KickUpper: A Tool For Making Better Crowdfunding Projects

Text Information Systems - Project Report

Amirhossein Aleyasen

University of Illinois at Urbana-Champaign

aleyase2@illinois.edu

Spring 2014

1 Introduction

The rise of crowdsourcing as a framework of harnessing crowd power leads entrepreneurs to use this power in building new ventures. This approach, named crowdfunding, has attracted attentions during recent years. Kickstarter [1], IndiGoGo [2] and RocketHub [3] are some of these crowdfunding platform. Increasing the prevalence of crowdfunding platforms, researchers started to explore the reason of success or failure a project in online crowdfunding project. This research has been extended to the enterprise level too [4]. Some researches have used non language features of a project in crowdfundig platforms such as project duration, connection to social networks and including a video to predict a project funding status [5, 6, 7, 8]. In addition to project features, some researches investigated the effect of updates in a project success [9]. However, all of these researches consider non-language based features of a project to determine its success or failure. A recent study [10] started to use some language features such as words and small phrases frequency by adding them to the current models to build an enhanced model. However, this research did not consider the effect of more large or sentences structure in success of a crowdfunding project.

In this project, we built a model in which the description of a project is taken into account to predict its success. To do so, we used a dataset of about 46K KickStarter projects to build a classifer based on n-grams (n = 3, 4, 5). Our classifier is able to predict success or failure a project by 73.7% f-measure, just based on its description text. We believe adding non-language features to this model would increase its performance. In the following, we start by explaining our method to build the classifier. Then, we demonstrate the results and conclude with some suggestions for future work. In addition we provide a web interface that allow users to enter their project descriptions and it provides feedback to users based on the classifier results.

2 Method

To build a classifier for a project based on its description language, we used a list of $45,815$ Kickstarter projects which includes the projects launched from June 2, 2012 [11]. Since only the projects that reach to their funding goal by their end date will receive their pledged money, the funding principle is based on all or nothing. Therefore, each project has a funded or unfunded status. In our dataset, 51.53% were funded while 48.47% were not funded. This balance helped us to have a fairly statistical analysis. The steps of building our classifier have been explained in the following sections.

2.1 Data Preprocessing

In order to preprocess the data, we first scraped the project pages to extract their description and funding status. To do so, we used jsoup [12], a java library for parsing HTML files. Then, we converted the text to lowercase and removed non letter and non digit characters. We also divided our dataset to a training (= 40K) and test set (= 5815) for using in the next steps.

2.2 Named Entity Extraction

Since just structure of the sentences were important for us not the proper names that used in the sentences, we extracted named entities of training set projects' descriptions using Stanford Named Entity Recognizer (NER) [13] for generalizing the sentences. We extract entity types: date, location, organization, person, time and money. We decided to replace the specific names and categories found here with general ones. This replacement would help to have a more generalized model based on the type of entities.

2.3 Tokenizing n-grams

In this project, instead of considering words and bigrams as the main pieces of the language, we considered larger and more complex structures to predict a project success. The main reason of doing so was the importance of longer structures in language in conveying a concept. While words or small phrases frequency might help in predicting a project success, it would not be such useful in helping an entrepreneur to write a successful project description. In other words, if someone wanted to write a project description, knowing to use what type of structures instead of words would help them more in creating an appropriate project description. Therefore, after preprocessing the projects, we tokenized each project description of the training set to every possible n-grams ($n = 3, 4, 5$). Then, we aggregate unique n-grams over all the training set projects by considering their frequency in funded and unfunded projects. Table 1 shows the frequency of n-grams before and after aggregation among the training dataset.

2.4 Index Creation

After tokenizing n-grams, we created index for these n-grams by using Lucene search engine [14]. To do so, we first traversed all the n-grams and removed the ones that were repeated less than 5 times. By doing so, we wanted to be sure exclude the specific n-grams which appeared in a few documents. Then, for

n-gram	before aggregation #	after aggregation #
3-gram	15,440,11	8,112,511
4-gram	14,620,090	11,351,863
5-gram	13,814,636	12,366,453

Table 1: The frequency of n-grams before and after aggregation in training dataset

the remained n-grams, we converted each of them to a document and use Lucene search engine to index these documents. Each indexed document has the related n-gram as its content (which is searchable) and three features: doc ID, frequency of funded and unfunded projects' descriptions the related n-gram appeared on them, respectively.

2.5 K-Nearest Neighbors

To build a classifier, we used KNN to vote for the funding status of a test project's description. To do so, we tokenized the test project's description to n-grams such as we have done for training projects' descriptions. Then, for each n-gram, we used Lucene search engine to find top similar n-grams (k = 15) to it. Then, based on these n-grams, we used KNN to determine the funding status of the input n-gram. We repeat this procedure for all the n-grams of a test project's description and calculate the average score of the description based on the scores of these n-grams. Based on the average score, we determined the funding status of the description.

