Capital Crunch: Predicting Investments in Tech Companies

Zifei Shan, Haowen Cao and Qianying Lin
Department of Computer Science, Stanford University
{zifei,caohw,qlin1}@stanford.edu

ABSTRACT

For many start-ups, lack of investment and capital has become the bottleneck for development. This phenomenon inspires us to use machine learning algorithms to find patterns in investment behavior from major investors. We propose to use various domain-specific features to predict which investors would potentially invest in a particular company. This would not only reveal important information about investment strategies and behaviors of investors, but also give startups ideas on where to seek potential investment and how to adjust their strategies so as to attract potential investors.

Our work is grounded in CrunchBase, an accessible knowledge base that maintains full records of company and people information.

There are two primary goals of our work:

(1) To predict whether an investor would invest in a particular start-up based on textual, topological and domain-specific signals from both the investor and start-up.

(2) To analyze and reveal the factors that would prompt an investor to invest in startups so as to shed light on the adjustments the start-ups could make to attract more investments.

Keywords

Machine Learning, Data Mining, Link Prediction, Startups

1. DATASET

We use data from *CrunchBase.com*, one of the biggest databases about information of companies. The current CrunchBase dataset includes 214,290 companies and 286,659 people.

1.1 Accessing data

CrunchBase provides indexing data and an API for full access of their data, yet the API has limited throughput. Due to the limitation, by now we would like to sub-sample the dataset, and we may get the full dataset in the future.

For data sub-sampling, a possible strategy is random sampling. However, it would have the potential drawback that we might not achieve consistency amongst different parties. (e.g. Facebook is sampled but its CEO Mark Zuckerberg is not). Therefore we propose to adopt the strategy where we start with a "seed set" of companies, and iteratively sample all related people and organizations to grow the network.

1.2 Data format

The detailed data format for people and companies is demonstrated below.

```
// people
"data": {
  "uuid": "a01b8d...",
  "properties": {
    "bio": ...
    "last_name": ...
    "first_name": ...
    ...
  }
  "relationship": {
    "degrees": {...}
    "experiences": {...}
    "news": {...}
    ...
  }
}
```

```
// company
"data": {
  "uuid": "770db0...",
  "properties": {
    "description": {...}
    "founded_on": {...}
    "name": {...}
    "number_of_employees": {...}
  }
  "relationships": {...}
  "board_members_and_advisors": {...}
  "acquisitions": {...}
  "competitors": {...}
  ...
}
```

2. PROPOSED MODEL

2.1 Data Model

The CrunchBase dataset has a variety of entities: organization, person, product, etc. There are also different relations including investment, acquisition, degree, founder, etc.

For simplification, we categorize organizations into **startups** and **investors**, and we care about predicting **investment** relationship between them.

The data model is defined below:

- $Startup(startupId, [attributes...])$
- $Investor(investorId, [attributes...])$
- $Investment(investorId, startupId, isTrue)$

Where we use features in *Startup* and *Investor* entities to predict *Investment* relations.

2.2 Problem definition

Our former problem is:

DEFINITION 1. *Problem: given the full **Startup** relation and **Investor** relation, predict **isTrue** value in **Investment** table, which determines if any given investor invests a startup.*

The desired output is a predicted probability between each investor and a startup. For example:

# investor	startup	probability
facebook	hello-doctor	0.95
google	hello-doctor	0.85
google	zynga	0.97
twitter	zynga	0.45

2.3 Proposed Features

A rich set of features can be applied to predict investments. They may include:

- Company attributes. e.g. date founded, number of employees.
- Attributes of correlated people. e.g. degrees of founders and employees.
- Linguistic features: information buried in company descriptions and biography of people.
- Network topology, e.g. make use of all relations including degree, founder and other investments. These feature may be only captured by a factor graph model discussed later.

2.4 Baseline and Oracle

A naive baseline model would be a random predictor that predict random labels based on some class priors.

A better baseline would be training an independent logistic regressor for each individual investor, that takes a feature vector of a start-up and predicts a label. The drawback of this model is that it can hardly utilize investor-based attributes and higher-level knowledge such as network topology.

