Automatically match people with jobs

Enrich personal data provided by people to create better matches

8vance

manage your talent

Author	:	**Tim Hermens**
Place	:	**Blerick**
Date	:	**06-06-2016**
Version	:	**0.12**

Student information

Name:	Hermens T.
Student number:	2210369
Graduation course:	Software Engineering (full-time)
Graduation period:	15-02-2016 to 24-06-2016

Company information

Name:	8vance Matching Technologies BV
Department:	Research & Development
Location:	Blerick, Kazernestraat 17

Company mentor information

Name:	Keuren P.
Function:	Neural Software Developer

School mentor information

Name:	Schouten G.

Thesis information

Title:	Automatically match people with jobs
Date of issue:	-
Confidential:	Strictly confidential (no distribution allowed)

Signed and approved by company mentor

Date:	-
Signature:	

Preface

This thesis contains a description of the project I've executed for the company 8vance Matching Technologies BV. This company is active in the data science area (also known as Big Data) of the IT industry. One of their products is called AIMA, which is a digital agent that is able to match profiles of people with jobs. The main objective of my project is to improve the quality of matches of people with jobs by enriching the profile data of people. This graduation project started in February 2016 and ended in June 2016.

This project has been one of the most challenging projects I've done, mainly because of my limited experience with the data science area. With help and insights from the company's data science experts (Sabrina Ziebarth, Lou Cremers, Paul Keuren, Jan Jacobs) I managed to overcome many challenges, for which I would like to express my gratitude. I want to thank Paul Keuren in particular for his thorough feedback and excellent support. And last but by no means least, I would like to thank Gerard Schouten for his close involvement and continuous support throughout the whole project.

Table of contents

Summary

(An informative summary with a maximum length of ONE page, summarizing the whole thesis.

Describe the company and the problem that needed to be tackled.

Describe the chosen approach with arguments.

Describe the results.

Describe the conclusions and recommendations.

DON'Ts:

Don't refer to other chapters of the thesis.

Don't go too in-depth on details.

Don't use difficult terms somebody may not understand. Anyone should be able to read the summary.

Don't add images or bulleted lists.)

Samenvatting

(The same rules apply from the previous chapter.)

Glossary

Term	Definition
AE (Analysis Environment)	This is one of 8vance's systems that scrapes and processes the profile data from social networks.
AIMA (Automatic Intelligent Matching Agent)	This is one of 8vance's products that's able to match people with jobs.
Big Data	Big data refers to the approach to data of “collect now, sort out later”…meaning you capture and store data on a very large volume of actions and transactions of different types, on a continuous basis, in order to make sense of it later. (Dutcher, 2014)
CR (Coverage Rate)	Coverage Rate stands for the percentage of user-specified data that can be translated to a value specified in a taxonomy.
Job DNA	The job DNA contains information about the required hard and soft skills, years of experience, languages, and other requirements the employee must fulfil to be eligible for the job.
ME (Matching Engine)	This is one of 8vance's products that can find similarities between profiles and create matches between the most similar profiles. For instance, it's able to match jobs and talents for 8vance's AIMA product, and it's able to mach real estates and buyers for 8vance's Domizz product.
PID (Project Initiation Document)	This is a document that contains information about a project and the approach and strategy that will be used to reach the project's goal.
SAD (Software Architecture Document)	This is a document that contains information about an application's software architecture.
Talents	People who are looking for a job.
Taxonomy	An hierarchical structure of data where one parent value can have links to an unlimited number of child values. For instance, *Fruit* could be a parent value and *Apple*, *Banana*, etc. could be linked child values. (Progress Sitefinity CMS Documentation, 2016)
URS (User Requirements Specification)	This is a document that contains the requirements of the stakeholders for an application.

1 Introduction

Imagine you won't ever have to search for a job that exactly matches your skill set and personal desires, but that job will automatically find you instead. It almost sounds too good to be true, but the company 8vance Matching Technologies is convinced this can become a reality. Better yet, they've already developed an early build of this system, which is called AIMA, that's able to automatically match jobs with talents[1] with an impressive accuracy.

8vance Matching Technologies is a relatively young and innovative organisation that's active in the data science area of the IT industry. With AIMA, they aim to render recruitment and career development smart, social, simple, cost-effective and fast. Two other notable products they're working on are Sjerlok and Domizz. Sjerlok is an artificial intelligence system that uses Big Data to track down stolen goods for insurance company Delta Lloyd. Domizz is an online platform for buying and selling real estates, involving an automatic digital real estate agent that automatically finds matches between estates and interested buyers. This project is focussed on AIMA.

All of the company's products are powered by Big Data. However, apart from benefits there're also problems with Big Data. To create matches between jobs and talents, data is collected of the jobs and talents from social networks such as LinkedIn and Xing as well as 8vance's internal database. Talent data acquired from social networks misses useful information more than 98 percent of the time. The most notable missing information are the talent's owned skills. It's extremely rare that talents have specified all of their owned skills. This is problematic because the matching quality between job and talent is the most dependant on the skills a talent owns. The more skills are missing, the higher the likelihood of a mismatch between job and talent. A mismatch means the failure of AIMA to deliver its promise which leads to unhappy customers. Therefore the focus and the goal of this project lies on complementing the skills of the talent data acquired from social networks to increase the quality of matches.

A selection of three methods are used that provide strategies to steer the project in the right direction and ultimately achieve the project's goal. The first one is Agile, which is a work methodology for incremental development and continuous improvement. Research plays a major role in this project and a majority of the requirements are dependent on the research, which makes Agile a good choice. Furthermore, Fontys' five strategies research method and phasing are methods that are used in this project. See chapter **4** to find more information about these methods.

This document contains a description of this project's process, results and key decisions that were made. Information about the company 8vance Matching Technologies can be found in chapter 2. Chapter 3 provides an in-depth description of the assignment and the project's goal. The approach of this project is addressed in chapter 4. Chapter 5 describes the orientation phase in which the Project Initiation Document (PID) was created and the initial research was executed. The research and solution phase is discussed in chapter 6, including topics such as: the User Requirements Document (URS), the Software Architecture Document (SAD), the creation of the algorithm and the algorithm library. Chapter 7 contains information about the completion phase, including a conclusion and recommendation of the research, the parts of the architecture that still needs to be implemented, and a summary of

[1] Talents are people who are looking for a job.

the user requirements that aren't or are partially implemented and thus need attention. And finally, chapter 8 contains the conclusion and recommendations of this project.

2 The company

2.1 Foundation and mission

8vance Matching Technologies BV is a relatively young and innovative organisation that's active in the data science (also known as Big Data) area of the IT industry. The company was founded at November 2012 with the mission to render recruitment and career development smart, social, simple, cost-effective and fast. There're five main reasons why the company wants to invest to accomplish this mission (based on the reasons specified in Figure 1).

Firstly, unemployment still remains in the top 10 world problems (Hutt, 2016). The partial cause of this problem is the fact that vacancies are spread all over the internet on a variety of websites (e.g. online job boards or a company's website) and the unemployed have trouble finding appropriate vacancies. The company sees an opportunity here to collect as many vacancies and profiles of people as possible from a variety of websites and store them on one central place. Since finding a perfect job (or employee for a company's recruiter) is a challenge in itself, the company wants to provide assistance in the form of an Automatic Intelligent Matching Agent (AIMA) that's able to match people with jobs. This contributes in decreasing the unemployment rate.

Secondly, a shocking number of 87 percent of the employees aren't happy at work. A Gallup research has revealed that only 13 percent of the employees are happy and engaged in their work. 63 percent are disconnected and not engaged in their work (130 million employees in Europe). 24 percent are even undermining the company and are actively disengaged in their work (50 million employees in Europe). (Crabtree, 2013) 8vance aims to solve this problem by being one of the first to combine soft and hard skills to find jobs that matches someone's skill set and personal values.

Thirdly, the company distinguishes itself from the competition by offering an even smarter, state-of-the-art matching engine. The company will be the first to combine both hard and soft skills to improve the matching quality between people and jobs.

Fourthly, according to researches, approximately half of all (current) jobs will disappear by 2030. (Frey T. , 2012), (Frey & Osborne, 2013) This means that finding new jobs will become increasingly more important in the coming years. This is when a service like AIMA could really shine.

And lastly, the matching engine that'll be developed to accomplish the mission can be used to accomplish a wide variety of incredible and innovative things (yet to be discovered). For instance, it would be possible to create a career assistance agent that's able to offer suggestions of skills you should achieve and/or educations you should follow to make progress in your career development. In other words, there's a lot of value that can be harvested.

Figure 1 - Reasons why 8vance want to create AIMA

2.2 Products

The company is working on several products. The following products are the three main products they're working on:

- **AIMA**. An automatic intelligent matching agent that's able to match talents with jobs. Since the project is focussed on this product, it'll be discussed more in-depth in the upcoming section.
- **Sjerlok**. An artificial intelligence system that uses Big Data to track down stolen goods for insurance company Delta Lloyd. (Vonk, 2015)
- **Domizz**. An online platform for buying and selling real estates. An automatic digital real estate agent finds matches between estates and interested buyers.

Because of the small number of employees and strict deadlines of the AIMA and Sjerlok products, the company had to cease the development of Domizz. However, an ex-CEO of www.jaap.nl is now working for 8vance and has taken up the responsibility to continue the development of Domizz.

2.2.1 AIMA

As said before, AIMA is an automatic intelligent matching agent that's able to match talents with jobs. This service is available on 8vance's website (www.8vance.com) after registering an account for free.

The target audience of this product are companies and talents around the world. The product will mainly be used by the companies' recruiters and the talents.

The general use of the product consists of the following three steps. (8vance Matching Technologies, 2016)

Step 1 - Registering

Companies can register with 8vance through a online wizard. The wizard prompts the user to insert the required information about the company it'll need for its matching algorithm. A short leadership test is also part of the registration, which improves the results of the matching algorithm. The company can start posting vacancies of jobs when it's registered. The wizard assists step by step in the creation of a full job DNA. The job DNA contains information about the required hard and soft skills, years of experience, languages, and other requirements the employee must fulfil to be eligible for the job. This way AIMA gathers all the required information to find better suited talents for the job.

