A Toy Story: Crafting Personalized Amazon Recommendations with Machine Learning

Ronne Kent Samridhi Singh

Advisor: Prof. Yifan Hu

Introduction

With the rapid growth of online retail platforms like Amazon, customers often face difficulty navigating extensive product catalogs to find items relevant to their interests and purchase history. Effective recommendation systems can significantly improve user experience, reduce choice overload, and boost customer satisfaction. However, accurately predicting a customer's next purchase remains a challenging task.

In this research, we aim to address this challenge by developing and comparing multiple machine learning-based recommendation models to find the best fitting model that accurately suggests Amazon toy products to customers based on selected features. Leveraging neural network architectures and a pre-crawled Amazon toys dataset from Kaggle, our objective is to create personalized recommendations that effectively capture user preferences and historical buying patterns.

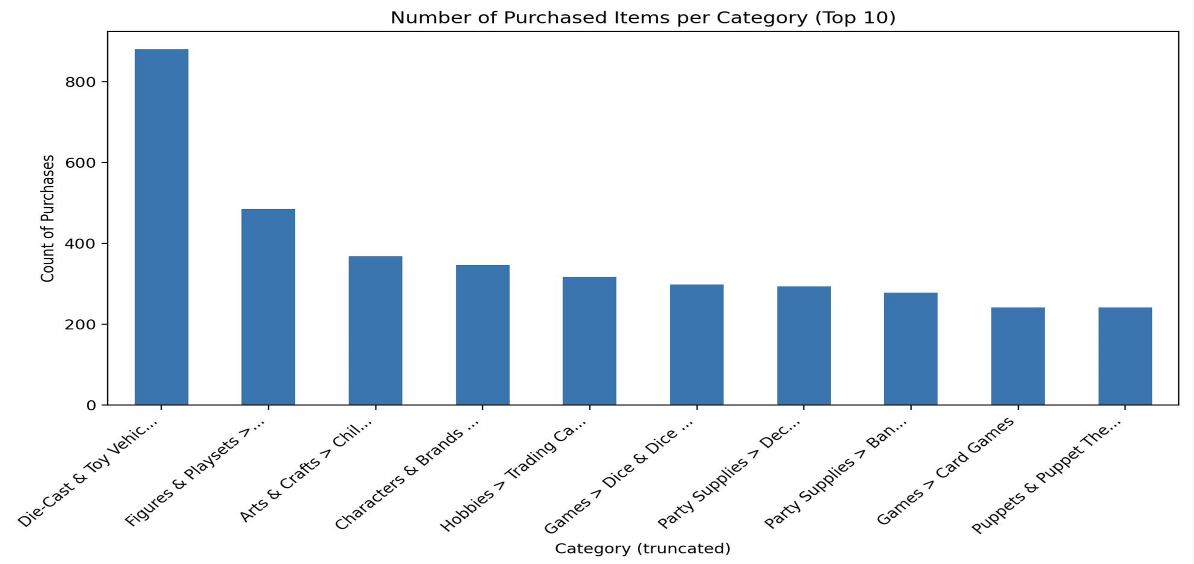
Goal and Motivation

Our aim was to build a robust recommender system for Amazon toy products using deep learning. With over 10,000 product listings and sparse user behavior data, traditional collaborative filtering is limited.
We explored **Linear Regression**, **content-based filtering** with **TF-IDF** and **deep learning** architectures to improve both precision and personalization.

Datasets

We used pre-crawled detailed dataset from **Kaggle** of toy products offered and purchased on **Amazon** to analyze purchases and create the recommendation model. The dataset includes multidimensional data the most important for our models being the product category, user ratings, price, and product name to support the predictive modeling for product recommendation.

We handled missing data by dropping certain fields where 90% of the field was missing, filling missing description and product_description observations with empty string, dropping rows where the category/sub-category is missing because that is the crucial piece of data needed and not able to back into. Missing price values were filled using median price per category.



