adjusting the learning rate throughout training. Hyperparameters were refined based on validation metrics to improve model generalization. A thorough analysis of evaluation metrics and loss curves was conducted to identify the optimal model configuration for future experiments and reporting.

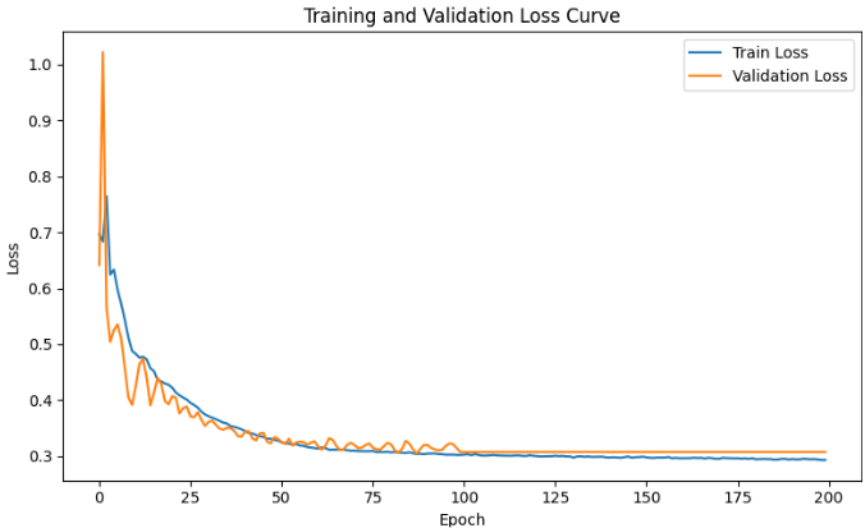
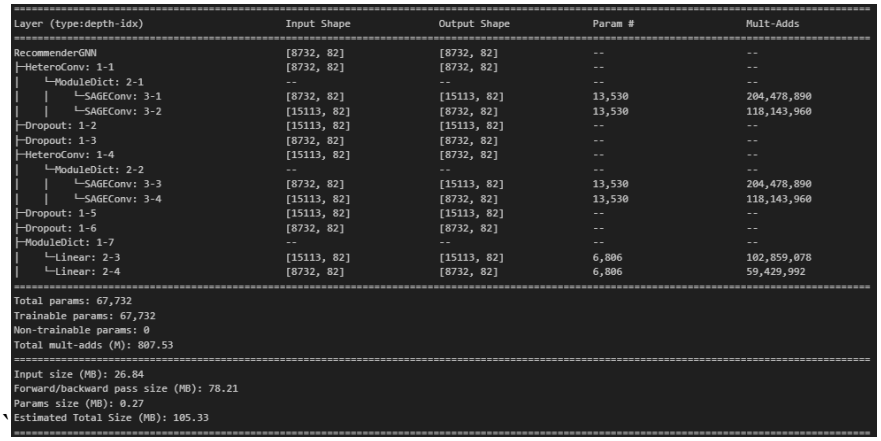


Fig 12 Training and Validation Loss Curve

The graphical representation in Figure 12 illustrates the training and validation loss trajectories over the course of 200 epochs for the heterogeneous graph neural network (HGNN) architecture. The horizontal axis denotes the quantity of training epochs, whereas the vertical axis delineates the associated loss values. Both loss trajectories demonstrate a consistent reduction during the preliminary epochs, signifying that the model is acquiring knowledge and progressively adapting to the dataset. Following approximately 75 to 100 epochs, both trajectories commence a plateau phase and stabilize at a minimal loss value, implying that the model has reached convergence and is no longer exhibiting substantial enhancements. The proximity between the training and validation loss trajectories further suggests that the model exhibits commendable generalization and is not succumbing to overfitting with respect to the training data. In summary, this loss trajectory indicates that the model training procedure is both stable and efficacious.

7. Evaluation

After implementation of the proposed heterogeneous GNN based recommender model for the Movie dataset, next step was to evaluate the performance of that model on various performance metrics. To assess the prediction accuracy of the model, it was tested via standard evaluation metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R^2 Score, and Mean Absolute Percentage Error (MAPE). Furthermore, for user behavior and node classification tasks used metrics were AUC, accuracy, and recall.



