Micro-Ventures

Predicting the Success of Potential Startups for Micro-Investments

1. Motivation and Background

A staggering 100 million businesses are launched annually, according to figures from GEM Global Report. **Struggling to grasp such a big number?** It comes to just over three businesses every second, or 11,000 per hour. What happens to all these ventures, you might wonder? Unfortunately **90 per cent of them will fail**, but that doesn't stop a rather impressive amount of money being thrown at them. In the US alone, a mind-boggling **$1,532 in venture capital is invested every second. Per year, that adds up to $48.3bn**.

1.1. What is Micro Venture Capital?

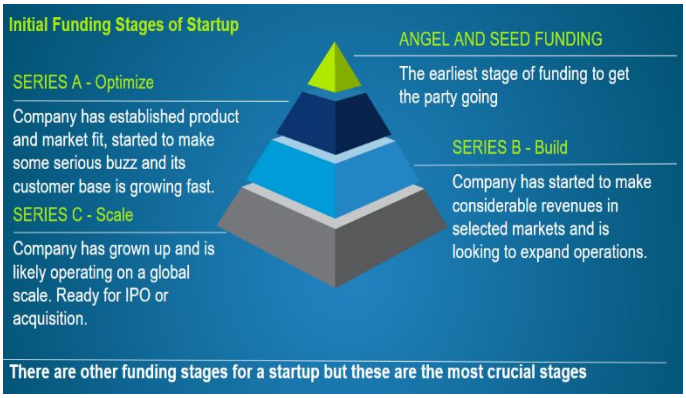
Micro venture capital is money invested to in early-stage emerging startups with amounts of finance that is typically less than that of traditional venture capital. In contrast to traditional venture capital which is money used to invest in companies looking to fund growth, micro venture capital consists of smaller seed investments, typically between $25K to $500K, in companies that have yet to gain traction. Small investors who do not have large investment are always looking for startup with potential to succeed, with major motivation to get high returns for equity when a company gets IPO or gets acquired. **A successful investment for a millennial is one in which a company gets acquired or get IPO (Initial Public Offering), likewise a company getting closed is a total loss.**

1.2. Related Work

Currently, there is one company called Microventures, which is an equity crowdfunding website offering investments in early stage companies for these millennial. **But what if we could use machine learning to predict a startup success?** Many researchers have tried to predict the acquisition, merger or IPO of companies from data at early stages, strategy usually adopted is to use financial and managerial data. One of the research that we followed in our project is **"A supervised approach to predict company acquisition with factual and topic features using Profiles and News Articles on Techcrunch"[1].** However, we take a different approach, ***we bind the company online presence with financial and managerial data to classify a potentially successful startups not to predict acquisition or IPO but to predict if it will reach Series C or beyond***.

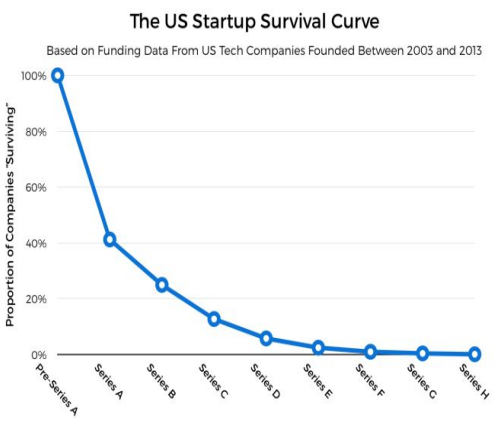
2. Problem Statement

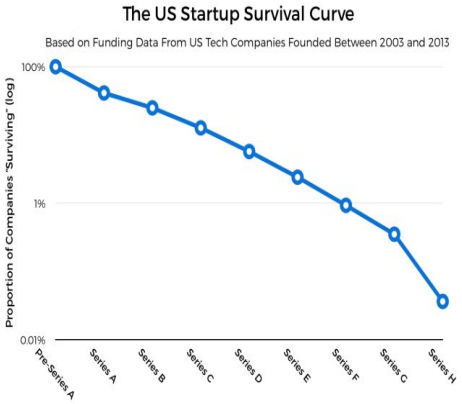
Predicting startup that might have IPO or acquisition after some years of their existence might not be relevant to small investors, as such a company would have already passed through several funding round getting many major funding and is no more looking for small investors to share equity. **This is where we are thinking differently! We are predicting these startup at their initial stages to predict Series C or above.**