Materials and Methods

1. TF-IDF + Cosine Similarity (Content-Based Filtering)

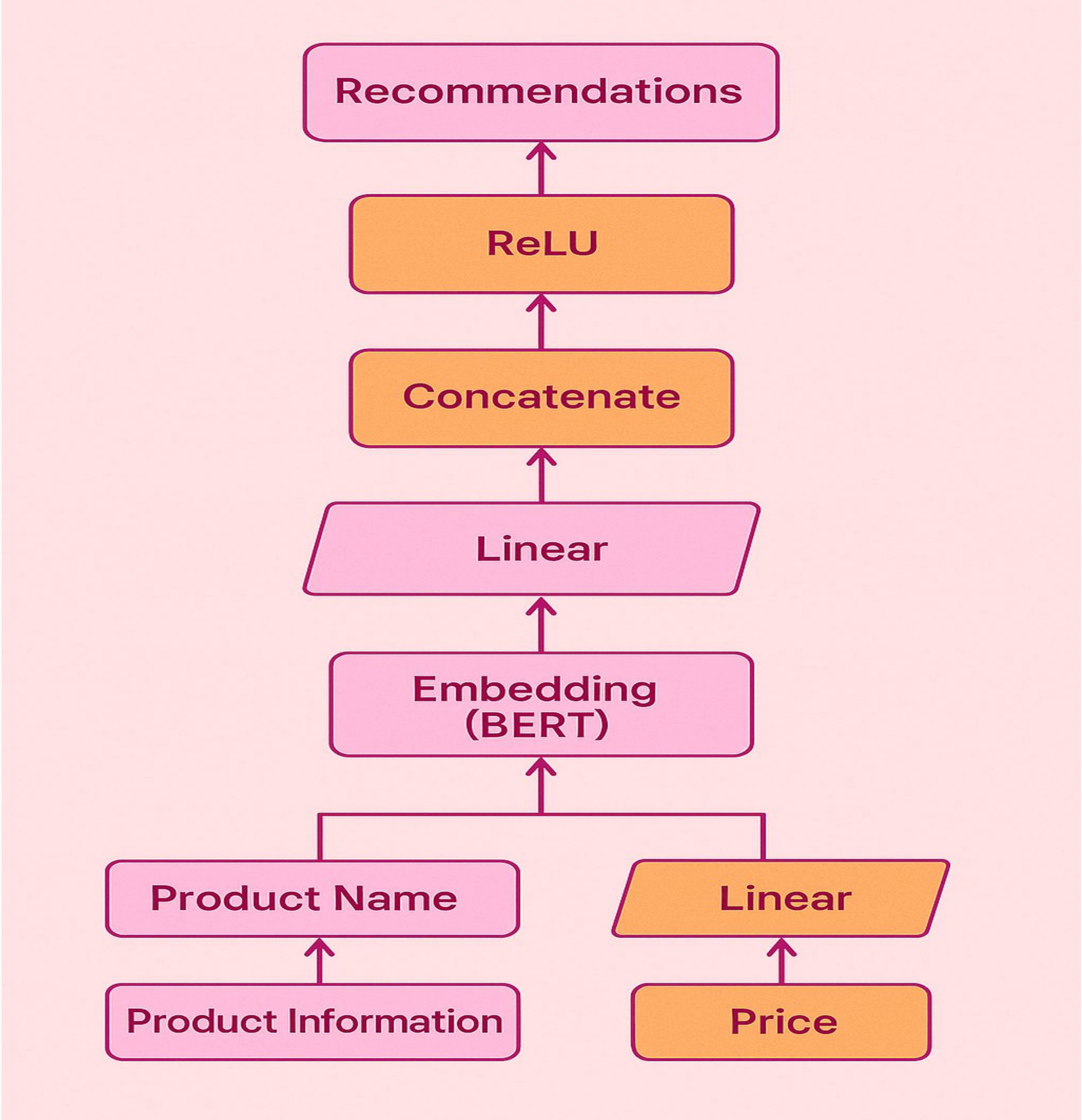
- We converted product descriptions into TF-IDF vectors to quantify the importance of words relative to the entire corpus.
- Pairwise cosine similarity was computed between a user's viewed/purchased product and all other items in the catalog.
- For each product interaction, we ranked and returned the top-k most similar items. This model was particularly effective when product descriptions were rich and unique.

