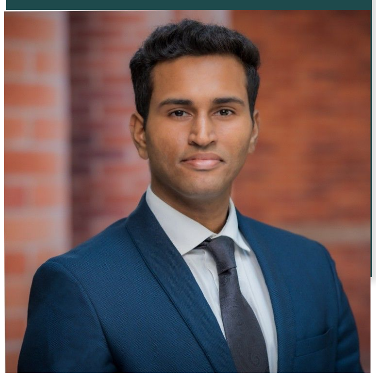
Predicting Kickstarter Success

Data Analytics and Machine Learning

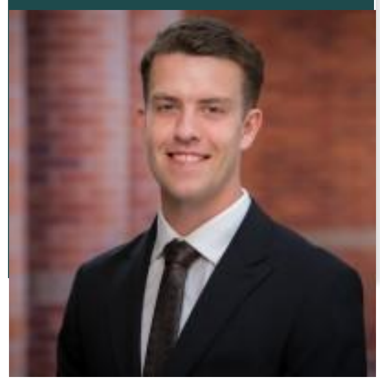
Meet the Team

Kushal Amilineni

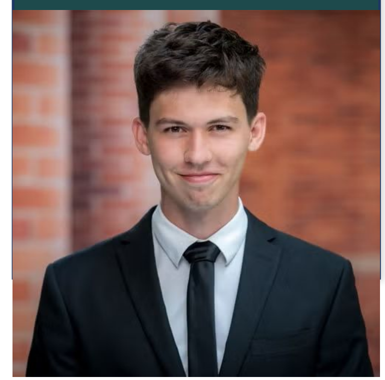


William Harrison

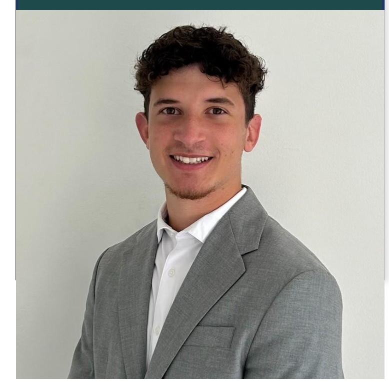
Sveinn Hauksson



David Pilik



Mathieu Jensen



Discussion Agenda

01
Motivation & Business Usefulness

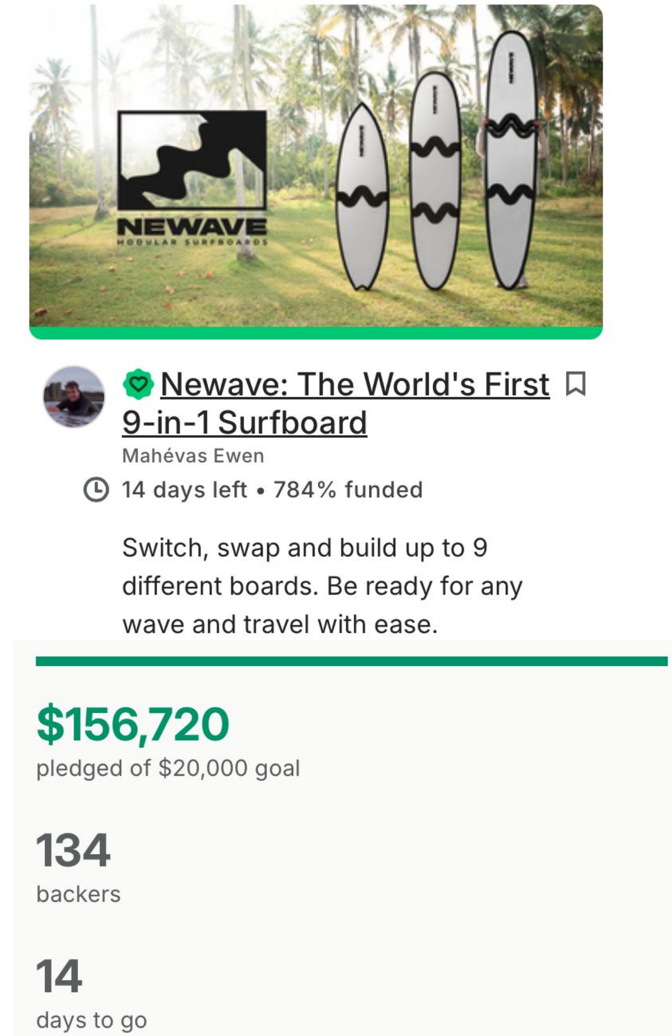
Why Predict Kickstarter Success?

