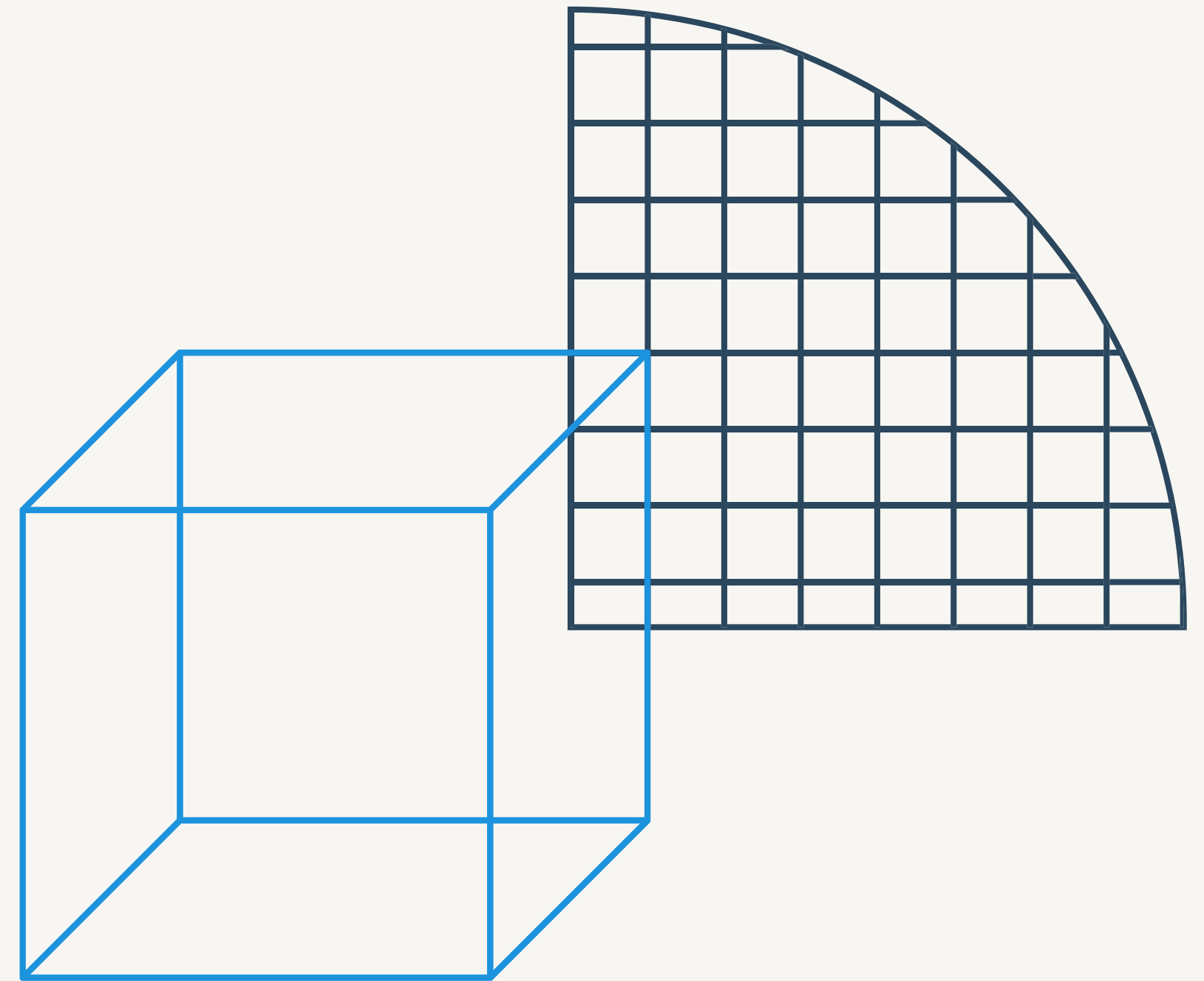


DEEP LEARNING- BASED STEAM STORE GAME RECOMMENDER SYSTEM

MSCA 31009 Machine Learning & Predictive Analytics
Final Project



SHIJIA HUANG



Agenda

- Problem Statement
- Data & Model Assumptions
- Exploratory Data Analysis
- Feature Engineering & Transformations
- Model Exploration
- Model Selection
- Hyperparameter Tuning
- Model Evaluation & Results
- Learnings & Future Work



Problem Statement



BACKGROUND AND CONTEXT

- **Steam Store:** A digital distribution platform for video games, offering a vast collection of games across various genres.
- **Features:** Includes community forums, game reviews, and multiplayer capabilities, enhancing user engagement and interaction.
- **User Base:** Boasts over **120 million** active users and a library of more than **30,000** games as of September 2021, providing a wide range of options for gamers to explore worldwide.

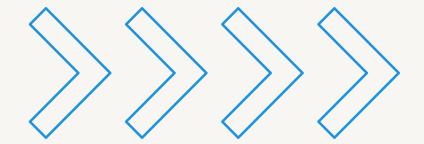
PROBLEMS AND QUESTIONS

- **Inadequate game recommender system:** The current Steam Store lacks an efficient recommendation system, resulting in users facing difficulties in discovering games that match their preferences.
- How can we leverage **user preferences** and **historical data** to create an accurate and personalized game recommender system on Steam Store?

RESOLUTION AND OBJECTIVE

- Develop and implement a **deep learning-based game recommender system** that utilizes advanced algorithms to analyze user preferences, historical data, and game features for accurate and personalized recommendations.
-

Data & Model Assumptions



+ DATA

- **Games:** Basic information about games, including ratings, pricing, and supported platforms (~50K).
- **Games Metadata:** Additional details about games such as descriptions and tags (~50K).
- **Users:** Includes information about user profiles, including the number of purchased products and published reviews (~7M).
- **Reviews:** Relationship between games and users, capturing user reviews and recommendations for specific products (~14M).

+ ASSUMPTIONS

- **Accurate data representation:** The model assumes the provided data accurately represent user preferences and game information.
- **Reliable user feedback:** Users' purchasing decisions and reviews are assumed to reflect their genuine recommendations for experienced games.
- **Effective user-game modeling:** The model assumes successful capture and utilization of user-game relationships for accurate and personalized recommendations.

+ LIMITATIONS

- **Incomplete data capture:** The available data may not fully capture the range of user preferences and game characteristics.
- **Limitations of historical user reviews:** Historical reviews may not accurately reflect current user preferences or trends.
- **Assumption of data influence:** The model assumes user preferences are solely influenced by the provided data, potentially overlooking external factors and individual context.

