Insert here your thesis' task.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Master's thesis

A Case Study and Proof of Concept of the Application of Machine Learning to Polarion's ALM Software

Bc. Michal Sláma

Katedra . . . softwarového inženýrství
Supervisor: Ing. Jurij Černikov

May 19, 2018

Acknowledgements

I would like to thank to my supervisour for his extraordinary leading and valuable advices during the whole process of writing this thesis.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended. In accordance with Article 46(6) of the Act, I hereby grant a nonexclusive authorization (license) to utilize this thesis, including any and all computer programs incorporated therein or attached thereto and all corresponding documentation (hereinafter collectively referred to as the "Work"), to any and all persons that wish to utilize the Work. Such persons are entitled to use the Work in any way (including for-profit purposes) that does not detract from its value. This authorization is not limited in terms of time, location and quantity. However, all persons that makes use of the above license shall be obliged to grant a license at least in the same scope as defined above with respect to each and every work that is created (wholly or in part) based on the Work, by modifying the Work, by combining the Work with another work, by including the Work in a collection of works or by adapting the Work (including translation), and at the same time make available the source code of such work at least in a way and scope that are comparable to the way and scope in which the source code of the Work is made available.

In V Praze on May 19, 2018 .....

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis

Sláma, Michal. *A Case Study and Proof of Concept of the Application of Machine Learning to Polarion's ALM Software*. Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2018.

Abstrakt

Machine learning (ML) is becaming essencial part any software application. Its same for application lifecycle management (ALM) which hides great opportunities to improve using, processing or behaviour of the whole system based on the ML principies. This work cointains description of ML principies relevat for using in the ALM environment. For the specific software is used Polarion which is wordwide successful enterprise solution for ALM. This thesis provides analysis it's core business and user cases and posible ways how to integrate ML to improve Polarion in different areas. As Polarion must that customer's data are not exposed to any possible thread ensure on all levels we will discuss the way how to achieve this goal by a different kind of architecture or implementation.

Klíčová slova Strojové učení, životní cyklus softwarových aplikací, ALM

Abstract

Sem doplňte ekvivalent abstraktu Vaší práce v angličtině.

Keywords Machine learning, application lifecycle management, ALM

Contents

List of Figures

Todo list

Introduction

"The revolution is just beginning, but it's real – and the time to act is now. In fact, it is yours for the taking to harness a broad platform, services and ecosystem to transform your business. A unified approach to application lifecycle management is not a futuristic technology trend. It's here today,and the good news is that you don't have to completely stop and reset, but can smoothly transition from squeezing the most out of your existing business processes to making your organization thrive."

Kurt Bittner
Analyst
Forrester Research

Author of this thesis has been working in software development for more than 10 years. Based on this experiences he got to a question how to manage and keep up to date info for large project where more than dozens people are involved. This is also why he chooses to work as a developer of Polarion.

We live in the world that is changing rapidly. No part of life of free to this changes and the software development needs more then others to adapt every day to new requirements and technologies. This demand leads to a new level of management for software development where tools, process, implementation, testing and reporting are organized on one place with goal to keep and improve traceability and productivity as high as possible. This comes hand by hand with automation in the form of ML that moves user experience and reporting to the next level. One of such product is Polarion[1] and this work will analyze it's user cases and find out what places are good candidates for using ML techniques to improve business value of the product.

Chapter 1

The aim of the thesis

The aim of this thesis is to analyze and identify machine learning (ML) use cases that would prove valuable for Polarion's application lifecycle management (ALM) software. A proof of concept prototype will be supplied for the selected use case.

1. Analyze and describe Polarion in order to identify suitable use cases to apply ML to.
2. Provide a review of ML frameworks and algorithms that are relevant for such an application.
3. Describe several use cases for ML and define their benefit to both ALM as a business and the users that deploy it.
4. Choose a scenario from the previous investigation and implement a proof of concept prototype.
5. Discuss the possibility of the full implementation and deployment of the previous prototype into the production environment.

Chapter 2

Polarion

Organizations are often struggling with the old processes of doing things. They focus on isolated process optimization instead of driving business value through comprehensive synchronization. With Polarion, customers have been able to get their teams out of their silos and orchestrate development efforts across the entire application lifecycle. This approach has empowered stakeholders to better perform tasks in context and quickly make sound decisions based on real-time access to information.

You can try Polarion on `https://polarion.plm.automation.siemens.com/`.

Now let's take a look what Polarion comes with and how improves old fashion processes.

2.1 ALM

"ALM is a paradox in the software engineering world, where engineers recognize the need for requirements management, change and configuration management, QA and test management, and so on, but are not familiar with the term ALM. This is a serious problem because ALM is necessary to manage software complexity, and the rise of embedded software in engineered products needs mature management processes and tools."
Michael Azoff
Principal Analyst

Polarion is a ALM enterprise solution to deal with modern-day challenges. It has emerged with the intent to fasttrack innovation, while safeguarding quality, functional safety and compliance to satisfy that speed in developing and delivering innovative applications is becoming essential to the success of businesses in any industry.

ALM points to these three aspects:

1. application has to be delivered as fast as possible and the time is a new strategy weapon
2. information technologies are fuel accelerating business success
3. errors are not forgiven and can go viral in instant. QA rules must be set and obeyed.

2.2 A unified solution

Polarion is a single solution providing a solution to build whole application from the ground. At the same time it is ensured that data and logic is in persistent state during entire process.

This helps with regulations. This basic word means a lot in the world of software development where processes are regulated by intern or external subject and have essential impact on the cost of product not only during development but after it's release.

What ALM comes with?

The main advantages attributed to ALM process (graphically shown in the picture 2.1):

- Agility through improved collaboration.
- Productivity through process integration.
- Auditability through traceability and accountability.
- Quality through transparency and automation.
- Innovation through unlocked team synergy.
- Predictability through better estimation and reporting.

Let's a look more close to some of these advantage.

2.2.1 Agility through improved collaboration

If faster time-to-market is a key success factor in today's competitive environment, real-time collaboration and contextual performance of tasks are the means to stay ahead. In many cases, lightweight Agile software development methods have replaced or augmented incremental waterfall methods to release products more frequently.

Polarion provides flexible support for Agile or Lean, as well as traditional and hybrid environments, including any customized Scrum, feature-driven development, Kanban, extreme programming, or rational unified process methodologies.

Figure 2.1: A unified solution for ALM [1]

The Polarion 100 percent browser-based architecture makes information universally accessible from anywhere for any collaborator. Collaboration is so easy and teams divided around the world can communicate to each other and solve tasks together.

2.2.2 Productivity through process integration

Major part of Polarion customers apply a combination of Agile and DevOps methodologies The Polarion solution is the perfect conduit to DevOps, allowing easy synchronization of development and delivery processes spanning requirements definition, feature development, quality testing, and maintenance. Any problem can be easily tracked back to the source and time of maintenance is so rapidly reduced even to real time fixes.

Polarion supports integration with other tools. This is done by extension or native integration. Thanks this customers can still use their own tools and data repositories and just integrate them with Polarion. Polarion has it's place on market for many years and during this time many extensions were created by customers or professional services and placed to official Polarion's marketplace from which can be freely downloaded. Common customer will find there with high probability a solution for his needs.