Talents can also register with 8vance through a online wizard. They have the opportunity to either register with their social network profile such as LinkedIn, or register a new account. When registering a new account, the wizard prompts the user to insert the required information about himself it'll need for its matching algorithm. After the registration, talents can start an automatic search to find the best matching jobs for them.

Step 2 - Matching and scouting

AIMA uses several methods to find talents. When a company activates one of their vacancies, AIMA searches the internet for public CVs to find talents (this is called scouting). She will establish who has worked where for how long and which competences have played an important role. AIMA may automatically add missing competences which are in line with a talent's working experience and educational background to improve the quality of matching (this part is the goal of this project).

In addition to publicly available information, talents can also create their own 8vance profile. Recruiters have the possibility to upload their database of talents and have AIMA match these data as well. AIMA creates a list of matches containing information about the found job matches or talent matches.

Step 3 - Advertising and social media

In a few steps, companies can create creative and efficient online campaigns for their vacancies, which includes a link to their own home page. Companies can also create advertisements on job sites and social media. Talents who have seen the advertisements and are interested, can register with 8vance so that they're immediately matched to the vacancy in question.

2.3 Business plan

Both the companies and the talents benefit greatly from AIMA's service. However, companies will overall have a greater benefit because of the following reasons (Stoffels, 2016):

- AIMA will reduce the costs of the process to find qualified talents up to 70 percent.
- AIMA accelerates the recruitment process up to 75 percent.
- AIMA reaches up to 50 times more candidates by combining profiles of talents from the companies' databases, 8vance's databases and social networks.
- AIMA minimizes the costs of mismatches because of the high matching quality.
- AIMA offers a better overview and transparency of the available talents.

Because of these added benefits for the companies and their financial interests, 8vance offers the companies a reasonable pricing model, which can be seen in Table 1. (Stoffels, 2016) The pricing is based around the number of jobs a company can submit that will be included in the job-talent matching process.

Single post plan	Submit less than 50,000 jobs Submit more than 50,000 jobs	€ 300.- € 900.-
Volume plan	5x Submit less than 50,000 jobs 5x Submit more than 50,000 jobs	€ 1,250.- € 3,500.-
License plan	6 months unlimited job submissions 12 months unlimited job submissions	€ 600.- per month € 500.- per month
Additional tools	Marketing, advertising and social media	Individual pricing, TBD

Table 1 - AIMA's pricing model for companies

The single post plan means a company can submit the given number of jobs all at once at one specific moment. The volume plan means a company can submit the given number of jobs all at once at five different moments. The license plan allows the company to submit new jobs at any given moment during the license period, meaning they can continuously update their available jobs without any further costs.

AIMA is completely free to use for the talents. However, additional future services like a career assistance agent, a generation of an auto-completed profile variant of the talent's profile, or the generation of smart views of the current market with a filtering of the talent's preferences and skills (see Figure 2), can be paid services.

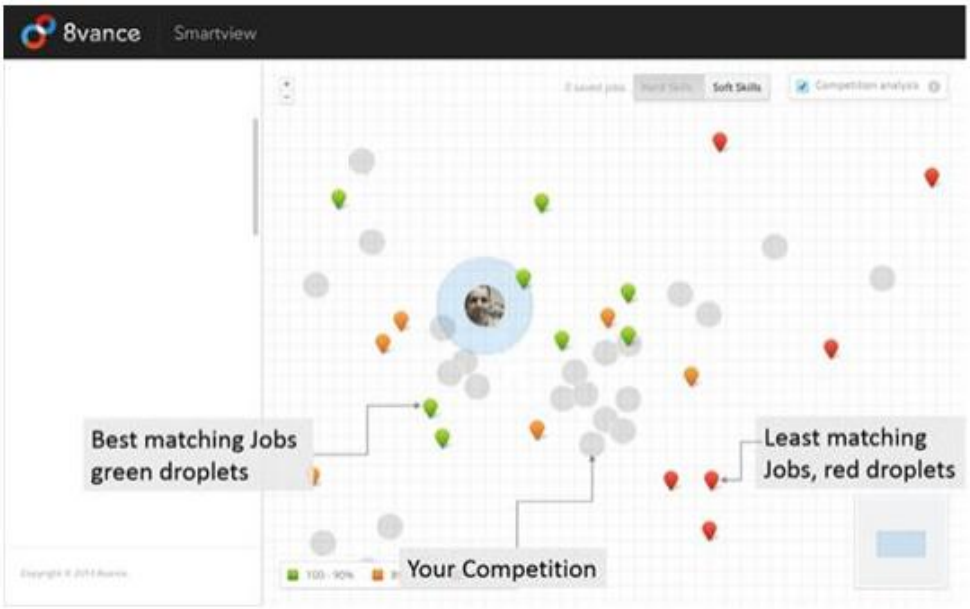

Figure 2 - Illustration of a smart view (job market). The coloured droplets indicate jobs the talent has a match with. Red droplets indicate a 70%-80% match, orange 80%-90% and green 90%-100%. The gray circles are positions of other talents (competition).

2.4 Organisation

The company currently consists of a total of 15 employees. An organisation chart of this company can be seen in Figure 13 in appendix **C**.

The company is situated in the Netherlands and Romania. The software development team is situated in Romania and the data science and marketing teams are situated in the Netherlands. I'm part of the development team in the Netherlands.

The main spoken language at the company is Dutch. However, there are some German employees who can't fully understand Dutch (yet). German or English is spoken when they're part of a discussion.

The work environment is very open, laid back and informal. The employees always have time for each other, either to help each other, or to have conversations on literally any subject at any time of the day. The company feels like one big happy family, as everyone (without any exception) lunches together, makes decisions together, has fun together, shares personal experiences, takes turns doing the shopping and dishes, etc. All in all, the atmosphere is very positive and you feel part of the family.

2.4.1 Manufactuur

As can be seen in Figure 3, the company's headquarter is situated in the Manufactuur building in Blerick in the Netherlands. The Manufactuur used to be one of the fabric halls of Leolux (a company that specializes in design furniture), but is now an open and transparent work floor for promising start-up companies. The Manufactuur currently houses over five companies, such as Accerion, Yubu and 8vance. (Kickstart Venlo, 2016)

Figure 3 - 8vance's headquarters in Blerick in The Netherlands

3 The assignment

3.1 Project goal

The company is working on a product called AIMA that matches jobs and talents as accurately as possible. Another product called ME (Matching Engine) is responsible for actually creating the matches between profiles. Essentially, AIMA uses the ME to create the matches between jobs and talents.

The ME uses a variety of algorithms to create the matches. The algorithms need to be trained with data initially for them to be able to create matches. The algorithms are trained on company, job and talent data that 1) are scraped from social network websites such as LinkedIn and Xing, and 2) are retrieved from 8vance's internal database. One major problem is that the talent data from the first data source misses useful information more than 98 percent of the time.

The information that's missing varies from talent to talent. The missing information can consist of followed educations, work experience, owned skills, the work industry, dates, and more. The most notable missing data in this list are the owned skills, because the matching quality is the most dependant on the skills a talent owns. The more skills are missing, the higher the likelihood of a mismatch between job and talent. This doesn't necessarily result in the failure of the ME, for it has found the best matches based on the incomplete information. But it does result in the failure of AIMA to find the best matches between talents and jobs, because the lack of skills leads to an incomplete picture of the talent's skill set which severely limits the possibilities of matches. Therefore the focus and the goal of this project lies on complementing the skills of the scraped talent data to increase the quality of matches.

The preferred solution to this problem comes in the form of an algorithm that's able to complement the missing skills for the provided talents. This solution can be used in a stand-alone tool to provide skill suggestions on request, and can be integrated in the scraping process to automatically complement the skills for newly scraped talent data.

3.2 Scope

The scope of this project is specified in Table 2, which is directly taken from the appendix Project Initiation Document (PID).

Belongs to the project	Doesn't belong to the project
A project plan.	Creating predictions of skills someone could own in the near future.
A research document where found solutions are discussed.	The information provided by the talents isn't checked on validity.
A solution that complements the missing skills for the provided talents.	The integration of the solution in the scraping tool.
A stand-alone tool in which the solution is integrated that provide skill suggestions and can be used to test the solution.	

A user requirements document.	
A software architecture document.	
Only the missing skills will be complemented for the talents from LinkedIn.	

Table 2 - Scope of the project

3.3 Constraints

The algorithm and stand-alone tools have some constraints. The algorithm (solution) has the following constraints:

- The scraping of the talent data is done in a Python application and the solution must be able to be integrated in this application. This means the solution must be compatible with Python.
- The prediction of missing skills for one profile at a time cannot take longer than one second (for on-demand use).
- The prediction of missing skills for one million profiles at a time cannot take longer than 24 hours (for automated use).

The stand-alone tool has the following constraints:

- The stand-alone tool that's developed must be platform-independent.
- The stand-alone tool that's developed must work locally (without internet connection).

3.4 Project changes

There's been one major change during the project. A software test plan first was also part of the deliverables, but was removed because of two reasons. The first one being there wasn't enough time to create a full-fledged test plan next to all the other, more important deliverables. The second reason is that primarily the quality of the found solutions need to be tested. Testing these solutions is already part of the research document, meaning the most important tests are already covered.

3.5 Research

During this project's research, the following main question needs answering: *Which solutions can be used to complement the missing skills for talents as accurately as possible?* To answer this question, the following sub-questions are answered:

- What data is available of the talents?
- Which selection of the talents' data can be used to help to determine the missing skills?
- What are possible solutions to complement the missing skills with help of this selection of talents' data?
- Which solutions perform best and is the company 8vance satisfied with?

The research framework that's taught in the research course at Fontys is used in this project (Van Turnhout, et al., 2014). This framework entails five different strategies that can be used to answer the research questions. Read the next chapter to find more information about this framework and why it's chosen.

4 The approach

4.1 Introduction

The approach of this project is described in this chapter. The work and research methods that were used in this project as well as the project planning and contact moment will be discussed. The original approach is written in the PID which can be found in appendix A.

4.2 Methods

The methods that were used in this project are: phasing, Fontys' five strategies research method, and Agile. Each method will be described below.