Exploratory Data Analysis

Games and Games Metadata

- **Platform Distribution**

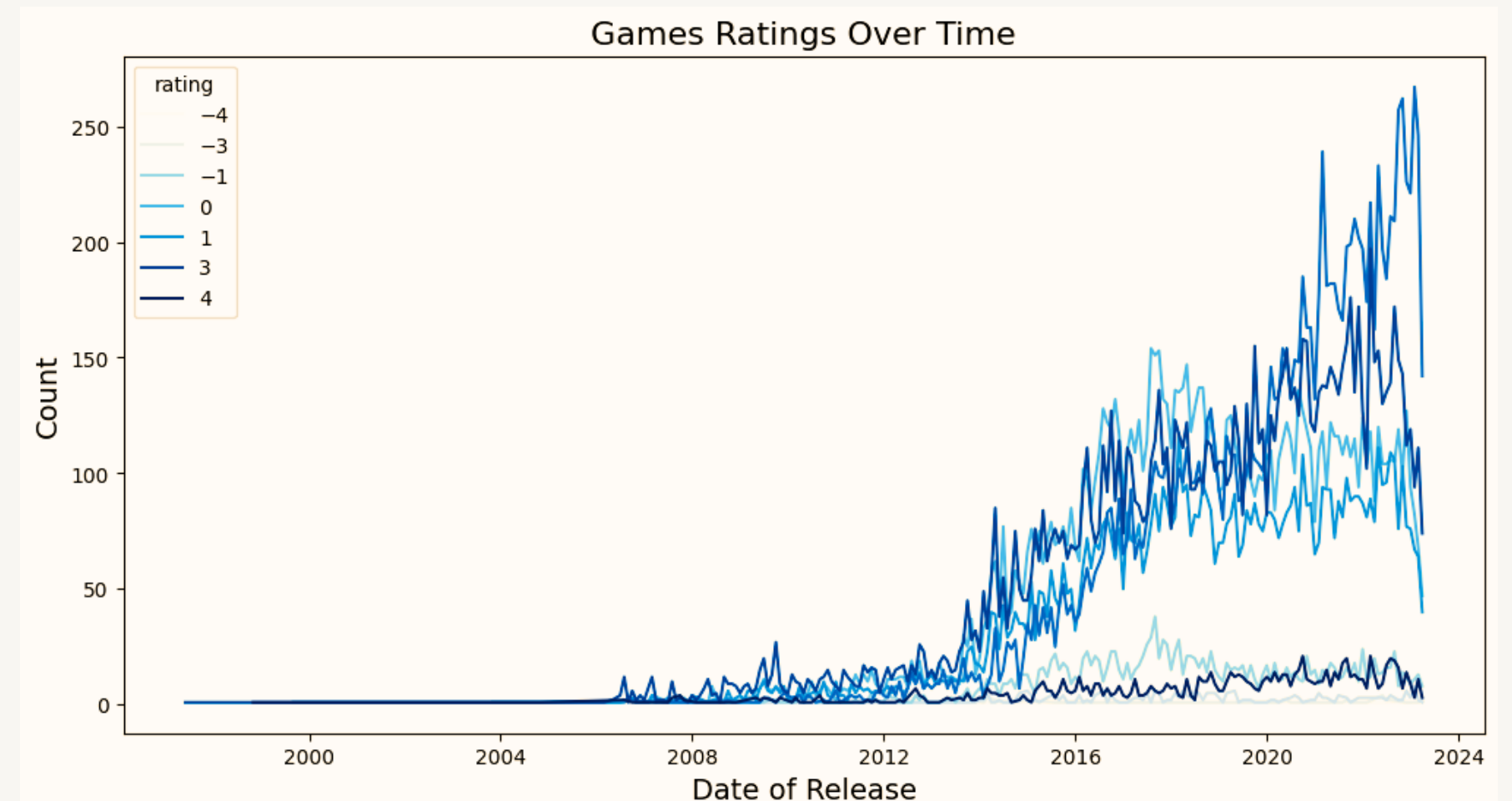
- The majority of games support Windows, followed by Mac and Linux.
- **Insight:** Consider platform preferences when designing the recommender system for personalized recommendations.

- **Popular Game Tags**

- Top tags include anime, action, and horror.
- **Insight:** Incorporate user preferences for these popular tags to improve recommendation accuracy.

- **Game Rating Trends**

- Ratings show overall positive sentiment and are increasing over time.
- **Insight:** Prioritize positive-rated games and leverage user feedback for more reliable recommendations.



Exploratory Data Analysis

Users and Reviews

- **Purchased Products vs. Reviews Published**

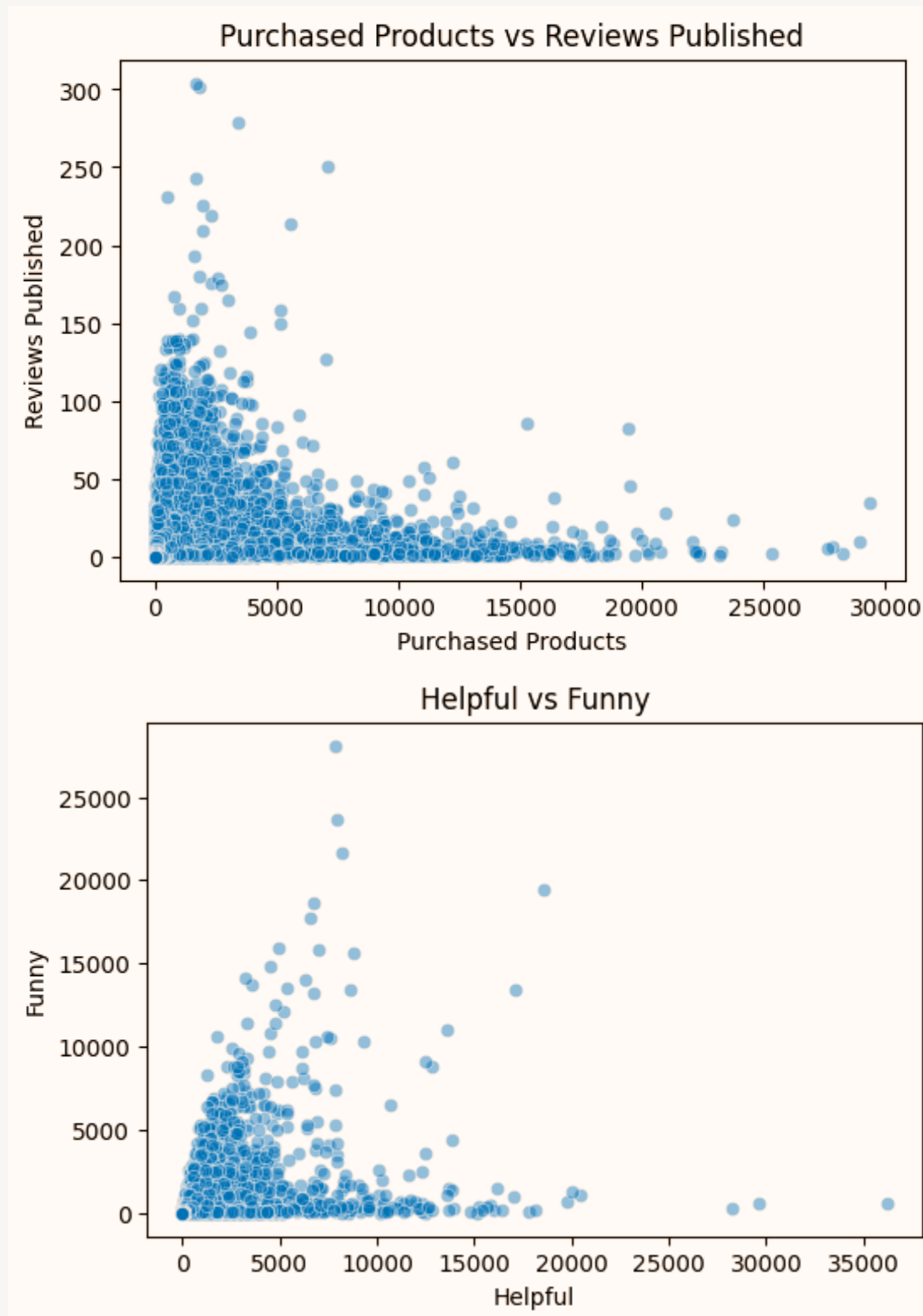
- A minor positive correlation was observed between purchases and user reviews.
- **Insight:** Users who purchase more products tend to be more active in providing reviews, which can be considered for personalized recommendations.