2.2.3 Auditability through traceability and accountability

"We chose Polarion ALM at Phoenix Contact in the Business Unit Automation to consolidate our very heterogeneous tool landscape – PVCS, Bugzilla, OneTree. With Polarion ALM we achieved transparency on all levels of development and we got fast acceptance in the teams. We now see exactly and in detail the status and the progress in our projects in the different project phases."
Andreas Deuter
Phoenix Contact Electronics

Every change is stored. Polarion stores all you need to track down what, how and by whom happened. In enterprise environment where is working hundreds and hundreds people it's hard to keep in touch who does what. A place where you see it all is priceless and helps to minimize risks of black holes when tasks is left or forgotten without any notice. Such a missing task can have very serious impact on cost or even release date itself.

2.2.4 Quality through transparency and automation

A big problem with this approach is that team members usually get the information about what they need to accomplish from static documents that tend to go out-of-date as quickly as they were created. But perhaps worst of all, changes and ad hoc decisions often fail to take into account the downstream impact.

Processes of Polarion provide way to track all requirement changes and so help to keep in touch with actual state of what needs to be done. For instance if product owner change a task developer is informed and can react by accepting this task or request more information about this change. In both cases every side knows what happened and what comes next.

Time when all operations were done only by people is over and automation plays important role in IT management. Polarion provides environment to run automatic jobs to build, test, check, ... or whatever customer may need. Customers are free to implements their own job and run them in the same way as native ones.

2.2.5 Automating proof of compliance

On the most important aspect is document workflow and will need this further in thesis.

You can understand to document as a normal word document but each task (in Polarion called work item) is external object. That means document is composed from its own text (headings, descriptions...) and work items that also have its description which is shown in the document. All theses together makes customer feel that is working with one unit. If customer wants to edit work item this can be done on document or via work item view easily accessible from document or another part of Polarion as working with work items is the main activity and processes are built on them. Let's look at the workflow shown in 2.2.

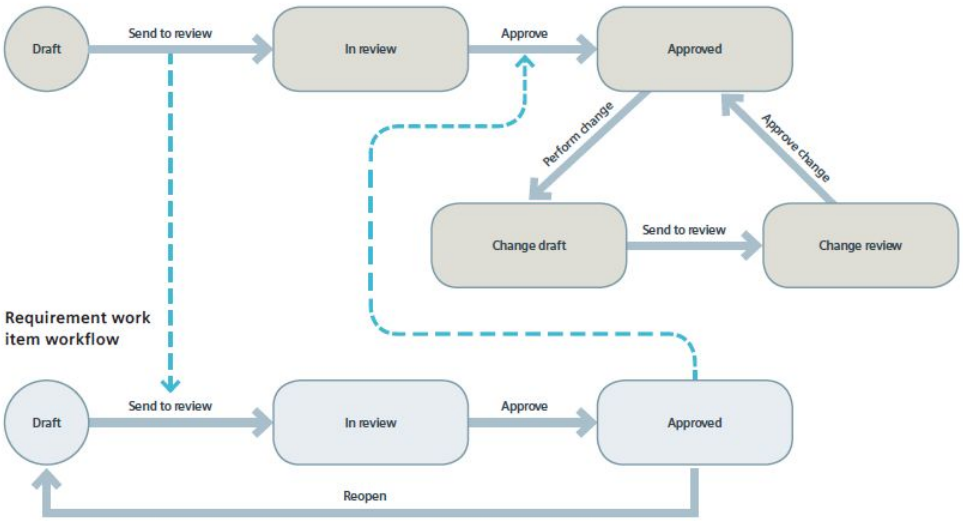

Figure 2.2: Document workflow [1]

Picture **??** shows two separated workflows. One for document and one for work items. Workflows itself does not need explanation but the important part is the relation between them. Document can not move to a next state unless all it's work items are also moved to a desired state. This keep traceability and collaboration consistence by pushing team to accept it's work and remove not desired drafts, fakes and other stuff that is not related to real work.

2.2.6 Innovation through unlocked team synergy

We live in information age. Information are all around us and the hardest part is to find what is important for us or company and what can be forgotten.

Team is a great way how to share this information but what if team has many people? Polarion contains way how to cooperate and improve your know how as fast possible and share it with other co-workers. Time to solve already solved issued is rapidly reduced and new task can be done based on results of previous work that leads to more accurate estimations and risk assessment.

2.3 Development process in complex or regulated environments

Most collaborators don't have the unified tools environment necessary to get them on the same page at the same time, however, and resulting disconnects have increasingly negative impact, disrupting industries with new records of regulatory warning letters, product failures, recalls, legal sanctions, loss in market position and associated cost explosions.

Topping the list of challenges in the new world of software driven innovation are the need for tight orchestration across disparate teams, growing regulatory demands and the increasing role of suppliers as innovation partners. Companies that are able to shift gears to meet the growing complexity will be well positioned to secure new market opportunities.

Common first step is to welcome agile approach. Companies have different expectations about it's benefits shows in the picture below 2.3.

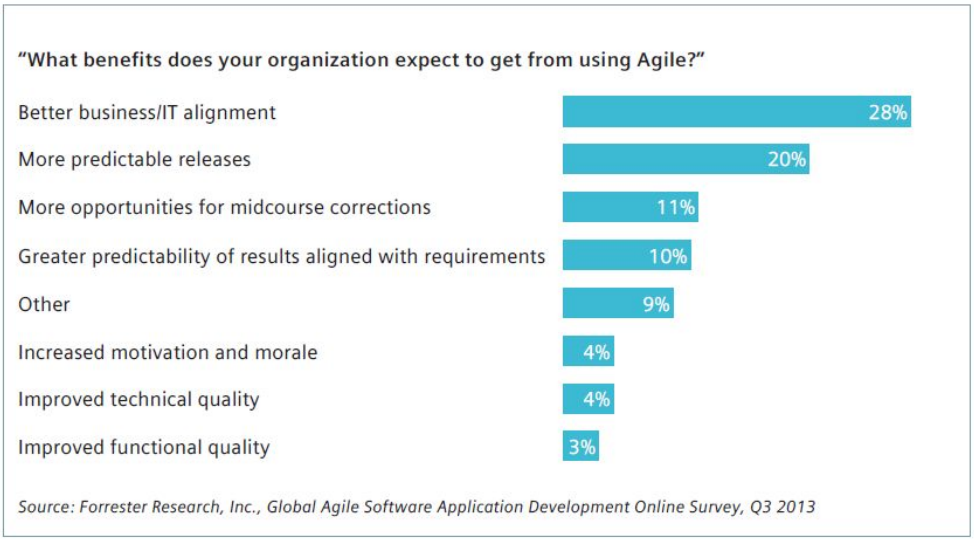

Figure 2.3: Companies expectation from using Agile[1]

Result is clear. Companies feel that IT is starting to be more and more complex and demands from business is harder to satisfy. Polarion support all

agile methods and is ready to help company improves its environment. More importantly this agile templates (how are called) can be modified based on customers demands to fit their needs and specify requirements.

"Siemens PLM Software's Polarion products presents the opportunity to allocate our complex and formal development rules via one state-of-theart tool. The modularity and flexibility make the adjustment to our needs simple and effective. The traceability and workflow features are convincing and really assist the everyday activities."
Christian Kettl
MTU Aero Engines

2.3.1 Integration of ALM and PLM

All the time we are talking about ALM but the main goal is to provide entire solution for product application lifecycle PLM where ALM is just a part of it. Where ALM is concentrated around application there PLM is the process of managing the lifecycle of a product itself from inception to final real item. Polarion is able to provide this functionality by integration with external tool that can use Polarion as the source of ALM work flow.

