Launch Hard or Go Home!

Predicting the Success of Kickstarter Campaigns

Vincent Etter, Matthias Grossglauser, Patrick Thiran

LCA 3/4
What is Kickstarter?

- **Crowdfunding** website
- Raise money to launch creative projects
- Each campaign has:
  - a funding **goal**
  - a campaign **duration**
- **All-or-nothing** funding model
A Success Story

• Since its launch in 2009:
  • 51,000 projects successfully funded
  • 44% success rate
  • $871,000,000 raised
  • 5,100,000 backers
Our Dataset

• Discover campaigns on the "Recently Launched" page

• Regularly get their status: number of backers, amount of money pledged

• Monitor Twitter in parallel for all tweets containing keyword kickstarter
Our Dataset

- From September 2012 to May 2013:
  - **16,042** campaigns
  - **48.24%** of successful campaigns
  - **1,309,295** backers
  - **$158,026,656** pledged
  - **737,398** "kickstarter" tweets
Preprocessing

• **Time** is normalized with respect to campaign **duration**

• Pledged **money** is normalized with respect to campaign's **goal**

• Statuses are **resampled** to have 1000 samples of each trajectory
Trajectories Overview
Our Predictors

• **Binary** classification: successful or failed?
• **Baseline**: campaigns' static attributes [1]

Two sources of information:

• **Money**: kNN, Markov
• **Social**: tweets, project/backer graph

Campaign Progress

• Predictions need to be made at different states of **progress**
• could be another **feature**
• or have **different** predictors
• We chose to build **separate** predictors
k Nearest Neighbors

• Take the \textit{series of amounts} of pledged money up to time $t \in [0, 1]$

• Find the \textit{k closest} projects

• Use \textit{proportion} of successful neighbors as probability of success
Markov Predictor

- **Discretize** time and money
- Define (time, money) as *state space*
- Express campaign trajectory as a *Markov Chain:*

\[ P_{m,m'}(i) = \mathbb{P}(M_{i+1} = m' \mid M_i = m) \]
Markov Predictor
Tweets Predictor

• Use only tweets about a campaign

• Extract several features:
  • # of tweets/retweets/replies
  • # of users who tweeted
  • Estimated # of backers

• Train a Support Vector Machine
Graph Predictor

- Can we use **backers** as source of information?

- Consider the project/backer graph:
Graph Predictor

- Extract several features:
  - # of backers
  - #:prop. of first-time backers
  - # projects with co-backers
  - #:prop. of these that are successful

- Train a Support Vector Machine
Evaluation

- **10-fold** cross-validation
- **70%** of samples for training
- **20%** of samples for validation
- **10%** of samples for testing
- **Median accuracy** and standard deviation over test set are reported
Prediction Accuracy

- kNN
- Markov
- Graph
- Baseline
- Tweets
Combined Predictor

• Money-based results are quite good
• Can we use social predictors to help?
• Train a SVM to combine individual predictions
Combiner Results

![Graph showing prediction accuracy vs. campaign progress with a highlighted Combiner line.]
Combiner Results
Conclusion

• Combiner improves *early* predictions

• This is where they are the most *useful*

• There is *potential* in social predictors
Future Work

- Improve graph predictor
- Analyze results in details
  - Typical false negative/positive
  - Important features
- Take dynamics on Twitter and project/backer graph into account
Thank you!

Data and real-time predictions on
http://sidekick.epfl.ch