Machine Learning / AI

VIDEO GAME SALES ANALYSIS

Renata Golemovic

October 2024

1. Use Case

The objective of this report is to develop and evaluate machine learning models for predicting the global sales of video games based on descriptive features such as year of release, platform, genre, and publisher.

Initially, the AutoGluon framework was used to automatically test and compare multiple algorithms in order to identify the best-performing model. After determining the optimal model, it was retrained separately on the same dataset to reduce memory usage and improve efficiency.

Finally, a Streamlit web application was developed to deploy the trained model, allowing users to input key game attributes (e.g., platform, genre, publisher) and instantly receive a prediction of whether the game is expected to achieve high or low sales.

The following chapters describe the data preprocessing, model selection, training, and performance evaluation process that led to the final deployment.

2. Data Loading and Preprocessing

In this step, the initial dataset containing video game information was loaded and filtered to include only the relevant columns: Platform, Genre, Publisher, and Global_Sales.

An outlier analysis was performed using boxplots and the IQR method to identify and remove extreme sales values. Additional filtering was applied to limit Global_Sales to more realistic ranges (below 1.0 and 0.4 million units).

The cleaned dataset was then saved and downloaded as a CSV file for further machine learning analysis.

3. Machine Learning – AutoGluon

The AutoGluon framework was used to automatically train and compare multiple classification models in order to identify the most effective one for predicting video game sales performance.
The target variable (Sales_Category) was created based on the median value of Global_Sales, categorizing each game as either good or bad.

AutoGluon evaluated several algorithms using f1-score as the main metric and provided a leaderboard ranking of model performance.

	model	score_test	score_val	eval_metric
0	LightGBM_BAG_L1	0.730904	0.734524	f1
1	WeightedEnsemble_L3	0.730904	0.734524	f1
2	WeightedEnsemble_L2	0.730904	0.734524	f1
3	LightGBMXT_BAG_L1	0.730416	0.730018	f1

	precision	recall	f1-score	support
bad	0.78	0.57	0.66	1693
good	0.65	0.83	0.73	1627
accuracy			0.70	3320
macro avg	0.72	0.70	0.70	3320
weighted avg	0.72	0.70	0.69	3320

Figure 3.2 Auto Gluon Results

The model evaluation results showed an overall accuracy of 70% and a macro F1-score of 0.70. The model performed better in identifying good sales (recall = 0.83) compared to bad sales (recall = 0.57), indicating an imbalance in predictive strength between the two categories.

The AutoGluon leaderboard revealed that the LightGBM_BAG_L1 model achieved the best performance with an F1-test score of 0.731, closely followed by the ensemble models (WeightedEnsemble_L2 and L3) with similar scores.

Although these results were acceptable, the limited ability of the model to distinguish between classes suggested that the input features did not capture all relevant relationships. Therefore, the next phase focused on feature engineering, aimed at enhancing the dataset with additional statistical and interaction-based variables to improve model accuracy and balance.

4. Feuture Engineering

To enhance model performance, several new features were generated.

Average sales were calculated for each Publisher, Genre, and Platform to capture historical success trends. Interaction features were created by combining categorical variables (e.g., Platform_Genre, Genre_Publisher) to represent cross-effects.

Rank-based features were also added to quantify the relative importance of each publisher, genre, and platform

The resulting dataset, enriched with these engineered features, was saved as video_game_sales_final.csv for model training.

Platform_Genre	Platform_Publisher	Genre_Publisher	Publisher_rank	Genre_rank	Platform_rank	Sales_Category
Wii_Sports	Wii_Nintendo	Sports_Nintendo	351.5	6004.5	4096.5	good
NES_Platform	NES_Nintendo	Platform_Nintendo	351.5	438.0	146.5	good
Wii_Racing	Wii_Nintendo	Racing_Nintendo	351.5	4240.0	4096.5	good
Wii_Sports	Wii_Nintendo	Sports_Nintendo	351.5	6004.5	4096.5	good
GB_Role-Playing	GB_Nintendo	Role-Playing_Nintendo	351.5	2892.5	49.0	good

	Platform	Genre	Global_Sales	Publisher	Publisher_avg_sales	Genre_avg_sales	Platform_avg_sales
0	Wii	Sports	82.74	Nintendo	2.563836	0.568247	0.705279
1	NES	Platform	40.24	Nintendo	2.563836	0.947577	2.561939
2	Wii	Racing	35.82	Nintendo	2.563836	0.593273	0.705279
3	Wii	Sports	33.00	Nintendo	2.563836	0.568247	0.705279
4	GB	Role-Playing	31.37	Nintendo	2.563836	0.628456	2.622887

Figure 4.1 Feuture Engineering

5. AutoGluon with Feuture Engineering

After applying feature engineering, the dataset was re-evaluated using the AutoGluon framework. The newly created features, such as average sales per publisher, genre, and platform, as well as interaction and ranking features, helped the model capture more complex relationships between categorical variables and sales outcomes.

