Technological advances have made launching a startup more accessible than ever before \cite{tweney2015}. Billions of consumers can be accessed through the Internet and launching a startup can be done from a bedroom. However, startups remain competitive and risky endeavours. Technology startups can be unprofitable for years so entrepreneurs look for incubators, accelerators, angel investors and venture capitalists to support them through this developmental period. Aside from funding, investors hold other key resources (e.g. information, networks) that accelerate startup growth \cite{croce2013}. Investors act as scouts, able to identify the potential of new startups, and as coaches, able to help startups realise that potential \cite{baum2004}.

It’s important for startups to continue to win over investors throughout their development, but this is not a trivial task. Raising funding rounds from investors can be challenging and time-consuming, as investors find it difficult to quickly evaluate startups as investment opportunities. Investors spend time seeking and evaluating signals of a startup's underlying quality because clear metrics of performance often do not exist or are difficult to capture \cite{shane2002}. The growing popularity of online databases like AngelList and CrunchBase, which offer information on startups, investments and investors, is evidence of a desire for a more efficient assessment of startup potential. By the end of 2014, over 1200 investment organisations (including 624 venture capital firms) were members of CrunchBase's Venture Program, mining CrunchBase's startup data to help inform their investment decisions \cite{patil2015}.

Venture capital investment does not come without its trade-offs. About 50% of venture capital-backed startups end in complete liquidation \cite{hall2010}. Venture capital investors are protected from these losses because the minority of their investments that do survive have outsized returns: 85% of venture capital returns come from just 10% of investments \cite{sahlman2010}. As the venture capital industry matures, investors optimise for this extreme risk-reward trade-off by pushing startups to grow rapidly, frequently raise follow-on funding rounds and make quick, centralised decisions \cite{manigart2002}. The rapid growth demanded by venture capital investors is generally incompatible with public company structures, due to strict reporting and compliance requirements \cite{wies2015}. Accordingly, venture-capital backed startups are delaying their Initial Public Offerings (IPO): time to IPO has doubled in the past 20 years \cite{nvca2016}.

Startups remaining privately-held for longer has the effect of shifting value creation to the private markets. For example, Microsoft’s market capitalisation grew 500-fold following its IPO in 1986 \cite{microsoft}, but for Facebook to grow to the same extent since its IPO in 2012 its valuation would exceed the capitalisation of the global equity market \cite{facebook}. Venture capital funding for late-stage privately-held startups is approaching all-time highs as investors enter the private markets \cite{nvca2016}. It’s important to understand how the factors that influence venture capital investment change throughout a startup’s development. There is a clear gap in the academic literature in learning how the factors that influence startup investment change throughout a startup’s development. Previous approaches to learning the factors that influence venture capital investment in startups have three common weaknesses:

1. Prior work are largely restricted in sample size \cite{croce2016, conti2011, dixon2014, gimmon2010, hoenig2014}. Abundant empirical evidence has suggested that the size of training data eventually becomes more critical than the sophistication of algorithms themselves or even careful feature selection \cite{caruana2008}. Large open databases (e.g. CrunchBase, AngelList) and social networks (Twitter, LinkedIn) offer larger samples than those generally studied in previous works. We expect that using data collected from these sources will lead to the discovery of additional features and higher accuracy in startup investment prediction.

2. Prior work has focused primarily on early-stage investment in startups, primarily equity crowdfunding \cite{beckwith2016, ahlers2015, cheng2016, yuan2016}, angel investing \cite{croce2016} and early-stage venture capital \cite{werth2013}. The functions and objectives of startups change through their development \cite{CITE}. We expect that the signals that attract investment in these companies will similarly change over time.

3. Prior work has focused on factual company details (e.g. the headquarters’ location, the age of the company, the number of founders) for startup investment predictive models \cite{beckwith2016, dixon2014}, gimmon2010} . Semantic text features (e.g. patents and media) \cite{thorleuchter2012, hoenen2014, yuan2016, wei2009, conti2013} and social network features (e.g. co-investment networks) \cite{wang2016, werth2013, bhat2011, cheng2016, yu2015} may also predict startup investment and performance. We expect that developing a comprehensive model that includes semantic text and social network features alongside factual company features in could lead to better startup investment prediction.