ALM-PLM integration benefits include:
- Integrated processes make cross-discipline synchronization very easy.
- Access to product and software requirements supports comprehensive understanding of the product definition.
- Bi-directional linking enables cross-discipline lifecycle management and audit readiness.
- Change propagation and automatic notification enable comprehensive change impact analysis.
- Synchronized testing and reporting support cross-functional defect management.
- Linked, versioned data architecture without data duplication delivers closed-loop decision making.
- Integration makes holistic compliance reporting for every aspect of the manufacturing process a reality.

2.4 Accelerate collaboration

Development environments to synchronize team efforts have proliferated. But most of them are cobbled together, posing a wide range of disadvantages. Leveraging Polarion flexibility, customers can choose from different configurations to provide all collaborators with the level of information and functionality they need, while keeping the total cost of ownership the lowest in the industry.

These are the most common:

- Difficulty linking and tracing artifacts across differently structured repositories.
- Problems of low visibility into project status, impact of changes and release predictability.
- Lack of a cohesive feedback loop that brings important context to every stakeholder.

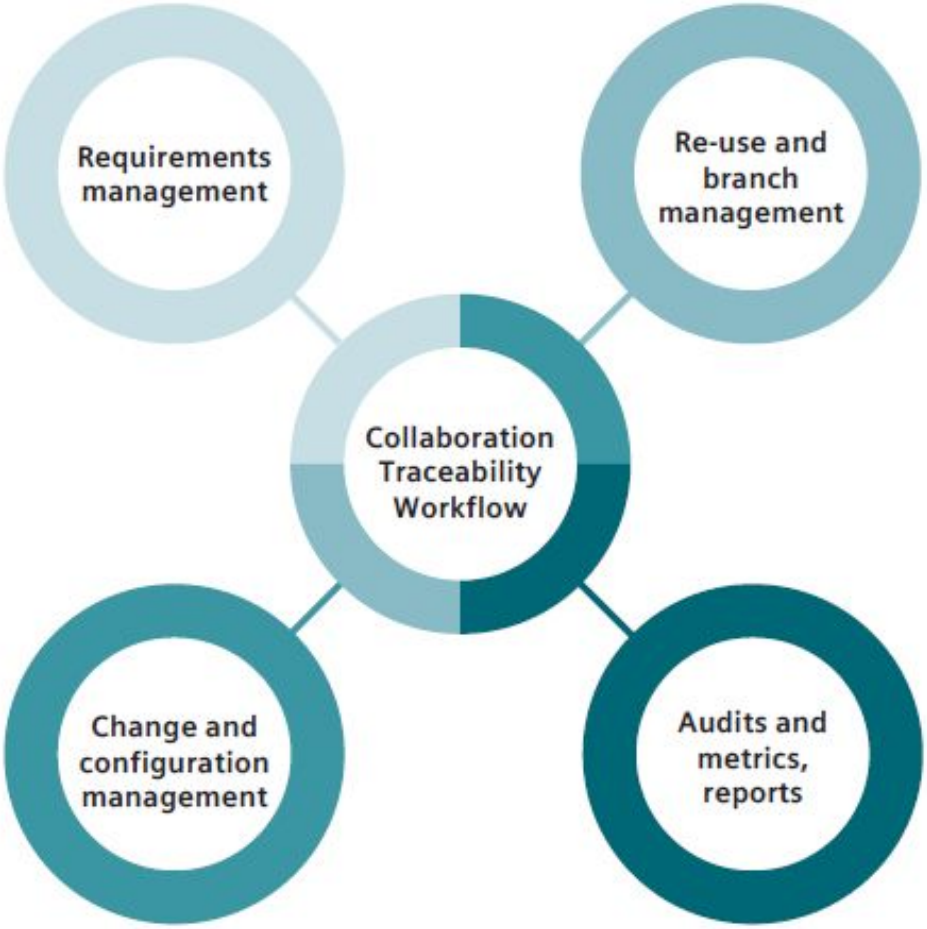

Figure 2.4: Collaboration traceability workflow [1]

2.4.1 The dilemma of requirements documentation

We need to be sure that all side speaks with dialect that respect specify domain. This requirement typically encompass varying pieces of content, including:

- Paragraphs to provide overviews and explain details.

- Lists and tables to detail structured data and rules.
- Images and models to illustrate requirements.
- Flow charts to describe a series of events.

2.4.2 "easy-as-Word functionality"

If we spoke that the base object in Polarion is Document we shall expect that customers will use document in the same way as if they were using in the Microsoft Word. Fortunately behaviour of document in Polarion is very similar in both using and how looks like.

Customers can use known buttons and tools from Microsoft Word to edit and interact with document and this speed up learning curve a lot.

2.4.3 Real-time access to content

Instead of Microsoft word document is Polarion document fully online and each change is immediately visible to everyone. Many users can simultaneously edit one item and Polarion then handle merging of changes. Of course that this can lead to conflict and in this case user have to handle his changes by own.

Consequently, companies that use Polarion Requirements are no longer forced to rely on meetings, sending emails, or circulating formal documents to make decisions, even with their partners and other external collaborators.

2.4.4 Tie in domain experts with their tools

As was said previously Polarion is expandable. Extensions can change or improve some existing functions or add completely new, transfer data from or to Polarion or connect it to external tools providing new functionality.

To complete the picture, connectors for popular third-party tools such as HP® Quality Center® and Atlassian® Jira® are available, and so is an open and fully documented Java API. As a result, a strong community of more than 100,000 members has formed and created extensions, integrations and customizations.

2.4.5 Deliver release predictability

Because every artifact change in the Polarion product is tracked and reported using the underlying configuration management system, customers automatically gain a complete audit trail of who did what, when and why, making it impossible to change anything without leaving a trace.

"Visual Diff" functionality is available to easily detect the changes between different states, and customers report that teams that take advantage of change management and impact analysis are much more successful.

2.4.6 Reduce time-to-marker

All previous parts have one main purpose to deliver product in the shortest time but ensure its quality and traceability.

2.5 Medical domain

One the strongest part of Polarion is in ability to satisfy regulation demands from various types of institutions that making development more complex then before.

Lets talk about medical domain.

Medical device product development work is a highly integrated and regulated process. Two key standards incorporated into medical device risk management are International Organization for Standardization (ISO) 14971:2009, which specifies the process for a manufacturer to identify the hazards associated with medical devices; and ISO Technical Information Report (TIR) 24971:2013, which provides guidance in addressing specific areas of ISO 14971 when implementing risk management. Europe has added to the mix with EN ISO 14971:2012, which is different in several important aspects, and is required if a company is selling medical devices into Europe. You can see this process in Medical risk management workflow picture 2.5.

It was just a example how problematic and regulated this domain is.

To describe how exactly is Polarion used in medical domain s beyond the scope of this thesis. But one point are data. Data are in medical domain valuable asset whose using is restricted by law and violation has serious financial and social impart. If we want to use this data we have to be use that all rules are obeyed and data are safe before abuse or stolen.

