CHAPTER 1

Discussion

In previous chapters, we outlined how we designed and evaluated a novel data mining system for automated Venture Capital (VC) investment screening. In this chapter, we discuss the merits and limitations of this project with respect to our system's design and performance, and its contribution to theory on startup investment performance.

1. System Design. We developed a data mining system that provides automated VC investment screening. Our system uses data collected from CrunchBase and PatentsView. We found that CrunchBase and PatentsView databases are large, comprehensive, and growing. However, CrunchBase records are sparse, long-tailed, and require cleaning. PatentsView features (do / do not) impact the system's performance. Our system generates a classification pipeline that is optimised to the dataset. We found that classification algorithm tuning had the greatest impact on performance during optimisation. In particular, Random Forests and Logistic Regressions were the most successful classifiers, Support Vector Machines and Artificial Neural Networks underperformed. We found that the top 3-5 pipelines generated from the optimisation process should be checked for their robustness over time. A limitation of our design is that the dataset slicing technique and the sparsity of the data sources may introduce biases. Our system is semi-autonomous but could be made fully autonomous with further development.

2. System Performance. We evaluated the performance of our VC investment screening system. We found that our system's performance is robust with respect to historical datasets (for 2012-16), which makes it suitable for forward-looking predictions. We found that the performance of our system was better or comparable to previous results from the literature. Peformance was positively related to longer forecast window (for a period of 2-4 years), later developmental stage (e.g Series C, Series D+), and breadth of target outcome (e.g. Exit). Our system's observed performance may have

been improved further if we had performed pipeline optimisation separately for each experiment. A limitation of our system is that some nuances of investment success are not fully captured (e.g. down-rounds, acqui-hires). However, we still believe that our system has performance that is practical for use in investment screening.

3. Model Evaluation. We developed a conceptual framework for startup investment performance based on the literature. This framework guided our data source selection and feature creation. We evaluated the framework using our data mining system. We found that models of startup investment performance generated by our system are robust with respect to time (for 2012-16) and forecast window (over 2-4 years), and vary with respect to developmental stage (e.g. Seed, Series A) and target outcome (e.g. IPO, Acquisition). Our system does have some limitations. We were unable to represent all factors from our conceptual framework (e.g. financial information), we may have understated some factors by not using more complex features (e.g. temporal relationships), and we were unable to generate structural models. Despite these limitations, we believe this project has made significant contributions to models of startup investment performance.

1.1 System Design

We developed a data mining system that provides automated Venture Capital (VC) investment screening using data collected from large online databases, CrunchBase and PatentsView. This system provides a multi-stage pipeline optimisation process that can automatically adapt to changes in the dataset or prediction task over time.

1.1.1 Data Collection & Manipulation

Our system uses data collected from CrunchBase and PatentsView online databases. Both data sources are fairly new (CrunchBase gained critical mass in 2012 and PatentsView was formed in 2015) and neither have been studied frequently in the literature. These data sources offer a significant improvement in terms of size and variety of features over previous data sources used in this research field (e.g. surveys, interviews, closed datasets). However, we found that CrunchBase was sparse, with many long-tailed features, and other abnormalities, and required considerable cleaning to remove irrelevant companies. We addressed these issues with a number of pre-processing steps in our classification pipeline. Unlike

CrunchBase, PatentsView is a government-regulated data souce, so it has fewer data quality issues. The greatest issue we had with PatentsView was matching companies between PatentsView and CrunchBase because companies often use variations on their names or have sub-entities. However, PatentsView data still produced a broad performance improvement to our system.

Our system uses historical CrunchBase datasets re-created through timestamps. While this technique provides significant benefits to our system, it also raises some concerns. We performed preliminary testing of our dataset slicing technique using last-updated timestamps and realised this would remove too many recently-updated records from the dataset. Instead, we used created-at timestamps which retain more records, but at the cost of possibly using features that were not originally available. While the impact of this effect is mediated by the relational database structure (e.g. acquisition and investment records have separate timestamps), it may artificially inflate historical results for companies that have had many later edits to their records. We tried to evaluate the impact of this technique by comparing a CrunchBase database collected in December 2013 with a slice from our primary dataset collected in September 2016. We found that there was only minimal variance in the number of records found in each relation. However, because the database schema changed dramatically between 2013 and 2016, we were unable to determine whether the completeness of each respective record is similar. As we collect more original database dumps, we will be able to better evaluate this technique.

A key benefit of our proposed VC investment screening system is that it will reduce the amount of manual effort required prior to the investment decision-making process. The implementation of the system described in this paper goes some ways to address this. Currently the system is semi-autonomous: it has little rrequirement for external input besides configuration of investment criteria (e.g. forecast window, developmental stage etc.), but still runs on-demand, rather than continuously. An improved implementation of the system would run in the background continuously, scheduling components of the system to run as needed to ensure the results are always optimised. Aside from scheduling, the CrunchBase data collection system could be improved. In the earlier stages of our project we developed a connector to CrunchBase's API that provided real-time access to their database. Although we abandoned this approach because of time constraints, it deserves further investigation. A time-stamped database produced in this manner would also allow for greater accuracy and analysis of temporal trends.

1.1.2 Pipeline Optimisation

A key component of our system is the pipeline optimisation process. While previous studies in this field have applied a few specific classification algorithms, we developed a pipeline optimisation process with the aim of greater accuracy and re-calibration of the system as the dataset changes over time. Our pipeline optimisation process is divided into two steps: pipeline creation and pipeline selection.