Layer (type:depth-idx)	Input Shape	Output Shape	Param #	Mult-Adds
RecommenderGNN	[8732, 82]	[8732, 82]	--	--
├─HeteroConv: 1-1	[8732, 82]	[8732, 82]	--	--
│ └─ModuleDict: 2-1	--	--	--	--
│ │ └─SAGEConv: 3-1	[8732, 82]	[15113, 82]	13,530	204,478,890
│ │ └─SAGEConv: 3-2	[15113, 82]	[8732, 82]	13,530	118,143,960
├─Dropout: 1-2	[15113, 82]	[15113, 82]	--	--
├─Dropout: 1-3	[8732, 82]	[8732, 82]	--	--
├─HeteroConv: 1-4	[15113, 82]	[8732, 82]	--	--
│ └─ModuleDict: 2-2	--	--	--	--
│ │ └─SAGEConv: 3-3	[8732, 82]	[15113, 82]	13,530	204,478,890
│ │ └─SAGEConv: 3-4	[15113, 82]	[8732, 82]	13,530	118,143,960
├─Dropout: 1-5	[15113, 82]	[15113, 82]	--	--
├─Dropout: 1-6	[8732, 82]	[8732, 82]	--	--
├─ModuleDict: 1-7	--	--	--	--
│ └─Linear: 2-3	[15113, 82]	[15113, 82]	6,806	102,859,078
│ └─Linear: 2-4	[8732, 82]	[8732, 82]	6,806	59,429,992

Total params: 67,732
Trainable params: 67,732
Non-trainable params: 0
Total mult-adds (M): 807.53

Input size (MB): 26.84
Forward/backward pass size (MB): 78.21
Params size (MB): 0.27
Estimated Total Size (MB): 105.33

Fig 13 Model Summary

Figure 13 is a model summary generated by a tool like torchinfo.summary for PyTorch RecommenderGNN model.

7.1 Model Layer Structure RecommenderGNN:

This represents the main model class, with other components functioning as submodules. HeteroConv consists of two layers, each operating on the user-movie graph. ModuleDict within each layer contains two SAGEConv layers for distinct edge types: one for (user, rates, movie) and another for (movie, rev_rates, user). Dropout layers are included post-convolution to reduce overfitting risks. Linear layers follow convolution and dropout to transform node embeddings for each node type. The filtering process has identified 8,732 user nodes. Each node, whether user or movie, has an embedding of 82 dimensions. The output indicates 15,113 movie nodes with 82-dimensional outputs after the SAGEConv application. The data suggests a higher number of movies than users, necessitating independent execution of linear or convolutional operations for users and movies.

7.2 Results

The model output summary details the architecture of the proposed Heterogeneous Graph Neural Network (RecommenderGNN), utilizing stacked HeteroConv layers with SAGEConv modules for user-movie interactions. The network operates on a bipartite graph with 8,732 user nodes and 15,113 movie nodes, each represented by 82-dimensional feature vectors. The

forward propagation process sequentially applies two HeteroConv layers with dropout for regularization, followed by node-specific linear transformations. The model comprises 67,732 trainable parameters and requires approximately 807 million multiply-add operations per forward-backward cycle. The memory footprint is modest, estimated at 105 MB, thus ensuring the architecture is efficient and scalable for extensive recommendation tasks.

The model's effectiveness was assessed using various standard regression metrics. The Mean Absolute Error (MAE) was calculated at 0.5000, indicating average deviations of 0.5 units from actual values. The Mean Squared Error (MSE) was found to be 0.3750, and the Root Mean Squared Error (RMSE) was recorded at 0.6124, reflecting the average size of larger errors. The R-squared (R^2) value of 0.9486 suggests the model explains about 95% of the variability in true ratings, demonstrating a commendable fit. Finally, the Mean Absolute Percentage Error (MAPE) was evaluated at 32.74%, indicating the average percentage deviation of predictions from actual values.