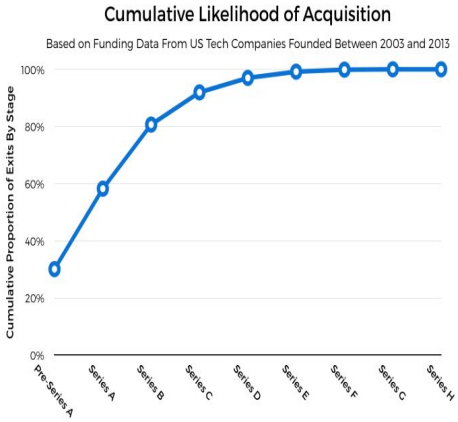


What do we predict?

We did some research on our end to see how we can classify a potentially successful startup in initial stages to see if it is going to get acquired or get IPO. A blog **"Here's how likely your startup is to get acquired at any stage" by Jason Rowley** based on funding data from around 15600 US based technology companies founded between 2003 and 2013 performs impressive analysis on funding stages of startups



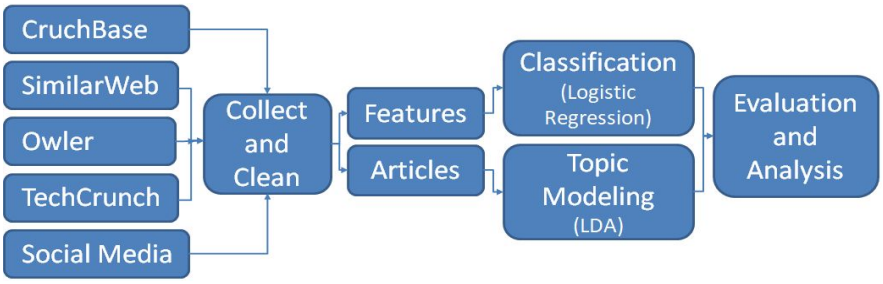




The first graph shows the proportion of company surviving next year after getting a particular type of funding, a company might not survive due to two major reason either it got acquired or it got closed. The second graph shows the companies that got a particular funding on logarithmic scale, which shows only 12% of companies make it to Series C and 1 % to Series F. The third graph shows that **92% of companies that raised Series C type funding are got acquired**. Our problem is to **predict if a company reaches a Series C funding based on its online presence, managerial and financial data**.

3. Data Science Pipeline

In this section, we describe our data science pipeline, which is demonstrated using a flow diagram in Figure on the right. Different stages of the pipeline is described as follows.



3.1. Data Collection

We collect all our data using web scraping and API for social networks. We scrape the startup list and their handle from **startups-list** and **yclist** for getting a list of 44,000 company names all around the world. Our main data sources are **Crunchbase, SimilarWeb, Owler, TechCrunch article** (58, 157 articles) and Social Media data API (Facebook and Twitter).

3.2. Data Cleaning

After scraping all the data, we used below steps to clean the scraped data, so that it can be used to form features for our final model :

Cleaning Scraped Data	Text Cleaning
1) Some companies have no of employee as a range, we took the mean. 2) Some of the categorical data was reduced for ex. major cities were included and rest were taken as others. 3) Entity resolution was done for scraped data from different sources like crunchbase, owler and similarweb.	1) Split the documents into tokens after converting them to lowercase. 2) Remove stop words using NLTK and Gensim library functions and also using stopwords list. 3) Lemmatize all words in documents. 4) Add bigrams and trigrams to docs (only ones that appear 20 times or more). 5) Filter out words that occur less than 20 documents, or more than 50% of the documents.

