-- Independent Work Report Fall, 2016 --

**A Graph Analysis of Investment Patterns in the Startup World**

Samhita Karnati

Advisor: Prof. Andrea LaPaugh

**Abstract**

*Abstract stuff*

**1. Introduction**

**2. Problem Background and Related Work**

Prior work can be largely grouped into two categories. The first is an analysis of general trends. These studies tend to be conducted by banks and consulting firms, seeking to pick out major trends from the past year to use them to predict similar broad trends in the coming year. For example, in Strategy&’s “2016 Technology Industry Trends” report, they notice that\_\_\_\_\_\_\_\_.

These observations are interesting as they pick out a trend relating to company category and investments. \_\_\_\_\_\_\_\_\_\_\_.

The other category of related work is event prediction regarding later-stage financial events like mergers & acquisition and IPOs. Most studies in this area use a combination of numerical and categorical features. An example is “A Supervised Approach to Predict Company Acquisition With Factual and Topic Features Using Profiles and News Articles on TechCrunch” from Carnegie Mellon and Microsoft Research. Here, numerical features include features like number of employees, number of competitors that got acquired, and number of products. To get the categorical, or “topic” features, the team treated “the news articles for each company as a finite mixture over an underlying set of topics, each of which in turn can be characterized by a distribution over words, and build models via such topic distributions using machine learning techniques.” Here, the topics that are more closely associated with acquisition will have higher probabilities for words that constantly occur in acquisition-related themes. Of particular interest to this project, the CMU/MSR study used funding round information and the company category as features for predicting M&A. In total, there were 22 features and a Bayesian Network was used as the primary learning algorithm. This study’s approach was largely successful with a high true positive rate between 60% and 79.8% and a false positive rate between 0% and 8.3%. This high success rate indicates that supervised learning methods are proven to work for event prediction.

Another study that falls within the category of event prediction and is more closely aligned goal-wise with this project is “Predicting Startup Funding via Twitter,” a final project from Harvard’s computer science department. As the title indicates, the goal with this study is to see if there is some correlation between PR in the form of tweets and startup funding rounds. After extracting features from the tweets based on sentiment and some data exploration, the study used different regression techniques and neural nets to identify trends. Their results indicated that there was a positive correlation between tweets and startup funding, but the authors indicated that they needed to combine these tweet-based features with numerical and factual features for specific event prediction.

**3. Approach**

This project seeks to extend past work in this area by addressing some of their shortcomings and by applying previously proven techniques to this slightly different problem. In all of these past studies, the data sets have been limited in some way or another. Firms publishing reports are typically constrained by their own deal flow, which can lead to biases and analysis only on a subset of company categories. The CMU/MSR study used TechCrunch and crunchbase, which provides a much more extensive dataset. However, they had a threshold for the minimum number of extractable features based on TechCrunch articles, resulting in a dataset with just under 6,000 companies of the 81,000 companies available on crunchbase at the time of the study (January 2012). In this project, we use the crunchbase dataset as of November 2016, which has complete information on 74,339 companies. This dataset is more extensive, more complete, and much larger than datasets previously used for similar studies.

Since past work has shown us that supervised learning methods can be used to predict specific events, we will use such methods to predict our event of question – funding. Since funding of a company is often one of the first things that it experiences, there tends to not be very much other information on companies at this early stage. Thus, the feature that will be used to predict future funding is past funding, as this is part of the limited information that we have across all companies. It is also important to note that while in previous studies, company category was used as a feature, we want to try and predict/assign categories. Again, we will use past funding to assign categories.

**4. Implementation**

Solving this problem can be broken down into five steps: data collection and organization, supervised learning methods, unsupervised learning methods, supervised clustering, and evaluation of these three methods.

**4.1 Data Collection and Organization**

As previously mentioned, the data used in this project is from crunchbase. More specifically, we will use the Daily CSV Export from November 11, 2016. The Daily CSV Export includes separate files with information about companies, people, funding rounds, acquisitions, and IPOs. The file that will be particularly useful for this project is the funding\_rounds csv. As the name suggests, it contains all the funding rounds for every company in the dataset along with metadata like company category and geographic location. After filtering out companies with missing information, we are left with 74,339 companies to use. These 74,339 companies were funded by \_\_\_\_\_ investors in \_\_\_\_\_ investments.