4.2.1 Phasing

Phasing is used for setting up the project so that the project's executor has more control over the project's process. The phases that are used are inspired by the phases from a method called Tienstappenplan. The phases are:

- Orientation phase;
- Research and solution phase;
- Completion phase.

These phases helps in structuring the project and securing quality. Similar phases are also commonly used for graduation projects in the IT industry because of its effectiveness. (Kempen & Bennink, 2016)

Orientation phase

The orientation phase has the PID as the final product. This phase exists of five steps:

- **External orientation**. The goal of this step is to acquire insights in the desires and goals of the company for the project. This step is performed in preparation for the intake interview.
- **Intake interview**. The company is interviewed in this step in order to determine the exact desires and goals of the company for the project.
- **Orientation activities**. The situation of the company at the moment before the project started is described. All the information that was obtained from the previous steps are gathered and an own opinion is formed on the desires and goals of the company.
- **Analysis**. An own interpretation of the project's goal and approach is formulated in the PID.
- **Feedback**. The company and school mentor evaluate the PID and determine whether or not the project may be started.

When all steps are taken, all of the project stakeholders will know what to expect of the project. The orientation phase also includes an initial research which is focussed on obtaining knowledge of the data that will be worked with.

Research and solution phase

The majority of the products are finished in this phase. This phase exists of three steps:

- **Planning and project organisation**. In this step, the planning and organisation as defined in the PID are realized.
- **Research and solution.** In this step, the project as described in the PID is executed and the Agile work method is followed.
- **Solution plan**. The goal of this step is to motivate why a particular found solution is better than another. This way, the company can better assess the value of the solutions.

Completion phase

This final phase exists of two steps:

- **Conclusion**. A conclusion and recommendation of the research is formulated which the company can use to continue the research and possibly improve AIMA.
- **Completion**. The thesis and products are finalized and handed over to the company. The concluding presentations for the company and school are prepared and executed.

4.2.2 Agile

During the *Research and solution* step in the research and solution phase, the Agile work methodology is used. This methodology is used because of several advantages.

Incremental development and continuous improvement

Agile realises the possibility for incremental development. The research is a long-running process where good solutions may or may not be found. Therefore it's impossible to say at the start of the project which things can and cannot be realised and/or achieved. Hence it's a good idea to start small and realize the most important requirements first. As the research progresses and more knowledge is required, requirements can be added or modified accordingly. (Moran, 2010)

Well-known

Agile is a well-known work methodology in the IT industry. Up to 37% of the companies use this methodology. Both companies and schools are familiar with Agile and acknowledge its strengths and usefulness. (Langley, 2016)

Transparency

Transparency plays a crucial role in Agile and is an important aspect why it's so effective. Agile forces a close communication between every stakeholder. Through communication, problems and weak points are discovered, discussed and resolved faster. This brings higher transparency and helps to prevent escalating problems. (Moran, 2010)

Quality

Testing and validating is an integral part of Agile. A requirement is only considered to be successfully implemented when it satisfies all the demanded quality requirements. (Waters, 2007)

Risk management

Agile development seeks to avoid the issues of "the customer got what they asked for, but it isn't what they wanted" because of a misunderstanding in the requirements. With Agile, working solutions are frequently delivered and inspected in order to avoid these issues. If there're misunderstandings, immediate corrections can be made. (Moran, 2010)

4.2.3 Fontys' five strategies research method

The research framework that's taught in the research courses at Fontys is used in this project. This framework entails five different strategies that can be used to answer the research questions. Figure 4 shows these strategies.

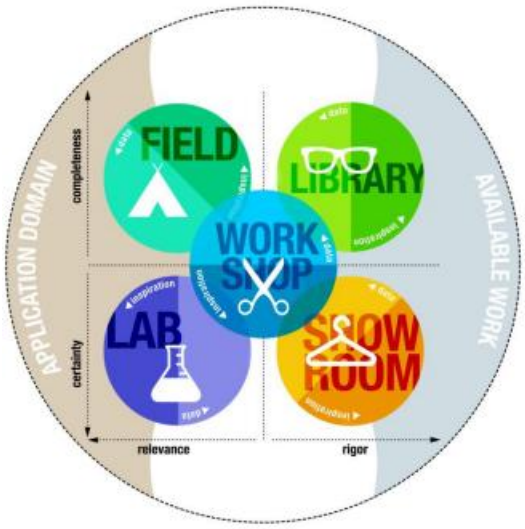

Figure 4 - The five research strategies

Let's have a brief look what each strategy means. (Kempen & Bennink, 2016) (Van Turnhout, et al., 2014)

- **Field**. You dive into the field to gather information around the application domain. The gathered information isn't always reliable, so it needs to be validated accordingly.
- **Lab**. In this strategy you test an aspect of your solution. This always involves measuring so you know your solution is the right one.
- **Workshop**. You get your hands dirty to explore new ways or to validate a solution. The way you go about this is methodical and structured.
- **Library**. You search for available work (literature or software) or knowledge (domain experts) that could serve as the foundation of your solution. In contrast to the Field strategy, this obtained information is reliable to a large extent.

- **Showroom**. The purpose of this strategy is to compare your solution to other solutions in order to measure the quality of your solution.

These strategies can be used to answer the research questions from different viewpoints. Each strategy also involves its own appropriate and unique sources (sources of Lab can be tables with test results, and sources of Library can be interviews with experts or research papers, etc.). Providing answers from these different viewpoints and using different sources make the answers more credible. (Kempen & Bennink, 2016) These are the main reasons why this research framework is chosen.

The Lab, Workshop and Library strategies are the followed strategies to answer the research questions. The Library strategy is mainly used to acquire information from 8vance's data science experts and their research as well as published research papers on the internet. The Workshop strategy is used to try out own or found solutions, and to validate them. The Lab strategy is used to compare the found solutions with each other and with the company's current skill suggestion solution.

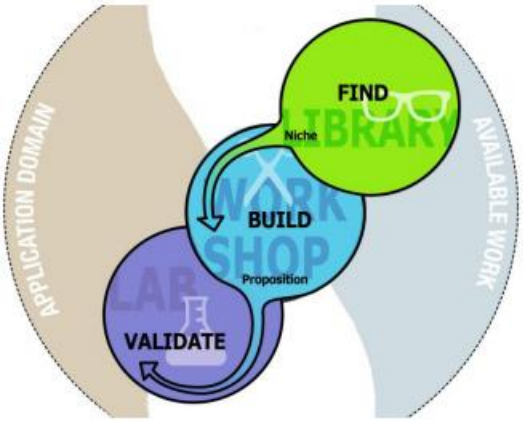

Figure 5 - Validated Solution research pattern

The combination of these strategies is also called the *Validated Solution* research pattern (see Figure 5). This pattern is particularly useful to solve deficiencies in existing solutions. This project aims to solve a deficiency in the AIMA product, which means the chosen strategies are appropriate choices. (Van Turnhout, et al., 2014)

Read the PID in appendix A for more information about the discussed methods.

4.3 Project planning

The planning is created based on the pre-known data and situation. The phasing in the TSP method made for a great foundation to be used for the planning. Table 3 contains a simplified version of the project planning which is based on the project planning of the PID, which can be found in appendix A.

Task	Days	Start	End
Orientation phase	**10**	**15-02-2016**	**26-02-2016**
PID	10	15-02-2016	26-02-2016
Initial research	5	22-02-2016	26-02-2016
Research and solution phase	**75**	**22-02-2016**	**03-06-2016**
Research document	75	22-02-2016	03-06-2016
URS	5	07-03-2016	11-03-2016
SAD	25	14-03-2016	15-04-2016
Create analysis tool	50	28-03-2016	03-06-2016
Implement algorithm	50	28-03-2016	03-06-2016
Implement algorithm library	50	28-03-2016	03-06-2016
Completion phase	**15**	**06-06-2016**	**24-06-2016**
Finalization of the products	15	06-06-2016	24-06-2016
Conclusion and recommendations	5	06-06-2016	10-06-2016
Final presentation preparation	13	08-06-2016	24-06-2016
Company presentation	1	10-06-2016	10-06-2016
Deadline thesis	1	14-06-2016	14-06-2016

Table 3 - Project planning

5 Orientation phase

5.1 Introduction

The orientation phase has the goal to acquire insights in the desires and goals of the company for the project and document them in the PID. This phase also includes an initial research which is focussed on obtaining knowledge of the data that will be worked with. The goal of this phase is to get an answer to the research's first sub-question: *What data is available of the talents?*

5.2 PID

The first activity in the orientation phase was to acquire insights in the desires and goals of the company for the project and document them in the PID. The following steps were taken to create the PID:

1. **External orientation**. The website and assignment description was analyzed to form an own opinion on the desires and goals of the company for this project. This step is performed in preparation for the intake interview.
2. **Intake interview**. An intake interview is taken with the company. The company's exact desires and goals for the project are determined.
3. **Orientation activities & analysis**. The situation of the company at the moment before the project started is described. All the information that was obtained from steps 1 and 2 are analyzed and used to describe the expected desires and goals of the company. This is formulated in the PID.
4. **Feedback**. The company evaluates the PID to determine whether or not the project description as defined in the PID meets their desires and goals. The school mentor also evaluates the PID and determines whether or not the described project lives up to be an appropriate graduation project.

When all these steps were taken, all of the project stakeholders knew what to expect of the project. The PID also serves as a nice guideline for the execution of the project.

5.3 Initial research

After the PID had been approved, an initial research was started to find out what the data looks like that will be worked with, and which other resources are available for use. This is done to get a better understanding and acquire more knowledge on the problem domain. Sabrina Ziebarth - one of 8vance's data science experts - provided 1000 scraped LinkedIn profiles that could be analyzed. With help of one of Sabrina's documents containing information about the scraped data and after analyzing the provided scraped profiles, all the profile data could be described (see Table 4).

The analysis also included a data syntax analysis, which means that the different occurrences of data is analyzed. This revealed that every data field except for profile_id and scraped_at can be empty. Additionally, the dates can all be specified in different formats, provided data in a profile can be specified in multiple different languages, and a lot of punctuation or other unexpected data is provided in data fields where you don't expect it.

