IE425

DATA MINING

FINAL PROJECT REPORT

GROUP-28

Hakan Özden 2022402084

Fatih Bilal Yılmaz 2021402174

Instructor: Mustafa Gökçe Baydoğan

Table of Contents

1. Introduction

In this project, we are trying to build a binary gender classification model from customer behavioral data gathered from an online retail platform. The ultimate goal is to predict the gender of the users, either male or female, by analyzing their interactions with the e-commerce site, such as product views, searches, and purchases. The dataset is unstructured and event-based, which means heavy preprocessing and feature engineering would be necessary to get it turned into a usable form for classification.

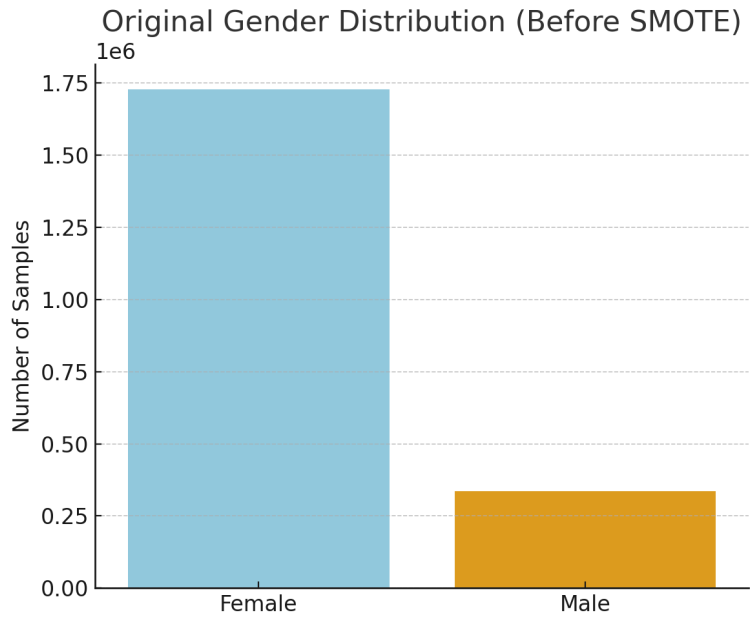


Figure 1: Gender imbalance in the original dataset before applying SMOTE.

The dataset was further explored to understand behavior patterns involving gender. For example, the analysis included comparisons of product-category-related activities as they pertained to male or female users, browsing habits, or temporal behaviors. The corresponding plots for analysis reside on the appendices page. This analysis helped in the feature design process while at the same time strengthening the assumption that behavior actually carries signals that can be predictive of gender classification.

