MÁSTER UNIVERSITARIO EN INGENIERÍA INDUSTRIAL

TRABAJO FIN DE MASTER

DEVELOPMENT OF A VENTURE CAPITAL AUTOMATIC OUTBOUND SOURCING PROCESS TO FIND AND TARGET STARTUPS ONLINE

Autor: Gonzalo de la Orden López

Director: Sebastián Fernandez Medrano

Madrid

Julio de 2019

AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESINAS O MEMORIAS DE BACHILLERATO

1º. Declaración de la autoría y acreditación de la misma.

El autor D. Gonzalo de la Orden López

DECLARA ser el titular de los derechos de propiedad intelectual de la obra:

"Development of a Venture Capital Automatic Outbound Sourcing Process to Find and Target Startups Online",

que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

2º. Objeto y fines de la cesión.

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

3º. Condiciones de la cesión y acceso

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar "marcas de agua" o cualquier otro sistema de seguridad o de protección.
b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
e) Asignar por defecto a estos trabajos una licencia Creative Commons.
f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente)*.

4º. Derechos del autor.

El autor, en tanto que titular de una obra tiene derecho a:

a) Que la Universidad identifique claramente su nombre como autor de la misma
b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
c) Solicitar la retirada de la obra del repositorio por causa justificada.
d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

5º. Deberes del autor.

El autor se compromete a:

a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.

c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.
d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción de derechos derivada de las obras objeto de la cesión.

6º. Fines y funcionamiento del Repositorio Institucional.

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusive del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a 13 de Julio de 2019

ACEPTA 

Fdo.: Gonzalo de la Orden López

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título "Development of a Venture Capital Automatic Outbound Sourcing Porcess to Find and Target Startups Online"

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el curso académico 2018/2019 es de mi autoría, original e inédito y no ha sido presentado con anterioridad a otros efectos. El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.

Fdo.: Gonzalo de la Orden Fecha: 13/ 07/ 2019

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: Sebastián Fernández Medrano Fecha: 13/ 07/ 2019

SAMAIPATA
Samaipata Ventures SGEIC S.A.
C/Velázquez 18,2º Izq
28001,Madrid.
A-87436812

DESARROLLO DE UN PROCESO DE VENTURE CAPITAL DE OUTBOUND SOUCING PARA ENCONTRAR Y CUALIFICAR STARTUPS ONLINE

Autor: Orden López, Gonzalo de la.
Director: Fernández Medrano, Sebastián.
Entidad Colaboradora: Samaipata Ventures SGEIC SA.

RESUMEN DEL PROYECTO:

El propósito de este proyecto es construir un proceso “lean” y escalable para encontrar y cualificar inversiones en etapa pre-semilla en el ecosistema del venture capital europeo a partir de fuentes de datos online.

En el **Capítulo 1 – Estado del Arte**, se puede encontrar una descripción exhaustiva de la situación actual del ecosistema tecnológico europeo, desde el desarrollo de la industria del Venture Capital (VC) en Silicon Valley a su asentamiento en Europa, la riqueza de talento emprendedor en las comunidades tecnológicas europeas, y una presentación de los diferentes procesos de sourcing típicos empleados por firmas de VC para encontrar potenciales startups para sus portfolios, así como las dificultades encontradas en el proceso en hubs tecnológicos tan fragmentados. El capítulo concluirá con una vista general del enfoque empleado en los capítulos siguientes.

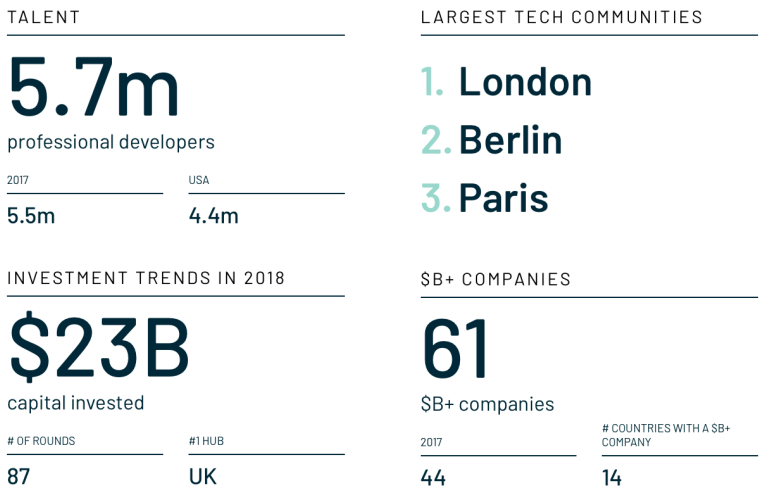

Figura 1. Figuras clave del ecosistema del VC europeo[1].

El **Capítulo 2 – Definición de un Pipeline de Ventas**, provée la estructura de un pipeline de ventas, proceso empleado típicamente dentro de los equipos de ventas y de marketing con el fin de cualificar y seleccionar oportunidades potenciales, para finalmente convertirlas en oportunidades reales al cerrar la

[1] https://2018.stateofeuropeantech.com

transacción. Este capítulo estudiará y proveerá una estructura y terminología para específica para modelar el proceso de inversión en firmas de VC, por el cuál cada startup pasará a través de una serie de escalones, cada uno de ellos constituyendo un filtro a partir de una serie de criterios. El pipeline construido en este capítulo servirá como estructura sobre la que se creará el proceso de outbound sourcing en los siguientes capítulos.

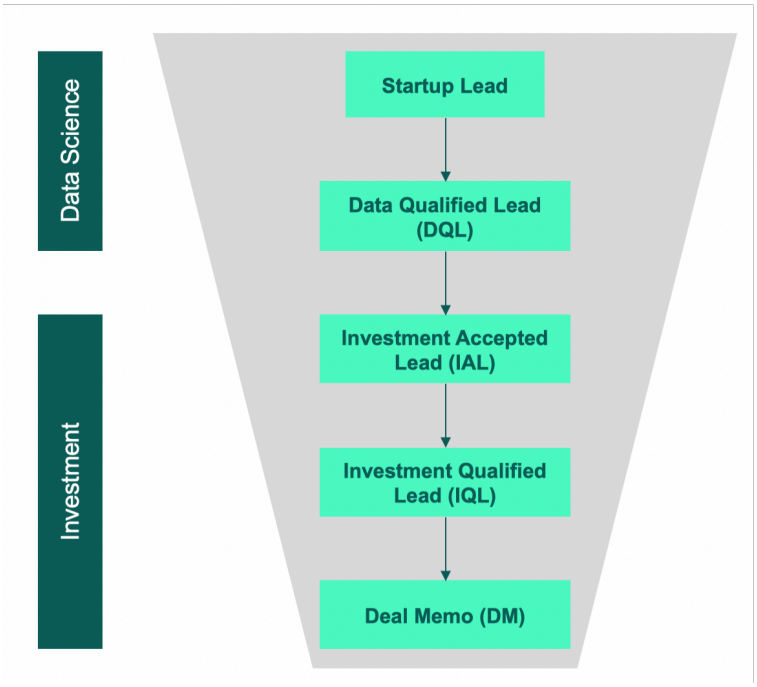

Figura 2. Representación de un embudo de inversión de una firma de VC.

En el **Capítulo 3 – Dimensionamiento del Mercado y Metodología de Extracción de Datos**, se realiza un análisis de los diferentes tipos de fuentes de startup leads online, tomando especial atención a la calidad de la información, la cantidad de startups encontradas en cada tipo de fuente, y el nivel de representación del ecosistema de la geografía de interés de las fuentes. Crunchbase y Pitchbook, dos bases de datos Premium extensamente empleadas en VC, son evaluadas junto con un conjunto de fuentes de datos públicos que incluirán, entre otras, paginas web de aceleradoras, incubadoras y otras firmas de VC.

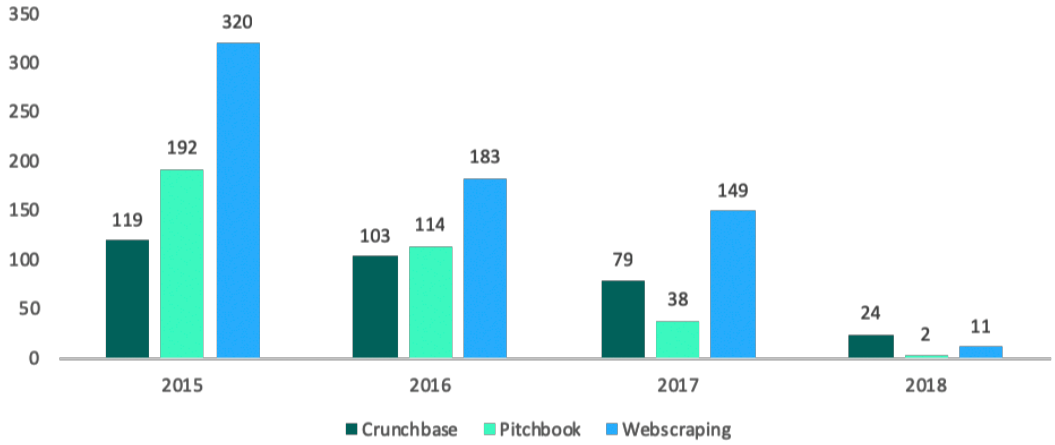

Figura 3. Número de startups por año de fundación registradas en cada tipo de fuente de datos revisada.

El capítulo 3 continuará con un estudio de diferentes alternativas para llevar a cabo el “scraping” de datos a partir de las fuentes online, con el fin de extraer la información de una forma automática y sentar las bases de la cualificación automática en el siguiente capítulo. El estudio continuará con un análisis comparativo entre herramientas de “scraping”, incluyendo plugins de buscadores y plataformas incorporadas para este propósito, y bots programados en entornos de desarrollo integrado mediante marcos de programación, como Scrapy y Apify.

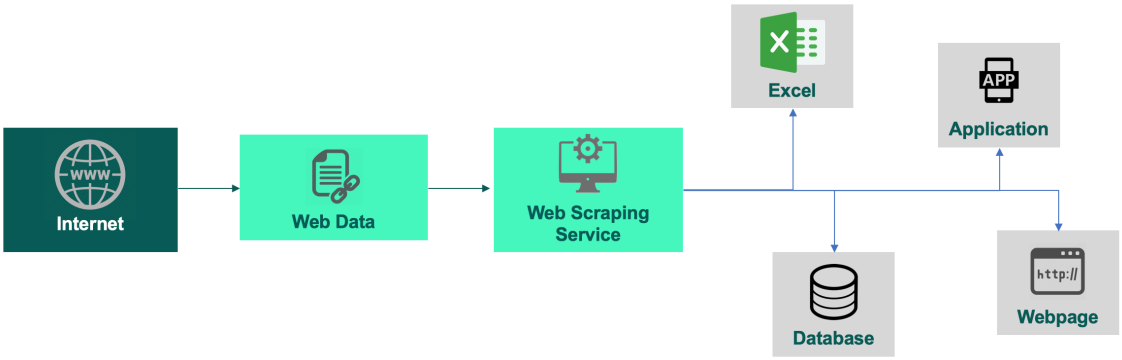

Figura 4. Funcionamiento de un proceso de Web-Scraping y extracción de datos online.

En el **Capítulo 4 – Desarrollo de un modelo de NLP para Clasificar Startup Leads**, se provee de un análisis sobre algoritmos y metodologías de machine learning para procesar y clasificar los datos extraidos en el capítulo anterior. En este capítulo se hará especial hicapíe en modelos de análisis de texto (NLP) de acuerdo con la naturaleza de los datos extraidos a partir de las fuentes de datos online. Finalmente, el capítulo proveerá la implementación de un modelo de regresión “Stacking”, que consistirá en una regresión de predicciones de dos modelos base: un modelo de “Naive Bayes” y una red neuronal convolucional. El capítulo terminará con la evaluación y validación de ambos modelos base de forma separada, y su combinación a través del meta-modelo.

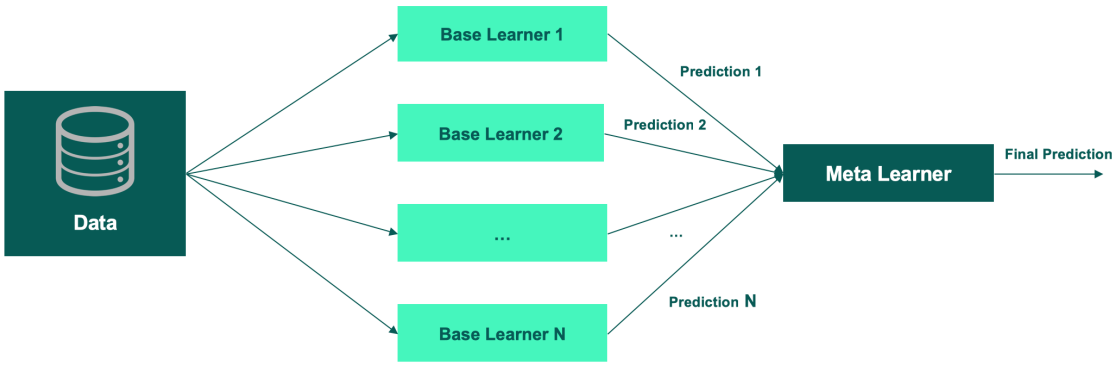

Figura 5. Estructura general de un modelo Stacking.

Finalmente, el **Capítulo 5 – Conclusiones**, proveerá una visión general del proceso, un análisis de sus ventajas y limitaciones, y las tasas de conversión resultantes para cada escalón del pipeline de inversión, provistas en la figura 6.

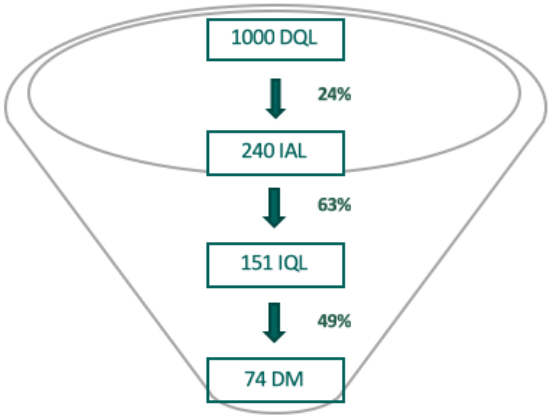

Figura 6. Tasas de conversión de cada escalón del proceso en el pipeline de inversión.

La figura 7 ilustra la estructura final del proceso de outbound sourcing.

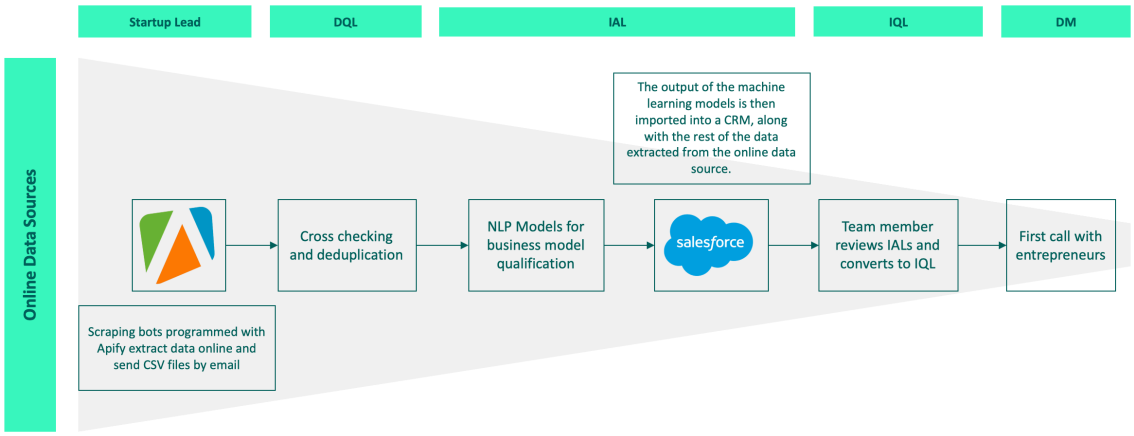

Figura 7. Estructura final del proceso de outbound sourcing.

DEVELOPMENT OF A VENTURE CAPITAL AUTOMATIC OUTBOUND SOURCING PROCESS TO FIND AND TARGET STARTUPS ONLINE

Author: Orden López, Gonzalo de la.
Director: Fernández Medrano, Sebastián.
Collaborating Institution: Samaipata Ventures SGEIC SA.

ABSTRACT:

The purpose of this project is to build a lean and scalable process to find and target prospective pre-seed investments in venture capital from online data sources across Europe.

On **Chapter 1 – State of the Art**, one can find an exhaustive description of the current situation in the European tech ecosystem, from the development of the Venture Capital (VC) industry in Silicon Valley to its settlement in Europe, the entrepreneurial talent richness of its tech communities, as well as a presentation of the different typical sourcing processes employed currently by VC firms, and the difficulties encountered in the process of finding prospective portfolio companies in such fragmented tech hubs. The chapter will conclude with a general overview of the approach to be followed in the subsequent sections.

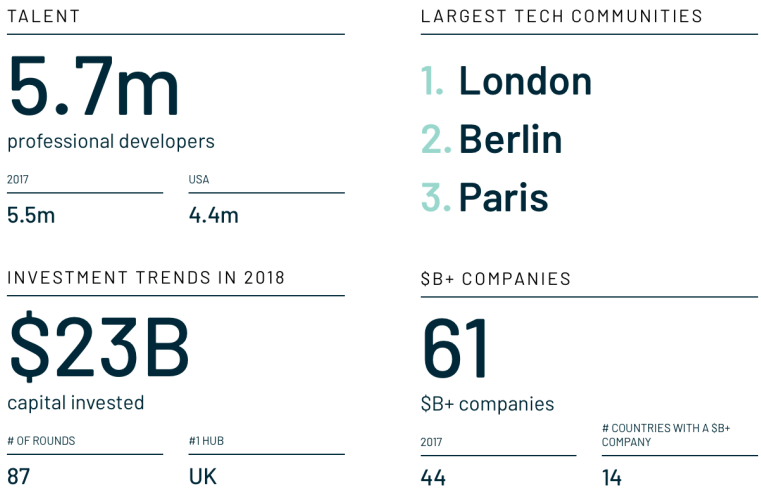

Figure 1. Key figures of the European VC ecosystem (Atomico, 2018)[1].

Chapter 2 – Definition of a Sales Pipeline, provides the structure of a sales pipeline, the process typically used within sales and marketing teams to qualify and select prospective opportunities and convert them into real opportunities, finally closed in a transaction. This chapter studies and provides a specific structure and terminology customized to approach the investment process in a

[1] https://2018.stateofeuropeantech.com

VC firm, in which each startup passes through a set of stages, each one containing a set of filtering criteria. The pipeline built in this chapter will serve as a structure in which to build up the outbound sourcing process in the following chapters.