What is Kickstarter

- → A leading crowdfunding platform where creators pitch projects directly to potential backers
- → Enables funding for creative ventures such as films, gadgets, games, and more

Campaign Properties

- → High failure rate - only about one in three campaigns reach their funding goal
- → Risk for creators - time, effort, and money spent on campaigns that never fund
- → Uncertainty for backers - difficulty distinguishing promising projects from less viable ones
- → Data-driven guidance - insights on optimal funding targets, duration, and launch timing
- → Strategic advantage - empower creators to make informed decisions and increase success odds



Who Benefits From This Study?

Project Creators

- → Optimize campaign parameters
- → Increase likelihood of funding success and avoid wasted efforts

Backers

- → Identify high-potential projects before marketplace hype
- → Allocate resources to campaigns with data-backed success odds

Platform Operators

- → Highlight promising campaigns to boost platform credibility
- → Improve overall success rates and user satisfaction

Researchers & Investors

- → Analyze patterns and trends in crowdfunding ecosystems
- → Develop new tools or services around predictive analytics

02
Data & Preparation

Understanding the Dataset

Source

→ Data extracted from Kickstarter (crowdfunding) projects

→ Contains 323,750 campaigns (rows) across all categories

→ Includes projects launched between approximately 2009–2017

Objective

→ Predict campaign outcome (state: “successful” vs “failed”) based on early project attributes

Key Takeaways

→ Large sample size which gives statistical power

→ Wide variety of product types (Music, Film & Video, Food, Publishing, etc.)

→ Majority of projects are “Success” or “Failure” so very little changes in data were needed

Core Variables and their type

Identifiers & Text

- → ID: unique integer per project
- → Name: project title (text)

Categorical Features

- → Category: "Narrative Film”
- → Main_Category: “Film & Video”
- → Currency: USD, GBP, EUR, NOK
- → Country: US, GB, CA

Numeric/Monetary

- → Goal: Funding goal
- → Pledged: Original Currency
- → USD Pledged: Converted to USD
- → Backers

Dates & Durations

- → Launched
- → Deadline
- → Duration_days: Deadline - Launched

New Feature Variables

- → log_goal_x_duration
- → category_sucess_rate
- → goal_per_day: goal / duration

03
Machine Learning Modeling

4 Classes of models

Class 1: Base Models

- → Logistic Regression
- → Random Forests
- → XGBoost
- → Decision Tree

Class 2: New Feature Engineering

- → Logistic Regression
- → Random Forests
- → XGBoost

Class 3: Data Upscaling & Advanced Models

- → Lasso Logistic Regression
- → Stacked Ensamble (LightGBM + RF + LogReg)
- → Random Forests
- → XGBoost

Class 4: Text Analysis

- → LightGBM
- → Stacked Ensamble (LightGBM + RF + LogReg)
- → XGBoost

Class 1

- Try base models for a performance baseline and feature importance
- Identify best and worst predictors
- Get ideas to move forward

Models

Logistic regression ✔	• Easy to interpret • Fast • Not great with nonlinear patters
Random Forest 1 ✔	• Non-linear model baseline
Random Forest 2 ✔	• RF with balanced weights
Random Forest 3 ✔	• RF without log_goal and duration • Testing feature dependence
Random Forest 4 ✔	• RF with only log_goal and duration
Random Forest 5 ✔	• GridSearchCV for hyperparameter tuning.
XGBoost ✔	• Upgrade RF • Sequential boosting, • More control
Decision Tree ✔	• Get insights • Not performance

Class 1: Top models

1st

XGBoost

2nd

Random Forest
GridSearchCV

3rd

Random Forest
Balanced Class Weight

4th

Logistic Regression

Logistic Regression

	precision	recall	f1-score	support
0	0.68	0.82	0.74	33553
1	0.61	0.42	0.50	22708
accuracy			0.66	56261
macro avg	0.64	0.62	0.62	56261
weighted avg	0.65	0.66	0.64	56261