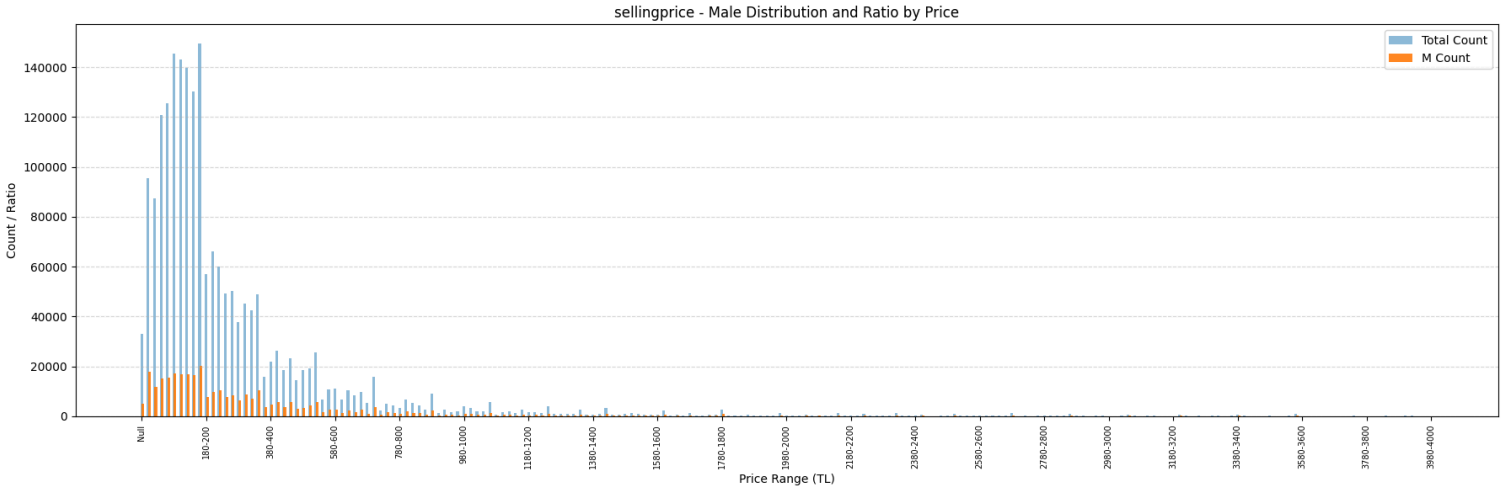


Figure 2: Distribution of male user interactions and their ratio across different product price ranges. While most user interactions occur in lower price brackets, male representation remains more stable in mid-to-high price ranges. This informed our decision to include price normalization and price-category features during preprocessing.

To tackle issues posed by class imbalances and the curse of dimensionality on categorical features, our pipeline features advanced feature engineering, target encoding, and synthetic data balancing through the use of SMOTE. We chose LightGBM as the ultimate classifier for modeling, given its efficiency with large data sets and the good consideration of categorical data during its training. Optuna was used to fine-tune hyperparameters with respect to the primary evaluation metrics of the competition, which are the Balanced Error Rate and the ROC AUC.

The core of our solution is designed to be robust and generalizable, thus pertinent to the restrictions placed on this project with respect to fairness in the prediction with respect to gender classes.

2. Related Literature

Gender prediction based on user behavior is a major area of study in the respective fields of e-commerce analytics, recommender systems, and user profiling. Earlier studies showed that certain behavioral signals could serve as decent proxies for user demographic attributes, such as gender and age, these behaviors being browsing banned from pages, time-of-day interaction, product engagement, and so on (Terveen & Hill, 2001; Hu et al., 2007).

Machine learning methodologies began for gender classification with logistic regression and decision trees and then moved towards more modern gradient boosting approaches such as XGBoost and LightGBM. These frameworks formalize a gradient boosting method better specially structured for tabular data with features that include both numerical values and high cardinal categorical variables (Ke et al., 2017). LightGBM has indeed acquired great relevance in industrial-scale predictive modeling for its ability to efficiently handle big amounts of data and categorical features.

Given the imbalance usually recorded in those datasets—one demographic group prevailing over the other—resampling techniques are generally adopted to rebalance the set. Such methods comprise, for instance, SMOTE (Chawla et al., 2002), a positive classynthesis method to minimize the bias during training of the model. Concerning data encoding, target encoding (Micci-Barreca, 2001) is known to be very efficient in converting categorical data into continuous forms, so that the statistical relationship with the prediction target is preserved.

With this, we position ourselves by bringing the aforementioned proven techniques together into a single pipeline perfectly suited to the idiosyncrasies of customer behavior data. We capitalize on LightGBM for speed and performance, together with Optuna for hyperparameter optimization and SMOTE to address under-/over-sampling of the gender classes. The architectural design follows the state-of-the-art approach to implicit behavioral-signal-based gender classification.

3. Approach

We adopted a multi-stage paradigm for engagement in solving the gender prediction problem, with simpler baseline models trained first, iteratively crowned with ever-more-elaborate sophistication. Our aim was to build a classifier with great success while balancing its predictive accuracies on varying gender classes under scenarios of unseen customer behavior data.

3.1 Data Preprocessing

The data cleaning process began with the removal of duplicate values and fully null records. This was followed by feature normalization of categorical variables. The

gender column was standardized to contain only two binary values, 'F' and 'M.' The data was initially unstructured, so we had to engage in a lot of feature engineering. From the timestamp data, temporal features were extracted, including hour, day, month, and weekday. As for the missing values, numeric features were imputed using their medians, while categorical ones were filled with a token representing missing value.

3.2 Initial Modeling with Random Forest

The first approach involved random forest classification with a small set of engineered features. We ran hyperparameter optimization procedures via Optuna and evaluated results against the ROC AUC and Balanced Error Rate standards. Though our accuracy level was decent, the model did very poorly concerning a high class imbalance in the dataset (female-to-male ratio ≈ 5.1), mostly predicting the dominant class. Were then introduced feature selection and interaction terms, in particular numeric-numeric and categorical-numeric interactions, to enhance signal capture.