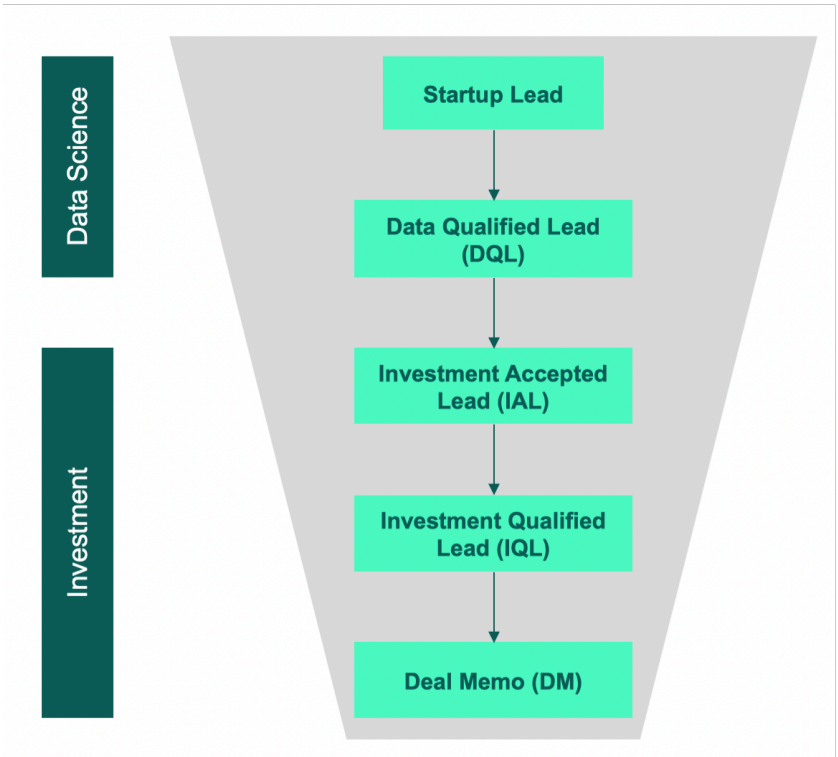

Figure 2. Representation of the Investment Funnel of a VC Firm.

On **Chapter 3 – Market Sizing and Data Extraction Approach**, there is an analysis about different types of online sources of startup leads, taking special consideration in the quality of the information, the quantity of startups found for each kind of source, and the level of representation of the sources as of the startup ecosystem of the geography of interest. Crunchbase and Pitchbook, two premium databases widely used in VC, will be assessed together with a set of public data sources, including websites of accelerators, incubators, and other VCs.

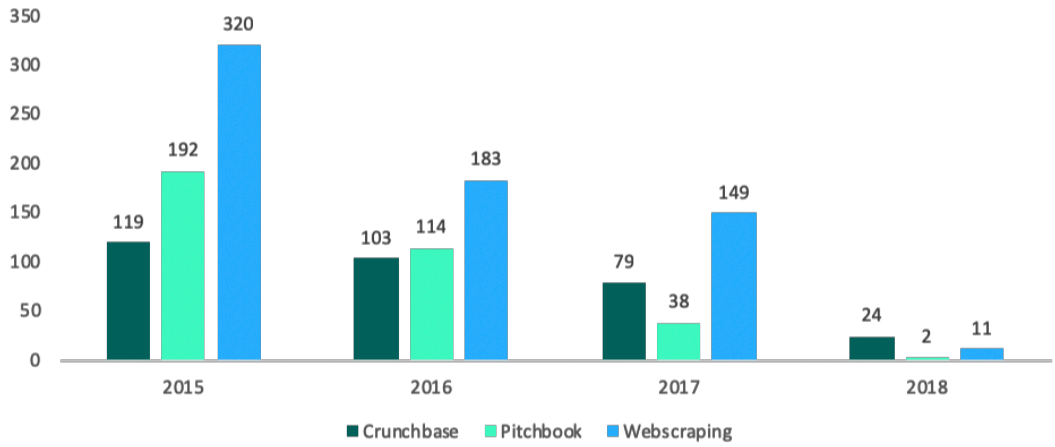

Figure 3. Number of startups per year of foundation registered for each data source reviewed.

Chapter 3 will continue with a study of different alternatives to perform data scraping of the online data sources, to extract the information automatically and be able to qualify it in more detail in the next chapter. The study will continue with a comparative analysis between scraping tools, including browser plugins and built-in platforms, and bots coded in an IDE using different programming frameworks, such as Scrapy and Apify.

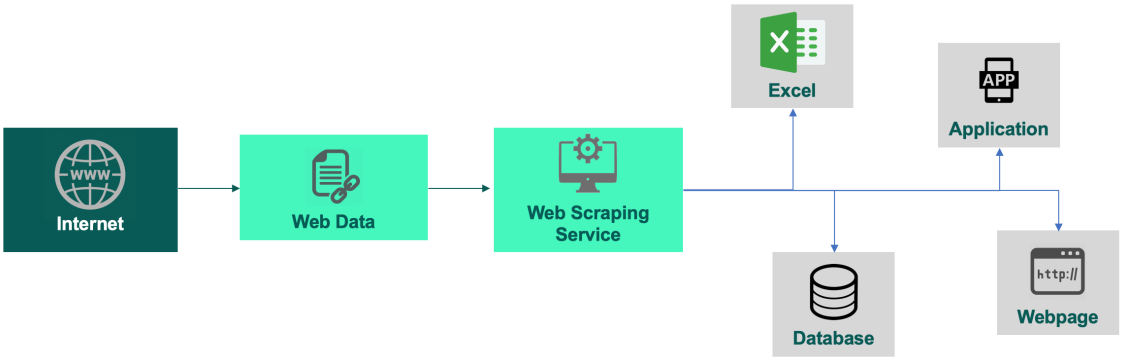

Figure 4. Functioning of a Web Scraping and Data Extraction Process.

On **Chapter 4 – Development of an NLP model to Classify Startup Leads**, there is an analysis about machine learning algorithms and methodologies to process and classify the data extracted in the previous chapter. Here, text analysis (NLP) models will be given a special focus due to the nature of the data extracted from the online data sources. Finally, this chapter will provide an implementation of a stacking regression model, consisting on a regression of predictions from two base learners: a Naïve Bayes model, and a convolutional neural network. The chapter will end up with an evaluation and validation of both base learners separately, and their combination through the meta-learner.

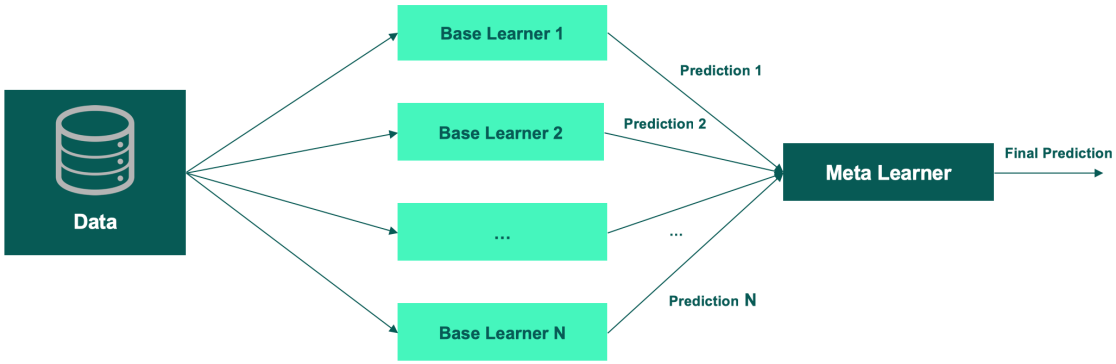

Figure 5. General structure of a stacking model

Finally, **Chapter 5 – Conclusions**, will provide a general overview of the process, an analysis of its advantages and limitations, and final conversion rates for each pipeline stage of the investment process, given in Figure 6.

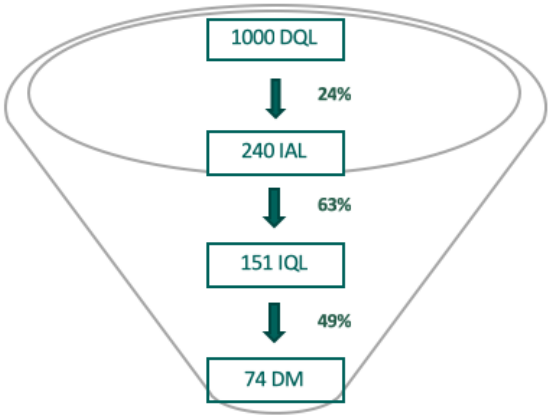

Figure 6. Conversion rates of the process for each stage in the investment pipeline.

Figure 7 depicts the final structure of the outbound sourcing process.

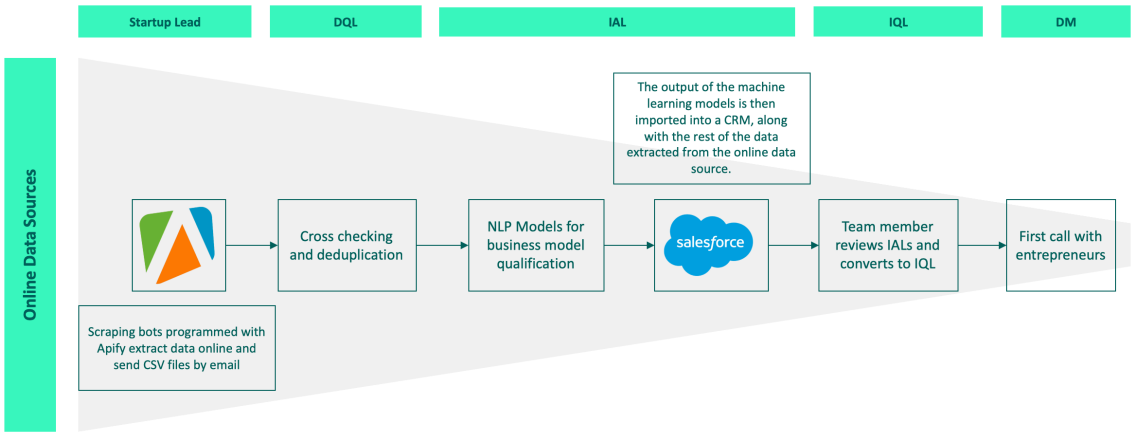

Figure 7. Final structure of the outbound sourcing process.

COMILLAS
UNIVERSIDAD PONTIFICIA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA – ICAI
Máster en Ingeniería Industrial

Development of a Venture Capital Automatic Outbound Sourcing Process to Find and Target Startups Online

– Gonzalo de la Orden López –

Director: Sebastián Fernández Medrano

Madrid
June, 2019

Index

List of Figures

“Success is not final, failure is not fatal:
It is the courage to continue that counts.”

Winston Churchill

Acknowledgments

To my family. Thank you for your unconditional support and faith in all my endeavors. Thank you for the education you have struggled to give me and for all your love and understanding. Without you, none of this would certainly have been possible.

To my director in this project, Sebastián. Thank you for placing your trust in me from day one, and for mentoring me with such an emotional intelligence.

To the entire team at Samaipata, for allowing me to work and learn from you in such a healthy work environment, and for changing my perspectives in so many ways.

Chapter 1 - State of the Art

1.1. Introduction to Venture Capital Investing

Venture Capital (VC) is a form of private equity investments focused on providing financing to emerging companies with high growth potential, but with extreme conditions of uncertainty.

Venture capital was born during the 1960s and 1970s in the US to start and expand companies that were exploiting breakthroughs in technology. Founded in the late 50s, Fairchild Semiconductor is considered the first venture-backed startup, funded by Arthur Rock, an early venture capitalist from New York. Figure 1 depicts the cover of the Time magazine of January 1984, where Arthur Rock is given credit as a venture capitalist.

Figure 1. Cover of the Times Magazine, January 1984.

The growth of the VC industry was fueled in the early 70s by the birth of many semiconductor companies in Santa Clara Valley, California, together with the first computer companies using their devices. Independent VC firms were created to give financing to the new wave of disruptive companies established in the valley.

The big successes in the decade of the 70s gave the green light to new agents, and from few dozens of players by early 80s, there were over 700 VC firms by the end of the decade. The number of investment firms multiplied, and so did the capital managed by these firms, which increased from $3 billion to $31 billion in a decade (NY Times, 1989).

Even though there was some friction in the VC industry during the early 90s due to excess supply of technology IPOs and a super competitive market for “hot” deals, the substantial increase in financing, together with outstanding results of some venture-backed companies, such as Netscape or Yahoo!, generated a tremendous interesting in investing in internet companies.

Andrew Metrick, a professor at Yale School of Management, refers to the first 15 years of modern VC industry beginning in the 1980s as the “pre-boom” period, referring to the boom that started in 1995 and lasted until the burst of the Internet Bubble in 2000 (Metrick, 2007). Figure 2 (Pwc, 2017) depicts the total investment amount in billions of dollars, together with the number of deals, from 1995 to 2017 in the US.

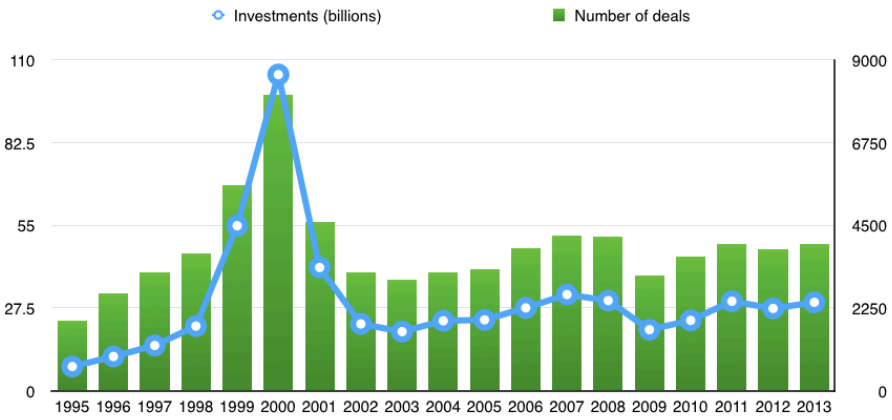

Figure 2. Total U.S. Venture Capital Investments.

Since the 1970s, VC firms have provided financing to some of the companies with the largest market capitalization, and with the most disruption in its market, in the world today, including the top five publicly traded companies by market capitalization (Statista, 2018). Figure 3 depicts the evolution of the top five publicly traded companies from 2001 (Visual Capitalist, 2016).

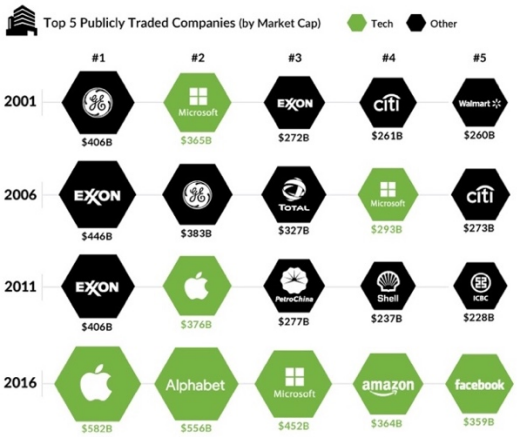

Figure 3. The Largest Companies by Market Cap Over 15 Years.

1.2. Venture Capital in Europe

Venture Capital is frequently associated with Silicon Valley, where many entrepreneurs and investors across the world arrive daily, and several billions of dollars are pulled into the US economy every year. However, in the last two decades, there has been a substantial international raise in other major global economies that have tried to emulate the entrepreneurial ecosystem of "the Valley."

1.2.1. European Share in the International Venture Ecosystem

Even though Europe has double the population of the US and a similar sized economy, European VC activity accounts for a fifth of that in North America, where the venture capital industry was established three decades earlier and consequently is currently significantly more developed. Figure 4 (World Economic Forum, 2017) depicts the share of top geographies by total venture capital invested in 2014.

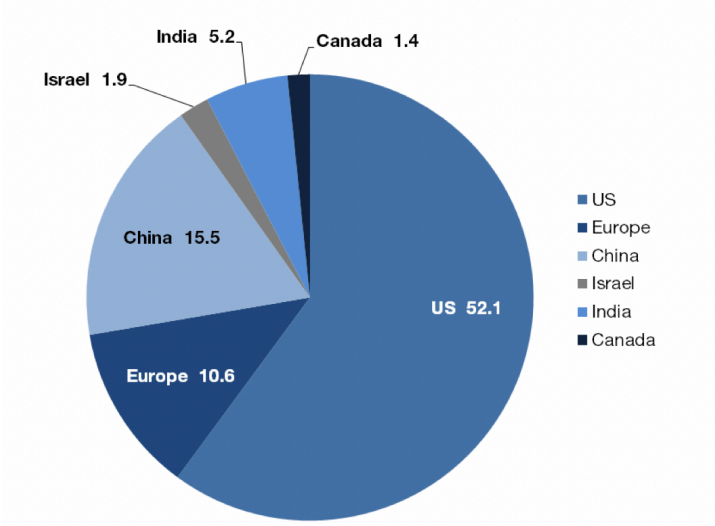

Figure 4. Top countries for total venture capital invested.

In Europe, the nature of entrepreneurship has changed significantly over the course of the last decade. From 2012 to 2017, more than 16,000 European companies have received venture capital financing, and Europe has seen startup valuations in excess of $1 billion, such as Skype, Delivery Hero, BlaBlaCar, Spotify and Cabify.

On the other hand, the European VC ecosystem benefits from lower valuations than in the US, since high competition drives prices up, and so called "unicorns" – billion-dollar companies, in the US are valued in average at 46 times their annual revenue, compared with 18 times for their counter parties in Europe (World Ecnomic Forum, 2017).

Figure 5 (Insights, 2016) depicts a European tech map with the most well-funded VC backed tech startups by country since 2014, according to CB Insights.

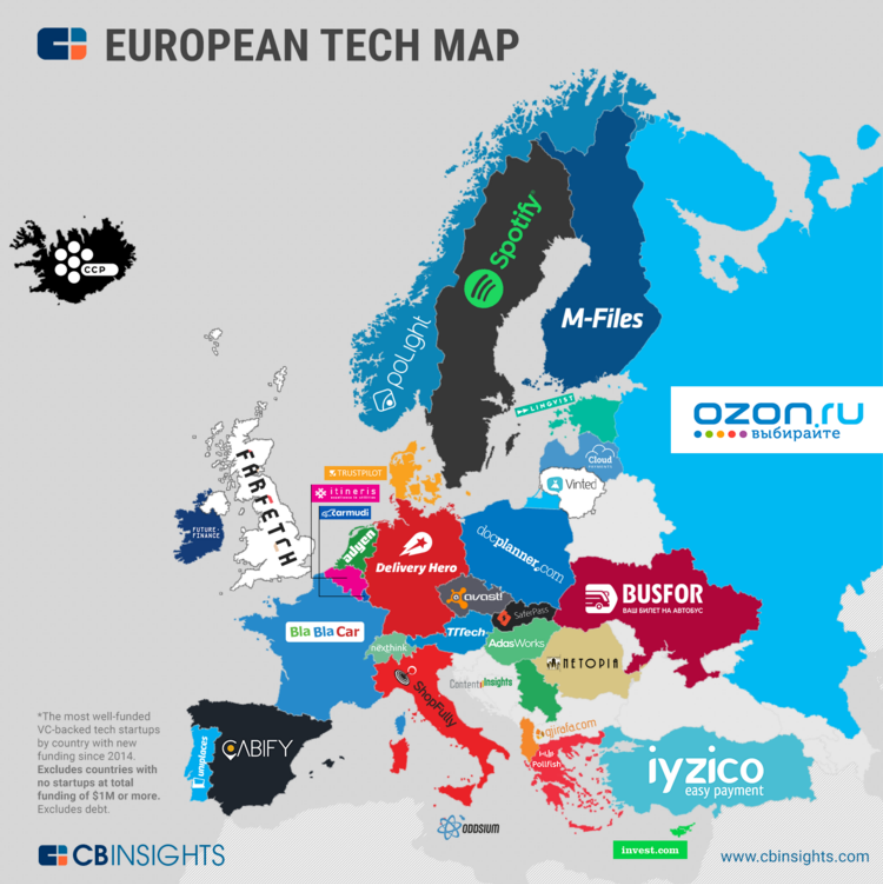

Figure 5. European tech map with the most well-funded VC backed tech startups by country (2014).

