Tom Steinman
Amy Ngo
Vy Ngo
Priyanka Viswanadharao

# Data Cats eCommerce Project Report

## Introduction

Understanding customer behavior in e-commerce is crucial for optimizing business strategies and enhancing user engagement. This project explores multiple data science techniques to analyze user browsing and purchasing patterns, build collaborative filtering recommendation systems, and implement time-sensitive recommendation strategies. The primary objectives were:

- Analyze user behavior and product affinity.
- Segment users based on behavior for targeted recommendations.
- Create a time-sensitive recommendation feature that adjusts based on seasonal trends and user activity.
- Develop a time forecasting analysis to visualize customer journeys and abandoned cart behavior, events and holidays.
- Build a predictor to suggest optimal purchase timing based on historical price fluctuations.
- Develop a recommendation system using collaborative filtering and two-tower techniques.
- Implement A/B testing analysis to compare recommendation strategies.

## Related Work

Recommendation systems are critical components in e-commerce platforms, aiming to provide users with personalized product suggestions to increase engagement and revenue. Current state-of-the-art solutions include:

1. **Collaborative Filtering (CF):** Collaborative filtering leverages user-item interaction data to recommend items based on similar user behavior or item properties. Techniques such as matrix factorization and neural collaborative filtering (NCF) have shown effectiveness in capturing latent user preferences, as demonstrated in various research papers (1)(2)(3).
2. **Two-Tower Models:** Google employs two-tower architectures to effectively handle large-scale recommendation data by separating user and item embeddings into distinct networks, enabling more robust personalization(10). Two-tower models also facilitate rapid retrieval of relevant items using nearest-neighbor search methods(10), as highlighted by Yi et al. (2019)(6), Xin et al. (2021)(4) and Ni (2023)(5).
3. **Hybrid Models:** Combining multiple recommendation strategies, including CF, content-based filtering, and temporal analysis, can provide a more comprehensive system(7)(8). Amazon's approach often integrates category-based recommendations with collaborative filtering to account for both popular items and personalized suggestions(9).

In this project, we implemented collaborative filtering using NCF, a two-tower architecture for user-item interactions, and a hybrid model that combines popular products and category-specific recommendations. Additionally, time-sensitive features such as hourly activity, weekday patterns, and purchase recency were incorporated to simulate context-aware recommendations.

## Methodology

The dataset was processed using the CS5530-eCommerce-DataCats.ipynb file, which handled missing values by filling them with 'unknown_category' and 'unknown_brand'. The preprocessing pipeline included:

1. Data Cleaning:
    - Null Value Replacement:
        i. Null values in 'category_code' and 'brand' columns were filled using the unique identifier ‘product_id’ to fill as many null values as possible
    - Data Splitting:
        i. Null values in the 'category_code' and 'brand' columns were filled with 'unknown_category' and 'unknown_brand'.
        ii. Null values in the 'category_code' and 'brand' columns were removed
2. Analyze Clean Dataset:
    - Create various models to analyze and choose between the two datasets
    - Use clustering to determine user engagement
    - Create journey visualizations and time forecasting
    - Create browsing and purchasing pattern visuals for interpretation
3. Analyze User Engagement Clustering
    - Used KMeans and elbow method into 5 groups by user engagement
4. Analyze browsing and purchasing patterns
    - Determined top products/categories purchased for hours, days of the week, and months
    - Used this data to create recommendation system based on time
5. Analyze Time Forecasting Model:
    - Based on purchases, the model, ‘Prophet’, captured: trends by detecting non-periodic changes in values; seasonality by detecting periodic changes based on time (daily, weekly, and holidays)
6. Create Recommendation Model:
    - The cleaned dataset, with filled null values was used to train multiple NCF and two-tower models to evaluate various recommendation strategies, including Standard (Top 5 Popular Products), Two-Tower, Category-Based, and Hybrid approaches, updating and .
7. Feature Engineering:
    - New features such as user activity level, purchase rate, and recency weight were calculated to improve recommendation models.
    - Event types were weighted to emphasize purchases over views and cart additions.
    - Temporal features such as hour and day of the event were encoded using sine and cosine transformations to capture cyclical patterns.

The models all used the filled brand and category datasets for model training. Further methodological details and model architectures are elaborated in the provided notebooks.

## Results and Discussion

The results of the Prophet model indicated that Black Friday is the most popular holiday for purchases to occur. Beyond that season, the model struggled to forecast future seasons due to the dataset only occurring in October and November.

Additional Forecasting Predictions:

- Daily: In a 24 hour day, the most purchases were made at the earliest hour of the day and decreased towards the afternoon hours. Then peaked again in the late hours of the note past 8:00 PM.
- Weekly: In the 7 day week, purchases were forecasted to peak on days such as Monday, Wednesday, and Friday. Friday being the most popular day, while Saturday is forecasted to be the least popular.

Potential Issues:

- Data might be skewed towards early Friday and late Friday even due to most purchases occurring on Black Friday. So, analysis is limited to the months of October and November rather than considering other annual holidays.

The results from the various NCF models and the two-tower models were compared to assess the effectiveness of each recommendation strategy using A/B/C/D testing. The implemented strategies include:

A. **Standard (Popular Products):** Recommendations based solely on the top purchased products.
B. **Two-Tower Model:** Personalized recommendations leveraging user and item embeddings to predict interactions. The Two-Tower approach used both October and November datasets at a 1% sampling rate, raising potential concerns about data leakage and bias in the evaluation.
C. **Category-Based Model:** Recommendations filtered by frequently interacted categories.
D. **Hybrid Model:** Combination of popular products, category-based recommendations, and personalized predictions.