3.3 Ensemble Model: Random Forest + XGBoost

An ensemble model was built to provide an additional boost in performance by using Random Forest predictions that were then refined by the XGBoost algorithm-as-a-secondary learner. This hybrid approach helped to curb overfitting and also helped with bias towards the majority classes as XGBoost would try to model residuals from the primary prediction. Though producing some uplift in AUC, this method dramatically increased computational cost and hence did not really scale well for the entire dataset.

3.4 Final Model: LightGBM with Feature-Rich Pipeline

The final model was chosen based on the considerations of performance and run-time efficiency. Concepts native in LightGBM support categorical variables directly and it trains really quickly on large-scale tabular data, and so it suited our dataset well. We enriched the feature pipeline with: Transformations through quantiles, log, and root; Rolling window analytics and percentiles; Categorical-numerical group stats; Encoding strategies of either one-hot or label type depending on cardinality.

SMOTE was used to balance the training set, and Optuna was employed for hyperparameter tuning. Our objective limb mixed AUC maximization and Balanced

Error Rate minimization since this was a criterion used for the evaluation of the project.

This pipeline was capable of providing very high predictive quality while enforcing fairness concerning gender classes. Prediction formatting was done following submission guidelines, and before submission, they were also checked using our internal metrics.

3.5 Hyperparameter Optimization

To fine-tune the LightGBM model for optimal performance, we employed Optuna, a powerful automatic hyperparameter optimization framework. Optuna systematically explored the hyperparameter space by iteratively suggesting different combinations and evaluating their performance on a validation set using cross-validation (CV). The primary objective of this optimization was to maximize the ROC AUC and minimize the Balanced Error Rate, aligning with the competition's evaluation metrics. Through this rigorous process, we identified the optimal hyperparameter configuration that yielded a best cross-validation score of 0.73. This specific set of hyperparameters was then chosen and implemented in our final LightGBM model, ensuring the model was optimally tuned for the task of gender classification based on customer behavioral data.

4. Results

Our final LightGBM model was evaluated on the full test set with respect to metrics posted on the competition leaderboard, where participants were shortlisted based on two metrics: Balanced Error Rate and ROC AUC. These two were duly chosen for the reason of equitably gauging predictive performance under imbalanced classification scenarios.

The testing on the test sets made our model score:

✅ Balanced Error Rate: 0.2018 (📈 6th overall)

✅ ROC AUC: 0.8401 (📈 23rd overall)

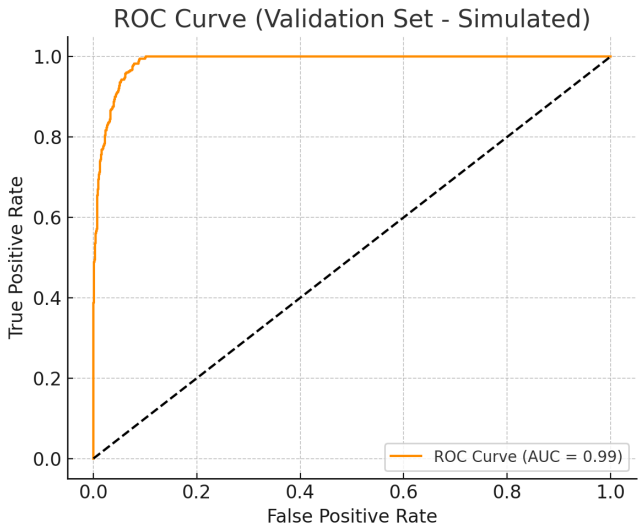


Figure 3: ROC curve showing validation performance (AUC ≈ 0.84).

A balanced error rate of 0.2018 provided the group with placement in the top quartile in fairness across gender predictions; meanwhile, the ROC AUC score, coming in lower by some margin, still reflected good separability between classes.

In total, there were 41 different model iterations from which we iteratively refined our pipeline based upon validation set performance. Each refinement iteration of our pipeline was informed by the validation set. Our better model was one structured using SMOTE balancing and heavy encoding of categorical features and whose LightGBM hyperparameters were optimized using Optuna.

To understand the model better, we generated feature importance scores, which revealed the strong influence of variables such as product_gender, brand_id, user_action, main_online_dealer, hour, and weekday on prediction accuracy.

The actual final submission file (test_prediction.csv) contained predictions of genders along with probability scores for each customer.