2. Logistic Regression Model For Predicting Purchase Categories

- We built a baseline logistic regression model to predict the category of next purchase and create a recommendation based on the following:
- **Target variable:** top_level_category derived from the input data of amazon category & sub-category
- **Features:** product name, price, average review rating
- Encoded categorical variables to be able to be processed by the logistic model
- Logistic regression evaluates the likelihood that a given product belongs to each top-level category based on its feature values

3. Deep Learning-Based Recommender (Prototype)

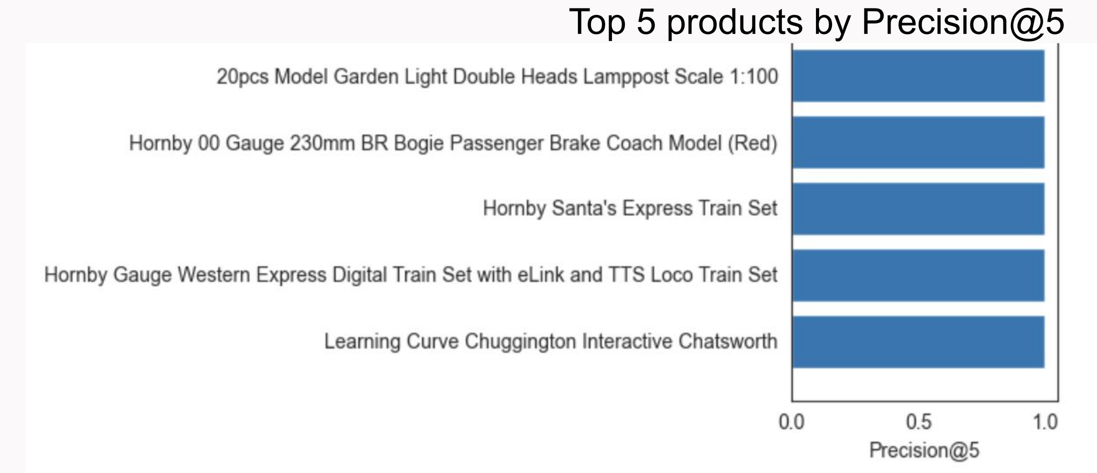
- **Embedding Layer**: For categorical variables like product category, brand, and manufacturer, we trained embeddings to capture latent semantics.
- **Dense Network**: Combined metadata features and learned embeddings were fed into fully connected layers with dropout and ReLU activations.
- **Loss Function**: Binary cross-entropy was used to predict the likelihood of co-purchase.
- The model was trained using mini-batch gradient descent with early stopping on validation loss.



Results

Our results show that **TF-IDF + Cosine Similarity** achieved a **Mean Average Precision@5 (MAP@5)** of **0.99**, indicating highly relevant top-5 recommendations. This performance demonstrates the strong discriminative power of content-based filtering when combined with descriptive product metadata.