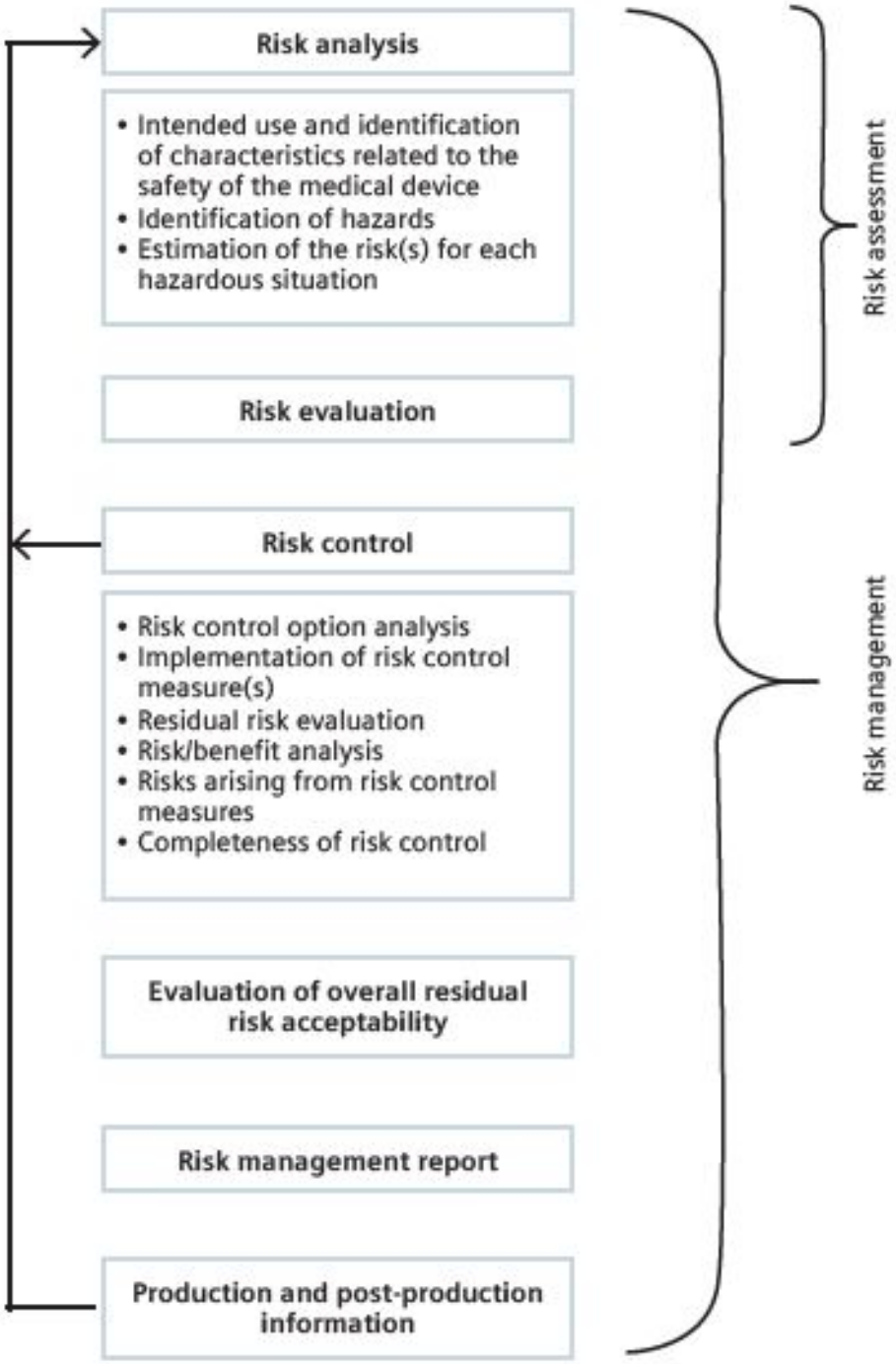

Figure 2.5: Medical risk management workflow [1]

Chapter 3

Machine learning

It's over 50 years since the first mention of ML by Arthur Lee Samuel in his study *Some Studies in Machine Learning Using the Game of Checkers*[8]. At that time it was just an idea IT was not ready to support it. But now we are fully in information age. Machines are able to calculate incredible fast and we are able to store all data what we need. From this ML has risen and companies from every corner domains have realised that ML can improve their product significantly or even create entirely new ones.

The path from just a theory to practical usage was long and hard [9].

- 1979 — Students at Stanford University invent the "Stanford Cart" which can navigate obstacles in a room on its own.
  - If you at it from today the achievement is a bit funny but on the other hand it is almost 40 years old.
- 1997 — IBM Deep Blue Beats Kasparov
  - You sure remember this event. It was also moment when author of this thesis heard about ML for the first time along side with public. Machines stopped to be perceived just like calculators and got a status of thinking think.
- 2016 — Beating Humans in Go
  - Google's AlphaGo was able to beat professional human player using a combination of machine learning and tree search techniques. What will be next?

During MLP 2018 [10] was presented a heartbreaking user case for face recognition. Try to imagine nature disaster leading to thousands and thousands refugees. Families are divided and with all that chaos around it is impossible to find each other. Depression is rising and may lead to panic or violence on himself or someone else. ML comes with simple idea. Just take a picture of yourself and it will find your relatives. How? ML has learnt on data from other families and find out what common family characters are and how can be used. Based on this ML is able to say with high probability if someone is related to someone.

What makes this examples so special? This user case uses ML in a new way but algorithms and frameworks were already here. Data was provided, model was created and it works. Sounds pretty easy and of course reality was a bit complicated but leave us a message where the current state of ML. Companies can use existing implementations without deep knowledge of complicated algorithms and may straightly start with their own problems, business case and see results in a short time.

It was said that with using of existing solutions PoF should not take more than 2 months according to learning task and processed data.

Learning can be done in different ways:

- Supervised learning: relies on data where the data are annotated and algorithm comes through them and tries to find similarities. For example, we want to distinguish between pictures of cats and dogs. Algorithm will go through on lots of annotated pictures and classify them. After it algorithm will be able correctly annotated new pictures but the main disadvantage is need of a large amount of preprocessed data.
- Semi-supervised learning: algorithm does not have any annotations. Again a large amount of data are provided to algorithm and with some characteristics what we want to find. In our case of cats and dogs we want to separate pictures into two groups. During this separation algorithm will learn rules how cats and dogs look like. Compared to the previous one we are good with raw data and no preprocessing is needed.
- Reinforcement learning: Chess play is good example of this learning when want to win. Algorithm gets set of plays with moves and results and it learns on this trying to find best moves leading to victory. In case of games this approach has a great advantage that algorithm can play with himself and play many games in a short time that is for people question of minutes or hours.

For the first Let's see what ML frameworks or libraries are available.

3.1 TensorFlow

TensorFlow[11] is an open source software library with strong support for machine learning and deep learning and the flexible numerical computation core is used across many other scientific domains. Developed by Google for its internal usage but fast became one the most used ML framework. The reason for this is the ease with which developers can build and deploy applications.

Main focused is on deep learning and has more tools to support reinforcement learning and other algorithms. Deep learning concept is shown in the figure 3.1.

