Content-Based Recommendation System (MovieRec) Case Study

Egemen Eroglu

November 13, 2025

1 Executive Summary

This project aimed to build and evaluate a series of content-based recommender systems using the Tag Genome dataset. We explored two primary paths:

1. **Qualitative Recommendation:** Generating the "best" list of recommendations for a target user (e.g. User 999999).
2. **Quantitative Prediction:** Building a Machine Learning model to accurately predict a user's 1- to 5-star rating for a movie.

Key Findings:

- **Best Qualitative Model:** The **Word2Vec (NLP)** model provided the most relevant, specific, and high quality recommendations, identifying the user's niche preferences.
- **Best Predictive Model:** The **CatBoost Regressor** was the clear champion in our offline evaluation, achieving the lowest Mean Absolute Error (MAE) of **0.9301**.
- **Conclusion:** We definitively proved that treating the 1-5 star rating as a **regression** problem is superior to treating it as a **classification** problem.

2 Findings: Qualitative Recommender Models

We built several models to generate a "Top 10" list for User 999999.

2.1 Initial Models and Noise Cleaning (TF-IDF)

Our first model, based on user reviews (TF-IDF), was polluted by generic and common words (*good*, *best*, *original*). This led to poor and seemingly random recommendations (e.g., horror movies for a comedy fan).

The most critical step of our project was **iterative feature engineering**. We used a bar chart of word frequencies to build a custom, aggressive list of **stop words**. This filtering was essential to find a clean signal.

2.2 Model 2: LSI/SVD (Topic Modeling)

By applying LSI to our clean TF-IDF matrix, we successfully denoised the data. This model correctly identified the user's preference for the broad **Comedy topic** in general, recommending films like *Minions* and *The Muppets*.

2.3 Model 3: Word2Vec (Review Semantics)

This was our most advanced and successful model. By learning the **semantic meaning** of words, it went beyond general topics to find the user's specific **"niche."** It correctly identified a strong preference for "Disney Channel Original Movies" (e.g., *Halloweentown II*, *Zenon: Z3*), providing the most relevant list.

3 Findings: Quantitative Predictive Models

We held a "bake-off" between numerous ML models to see which could most accurately predict the `targets` (1-5 star rating) from the data in `processed/10folds`.

3.1 Regression vs. Classification

We explicitly tested whether to treat the 1-5 stars as a regression or classification problem. The regression model was the clear winner.

- **Optimized XGBoost Regressor MAE:** 0.9631
- **Optimized XGBoost Classifier MAE:** 0.9900

This proves that the model performs better when it can understand the "distance" between ratings (i.e., that 4 is closer to 5 than 1 is).

3.2 The Model Bake-off

The `CatBoost Regressor` was the definitive champion, outperforming our tuned `XGBoost` model without any complex tuning.

Table 1: Final Predictive Model Comparison

Rank	Model	Model Type	MAE (Lower is better)
1.	**CatBoost Regressor**	Gradient Boosting	**0.9301**
2.	**XGBoost Regressor (Tuned)**	Gradient Boosting	**0.9631**
3.	**XGBoost Regressor (Original)**	Gradient Boosting	**0.9771**
4.	**Random Forest Regressor**	Bagging	**0.9848**
5.	**XGBoost Classifier (Tuned)**	Gradient Boosting	**0.9900**
6.	**XGBoost Classifier (Original)**	Gradient Boosting	**1.0299**
7.	**AdaBoost Regressor**	Boosting (Classic)	**1.0718**
8.	**Linear Regression (Baseline)**	Linear	**1.0872**

4 Final Conclusion

Our exploration was highly successful. We concluded that:

1. To **generate high-quality recommendations**, `Word2Vec` is superior because it understands the semantic *niche* of the content, not just keywords.
2. To **predict user ratings**, the `CatBoost Regressor` is the most accurate model, validating the power of modern gradient boosting.