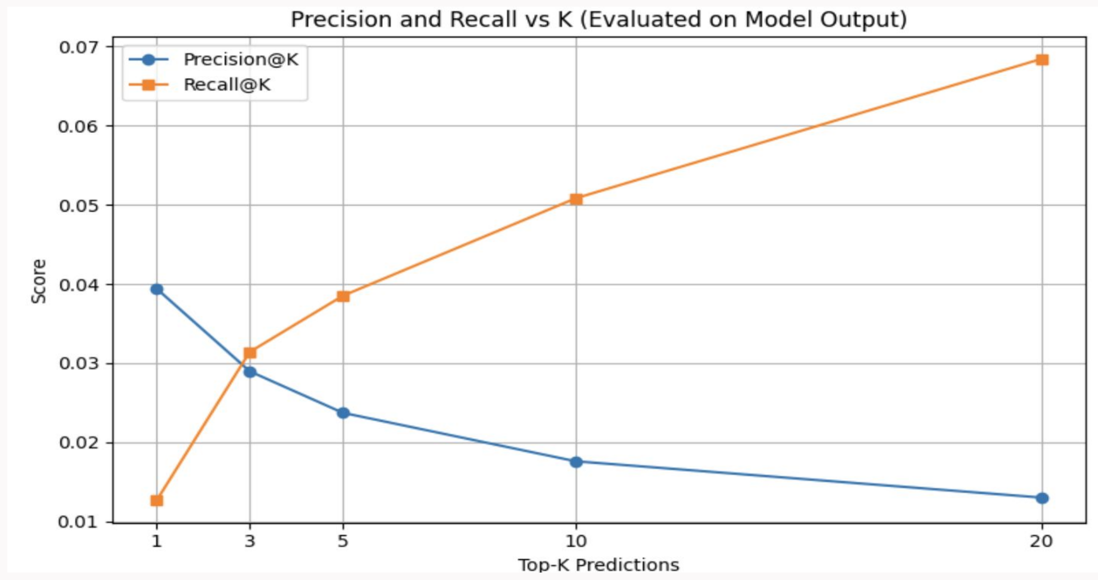
Top 5 products by Precision@5



The **logistic regression** resulted in accuracy score of **0.79**, indicating relatively high relevant recommendations. However, it is important to note that not all categories of products have substantial data resulting in some categories having more accurate product suggestions than others.

```
Query Features: {'price': 49.99, 'product_name': 'Settlers of Catan Board Game', 'average_review_rating': 4.7}
Predicted Category: Games
Recommended Products:
 - Pocket / Travel Poker Dice game
 - Chessex Dice: Polyhedral 7-Die Opaque Dice Set - Yellow with Black [Toy]
 - Trefl Frozen Dominoes
```

The **deep learning** recommender model was trained over 10 epochs using metadata inputs such as product name, description, and price, with a multi-label classification objective to predict co-purchased items. The model was evaluated on **Precision@K** and **Recall@K** metrics using top-K ranked predictions over ~28,000 product labels.



- **Precision@1** of 3.96% indicates that nearly 1 in 25 top-ranked recommendations is co-purchased in the ground truth.
- **Recall@20** approaching 7% suggests the model retrieves relevant products with reasonable coverage, despite extreme label sparsity (~0.02% positives).
- **Performance** significantly **exceeds random baseline**, which would yield **~0.0002 Precision@5.**

These results demonstrate that the model has successfully learned latent product relationships from sparse co-purchase signals and high-dimensional metadata.

Conclusions

Our study explored three primary approaches—TF-IDF with cosine similarity, logistic regression, and deep learning—for building a product recommendation system tailored to the toy category on Amazon. Each method brings unique strengths and is suited for specific real-world scenarios:

1. **TF-IDF + Cosine Similarity: Fast, Lightweight, and Cold-Start Friendly**
   - **Conclusion**: This method yielded the **highest precision (MAP@5 = 0.99)** by leveraging semantic information in product descriptions. Its strength lies in understanding product context without needing user behavior data.
   - **Limitations**: Doesn't personalize recommendations. Similarity is based only on item content, not on user preferences or behavior.
2. **Logistic Regression: Simple, Interpretable Baseline for Product Prediction**
   - **Conclusion**: While logistic regression provided a reasonable baseline, it was limited in capturing product relationships and contextual dependencies between product categories for categories where there was less robust data.
   - **Limitations**: Poor performance on large or complex datasets with high-dimensional feature interactions. Also struggles with sparsity in certain category labels.
3. **Deep Learning-Based Recommender: High-Potential for Personalized Recommendations**
   - **Conclusion**: The deep learning-based recommendation engine shows strong early-stage promise for generating personalized, scalable product suggestions in a large catalog setting. While precision may appear modest in absolute terms, the model's performance represents a 100x improvement over random in a highly sparse, multi-label environment with thousands of possible outputs.
   - **Limitations**: Requires large amounts of labeled data, hyperparameter tuning, and significant computational resources for training and inference.
   - **Impact**: This prototype lays the groundwork for building intelligent recommendation layers for e-commerce platforms, media services, and digital marketplaces. By learning from co-purchase behavior and metadata, such models can drive personalized discovery, increase cross-sell opportunities, and ultimately improve customer satisfaction through relevant product suggestions.

References

- McAuley, J., Pandey, R., & Leskovec, J. (2015). **Inferring networks of substitutable and complementary products**. *Proceedings of the 21th ACM SIGKDD*.
- Aggarwal, C. C. (2016). *Recommender Systems: The Textbook*. Springer.
- Kaggle Dataset: Toy Products on Amazon Dataset

Acknowledgement and More Information

We thank **Northeastern University** and professor **Yifan Hu** for academic guidance and support.

Ronne Kent – ronne.kent@northeastern.edu
Samridhi Singh – singh.samri@northeastern.edu