This study will develop software that collects and processes information on startups to predict their likelihood of raising investment at different stages in their development. If successful, this study has the potential for scholarly, policy and firm-specific implications. We propose a theoretical framework for startup investment, based on work by Baum & Silverman (2004) \cite{baum2004} and Ahlers and colleagues (2015) \cite{ahlers2015}. Our theoretical framework models startup investment success as a product of two factors: venture quality and investment confidence. We will test this framework with respect to startup development, using cross-sectional and longitudinal analyses. We hope that this study provides interesting insights for entrepreneurs, policy makers, and investors and improve their understanding of the determinants of startup investment, especially for later-stage startups. Ultimately, we hope that this study encourages greater investment in startups.

The paper proceeds as follows. The next section explores theoretical models of technology startups and startup investment (Section~1). Thereafter, we review empirical evidence of features linked to startup investment (Section~2). We then determine how to collect the data to test those features (Section~3) and evaluate machine learning algorithms to find those that suit this startup investment prediction task (Section~4). The final section summarises our main results and concludes.

**1. Theoretical Background**

**1.1 Technology Startups**

**1.1.1 Resource Discovery**

**1.1.2 Developmental Lifecycle**

**1.2 Startup Investment**

**1.2.1 Effect on Startup Performance**

**1.2.2 Investment Assessment**

**1.2.2 Risk Management**

**1.3 Rationale**

**2. Feature Selection**

**<Figure 1 about here>**

**2.1 Venture Quality**

**2.1.1 Human Capital**

**2.1.2 Social Capital**

**2.1.3 Intellectual Capital**

**2.2 Investor Confidence**

**2.2.1 Third Party Validation**

**2.2.2 Historical Performance**

**2.2.3 Context**

**2.3 Feature Summary**

**<Table 1 about here>**

**3. Data Sources**

Predicting startup investment is a complex task. As reviewed in the previous section, there are many features that can influence startup investment decisions. Capturing the diversity of these features is critical to developing accurate models. Accordingly, this task will likely involve data collection from multiple data sources. Appropriate selection of these data sources is important because different data sources provide insights into different actors, relationships and attributes \cite{formsma2012, song2012}. In Table 2 we have outlined the general characteristics of a selection of relevant data sources and how they can contribute to the features we’re studying. In this section, we describe the desired characteristics of data sources for this task, review potentially relevant data sources, and ultimately determine which data sources are most likely to suit the characteristics of this task.

**<Table 2 about here>**

**3.1 Source Characteristics**

Online data sources typically capture a wide spectrum of businesses and entrepreneurs with varying data quality. It is important to filter entities to some extent before further analyses are performed. For this study, we will restrict our scope to companies that are based in the United States, because this is likely to be the largest grouping (as in \cite{alexy2012}). The following sections describe common data sources used in entrepreneurship research, particularly data sources that focus on technology startups in the United States.

**3.1.1 Surveys and Interviews**

Researchers in the field of entrepreneurship have historically relied heavily on surveys and interviews for data collection. Entrepreneurs may be more likely to provide private and confidential information in surveys and interviews than in other contexts. Information about human capital (e.g. founder, directors’ and staff capabilities), strategic alliances, and financial performance may be difficult to capture elsewhere. The trade-off for access to these features is that surveys and interviews are generally time-consuming and costly to implement. While online surveys address some of these issues there is still an issue of motivating potential participants to contribute. Startup databases and social networks can be far more efficient for data collection than surveys and interviews. For participants interacting with these sources, data collection is a secondary function so the participants do not require prompting from the researcher. Generally, the researcher can also collect the results from these sources automatically and at scale.

**3.1.2 Startup Databases**

Startup databases collect and store information about startups, investors, media coverage and trends. Most startup databases are closed systems that require expensive commercial licenses to use (e.g. CB Insights, ThomsonOne, Mattermark). CrunchBase and AngelList are two large crowd-sourced and free-to-use alternatives. CrunchBase and AngelList provide free Application Program Interfaces (API) for academic use. Dataset crawlers can be developed to traverse these APIs and collect data systematically. The advantages of such a crawler is that it can selectively collect data from nodes with specific attributes, or collect a random sample (as in \cite{jang2015}), or traverse the data source indefinitely, updating entries as new data becomes available. CrunchBase also provides public pre-formatted database snapshots which allows easier access to the dataset.