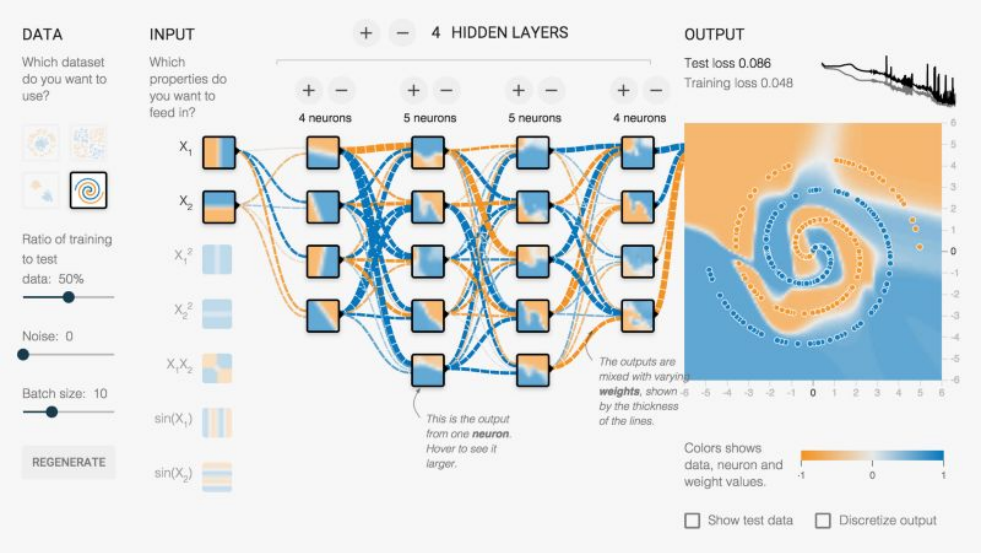

Figure 3.1: Deep learning concept of Tensorflow [2]

Tensorflow has much better performance and customizability, but the learning curve is much steeper as you are not just plugging in data and labels into a constructor, but are actually creating the layers which will make up your model. For its using you need solid understanding of machine learning and mathematical concepts especially linear algebra and calculus.

3.2 PyTorch

A Python version of Torch, known as Pytorch, was open-sourced by Facebook in January 2017. It is based on dynamic computation graphs. This has a lot of useful benefits for certain types of RNN, situations where you need to generate weights and things where the very structure of the network changes.

PyTorch[12] also has a really nice interface and has the support of Facebook. And contains lots of modular pieces that are easy to combine. The downsides are that it's a relatively newer framework, so there's not many integrations with it, not much community and not many papers implemented in it. And note that dynamic computational graphs will make many things rather inefficient because lacking static optimizations.

3.3 Keras

Keras was created by Francois Chollet, a software engineer at Google and is a deep-learning library that sits atop TensorFlow and Theano, providing an intuitive AP that is inspired by Torch and its development is fast growing.

On the other hand as Keras[13] is good in rapid prototyping it lack in flexibility.

3.4 Caffe2

Caffe2[14] with C++ engine is a successor to the original Caffe and is the second deep-learning framework to be backed by Facebook after Torch/PyTorch. The main difference is that Caffe2 is more scalable and light-weight but rather limited in flexibility. Good for smartphone inference.

3.5 Amazon web services

AWS provides a low-cost, scalable and highly reliable infrastructure platform in the cloud. This has been adopted by thousands of businesses globally. Australia, the US, Japan, Europe, Singapore and Brazil are among the data center locations. The locations are widespread to make sure the system is robust and secured against the impact of outages or other such problems.

Advantages[15]

- Security.
  -AWS conducts regular audits to ensure its infrastructural security. It has implemented best practices in security and also provides documentation on how to deploy the security features. It ensures the availability,

integrity and confidentiality of your data and provides 'end to end' privacy and 'end to end' security.
- Cost-Effectiveness
  -You consume only as much storage or computing power as required. No upfront investment or minimum expenditure is required. Generally, it is not easy to predict the requirements for the resources. So, you might allocate fewer resources than required and impact customer satisfaction or you might allocate excessive resources and not be able to maximize return on investment (ROI).
- Flexibility and Openness
  - You can use the programming languages, architectures, operating systems and databases you are familiar with. In this manner, there won't be any need for your IT personnel to pick up new skills and the overall time to market and productivity will improve.
- Elasticity and Agility
  - allows you experiment and innovate quickly through its huge global cloud infrastructure. You can quickly scale up or scale down on the basis of demand. You can also use new applications, rather than wait for months for hardware.

Common company solves a problems those can be group by[16]
- Overspending on hardware and storage capacity.
- Business leaders want IT to help preserve cash.
- A non-standardized IT environment and platform is expensive from a security, support and training perspective.

All these problems are solved by AWS.

AWS provides wide range of services shown in the picture below 3.2.

3.5.1 Frameworks and infrastructure

At the low end computation customers can choose from the most used frameworks:
- Mxnet
- TensorFlow
- Caffe2
- Keras
- CNTK
- PyTorch
- Gluon

Using is then just about calling united interface without dependence with framework was chosen. It bring great for testing different approaches in a short time.

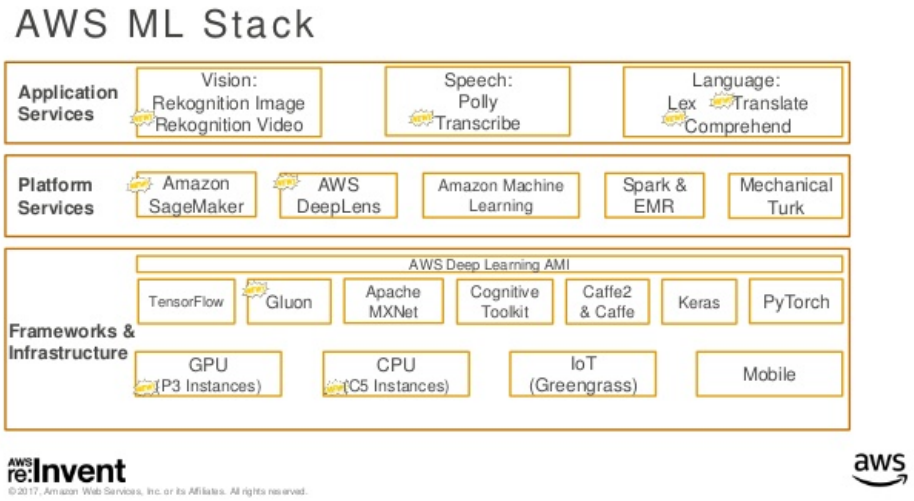

Figure 3.2: AWS ML stack [3]

3.5.2 Application services

It is SaaS layer with applications ready for use. Customers do not need to implement their own solutions to test their data in common cases and can start to use it immediately.

3.5.3 Platform services

On this level resides what we can expect except one and this is Amazon SageMaker.

Amazon realized that developers need a fast way how to build, train and deploy machine learning models with as litter effort as possible. And for this purpose SageMaker was created.

Let's look closer on Amazaon SageMaker.

3.5.4 SageMaker

Amazon SageMaker is a fully-managed platform that enables developers and data scientists to quickly and easily build, train, and deploy machine learning models at any scale with easy to use GUI.

Amazon SageMaker runs on a fully managed elastic compute server. This relieves the data scientist/ developer from DevOps concerns. Amazon SageMaker fully takes care of health checks, and outline infrastructure maintenance tasks via the built-in "Amazon CloudWatch monitoring and logging" service. Machine learning algorithms are provided pre-optimized particularly enhanced

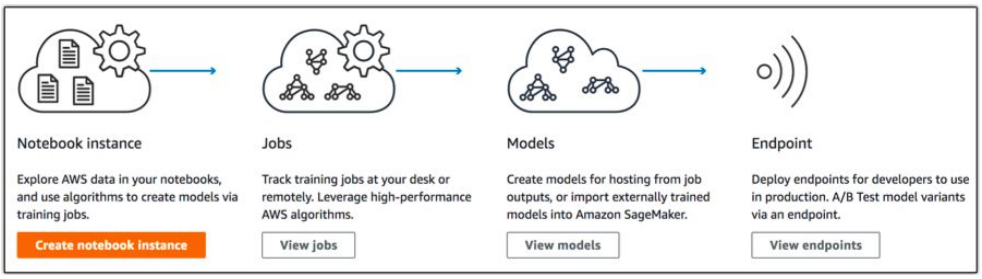

Figure 3.3: Amazaon machine learning stack [3]

to run on Amazon's compute servers and customers need only simply connect them to their data source. Trained models can be deployed for production directly from Amazon SageMaker thta deploys the model as well as implements a secure HTTPs endpoint for the application. Again from customers point of view DevOps are not needed. The last but not the least billing is based on utilization that are mostly dependent on customer use-case and demand peculiarities.