- **Helpful vs. Funny Ratings**

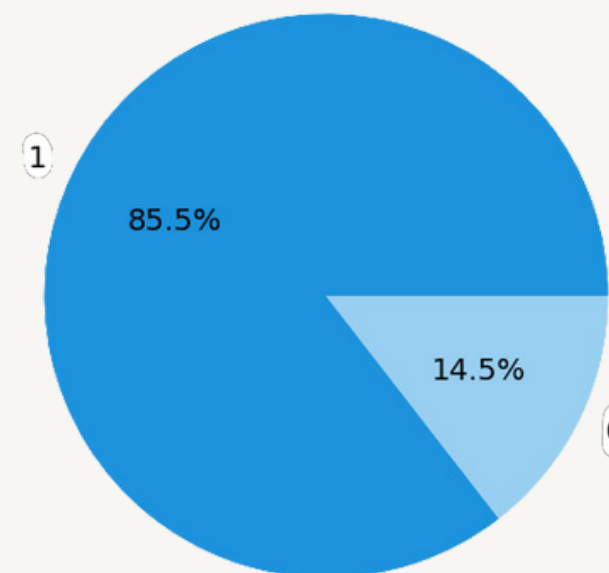
- A minor positive correlation was found between helpfulness and funniness ratings.
- **Insight:** Consider ratings of helpfulness and funniness to enhance recommendation evaluation.

- **Imbalanced Target Variable (Recommended by User)**

- A higher percentage of 1 (recommended) compared to 0 (not recommended).
- **Insight:** Addressing the class imbalance to ensure a fair representation of recommended and non-recommended games in the model. Proper resampling techniques and evaluation metrics can be employed to maintain model performance.

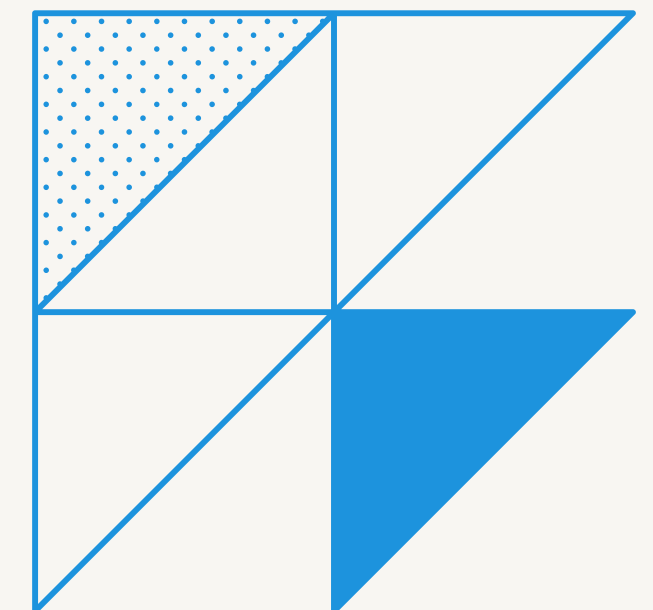
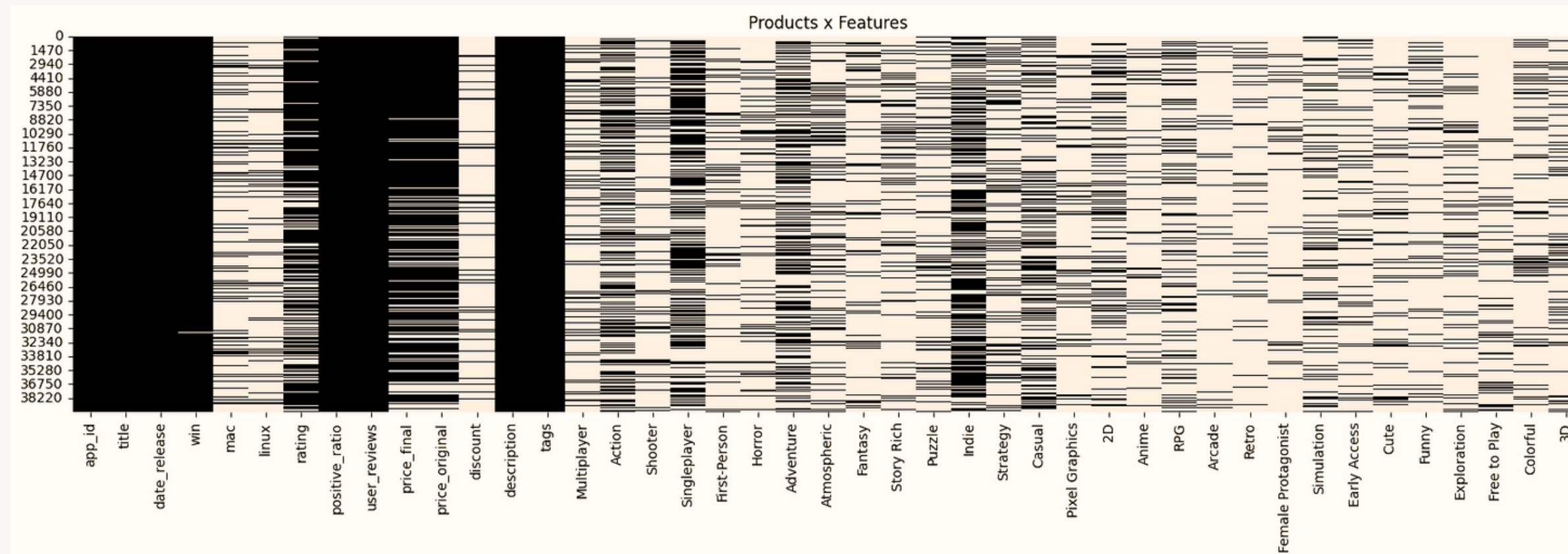