1.2.2. European Increase in VC Funding

Even though the European economy remains heavily dependent on traditional industries such as construction, industrial manufacturing and transportation, Europe is experiencing an ever-widening gap in the indexed growth rates of the tech and non-tech industries. As it can be seen in figure 6 (Atomico, 2018), European tech has grown 194% of its relative value in 2002.

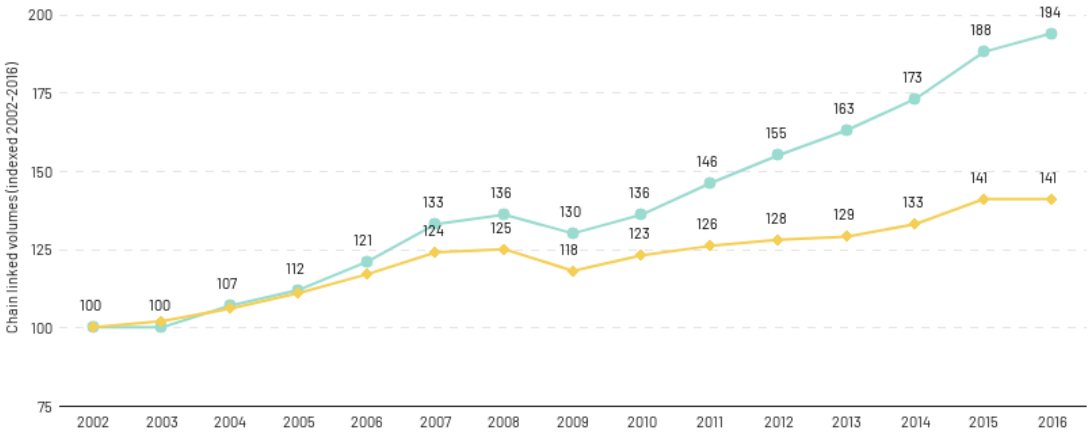

Figure 6. Chain linked volumes of tech and non-tech GVA (indexed 2002-2016).

Figure 7 illustrates the European VC financing activity from 2010 to 2017 (Pitchbook, 2018).

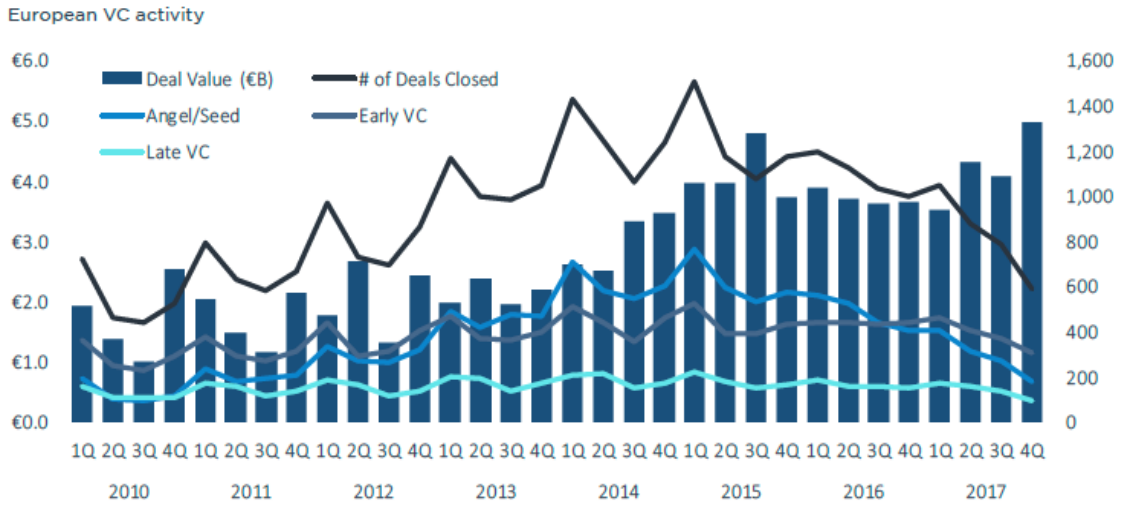

Figure 7. European VC activity.

1.2.3. European Pool of Tech Talent

One of Europe's strengths as an international tech force is its deep talent pool. An increasing number of cities considered tech hubs are distributed across all Europe, and they are connected by Europeans and non-Europeans alike. Also, the rise in funding and the growth of Europe's tech hubs is attracting European founders back from Silicon Valley, and students looking to fund their own companies.

According to StackOverflow (StackOverflow, 2018), Europe's workforce registered a total of 5.7 million european developers in 2018, compared to 4.4 million the same year in the US. The countries with a higher weight in this number were Germany, UK and France. Spain ranked the 6th in Europe. Figure 8 depicts the number of professional developers by country between 2017 and 2018 (Atomico, 2018).

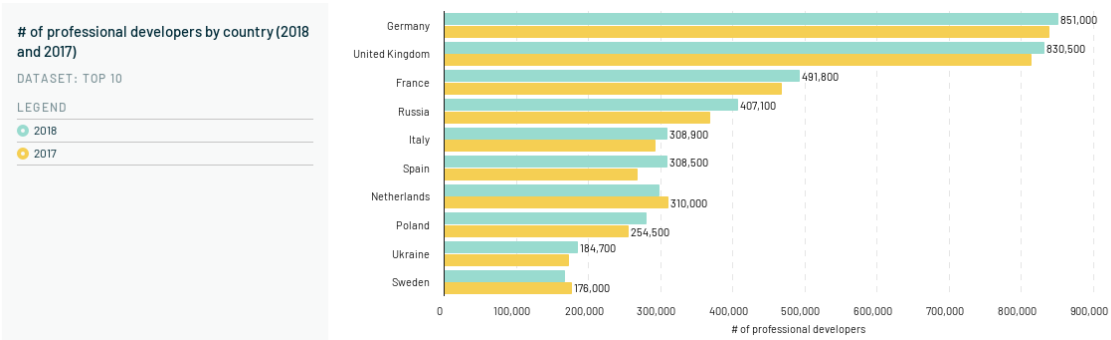

Figure 8. Number of professional developers in Europe by country (2018 and 2017).

Figure 9 below provides a density map of the distribution of professional developers across Europe (Atomico, 2018).

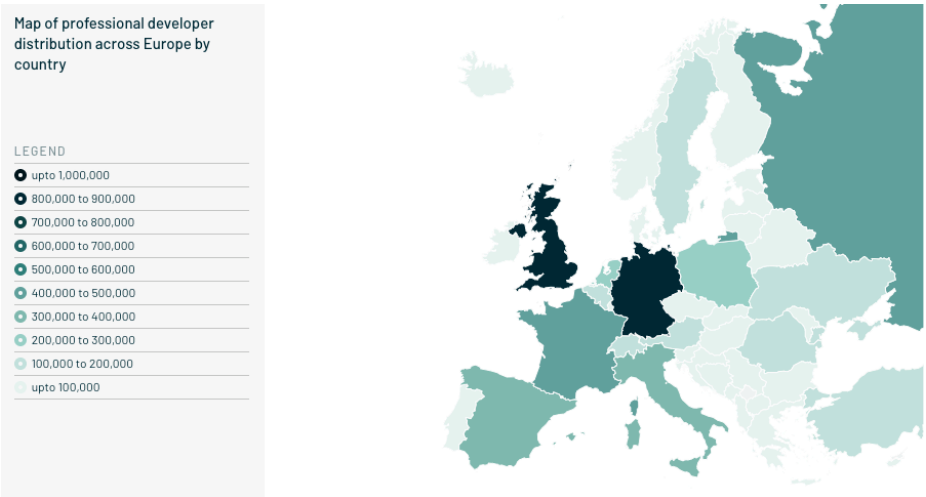

Figure 9. Map of professional developer talent pool across Europe by Country.

1.3. Finding Deals from the Perspective of a VC Firm

From the perspective of its investors, a startup's lifecycle spans from idea to an eventual exit through an initial public offering (IPO), acquisition, or liquidation. In fact, most VCs define themselves by the stage of financing in which they concentrate their investments. Today, the standard labeling of the rounds is related to the stage of the company as follows.

- Early-stage: Pre-seed, Seed and Series A.
- Mid-Stage: Series B and Series C.
- Late-Stage: Series D or later.

Figure 10 (Simon Duchatelet, The World Bank, 2018) provides a visual representation of the different stages of venture financing, from seed capital to IPO and secondary offerings.

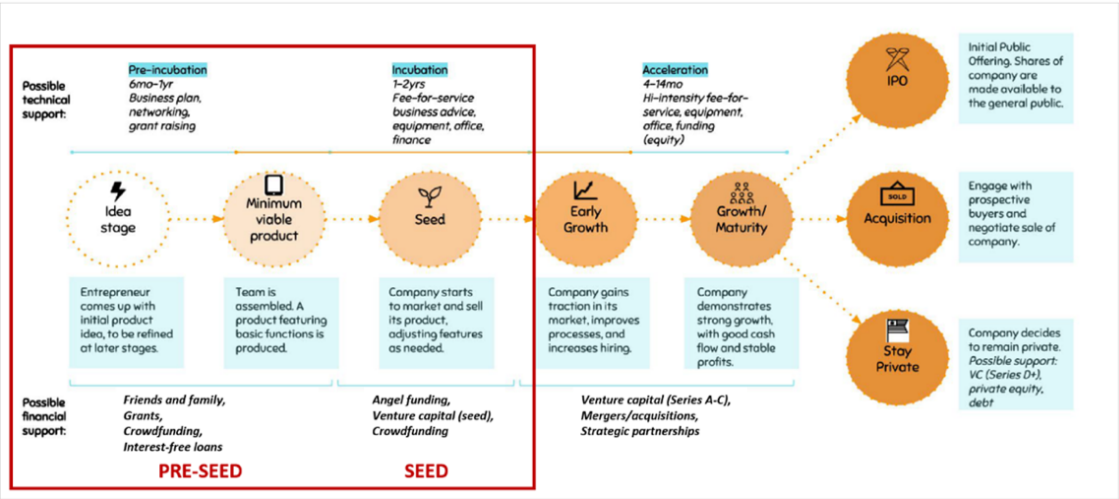

Figure 10. The different start-up funding rounds and stages of development.

For discovering prospective portfolio companies, it is imperative for the VC firm to consider the stage of financing in which to invest since the amount and perceptibility of the startups vary considerably on this factor. To better understand this, figure 11 (Jason

Rowley, Crunchbase News, 2017) depicts the proportion of startups in the US that, having raised capital in previous rounds, do so in the next one.

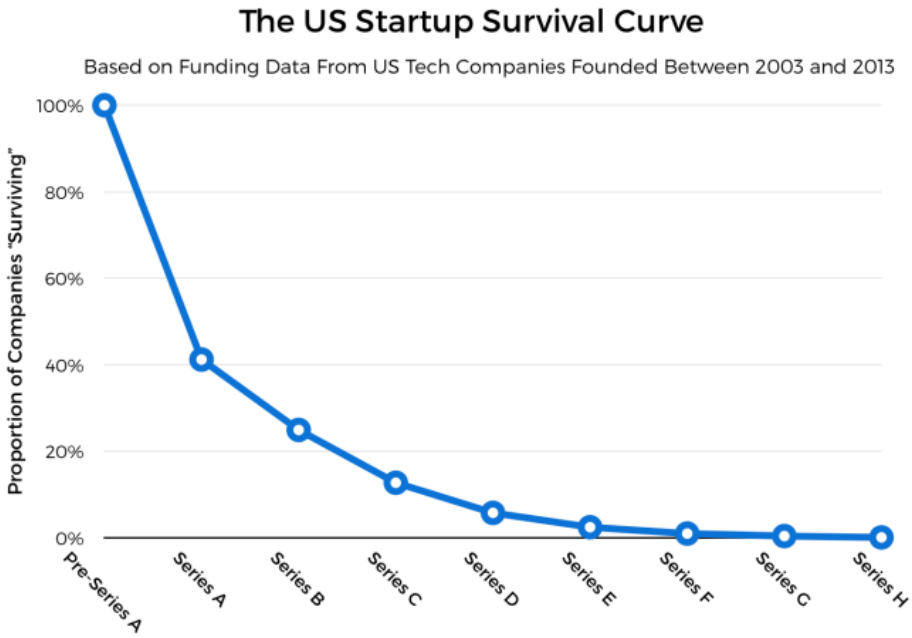

Figure 11. The US Survival Curve.

Therefore, out of 100 startups that complete an early stage round in the US, it can be expected to see approximately 40 that make it to the next round of financing (Series A). Hereafter, the percentage drops dramatically on each stage onwards. It inevitably follows that, since the number of companies in earlier stages is significantly higher than in later stages and their mortality is much higher, it is a more difficult task to detect and evaluate all of them.

As previously mentioned, the growing number of talented entrepreneurs in Europe is attracting increasing amounts of VC funding. For a VC firm specialized in early-stages, it is a differential competitive factor to be able to detect investment opportunities at the very start of the fundraising process, especially in the case of desirable deals, that meet the target financing amount rapidly, and many firms compete to be a part of their cap table. Moreover, the fact that entrepreneurs are very scattered around Europe adds another layer of complexity to this concern.

Most VC firms strive to find early-stage companies that fit with their investment criteria: business model, stage of financing, industry, etc. Many of these firms rely on pure inbound, consisting on: attending events, building a network of entrepreneurs in the VC ecosystem, and continuously communicating with potential sources of startups, such as accelerators and other VC firms. The process of finding new prospects is considered, in the vast majority of the cases, the bottleneck of the investment process.

A valuable approach to shape the startup detection process of a VC is through a sales pipeline, initially used by sales teams to control its process of finding prospects, called leads in sales terminology, and closing new clients. In the context of venture capital, we will talk about startup leads to refer to prospective startups that have been identified but not yet closed as an investment.

In VC, startup leads can come from different sources or activities, depending on which we will talk about inbound or outbound leads.

- Inbound leads are leads found passively. They include any startup founder looking for financing and contacting the VC firm directly, intro calls from a previously established network in the industry, online forms, etc.

- Outbound leads are different from inbound leads in that it is the VC who actively arrange a plan to discover them. We differentiate between manual and automatic outbound.

    - Manual outbound: leads generated from a previously built and nourished network of other VCs, Business Angels (BAs), accelerators, incubators, entrepreneurs, etc.
    - Automatic outbound: any automated process that relies on technology to find and place the startup leads in the sales pipeline.

This document will have the purpose of building a semi-automated standardized process to screen online data with data crawling bots and, due to the significant amount of data that this process will scrape, providing a layer of automated intelligence through machine learning algorithms.

1.4. Approach for Building an Automated Outbound Sourcing Process

As mentioned in the previous section, this document aims to build a scalable process to detect and filter prospective startups from the perspective of a VC firm. For this purpose, the next steps will be followed:

1. Determination of an appropriate sales pipeline to approach the entire process of the initial segment of the investment process from initial detection of prospective portfolio startups to the first call with their entrepreneurs by a member of the investment team.

   A sales pipeline will be instrumental in providing a visual overview of the outbound process and in managing and delivering business forecasts on how many deals are expected to be closed, given a certain quantity of prospects on the top of the investment funnel.

2. Market sizing of the startup ecosystem with an online presence in the geography of study, in this case, Spain.

   The market sizing will consist on assessing a set of online data sources, including premium databases, and public websites (websites of accelerators and incubators, public databases, social networking services, etc.), ending with a comparative ranking based on a set of parameters.

3. Study of technologies and programming techniques to automate the process of extracting the data from the set of online data sources, into a manageable file that a set of machine learning algorithms will process.

4. Implementation of machine learning models to automate the process of assessing the data extracted online from startups. This implementation will follow a previous study of the quality and reliability of the data retrieved by the scraping bots.

5. The data, once retrieved and classified, will be imported into a CRM, where members of the investment team will be able to review the information in an organized manner and will decide whether to follow with the appropriate due diligence on the company or not.

Chapter 2 – Definition of a Sales Pipeline

2.1. Sales Pipeline Management

In most cases, the process of discovering prospective portfolio startups is considered the bottleneck of the VC firm. From that set of startups, those who adhere the firm's investment thesis are approached to do a deal memo: a 30 minutes call involving a member of the VC team and the company founders to allow the VC to understand the startup business.

Doing deal memos requires a significant portion of the week of an investment professional, whose central tasks are finding brilliant entrepreneurs, building a network in the entrepreneurial ecosystem, and performing due diligence of businesses.

By disposing of a methodology to visualize and measure the number of startups in each part of the investment process, especially in the first stages, the firm can make forecasts based on conversion rates from stage to stage, and therefore manage the firm's resources in a more efficient way.

Due to the similarity in the process with that of a sales team, which also needs to find prospects and close new clients, this chapter aims to extrapolate a compelling sales pipeline with a sound terminology for the case of the outbound process of a VC firm. Each stage down the pipeline will designate a higher level of qualification of the startup as a potential investment by the VC.

2.1.1. Introduction of a Sales Pipeline

In sales terminology, the sales pipeline is a systematic and visual representation for selling a product or service (Pipedrive, 2019). In sales, it is beneficial to understand at which stage each prospective customer is, whether the company has enough prospects

on board to achieve a quota and whether an opportunity needs any special consideration.

A sales pipeline is a composite of individual customer sales' funnels. A sales funnel pictures every stage in the process that a sales rep takes to move a deal from start to close. Figure 12 depicts a standard representation of a sales funnel (Freshworks, 2019).

Figure 12. Standard representation of a sales funnel.

A prospect, or lead, starts at the top of the funnel and subsequently advances through a series of sequential stages, subject to meeting specific criteria at each until it reaches the bottom, considered the closing of the lead, which means that the prospect has purchased the product or service involved in the sale. Figure 12 includes the following stages.

1. Qualify: qualifying leads accurately in the first stage enables an effective sales process. Sales teams use techniques such as BANT, to determine if a prospect has the Budget, Authority, Need and Timeframe to qualify as a potential purchase. Only those leads that meet the fitness criteria will be considered prospective opportunities and will descend through the funnel.

2. Meet: a sales rep approaches qualified leads by setting up a call or demo of the product or service.

3. Propose: once a sales rep meets the prospect, the lead passes to a sales employee, who presents a deal proposal and additional required agreements.

4. Close. The final stage is to close the deal. The prospect may either turn into a customer or leave the pipeline as a lost opportunity.

There are, however, many different nomenclatures in sales and marketing assigned to the different stages of a customer's journey, and the one given here pretends to be just an example. The sales pipeline represents the aggregate of all the customer's sales funnels, and provides a general overview of the health of the process.

2.1.2. Internal Teams and the Sales Funnel

Figure 13 depicts a particular representation, with a specific nomenclature, of the sales funnel of a company with separate marketing and sales teams.