3 Results

In order to evaluate our classifier model, we measure the precision, recall and its f-measure in identifying the funding statuses of the test set projects. To understand the effect of ngrams size on the performance, we have done multiple experiments on the different combinations of extracted n-grams size from training set and test set. Table 2 shows the results of our classifier over 5815 test projects. As the results show, the performance of the classifier is almost the same over different n-gram sizes. F-measure 73.7% is the best result for the classifier in predicting test projects' funding status.

Since there is no previous method which just used language of a project to predict its success, there

Training n-grams	**Testing n-grams**	**Precision**	**Recall**	**F-measure**
3-grams	3-grams	61.1%	92.9%	**73.7%**
4-grams	3,4-grams	60.9%	88.9%	72.3%
4-grams	4-grams	62%	89.4%	73.2%
5-grams	4,5-grams	58.2%	90.7%	70.9%

Table 2: The Performance of the classifier over different combinations of n-grams

was no baseline method to compare our classifier with. However, this project showed that language has an important effect in the success of a project. Therefore, we believe adding more language features to the existing models would improve their accuracy significantly.

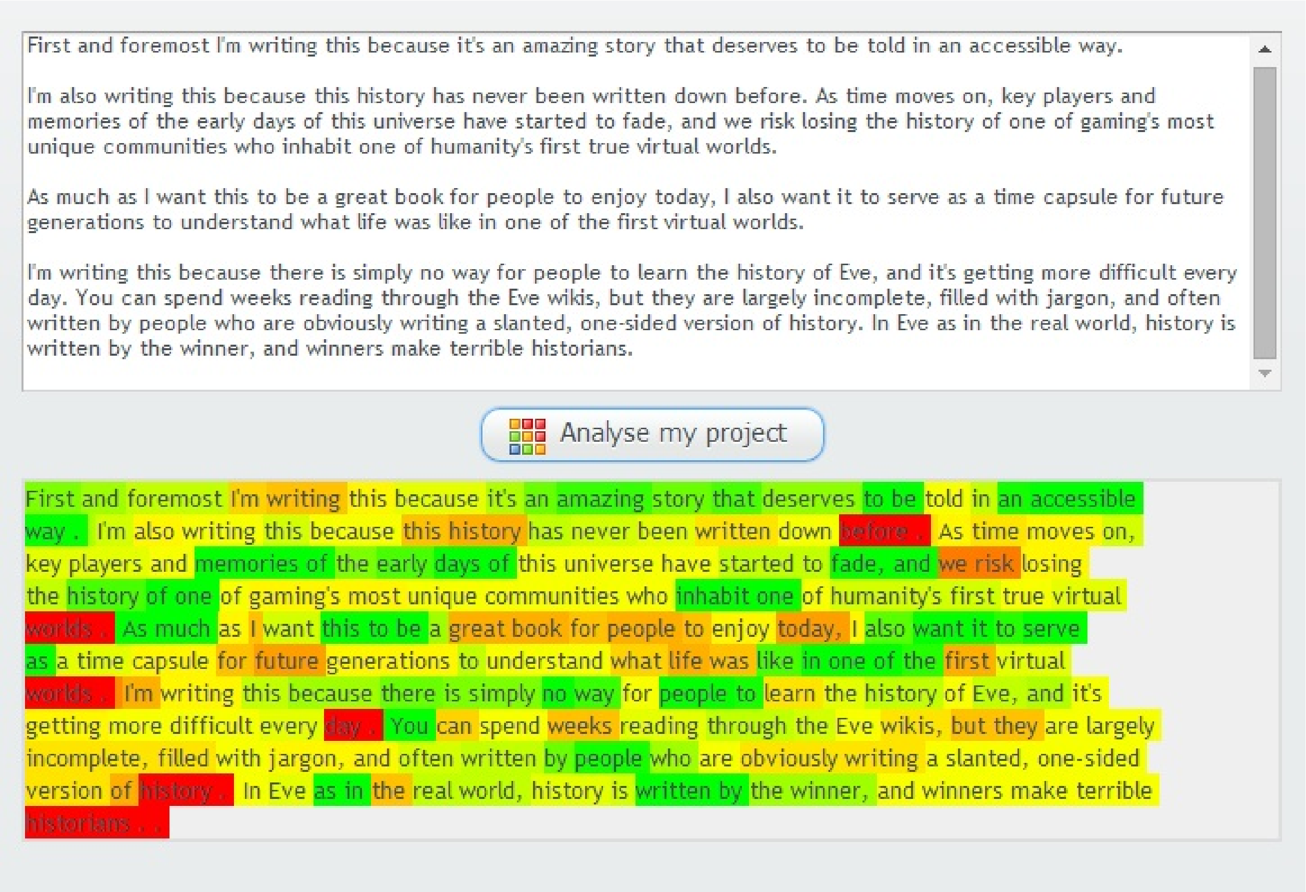
3.1 Web Interface

As it was mentioned earlier, the goal of building this classifier was not only predicting the success of a project, but also help an entrepreneur to write their project's description with more insight. Therefore, we designed an web interface [1], which an entrepreneur can enter their project's description and the application identifies which phrases sounds appropriate or not based on the phrases in the training dataset. To do so, we colored each word based on its score in our classifier from dark green (very appropriate) to dark red (not appropriate). Figure 1 shows an example of this analysis over a real KickStarter project.