**CrunchBase.** CrunchBase is an open online database of information about startups, investors, media coverage and trends, focusing on high-tech industry in the United States. It relies on its online community to edit most pages. CrunchBase is a comprehensive database, with almost complete coverage of startups and investors in the Internet sector, including the relationships between them \cite{alexy2012}. However, it has been noted that the CrunchBase corpus is sparse with many missing attributes \cite{xiang2012, zhao2015}. CrunchBase has three provisions to prevent and remediate inaccurate crowd-sourced entries \cite{crunchbase2014}. First, all users are required to authenticate their CrunchBase accounts with a social media account which allows CrunchBase to verify a user's identity. Second, every change goes through a machine review, which flags significant or questionable updates for moderation. Third, well-known startups have their editing privileges locked and require email verification.

**AngelList.** AngelList is a promising new source of startup data, combining the functionality of an equity crowdfunding platform, a social networking site and an online startup database. As an equity crowdfunding platform, users create profiles for their startups on AngelList, and use the platform to attract investment. Investors use the platform to identify investment opportunities and can invest directly through AngelList, often alongside other investors in investment syndicates. AngelList is also an online startup database. It has a data-sharing agreement with CrunchBase (which results in significant overlap between the two sources), though AngelList is a less substantial database. Importantly, AngelList also tracks ``startup roles” (e.g. founders, investors, employees) with a creation time, start time and end time. This means that, unlike CrunchBase, AngelList’s networks can be re-created through time. For example, Britz et al. \cite{britz2013} analysed the formation of relationships on AngelList (looking at edge-creation time-stamps) to perform a longitudinal study of community growth and development.

**3.1.3 Social Media**

Social media platforms allow people to network and communicate online. As a side-effect, they also capture useful information about peoples’ identities and relationships. There are many diverse social networking sites: Facebook, friendships; LinkedIn, professional relationships; Twitter, micro-broadcasts; Snapchat, ephemeral video. Two social media platforms have been focused in on for academic research on entrepreneurship: LinkedIn and Twitter.

**LinkedIn.** LinkedIn is commonly used in online social network studies of entrepreneurship because it is a social network primarily used for professional networking \cite{formsma2012, song2012}. LinkedIn captures information about founders, directors and staff capability including past employment and education which is difficult to collect elsewhere. In addition, it can capture the professional social reach of founders and investors. Unfortunately, as of May 2015, the LinkedIn API no longer allows access to authenticated users' connection data or company data \cite{trachtenberg2015}, making it virtually impossible to use this site for social network analysis, without resorting to semi-automatic HTML parsing techniques (which is against the Terms of Service).

**Twitter.** Twitter is a social networking site and micro-blogging site that is often used by entrepreneurs to promote their personal and business brands and rapidly share news and opportunities. Users can send and read public messages (called tweets) of 140-character length. Twitter is a directed network where users can follow other users without gaining their permission to do so. Twitter’s public API provides access to social network topological features (e.g. who follows who) and basic profile information (e.g. user-provided description). However, Twitter’s public API only provides Tweets published within the last 7 days. To access historical Twitter data requires a commercial license to Gnip (also owned by Twitter).

**3.2.4 Patent Filings**

Startups file patents to apply for a legal right to exclude others from using their inventions. In 2015, the US Patents Office (USPTO) launched PatentsView, a free public API to allow programmatic access to their database. PatentsView holds over 12 million patent filings from 1976 onwards. The database provides comprehensive information on patents, their inventors, their organizations, and locations. One limitation might be difficulty matching identities across multiple data sources because registered company names (as reported in PatentsView) are not always the same as trading names (as reported in other data sources).

**3.2.5 Financial Reports**

There are many difficulties in searching for private company financial information. Unlike public companies, private companies are typically not required to file with the Securities and Exchange Commission (or international-equivalent). There are a few proprietary databases that provide this data but typically commercial licenses are prohibitively expensive. These databases also don’t tend to cover early-stage companies. For example, PrivCo is a source for private company business and financial intelligence that covers over 500,000 private companies. PrivCo focuses its coverage on U.S. Major Private Companies with at least $50-100 million in annual revenues but also has some coverage on smaller but high-value private companies (like startups).