As we are trying to predict specific relationships – funding – and use these relationships to assign categories, a data organization scheme that demonstrates relationships is the most intuitive and would be the most useful. Thus, we organize our data as a directed graph with nodes representing companies and investors and edges from investors to companies representing investments. The nodes’ attributes are category and location and the edges are weighted based on the dollar amount transacted.

**4.2 Supervised Learning**

We can translate the problem of predicting investments in the context of our digraph as a link-prediction problem. In other words, we want to predict whether or not a given transaction will occur. Thus, we will take a random 90% of the edges in our digraph as our training set and try to predict the remaining 10% (our testing set). We will approach the link-prediction problem by using three similarity measures and scores.

The first similarity score we will use is a modified Jaccard score. When used in undirected graphs, the Jaccard similarity score of two nodes A and B is the intersection of the sets of neighbors of A and B, divided by the union of the sets of neighbors of A and B:

However, given our directed graph construction, if A is a company node and B is an investor node, the neighbors of A will be investors and the neighbors of B will be companies. Thus, there will be no overlap between the two. This score makes the assumption that nodes with similar neighbors will be more similar to each other. We can extend this assumption to our case by asserting that investors who have invested in a larger number of similar companies are more similar to each other. Given this, we can modify the Jaccard similarity score for our directed graph. With a company node A and an investor node B, we will calculate the following:

In doing this calculation, we compare the company’s neighbors’ neighbors (other companies who were invested in by the investors of our company A) with the investor’s neighbors (companies that our investor B invested in).

Next, we will use the Adamic-Adar index. This measure is similar to the Jaccard similarity score, except that we weight rare features heavily. A classic application of this measure where it works better than the Jaccard score is in Facebook friend prediction. If somebody has thousands of friends, their relationships are assumed to be less telling than those of someone with only a few hundred friends. However, in this application, this assumption might not be valid; investors who have invested in lots of companies are more likely to invest, and so their investments should not be weighted less due to this quantity. It is still useful to calculate these indices, as there might be some latent variable at play we are not considering. The Adamic-Adar index \_\_\_\_\_\_\_\_\_\_

The final similarity score we will use for our link-prediction problem is the preferential attachment score. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**4.3 Unsupervised Learning**

In addition to specific event prediction, we also want to investigate the relationship between investments and company category. Our hypothesis, based on observations cited in technology trends reports (like Strategy&’s, see Problem Background and Related Work), is that our investments digraph will cluster based on category. This is because companies in the same category have similar investing patterns and have been invested in by the same entities.

In order to test this hypothesis properly, we need to remove significant confounding variables, namely region, as investors tend to invest in companies in the same geographic area as them. We will only use transactions where the company node is in the SF-region, as this is the single largest region in terms of transactions. Doing so leaves us with 15,183 company nodes, which is roughly 20% of all companies in the dataset.

As prior work has shown that k-means clustering was the most effective clustering method used, we will use k-means clustering for this project. K-means clustering works by partitioning a set of n observations into k sets in a way that minimizes the within-cluster sum of squares (WCSS). WCSS is the sum of distance functions of each point in the cluster to the center of the cluster, so the goal here is to minimize the within-cluster Euclidian distance. We can mathematically model this goal as follows:

If there is a strong correlation between company category and investment, we would expect to see roughly the same number of clusters as there are company categories. There are 550 unique first-order categories in this dataset and so we will use this number as our first cluster parameter. Afterwards, we will use k-fold cross-validation to further examine the structure of the graph.