Data field	Description
Crawled_at	Contains the date the profile has been scraped at.
Educations	Contains a list of followed educations. Each education contains the following data: - **Date_start:** The starting date. - **Date_stop**: The stopping date. - **Degree**: The acquired degree after graduation. - **Degree_major:** A combination of the acquired degree after graduation and the followed major program. - **Institution**: The institution's name. - **Major**: The followed major program.
Experiences	Contains a list of past and present working experiences. Each experience contains the following data: - **Company**: The company's name worked at. - **Company_url**: The company's website. - **Date_start**: The starting date. - **Date_stop**: The stopping date. - **Function**: The carried out function. - **Location**: The company's location.
Industry	Contains the industry the person is currently active in.
Languages	Contains a list of languages the person is proficient at.
Locality	Contains the person's current home address.
Profile_id	Contains the profile's id.
Skills	Contains a list of skills the person owns.
Slogan	Contains a free text description of the person's current state (typically contains the function the person is carrying out and the company name).
Summary	Contains a free text description of the person about himself.
Summary_education	Contains a list of institution names where the person is currently studying.
Summary_past	Contains a list of company names of former companies the person has worked for.
Summary_present	Contains a list of company names of companies the person is currently working for.

Table 4 - LinkedIn profile data

5.4 Conclusion

This initial research truly showed the importance of pre-processing the data in order to make sense of it. It's clear that the data needs to be pre-processed in order to get qualitative data. You can't work with data that can't be made sense of.

6 Research and solution phase

6.1 Introduction

The majority of the products are finished in this phase. The project that's described in the PID is executed and the Agile work method is followed.

This chapter includes the following topics:

- User requirements specification.
- Software architecture document.
- Implementation.
- Depth research.

The goal of this phase is to get an answer on the research's second, third and partially the fourth sub-questions, which are:

- Which selection of the talents' data can be used to help to determine the missing skills?
- What are possible solutions to complement the missing skills with help of this selection of talents' data?
- Which solutions perform best and is the company 8vance satisfied with?

6.2 User requirements specification (URS)

This phase started with the specification of the requirements of the company's preferred solution in a URS. As described in the assignment chapter, the preferred solution comes in the form of an algorithm that's able to complement the missing skills for the talents. This solution can be used in a stand-alone tool to provide skill suggestions on request, and can be integrated in the scraping process to automatically complement the skills for newly scraped talent data.

By specifying these requirements in a URS, stakeholders can verify whether or not the depicted product contains all necessary features. The URS will be a contractual agreement, meaning the company cannot demand features not in the URS, whilst the developer cannot claim the product is ready if it doesn't meet an item of the URS. (Wikipedia, 2016)

The URS was created in co-operation with the stakeholders. One side of the stakeholders was primarily interested in the stand-alone tool, whilst the other side was interested in the automated solution. The main requirements for the automated solution are pretty straight-forward, for instance:

- The algorithm can predict missing skills for the provided profiles.
- The algorithm can calculate a certainty score for every predicted skill per profile.

The stand-alone tool serves as an analysis tool to analyze the results of an algorithm's predictions. The requirements for the stand-alone tool are more diverse, for instance:

- The user can start a prediction process for the provided profiles. This process returns the predicted skills, the certainty score per skill, and the execution time.
- The user can provide profiles to be used as the data source.
- The user can specify an algorithm he wants to analyze.
- The user can save and load the results of a prediction process.

The third requirement stands out the most, because it means that not only the solutions of this project can be analyzed with the tool, but also any other future solutions.

6.3 Software architecture document (SAD)

After the URS had been specified and approved by the stakeholders, a SAD was created. The goal of this SAD is to ensure the longevity of the system by considering and integrating important quality attributes. This document can be used as a reference guideline when the system lacks performance and needs to be scaled horizontally.

The architecture serves as the foundation of the system. It defines a structured solution to meet all the technical and operational requirements, while optimizing the quality attributes. Further, it involves a set of significant decisions related to software development and each of these decisions can have a considerable impact on quality, maintainability, performance, and the overall success of the system. The architecture is designed as such that it incorporates all the fundamental quality attributes. *This ensures the system's foundation meets the non-functional requirements, meaning the architecture can never be the limiting factor of the system.*

The software architecture is shaped by several quality attributes taken from the ISO 25010 model. This model was recommended by Paul Keuren - 8vance's neural developer expert. The official ISO 25000 website perfectly summarizes why this model is powerful for creating a software architecture: *"The quality of a system is the degree to which the system satisfies the stated and implied needs of its various stakeholders, and thus provides value. Those stakeholders' needs (functionality, performance, security, maintainability, etc.) are precisely what is represented in the quality model, which categorizes the product quality into characteristics and sub-characteristics."* (ISO 25000, 2015)

The SAD also follows the Kruchten 4+1 architecture model. The problem of most architectures is that they represent a bit of everything and fail to address the concerns of all stakeholders. This model was chosen because it addresses specific set of concerns of interest to different stakeholders in the system. (Kruchten, 1995) The SAD describes the five view points of this model:

- Use case view. Describes the functional requirements with a significant impact on the architecture.
- Logical view. Describes the different layers of the system.
- Process view. Describes the concurrency and synchronization aspects of the system.
- Implementation view. Describes the technical implementation of the layers of the system which were discussed in the logical view.
- Deployment view. Describes the mapping of the system onto the hardware and shows the system's distributed aspects.

6.3.1 A closer look at the architecture

Table 5 contains some of the most important quality attributes based on the ISO 25010 model for the system. These quality attributes were taken from the SAD that describes a multitude of quality attributes sorted by importance.

Quality type	Description
Performance	The algorithm is able to predict missing skills of 1 million profiles in 24 hours.
Adaptability/ scalability	The algorithm must be adaptable in different or evolving software products that support Python 2.7. Additionally, the system must support hardware scalability, meaning multiple servers could be introduced that are used to make predictions with the algorithm to surpass the 1 million profiles per day prediction goal.
Interoperability	The algorithm and 8vance's Analysis environment as well as the algorithm and analysis tool must be able to exchange information with each other.
Availability	The algorithm must be available to be used to create predictions at all times.

Table 5 - Important quality attributes of the system

Performance, adaptability/scalability, interoperability and availability are some of the most important quality attributes. It's important that the system's base architecture covers as many of the important quality attributes as possible. Any other important remaining quality attributes should be covered by choosing other sub architectures or design patterns.

The architectural designs or patterns that are discussed in SAD were chosen based on the quality attributes they offer. Let's have a brief look at the most interesting architectural designs or patterns that are discussed in the SAD.

System overview

Let's first have a look at the overall abstract context view of the system.

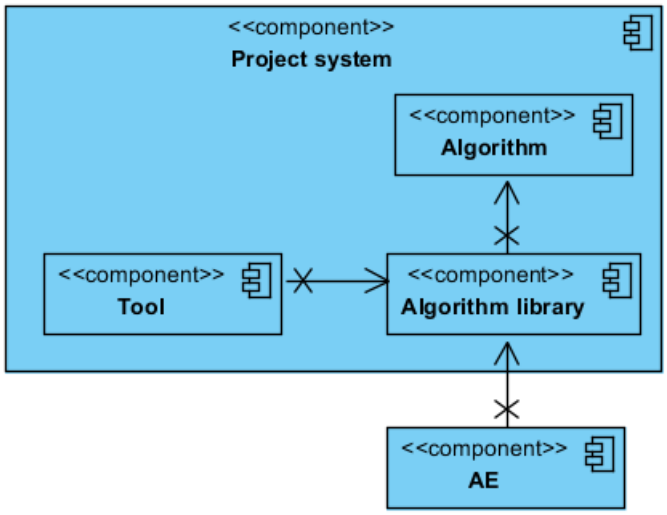

Figure 6 - Abstract context view of the system

In Figure 6, you can see a abstract context view of the system. The purpose of this context view is to provide an overall overview of other systems the project system communicates with. The stand-alone tool and algorithm are part of the project system, as is already known. The Analysis Environment (AE) is a system that's currently being used by the company to scrape and process the profile data. The stand-alone tool as well as the AE will use/communicate with the algorithm.

The Algorithm library is a newcomer. This component servers as an intermediate between the Tool, AE and the algorithm. It handles all the control checks and operations which are needed to communicate with the algorithm. The response of the algorithm is post-processed in such a way that the stand-alone tool and AE can use it as it wants to (e.g. for predictions: the tool expects to receive a list with *human-readable* skill predictions, certainty scores and execute time, whilst the AE expects a JSON array with skill predictions and certainty scores).

There're three other reason why this intermediate component has been introduced:

- The algorithm can be changed at any time, meaning a direct connection from the AE to the algorithm isn't desired. Either the AE would have to support additional functionality to dynamically load new algorithms, or a new component is introduced that does this job. The latter option is preferred.
- If the data scraping process is too slow, multiple AE systems can be introduced in the future to increase the performance of the data scraping process. These systems can easily be connected to the algorithm library to enable skill predictions.
- If the skill prediction process is too slow, multiple servers can be introduced in the future that run the algorithm library to increase the performance of the skill prediction process.

Choosing the architectural designs and patterns

The core functionality of the system all revolves around the Algorithm library component, which is why the base architecture is chosen based on this component.

In order to select the best architecture, research was done to compare the most promising and popular architectures that satisfy the most important quality attributes. The architecture shouldn't be too complex which would require additional development time.