ROC AUC Score: 0.6918119942756729

Report Insights

→ Predicts failed projects better than

→ It struggles to identify successful projects

→ Good for baseline

Average F1 Score

0.620

Model Insight

→ Logistic Regression is a linear

→ Interpretable model for binary outcomes

→ Good simple a baseline model

→ Not great at capture nonlinear patterns

Random Forest
Balanced Class Weight

	precision	recall	f1-score	support
0	0.73	0.65	0.69	33553
1	0.56	0.65	0.60	22708
accuracy			0.65	56261
macro avg	0.65	0.65	0.64	56261
weighted avg	0.66	0.65	0.65	56261

Random Forest ROC AUC: 0.706908681582278

Report Insights

→ Improved success prediction

→ Recall same for both

→ Higher ROC AUC (0.71) shows better probability ranking

Average F1 Score

0.645

Model Insights

→ Random Forest is an ensemble of decision trees

- Reduces overfitting
- Captures **nonlinear patterns**

→ Adjusts for class imbalance

- Gives more weight to minority class errors

Random Forest
GridSearchCV

	precision	recall	f1-score	support
0	0.74	0.65	0.69	33553
1	0.56	0.66	0.60	22708
accuracy			0.65	56261
macro avg	0.65	0.65	0.65	56261
weighted avg	0.67	0.65	0.66	56261

ROC AUC Score: 0.709128454809212

Report Insights

→ Very close to the untuned Random Forest

→ Success recall improved slightly

→ Good for baseline

Average F1 Score

0.645

Model Insights

→ Tests multiple depths to find the best combo
  - but gains were marginal.

→ Takes a long time

→ Good simple a baseline model

→ Not great at capture nonlinear patterns

XGBoost

	precision	recall	f1-score	support
0	0.747	0.636	0.687	33553
1	0.559	0.682	0.614	22708
accuracy			0.654	56261
macro avg	0.653	0.659	0.650	56261
weighted avg	0.671	0.654	0.657	56261

XGBoost ROC AUC: 0.7191564239101349

Report Insights

→ Best ROC AUC and F1 score so far

→ Balanced performance

→ F1 for success improved

Average F1 Score

0.6505

Model Insights

→ Gradient boosting method

- Corrects past mistakes by building trees sequentially

→ Strong predictive performance in tabular data

Class 2

1st

XGBoost

2nd

Random Forest
GridSearchCV

3rd

Logistic Regression
Tuned Threshold

Random Forest
GridSearchCV

	precision	recall	f1-score	support
0	0.734	0.667	0.699	33645
1	0.564	0.640	0.599	22616
accuracy			0.656	56261
macro avg	0.649	0.653	0.649	56261
weighted avg	0.665	0.656	0.659	56261

ROC AUC Score: 0.7138627804208226

Report Insights

→ F1 for failures (0.70), highest across all models

→ ROC AUC of 0.71+

→ Success detection is solid (F1 = 0.60)

Average F1 Score

0.6490

Model Insights

→ Same model as in Class 1

→ Benefits from controlled tree depth & min splits

→ Takes a while

XGBoost

	precision	recall	f1-score	support
0	0.748	0.639	0.689	33645
1	0.559	0.681	0.614	22616
accuracy			0.655	56261
macro avg	0.653	0.660	0.651	56261
weighted avg	0.672	0.655	0.659	56261

XGBoost ROC AUC: 0.7184573232143624

Report Insights

→ Improved success recall (0.68)

→ Excellent failure detection (F1 = 0.69)

→ Most balanced model so far

Average F1 Score

0.6515

Model Insights (Same model)

→ Gradient boosting method

- Corrects past mistakes by building trees sequentially

→ Strong predictive performance in tabular data

Class 3: All worse than Class 2

XGBoost

Random Forest
GridSearchCV

Lasso Logistic Regression

Class 4

XGBoost

LightGBM+RF+LogReg

LightGBM

LightGBM

	precision	recall	f1-score	support
0	0.706	0.810	0.755	33645
1	0.639	0.499	0.560	22616
accuracy			0.685	56261
macro avg	0.673	0.655	0.658	56261
weighted avg	0.679	0.685	0.677	56261

ROC AUC Score: 0.736031545533871

Report Insights

→ Best ROC AUC of all models so far (0.736)

→ Excellent failure prediction (F1 = 0.755)

→ Weak recall on successes (0.499)

Average F1 Score

0.6575

Model Insights

→ Gradient boosting framework

→ Fast performance on large datasets

→ Ideal for structured/tabular data

LightGBM+RF+LogReg

	precision	recall	f1-score	support
0	0.710	0.802	0.753	33645
1	0.635	0.514	0.568	22616
accuracy			0.686	56261
macro avg	0.673	0.658	0.661	56261
weighted avg	0.680	0.686	0.679	56261