	model	score_test	score_val	eval_metric
0	WeightedEnsemble_L2	0.741384	0.745154	f1
1	LightGBMXT_BAG_L1	0.740385	0.743804	f1
2	LightGBM_BAG_L1	0.735115	0.740987	f1
3	RandomForestGini_BAG_L1	0.731620	0.727751	f1
4	LightGBMXT_BAG_L2	0.730245	0.815142	f1
5	WeightedEnsemble_L3	0.730245	0.815142	f1

```
              precision    recall  f1-score   support

         bad       0.74      0.70      0.72      1649
        good       0.71      0.75      0.73      1609

    accuracy                           0.73      3258
   macro avg       0.73      0.73      0.73      3258
weighted avg       0.73      0.73      0.73      3258
```

Figure 5.1 AutoGluon with Feuture Engineering

After applying feature engineering, the model's performance slightly improved. The enhanced dataset, including average sales per publisher, genre, and platform, as well as interaction and ranking features, led to an accuracy of 73% and a macro F1-score of 0.73. Both precision and recall became more balanced (0.73 for each), showing that the new features allowed the model to better capture sales patterns across both categories.

However, the moderate improvement also reflects a limitation of the dataset itself. Since the input variables do not show strong linear or categorical correlations with global sales, the predictive power of any model remains inherently constrained. Despite this, the achieved results are considered satisfactory, demonstrating that the model can capture general sales tendencies even with limited feature–target relationships.

6. Model Deployment – LightGBM Integration with Streamlit

After identifying LightGBM as the best-performing model within the AutoGluon framework, the algorithm was retrained separately outside of AutoGluon. This approach reduced memory consumption, allowed full control over hyperparameters, and made it easier to export and reuse the model in external applications.

The retrained LightGBM classifier achieved comparable performance to the AutoGluon version, confirming the consistency and robustness of the selected model. The trained model was then serialized and saved as lightgbm_sales_classifier.pkl, including its key components, feature list, label mapping, and optimal decision threshold.

=== Classification report @ best threshold ===

	precision	recall	f1-score	support
bad	0.81	0.57	0.67	1646
good	0.66	0.86	0.75	1613
accuracy			0.71	3259
macro avg	0.74	0.72	0.71	3259
weighted avg	0.74	0.71	0.71	3259

Figure 6.1 LightGBM classifier

The retrained LightGBM model achieved an accuracy of 71% and a macro F1-score of 0.71, with the best balance found at an optimized decision threshold. The model performed strongly in predicting good sales (recall = 0.86, F1 = 0.75), while recall for bad sales (0.57) was slightly lower, indicating a minor imbalance in identifying low-performing games. Nevertheless, the overall results remain stable and consistent with the AutoGluon baseline.

The finalized model was saved as lightgbm_sales_classifier.pkl and integrated into a Streamlit web application, enabling real-time predictions. By entering Platform, Genre, and Publisher, users can instantly receive a forecast on whether a game is likely to achieve good or bad sales performance, completing the end-to-end predictive workflow.

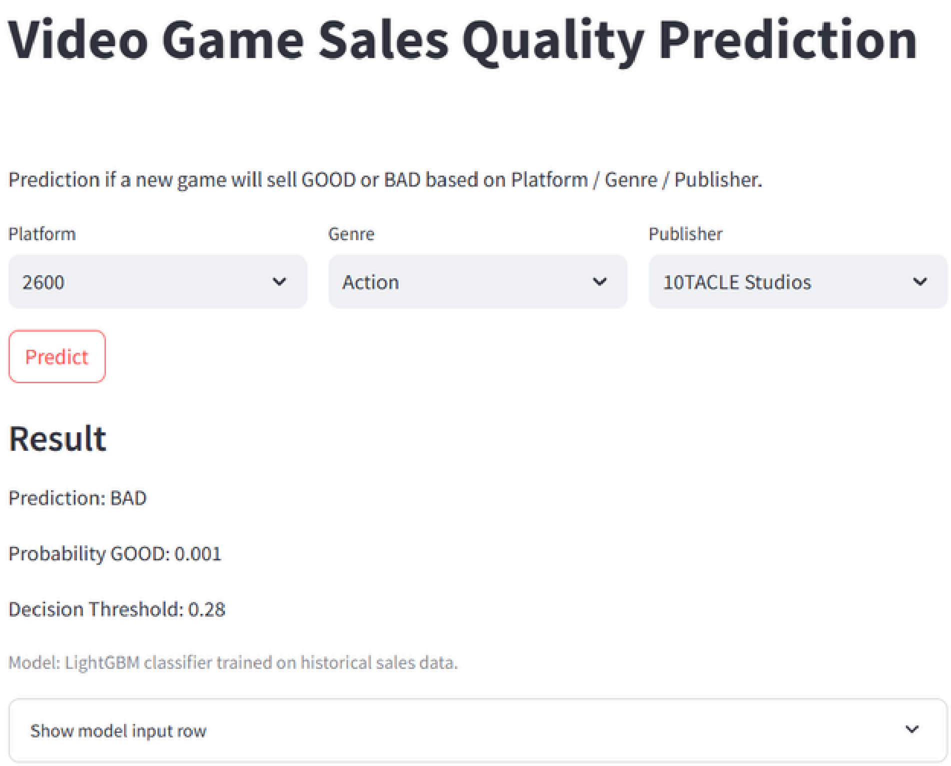

Figure 6.2 Streamlit APP