3.6 Microsoft Cognitive Toolkit

The Microsoft Cognitive Toolkit[4] written in C++ is a unified deep learning toolkit that describes neural networks as a series of computational steps via a directed graph. CNTK allows users to easily realize and combine popular model. CNTK has been available under an open-source license since April 2015 and is Microsoft's response to Google's TensorFlow.

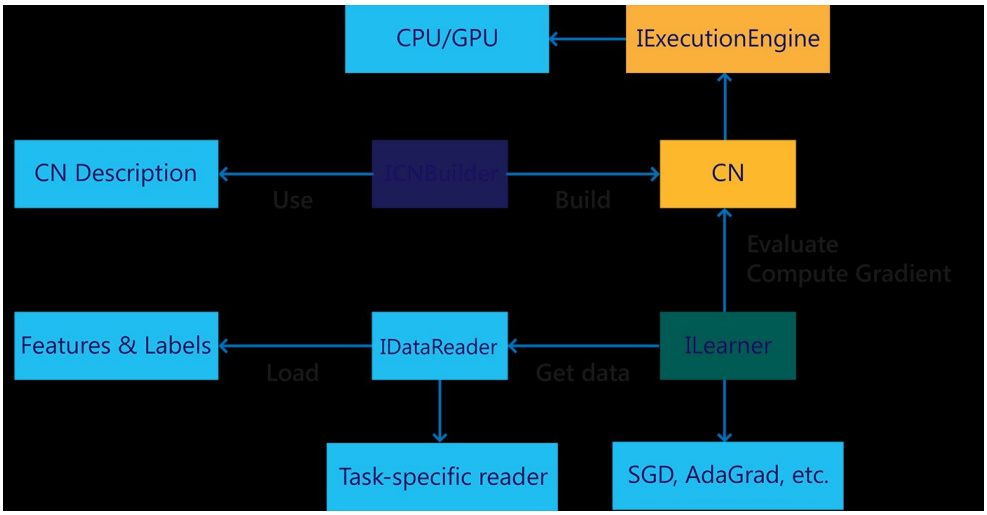

Figure 3.4: CNTK architecture [4]

Thanks to architecture it is very flexible, allows distrubted training and supports main programming languages but lack visualizations.

3.7 Apache Spark MLlib

MLlib[17] is Apache Spark's scalable machine learning library.

Common[**?**] problem of prototyping in Python or R is that moving from development to production environments requires extensive re-engineering.

For this Spark provides data engineers and data scientists with a powerful, unified engine that is both fast (100x faster than Hadoop for large-scale data processing) and easy to use. This allows data practitioners to solve their machine learning problems interactively and at much greater scale.

The advantages of MLlib's design include:

- Simplicity - Simple APIs familiar to data scientists coming from tools like R and Python. Novices are able to run algorithms out of the box while experts can easily tune the system by adjusting important knobs and parameters.
- Scalability - Ability to run the same ML code on your laptop and on a big cluster seamlessly without breaking down. This lets businesses use the same workflows as their user base and data sets grow.
- Streamlined end-to-end - MLlib is on top of Spark and it makes possible to tackle these distinct needs with a single tool instead of many disjointed ones. The advantages are lower learning curves, less complex development and production environments, and ultimately shorter times to deliver high-performing models.
- Compatibility - Data scientists often have workflows built up in common data science tools, such as R, Python and so on. MLlib provides tooling that makes it easier to integrate these existing workflows with Spark.

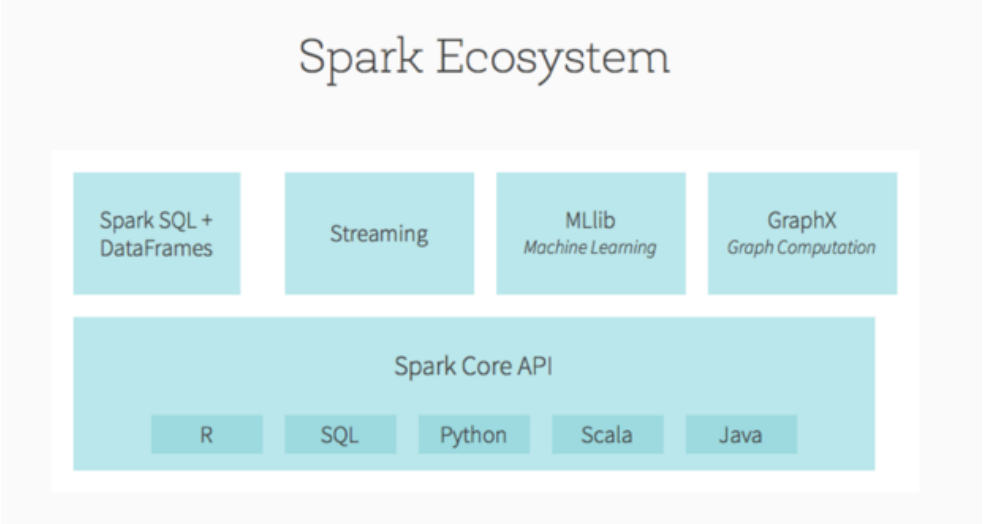

Figure 3.5: Spark ecosystem [5]

3.8 MxNet

Along[18] side previous frameworks Apache's MxNet is a modern open-source deep learning framework used to train, and deploy deep neural networks supporting multiple programming languages.

Supports an efficient deployment of a trained model to low-end devices for inference, such as mobile devices, IoT devices, Serverless or containers which should used the models trained on a higher-level environments because of limited CPU an RAM resources.

It has been chosen by AWS be part of their ML on demand infrastructure.
Now let's switch to some specific algorithm.

3.9 Latent Dirichlet allocation

LDA[19] is a type of topic modeling algorithm. The purpose of LDA is to learn the representation of a fixed number of topics, and given this number of topics learn the topic distribution that each document in a collection of documents has.

In LDA, each document may be viewed as a mixture of various topics. The sparse Dirichlet priors encode the intuition that documents cover only a small set of topics and that topics use only a small set of words frequently. In practice, this results in a better disambiguation of words and a more precise assignment of documents to topics.