ROC AUC Score: 0.7369901830863387

Report Insights

→ Great for failures, with F1(0) = 0.753

→ Bad success prediction, F1(1) = 0.568

→ Strong ROC AUC (0.737)

Average F1 Score

0.6605

Model Insights

→ Combines multiple base models

→ Complementary strengths

→ Boost performance when base models differ

XGBoost

	precision	recall	f1-score	support
0	0.755	0.656	0.702	33645
1	0.572	0.683	0.623	22616
accuracy			0.667	56261
macro avg	0.664	0.670	0.663	56261
weighted avg	0.682	0.667	0.670	56261

ROC AUC Score: 0.7335472276993976

Report Insights

→ The best model

→ Good balance

→ ROC AUC 0.734

Average F1 Score

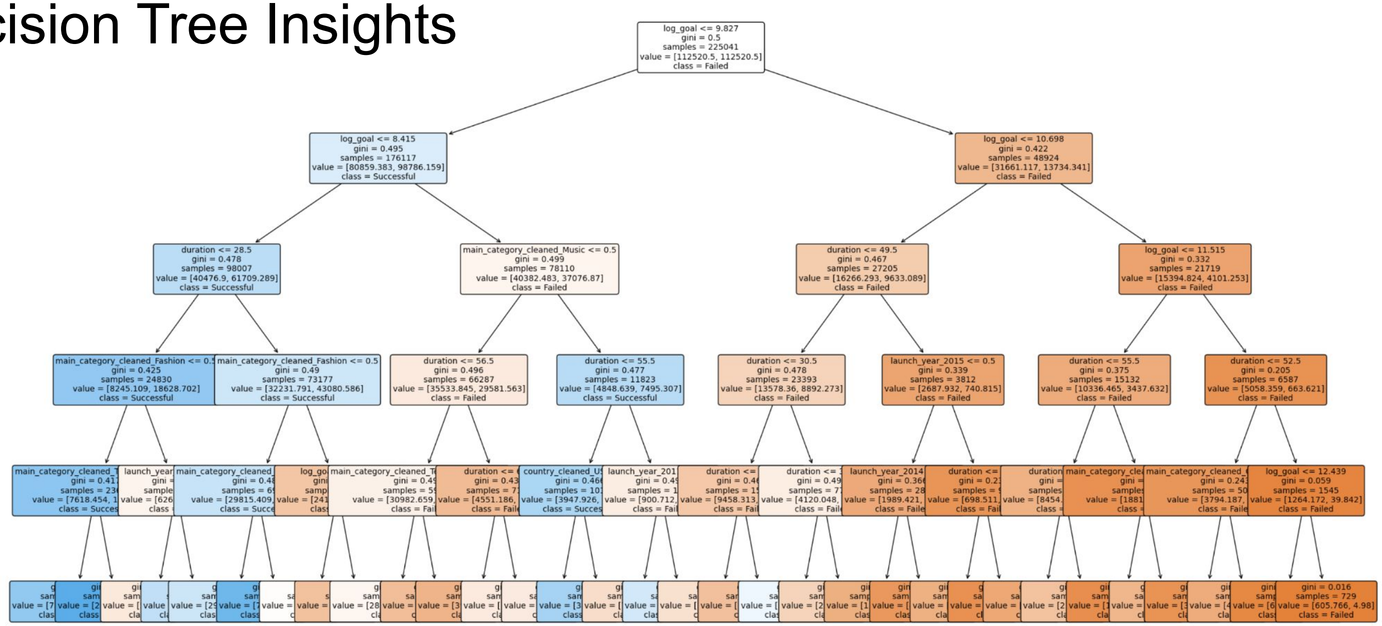
0.6625

Model Results Summary

Rank	Model	Class	Avg F1	F1 (Success)	F1 (Failure)	ROC AUC
1	XGBoost (with TF-IDF)	4	0.6625	0.631	0.694	0.734
2	LightGBM + RF + LogReg (Stacked)	4	0.6605	0.568	0.753	0.737
3	LightGBM	4	0.6575	0.499	0.755	0.736
4	XGBoost	2	0.6515	0.680	0.690	0.715
5	Random Forest (GridSearchCV)	2	0.6490	0.600	0.700	0.710