3.3. Feature Extraction

In this project, feature extraction happens in two phases, as described in the following subsections. We generate features from the collected data from the websites as well as from the collected articles from techcrunch.

Features from Websites: In the first phase, we extract the available features from the above mentioned sources through web scraping, cleaning and resolving conflicts between data sources. **One of the main problem for this particular problem is, available data is very sparse.** We discard a few columns with almost no values. To learn more about the data, we observe the correlation among the features. Figure 3 represents the correlation among the features that we are using in this project.

If we look at Figure 3, we will see for the most of the features, the correlation is very low. There could be two possible reasons for this. Firstly, The features really have a very low correlation to each other, and the second one is the sparsity of the data. Dark blue squares on Figure 3. represents zero or almost no correlation, light blue squares represents moderate (0.3 to 0.5) correlation, while the red square stands for higher range of correlation (>0.5).

Features from Articles: In the second phase, we use topic modeling for extracting a few features from the techcrunch articles. Each document in the pool of articles is represented as a distribution over topics and each topic is represented as a distribution over words. The central idea is to treat the news for each company as a finite mixture over an underlying set of topics, each of which is in turn characterized by a distribution over words, and build models via such topic distributions using machine learning techniques. We adopt the latent Dirichlet allocation (LDA) to build the composite topical features. We tried different no of topics and found out that the LDA gave best results when number of topics remains within ten. Table 1 represents the most important topics found by the LDA model.

Topics	Words
Social Media	User, Facebook, Social, Twitter, Friend, Photo
Business	Business, Start-up, Technology, Platform, Founder
Mobile Devices	Device App, Mobile, Apple, Android, Nokia, Kindle, iPad
Stocks	Million, Billion, Revenue, Share, Stock, IPO
Advertising	Google, Ad, Search, Advertising, Web, Yahoo
Funding	Venture, Investor, Round, Capital, CEO, Raised

Table 1. Topics found in the techcrunch articles using LDA topic modeling

We use these topics as additional features to feed into the machine learning model for predicting the success of a start-up in the hope that, it will enhance the predictive performance of the trained model.

3.4. Machine Learning for Prediction

We use the features extracted in the previous step of our pipeline to train a classifier. For this project, we use Logistic Regression for the classification purpose. As we have already mentioned, we want to predict the companies that will pass series C. In the extracted data, we have multiple categories of companies. At the beginning, we used the whole dataset to train our model and evaluated performance. We also trained different model for different categories of businesses, because different categories of business have different factors that affects their success.

4. Methodology

We used the following tools and technologies at different stage of our product

Data Collection : Selenium Webdriver browser scripts ,BeautifulSoup, Lxml , urllib2, Facebook/Crunchbase Rest API's

Data Integration : Pandas, Mysql

Machine Learning : We mainly used scikit-learn. We tried a couple of classification models like Naive Bayes, Decision Tree, SVM, Logistic Regression. We found out that Logistic Regression gave best results.

NLP : Gensim, NLTK, stop_words, LDA
Visualization : pyLDAvis, Matplotlib, Seaborn, Plotly

5. Evaluation

After training our model, we need to do performance evaluation, for which we split the data into train and test set according to 1:4 ratio randomly and repeat this process ten times to get an average result. Testing our model using the test dataset, provided the accuracy of more than 90% for the whole dataset as well as for all the different categories. However, for this specific problem, accuracy of prediction is not the main concern. If we think from the point of view of an investor, we understand that, an investor will not want to invest in a start-up that has a high probability to fail. Therefore, while predicting the success of start-ups we need to carefully look at the rate of true positives and false positives. Most of the investors will want to look at the false positives. **Because false positive means investing in a startup that is predicted as a success but eventually fails.** We also look at the area under the ROC curves to get an overall impression about the model.