Target Variable - Recommended



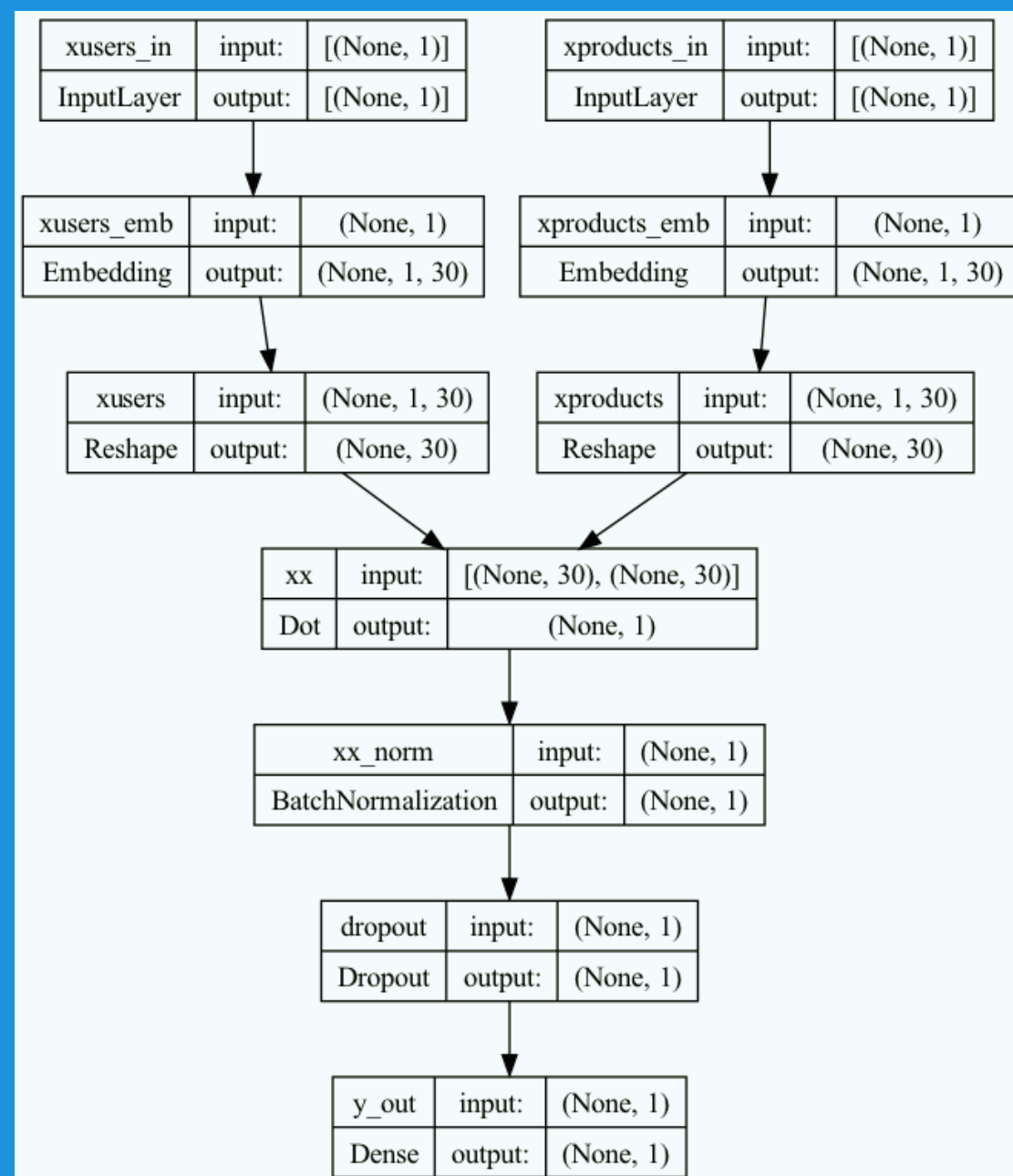
Feature Engineering & Transformations

- **Categorical Columns Encoding:** Encoded binary columns as 0s and 1s and scaled the rating column from -4 to 4 for standardized representation.
- **Top 30 Frequent Tags:** Selected and encoded the top 30 most frequent tags as new game features using binary encoding.
- **Standardization:** Applied MinMaxScalar to the feature-engineered games data for consistent scales.
- **Review Filtering:** Filtered user reviews to include only users who reviewed more than 1% of games.
- **User-Game Interaction Matrix:** Created a user-game interaction matrix by pivoting the filtered reviews data where rows represent users and columns represent games.
- **Partitioning:** Split the user-game interaction matrix by columns (games) as train and test sets for predicting the recommendation of unseen games.
- **Oversampling:** Randomly sampled minority class (0) for the train set



Model Exploration

Content-Based Filtering & Collaborative Filtering (CF) with Embeddings



CF WITH EMBEDDINGS

+ Baseline Model: Content-Based Filtering

- Utilizes **game features** and user preferences for personalized recommendations.
- Overcomes **cold-start problems** by relying on game features, making it suitable for new users or games with limited data.
- Choosing content-based filtering as a **baseline model** allows for comparison against more advanced recommender systems

+ Collaborative Filtering (CF) with Embeddings

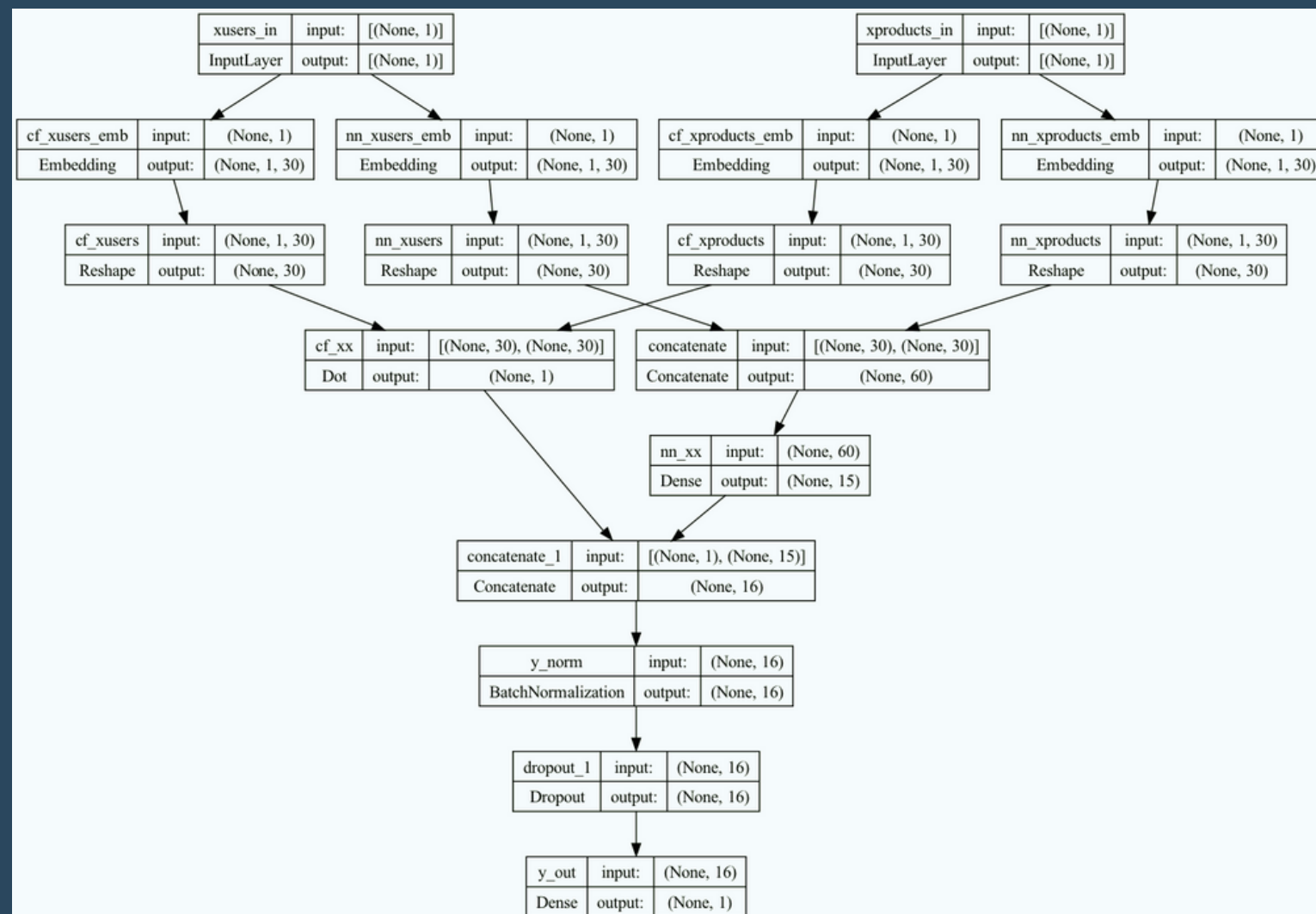
- Leverages **user-item interactions** to capture the underlying patterns and preferences in the data.
- Captures intricate relationships between users and items by mapping them to a shared **latent space**.
- Handles data **sparsity issues** by learning representations that generalize well across users and items
- Incorporates **batch normalization layer** for improved model training, convergence, and regularization.