**3.3 Source Evaluation**

We evaluated relevant data sources for their suitability to the current task, startup investment prediction. The startup databases CrunchBase and AngelList provide the most comprehensive set of features for this task. There are small differences between the features recorded in each. CrunchBase has slightly more coverage and tracks media mentions better but lacks AngelList’s social network and timestamping. At least one startup database should be used in this study, but it is less significant which one is used. Of the other data sources reviewed, patent filings and Twitter are the most promising. The US Patent Office has a free public API with comprehensive patent information, though it could be difficult matching identities across multiple sources because registered company names are not always the same as trading names. Twitter provides social network topology and basic profile information through its free API but doesn’t provide access to historical tweets. Other data sources are less promising because of access issues. Although Surveys and Interviews provide insights into human capital and past performance, manual data collection doesn’t suit the scope of this study. LinkedIn cannot be easily collected now that the API is deprecated. Financial reports are too expensive to access for the purposes of this study.

**4. Learning Algorithms**

Machine learning is characterised by algorithms that can improve their ability to reason about a given phenomenon given greater observation and/or interaction with said phenomenon. Mitchell provides a formal definition of machine learning in operational terms: ``A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E." \cite{mitchell1997}. Machine learning algorithms can be classified based on the nature of the feedback available to them: supervised learning, where the computer is presented with example inputs and desired outputs; unsupervised learning, where no labels are provided and the computer must find structure in its input; and reinforcement learning, where a computer interacts with a dynamic environment to perform a certain goal. These algorithms can be further categorised by desired output: classification, supervised learning that divides inputs into two or more classes; regression, supervised learning that maps inputs to a continuous output space; and clustering, unsupervised learning that divides inputs into two or more classes (basically, unsupervised classification). In this section, we describe the characteristics of the startup investment prediction task, review common machine learning algorithms, and ultimately determine which algorithms are most likely to suit the characteristics of this task.

**4.1. Task Characteristics**

Machine learning tasks are diverse and can be approached in many ways. For the current study, we will manipulate and combine the data collected from our data sources into a labelled data set appropriate for the application of supervised machine learning algorithms. Primary labels will be whether a startup received funding at each funding round, though measures of startup performance may also be investigated (e.g. survival time, exit). The key objective of machine learning algorithm selection is to find algorithms that make assumptions that are consistent with the structure of the problem (e.g. tolerance to missing values, mixed feature types, imbalanced classes) and suit the constraints of the desired solution (e.g. time available, incremental learning, interpretability). In Table 5 we have outlined some of the general characteristics of supervised learning tasks and identified the characteristics that are relevant to the current startup investment prediction task.

**< Table 3 about here >**

**4.1.1 Problem**

**Missing values.** Data sets often have missing values, where no data is stored for a feature of an observation. Missing data can occur because of nonresponse or due to errors in data collection or processing. Missing data has different effects depending on its distribution through the data set. Public data sets, like CrunchBase and social networks, are typically sparse with missing entries despite their scale.

**Mixed feature types.** Data sets can contain data with distinct primitive natures: real-valued, interval, counts, rank, binary, ordinal, categorical and multi-categorical types. The simplest way to handle mixed data types is to convert into a unified type (e.g. real-valued, binary). However, this process partially destroys type-specific information. We expect mixed-feature types in our dataset as we will be handling data from databases, social networks and semantic text analysis.

**Irrelevant features.** Despite best efforts to only include features that have some theoretical relevance, most machine learning tasks will include irrelevant features. Irrelevant features are those that have no underlying relationship with classification. However, depending on the way they are handled they may affect classification or slow down the learning algorithm. We expect irrelevant features in our dataset because our proposed framework includes features that haven’t been tested in the literature.

**Imbalanced classes.** Data sets are not usually restricted to containing equal proportions of different classes. Significantly imbalanced classes are problematic for some classifiers. In the worst case, a learning algorithm could simply classify every example as the majority class. Our dataset is not dramatically imbalanced overall, but when looking at funding status for different funding rounds it is significantly imbalanced.

**Small training set.** Machine learning techniques generally work better with more data \cite{caruana2008}. Problems of small-data are numerous, but mainly revolve around high variance: over-fitting is harder to avoid, noise is an issue and outliers become more significant. Some machine learning techniques are better at dealing with these problems with others. Our dataset is reasonably large (500,000 startups and 150,000 funding rounds) so this is not a relevant characteristic.

**High dimensionality.** In extremely high dimensional data sets, the number of features is larger than the number of observations. When dimensionality increases to the extent, the volume of the feature space increases so fast that the available data becomes sparse. This phenomenon is known as the curse of dimensionality or ``Hughes effect” \cite{hughes1968}. We do not expect our data set to be highly dimensional and where possible will transform the data into concise features.