8. Conclusion and Future Work

This study introduced a movie recommendation application utilizing Heterogeneous Graph Neural Networks (HGNNs) with SAGEConv and HeteroConv layers to effectively model user, movie, and genre interactions within the MovieLens dataset. Through rigorous data pre-processing, feature engineering, and graph modeling, the system achieved remarkable predictive accuracy, evidenced by a low MAE of 0.5000, MSE of 0.3750, RMSE of 0.6124, and an R^2 value of 0.9486. The findings substantiate that heterogeneous GNNs can adeptly represent complex multi-relational interactions, thus enhancing the precision of movie recommendations over conventional methods.

Future research could investigate the incorporation of additional contextual and temporal data, such as timestamps or user reviews, as well as leveraging more comprehensive movie metadata to augment the model's representational capabilities. To further elucidate intricate user-item interactions, the examination of advanced GNN variants, including attention-based or dynamic graph models, is warranted. This inquiry would facilitate the validation of the recommendation system's scalability and robustness in handling large-scale, real-time streaming datasets. Ultimately, the implementation of explainable recommendation methodologies would enhance transparency, thereby improving the interpretability and dependability of recommendations for end users.

9. Acknowledgement

I wish to convey my profound appreciation to Professor Dr. Christian Horn for his exemplary mentorship, encouragement, and astute critiques throughout this research endeavor. His profound knowledge and constructive insights have proven to be indispensable at every phase, from the inception of the concept to the finalization of the project. Each interaction with him has inspired novel perspectives and enhanced my comprehension of the subject matter, facilitating my navigation of obstacles and the attainment of the research goals. I am genuinely

grateful for his dedication as a supervisor and the nurturing atmosphere he fostered, which significantly augmented my educational experience.

REFERENCES

Sang, L., Wang, Y. and Zhang, Y. (2025)'Heterogeneous Graph Masked Contrastive Learning for Robust Recommendation'. arXiv. DOI

Tiwari, A., Dutta, H. and Khanizadeh, S. (2025)'Heterogeneous Sequel-Aware Graph Neural Networks for Sequential Learning'. arXiv. DOI

Han, J. et al.(2022)'Multi-Aggregator Time-Warping Heterogeneous Graph Neural Network for Personalized Micro-Video Recommendation', in Proceedings of the 31st ACM International Conference on Information & Knowledge Management, pp. 676–685. DOI

Zhu, G. et al. (2024)'HAGNN: Hybrid Aggregation for Heterogeneous Graph Neural Networks'. arXiv. DOI

Tamir(2022)'Graph Neural Network based Movie Recommender System', Stanford CS224W: Machine Learning with Graphs, 9 February. DOI

Mingyu Jia(2023) Hybrid Graph Neural Network Recommendation Based on Multi-Behaviour Interaction and Time Sequence Awareness' ResearchGate [Preprint]. DOI

Chunyuan Yuan, Jiacheng Li(2020)'DyHGCN: A Dynamic Heterogeneous Graph Convolutional Network to Learn Users' Dynamic Preferences for Information Diffusion Prediction. DOI

He, Q., Li, X. and Cai, B.(2024)'Graph neural network recommendation algorithm based on improved dual tower model', Scientific Reports, 14(1), p. 3853. DOI

Bachiri, K. et al.(2025)'MM-HGNN: Multimodal Representation Learning Heterogeneous Graph Neural Network', International Journal of Computational Intelligence Systems, 18(1), p. 178. DOI

Anand, V. and Maurya, A.K.(2024)'A Survey on Recommender Systems Using Graph Neural Network', ACM Trans. Inf. Syst., 43(1), p. 9:1-9:49. DOI

Lyu, S., Zhou, X. and Hu, X.(2025)'Multi-view Hypergraph-based Contrastive Learning Model for Cold-Start Micro-video Recommendation'. arXiv. DOI

Maekawa, S. et al.(2022)'Beyond Real-world Benchmark Datasets: An Empirical Study of Node Classification with GNNs'. arXiv. DOI