Model Exploration

Neural Collaborative Filtering (NCF) & Hybrid Model

+ Neural Collaborative Filtering (NCF)

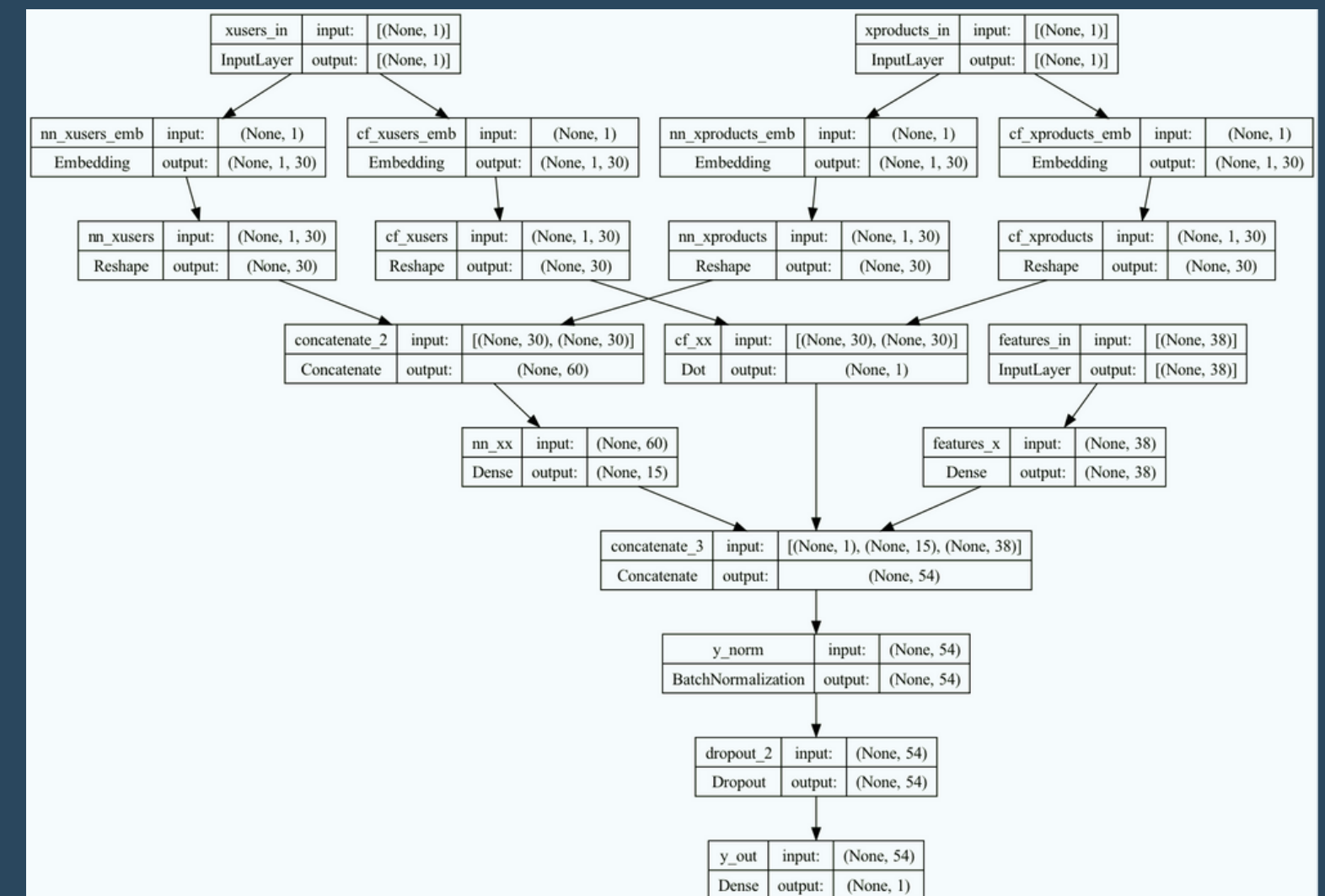
- Incorporates **neural networks** to model user-item interactions and enhance collaborative filtering for improved recommendation performance.



NEURAL COLLABORATIVE FILTERING (NCF)

+ Hybrid: NCF with Content-Based Filtering

- Integrates **multiple recommendation techniques** including collaborative filtering, content-based filtering, and deep learning, to leverage their respective strengths.