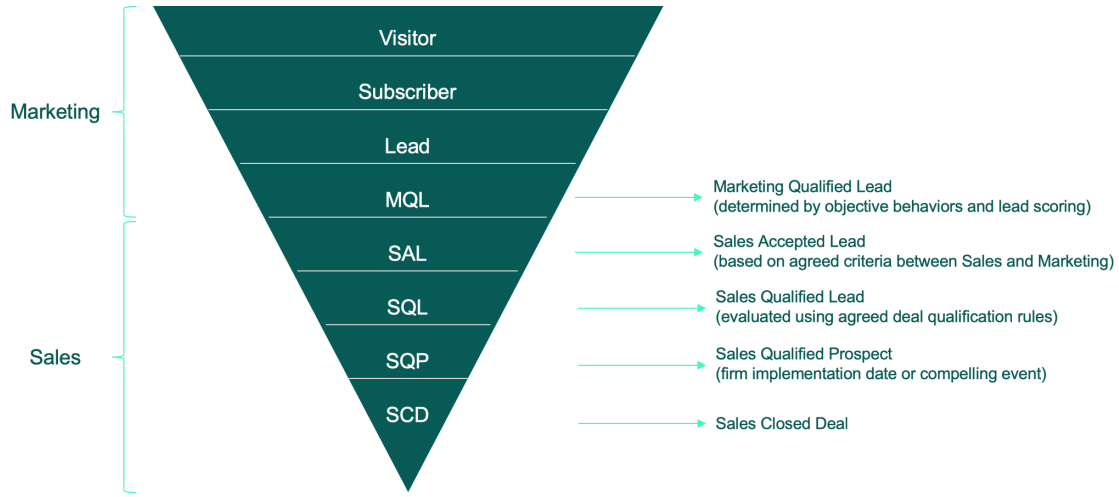

Figure 13. Example of Sales Funnel employed in Marketing and Sales.

Figure 13 separates the marketing and sales teams in different parts of the funnel. The marketing unit is responsible for creating awareness of the products and services and detecting prospects, which position in the funnel as MQL, or Marketing Qualified Leads, through objective behaviors and a scoring.

The MQLs that the sales team accepts based on agreed criteria are then converted to SALs, or Sales Accepted Leads, and, if they meet the criteria specified, subsequently to Sales Qualified Leads Sales Qualified Prospects, and finally Sales Closed Deals.

2.2. Investment Pipeline of Startup Leads

In this section, a terminology for each stage of the investment funnel will be built by adapting the nomenclature of a sales funnel to the specific process of finding and evaluating startups carried out by a VC in the initial stages.

Even though a VC does not typically have marketing and sales teams, the functioning of these teams is analogous to that of a data science and an investment team, respectively, as indicated below:

- A marketing team is comparable to a data science team in that the latter will also use a objective set of criteria and lead scoring to market prospects. In this setting, the data science team will be responsible for screening data sources and use unbiased techniques to determine which leads should descend through the funnel, and which should not.

- The activity of the investment team has a component that is somewhat similar to that of a sales team in that it will target the leads that the data science team passes through the funnel. The investment team will begin communications with the entrepreneurs, to determine whether the company meets specific standards of the VC firm (e.g., business model, industry, market size, competition, etc.), and vice versa, since the VC firm is competing with other firms to attract to their portfolios the best startups, whose cap table is limited and usually divided among various VCs.

2.2.1. Business Model Specialization of the VC

In general, any company selling a product or delivering a service will target specific segments of prospective customers. Nowadays, this is even more applicable as more and more companies target customers with cookies and artificial intelligence to add efficiency into their marketing scheme.

In a similar fashion, a VC, aside of concentrating in startups, is usually specialized in specific segments of technologies, industries or business models. To better resemble this specialization, for the scope of this document the automatic sourcing process will target early stage startups whose business model is a marketplace or a digital native vertical brand. Both business models are briefly described below:

- Marketplaces: online platforms that aggregate demand and offer for particular products or services.
- Direct to consumer digital native brands: described in a very simply manner, brands with strong online communities around particular products, completely or partially distributed online.

2.2.2. Nomenclature of the Investment Funnel of the VC

Here, a sales funnel will be adapted to the operations of the VC. Each stage of the funnel will refer to a particular status in the evaluation process carried out by the investment team. As mentioned in the previous section, instead of marketing and sales teams, it makes more sense for a VC to talk about data science and investment teams, respectively.

Figure 14 depicts a compelling illustration of the investment funnel of a VC firm for the first stages of the investment process, from discovery of the startup, to 30 minutes call with the entrepreneurs.

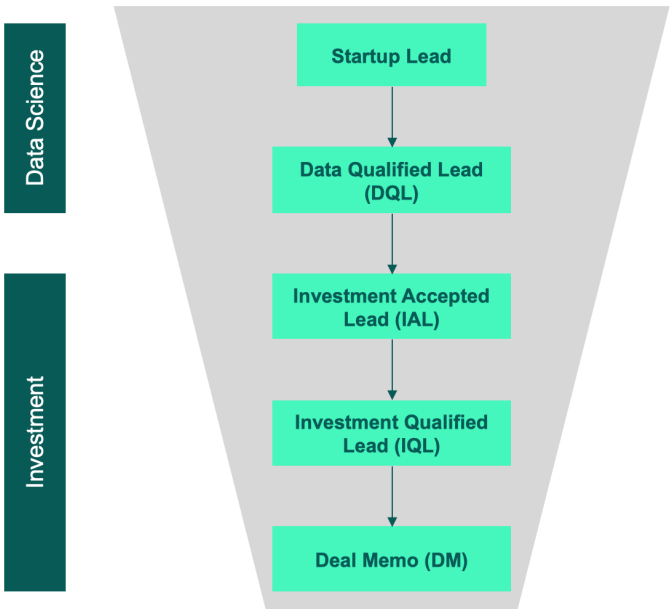

Figure 14. Representation of the Investment Funnel of a VC Firm.

The investment funnel depicted in Figure 14 will be employed to model the initial interactions with prospective portfolio companies. It is, however, limited to the first portion of the investment process, and it is critical noting that there are several more steps that involve due diligence and company-specific appraisals after a deal memo.

The stages of the investment funnel defined in Figure 14 are explained more in detail as follows:

1. Startup Lead: a prospective startup contact of any kind, usually related to a person involved in the company's operations. Since the related person is the one that will be contacted if it advances down the funnel, it is desirable for her to be the CEO or one of the founders.

2. Data Qualified Lead (DQL): deduplicated startup leads cross-checked with the VC firm's list of past companies reviewed.

3. Investment Accepted Lead (IAL): DQLs that meet established unbiased criteria, are converted into IALs. The rules implied here will rely on a set of natural language processing algorithms developed from a training set of past due diligence carried out by the VC firm.

4. Investment Qualified Lead (IQL): the list of IALs will be reviewed manually for true positives, or companies that have been correctly tagged as prospects by the machine learning models involved in the previous stage. True positives are converted into IQLs.

5. Deal Memo: IQLs that successfully fit within a set of agreed qualification rules determined by the investment team are contacted by the VC for a 30 minutes call, the first personal interaction with the founders.

After a deal memo call with the founders, if there is a match between the VC firm and the startup, the company will have to comply with to further due diligence stages, usually involving higher levels of seniority in the organizational hierarchy of the VC firm, to finally be closed as an investment.

2.3. General Overview of Chapter 2

The definition of a specific sales pipeline to model the process that a VC firm encounters when approaching prospective portfolio startups has many advantages. It defines a standard method to approach the initial stages of the investment process, providing names, and allowing the firm to measure and manage productivities at each step, in a similar fashion to a lean manufacturing plant, that involves a set of machines and processes in a specific order to build its product.

The next steps will involve encountering efficient methodologies for each stage of the process:

1. Find online sources of prospective startup leads. This process will entail researching online data sets, including subscription-based databases and public websites, to finally target the ones that provide large amounts of data with a minimum of quality.

2. Research on scalable and automatic ways to extract the information contained in the sources targeted. This research will include studying different types of web scraping algorithms, that allow performing repetitive tasks on the internet and increase productivity.

3. Research and define a set of machine learning techniques that better suit the data extracted, to use an initial objective scoring that serves to the purpose of maintaining the objectivity in the process, saving labor time reviewing all the extracted information manually, and converting DQLs into IALs.

The result will be a lean, scalable, and standardized process for detecting prospective investments online periodically. This process will help to better manage resources by focusing on specific bottlenecks, and to have a better vision of the market by picturing a representative sample of the entire startup ecosystem for each geography.

Chapter 3 – Market Sizing and Data Extraction Approach

3.1. Sources of Startups to Perform the Market Sizing

There are a large number of online-based data aggregators that allow users to find companies according to specific criteria and do research on particular industries, markets, or business models. The filtering measures of this type of platforms include headquarters country, business model, industry, and total funding amount. The majority of these platforms aggregate only user-generated content: millions of users use them to promote their organizations, analyze industry trends, and make better-informed investment decisions.

The fact that most of all, the online data aggregators are public and only contain a partial amount of the population set diminishes their reliability as a unique source of outbound sourcing. Thus, it is convenient to aggregate the data provided on scattered sources to increase the competitive advantage of the process.

The purpose of this section is to build a framework to assess online databases based on the number of startups registered and their founded year which is considered one key factor due to the substantial gap of time from founded date to registered date. For this purpose, we will use the following databases.

- CrunchBase: a business information platform with hundreds of thousands of public and private companies registered globally.

- Pitchbook: a premium web-based platform providing data and analytics to venture capital and private equity firms. The main difference with Crunchbase is that Pitchbook actively collects, organizes and analyzes deals data, instead of relying just on user-generated content, and that the information is only accessible under subscription.

- Public websites: any online website providing information about startups. These sources including sites of other VCs, accelerators, incubators, networks of business angels, or publicly available databases.

3.1.1. Crunchbase

Using a Crunchbase Pro license to filter the companies with headquarters in Spain, and with funding status including seed, early-stage venture, and late-stage venture, approximately 949 companies result as of March 2019. Figure 15 provides a plot depicting the number of startups by founded date for the given sample.

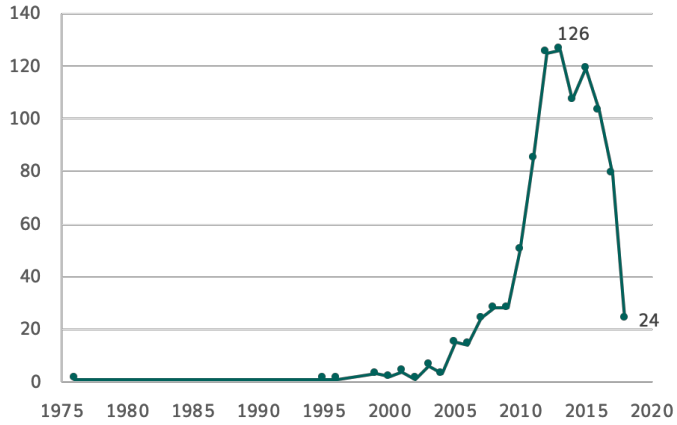

Figure 15. Companies registered in Crunchbase per year of foundation.

As seen in Figure 15, the number of startups registered by year of foundation significantly starts to increase in 2010 and plummets in 2015. However, the number of startups born in Spain is known to have increased in the last five years since the level of investments, the GDP per capita and the number of VC firms and accelerators have also increased in the previous few years. Figure 16 (Asociación Española de Capital, Crecimiento e Inversión (ASCRI), 2018) depicts the total investment amount in Spain, in millions of euros, by type of investment entity, from 2006 to 2017.

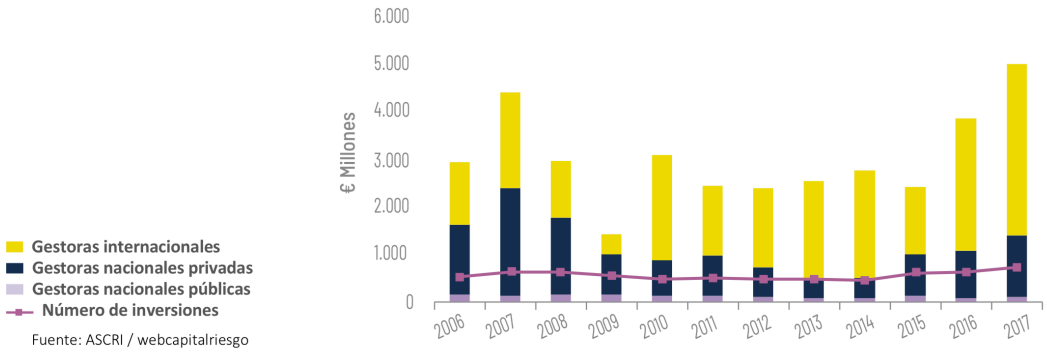

Figure 16. Private Equity investment amount in Spain per year by type of entity.

As seen in Figure 16, the investment amount has almost doubled from 2015 to 2017, which is inconsistent with the number of companies obtained for the same years in Crunchbase, considering a positive correlation should be expected between the two variables. The substantial decrease in the quantity of companies founded after 2015

registered in Crunchbase can be due to a lag between the company's foundation and the time of registration in the database.

3.1.2. Pitchbook

Using Pitchbook to assess the number of companies per year of foundation in Spain results in a significantly lower amount compared with that of Crunchbase. Figure 17 depicts the number of companies per year of foundation from 2015 registered both in Crunchbase and Pitchbook.

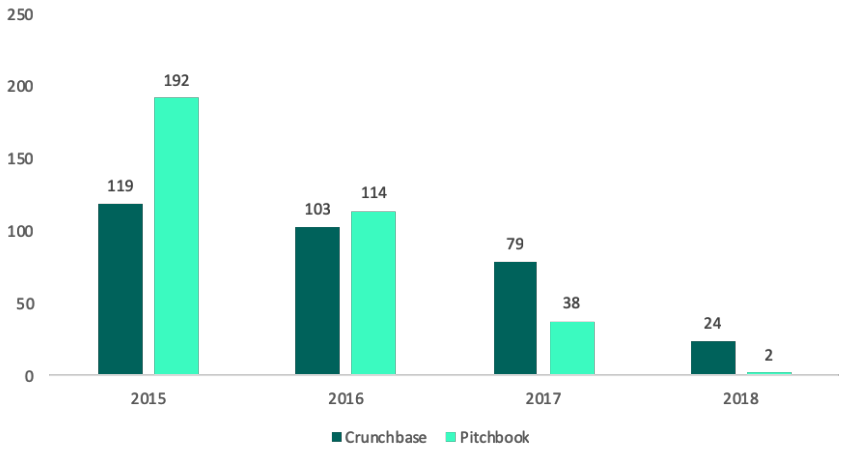

Figure 17. Comparison of number of companies registered in Crunchbase and Pitchbook per founded year.

Even though the quality of the information in Pitchbook is significantly higher and more accurate, the fact that Pitchbook does not depend on user-generated content results in a smaller set of startups compared to Crunchbase, especially in most recent years.

Pitchbook also has direct competitors in its price target and type of customers, principally CB Insights. However, a comparison between the two reveals that Pitchbook has a higher volume of startups registered in Europe than in the US compared to CB Insights, as of October 2018. The figure below depicts the difference between Pitchbook and CB Insights in terms of volume of companies registered for each of the five countries represented.

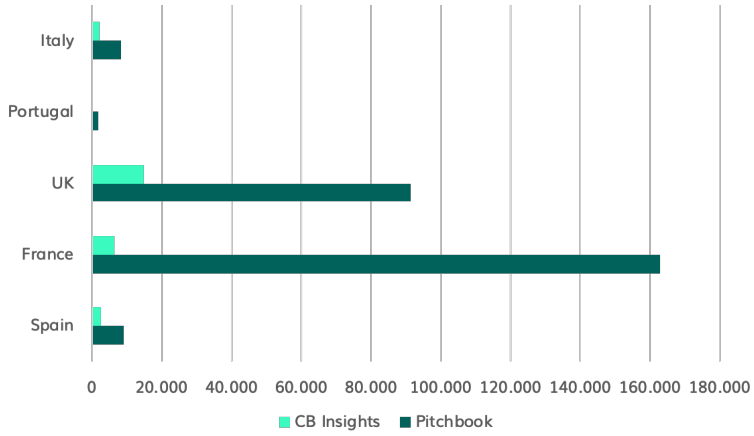

Figure 18. Difference in volume of companies registered between Pitchbook and CB Inisghts.

3.1.3. *Websites of Sources of Deals*

In this section, a list of websites has been elaborated using Crunchbase by filtering for early-stage investors in Spain (including accelerators, incubators, and VCs), obtaining 553 sources in total.

Subsequently, the list of websites has been analyzed on a one-by-one basis to scrape all the startups contained in the source. In the end, the screening has resulted in a set of 820 companies after deduplicating for the same companies registered on different websites. The figure below depicts the number of companies obtained from each type of source per year of foundation.

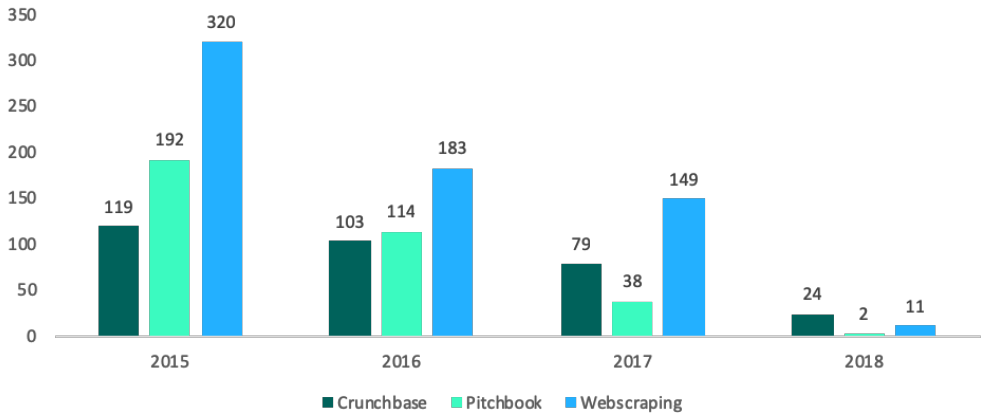

Figure 19. Number of companies registered in sources websites per year of foundation.

As depicted in Figure 19, the number of companies obtained from public websites decreases from 2015 to 2018 in a 96,56%, flagging the effect of the time-lag between a company's foundation and its apparition in a website.

It is also worth noting how many startups each type of source has contributed to the overall result. Figure 20 depicts the total number of sources reviewed, and the number of startups identified per type of source.

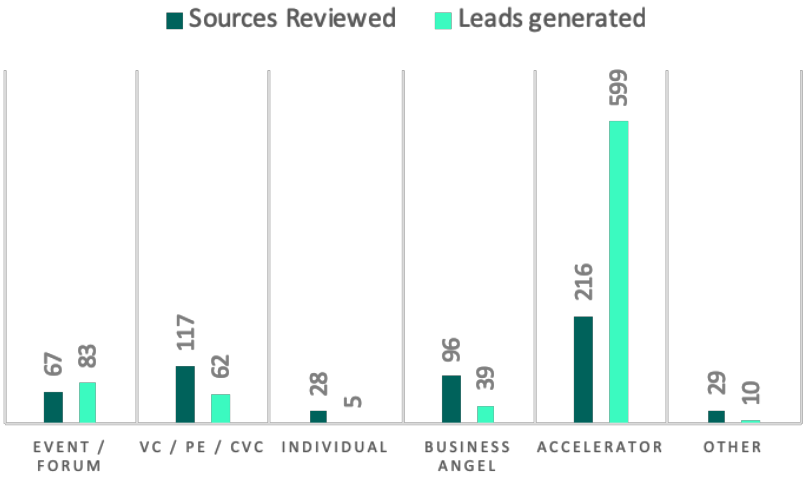

Figure 20. Distribution of startups detected per type of source.