Pipe-and-filter

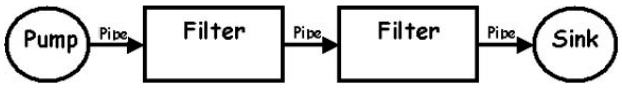

Figure 7 - Pipe-and-filter pattern

Initially, one of the most promising architectures was the pipe-and-filter architecture. Its strength lies in its performance and modifiability. The pipe and filter pattern makes use of parallelism which provides concurrency and high throughput, resulting in great performance. This architecture design is particularly useful for pre- and post-processing the profile data, and creating skill predictions. These steps can be seen as separate filters which are linked together and executed in a one-way order. However, having the flexibility to switch the filters isn't particularly useful for this system because there simply won't be a need to switch them. The pipeline always has to start with

pre-processing the data and stop with post-processing the data. The filters in between call the algorithm functionality to predict the missing skills. This order of filters won't ever change, so the strengths of this architecture design like modifiability and reusability aren't used. This means the strength of this design currently lies only in the performance. Additionally, this architecture makes it quite hard to handle exceptions and complex workarounds would need to be implemented in order to achieve it, which could negatively affect the performance. (Open Universiteit, 2016) (Keuren, Incidental questions, 2016) (March, 2003) (Sanders, Massingill, & Mattson, 2005)

MVC

Model-View-Controller (MVC) decomposes a given software application into three interconnected parts. The Model encapsulates the underlying data and business logic. The Controller responds to user actions and direct the application flow. The View formats and presents the data from model to user. This architecture is normally used in applications with a graphic user interface (GUI). However, the Algorithm library doesn't have a GUI, but it doesn't mean MVC can't be used. Better yet, the three different layers lend themselves perfectly for the algorithm. Both the AE and stand-alone tool want to execute the algorithm and expect similar input and different output interfaces. The View layer of MVC can be used to define this interface (one View for the AE, and one for the tool) and do the post-processing of the data for the AE and tool . The Controller layer of MVC can be used to check the validity of the profile data. The Model layer of MVC can be used to pre- and post-process the data and execute the algorithm's functionality like predicting missing skills. This layer always holds the knowledge data/models, so changes to the algorithm can be made in this layer without changing anything else. (Tutorialspoint, 2016) (Keuren, Incidental questions, 2016)

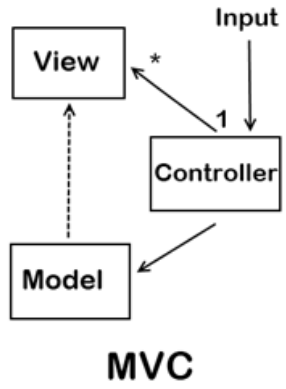

Figure 8 - MVC design pattern

The decision

Ultimately, the MVC architecture was chosen as the base architecture. The three M, V, C layers just lend themselves perfectly for the Algorithm library. This architecture didn't really have a great deal more advantages over other architectures, but it simply didn't have as many disadvantages. MVC has a separation of concerns, allowing changes to be made to the different layers without needing to change anything else. If performance is a concern, the Model layers which do heavy algorithm computing can be distributed on different physical tiers.

The architectures that were considered can also be found in Table 6. This table also includes architectures to achieve communication in a distributed system (the fifth and sixth architecture) and architectures to achieve a high

availability in a distributed system (the last three architectures). Based on the advantages, disadvantages and covered quality attributes, the Client-Dispatcher-Server and Load-balanced cluster architectures were chosen.

Architectural style or design pattern	Performance	Adaptability/scalability	Testability	Modifiability	Reusability	Modularity	Availability
Pipe and filter							
Layered (n-tier)							
Blackboard							
MVC							
Message broker							
Client-Dispatcher-Server							
Load-balanced cluster							
Asymmetric server clustering							
Failover clustering							

Table 6 - Degree to which an architectural style or design pattern supports the most important architectural-related quality attributes

The stand-alone tool change

The stand-alone tool has had one major change about what it was going to be. Initially, it was going to be a full-fledged GUI desktop application so that it could easily be used by anyone. After all, the tool would primarily be used to analyze the results of an algorithm's prediction, which would be interesting for 8vance's algorithm developers and other stakeholders who are just interested to see the quality of the predictions.

However, after several discussions it became clear that the tool wouldn't ever be used by other stakeholders other than the algorithm developers. It was decided that the tool had to solely focus on helping the algorithm developers to develop their algorithms. This means the tool doesn't necessarily have to be a GUI desktop application.

Ultimately, the choice fell on the tool becoming a Jupyter notebook. This decision was made because of the following reasons: (Guo, 2013)

- A Jupyter notebook adds the flexibility to change/improve the analysis code on the fly, so that the analysis results contain the exact information you're looking for. This is very useful if the standard analysis

implementation doesn't reveal the information you're looking for, or you want to change the presentation of the output.

- You can easily perform and save the results of multiple analysis in different cell blocks and take notes directly beneath each analysis' output. You don't have to bother with - for instance - creating image files that contain the analysis results and remembering the exact conditions that produced the files.
- A Jupyter notebook containing descriptive notes above each code block is fast and easy to use. For instance, you can quickly change the data source or algorithm to be analyzed by modifying their respective variables.

6.4 Implementation

After the base architecture had been chosen and an abstract UML model of the system based on this architecture was created, the implementation of the algorithm library started. Further work on the SAD ceased because the research needed to be started as soon as possible. Once the library has been implemented, data sources and algorithms found from the research can easily be hooked on the library and pre-processing or prediction operations can be performed. The results of these operations can then be compared to each other to find the best-performing algorithms.

Python was chosen as the programming language because the solutions need to be compatible with the company's AE which is written in Python 2.7, and the company has a lot of experience in Python (other products like the ME are also written in Python). Furthermore, Python is one of the most used programming languages for data science work and has many available modules that on this area. (Jeevan, 2015)

6.5 Depth research

The research started after the algorithm library had been implemented. The goal of this research is to acquire answers on the remaining three research sub-questions:

- Which selection of the talents' data can be used to help to determine the missing skills?
- What are possible solutions to complement the missing skills with help of this selection of talents' data?
- Which solutions perform best and is the company 8vance satisfied with?

6.5.1 The best selection of talent data

In order to find the best selection of talent data, all of the available profiles' data fields were analyzed and the usefulness of every data field was documented. The usefulness of a data field is determined by the expected correlation strength between the data field and a skill. Sabrina Ziebarth - 8vance's data science expert - helped with this. Table 7 contains the result of this analysis.

Social network	Locality	Industry	Summary	Slogan	Skills	Languages	Experiences	Educations	Academic degree	Interests / topcis	Wants	Publications	CV	Keywords	Disciplines
LinkedIn															
Xing															
Academia															
Researchgate															
About.me															
Medium.com															
Zoominfo															

Useful	Likely useful	May be useful	Likely useless	Highly likely useless

Table 7 - Usefulness of the data fields per social network. Totally useless data fields are excluded.

On basis of this table and the quality of the profile data, a conclusion could be made that the LinkedIn data has the most useful data fields. The *Industry*, *Skills*, *Experiences*, and *Educations* are the expected data fields that have a high correlation with skills.

6.5.2 Found solutions to complement the missing skills

The approach to find the missing skills was: find a solution that can calculate the probability rate of a talent owning a particular skill based on the specified information in his profile's data fields that have a high correlation with skills.

Before these solutions can be found, the profile data first needs to be pre-processed. The necessity of pre-processing the data was already revealed in the initial research.

Pre-processing

In its scraped state, the profile data doesn't provide much use to find correlations in the data. There're data fields that have different values, but that mean the same thing. For instance, if someone has `powerpoint` as his skill and someone else has `microsoft powerpoint` as his skill, they won't have the same skills because the values are different even though they mean the same. There were three approaches to pre-process the data.

Approach 1: Use available taxonomies

The company has created a multitude of taxonomies that map similar meaning values in the profiles to a identifying value. For instance, `powerpoint` and `microsoft powerpoint` are mapped to the `presentation and`

`visualisation (powerpoint)` skill. The strength of pre-processing can be seen in Figure 9 where the number of unique original skills has been condensed by 99,5% (!), meaning there's much higher chance people have similar skills.

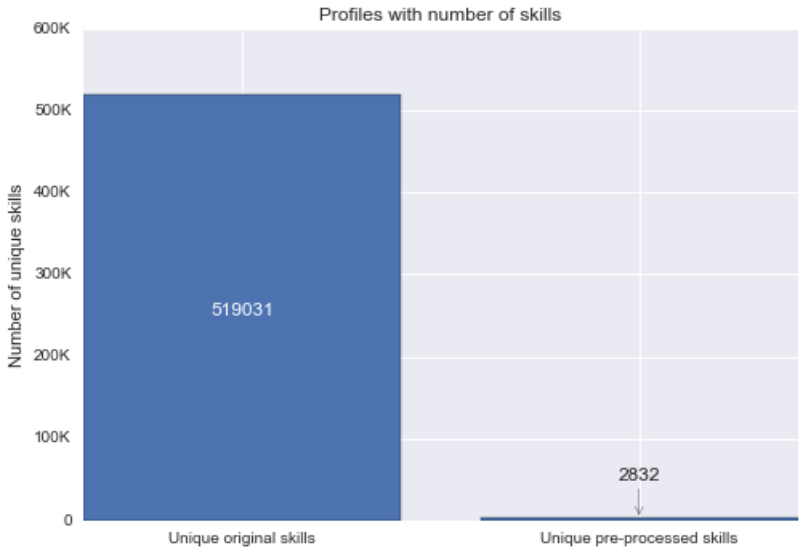

Figure 9 - Number of unique original skills versus number of unique pre-processed skills in one million profiles.

Not all the taxonomies are useful. Only the taxonomies that had something to do with the useful data fields were chosen, which can be seen in Table 8.

Taxonomy	CR (%)	Description
Education_type	4.75%	Contains types that specify all the expected specifications of all the educations' majors in English, Dutch and German.
Function_type	7.32%	Contains types that categorize similar-meaning function names together, specified in English, Dutch and German.
Industry_category	100%	Contains categories that categorize broadly similar industry types together, specified in English, Dutch and German.
Skills_&_competencies	63.8%	Contains all the expected specifications of skills, specified in English, Dutch and German.

Great	Good	Average	Bad	Terrible

Table 8 - List of most useful taxonomies. CR means Coverage Rate.

The coverage rate (CR) of taxonomies on the profile data varies from taxonomy to taxonomy. CR stands for the percentage of user-specified data that can be translated to a similar value specified in the taxonomy. The CR was measured based on the profile data in one million profiles.

The CRs of educations and functions are pretty bad, which means solutions must be found to increase them because only 4.75% of the educations and 7.32% of the functions can be detected with the taxonomy and thus have meaning. Due to time restrictions, only a solution for the educations was attempted to find.