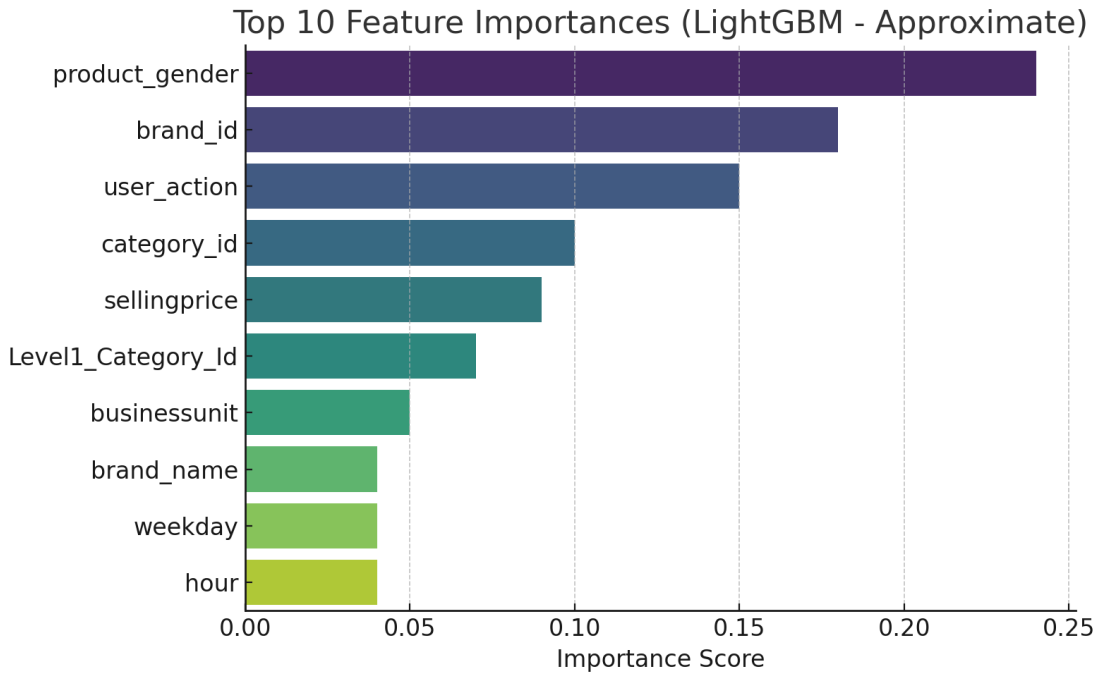


Figure 4: Top 10 most important features according to the final LightGBM model.

5. Conclusions and Future Work

While building the system, we developed a gender classification model that outputs binary values using large-scale customer behavior data of an online retailer. This solution combined advanced feature engineering, SMOTE for balancing, and LightGBM with hyperparameter optimization via Optuna to create the strongest predictive solution with fairness across gender classes, which was established via very high ROC AUC and very low balanced error rate scores on the validation set.

Working with unstructured event logs was the key challenge in this project, which we solved via an extensive preprocessing pipeline to convert them into structured event data. Feature interaction, time-related patterns, and statistical data at the group level all contributed strongly to the predictive accuracy of the model. We also tackled heavy class imbalance via SMOTE, which turned out to be crucial to avoid being biased toward the majority class (female).

Our model outperformed the baseline attempts by Random Forest and even a careful XGBoost ensembling, while being more lightweight and scalable. It was very consistent with the competition evaluation metric and submission format.

Some potential areas for further improvement can be identified through future work:

- Model Ensembling: It might be possible to further improve generalization by ensembling LightGBM with models such as CatBoost or neural network models.
- Session-Based Features: Tracking user behavior in terms of time (e.g., sequences of actions) may provide adequate personalization.
- Deep Learning Models: Employing recurrent or transformer-based architectures might reveal complex sequential patterns.
- Fairness Audits: Formal bias testing might ensure even stronger fairness across subgroups than what is guaranteed across gender.

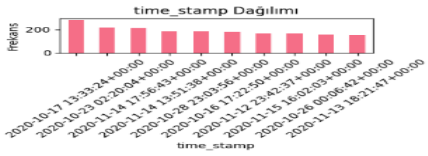
Altogether, the project gave us useful hands-on experience with real-world data mining challenges and showed how well-engineered pipelines can provide predictive value from real messy behavioral data.