As observed in Figure 20, the websites of accelerators are the ones that contribute the most to the overall result. The Pareto diagram in Figure 21 illustrates how this type of website alone account for more than 70% of the total startups detected.

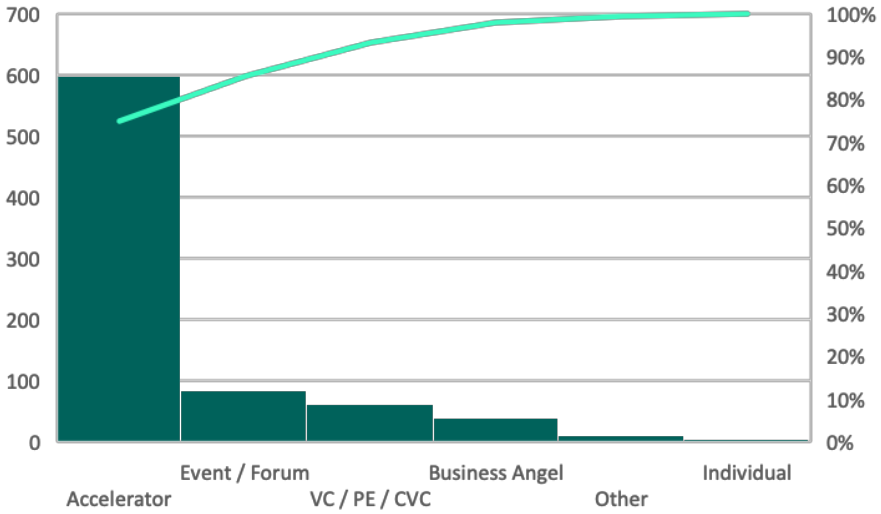

Figure 21. Pareto diagram: startups per type of source.

3.1.4. Comparative Analysis of Online Data Sources

Finally, this section proceeds to compare all the three online data sources to assess the quality and the quantity of information contained in each one of them. For this evaluation, the following variables will be considered to measure the reliability of each type of source, considering the VC has a limited budget and limited time constraints: number of startups registered, the price per month of the data source, and time required to perform the extraction, not considering scraping tools.

Additionally, taking the perspective of an early-stage VC firm, which will typically be more concerned about companies founded recently, a foundation year limit will be set on 2015, thus considering for this analysis just the companies that started operations from 2015 onwards.

	Number of Startups Retrieved	**Price/month ($)**	**Time required**
Crunchbase	325	29	Low (1)
Pitchbook	346	1250	Low (1)
Websites	663	0	High (3)

Considering the number of startups and the price per month ceteris paribus, the process of extracting startups from websites outperforms the rest. However, this methodology is significantly more time consuming when carried out manually, since the information is fragmented in 553 sources, each one being tackled to be able to identify every startup. We also need to balance the price of Crunchbase's and Pitchbook's licenses when

assessing the convenience of each data source, which depends purely on a budget of the firm.

Figure 22 depicts the information contained in the previous table, and the tradeoff between each parameter: price, time, and volume.

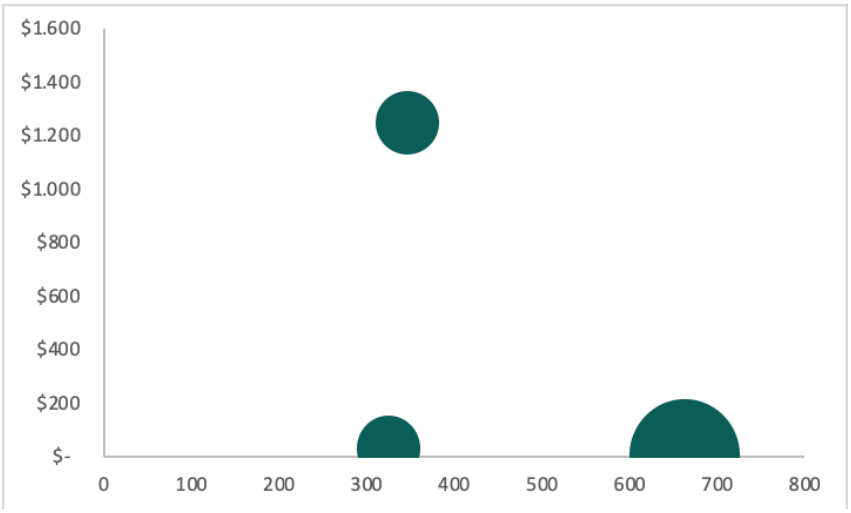

Figure 22. Diagram comparing price, volume and processing time of each online data source.

In the absence of an upper limit on the budget, which inflicts a constraint, it is useful to use both Crunchbase and Pitchbook, the latter being significantly more expensive. The information contained in Pitchbook is more relevant to perform due diligence of specific deals, due to the accuracy and quality of the data, and the fact that they provide not only data of particular companies, but also a series of additional services, such as industry or trend reports.

Additionally, extracting data from Crunchbase and Pitchbook is reduced to entering a query and downloading the results, which is substantially less time consuming than scraping websites on a one-by-one basis looking for startups.

On the other hand, considering the time needed to extract the data from different websites, there are a series of methodologies we can use to automatize this process by programming a set of periodic crawlers for each site. This issue will be the aim of the following section.

3.2. Public versus Non-public Data Sources

As seen in the previous section, the three main issues encountered with the data sets previously discussed are the following ones:

- The high price of premium databases, especially Pitchbook.
- The low number of registered startups in premium databases (i.e., Crunchbase and Pitchbook).
- The high labor-costs of aggregating data from multiple websites.

Since the premium databases have a fixed cost and they allow to download data using queries, the approach here will consist on automatizing the aggregation of data from

different websites using web crawlers, thus reducing the time constraint of supervising the set of sites.

Data scraping, or data crawling, is a software technique in which a computer program extracts data from human-readable output coming from another program.

Several methods can be used to extract the data, depending on the nature of the source of data: non-public websites, available by login-in (e.g., LinkedIn, AngelList, etc.), and public data sources, available without a login.

3.2.1. Public Data Source

This category includes websites of accelerators and incubators, venture capital firms, venture builders, and public startup databases (e.g., Startupxplore, Product Hunt, etc.). The following methodologies allow to extract data from public websites:

- Browser plugins, which need to be given the scraping structure to be used, together with the CSS selectors of the data to be extracted. Many plugins fall into this category; the majority of them are very straightforward to use. However, the ease of use is a tradeoff with flexibility and customization. A good example is Webscraper.io (Webscraper.io, 2019), which also supports periodic cloud-scraping of pre-defined websites.
- Scraping platforms, allowing to build crawlers using a programming language, usually Javascript, and retrieve the data on a timely basis, e.g., Apify (Apify, 2019).
- Programming in IDEs using open-source frameworks built for the specific purpose of scraping. One of the most relevant programming libraries to make scraping bots is Scrapy, built for Python, currently one of the most versatile programming languages.

3.2.2. Non-Public Data Sources

This challenge is encountered in social networking websites such as LinkedIn, who have the right to block an account if such abuses are reported. There have been legal cases involving companies scraping data from this type of websites, such as HiQ Labs vs. LinkedIn (M Lex Market Insight, 2018), the results of which are currently not concluded.

Since founders usually publish their company's data as soon as they start the business, the benefits of using networking and social platforms are notable. Several programming techniques can be used to automate logins and scrape the data. However, these methodologies will not be considered in this document since the legal boundaries of these activities are not yet clear.

This type of data source is more challenging than the previous one due to the following two main challenges:

- Anti-bot systems implemented to detect abuses in the use of data, annoying user behaviors, and massive requests to their servers. Usually, this issue can be offset by using IP rotation, which is a technique used to change the IP periodically and thus avoid being detected. However, in cases that need a login to access the data, the owner of the website can measure the number of interactions of an

account and thus identify abnormal behaviors, such as doing requests every five seconds for hours.
- Input and output interactions. Many built-in crawlers do not support input/output operations, referring to automatically completing inputs and using the results of queries in websites. Thus, it is required to use a programming language with a specific function to input the login information with a session cookie.

3.3. Building a Sample Set

Since scraping data from non-public data sources present legal barriers, usually adhered to in terms of service, a particular focus will be made here on public data sources and the scraping methodologies defined for this type of data source.

To connect the set of crawlers that will extract the data on a timely basis, and exemplify how a website could be considered a potentially reliable source of startup leads, a set of websites has been elaborated following the criteria specified below:

- Legality: the site does not require a login.
- A short time lag from foundation to publication: the website has to contain startups founded after 2016, reflecting early stage opportunities.
- Volume: the site has to include a minimum of 10 startups.

The result has been the set of websites described in the following table.

	Name	Type	Nº of Companies (Approx.)	Description
LANZADERA	Lanzadera	Accelerator	150	Juan Roig's (owner of Mercadona) startup accelerator, based in Valencia. They have backed more than 280 startups since 2013, and currently have agreements with multi-nationals, such as Volkswagen, Nestle or CBRE, to provide external innovation. (Lanzadera, 2019)
wayra.	Wayra	Accelerator	100	Wayra is Telefonica's start-up accelerator. They back entrepreneurs in Latin America and Spain since 2011. (Wayra, 2019)

STARTUPXPLORE	Startupxplore	Database	15.000	Community of startups and investors in Spain. Even though it is a social service, the data is publicly available. (Startupxplore, 2019)
BBOOSTER	Bbooster	Accelerator	25	Venture capital funds Manager born in 2010 as the first startup accelerator in Spain. (Bbooster, 2019)
DEMIUM startups	Demium Startups	Incubator	50	Start-up incubator specializing with offices in Valencia, Madrid, Barcelona, Bilbao and Malaga. Demium creates startups from scratch. (Demium Startups, 2019)
VA	Tetuan Valley	Incubator	50	Tetuan Valley is a non-profit incubator program to promote local entrepreneurship and regional development in the area of Madrid. (Tetuan Valley, 2019)
Google	Google Campus	Accelerator	50	Google's startup acceleration program. It involves free trainings and mentoring, and a diverse community of like-minded entrepreneurs. (Google Campus, 2019)
DATACITY - BY NUMA -	DataCity	Accelerator	150	6-9 month's program to help entrepreneurs to create and test solutions. (DataCity, 2019)
Seed Rocket	SeedRocket	Accelerator	100	Seed funding venture program for entrepreneurs with technology-based startups. (Seed Rocket, 2019)

ESADE	ESADE Ban	Accelerator	25	ESADE's startup-acceleration program. (ESADE Ban, 2019)
ie ENTREPRENEURSHIP	IE Venture Labs	Accelerator	25	IE's startup-acceleration program. (IE Venture Labs, 2019)
IMPACT GROWTH	Impact Accelerator	Accelerator	100	Delocalized acceleration program, supported by a combination of equity-free cash and VC funding. (Impact, 2019)
endeavor	Endeavor	Accelerator	50	Non-profit organization that supports high-impact entrepreneurs around the world. (Endeavor, 2019)

3.4. Crawling Approaches: Tools vs Code

There are several ways to extract structured data from websites in an automatic way using so-called bots. We will differentiate between third-party tools and programming bots with scraping libraries.

- Tools: third-party providers of scraping services. This category includes browser plugins, cloud platforms, and scraping solutions that do not involve any programming.

- Programming libraries: this category includes collections of pre-compiled routines that programs can use, and stored in object formats.

3.4.1. Tools

Tools are an easy and fast approach to do web scraping. Since there is no need to code, but to provide the structure in which to extract the data, there are many advantages to using tools. Some of the features to account for when selecting the appropriate tool are the following:

- Ease of use, referred to the level of difficulty to understand the tool's functionalities. Scale: 1 (easy), 2 (medium), and 3 (hard).
- Cloud scraping referred to the possibility of integrating cloud technologies to perform crawling of websites.
- API: possibility to manage scrapers through an API.

- IP Rotation. A. A common issue faced when scraping websites are getting blocked while scraping due to abnormal user behaviors. Several techniques can be used to rotate IP addresses with switching user agents to mimic the conduct of a regular user (ScrapeHero, 2018).
- Interactive extractors allow users to record specific actions for different situations and have them play them back automatically (Import.io, 2019).
- Price per month of the standard license.

The table below provides a comparison of some of the more popular scraping tools, with an analysis of the features defined above.

Tool	Ease of use	Cloud Scraping	API	IP Rotation	Interactive	Standard Basic Price
Web Scraper	1	Yes	Yes	Yes (5.000 credits)	No	$50/mo.
import.io	1	Yes	Yes	Yes (5.000 credits)	Yes	$299/mo.
Octoparse	2	Yes	Yes	Yes	Yes	$75/mo.
parsehub	2	Yes	Yes	Yes	Yes	$149/mo.
mozenda	2	No	No	No	Yes	$250/mo.

The use of tools instead of code has the following advantages from an organization.

- Tools do not require programming skills.
- Unlike code, they do not require maintenance.
- Tools allow to deploy crawlers in a matter very fast, and they usually do not have bugs.
- Most of the tools defined above have APIs, meaning that the crawlers can be integrated as a part of a larger chain of the process.

However, in some cases they have some limitations as defined below.

- Tools are not completely customizable, and some cases may require an additional level of complexity, such as automating logins.
- Tools are limited to the extent of its features, while by programming bots we can add any feature by integrating other technologies, such as IP rotation, periodic crawlers, connection to databases, etc.

3.4.2. Code

Several libraries can be used to build scraping bots. Some of the most common programming languages are Python and Javascript. Two of the leading libraries are the following:

- Scrapy: open source framework for extracting data from websites using Python.
- Apify: web scraping and automation platform to turn websites into APIs in a using Javascript.

The public database Startupxplore has been used to exemplify the process of building a scraping bot, made from the online platform of Apify following the process defined below:

- Filter for companies according to specific criteria, resulting in a search URL.

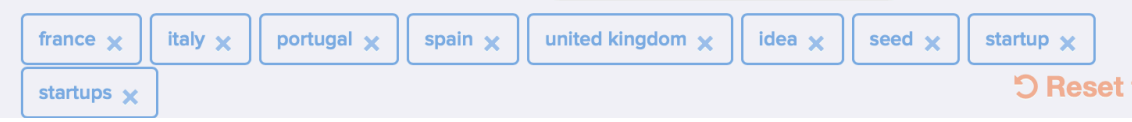

Figure 23. Example of company filters in Startupxplore.

- Login in Apify and create a crawler. Define the start URL with the one resulting from doing the query specified in the previous step. There are more configurations available, such as clickable elements or pseudo-URLs, but they are out of the scope of this example.

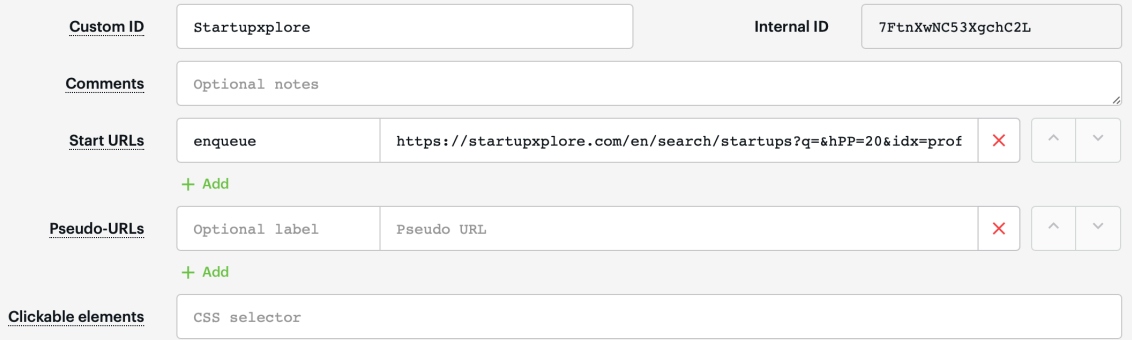

Figure 24. Apify crawler configurations.

- Code the scraping function with Javascript, as defined in the code below. The scraping bot will enter into each element of the CSS class “profile-link”, and retrieve the name of the company – “title”, a brief description of the company's business model – “elevPitch”, the website – “website”, the founder’s first name - “firstName”, and the last name – “lastName”.

```
function pageFunction(context) {
    // called on every page the crawler visits, use it to extract data from
it

// Save the context and create the results array that will return the set of
data once the bot has finished the extraction process.
    var $ = context.jQuery;
    var results = []
```

```
// Get each result of the initial query using the CSS selector that defines
the set of links, in this case “profilelink”.

    $('.profilelink').each(function() {
            context.enqueuePage({
                url: $(this).attr('href'),
                label: 'LIST'
            });
        });

// Create a function to be able to extract the set of links contained in the
pagination of the webstie.

    $('.ais-pagination--link').each(function() {
        context.enqueuePage({
            url: $(this).attr('href')
        });
    });

// Once we have collected all the result-links, extract the structured data
contained in each one of them by using their corresponding CSS selectors.

    if (context.request.label === "LIST") {
        var founder = $('.profilelink').attr('title');
        var vecFounder = [2];
        vecFounder = founder.split(" ");
        if(vecFounder[0] == null) vecFounder[0] = 'TBC'
        if(vecFounder[1] == null) vecFounder[1] = 'TBC'

// Add an array with the data extracted from each profile to the array of
output results.

        results.push({
            title: $('.content-profile-principal h1').text(),
            elevPitch: $('.about-profile em').text(),
            website: $('.website a').attr('href'),
            firstName: vecFounder[0],
            lastName: vecFounder[1]
        });
    }

// Once the process has finished, return the array of results.
    return results
}
```

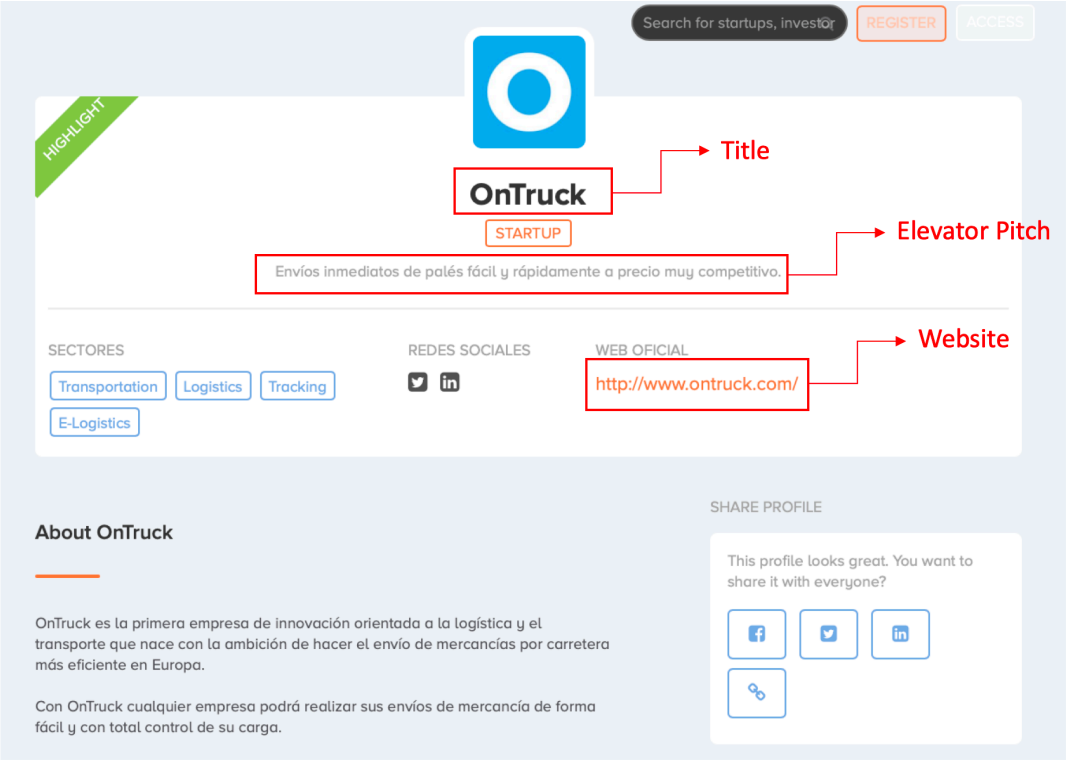

Figure 25. Startupxplore's example of company layout and crawler output (I).