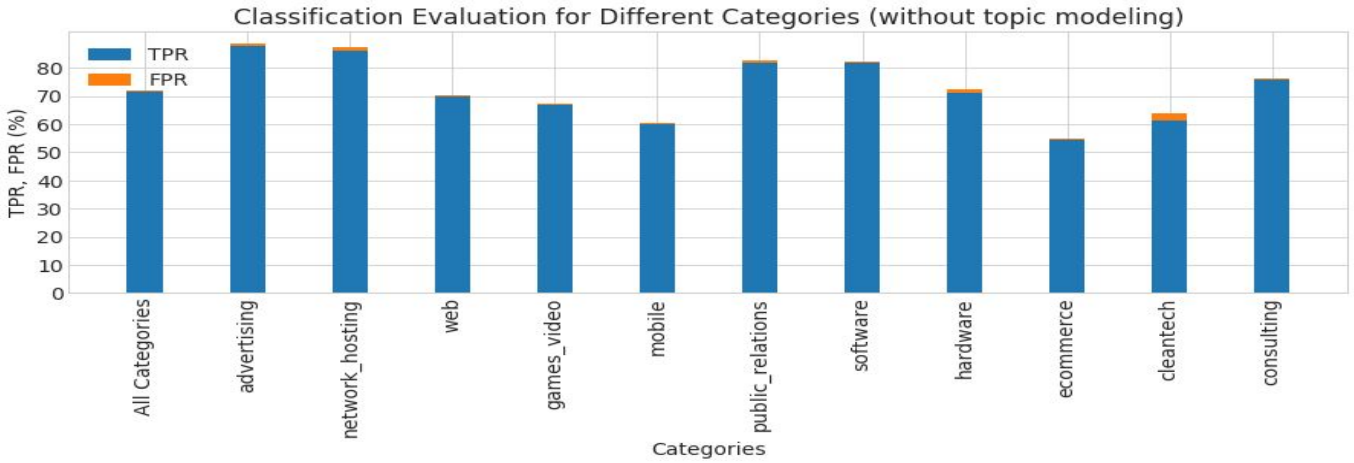


Figure 4. TPR and FPR for different categories of start-ups

Figure 4 represents the true positive rates and false positive rates for different models that are trained for different categories of startups. If we look more carefully at the figure, we observe that, most of the categories have true positive rates between 60% to 80%. For the overall dataset, the true positive rate is around 72%. The most important observation is, the false positive rate is really low. For most of the categories false positive rate is below 1%. Table 3 provides a comprehensive representations of all the results.

As we already discussed in the previous section, we also use the topics found from the topic modeling as additional features for our machine learning model. As expected, these additional features enhances the performance of our model in terms of increasing the true positive rate.