**4.4 Supervised Clustering**

**5. Evaluation**

**5.1 Evaluating Supervised Learning for Link Prediction**

**5.2 Evaluating Unsupervised Learning for Company Category**

When evaluating the success of a clustering algorithm, there are two scores that are often used. First, we want to check for homogeneity within each cluster. In other words, we need to check that all the companies within a given cluster are of the same category. A common way to do this is to use the root-mean-square standard deviation (RMSSTD) index. The RMSSTD index is the variance of the clusters and thus measures homogeneity of the clusters. The index can take a value between 0.0 and 1.0, where 0.0 means that there are no differences within the clusters (variance is low) and 1.0 means that there is no similarity within the cluster (variance is high). Formally, the index is defined as:

Breaking this down further, the term is the within sum of squares of the *j*th variable. The within cluster sum of squares is calculated as follows:

Scikit-learn has a method that gives homogeneity scores based on the RMSSTD index that we will use to compute scores for each of the 550 clusters.

The other way we can evaluate the success of these clustering algorithms is to check that all the companies of a given category are in the same cluster. Thus, this score measures completeness. For each of the 550 categories defined in the dataset, we will find the cluster that has the most companies of the given category. Then, we will divide the number of companies of the given category by the total number of companies of this category in the full dataset. This fraction is the completeness score and we will have one score for each of the 550 categories of the dataset. If , this means that there is no cluster that includes a significant portion of the companies of a given category and if , then there is one cluster that has all of the companies of a given category.

***5.2.1 Homogeneity Evaluation***

***5.2.2 Completeness Evaluation***

After evaluating the completeness scores for each of the 550 categories, only 21 categories had a non-zero value. Table \_\_ shows the completeness scores for these 21 categories:

|  |  |  |  |
| --- | --- | --- | --- |
| **Company Category** |  | **Companies in Category** | **Companies in Cluster** |
| Medical Device | 0.333333 | 3 | 1 |
| Optical Communication | 0.142857 | 7 | 1 |
| Home Automation | 0.1 | 10 | 1 |
| Diabetes | 0.090909 | 11 | 1 |
| Compliance | 0.058824 | 17 | 1 |
| Cloud Security | 0.045455 | 44 | 2 |
| Ad Targeting | 0.026316 | 38 | 1 |
| Information Services | 0.020833 | 48 | 1 |
| Biotechnology | 0.017241 | 754 | 13 |
| Information Technology | 0.015267 | 131 | 2 |
| Financial Services | 0.013514 | 74 | 1 |
| Crowdsourcing | 0.010753 | 93 | 1 |
| Biopharma | 0.010101 | 99 | 1 |
| Electronics | 0.009217 | 217 | 2 |
| Automotive | 0.007092 | 141 | 1 |
| Health Care | 0.006682 | 449 | 3 |
| Internet | 0.002755 | 363 | 1 |
| Analytics | 0.002010 | 995 | 2 |
| Enterprise Software | 0.001838 | 544 | 1 |
| Apps | 0.001825 | 548 | 1 |

Table \_\_\_: Non-Zero Completeness Score Information

Given that 21 of the 550 (or 3.8%) of the categories were clustered to some non-zero degree of completeness, this result seems to show that clustering this investment graph did not yield clusters that align with company category. However, when we compare these results with what the completeness scores would be assuming an evenly distributed and random clustering, we see that these 21 categories actually outperform the baseline. We calculate the baseline by taking the total number of companies in a given category and dividing it by 550 (giving the number of companies of this category in a given random cluster) and then dividing this by the total number of companies in a given category. Thus, we get:

and all of the categories in the above table received completeness scores of greater than this baseline threshold.

It is also important to take into account the total number of companies in a given category. Even though a completeness score of 0.333333 for the Medical Device category seems very high, there are only three companies in this category, which means that this score represents that there was a cluster with one of these three companies. This is not impressive. The fourth column in Table \_\_ shows the number of companies for each category that were in the cluster used to compute the completeness score. Almost all the categories have one or two companies in a given cluster and not that many companies over all.

The one interesting category is Biotechnology, where 13 of 754 companies were in the same cluster. Upon further investigation into this cluster where these 13 Biotechnology companies were clustered together, the total number of companies in this cluster is 38. This means that this cluster is not a massive cluster that just happened to capture 13 companies of the same category, and this result is significant.

**5.3 Evaluating Supervised Clustering**

**6. Summary**

**6.1 Conclusions**

**6.2 Future Work**

**6.3 Implications**

**References**