**4.1.2 Desired Solution**

**Predictive power.** Predictive power is the ability of a machine learning algorithm to correctly classify new observations. If a model has no predictive power, then fundamentally the model is not representing the underlying statistical process being studied. For this reason, predictive power is a dominant characteristic. However, predictive power is far from the only important characteristic in machine learning selection. If algorithms provide similar predictive power, then other selection criteria become more significant.

**Interpretability.** Interpretability is the extent to which the reasoning of a model can be communicated to the end-user. There is a trade-off between model complexity and model interpretability. Some models are a ``black box” in the sense that data comes in and out but the model cannot be interpreted. For this study, it is important that we understand the determinants of the model so we are seeking algorithms that are highly interpretable.

**Incremental learning.** Incremental learning is where learning occurs dynamically whenever new observations are made and the algorithm adjusts what has been learned per the new observations. The key driver behind the need for incremental learning is when the underlying source generating the data is changing. It’s plausible that, as a system, the drivers behind startup investment are changing over time.

**Ease of tuning.** Machine learning algorithms have hyperparameters that must be tuned to ensure the model does not overfit its dataset. Some algorithms have many hyperparameters and tuning can be a computationally expensive process. We are not under significant time-pressure for this study and our dataset is not prohibitively large so this characteristic is not especially relevant.

**Computational speed.** The amount of time and computational resources necessary to train a model varies a great deal between algorithms. Training time is often closely tied to predictive power. In addition, some algorithms are more sensitive to the number of data points than others. When time is limited it can drive the choice of algorithm, especially when the data set is large. For this task, with a relatively moderate data set and without significant time-pressure, this characteristic is not especially relevant.

**4.2 Algorithm Characteristics**

Supervised machine learning are algorithms that can reason about observations to produce general hypotheses, and then make predictions about future observations. Supervised machine learning algorithms are diverse, from symbolic (Decision Trees, Random Forests) to statistical (Logistic Regression, Naïve Bayes, Support Vector Machines), instance-based (K-Nearest Neighbours), and perceptron-based (Artificial Neural Networks). The meta-review in Table 6 compares common supervised learning algorithms across general characteristics of the two domains mentioned in the previous section: assumptions about the structure of the problem and constraints of the desired solution. The following sections describe each candidate learning algorithm, critique their advantages and disadvantages, and present evidence of their effectiveness in relevant applications.

**< Table 4 about here >**

**4.2.1 Naïve Bayes**

Naïve Bayes is a simple generative learning algorithm. It is a form of Bayesian Network that models features by generating a directed acyclic graph, with the strong (naïve) assumption that all features are independent. While this assumption is generally not true, it simplifies estimation which means Naïve Bayes is more computationally efficient than other learning algorithms. Naïve Bayes can be a good choice for datasets with high dimensionality and sparsity as it estimates each feature independently. Naïve Bayes has been found to sometimes outperform more complex machine learning algorithms because it is reasonably robust to violations of feature independence, at least with regards to classification (Niculescu-Mizil & Caruana, 2005). However, Naïve Bayes is known to be a poor estimator of class probabilities, especially with highly correlated features (Zhang, 2004). Naïve Bayes was used alongside Logistic Regression, Decision Trees and Support Vector Machines to predict success in equity crowdfunding campaigns on the AngelList data set \cite{beckwith2016}. None of these models performed well. The algorithm that best predicted funded startups was Naïve Bayes with a Precision of .41 and Recall of .19, which means that only 19% of funded startups were classified correctly by the model. The author suggests the poor performance of their algorithms is caused by insufficient features captured in their training set, missing features relating to Intellectual Capital, 3rd Party Validation or Historical Performance. These features are included in the theoretical framework proposed by the current study.

**4.2.2 Logistic Regression**

Regression is a class of statistical methods that investigates the relationship between a dependent variable and a set of independent variables. Logistic regression is regression where the dependent variable is discrete. Like linear regression, logistic regression optimises an equation that multiplies each input by a coefficient, sums them up, and adds a constant. However, before this optimisation takes place the dependent variable is transformed by the log of the odds ratio for each observation, creating a real continuous dependent variable on a logistic distribution. A strength of Logistic Regression is that it is trivial to adjust classification thresholds depending on the problem (e.g. in spam detection (Hastie et al., 2008), where it is important that specificity is high). It is also simple to update a Logistic Regression model using online gradient descent, when additional training data needs to be quickly incorporated into the model. Logistic Regression tends to underperform against more complex algorithms like Random Forest, Support Vector Machines and Artificial Neural Nets in higher dimensions \cite{caruana2008}. This underperformance is observed when Logistic Regression is applied to startup investment prediction tasks \cite{xiang2012, beckwith2016, bhat2011}. However, weaker predictive performance hasn’t prevented Logistic Regression from being commonly used. Its simplicity and ease-of-use means it is used more casually, often being used without justification or comparative evaluation of its use \cite{gimmon2010, huang2015}.