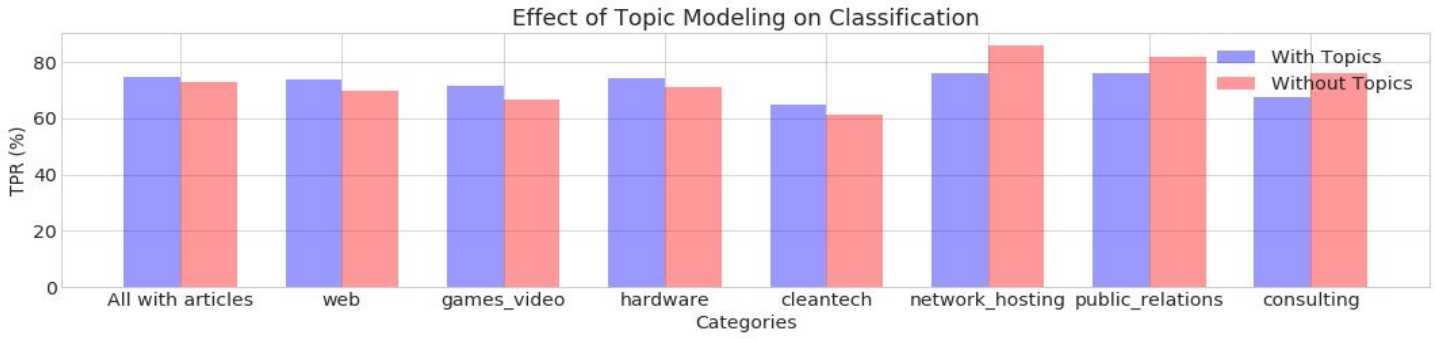


Figure 5. Comparison of TPR for different categories, with and without topic modeling

Figure 5 shows a comparison between true positive rates for different categories of startups with and without the topic features. As we can see from the figure, the first couple of bars represent only the companies who have articles on them. We see a slight improvement in terms of true positive rates while using the topic modeling. For the other categories like 'web', 'games_video', 'hardware', 'cleantech' we have improved performance while using the topics as additional features. However, for the categories like 'public_relations' and 'consulting' we see that, using topic makes the model perform worse. From this result we observe that, for the companies that fall into category of technology related company can be predicted better using the topics from topic modeling. However, the success of non technology companies are worse predicted, while using the topics as features. The reason behind this may be, the lack of articles in these non-technical categories in techcrunch.

An investor would also like to know about the important features behind the success of a startup, so that the investor can have an overall idea about how the company might do. This will provide the investor with some intuitive understanding about the success of the company, without relying blindly on a machine learning model. We wanted to answer this question by finding out the weights of the features for our machine learning model.

As we have trained multiple models for different categories of businesses, it is possible to have differences in the serial of important features. However, we observe that, for all the categories, except 'public_relations' the top most features remains the same. Figure 6. shows us the important features for the majority of models we trained for different categories. The features that found to be important are self explanatory. However, only for the category 'public_relations' the most important feature is found to be 'the number of employees'.

6. Data Product

Our data product will predict if a company reaches Series C funding based upon its online presence, managerial and financial data. We can do consulting using our data product. Basically, we can tell micro-investors which startup's have a higher chances of success, so that they can invest smartly. The way our product work is from the name and social handle of the startup, it scrape all the data about it and also converts them into features. Now, these various features (managerial, financial and online presence) will be passed to our trained model, which will predict the success/failure of the startup.

7. Lesson Learnt

During this project we had quite a few important learnings. As we have collected our data through web scraping, the data was not very well organized. Moreover, the websites we used as our data sources are mostly crowdsourced. As a result, a huge portion of the collected data was blank. This is not a very special case we working with, as we learnt most of the real world data has this problem of data sparsity and a data scientist or engineer should be able to deal with this problem. Moreover, collecting data through web scraping is difficult, especially for the websites who does not allow scraping over a certain range. Having data from multiple sources also makes the data integration part very difficult, which demands tedious effort to make a organized data set.

8. Summary

In this project, we use publicly available data to predict the success of a startup in its early ages to help micro-investors to make a more informed decision about their investment. We use logistic regression for classifying the companies that goes beyond series C. We also perform topic modeling on the articles found from techcrunch. We observe that, for most of the categories our model achieve true positive rate from 60% to 80% while the false positive rates remains as low as 1% for most of the cases. While using the topics found from the pool of techcrunch articles as additional features, the performance of the models in terms of the true positive rate was enhanced for the companies that fall into technology category.

References

[1] Guang Xiang, Zeyu Zheng, Miaomiao Wen, Jason Hong, Carolyn Rose, Chao Liu. A Supervised Approach to Predict Company Acquisition with Factual and Topic Features Using Profiles and News Articles on TechCrunch

[2] Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. Journal of Machine Learning Research 3:993–1022.

[3] gensim - topithc modeling for humans [https://radimrehurek.com/gensim/]