04
Key Patterns & Visual Insights

Class 1 Decision Tree Insights



5 Kickstarter Paths

Path 1
Low goal
Short duration
Broad category

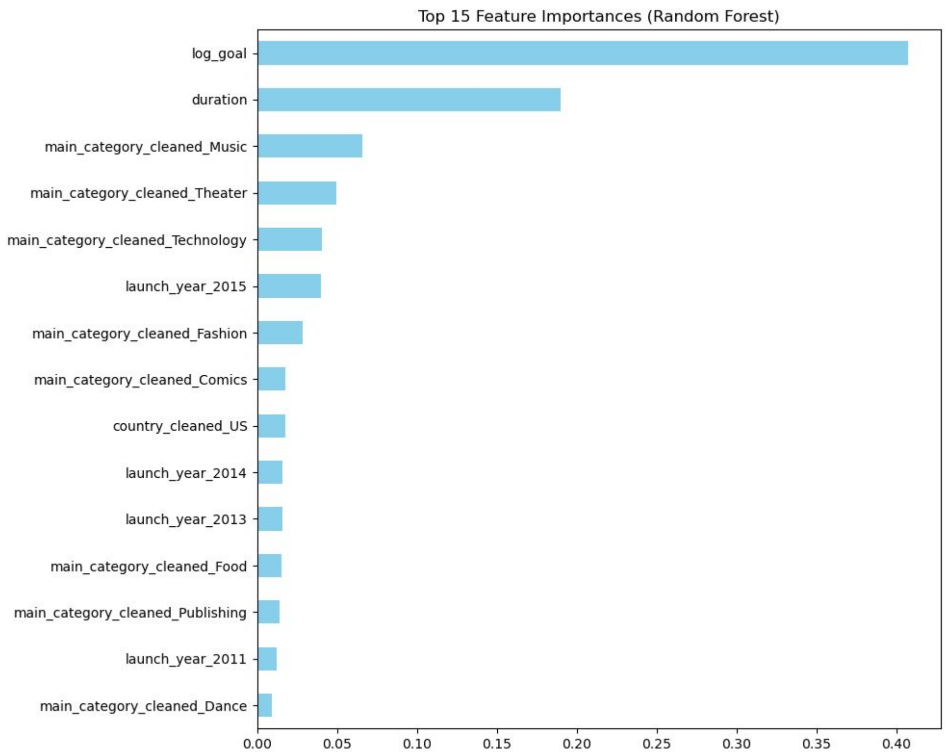
Path 2
Low goal
Music category

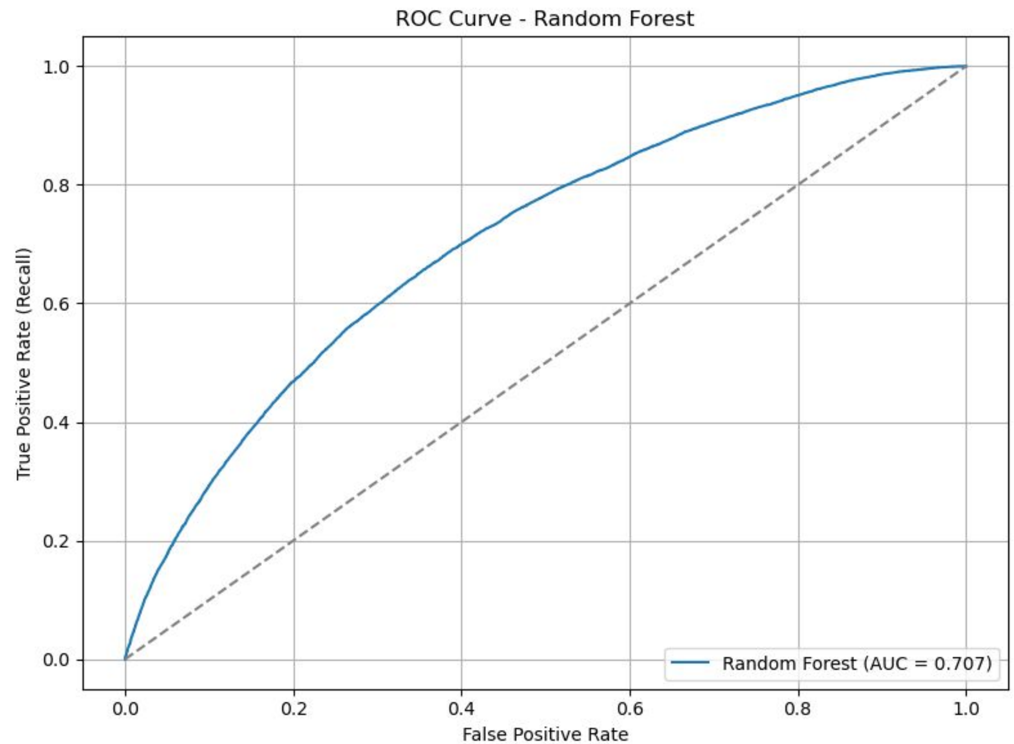
Path 3
High Goal
Long Duration
2016 Launch