Approach 2: Create own taxonomies

An education is defined by the combination of the educational degree and major (Ziebarth, Incidental questions, 2016). This means two taxonomies had to be created: one for the degrees and one for the majors. The degree taxonomy would be created first. To create it, a set of user-specified degrees was retrieved from the one million profiles that occur at least 20 times. If something occurs at least 20 times, it means that data is statistically relevant (Keuren, Incidental questions, 2016). Every degree in this massive set of user-specified degrees is checked manually and an appropriate actual degree is assigned to it. Both Google and 8vance's data science experts were used as the sources to find the appropriate degrees.

After approximately 2 weeks, roughly 1875 user-specified degrees were mapped to actual degrees (and to some majors as well, see Table 9). The taxonomy covered about 17% of the user-specified degrees. Mapping the other 83% would take too long, and a taxonomy of majors also still had to be created. This meant another solution had to be found that would speed up the process. With this taxonomy, the CR of educations already impressively increased from 4.75% to 41.8%.

Degree	Major	User-specified degree
bachelor	-	bachelor with honours|bs.c|bachelor|bs.c. |
bachelor	networks and communications	bachelor of communications|ba communications|

Table 9 - Snippet of the education taxonomy. The | symbols serve as separators. As the user-specified degrees can contain information about the degree and major, both the degree and major can be extracted.

Approach 3: Code-based pre-processing

The goal of this third approach was to chain text pre-processing methods together and use it to pre-process the degrees and majors. The pre-processing chain can be seen in Figure 10. The purpose of this method is to remove as much useless information as possible and maximize the chances of a match between pre-processed degrees and majors. The pre-processed degree and major are joined together which defines an education.

However, this method didn't prove to be useful because it ended up with more than 30,000 unique combinations of pre-processed degrees and majors and an underwhelming amount of matches. The main problem with this approach is that it can't detect a similarity of words that are spelled differently, but mean the same thing.

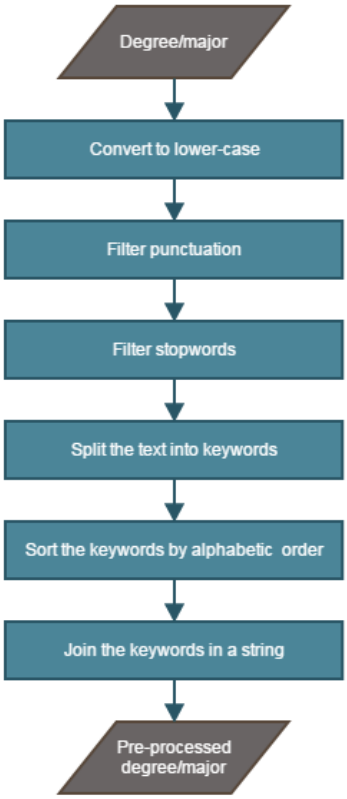

Figure 10 - Pre-processing chain to pre-process degrees and majors.

Approach 4: Algorithm-based pre-processing

This approach aims to solve the problem of the previous approach: to detect similar-meaning words that are spelled differently and translate them to one word. Word2vec was the algorithm of choice to find and detect similar-meaning words based on their semantic similarity. (Google, 2013) (Mikolov, Chen, Corrado, & Dean, 2013) The Word2vec algorithm works as follows: (Google, 2013)

1. It takes a text corpus as input and produces the word vectors as output.
2. It constructs a vocabulary from the text data and learns vector representation of words.
3. The resulting vector representations can be used to find distances to other vector representations in order to find the closest similar words.

The Word2vec algorithm was trained on user-specified degrees and majors of five million profiles. The degrees and majors were converted to lower case and stop words and punctuation were filtered (see Figure 11).

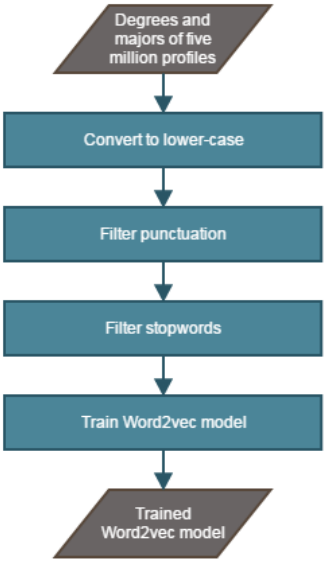

Figure 11 - Process of training the Word2vec model.

The next step is to use the trained Word2vec model to translate similar-meaning words to one word. Firstly, the words in the user-specified majors were translated. This process can be seen in Figure 12.

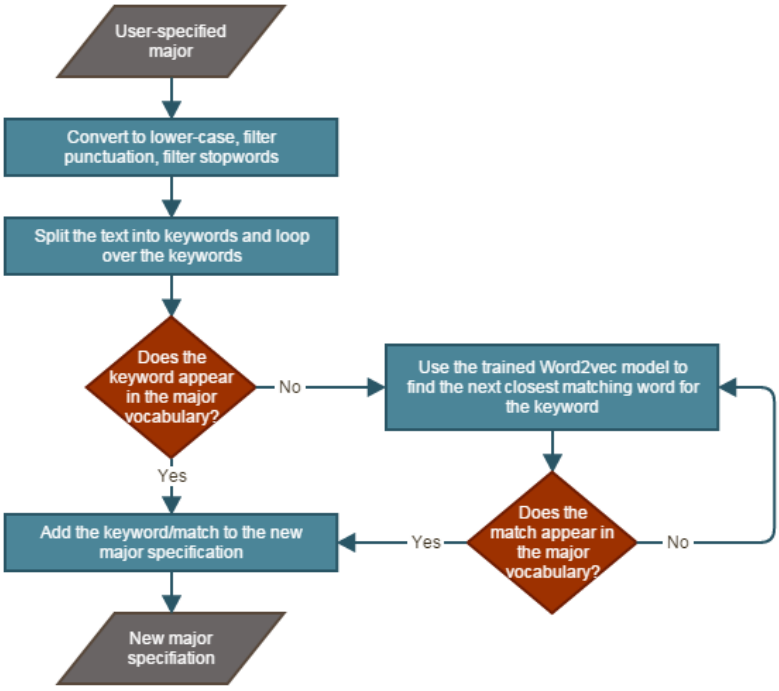

Figure 12 - Process of translating keywords in user-specified majors to keywords that exist in the major vocabulary.

The major vocabulary is a list of keywords which are extracted from the majors specified in 8vance's education taxonomy. So if a user-specified major contains keywords that don't occur in any major specification in this

taxonomy, these keywords are translated to the closest matching keywords that do appear in the major specifications.

Now that the majors have been translated, a total of eight potential solutions were tested to try and predict the closest matching major to the translated user-specified major. Table 10 contains an overview of each solution's quality, which was determined by comparing the predicted closest matching major of the user-specified major with the expected major. The seventh solution appeared to be the best performing solution of them all. Refer to Appendix **B** to find more information about the compared algorithms.

User-specified major	Tfidf weighting n=3 score > 0.7	Word2vec n=3 score > 0.7	Word2vec n=2 score > 0.7	Word2vec	Tfidf	Levenshtein	Pipeline count vectorizer, tfidf, mulinomialnb	Pipeline count vectorizer, tfidf, sgdclassifier
Architecture								
Technique traduction économique éditoriale								
Professionnelle traduction								
Translation								
Computer science								
Accounting business management								
Computer general information sciences								
AVERAGE QUALITY								

Excellent | Good | Mediocre | Bad

Table 10 - An overview of the quality of major predictions based on the user-specified major. Eight different combinations of algorithms and parameters were tested.

Now that the user-specified majors can be predicted to actual majors, the CR of educations almost doubled, from 41.8% to 79.8%. Compared to the initial situation where only 4.75% of the educations could be used for predicting

skills, now 79.8% of the educations can be used. However, the predictions sometimes aren't that good (see Table 10) which could mean a major has wrongfully been assigned to a profile. This is the risk of this solution.

Unfortunately due to a lack of time, the same calculations couldn't be done for the user-specified degree. The approach to find the closest matching degree to the user-specified degree would be similar to the major's approach discussed in this section. These calculations could also be useful for other data fields such as the functions.

Missing skill predictions

Now that a large portion of the data is pre-processed and ready to be used, solutions to predict missing skills could be found and tested with the data. A lot of different solutions have been tested which didn't yield good enough results, such as:

- Calculate the closest matching profiles to a profile based on the profile data, and return the most occurring skills in the matching profiles as the skill prediction. One of the reasons this solution didn't work is because the best matches were almost always based on the similarity in skills. This meant the best matching profiles only had approximately 0-10 other skills that could be predicted for a profile. Most of these other skills often only occurred once in one of these matching profiles, meaning it would be unlikely the profile these skills were predicted for would actually have these skills (flawed prediction). If these low-occurring skills would be filtered, only 0-3 skills would remain.

The better approach proved to be a statistical calculation that measures the likelihood of a profile owning a particular skill based on the occurrences of a skill in regards to all the variations in profile data. Let's have a look at an example to show it works.

Example

One of the profiles used to test the skill prediction solutions is the one that can be seen in Table 11. This profile was chosen because it appears in the IT industry. The quality of the skill prediction can be better assessed in this industry due to the (developer's) broader knowledge in the IT domain.

Industry	Program development	
Educations	**Education #1**	
	Degree:	Not-specified
	Major:	Neural computing
	Major_certainty:	0.865409523249
	Education #2	
	Degree:	Bachelor
	Major:	Neural computing
	Major_certainty:	0.69803828001
Experiences	**Experience #1**	

	Function:	Software developer
	Sector:	ICT-AV
	Level:	5
Skills	Java, Spring framework, Object-relational mapping, Javascript, Jquery, Ajax, SQL, MySQL, CSS, PHP, Web development, HTML, HTML5, C# - iso/iec 23270, Web design, Windws	

Table 11 - Simplified overview of one of the test profiles for testing skill prediction solutions. The Major_certainty data field indicates the level of certainty of the person having the specified major (from 0 to 1).This score was calculated with the pre-processing solution discussed in the previous section.

To predict the missing skills for this profile, the idea was to calculate the occurrences of skills in regards to the data fields specified in Table 11. So for instance, this profile has specified *Program development* as the industry. All skills that are related to this industry are retrieved and the likelihood of the profile having any of those skills is measured. There were two different approaches to measure the likelihood score per skill.