Pipeline creation performs a broad search and evaluation of candidate pipelines with varying hyperparameters. This search is performed across the pre-processing steps of the pipeline and also the classification algorithms. We found that classification algorithm tuning had the greatest impact on performance during optimisation. It appears that very little optimisation of the pre-processing steps was needed. In aggregate, the performance of the classification pipeline was not improved by the pre-processing steps. However, it is likely that the effect of the pre-processing steps was also highly dependent upon the classification algorithm (e.g. Random Forests are more resilient to low orthogonality than Naive Bayes), which may have reduced the aggregate effect. Nonetheless, optimisation of the pre-processing steps should still improve the overall robustness of the optimisation process as the dataset and prediction tasks change.

Our literature review suggested that Random Forests would be the most successful classifier, probably followed by Artifical Neural Networks and Support Vector Machines. We found that Random Forests and Logistic Regressions performed best and Artifical Neural Networks and Support Vector Machines underperformed. Random Forests may have outperformed the other algorithms due to its robustness to missing values and irrelevant features. Learning curves also revealed that, unlike most of the other classifiers, Random Forests was least likely to converge early, which suggests that with larger training sets it should perform better. Logistic Regression. It was surprising that Support Vector Machines and Artifical Neural Networks underperformed the other algorithms. However, these algorithms are far more difficult to accurately tune than these other algorithms, so it may reflect that our search process was too limited. In production, we could perform the search process over longer iterations which would likely result in better performance from these algorithms.

The second step in our system's pipeline optimisation process is pipeline selection. In this component, we rank the candidate pipelines generated previously and evaluate the best pipelines (finalist pipelines) over a number of different dataset slices. This process ensures that our final pipeline is robust in their performance with respect to time. We don't observe significant variance in the pipelines on aggregate against the dataset slices, but there is variance within the individual pipelines. The former result suggests that our pipelines produce models that are robust with respect to time (which is reinforced in the final evaluation). The latter result justifies this step in our process. In our preliminary evaluation of this test we selected the top ten candidate pipelines. Although there is still a strong positive correlation between the pipelines initial ranking and their scores, we can see that there are some individual deviations. Importantly, the top-ranked pipeline from the first stage actually has a lower median score than the second-ranked pipeline. These results suggest that it is optimal to evaluate the top 3-5 candidate pipelines in this manner.

1.2 System Performance

1.2.1 Experimentation

1.2.1.1 Time Slices

1.2.1.2 Forecast Window

1.2.1.3 Developmental Stage

1.2.1.4 Target Outcome

1.2.2 Limitations

1.2.2.1 Configuration

We evaluated the performance of the system across a range of variables, including size of training set, date of training set, duration of forecast window, company developmental stage, and target outcome. For each of these experiments, we manipulated these variables during the model fit and prediction step of our system design. To reduce the time taken by our experiments, we used the same optimised pipeline for each experiment (for the configuration, see Appendix**??**). This pipeline optimisation step takes the vast majority of time of our system (84.8%). By using a pipeline optimised for different objectives we are likely to have under-reported the performance of our system. In future research, it would be interesting to determine the extent to which the results of our pipeline optimisation changes with respect to these variables and the extent to which our results improve.

1.2.2.2 Outcome Nuances

The target outcomes that we used as objectives for our system (e.g. Exit, Acquisition) are rough proxies for the underlying success of the investment. A Venture Capital (VC) firm typically measures investment success on a financial basis (e.g. Internal Rate of Return (IRR) or multiple returned). While most funding rounds are generally at higher valuations than the previous round, some funding rounds are not - these are termed 'down-rounds'. Likewise, although most acquisitions are performed at higher valuations, sometimes they are not - these are often termed 'acqui-hires'. As our publicly-sourced dataset has little information about valuations at funding rounds or during acquisitions (valuation is considered more sensitive than quantum raised), our system has little ability to distinguish between succesful activity and down-rounds or acqui-hires. These discrepancies limit the performance of our system. In Appendix**??** we present four case studies that highlight the nuances of our system's performance. In future, applying sentiment analysis to media coverage of funding rounds, acquisitions or IPOs may indicate whether the activity was genuinely successful.

1.3 Model Evaluation

As a by-product of the evaluation of our system, we provide a comprehensive study of the determinants of startup investment performance. These models have their basis in the literature, but are evaluated on datasets larger and broader than any previous empirical evaluation of startup investment performance.

1.3.1 Experimentation

Through our experimentation, we produced models that describe features associated with startup investment performance over time, with different forecast windows, developmental stages, and target outcomes.

1.3.1.1 Time Slices

Robust with respect to time (for 2012-16)

1.3.1.2 Forecast Window

Robust with respect to forecast window (over 2-4 years)

1.3.1.3 Developmental Stage

Variable with respect to developmental stage (e.g. Seed, Series A)

1.3.1.4 Target Outcome

Variable with respect to target outcome (e.g. IPO, Acquisition)

1.3.2 Limitations

1.3.2.1 Missing Features

Missing features from framework - alliances, media, awards & grants, finances

1.3.2.2 Homogeneous Features

Largely homogeneous features - we didn't use semantic text, social network features

1.3.2.3 Simple Structure

Unable to determine complex structure - feature hierarchy, temporal relationships (edited)