Path 4
High Goal
Tech/Photography

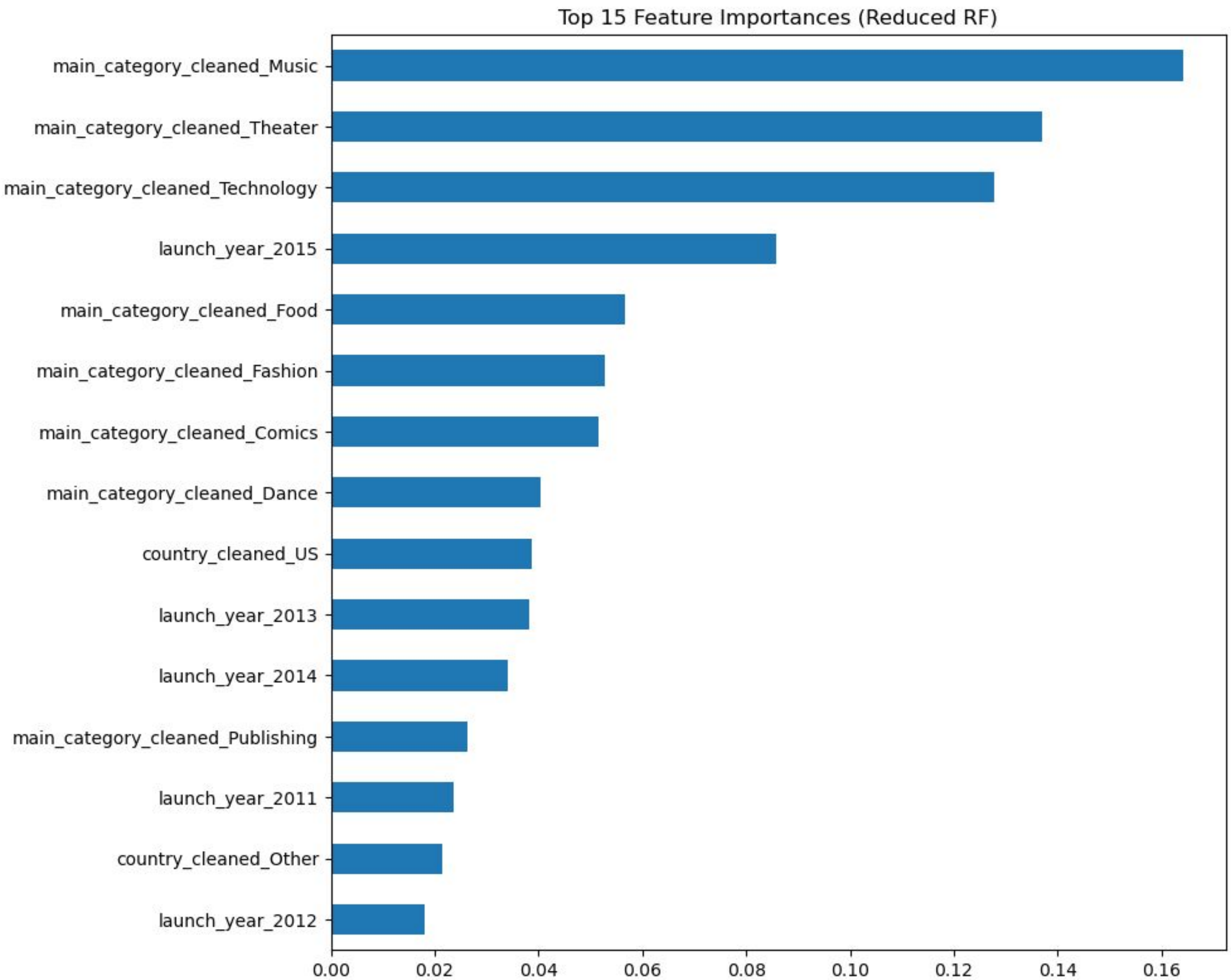
Path 5
Medium goal
Short duration
US-Based