First we need to select number of topics to discover. LDA will go through each of the words in each of the documents, and it will randomly assign the

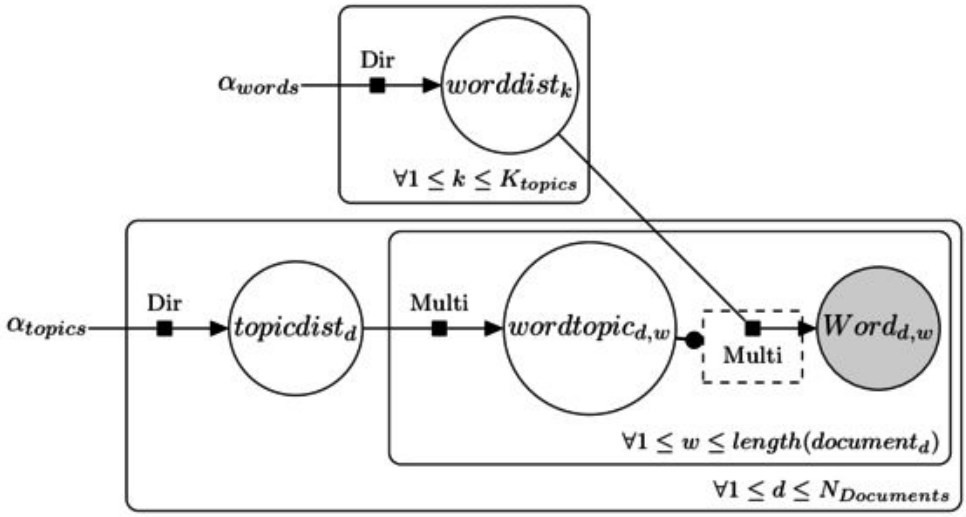

Figure 3.6: Plate notation (Bayesian inference) for LDA [6]

word to one of the K topics. After this step we will have topic representations (how the words are distributed in each topic) and documents represented in terms of topics. This random form is not very optimal or accurate. To better this representation LDA will analyze per document what is the percentage of words within the document that were assigned to a particular topic. And for each word in the document, LDA will analyze over all the documents, what is the percentage of times that particular word has been assigned to a particular topic.

Chapter 4

Creating work items with ML

Chapter 5

ML in production environment

Chapter 6

General Data Protection Regulation

Data are fuel for ML but their use can be limited especially if we talk about customer's data with sensitive information. Now we face new regulation in form of GDPR that will change rules how to threat and store data. What GDPR is?

GDPR[20] is a legal framework that sets guidelines for the collection and processing of personal information of individuals within the European Union (EU). The GDPR sets out the principles for data management and the rights of the individual, while also imposing fines that can be revenue-based. The General Data Protection Regulation covers all companies that deal with data of EU citizens, so it is a critical regulation for corporate compliance officers at banks, insurers, and other financial companies. GDPR will come into effect across the EU on May 25, 2018.

Data types affected by GDPR are:
- Basic identity information such as name, address and ID numbers
- Web data (location, IP address, cookie data, ...)
- Health and genetic data
- Biometric data
- Racial or ethnic data
- Political opinions
- Sexual orientation

What companies will be affected by:
- A presence in an EU country.
- No presence in the EU, but it processes personal data of European residents.
- More than 250 employees.

- Fewer than 250 employees but its data-processing impacts the rights and freedoms of data subjects, is not occasional, or includes certain types of sensitive personal data. That effectively means almost all companies.

GDPR extends rights of data subjects that can be summarized like:
- the right to access
- the right to be forgotten, a.k.a. right to erasure
- the right to data portability

In more detail look at the picture below 6.1.

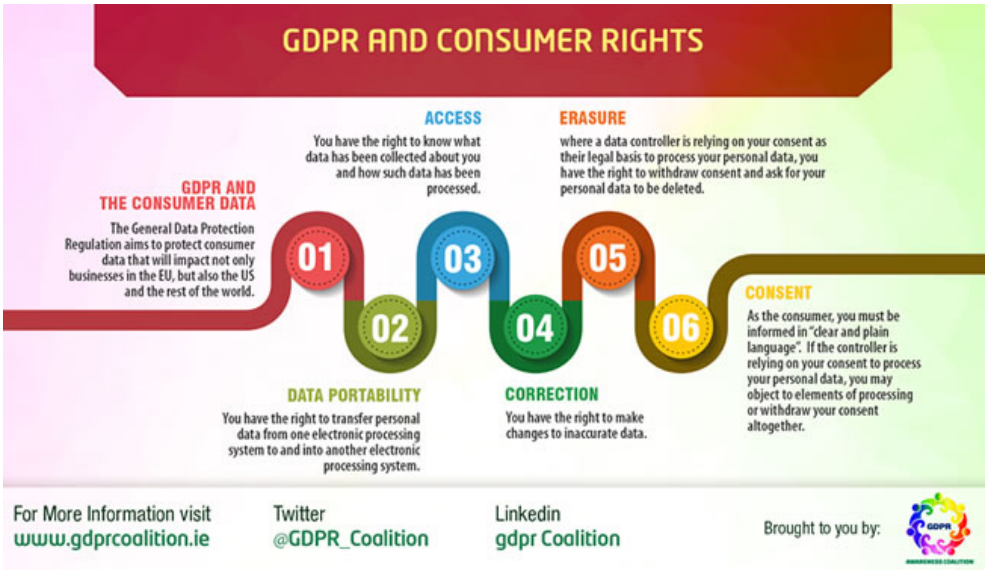

Figure 6.1: GDPR consumer rights [7]

- The data subject's right of access which means
  - the right to know whether data concerning him or her are being processed and
  - if so, access it with loads of additional stipulations.
- The data subject's right to rectification. When personal data are inaccurate, then controllers need to correct them indeed. The previously mentioned right to erasure or right to be forgotten with additional stipulations, among others if personal data has been made public.
- The data subject right to restriction of processing. Simply said, the right of the consumer or whatever you call the natural person under the scope of the GDPR, to limit the processing of his/her personal data with, once more, several rules and exceptions of course.
- The right to be informed. In general, the GDPR asks controllers and so on to inform data subjects on several matters. Providing clear and correct information is a key duty in many regards. Simply said, the

GDPR wants consumers to know because if you don't know you can't decide. The controller must inform recipients who got these data, where feasible. And then the data subject also has a right, even if not strictly called a right, to ask *who are all these recipients who got to see my data.*

- The right to data portability. This is again one of those data subject rights that are in the infographic and which we covered more in depth previously.
- The data subject's right to object. That does indeed mean what it says: data subjects can say they don't want the personal data processing to be done or going on. Direct marketers and people who do profiling should pay a lot of attention to the right to object as it's a lot about them and certainly profiling with automated means.
- The data subject right not to be subject to a decision based solely on automated processing, including profiling, which produces legal effects concerning him or her or similarly significantly affects him or her.

What we need to be careful about and is not part of current Data Protection Directive?

- Data breach notification: Controllers and processors are now required to notify supervisory authorities within 72 hours of learning of a breach and to notify the people to whom the data applies *without undue delay.*
- Explicit consent: GDPR requires that at the time you collect personal data, explicit consent must be given by the data subject. This means organisations can no longer bury generic consent in a long form full of legalese. Instead, organisations must offer specific information on what data is collected, how the data will be stored and processed, and must use clear and plain language. Nothing short of opt-in will do, and it must be as easy to withdraw consent as to give it.
- Data transfer out of the EU: Personal data must not leave the EU unless you have approval from the supervisory authority, or where the data subject is informed of the data transfer and associated risks and authorises the transfer.
- Data protection officer (DPO) appointment: If you process data on a large scale then you must appoint, hire, assign or contract with a DPO, who is your representative to the supervisory authorities that monitor and ensure compliance with the regulation.