Approach 1: Tf-idf weighting

The first likelihood score was measured based on the Tf-idf weighting formula, recommended by 8vance's data science expert Sabrina Ziebarth. Tf-idf is a statistical measure used to evaluate how important/identifying a source value is to another target value. The importance increases proportionally to the number of times the source value appears with the target value, and is offset by the frequency of the source value appearing with all of the target values. (Tfidf, 2016) The Tf-idf weight was measured for 11 data field combinations with regards to the skill data field, specified in Table 12. These data field combinations were expected to yield the best results in predicting the missing skills.

Combination ID	Source value	Target value(s)
1	Skill	Industry
2	Skill	Education (degree + major)
3	Skill	Experience (function)
4	Skill	Experience (sector)
5	Skill	Experience (level)
6	Skill	Skill
7	Skill	Industry & Education (degree + major)
8	Skill	Industry & Experience (function)
9	Skill	Industry & Experience (sector)
10	Skill	Industry & Experience (level)

11	Skill	Industry & Skill

Table 12 - Data field combinations to measure Tf-idf weighting. The Skill-Skill combination measures the weighting based on a skill appearing with other skills. The Skill-Industry & Education (degree + major) combination measures the weighting based on a skill appearing with both an industry and education (an education is specified by the combination of a degree and major).

The combinations are specified with a limit of two target values. It wasn't useful to create deeper combinations with three or more target values because it would create too small groups. For instance, there aren't enough profiles that have the same industry, education and experience. However, having deeper combinations would result in even better results. (Jacobs, 2016)

After all the weights are calculated for all the combinations specified in Table 12 for the one million profiles, the Tf-idf weight of all 2832 skills can be retrieved from these combinations and added together to get a final weight for all the skills:

$$S_{p,s} = C1_{p,s} + C2_{p,s} + C3_{p,s} + C4_{p,s} + C5_{p,s} + C6_{p,s} + C7_{p,s} + C8_{p,s} + C9_{p,s} + C10_{p,s} + C11_{p,s} \text{ for } s = 0 \ldots 2831$$

$S_{p,s}$: Final skills and their relevant Tf-idf weight added together from the 11 data field combinations for a profile

$C1...11_{p,s}$: Skills and their relevant Tf-idf weight based on the relevant data field combination for a profile

The skills with the highest weights are the skills the algorithm is most certain about the profile has. However, there isn't a certainty score to determine the exact certainty, which is a big downside. Table 13 contains the top 10 predicted skills for the example profile for this solution.

Predicted skill	Prediction weight	Prediction quality
XML	5.512655	
Revision control systems	4.927785	
Software development	3.999552	
Agile software development	3.987702	
Microsoft SQL server	3.775231	
OOP	3.769749	
C++ - iso/iec 14882	3.701439	
.net	3.526401	
Linux	3.453740	
Web services	3.404043	

Excellent	Good	Mediocre	Bad

Table 13 - Top 10 predictions of the Tf-idf solution. The predictions are in line with the profile, meaning they are all good predictions.

As can be seen, the solution performs pretty well. However, since a certainty percentage is missing - which is one of the requirements - another solution had to be found.

Approach 2: Percentage occurrence

The second approach to calculate the likelihood score was advised by one of 8vance's data science expert acquaintances. He strongly disagreed with the first approach to work with weights, because weights don't show any form of likelihood of a person owning a skill. He said that all the domain knowledge you need to solve the missing skill problem is in the data. Calculating the probability of someone owning a particular skill is a simple, effective and likely the best solution to solve the problem. Jan Jacobs - 8vance's neural developer - also agreed with this.

The percentages were also calculated with the same data field combinations specified in Table 12, as it has proven to yield very good results. Table 14 contains the top 10 predicted skills for the example profile for this solution.

Predicted skill	Certainty score (%)	Weighting score	Prediction quality
Microsoft office	44.2501	0.012862	
XML	40.3846	0.010957	
Revision control systems	38.4615	0.007619	
Software development	28.3601	0.005580	
Project management	28.1734	0.021515	
Adobe photoshop	28.1665	0.005060	
Adobe creative suite	28.1665	0.000385	
Web applications	28.1665	0.003316	
Wordpress	28.1665	0.000133	
Polymer chemistry	28.1665	0.000272	

Excellent	Good	Mediocre	Bad

Table 14 - Top 10 predictions of the percentage solution.

As can be seen, this solution seems to perform worse. The prediction quality isn't as good as the previous solution. However, when a *Weighting score* column is added (which depicts the Tf-idf weighting of the specified skill in the

profile's *Program development* industry), you can see that the low scoring weighted skills are the bad predicted skills. When the low scoring weights have been filtered, the following result is presented:

Predicted skill	Certainty score (%)	Weighting score	Prediction quality
Microsoft office	44.2501	0.012862	
XML	40.3846	0.010957	
Revision control systems	38.4615	0.007619	
Software development	28.3601	0.005580	
Project management	28.1734	0.021515	
Adobe photoshop	28.1665	0.005060	
Web applications	28.1665	0.003316	
Management	25.3576	0.017212	
Microsoft SQL server	25.0000	0.008220	
Strategic planning	23.8390	0.018205	
Agile software development	23.4387	0.005965	

	Excellent		Good		Mediocre		Bad

Table 15 - Optimised result of the percentage solution.

The overall prediction quality is a lot better. With the added certainty score, this solution seems to be the better one of the two.

Comparison of the best found solution to the company's current solution

The company also has its own solution to predict skills. Let's have a look at the top 10 skill predictions of their solution for the same profile.

Predicted skill	Weighting score	Prediction quality
Struts	13.417104	
Gimp	10.501582	
Usability engineering	9.773890	
Open office	9.751774	
Product strategy	9.719967	

PMI	9.570196	
Entity framework	9.442973	
Relational databases	9.332178	
User research	9.297548	
Computer games	9.209280	

	Excellent		Good		Mediocre		Bad

Table 16 - Top 10 skill predictions of the company's solution.

The overall quality of predictions of this solution is on the same level as the other two discussed solutions. However, the percentage solution also adds a probability ratio of the person owning the skills, whereas a weighting score doesn't tell anything about the probability of the person owning the skills. Therefore the percentage solution has a slight edge over the other solutions.

6.6 Conclusion

Let's get back to the three research's sub-questions that would be answered in this chapter.

- Which selection of the talents' data can be used to help to determine the missing skills?

Table 7 contains the overview of the usefulness of the data fields which could be used to help determine the missing skills. The most useful data fields ended up being the *Industry*, *Experiences*, *Educations* and *Skills.*

- What are possible solutions to complement the missing skills with help of this selection of talents' data?

Before predicting the missing skills, the data needed to be pre-processed first. The data was pre-processed as follows:

1. The company's own available taxonomies is used to pre-process a large portion of the data.
2. An own taxonomy of educations was created to improve the pre-processing quality for educations.
3. The Word2vec algorithm was used to find semantic similar words based on the education's majors. Any words that didn't appear in the company's education taxonomy but that were specified in the major data field, were replaced with the semantic similar words that do appear in the company's major taxonomy.
4. The pre-processed major is translated to an actual major defined in the company's education taxonomy with help of a pipeline that exists of a CountVectorizer, Tf-idf and MultinomialNB algorithm.

After this pre-processing, the skill prediction solutions could be tested. A Tf-idf algorithm and a percentage-based algorithm were the most promising solutions to predict the skills.

- Which solutions perform best (and is the company 8vance satisfied with)?

The percentage-based algorithm, Tf-idf algorithm and the company's own algorithm seemed to have a similarly good performance. However, the percentage-based algorithm had a slight edge over the others because it's the only one that can calculate a probability ratio for a person owning a particular skill.

(Start with a introduction, explaining the details of this chapter.

Topics of importance, in this order:

- *Requirements specification. Explain why and mention a range of important and unimportant requirements (for competence reasons).*
- *Software architecture document. Explain why (kruchten, ISO 25010) and refer to a simplified model of the chosen architecture (must be simple enough to understand for anyone). Also mention the architectures that were also considered. Also mention the things of the architecture I will be responsible for to implement and the things 8vance will be responsible for.*
- *Building the algorithm library. Started with it after the architecture was partially created. This library contains four major features: pre-processing the data, using an algorithm to predict skills, and post-processing the data.*
    - *Pre-processing of the data. Explain why it's crucial. This includes the degree model (interview), using existing taxonomies (interview), and normalizing the data. Also mention the problem of creating an own model and the lack of a 'major' model.*
    - *Algorithms to predict skills. Mention the algorithms that were considered and tested. Explain why one algorithm was better than the other and which one was the best. Explain how the algorithms were tested and briefly explain how the best algorithm works.*
    - *Post-processing of the data. Explain why it's crucial. Maybe I'll leave this out because there isn't much interesting to say about this because it's very similar to the pre-processing section.*
- *Building the algorithm analysis "tool". Mention the problem with the first architecture of this tool (a complete GUI application with a 3-layer architecture) and why this architecture wasn't necessary. Explain the use of this tool and why it's important.*

End with a conclusion, briefly mentioning the results of the topics and the next steps to take. Important questions to answer are:

- *Does the found solution solve the problem?*
- *Is the found solution satisfactory to the company?)*

7 Completion phase

(Completion phase might not be the best translation of "invoeringsfase", but it will do for now.

Start with a introduction, explaining the details of this chapter.

Topics of importance, in this order:

- *Conclusions and recommendations of the research. Briefly mention the best found solution and its flaws. Mention recommendations for the company to get possibly better results.*
- *Finalization of the architecture document. (especially the deployment view which isn't created yet) What part of the architecture is and isn't implemented?*
- *List of implemented, partially implemented and not implemented requirements.*

End with a conclusion, briefly mentioning the results of the topics. A conclusion may be unnecessary for this chapter though.)

8 Conclusion and recommendations

(The reader must be capable to understand this chapter without reading any of the intermediate chapters. This chapter cannot suddenly introduce new information out of nowhere.

Determine whether or not the project has been successful by reviewing the found solutions. Mention recommendations to improve the solutions.)

Evaluation

(Use of "I" is mandatory in this chapter.

Reflect on your own work process and experiences. Describe what I've learned, what I enjoyed, and what were the most important learning moments. Emphasize especially how I solved my mistakes.