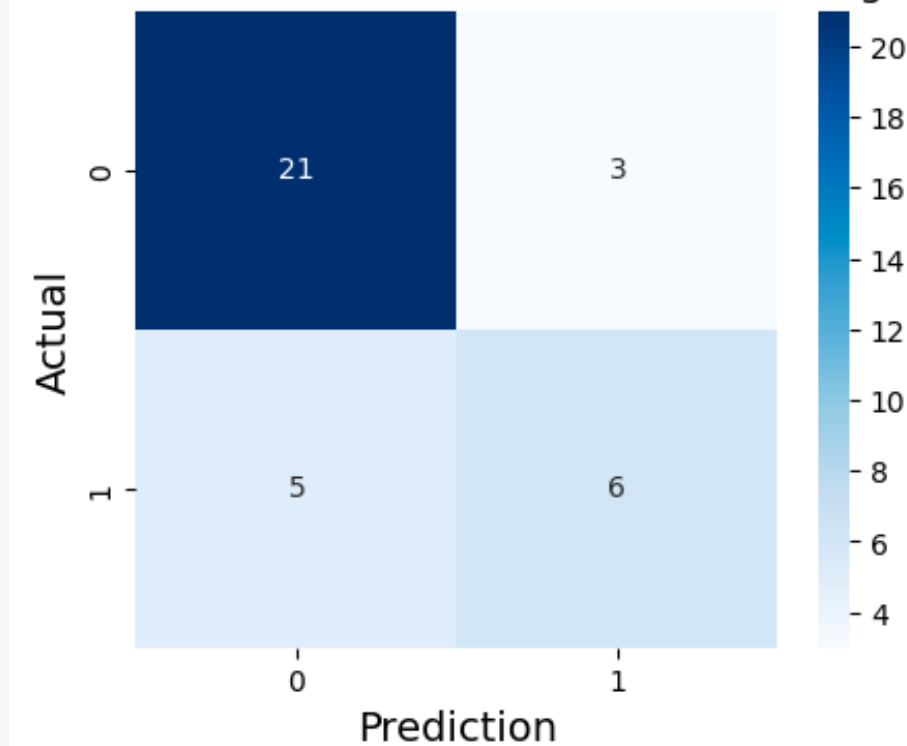


HYBRID MODEL: NCF WITH CONTENT-BASED FILTERING

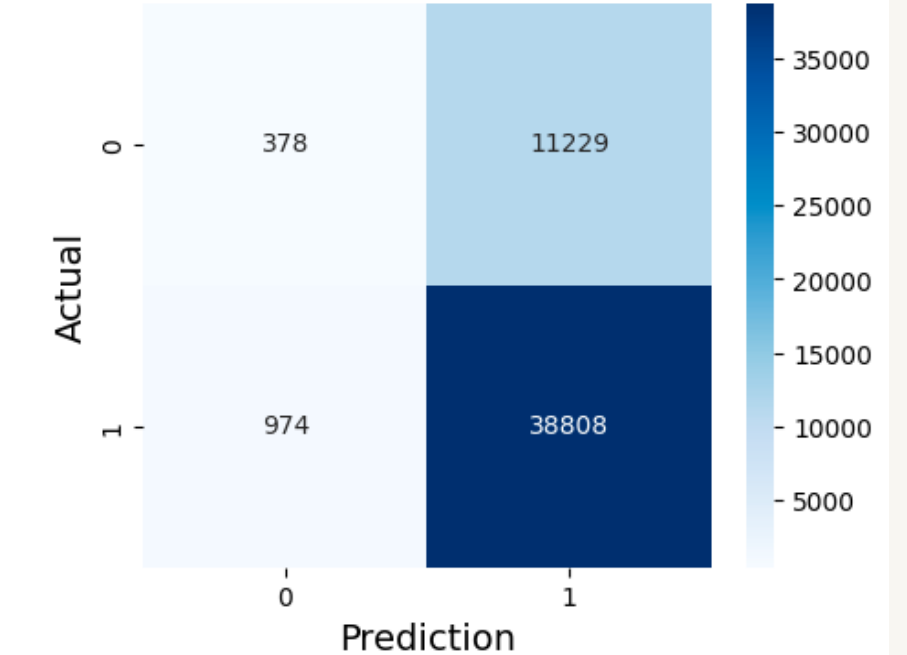
Model Selection

- Based on the confusion matrix and classification report, I choose the **Hybrid Model** as the final model since it has the highest training accuracy (80%), test accuracy (74%), precision, and recall.
- The Content-Based Filtering, Collaborative Filtering (CF), and Neural Collaborative Filtering (NCF) models have lower accuracy, precision, and recall. And they are all **severely overfitted**.
- I then improved the performance of the hybrid model by **tuning the hyperparameters**.

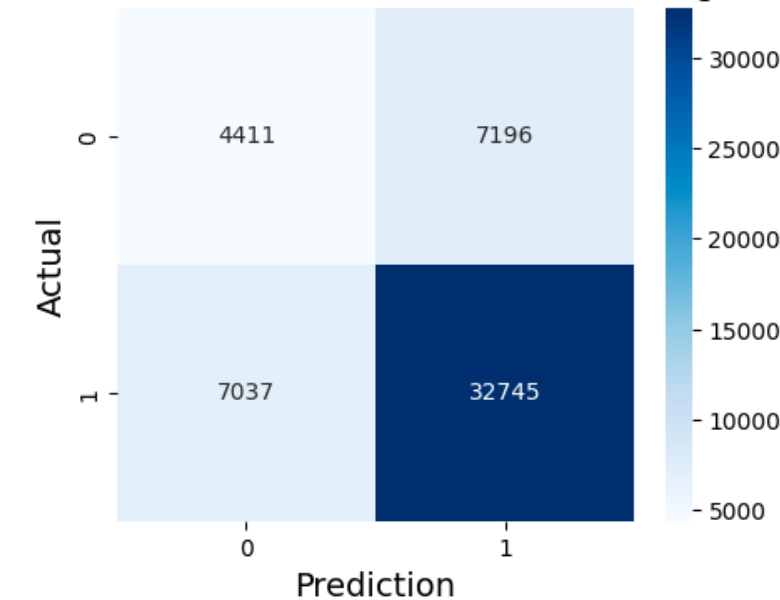
Confusion Matrix - Content-Based Filtering



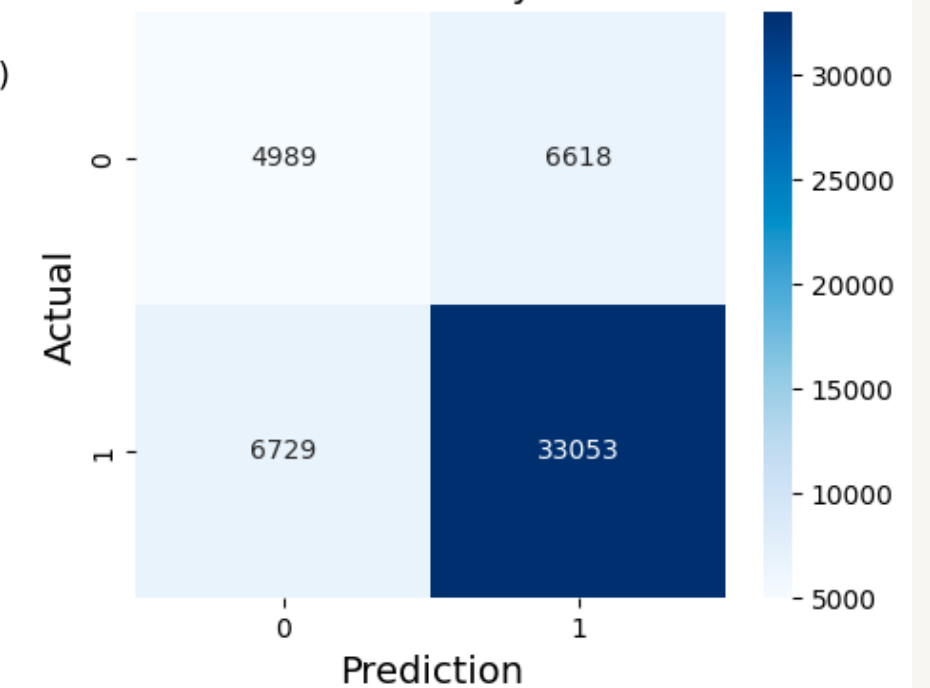
Confusion Matrix - Collaborative Filtering (CF)



Confusion Matrix - Neural Collaborative Filtering (NCF)

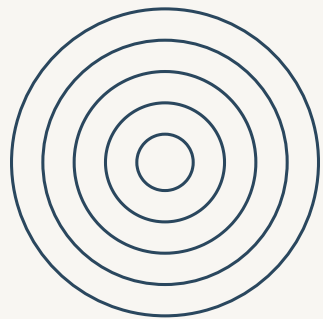


Confusion Matrix - Hybrid Model



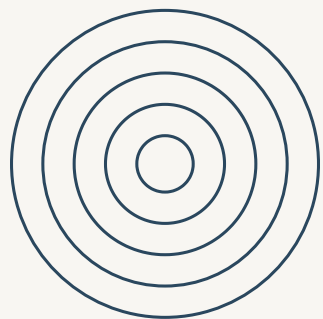
Hyperparameter Tuning

Tuning Hyperparameters for Hybrid Model Using Random Search



BEST HYPERPARAMETERS

- Embedding Size: 30
- Batch Size: 64
- Epochs: 3



MODEL IMPROVEMENT

- Loss: 0.65 -> 0.54
- Accuracy: 0.74 -> 0.75
- Precision: 0.63 -> 0.64
- Recall: 0.63 -> 0.64
- F1 Score: 0.63 -> 0.64