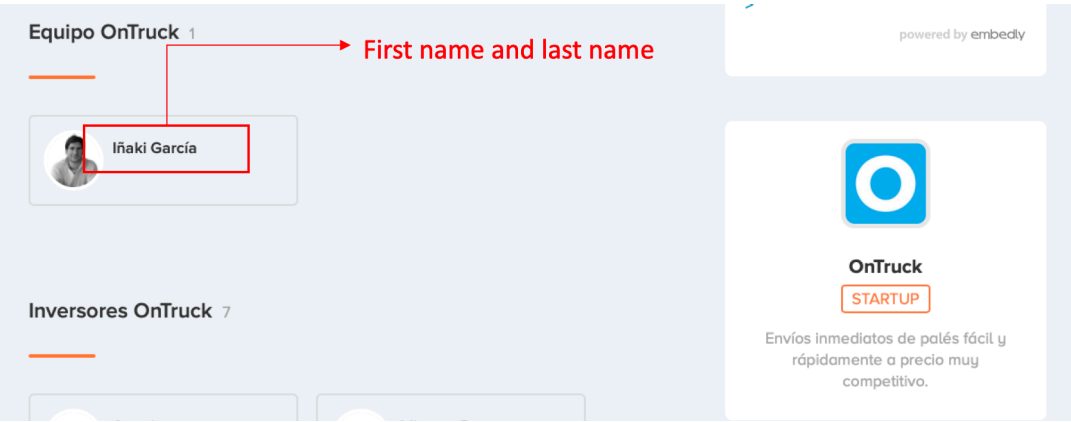

Figure 26. Startupxplore's example of company layout and crawler output (II)

3.5. Data Collection

Once extracted the data, it is possible to additionally program a periodic schedule for the bot's execution and determine how to receive the set of data and the file's format.

In the case of this research, the option employed has been a set of periodic crawlers to send the resulting data in a comma-separated values' format by email. This CSV file will then be cleaned and classified according to its business model before being imported into the CRM.

3.6. Data Enrichment

Any of the extraction processes defined in the previous sections can generate a large volume of structured data. However, it is critical to be able to enrich the information extracted to get further insights. For this purpose, the following two approaches can be used:

- Use of third parties' data enrichment tools, such as Clearbit (Clearbit, 2019), which provides data enrichment for B2B marketing and sales teams. Clearbit enriches the leads with information from multiple sources and adds new layers of intelligence to the data extracted, such as emails, phone numbers, locations, etc.
- Cross-checking of data records extracted from multiple sources by the company's or founder's name: for example, if data from a company has been obtained from source A, it can be cross-checked with source B to see if it contains additional information.

3.7. General Overview on Chapter 3

This chapter has reviewed and evaluated the different types of online data sources of startup leads, together with the methodologies available to build a lean and scalable data extraction process using bots.

The extraction processes defined in previous sections generate a large volume of data. The next chapter will study different machine learning techniques that will automatically filter the data that retrieved online according to a given set of enterprise data tagged by the attractiveness of a company's business model.

Chapter 4 – Development of an NLP Model to Classify Startup Leads

4.1. Introduction to Artificial Intelligence, Machine Learning, and Deep Learning

Artificial Intelligence (AI) is the umbrella of different approaches to data analytics, such as Machine Learning models, or Deep Learning Networks. The term was first coined by John McCarthy in his 1955 proposal for the Dartmouth Conference, considered the first conference on artificial intelligence, the aim of which was to explore alternatives for making a machine that could reason like a human, and could abstract thought, self-improvement, and problem-solving. Figure 27 provides a picture of John McCarthy, considered the father of AI.

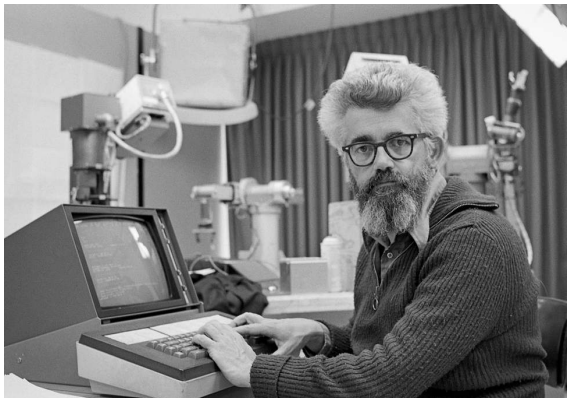

Figure 27. John McCarthy: Computer scientist known as the father of AI.

Advances in affordable high computing power and big-data availability made artificial intelligence thrive in the 1980s. Machine learning appeared as a subset of artificial intelligence to give computers the ability to learn automatically from patterns or features in large data sets. Meanwhile, deep learning appeared in the 2010s as a subset of machine learning to enable computers to solve more complex problems by making the computation of multi-layer neural networks feasible (Genç, 2019).

The popularity milestone for artificial intelligence came with AlphaGo, an artificial intelligence program developed by Google DeepMind to play the ancient board game Go. The program, using a machine learning algorithm called reinforcement learning, and incorporating deep-learning, won in March 2016 in a five-game match to Lee Sedol, a professional Go player. Ever since, AI became a daily topic in the tech news: self-driving cars, Amazon's Alexa, Apple's Siri, Netflix recommendation algorithms, and so on.

Figure 28 depicts the evolution of the different branches of artificial intelligence from the 1950s until today (Jeffcock, 2018).

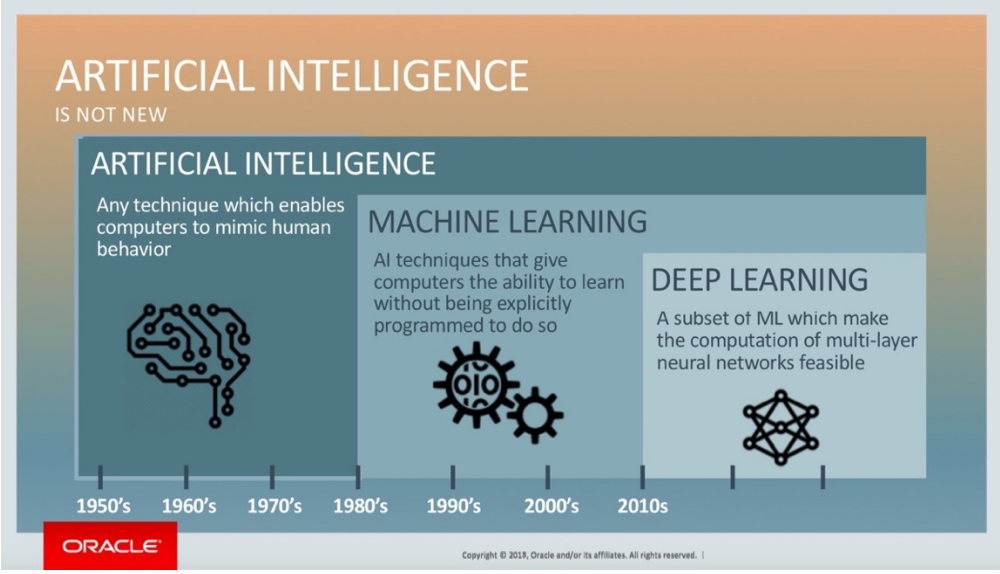

Figure 28. Evolution timeline of AI, machine learning and deep learning.

The spectrum of problems that businesses face is vast, and the machine learning alternatives to solve them are very wide as well. The following table lists few real-life applications of machine learning.

Application Field	Example
Agriculture	Precision agriculture (PA): the use of technology for the treatment of soil and by-demand plant growth. Instead of treating each field uniformly as in conventional farming, using precision agriculture, crops can be sprayed and fertilized based on variable rates according to specific requirements of the site.
Brain-machine interfaces	Brain-computer interface (BCI): direct bidirectional communication interface between a brain and an external device.
Computer networks	

	Network simulations: simulation technique to emulate the behavior of a network by calculating interactions between its different entities (nodes, routers, access points, etc.).
Computer vision	Autonomous planning of robots to navigate in unknown environments.
Economics	Agent-based computational economics (ACE): study of economic processes and dynamics using interacting agents.

4.2. Approach for Building the NLP Model

As seen in previous chapters, in sales terminology, a lead is a person or business who may eventually become a client of a given company. In the case of a VC firm, it is useful to use the term lead to target prospective startups in which to invest. Even though the VC firm invests in them, startups are taken care as clients for their eventual return. Besides, investing in the best startups implies nowadays having a sales-oriented investment process.

Today, sales teams need to manage larger and larger volumes of leads, classify them, and then target the ones that meet specific criteria. According to HubSpot's (Hubspot, 2014), outbound marketing generated in 2014 twice as many leads compared to inbound marketing.

Similarly, even though inbound sourcing is a crucial source of leads in venture capital, outbound sourcing is a much more scalable source of prospects. Since the inbound sourcing process is very time consuming per unit of output for the investment team, which must be actively involved in the process of finding new startups, defining an outbound sourcing process to find prospects is vital to ensure the business scalability of the VC firm, who usually has fewer than 15 employees (Vault, 2018).

In the previous chapter, we defined a scalable process to extract leads from different online data sources. However, the large volume of data that the latter generates makes it appropriate to establish an intermediate layer to qualify the leads automatically. For this purpose, this chapter will include several machine learning methodologies to classify the data, with a particular focus on natural language processing algorithms, or NLP models: Deep Learning and Naive Bayes.

This chapter will also contain an analysis of ensemble techniques, based on the fact that by diversifying on a set of different models, to unify the results given by the various models, and make better predictions.

Stacking is an ensemble model technique used to combine information from multiple predictive models to generate a new model. The stacking model will use predictions from a naïve Bayes and a Deep Learning model (base models) to build a new one based on a linear regression of both (stacked model), to make predictions on the test set.

Often, the stacked model will outperform each of the individual models due to its soothing nature and ability to highlight each base model where it performs best and discredit each base model where it performs poorly.

4.3. NLP Models for Text Categorization

The information extracted from online data sources is, in most cases, limited to just a company's name, a founder's name, a website, and a short description of the business operations. Since we need the methodology to be consistent with very different online data sources, structured in variable ways, and the least common multiple is a description of the business, we will focus on adding a Natural Language Processing (NLP) layer to add intelligence to the process in an automatic way. In the process of building NLP models, we will use a set of data from a VC's CRM for training and testing.

Adding NLP layers to subtract information from large volumes of data automatically is widespread use of machine learning in many different industries, from online advertising up to trading the markets. To illustrate how an intermediate NLP layer can be useful to automatize repetitive tasks, figure 29 depicts the working example of a chatbot (Gill, 2019).

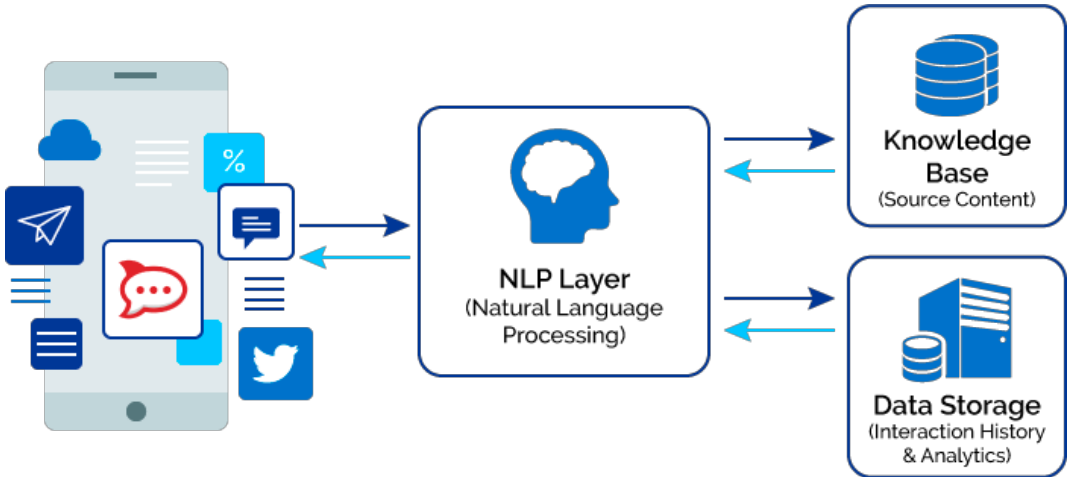

Figure 29. Working of a Chatbot.

In the example depicted in figure 29, when the chatbot is asked a question, it will respond based on a knowledge base, or source of content, that is available to it. In conversations where the chatbot is not capable of understanding a question based on the knowledge base, it will redirect the conversation to a human operator or switch the answer. However, it will always learn from the interaction, growing its scope, and gaining relevance.

NLP models can be applied in many ways using various combinations of technique. A set of text categorization techniques will be studied to qualify startup leads, and convert DQLs to IALs. These techniques will use a CRM's data set consisting of business descriptions previously tagged manually by a member of the investment team as attractive (or fit with investment criteria), or not. Text categorization is a very typical use of supervised machine learning and allows developers to save time by automatically tagging lengthy texts into pre-defined categories.

Figure 30 exemplifies how text categorization by topic works. The general principle is that a set of articles passes through an NLP layer which, according to the words contained in it, its frequency, and training set with a collection of articles already tagged, classifies them into a pre-defined set of categories.

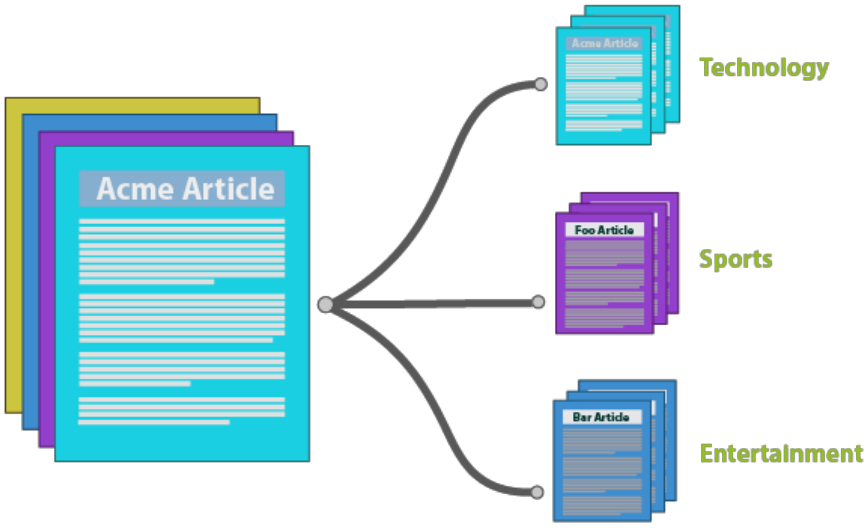

Figure 30. NLP for text classification by topic.

4.4. Bag of Words (BoW)

Most machine learning algorithms rely on numerical data as input values, but business descriptions are usually dense, relenting the computation. This section will introduce the Bag of Words (BoW) concept, which is a term used to specify the problems that have a "bag of words" or a collection of text data that needs to be processed. The idea is to take a piece of text and count the frequency of the words in that text.

It is essential to consider that the BoW concept treats each word individually and the order in which the words occur in the sentence does not matter; thus losing information on specific sequences of words that may have a particular meaning together. A set of descriptions or documents is converted into a matrix, with each separate text being a row, and each word (token) being a column. The corresponding pair (row, column) values will be the frequency of occurrence of each word or token in the text.

As an example, consider the following four text messages:

["Hello, how are you," "Win money, win from home," "Call me now," "Hello, call you tomorrow?"]

The frequency distribution matrix depicted in figure 31 (Udacity, 2018) results from the set of sentences with quotation marks defined in the table above.

	are	call	from	hello	home	how	me	money	now	tomorrow	win	you
0	1	0	0	1	0	1	0	0	0	0	0	1
1	0	0	1	0	1	0	0	1	0	0	2	0
2	0	1	0	0	0	0	1	0	1	0	0	0
3	0	1	0	1	0	0	0	0	0	1	0	1

Figure 31. Example of Bag of Words (BoW).

As seen in figure 31, sentences are numbered in the rows, and each word is a column name, with the corresponding pair value being the frequency of that word in the document.

Using Python, a programmed way to handle the bag of words problem is by using the Sklearn "CountVectorizer" method (Scikit-learn, 2019), which does the following:

- It tokenizes the string (separates the string into individual words), and gives an integer ID to each token.
- It counts the frequency of occurrence of the tokens.

4.5. Naïve Bayes

Bayes theorem is one of the first probabilistic inference algorithms in history. It was developed by Reverend Bayes, who tried to use it to infer the existence of God (Dan Kopf, QUARTZ, 2018). It still performs very well in some instances. In short, the Bayes theorem calculates the probability of a particular event happening based on the probabilistic joint distributions of other events (for example, the appearance of certain words in a text).

Naive Bayes is an NLP text classification algorithm based on the concept of conditional probability. This algorithm has excellent benefits, such as being easy to implement, and very fast to train.

Some definitions that we need to consider when using the Bayes theorem are the following.

- Prior: the initial guess, since it is all we could infer before the arrival of new information.
- Posterior: the second guess, inferred after further information has arrived.

This model is called naive Bayes because the computation of the probabilities for each hypothesis is simplified. Thus, instead of calculating the values of each attribute value, they are assumed to be conditionally independent, given the target value.

$$P(d_1, d_2, d_3, \dots \mid h) = P(d_1 \mid h) \cdot P(d_2 \mid h) \cdot P(d_3 \mid h) \cdot P(\dots \mid h)$$

Of course, this phenomenon is improbable to be observed in real data, since words are generally dependent on one another. However, this approach performs very well on data where this assumption does not hold.

The Naïve Bayes method can be summarized in the following three steps (Synced, 2017).