Also mention and reflect on my personal learning goals.)

Bibliography

8vance Matching Technologies. (2016). *How it works*. Retrieved May 17, 2016, from 8vance: https://www.8vance.com/howitworks/

Crabtree, S. (2013, October 8). *Worldwide, 13% of Employees Are Engaged at Work*. Retrieved May 18, 2016, from Gallup: http://www.gallup.com/poll/165269/worldwide-employees-engaged-work.aspx

Dutcher, J. (2014, September 3). *What is Big Data?* Retrieved June 2, 2016, from Berkeley school of information: https://datascience.berkeley.edu/what-is-big-data/

Frey, C. B., & Osborne, M. A. (2013, September 17). *The future of employment: How susceptible are jobs to computerisation?* Retrieved June 2, 2016, from oxfordmartin: http://www.oxfordmartin.ox.ac.uk/downloads/academic/The_Future_of_Employment.pdf

Frey, T. (2012, February 3). *2 Billion jobs to disappear by 2030*. Retrieved June 2, 2016, from FuturistSpeaker: http://www.futuristspeaker.com/business-trends/2-billion-jobs-to-disappear-by-2030/

Google. (2013, July 30). *Word2vec*. Retrieved June 3, 2016, from Google: https://code.google.com/archive/p/word2vec/

Guo, P. J. (2013, July). *First impressions of the IPython notebook*. Retrieved June 2, 2016, from pgbovine: http://pgbovine.net/ipython-notebook-first-impressions.htm

Hutt, R. (2016, January 21). *What are the 10 biggest global challenges?* Retrieved May 8, 2016, from World Economic Forum: https://www.weforum.org/agenda/2016/01/what-are-the-10-biggest-global-challenges/

ISO 25000. (2015). *ISO/IEC 25010*. Retrieved June 1, 2016, from ISO 25000: http://iso25000.com/index.php/en/iso-25000-standards/iso-25010

Jacobs, J. (2016). Incidental questions. (T. Hermens, Interviewer)

Jeevan, M. (2015, September 4). *How I chose the right programming language for data science*. Retrieved June 2, 2016, from Big data made simple: http://bigdata-madesimple.com/how-i-chose-the-right-programming-language-for-data-science/

Kempen, P., & Bennink, H. (2016). *Competent afstuderen met het Tien Stappen Plan.* Retrieved March 7, 2016, from Noordhoff: http://hoadd.noordhoff.nl/sites/7745/_assets/7046d91.pdf

Keuren, P. (2016). Incidental questions. (T. Hermens, Interviewer)

Keuren, P. (2016). Incidental questions. (T. Hermens, Interviewer)

Kickstart Venlo. (2016). *Huisvesting*. Retrieved May 17, 2016, from kickstartvenlo: https://www.kickstartvenlo.nl/huisvesting/

Kruchten, P. B. (1995, November). *The 4+1 View Model of Architecture.* Retrieved June 2, 2016, from ics: http://www.ics.uci.edu/~andre/ics223w2006/kruchten3.pdf

Langley, M. A. (2016). *The High Cost of Low Performance.* Retrieved May 23, 2016, from pmi: http://www.pmi.org/~/media/PDF/learning/pulse-of-the-profession-2016.ashx

March, D. L. (2003, April 11). *Pipe-and-filter style*. Retrieved June 1, 2016, from Desales: http://www4.desales.edu/~dlm1/it533/class03/pipe.html

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013, September 7). *Efficient estimation of word representations in vector space.* Retrieved June 3, 2016, from Arxiv: http://arxiv.org/pdf/1301.3781.pdf

Moran, D. (2010, April 27). *Top 10 Reasons to Use Agile Development*. Retrieved May 23, 2016, from devx: http://www.devx.com/enterprise/Article/44619

Open Universiteit. (2016). *Architectural patterns.* Retrieved June 1, 2016, from OU: http://portal.ou.nl/documents/114964/2986739/IM0203_03.pdf

Progress Sitefinity CMS Documentation. (2016). *Hierarchical taxonomies*. Retrieved June 2, 2016, from Sitefinity: http://docs.sitefinity.com/for-developers-hierarchical-taxonomies

Sanders, B. A., Massingill, B. L., & Mattson, T. G. (2005, March 11). *The algorithm structure design space in parallel programming*. Retrieved June 1, 2016, from Informit: http://www.informit.com/articles/article.aspx?p=366887&seqNum=8

Stoffels, H. (2016). Incidental questions. (T. Hermens, Interviewer)

Tfidf. (2016). *What does tf-idf mean?* Retrieved June 6, 2016, from Tfidf: http://www.tfidf.com/

Tutorialspoint. (2016). *Interaction-oriented architecture*. Retrieved June 1, 2016, from Tutorialspoint: http://www.tutorialspoint.com/software_architecture_design/interaction_oriented_architecture.htm

Van Turnhout, K., Bennis, A., Craenmehr, S., Holwerda, R., Jacobs, M., Niels, R., et al. (2014, October). *Design patterns for mixed-method research in HCI.* Retrieved June 2, 2016, from ralphniels: http://www.ralphniels.nl/pubs/turnhout-designpatterns-nordichi14.pdf

Vonk, M. (2015, October 13). *Sjerlok spoort gestolen goederen op voor Delta Lloyd*. Retrieved May 17, 2016, from amweb: http://amweb.nl/branche-724322/sjerlok-spoort-gestolen-goederen-op-voor-delta-lloyd

Waters, K. (2007, June 11). *10 Good Reasons To Do Agile Development*. Retrieved May 23, 2016, from allaboutagile: http://www.allaboutagile.com/10-good-reasons-to-do-agile-development/

Wikipedia. (2016, May 22). *User requirement document*. Retrieved June 1, 2016, from Wikipedia: https://en.wikipedia.org/wiki/User_requirements_document

Ziebarth, S. (2016). Incidental questions. (T. Hermens, Interviewer)

Ziebarth, S. (2016). *people site for scrape2-1.* Venlo.

Appendices

A. Project Initiation Document (PID)

(Contains the whole PID document.)

B. Major prediction algorithms

(Contains information about the algorithms that were tested to predict the closest matching major to a user-specified major)

C. Organisation chart

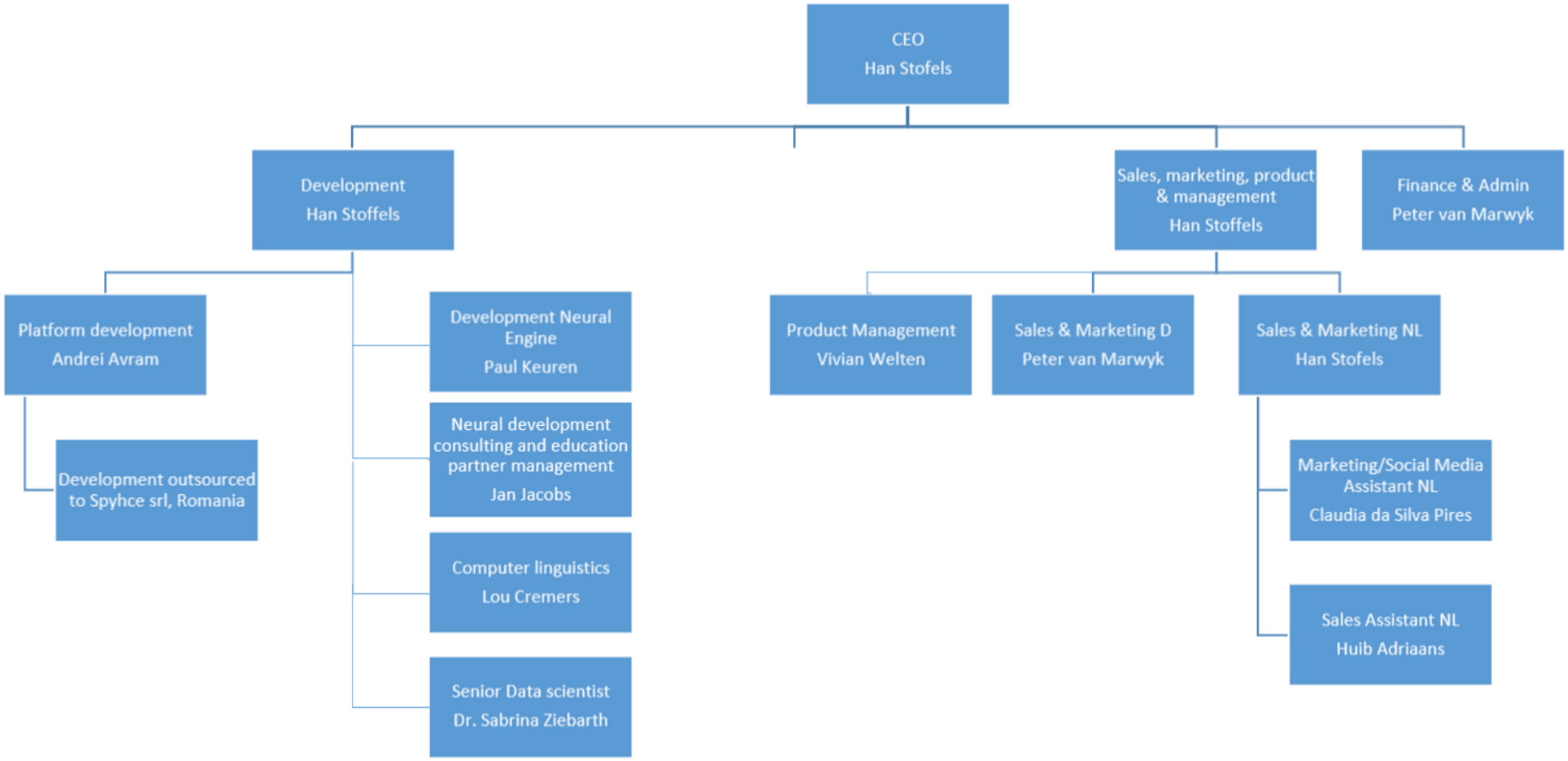

Figure 13 - Organisation chart of the company 8vance.

D. Other appendices

(This section is just a placeholder for other appendices. Only add documents or preferentially parts of documents if they offer relevant information for the reader.)