EMBEDDING SIZE

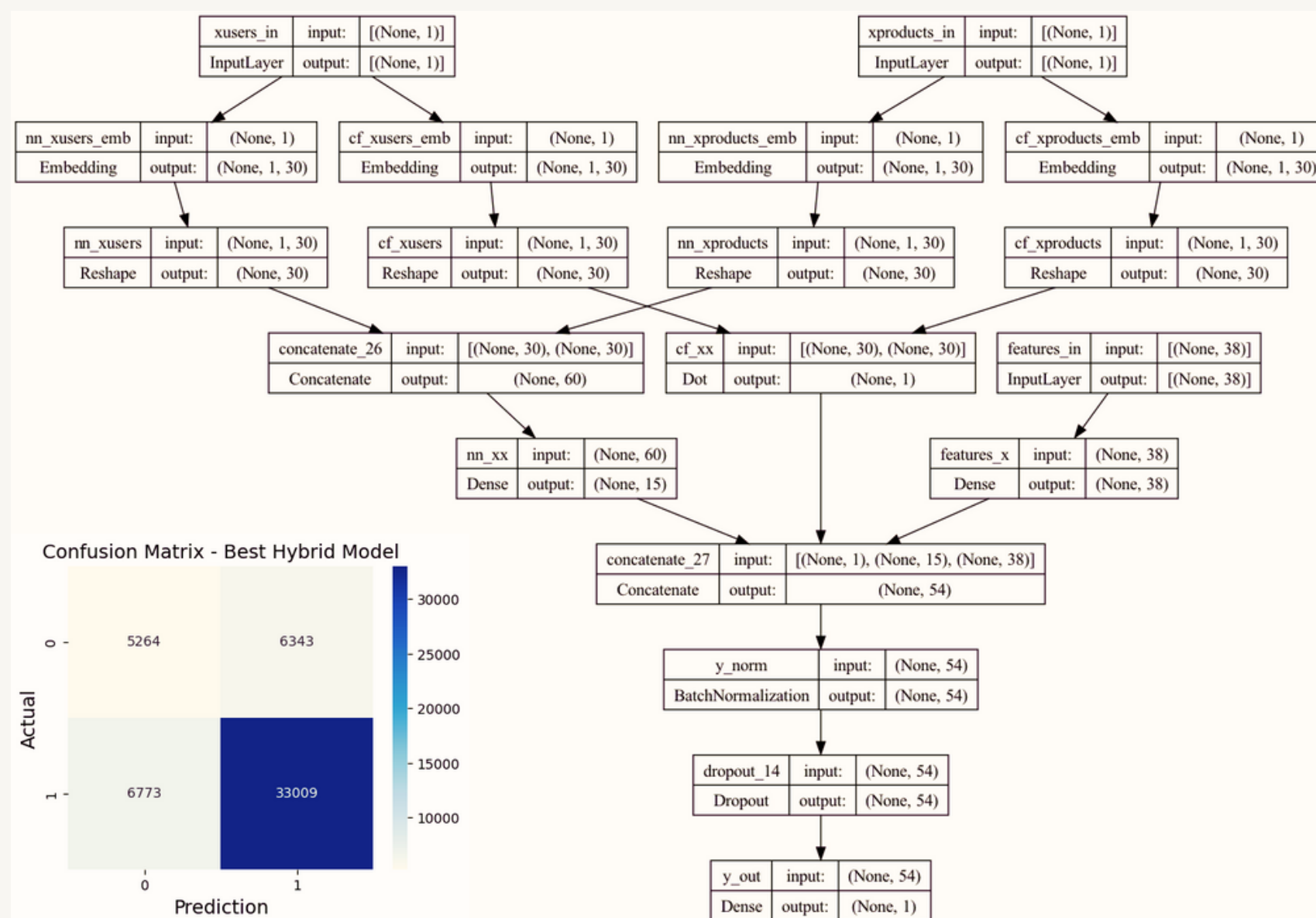
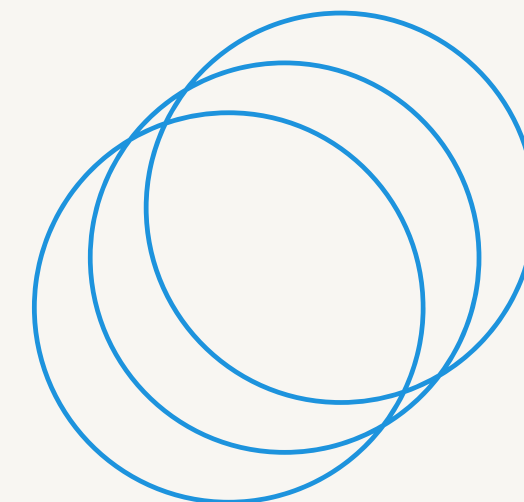
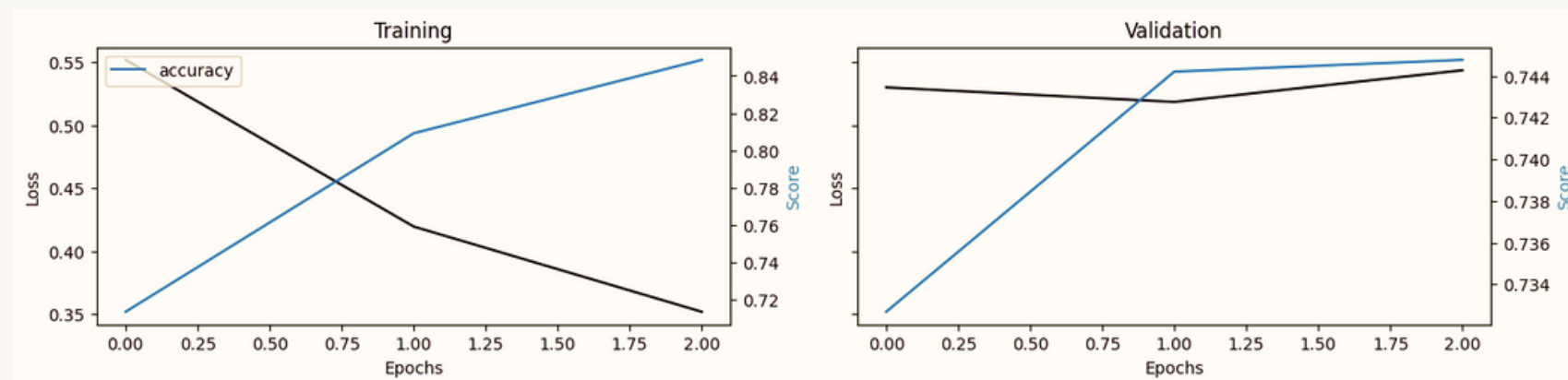
- Search Space: 10, 20, 30, 40, 50
- Increasing the embedding size can capture more complex patterns but may also increase the model's complexity and training time.