4 Conclusion

While recent researches started to investigate different factors' effects on crowdfunding projects' success, we still know little about the effect of language structure on these projects' success. Here, we proposed a classifier to consider the language specifically n-grams in predicting the funding status of a project. Achieving 73.7% f-measure, we believe adding non-language based features would increase this accuracy. We also suggest to consider sentence structures in future models to reach higher accuracy. Finally, we believe in addition to predicting success of a project, the methods should be able to suggest what the project author should do to improve it. To do so, we suggest using application such as KickUpper to help

[1] http://palm.cs.illinois.edu:8080/kickupper/



First and foremost I'm writing this because it's an amazing story that deserves to be told in an accessible way.

I'm also writing this because this history has never been written down before. As time moves on, key players and memories of the early days of this universe have started to fade, and we risk losing the history of one of gaming's most unique communities who inhabit one of humanity's first true virtual worlds.

As much as I want this to be a great book for people to enjoy today, I also want it to serve as a time capsule for future generations to understand what life was like in one of the first virtual worlds.

I'm writing this because there is simply no way for people to learn the history of Eve, and it's getting more difficult every day. You can spend weeks reading through the Eve wikis, but they are largely incomplete, filled with jargon, and often written by people who are obviously writing a slanted, one-sided version of history. In Eve as in the real world, history is written by the winner, and winners make terrible historians.

First and foremost I'm writing this because it's an amazing story that deserves to be told in an accessible way . I'm also writing this because this history has never been written down before . As time moves on, key players and memories of the early days of this universe have started to fade, and we risk losing the history of one of gaming's most unique communities who inhabit one of humanity's first true virtual worlds . As much as I want this to be a great book for people to enjoy today, I also want it to serve as a time capsule for future generations to understand what life was like in one of the first virtual worlds . I'm writing this because there is simply no way for people to learn the history of Eve, and it's getting more difficult every day . You can spend weeks reading through the Eve wikis, but they are largely incomplete, filled with jargon, and often written by people who are obviously writing a slanted, one-sided version of history . In Eve as in the real world, history is written by the winner, and winners make terrible historians . .

Figure 1: An example of language analysis over a real KickStarter project description by KickUpper applications

an entrepreneur in achieving their goal.

5 Acknowledgment

With special thanks to Motahareh Eslamimehdiabadi, PhD Candidate in Computer Science Department, for her ideas for developing the model and classifier and also revising the report.

References

[1] "Kickstarter." https://www.kickstarter.com/.

[2] "Indiegogo." https://www.indiegogo.com/.

[3] "Rockethub." http://www.rockethub.com/.

[4] T. S. S. D. . L. C. M. Muller, W.r Geyer, "Crowdfunding inside the enterprise: Employee-initiatives for innovation and collaboration," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI 13, ACM, 2013.

[5] K. H. M.D. Greenberg, B. Pardo and E. Gerber, "Crowdfunding support tools: Predicting success and failure," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI EA '13, ACM, 2013.

[6] I. K. B. S. K. Chen, B. Jones, "Kickpredict: Predicting kickstarter success," in *Dept. of Computing and Mathematical Sciences, California Institute of Technology*.

[7] E. Mollick, "The dynamics of crowdfunding: An exploratory stud," *Journal of Business Venturing*, 2013.

[8] M. G. V. Etter and P. Thiran, "Launch hard or go home! predicting the success of kickstarter campaigns," COSN13, ACM, 2013.

[9] H. R. W. F. S. H. B. B. A. Xu, X. Yang, "Show me the money! an analysis of project updates during crowdfunding campaigns," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, ACM, 2014.

[10] T. Mitra and E. Gilbert, "The language that gets people to give: Phrases that predict success on kickstarter," in *Proceedings of Computer Supported Cooperative Work*, CSCW '14, ACM, 2014.

[11] P. Jeanne, "Kickstarter failures revealed! what can you learn from kickstarter failures?," Retrieved Sep. 19, 2012. http://www.appsblogger.com/kickstarter-infographic.

[12] "jsoup: Java html parser." http://jsoup.org/.

[13] "Stanford named entity recognizer." http://nlp.stanford.edu/software/CRF-NER.shtml.

[14] "Apache lucene." http://lucene.apache.org/.