1. Feature Engineering. The first step will be to extract features from the text, and since the model needs numerical features as input for the classifiers, the solution is to use the bag of words (BoW) to count word frequencies. Then, the probability will be calculated using word frequencies through the Bayes' theorem, defined below:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

As an example, let's consider the simplified business description: "Online platform to connect drivers." The probability that the business model associated with the description is a marketplace can be calculated as follows:

$$P(marketplace|"Online\ platform\ to\ connect\ drivers") = \frac{P("Online\ platform\ to\ connect\ drivers"|marketplace) \cdot P(marketplace)}{P("Online\ platform\ to\ connect\ drivers")}$$

However, to obtain $P("Online\ platform\ to\ connect\ drivers"|marketplace)$, it will be necessary to count the occurrence of "Online platform to connect drivers" in the "Marketplace" category, and since it may not appear as such in the training set at all, the probability will be zero, consequently making $P("Online\ platform\ to\ connect\ drivers"|marketplace)$ zero as well.

2. Being Naïve. Naïve Bayes model assumes that every word is independent of one another. Therefore, it will look at individual words in the training set instead of the entire sentence. Thus, the probability of the previous sentence will be calculated as follows:

$$P("Online\ platform\ to\ connect\ drivers") = P(Online) \cdot P(platform) \cdot P(to) \cdot P(connect) \cdot P(drivers)$$

$$P("Online\ platform\ to\ connect\ drivers"\ |\ marketplace) = P(Online|marketplace) \cdot P(platform|marketplace) \cdot P(to|marketplace) \cdot P(connect|marketplace) \cdot P(drivers|marketplace)$$

3. Calculating the probabilities: the model will compare the probabilities associated with each category and determine which one is higher. Since this case will consist on a binary categorization, a probability above 0.5 for a positive business model will convert a startup lead from DQL to IAL.

   It is worth noting that it may be the case that the word "drivers" does not exist in the category $marketplace$, thus $P(drivers\ |\ marketplace) = 0$, leading $P(Online\ platform\ to\ connect\ drivers|marketplace) = 0$.

   Since this issue will wipe out all the information of the other probabilities, the technique Laplace smoothing will be used.

4. Laplace smoothing is a technique that is employed to smooth categorical data that implements a pseudo-count in every probability estimate. Therefore, no probability will be zero.

4.6. Deep Learning and Convolutional Neural Networks

Deep learning is a sub-set of machine learning that mirrors the functioning of the human brain, which uses a distributed approach to problem-solving. Similar to the structure of the neurons in a brain, a deep learning model is organized in consecutive layers, each layer receiving information from a previous layer, and passing information downstream to the next one.

Deep learning models learn multiple levels of representation, which is one of its main advantages. The information is then constructed level-by-level through composition, and the lower level of representation can be shared across tasks.

One of common use of deep learning is for image classification. Figure 32 provides an example of how a deep learning neural network works (PNAS, 2019).

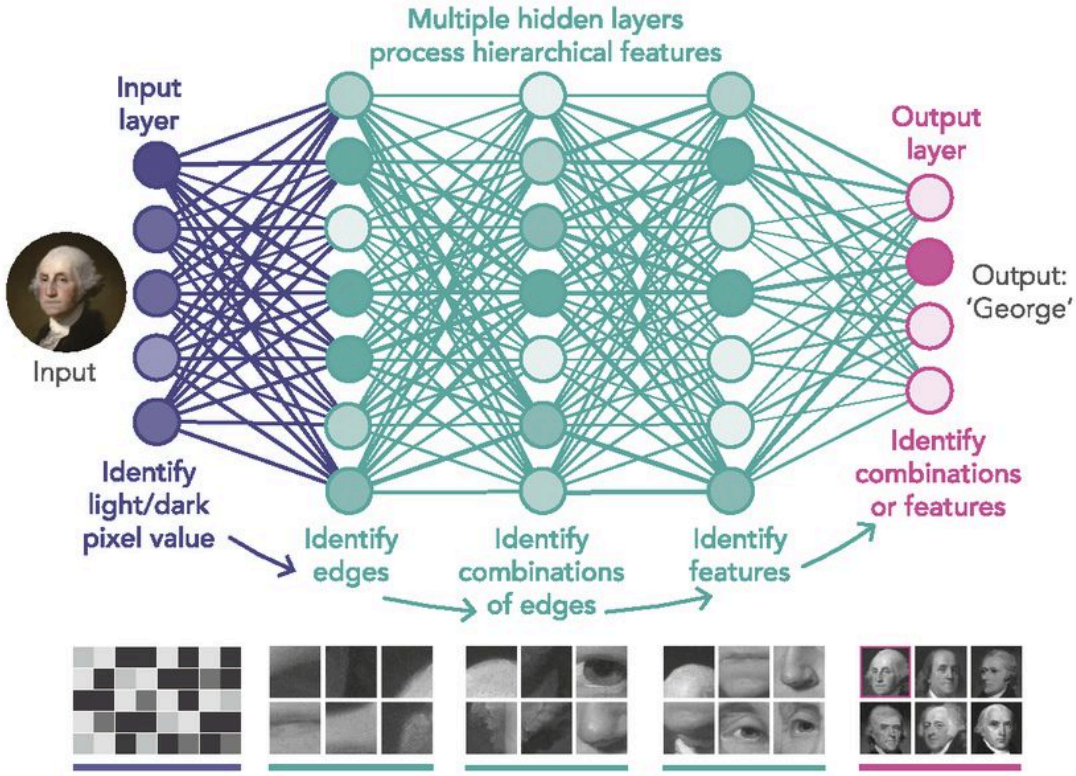

Figure 32. Deep Neural Network for Face Recognition.

As seen in figure 32, the neural network starts identifying simple patterns in the image, such as edges, then combinations of edges and finally, sophisticated features in the picture. Though the example of image classification is different from the one of classifying text, the process is quite similar.

To classify text, a class of deep learning model called convolutional neural network, CNN's will be used. CNN's are a regularized version of multilayer perceptrons, defined as fully connected networks in which each node is connected to each in the next layer.

Weston and Collobert (Saravia, 2018) were among the first researchers to use a CNN based framework in NLP tasks. Their method consisted of transforming words into vector representations via lookup tables, resulting in a word embedding that learned weights during the training of the network. The process involved in the classification task of a sentence will be the following:

1. Sentences are first tokenized into words, which will then be transformed into a word embedding matrix that will constitute the input layer.

2. Convolutional filters consisting of filters of all possible sizes of windows will then be implemented on the input layer to produce a feature map.

3. The feature map will then be followed by a max-pooling operation, applying a max operation on each filter to obtain a fixed length of the output, thus reducing the dimensionality of the output and amount of computation in the network.

4. This procedure will then classify the sentence into a pre-defined set of categories.

Figure 33 provides a general representation of a convolutional neural network applied for text classification (WILDML, 2015)

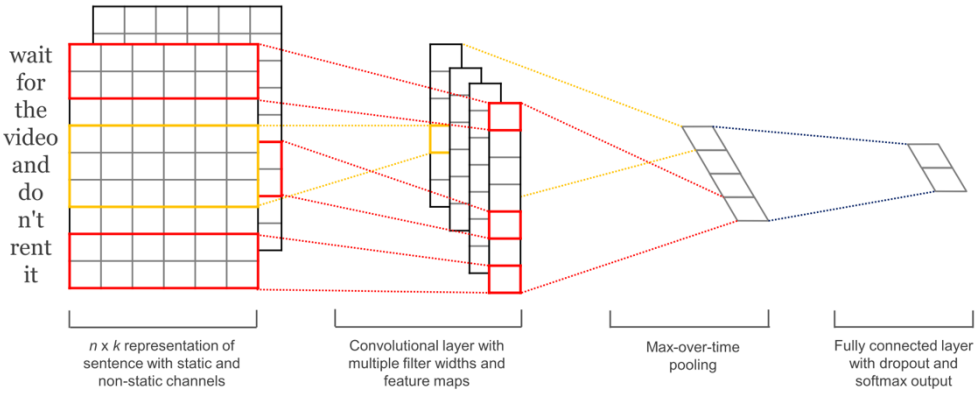

Figure 33. Model Architecture with two channels for an example sentence.

4.7. Stacking Methods

Ensemble methods are useful to improve predictive performance in machine learning problems. Stacking is an ensemble technique that is used to combine predictions from a set of models trained on a dataset. The output predictions from the collection of models are combined to train a new model, defined as meta-learner.

The following figure (DeFilippi, 2018) depicts the structure of a stacking method that combines multiple learners' predictions to generate a single final forecast.

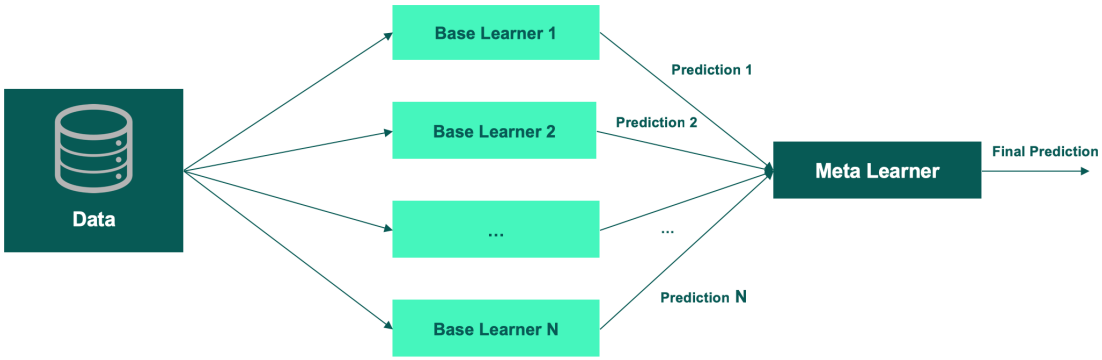

Figure 34. Functional representation of a stacking method.

Simple linear models can be used to combine predictions from sub-models to a weighted sum using logistic regression or linear regression. Once we have trained and fitted both base learners, we will mix their probability predictions through the ensemble method of "stacking," for which we will use a linear stacking regression.

A stacking regression is a method to form linear combinations of a set of predictors employed to give improved prediction accuracy. The idea behind is to use cross-validation data and least squares to determine the coefficients in the combination (Leo Breiman, UC Berkeley, 1996). The general representation of linear regression, with the array of independent variables x_i^T, the array of regression coefficients β, and the error term ε_i, for a given observation y_i, is given as follows:

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x_i^T}\beta + \varepsilon_i, \qquad i = 1, \ldots, n$$

$$\mathbf{y} = \mathrm{X}\beta + \boldsymbol{\varepsilon}$$

Linear regressions are widely used in machine learning. They model the relationship between a dependent variable Y, and one or more independent variables X, using the best-fit regression hyperplane, i.e., the one that minimizes the mean squared error, given by the following formula.

$$MSE = \frac{1}{n}\sum_{i-1}^{n}(y_i - \widehat{y_i})^2$$

For the case of study of this research, the data will consist of startups' business descriptions, retrieved from online data sources using scraping techniques. Then, the descriptions will be vectorized using the bag of words methodology, the output of which will be nurtured into different base learners. Each base learner which will give a prediction, consisting of a probability that the business model is similar to that of companies previously analyzed by the investment team, and stored in a CRM. Finally, the predictions of each base learner will be combined using a linear regression model, which will provide a final forecast, determining whether the company is converted from DQL to IAL or not.

4.8. Metrics for Evaluating the Performance of the Model

To evaluate the performance of our model, we will use different metrics to determine how good it classifies the business descriptions. For this purpose, we will use confusion matrices. A confusion matrix describes the performance of a model used for binary classification problems. It provides a whole set of metrics to be analyzed. Figure 35 depicts a confusion matrix with a set of metrics associated with each measure.

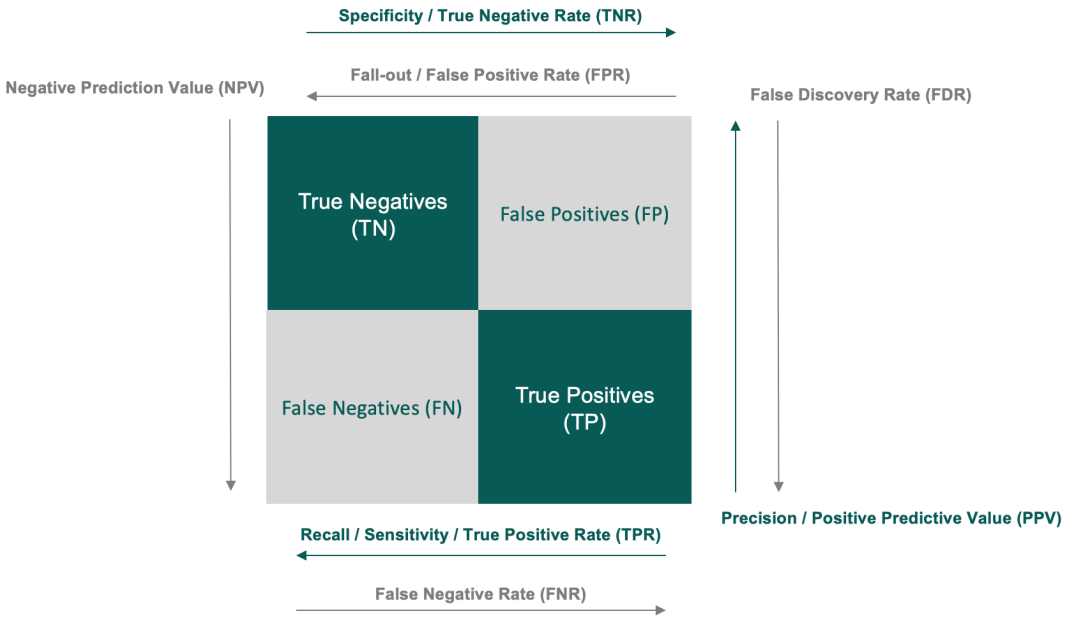

Figure 35. Representation of a Confusion Matrix (Kapoor, 2017).

Each array in the visualization above defines the name of a metric: precision and recall. Precision measures the percent of true positives out of all the true and false positives, and recall measures the percentage of true positives out of all true positives and false negatives. On the other hand, accuracy measures the rate of correctly classified points out of the total points.

$$Precision = \frac{True\ positives}{True\ positives + False\ positives}$$

$$Recall = \frac{True\ positives}{True\ positives + False\ Negatives}$$

$$Accuracy = \frac{Correctly\ classifed\ points}{Total\ points}$$

To simplify, a combination of both precision and recall based on a weighted average of both will be used. The arithmetic mean could have been taken, but it would not be any different from the accuracy. Thus, the harmonic mean will be used, which is always lower than the arithmetic average, so it is closer to the smaller number than to the higher number, providing a more conservative metric. The harmonic mean is calculated as follows:

$$H = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \cdots + \frac{1}{x_n}} = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}}$$

Thus, the F1 Score is defined as follows.

$$F1\ Score = 2\ \frac{Precision * Recall}{Precision + Recall}$$

It is worth noting that the F1 score is more useful than accuracy to measure performance in uneven class distributions. Accuracy will work best when false positives and false negatives have a similar cost. In our case, false negatives will have a much higher cost than false positives, since losing a prospective investment due to a misclassification of

the model is much worse than manually reviewing a company that does not meet the investment criteria of the VC firm. Thus, we will use both accuracy and F1, giving a larger weight to the latter when evaluating the performance of each model.

4.9. Implementation in Pyhton through a Jupyter Notebook

4.9.1. Exploratory Visualization of the Training Set

The data set contains 9837 total rows. From these rows, 42.3% correspond to prospective portfolio companies. We have split the description column into words, then we have removed “stopwords”, or words that are common in everyday English but do not provide relevant meaning to the text, such as “the” or “and,” and counted the frequency of the words obtained.

Figure 36 depicts the frequency distribution per word for the most 1,000 frequent words in the set.

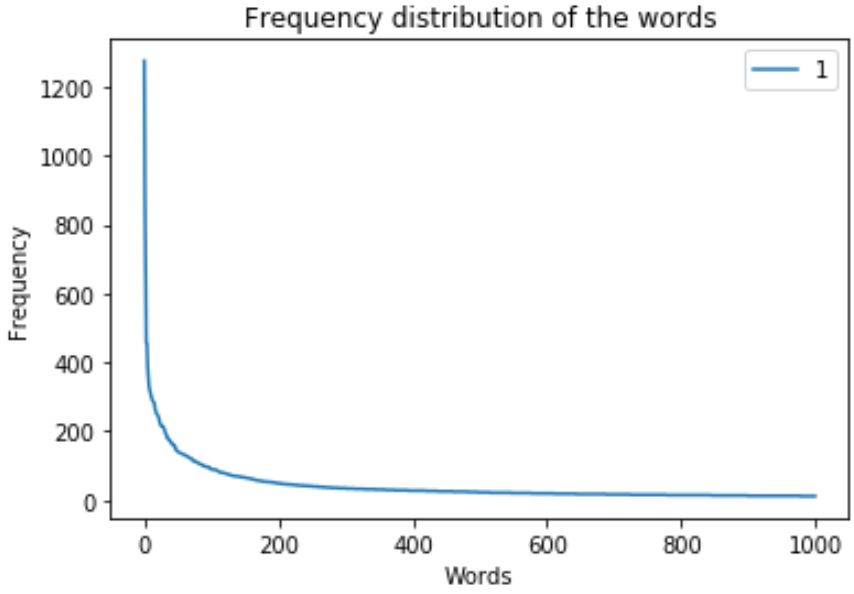

Figure 36. Frequency distribution of the words in the data set.

As seen in figure 36, a little number of words is persistent in the descriptions, while the majority of the words appear in the set a few times.

The top 10 most repeated words are given in figure 37.

	Word	Frequency
0	platform	1275
1	online	863
2	marketplace	678
3	company's	456
4	company	454
5	mobile	387
6	provider	357
7	users	331
8	application	321
9	designed	312

Figure 37. Most repeated words in the training set.

As seen in the table above, words such as marketplace (a word explicitly describing a target company) will rank among the most repeated ones. The presence of representative words will help the models make better predictions since they will help identify more easily specific business models. For example, a business description containing the word “marketplace” will probably be given a higher probability of being a prospective portfolio company.

4.9.2. Methodology and Implementation of the Models

The following section provides the implementation of both base learners and the stacking regression model, using a set of commonly used machine learning libraries in Python, and a Jupyter Notebook.

- Import Libraries and Training Set

Here, the set of libraries that will be used onwards in the implementation is imported into the notebook, together with an excel file containing the data set that will be used for training and testing the models. The data set will include a set of descriptions with a tag, specifying whether the corresponding business model is a fit with the investment criteria. The label will be "1" if the business model related to the description is marketable by the VC and "0" if it is not.

```
# Import libraries

import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
import pickle
from keras.preprocessing.text import Tokenizer
from keras.models import Sequential
from keras.layers import Activation, Dense, Dropout
from sklearn.preprocessing import LabelBinarizer
import sklearn.datasets as skds
from pathlib import Path
from keras.utils import to_categorical
```

```
# Import the training set into a pandas dataframe
df = pd.read_excel('Descriptions_Training_Updated.xlsx')

# Visualize descriptive statistics about the dataframe
df.describe()
```

The previous code will return a few descriptive statistics about the data set, which are given in figure 38.

	Model
count	9837.000000
mean	0.422893
std	0.494044

Figure 38. Descriptive Statistics of the Training Set employed to build the NLP models.

As seen in figure 38, the data set contains 9837 business descriptions, 42.3% of which are a fit with the investment criteria of the VC firm. It is worth recalling that members of the investment team have manually tagged the business descriptions of the training set.