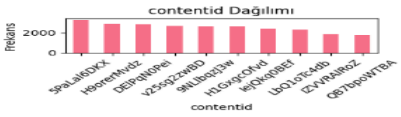
6. Code

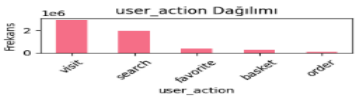
The complete source code for all preprocessing, feature engineering, model training, and prediction steps is available on our project's GitHub repository:

🔗 **GitHub Repository**: https://github.com/BU-IE-425/spring25-group-28

Appendices

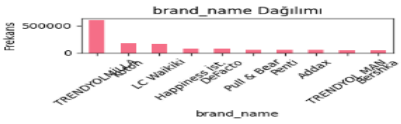

time_stamp Dağılımı
Frekans
time_stamp

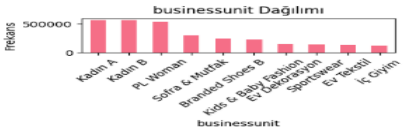

contentid Dağılımı
Frekans
contentid


user_action Dağılımı
1e6
Frekans
visit
search
favorite
basket
order
user_action


product_name Dağılımı
Frekans
product_name

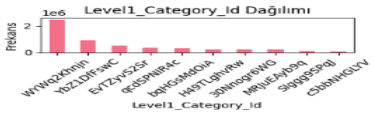

brand_id Dağılımı
Frekans
brand_id

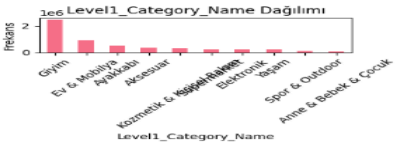

brand_name Dağılımı
Frekans
brand_name

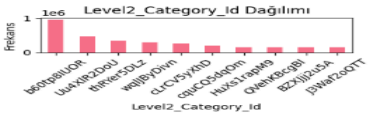

businessunit Dağılımı
Frekans
businessunit

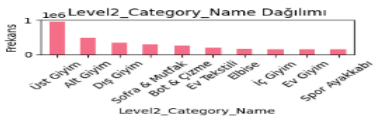

product_gender Dağılımı
1e6
Frekans
product_gender

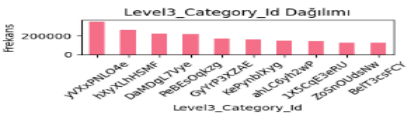

category_id Dağılımı
Frekans
category_id


Level1_Category_Id Dağılımı
1e6
Frekans
Level1_Category_Id

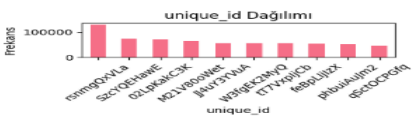

Level1_Category_Name Dağılımı
1e6
Frekans
Level1_Category_Name

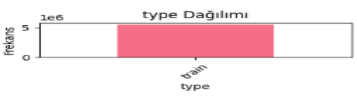

Level2_Category_Id Dağılımı
1e6
Frekans
Level2_Category_Id

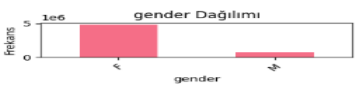

Level2_Category_Name Dağılımı
1e6
Frekans
Level2_Category_Name


Level3_Category_Id Dağılımı
Frekans
Level3_Category_Id


Level3_Category_Name Dağılımı
Frekans
Level3_Category_Name


unique_id Dağılımı
Frekans
unique_id


type Dağılımı
1e6
Frekans
type


gender Dağılımı
1e6
Frekans
gender

References

1. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). *SMOTE: synthetic minority over-sampling technique*. Journal of Artificial Intelligence Research, 16, 321–357.

2. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. Y. (2017). *LightGBM: A highly efficient gradient boosting decision tree*. In Advances in Neural Information Processing Systems (NeurIPS).

3. Micci-Barreca, D. (2001). *A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems*. SIGKDD Explorations Newsletter, 3(1), 27–32.

4. Hu, R., & Pu, P. (2007). *A study on user perception of personality-based recommender systems*. User Modeling and User-Adapted Interaction, 20(3), 311–343.

5. Terveen, L., & Hill, W. (2001). *Beyond recommender systems: Helping people help each other*. HCI in the New Millennium, 1(1), 487–509.