**4.2.3 K-Nearest Neighbours**

K-Nearest Neighbours is a common lazy learning algorithm. Lazy learning algorithms do not perform explicit generalisation, but compare new instances with instances from training stored in memory. K-Nearest Neighbours is based on the principle that the instances within a dataset will exist near other instances that have similar characteristics (Cover and Hart, 1967). K-Nearest Neighbours models depend on how the user defines distance between samples; Euclidean distance is a commonly used metric. K-Nearest Neighbour models are stable compared to other learning algorithms and suited to online learning because they can add a new instance or remove an old instance without re-calculating \cite{kotsiantis2007}. A shortcoming of K-Nearest Neighbour models is that they can be sensitive to the local structure of the data and they also have large in-memory storage requirements. K-Nearest Neighbours was compared to Artificial Neural Networks to predict firm bankruptcy (Ahn & Kim, 2008). K-Nearest Neighbours is attractive in bankruptcy prediction because it can be updated in real-time. By optimising feature weighting and instance selection, the authors managed to improve the K-Nearest Neighbours algorithm to the point where it outperformed the Artificial Neural Network.

**4.2.4 Decision Trees**

Decision Trees use recursive partitioning algorithms to classify instances. Each node in a Decision Tree represents a feature in an instance to be classified, and each branch represents a value that the node can assume. Methods for finding the features that best divide the training data include Information Gain (Hunt et al., 1966) and Gini Index (Breiman et al., 1984). Decision Trees are close to an ``off-the-shelf” learning algorithm. They require little pre-processing and tuning, are interpretable to laypeople, are quick, handle feature interactions and are non-parametric. However, Decision Trees are prone to overfitting and have poor predictive power \cite{caruana2006}. These shortcomings have been addressed with pruning mechanisms and ensemble methods like Random Forests, respectively. Decision Trees were compared with Naïve Bayes and Support Vector Machines to predict investor-startup funding pairs using CrunchBase social network data \cite{liang2015}. Decision Trees had the highest classification accuracy and the authors suggest they are particularly useful in this application because their reasoning is easily communicated to startups.

**4.2.5 Random Forests**

Random Forests are an ensemble learning technique that constructs multiple Decision Trees from bootstrapped samples of the training data, using random feature selection \cite{breiman2001}. Prediction is made by aggregating the predictions of the ensemble. The rationale is that while each Decision Tree in a Random Forest may be biased, when aggregated they produce a model that is robust against over-fitting. Random Forests exhibit a performance improvement over a single Decision Tree classifier and are among the most accurate learning algorithms \cite{caruana2006}. However, Random Forests are more complex than Decision Trees, taking longer to create predictions and producing less interpretable output. Random Forests were used to predict private company exits using quantitative data from ThomsonOne \cite{bhat2011}. Random Forests outperformed Logistic Regression, Support Vector Machines and Artificial Neural Networks. This may be because the data set was highly sparse, and Random Forests are known to perform well on sparse data sets \cite{breiman2001}.

**4.2.6 Support Vector Machines**

Support Vector Machines are a family of classifiers that seek to produce a hyperplane that gives the largest minimum distance (margin) between classes. The key to the effectiveness of Support Vector Machines are kernel functions. Kernel functions transform the training data to a high-dimensional space to improve its resemblance to a linearly separable set of data. Support Vector Machines are attractive for many reasons. They have typically high accuracy \cite{caruana2006}, theoretical guarantees on limiting overfitting, and with an appropriate kernel they can work well even if data isn’t linearly separable in the base feature space (though this is an issue with a linear kernel). Support Vector Machines are computationally intensive and relatively complicated to tune effectively (compared to Random Forests, for example). Support Vector Machines were compared with back propagated Artificial Neural Networks in predicting the bankruptcy of firms using data provided by Korea Credit Guarantee Fund \cite{shin2005}. Support Vector Machines were found to outperform Artificial Neural Networks at this task, especially because it was on a small data set.