- Data Preprocessing

Since the base learners only work with numbers instead of words, the data set will need to be processed into a bag of words. For this purpose, it will use the "Count Vectorizer" method, whose implementation procedure is depicted as follows.

```
# Divide the data set into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(df['Description'],
                                                    df['Model'],
                                                    random_state=1)

# Instantiate the CountVectorizer method
count_vector = CountVectorizer()

# Fit the training data and then return the matrix
training_data = count_vector.fit_transform(X_train)

# Transform testing data and return the matrix. Note we are not fitting the testing dat
testing_data = count_vector.transform(X_test)
```

- Base Learner 1: Naïve Bayes

Once we have loaded the training data into the variable 'training_data' and the testing data into the variable 'testing_data,' the MultinomialNB classifier will be instantiated and fit the training data into the classifier using 'naive_bayes.' The implementation of this model is very straightforward since the model architecture is already provided in Sklearn.

```
naive_bayes = MultinomialNB()

naive_bayes.fit(training_data, y_train)

MultinomialNB(alpha=1.0, class_prior=None, fit_prior=True)
```

Then, it will make probability predictions on the testing data, and store them in a variable named pred_Naives_p. These predictions will be used in the stacking regression together with the probability predictions of the convolutional neural network.

```
pred_Naives = naive_bayes.predict(testing_data)
```

- Base Learner 2: Convolutional Neural Network

Keras Sequential model API (Keras, 2019) lets to define deep learning models easily. It provides a straightforward configuration for the shape of the input data and the type of layers that make up the model. The model below has been built after several trails with vocab size, epochs, and dropout layers.

```
model = Sequential()
model.add(Dense(512, input_shape=(16307,)))
model.add(Activation('relu'))
model.add(Dropout(0.3))
model.add(Dense(512))
model.add(Activation('relu'))
model.add(Dropout(0.3))
model.add(Dense(2))
model.add(Activation('softmax'))
model.summary()

model.compile(loss='categorical_crossentropy',
          optimizer='adam',
          metrics=['accuracy'])

model.fit(training_data, to_categorical(y_train),
                batch_size=100,
                epochs=30,
                verbose=1,
                validation_split=0.1)
```

Similar to the base learner 1, base learner 2 will make probability predictions on the testing set and store them in an array named “predDN_p.”

```
predDN_p = model.predict_proba(testing_data)[:,1]
```

- Meta Learner: Stacking Regression

It is imperative that the base learners produce uncorrelated predictions. Stacking works best when the forecasts that are combined are all skillful, but skillful in different ways, which may be achieved by using algorithms that use very different internal representations or models trained on different representations or projections of the training data.

The code below provides the implementation of the stacking regression model, built from the probability predictions of both Naïve Bayes' model and CNN's model.

```
X_ens = pd.DataFrame(pred_Naives_p, predDN_p)
y_ens = pd.DataFrame(y_test)

X_train_ens, X_test_ens, y_train_ens, y_test_ens = train_test_split(X_ens, y_ens, random_state=1)

ens_model = LinearRegression()
ens_model.fit(X_train_ens,y_train_ens)
```

The resulting model named ens_model will be able to make combined predictions from both base learners. The following section will assess the performance of the models.

4.9.3. Models' Evaluation and Validation

The performance of the base learners and the meta learners, measured by the F1 score and the accuracy, is provided in the following table.

	F1 Score	**Accuracy**
Base Learner 1	0.840	0.805
Base Learner 2	0.826	0.794
Meta Learner	0.859	0.822

The performance of the meta-learner is reasonable, and it aligns with the solution expectation. There is an improvement of a few percentage points in both the accuracy and the F1 score, compared to the base learners.

On the other hand, it can be concluded that the final model is more robust than the base learners alone, since it relies on two different algorithms for its predictions, and it is less prone to fall in the errors of a single algorithm. Besides, this leads the model to generalize better to unseen data, compared to a unique model.

4.10. General Overview of Chapter 4

The workflow employed is defined below:

1. Data preprocessing: the dataset has been preprocessed by getting some metrics and trying to get a better understanding of the probability distribution of the set. On the other hand, since the models need numbers as inputs instead of words, we have converted the business descriptions into a bag of words.

2. Naive-Bayes implementation: implementation of the multinomial Naive Bayes in Sklearn. This particular classifier has proven suitable for classification due to its simplicity and its high performance on the testing set.
3. CNN model implementation: implementation of a convolutional neural network as the second base learner. The Keras sequential model has been used to specify the layers in the model. Here, different model architectures have been tested iteratively to obtain the model with the best performance.
4. Stacking Model Implementation: the raw probabilities of the base learners' predictions have been passed as inputs into a stacking regression model.

Finally, the overall performance of the models was reasonably good, and the results obtained with the final stacking linear regression model improved a few points the F1 score and the accuracy.

As a potential improvement, we could have added new models to add valuable insights to the descriptions of the dataset — for instance, recurrent neural networks, which have proven useful in several NLP tasks. The idea behind RNNs is to make use of sequential information. In a traditional neural network, we assume that all inputs (and outputs) are independent of each other. However, for many tasks, that does not work, since in cases in which we want to predict the next word in a sentence, it is better first to know which words came before it.

On the other hand, it is worth noting that a good description of the business operations is fundamental to obtain reasonably good predictions. Also, descriptions that vary too much with those of the training set will present issues in the classification. Therefore, it is essential to use a training set with descriptions that are similar to the ones to be extracted from the online data sources.

Chapter 5 – Conclusions

5.1. General Overview of the Process

This document aimed to provide an alternative to inbound sourcing of startups in venture capital. For this purpose, a standard outbound sourcing process has been proposed, similar in structure and terminology to the one employed by marketing and sales teams, and supported by an NLP layer to provide intelligence to the broad set of data that the process generates.

A general overview of the entire process is depicted in figure 39.

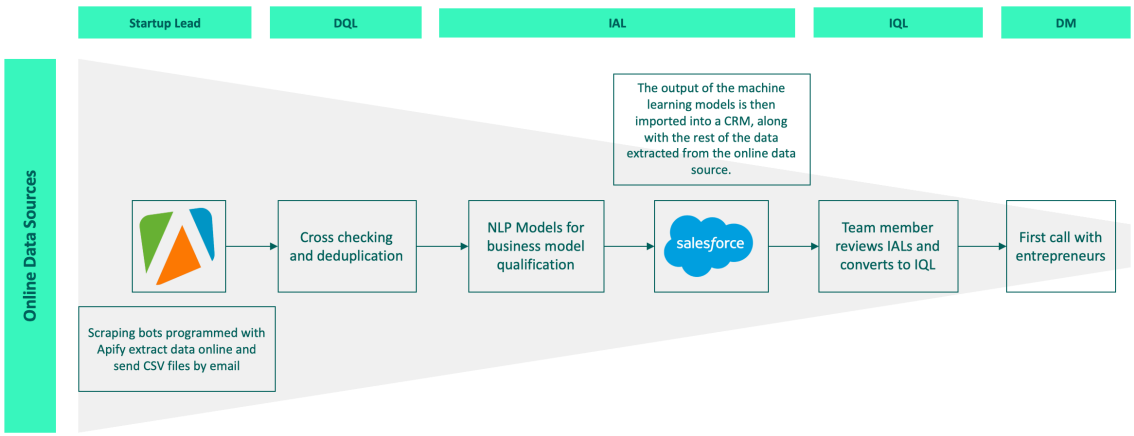

Figure 39. Overview of the outbound sourcing process.

The process has empirically yielded the conversion rates from stage to stage depicted in the funnel of figure 40.

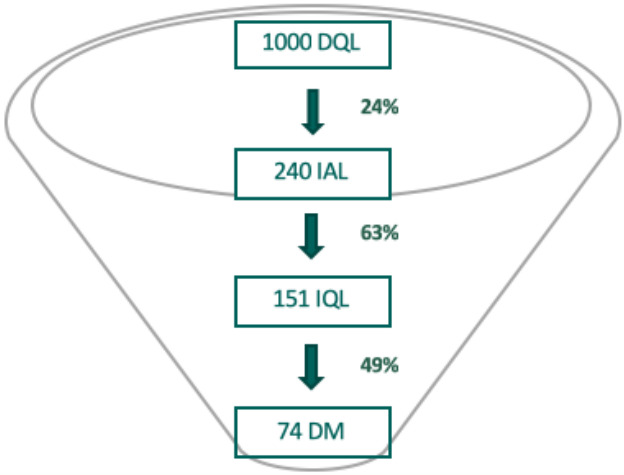

Figure 40. Conversion rates of the outbound sourcing processs.

As it can be seen in the figure above, from a set of 1,000 deduplicated DQLs, the NLP model trained with data from the CRM converts a 24% into IALs, which are then manually reviewed by a member of the team to determine whether there is effectively a fit with the investment criteria or not. From the set of IALs, 63% will be true positives and therefore will be converted into IQLs. 49% of the IQLs will be considered attractive enough by a member of the investment team and will be contacted to schedule a 30 minutes meeting with the founders. Thus, 7.4% of all the deduplicated leads retrieved from online data sources will be contacted to schedule a deal memo.

The numbers given by the conversion rates above are attractive from the perspective of a VC firm. The process relies on automatic techniques to retrieve prospective startups from online sources and on NLP models to filter the ones that match the investment thesis of the firm, saving time to the traditionally limited numbered team of a VC, and widening what is considered its bottleneck.

5.2. Advantages and Limitations

5.2.1. Advantages

Among the most relevant advantages of the standardized sourcing process defined in this paper are the following:

- The method provides a scalable way to find prospective portfolio companies in new geographies.
- It allows managing resources (capital and labor) more efficiently by providing a way to widen or shorten the bottleneck of the investment process.
- It allows the VC to visualize the status of the investment funnel at each stage of the process, giving room to lean analytics methodologies.
- It allows the investment team to concentrate on doing the due diligence of prospective startups.

5.2.2. Limitations

Some of the limitations encountered by the proposed outbound sourcing process are the following:

- By relying on machine learning algorithms to filter the extracted data, there is room to false negatives, referring to companies that are prospective portfolio companies but that are not converted into IALs, and therefore are not reviewed by any member of the VC firm, translating in missing opportunities. For this reason, it is necessary to do periodic tests of false negatives and false positives to evaluate the performance of the NLP model, and determine solutions, such as adjusting its hyper-parameters, changing its structure, or adding new base learners to the stacking regression model.
- Though the retrieval and the classification steps in the sourcing process are considered automatic, it is still necessary to program the scraping bots for each online data source and build the machine learning algorithms and make the appropriate adjustments. Besides, these steps are not connected between each other, so the intermediate layers must be manually processed.

5.3. Final Comments

The sourcing process defined in this paper is currently being used as a beta by Samaipata, a pan-European VC firm based in Madrid, to support its outbound of finding and targeting prospective startups for its portfolio.

It is worth noting that any electronic sourcing in venture capital is currently being considered state of the art, since very few firms are known to be using automatic techniques to screen prospective portfolio companies, and almost all of them rely purely on inbound sourcing.

Bibliography

Apify. (2019, 3). *Apify.com.* Retrieved from https://apify.com/

Asociación Española de Capital, Crecimiento e Inversión (ASCRI). (2018). Retrieved from Informe de actividad Venture Capital & Private Equity en España: https://www.ascri.org/wp-content/uploads/2017/05/Informe-ASCRI-2018-1.pdf

Atomico. (2018). *The State of European Tech*. Retrieved from https://2018.stateofeuropeantech.com/chapter/tech-european-economy/article/tech-motor-gdp-growth/

Bbooster. (2019). *Bbooster*. Retrieved from http://www.bbooster.org/en/

Childs, M. (2011). *Independent*. Retrieved from John McCarthy: Computer scientist known as the father of AI: https://www.independent.co.uk/news/obituaries/john-mccarthy-computer-scientist-known-as-the-father-of-ai-6255307.html

Clearbit. (2019). *Clearbit*. Retrieved from https://clearbit.com

Crunchbase. (2019). San Francisco, USA.

Crunchbase. (2019, January). Retrieved from Crunchbase Support: https://support.crunchbase.com/hc/en-us/articles/115010477187-Crunchbase-Rank-CB-Rank-

Dan Kopf, QUARTZ. (2018, June). *The most important formula in data science was first used to prove the existence of God*. Retrieved from https://qz.com/1315731/the-most-important-formula-in-data-science-was-first-used-to-prove-the-existence-of-god/

DataCity. (2019). *DataCity*. Retrieved from https://www.datacity.numa.co

DeFilippi, R. R. (2018). *Boosting, Bagging, and Stacking—Ensemble Methods with sklearn and mlens*. Retrieved from https://medium.com/@rrfd/boosting-bagging-and-stacking-ensemble-methods-with-sklearn-and-mlens-a455c0c982de

Demium Startups. (2019). *Demium*. Retrieved from https://demiumstartups.com/en/

Endeavor. (2019). *Endeavor*. Retrieved from https://endeavor.org

ESADE Ban. (2019). *ESADE*. Retrieved from https://www.esadealumni.net/en/entrepreneurship/entrepreneurs

Freshworks. (2019). *Freshworks*. Retrieved from Sales Funnel: https://www.freshworks.com/freshsales-crm/sales-funnel/

Genç, Ö. (2019). *Medium*. Retrieved from Notes on Artificial Intelligence, Machine Learning and Deep Learning for curious people: https://towardsdatascience.com/notes-on-artificial-intelligence-ai-machine-learning-ml-and-deep-learning-dl-for-56e51a2071c2

Gill, N. S. (2019). *Upwork*. Retrieved from https://www.upwork.com/hiring/for-clients/artificial-intelligence-and-natural-language-processing-in-big-data/

Google Campus. (2019). *Google Campus*. Retrieved from https://www.campus.co

Hubspot. (2014). *State of Inbound*. Retrieved from https://cdn2.hubspot.net/hub/53/file-1589882006-pdf/HubSpot-State-of-Inbound-2014.pdf

IE Venture Labs. (2019). *IE*. Retrieved from https://www.ie.edu/entrepreneurship/programs-initiatives/initiatives/venture-lab/

Impact. (2019). *Impact*. Retrieved from https://www.impact-accelerator.com

Import.io. (2019). Retrieved from Creating an interactive extractor.: https://help.import.io/hc/en-us/articles/360000055751-Creating-an-Interactive-Extractor

Insights, C. (2016). *European tech map with the most well-funded VC backed tech startups*. Retrieved from https://www.cbinsights.com/research/top-startups-europe-map/

Jason Rowley, Crunchbase News. (2017). Retrieved from TechCrunch: https://techcrunch.com/2017/05/17/heres-how-likely-your-startup-is-to-get-acquired-at-any-stage/

Jeffcock, P. (2018, July). *Oracle Blogs*. Retrieved from What's the Difference Between AI, Machine Learning, and Deep Learning?: https://blogs.oracle.com/bigdata/difference-ai-machine-learning-deep-learning

Kapoor, S. (2017). *Visualizing the Confusion Matrix*. Retrieved from https://www.sanyamkapoor.com/machine-learning/confusion-matrix-visualization/

Keras. (2019). *The Sequential model API*. Retrieved from https://keras.io/models/sequential/

Lanzadera. (2019). *Lanzadera*. Retrieved from https://lanzadera.es

Lanzadera Accelerator. (2019). *LANZADERA*. Retrieved from https://lanzadera.es/proyectos/

Leo Breiman, UC Berkeley. (1996). *Stacked Regressions*. Retrieved from https://statistics.berkeley.edu/sites/default/files/tech-reports/367.pdf

M Lex Market Insight. (2018, March). *M Lex Market Insight*. Retrieved from https://mlexmarketinsight.com/insights-center/editors-picks/antitrust/north-america/linkedin,-hiq-spar-at-ninth-circuit-in-data-scraping-case

Mariya Yao, Forbes. (2018, April). *Forbes*. Retrieved from https://www.forbes.com/sites/mariyayao/2018/04/10/14-ways-machine-learning-can-boost-your-marketing/#53099c7611b6

Metrick, A. (2007). Venture Capital and the Finance of Innovation. John Wiley & Sons.

NY Times. (1989). *Venture Capital Loses its Vigor*. Retrieved from https://www.nytimes.com/1989/10/08/business/venture-capital-loses-its-vigor.html

Pipedrive. (2019). Retrieved from Sales Pipeline Management: https://www.pipedrive.com/en/resources/sales-pipeline-management

Pitchbook. (2018). *13 charts explaining Europe's VC industry*. Retrieved from https://pitchbook.com/news/articles/13-charts-explaining-europes-vc-industry

Proceedings of the National Academy of Sciences of the USA. (2019, 01 22). Retrieved from https://www.pnas.org/content/116/4/1074

Pwc. (2017). Total U.S. Venture Capital Investments.

Saravia, E. (2018). *Medium*. Retrieved from Deep Learning for NLP: An Overview of Recent Trends: https://medium.com/dair-ai/deep-learning-for-nlp-an-overview-of-recent-trends-d0d8f40a776d

Scikit-learn. (2019). *Scikit-Learn Documentation*. Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

ScrapeHero. (2018). Retrieved from How To Rotate Proxies and IP Addresses using Python 3: https://www.scrapehero.com/how-to-rotate-proxies-and-ip-addresses-using-python-3/

Seed Rocket. (2019). *Seed Rocket*. Retrieved from https://www.seedrocket.com

Simon Duchatelet, The World Bank. (2018, August). Retrieved from The World Bank: https://blogs.worldbank.org/psd/voices/why-providing-pre-seed-and-seed-capital-essential-step-bringing-west-africa-and-sahel-s

StackOverflow. (2018). *stackoverflow*. Retrieved from https://stackoverflow.com/

Startupxplore. (2019). *Startupxplore*. Retrieved from https://startupxplore.com/en

Statista. (2018). *The 100 largest companies in the world by market value (in billion US dollars).* Retrieved from https://www.statista.com/statistics/263264/top-companies-in-the-world-by-market-value/

Synced. (2017, July). *Medium*. Retrieved from Applying Multinomial Naive Bayes to NLP Problems: A Practical Explanation: https://medium.com/syncedreview/applying-multinomial-naive-bayes-to-nlp-problems-a-practical-explanation-4f5271768ebf

Tetuan Valley. (2019). *Tetuan Valley*. Retrieved from https://www.tetuanvalley.com

Udacity. (2018). *Udacity Machine Learning Engineer Nanodegree*. Retrieved from https://www.udacity.com/course/machine-learning-engineer-nanodegree--nd009t

Vault. (2018). *Venture Capital*. Retrieved from http://www.vault.com/industries-professions/industries/venture-capital.aspx

Vault. (n.d.). *Venture Capital*. Retrieved from http://www.vault.com/industries-professions/industries/venture-capital.aspx

Visual Capitalist. (2016). *The Largest Companies by Market Cap Over 15 Years*. Retrieved from https://www.visualcapitalist.com/chart-largest-companies-market-cap-15-years/

Wayra. (2019). *Wayra*. Retrieved from https://es.wayra.co

Webscraper.io. (2019). *Webscraper.io*. Retrieved from https://www.webscraper.io

Webscraper.io. (2019, 3). *www.webscraper.io/.* Retrieved from https://www.webscraper.io/

WILDML. (2015, December). *Implementing a CNN for Text Classification in TensorFlow*. Retrieved from http://www.wildml.com/2015/12/implementing-a-cnn-for-text-classification-in-tensorflow/

World Ecnomic Forum. (2017). *Europe's venture capitalists are closing the gap with Silicon Valley*. Retrieved from https://www.weforum.org/agenda/2017/11/europe-venture-capitalists-silicon-valley/