As an improvement, we propose to apply a factor graph model that correlates features across this graph. Specifically, each *Investment* relation is a boolean variable that we are predicting, and features in both sides of investors and startups can be correlated. We can further design features for more complex correlations.

2.5 Getting training data

To train the predictor, we take ground truth investments in CrunchBase as positive training examples, that is, if an investor I has invested in a startup S, we obtain a training example $(I, S, true)$ in *Investment* relation.

For the negative training examples, it might not be desirable to simply label all pairs of $(investor, startup)$ that do not have a known investment as negative, because (1) this makes positive examples extremely sparse and introduce a data skew, and (2) even if an investor have not invested a startup right now, it is still possible that the invest will happen in the future. How to effectively label negative examples is still open to us. For now we propose to take random pairs of investors and startups with no known investment happening.

3. EVALUATION

We will evaluate our models based on ground truth investments.

Specifically, we hold out a fraction of training examples, and predict these relations and get measures including precision and recall. We may want to define our own measure to encourage aggressive predictions and punish false negatives more than false positives.

Another measure to try is the accuracy in top-K prediction: we hope to get a trained system where the most confident predictions are very likely to be true.

An further evaluation method is the calibration plot where we layout the predicted probabilities and the accuracy in testingset in buckets. See Figure 1. A discussion of interpreting calibration plots is in the paper [3].

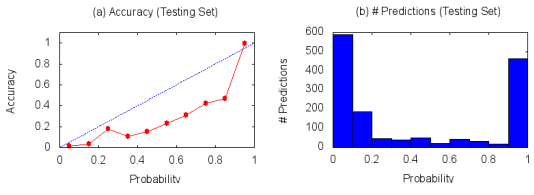

Figure 1: A sample calibration plot

4. CHALLENGES

We see several challenges in the project:

Data sparsity. To predict pairwise investment relationship between investors and startups, training data might be very sparse and highly skewed: most investors only invest a few companies. We may reduce feature space by carefully engineering features through error analysis, or trying methods like SVD. More advanced models, e.g. collaborative filtering, or a joint inference model that correlates prediction on all investors, would also help tackling the sparsity issue.

Feature Extraction. Designing and extracting features are the key to a successful predictor. To fully utilize the information hidden from raw text such as company descriptions and founder bios, we propose to adopt state-of-the-art natural language processing methods for feature extraction, including named entity recognition and dependency parse.

Model high-level knowledge. Some high-level knowledge would be hard to capture by a simple logistic regression model. We propose to use factor graphs to model the correlations between different investors, company and people, etc.

Scalability. If we are building a factor graph model with a large feature space, it would be hard to do learning and inference on the graph. We propose to use DeepDive [3], a highly scalable inference engine to tackle the problem.

5. RELATED WORK

In the paper [1], the author discussed a methodology to match proposals from start-ups to the potential investors on Kickstarter with linear regression, SVM-linear, SVM-poly and SVM-RBF, with an accuracy rate of 82% for static data features and 73% for dynamic data features. Their features are mostly updates made to the tweets, number of comments and so on, and we could widely expand the feature set.

Another paper [2] used a discriminant analysis to classify the potentially successful and unsuccessful companies. Their feature sets are worth noting, including individual characteristics of the entrepreneurs, the efforts by entrepreneurs (i.e. whether they actively look for resources and help), degree of innovation and so on. Though this paper is more on the social science side, we would like to scrutinize the feature sets so as to explore more meaningful and insightful features. For example, we could extend individual characteristics to how many start-ups the CEO has founded and their histories.

6. REFERENCES

[1] J. An, D. Quercia, and J. Crowcroft. Recommending investors for crowdfunding projects. In *Proceedings of the 23rd international conference on World wide web*, pages 261–270. International World Wide Web Conferences Steering Committee, 2014.

[2] W. Gartner, J. Starr, and S. Bhat. Predicting new venture survival: an analysis of âĂIJanatomy of a start-up.âĂİ cases from inc. magazine. *Journal of Business Venturing*, 14(2):215–232, 1999.

[3] C. Zhang, C. Ré, A. A. Sadeghian, Z. Shan, J. Shin, F. Wang, and S. Wu. Feature engineering for knowledge base construction. *arXiv preprint arXiv:1407.6439*, 2014.