BATCH SIZE

- Search Space: 32, 64, 128, 256, 512
- Larger batch sizes can provide more stable gradient estimates and potentially faster training. However, excessively large batch sizes may lead to memory limitations or slower convergence.

EPOCHS

- Search Space: 3, 5, 7, 10, 15
 - Increasing the number of epochs allows the model to see the data more times, potentially improving its ability to learn complex relationships but may lead to overfitting.
-



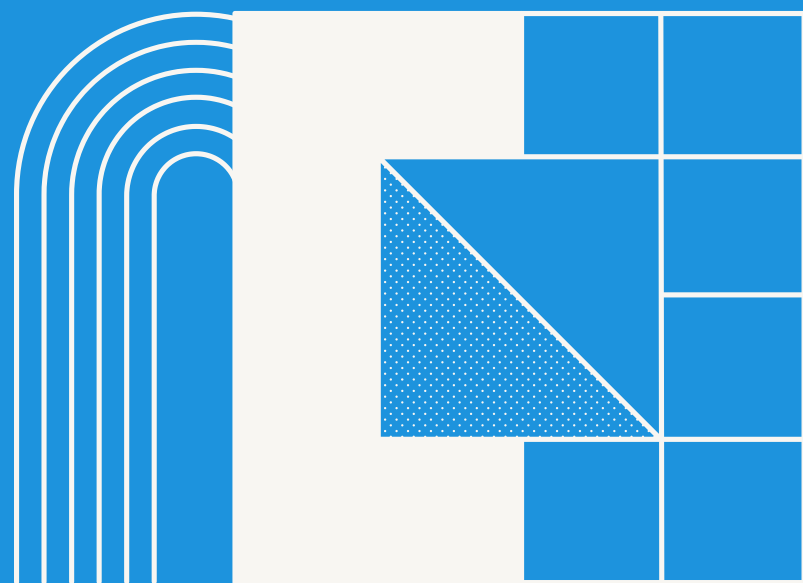
BEST HYBRID MODEL WITH EMBEDDING SIZE = 30

Model Evaluation & Results

BEST MODEL PERFORMANCE

- Loss: 0.54
- Accuracy: 0.75
- Precision: 0.64
- Recall: 0.64
- F1 Score: 0.64

Learnings & Future Work

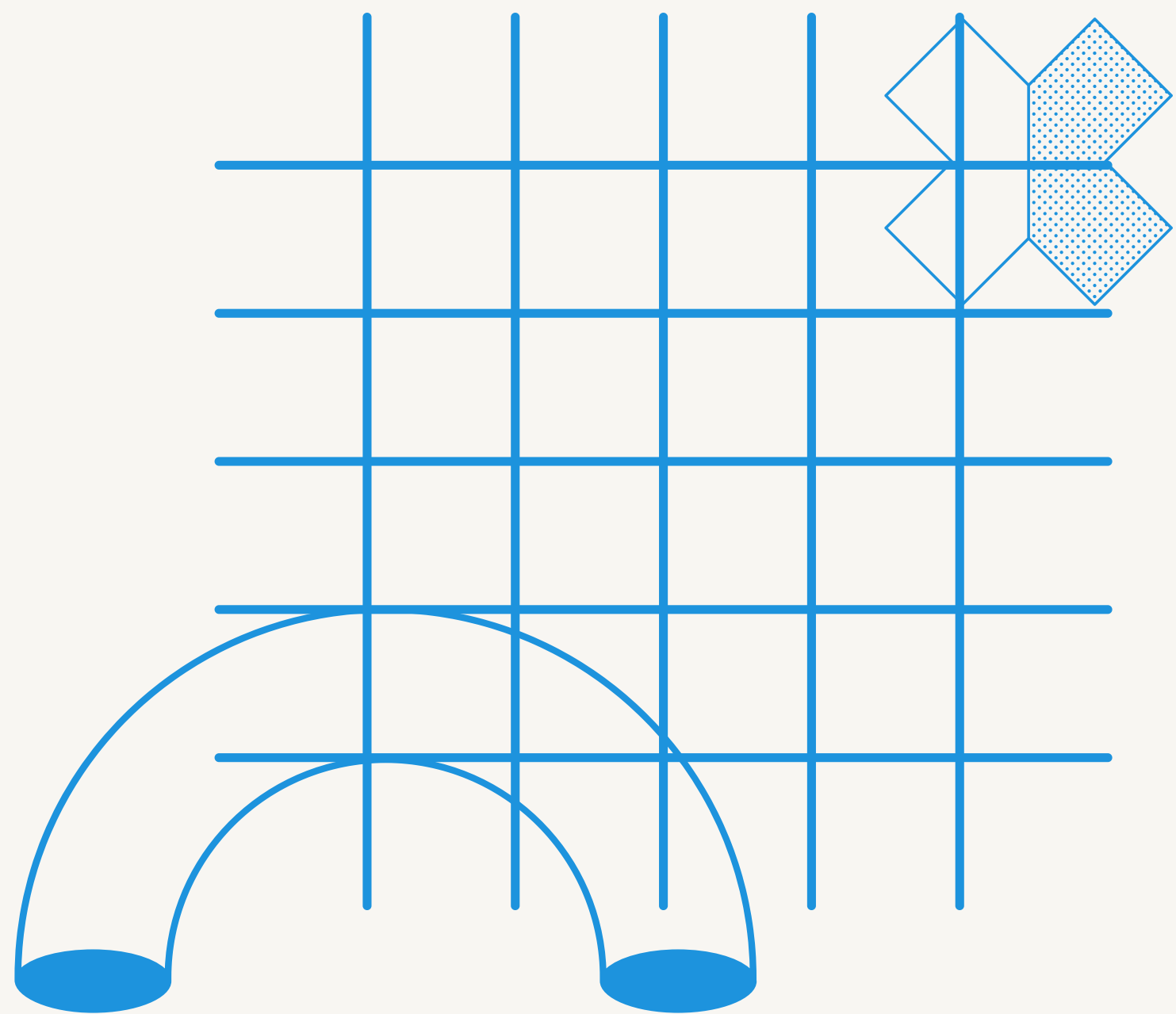


LEARNINGS FROM METHODOLOGY

- **Hybrid models improve recommendations:** Combining multiple techniques enhances accuracy and diversity.
- **Evaluation metrics guide model selection:** Choosing suitable metrics helps assess performance and select the best models.
- **Iterative refinement enhances performance:** Continuous experimentation and evaluation lead to improved recommendations.

FUTURE WORK

- **Incorporate Contextual information:** Personalize recommendations by considering factors like time, location, and user behavior.
- **Solve Cold-start problem:** Address the challenge of limited data for new users or items using content-based approaches and auxiliary information.
- **Utilize User feedback:** Incorporate explicit ratings and implicit interactions to refine the recommender system and enhance personalized recommendations.



Thank you!

SHIJIA HUANG

Email: shijia@uchicago.edu

References

- GitHub Repository: <https://github.com/slvhuang/DL-Steam-Game-Recommender-System>
- Data Source: <https://www.kaggle.com/datasets/antonkozyriev/game-recommendations-on-steam>
- <https://towardsdatascience.com/modern-recommendation-systems-with-neural-networks-3cc06a6ded2c>