Evaluation Metrics:

- CTR (Click-Through Rate)
- Precision
- Recall
- F1 Score
- MRR (Mean Reciprocal Rank)
- NDCG (Normalized Discounted Cumulative Gain)
- Coverage

The best-performing strategy in terms of CTR was the Standard approach, which capitalized on high-frequency products. However, the Two-Tower model demonstrated improved precision and recall, indicating better personalization despite a lower CTR. The Hybrid approach attempted to balance both popularity and personalization, resulting in moderate performance across all metrics.

Potential Issues:

- Data Leakage: Using October and November datasets simultaneously for both training and evaluation in the final two-tower model could lead to data leakage, inflating model performance.
- Sampling Bias: The use of only 1% of the dataset may limit the model's ability to generalize effectively to the entire user base.

Detailed performance metrics and comparison plots are provided in the CS5530-eCommerce-DataCats.ipynb file for each model, with individual model evaluations and A/B/C/D testing analysis documented for further reference. The results from the various NCF models and the two-tower models were compared to assess the effectiveness of each recommendation strategy. A rough estimation of the impact on company revenue was generated in comparing each method, as well as combining them without competition to estimate revenue.

## Conclusion and Future Work

The implementation of multiple recommendation strategies, including customer segmentation, time/event analysis, behavior analysis, and time-sensitive recommendations, provided comprehensive insights into user behavior and purchasing patterns. The Two-Tower model, in particular, leveraged user and item embeddings to provide more personalized recommendations. However, the use of both October and November datasets during training introduced risks of data leakage and bias, potentially impacting model performance.

To address these concerns and further refine the recommendation system, future work should focus on:

- **Integration of Multiple Strategies:** Combining the strengths of the Two-Tower model, category-based recommendations, and time/event analysis to improve overall system effectiveness. Integrating a “cold-start” approach into the approach would be beneficial(8).
- **Enhanced Customer Segmentation:** Utilizing historical purchase data, product interactions, and browsing patterns to develop targeted recommendation strategies for distinct user groups created in the user segmentation KMeans model.
- **Temporal and Event-Based Analysis:** Further leveraging significant shopping events like Black Friday to refine the model’s responsiveness to seasonal trends and pricing fluctuations to account for data drift.
- **Dynamic Pricing and Purchase Timing Predictions:** Implementing a dynamic pricing model that adjusts recommendations based on historical price changes and optimal purchase timing incorporating the time recommendation system.
- **Data Partitioning and Structured Evaluation:** Establishing a more rigorous training/testing split using October data exclusively for training and only providing sale pricing for November.

By combining these approaches, the recommendation system can achieve a more balanced and effective strategy that leverages both popular products and personalized recommendations while mitigating overfitting and data leakage risks.

## References


(1) Li, Y., Lu, L., & Xuefeng, L. (2005). A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in E-Commerce. *Expert Systems with Applications, 28*(1), 67–77. https://doi.org/10.1016/j.eswa.2004.08.013

(2) Huang, Z., Zeng, D., & Chen, H. (2007). A comparison of collaborative-filtering recommendation algorithms for e-commerce. *IEEE Intelligent Systems, 22*(5), 68–78. https://doi.org/10.1109/MIS.2007.4338497

(3) Kumar, M. R., Vishnu, S., Roshen, G., Kumar, D. N., Revathi, P., & Baster, D. R. L. (2024). Product recommendation using collaborative filtering and K-means clustering. *2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT)*, 1722–1728. https://doi.org/10.1109/IC2PCT60090.2024.10486625

(4) Xin, S., et al. (2021). ATNN: Adversarial Two-Tower Neural Network for New Item's Popularity Prediction in E-commerce. *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, 2499–2510. https://doi.org/10.1109/ICDE51399.2021.00282

(5) Ni, J. (2023). *Customized E-commerce B2C Recommendation System for Amazon* (Master's thesis, University of Chicago). University of Chicago Knowledge Archive. https://doi.org/10.6082/uchicago.6096

(6) Yang, J., Hong, L., Yi, X., Cheng, D. Z., Li, Y., Wang, S. X., Xu, T., & Chi, E. H. (2020). *Mixed Negative Sampling for Learning Two-Tower Neural Networks in Recommendations*. Retrieved from https://yangli-cs-ucsb.github.io/paper/www20_yang.pdf

(7) Chen, R., Hua, Q., Chang, Y.-S., Wang, B., Zhang, L., & Kong, X. (2018). A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access, 6*, 64301-64320. https://doi.org/10.1109/ACCESS.2018.2877208

(8) Lee, W.-M., & Cho, Y.-S. (2023). A flexible two-tower model for item cold-start recommendation. *IEEE Access, 11*, 146194-146207. https://doi.org/10.1109/ACCESS.2023.3346918

(9) Hardesty, L. (2019, November 22). *The history of Amazon's recommendation algorithm: Collaborative filtering and beyond*. Amazon Science. Retrieved from https://www.amazon.science/the-history-of-amazons-recommendation-algorithm

(10) Wortz, J., & Totten, J. (2023, April 19). *Scaling deep retrieval with TensorFlow Recommenders and Vertex AI Matching Engine*. Google Cloud. Retrieved from https://cloud.google.com/blog/products/ai-machine-learning/scaling-deep-retrieval-tensorflow-two-towers-architecture