**4.2.7 Artificial Neural Networks**

Artificial Neural Networks are a computational approach based on a network of neural units (neurons) that loosely models the way that the brain solves problems. An Artificial Neural Network is broadly defined by three parameters: the interconnection pattern between the different layers of neurons, the learning process for updating the weights of the interconnections, and the activation function that converts a neuron's weighted input to its output activation. A supervised learning process typically involves gradient descent with back-propagation. Gradient descent is an optimisation algorithm that updates the weights of the interconnections between the neurons with respect to the derivative of the cost function (the weighted difference between the desired output and the current output). Back-propagation is the technique used to determine what the gradient of the cost function is for the given weights, using the chain rule. Artificial Neural networks tend to be highly accurate but are very slow to train and require significantly more training data than other machine learning algorithms. Artificial Neural Networks are also a black box model so it is difficult to reason about their output in a way that can be effectively communicated. Artificial Neural Networks have been rarely applied to startup investment or startup performance prediction tasks, probably because research in this area has used relatively small and low dimensional data sets. As one author put it ``More complex classification algorithms—artificial neural networks, Restricted Bolzmann machines, for instance—could be tried on the data set, but marginal improvements would likely result.” \cite{beckwith2016}. However, the current study seeks to address both of those issues so Artificial Neural Networks may be more relevant.

**4.3 Algorithm evaluation**

We evaluated common supervised learning algorithms for their suitability to the current task, startup investment prediction. In Table 7 we produce a ranking based on cross-referencing the task characteristics with the characteristics of the common algorithms. While this gives us some directionality of fit, we hesitate to rule in or out algorithms purely based on this ranking. Algorithm selection is complex and preliminary empirical testing will provide clarity as to which algorithms should be used. In addition, larger training sets and good feature design tends to outweigh algorithm selection \cite{caruana2008}. With those concessions in mind, we continue to review our findings. Our findings suggest that we should expect Random Forests, Support Vector Machines and Artificial Neural Nets to produce the highest classification accuracies. An ensemble of these high-performing methods may also provide an accuracy improvement, though at the cost of computational speed and interpretability. Random Forests could be expected to slightly outperform the other two algorithms due to robustness to missing values and irrelevant features and native handling of discrete and categorical data. However, Random Forests are not highly interpretable so Decision Trees and Logistic Regression might be preferable for early, exploratory analysis of the dataset.

**< Table 5 about here >**

**5. Conclusion**

A literature review was conducted to determine how to learn the factors that influence investment success for startups. First, we explored theoretical models of technology startups and startup investment (Section~1). Thereafter, we reviewed empirical evidence of features linked to startup investment (Section~2). We then determined how to collect the data to test those features (Section~3) and evaluated machine learning algorithms to find those that suit this startup investment prediction task (Section~4).

Venture capital funding for late-stage privately-held startups is approaching all-time highs as investors enter the private markets \cite{nvca2016}. It’s important to understand how the factors that influence venture capital investment change throughout a startup’s development. There is a substantial research gap around accurately predicting startup investment success. Existing approaches in the literature were assessed to have three common limitations: small sample size \cite{croce2016, conti2011, dixon2014, gimmon2010, hoenig2014}, a focus on early-stage investment \cite{beckwith2016, ahlers2015, cheng2016, yuan2016, croce2016, werth2013}, and sparse use of features \cite{ahlers2015, an2015, cheng2016, croce2016, werth2013, gimmon2010, thorleucter2012}. Although individual studies addressed some of these limitations, none attempted to synthesise their findings into a standalone study and piece of software.

This study will develop software that collects and processes information on startups to predict their likelihood of raising investment at different stages in their development. If successful, this study has the potential for scholarly, policy and firm-specific implications. We propose a theoretical framework for startup investment, based on work by Baum & Silverman (2004) \cite{baum2004} and Ahlers and colleagues (2015) \cite{ahlers2015}. Our theoretical framework models startup investment success as a product of two factors: venture quality and investment confidence. We will test this framework with respect to startup development, using cross-sectional and longitudinal analyses. We hope that this study provides interesting insights for entrepreneurs, policy makers, and investors and improve their understanding of the determinants of startup investment, especially for later-stage startups. Ultimately, we hope that this study encourages greater investment in startups.