Class 1: Random Forest

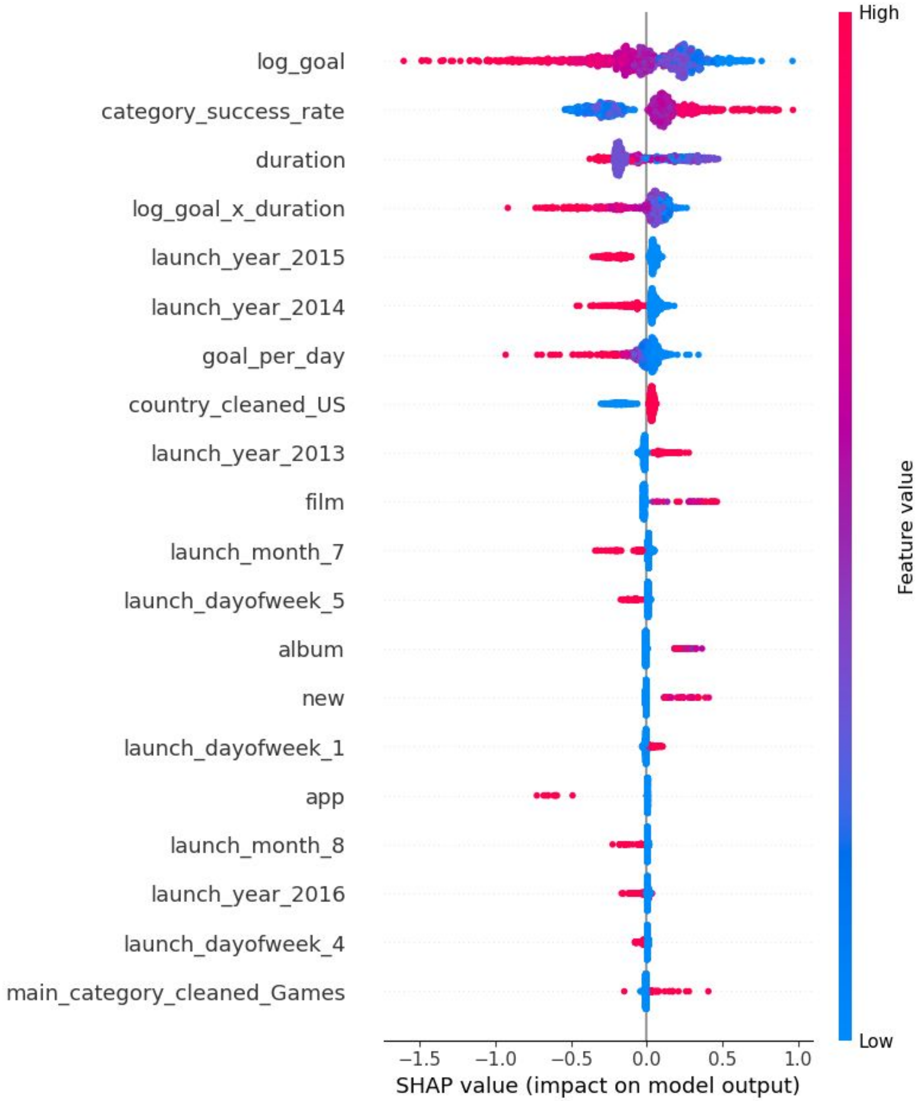


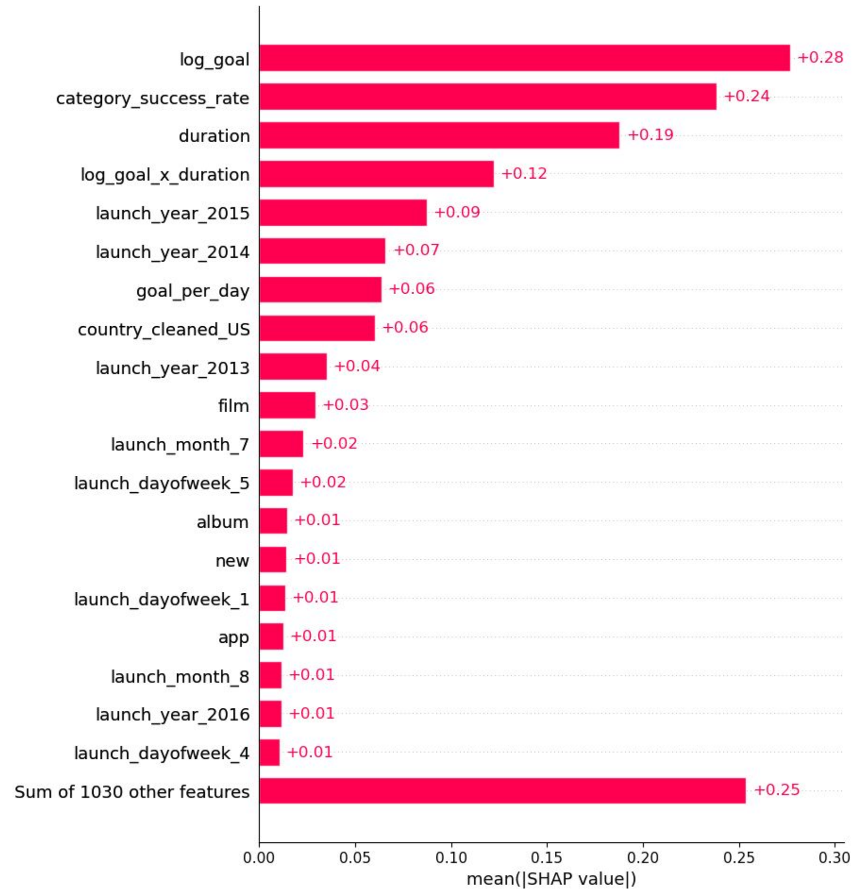


Class 1: Random Forest



Class 4: XGBoost with TF-IDF





Class 4: XGBoost with TF-IDF

🔤 Top 20 Word Features by SHAP:

	word	mean_abs_shap
327	film	0.029278
33	album	0.014634
604	new	0.014194
52	app	0.012675
56	arduino	0.009013
106	book	0.007590
254	documentary	0.007368
193	com	0.007250
235	debut	0.006895
720	record	0.006384
973	wireless	0.005580
943	volume	0.005521
790	short	0.005187
118	brewing	0.004350
226	dance	0.003863
931	video	0.003701
513	length	0.003578
507	leather	0.003354
547	magnetic	0.003340
356	funding	0.003196

05
Recommendations & Conclusion

Recommendations For Campaign Creators

Data-driven strategies to boost your campaign's chance of success

What Works?

Set Realistic Funding Goals	Campaigns with moderate goals succeed more often
Optimize Campaign Duration	Campaigns with a conservative timeline tend to perform best
Launch in Right Macro Conditions	Launching in a good economy has a positive effect on campaign success
Craft Clear and Engaging Titles	Strong, action-oriented titles correlate with success
Choose High-Performing Categories	Campaigns in Games, Design, and Technology have higher success rates
Build Early Momentum	Fast early pledges strongly predict overall success

Why These Recommendations?

→ Our machine learning models reveal that small tweaks in campaign setup like launch timing, goal setting, and communication; significantly shift success odds

→ Combining data-driven insights with intuitive design choices gives creators a measurable edge

Bonus Tip

→ Leverage predictive tools: Predictive models like ours can help creators pre-test their campaign setups and optimize before launch

→ Use Strong Visuals and Media: Campaigns with high-quality images and videos have much higher engagement and funding rates

Conclusions & Business Insights

✔ **1. Machine Learning Effectively Predicts Kickstarter Success**
Our models had an average F1 score of 0.6625, showing strong predictive power using campaign data and text features

✔ **2. Text Features Boost Predictive Accuracy**
Incorporating TF-IDF on campaign titles and summaries improved model performance, revealing the importance of strong messaging

✔ **3. Key Drivers of Success Are Actionable**
Goal size, campaign duration, launch timing, and category selection emerged as critical factors

✔ **4. Practical Insights for Stakeholders**
Creators can optimize campaign design, while platforms and backers can better identify high-potential projects

Our study demonstrates how data-driven strategies can materially increase success rates on crowdfunding platforms

Thank You