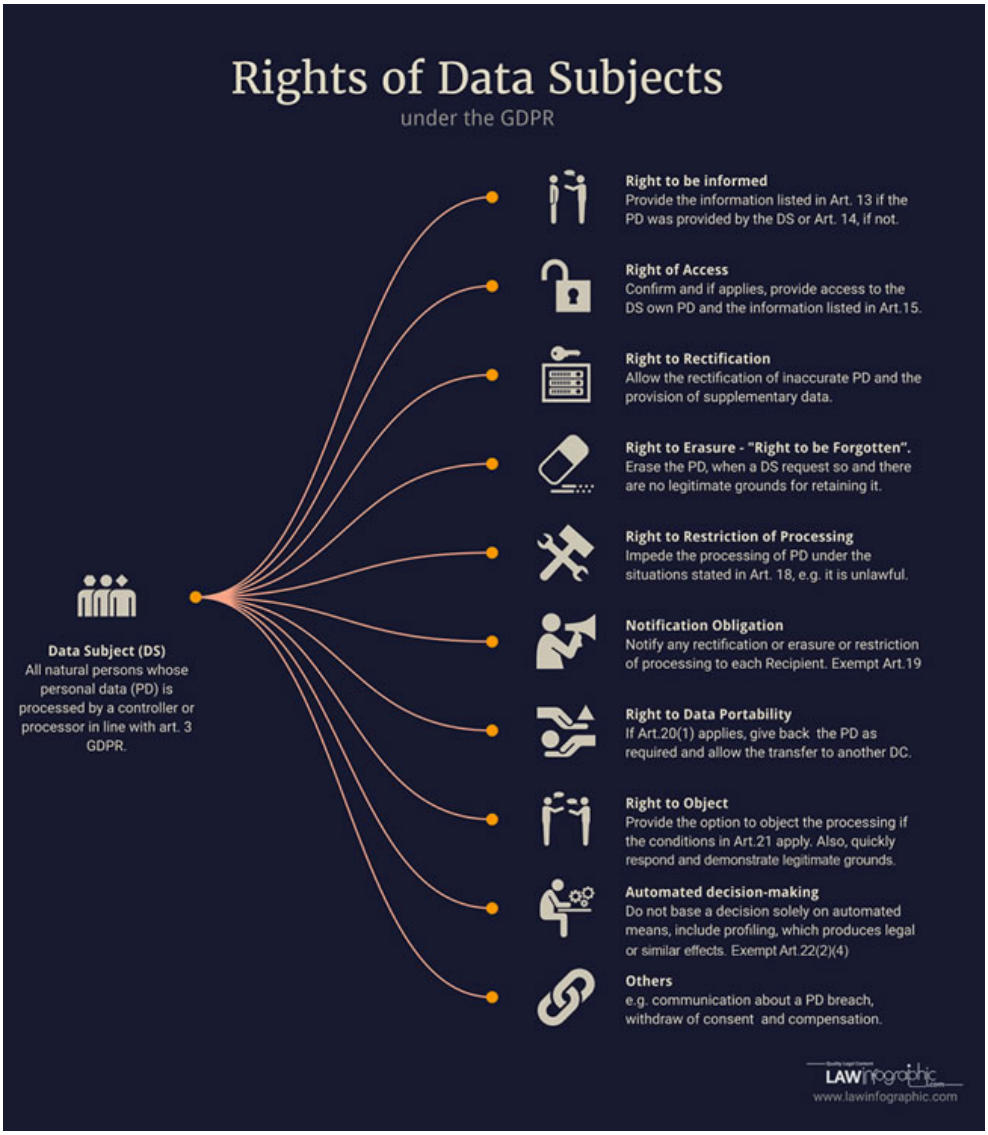

Figure 6.2: GDPR consumer rights in detail [7]

IT with ML is very competitive environment where companies around the world fight for primacy in research and business use. And as GDPR affects only EU this can disadvantage EU companies with their world competitors and their moving our of EU borders. Still true impact of GDPR is unclear and only future shows.

For Polarion ALM (Polarion) GDPR leads to more strictly regulation already so strict environment. Transfer data to external services seems to be quite hard and for this Polarion should implement its own ML solution that customers can install in their own secured environment and work with it without sending data to external repositories.

Chapter 7

Validity

Chapter 8

The value of the enhancement

Conclusion

Bibliography

[1] Polarion ALM. [online], [cit. 2018-05-01]. Dostupné z: https://polarion.plm.automation.siemens.com/

[2] Deep Learning with Tensorflow Part 1 theory and setup. [online], [cit. 2018-05-06]. Dostupné z: https://towardsdatascience.com/deep-learning-with-tensorflow-part-1-b19ce7803428

[3] Machine Learning on AWS. [online], [cit. 2018-05-06]. Dostupné z: https://aws.amazon.com/machine-learning/

[4] The Microsoft Cognitive Toolkit. [online], [cit. 2018-05-06]. Dostupné z: https://www.microsoft.com/en-us/cognitive-toolkit/

[5] Why you should use Spark for machine learning. [online], [cit. 2018-05-06]. Dostupné z: https://www.infoworld.com/article/3031690/analytics/why-you-should-use-spark-for-machine-learning.html

[6] Experiments with Latent Dirichlet Allocation. [online], [cit. 2018-05-06]. Dostupné z: https://mollermara.com/tag/statistics.html

[7] Data subject rights and personal information: data subject rights under the GDPR. [online], [cit. 2018-05-06]. Dostupné z: https://www.i-scoop.eu/gdpr/data-subject-rights-gdpr/

[8] Samuel, A. L.: Some Studies in Machine Learning Using the Game of Checkers. [online], [cit. 2018-05-02]. Dostupné z: https://ieeexplore.ieee.org/document/5392560/

[9] Marr, B.: A Short History of Machine Learning. [online], [cit. 2018-05-03]. Dostupné z: https://www.forbes.com/sites/bernardmarr/2016/02/19/a-short-history-of-machine-learning-every-manager-should-read/

[10] Lanzetta, M.: Social Good at Cloud Scale.

[11] TensorFlow. [online], [cit. 2018-05-05]. Dostupné z: https://www.tensorflow.org/

[12] PyTorch. [online], [cit. 2018-05-06]. Dostupné z: https://pytorch.org/

[13] Keras. [online], [cit. 2018-05-06]. Dostupné z: https://keras.io/

[14] Caffe2. [online], [cit. 2018-05-06]. Dostupné z: https://caffe2.ai/

[15] What is Amazon Cloud, Its Advantages and Why Should You Consider It. [online], [cit. 2018-05-06]. Dostupné z: https://www.netsolutions.com/insights/what-is-amazon-cloud-its-advantages-and-why-should-you-consider-it/

[16] Benefits of Amazon Web Services (AWS). [online], [cit. 2018-05-06]. Dostupné z: http://2ndwatch.com/blog/benefits-of-amazon-web-services-aws/

[17] Apache Spark MTlib. [online], [cit. 2018-05-06]. Dostupné z: https://spark.apache.org/mllib/

[18]

[19] Topic modeling with LDA introduction. [online], [cit. 2018-05-06]. Dostupné z: https://algobeans.com/2015/06/21/laymans-explanation-of-topic-modeling-with-lda-2/

[20] General Data Protection Regulation (GDPR). [online], [cit. 2018-05-06]. Dostupné z: https://www.investopedia.com/terms/g/general-data-protection-regulation-gdpr.asp

Appendix A

List of abbreviations used

ALM Application lifecycle management

Polarion Polarion ALM

ML Machine learning

pof Proof of Concept

RNN Recurrent neural network

AWS Amazon web services

GUI Graphic user interface

CNTK Microsoft Cognitive Toolkit

LDA Latent Dirichlet allocation

GDPR General Data Protection Regulation

Appendix B

CD contains

- `readme.txt` ..... stručný popis obsahu CD
- `thesis` ..... zdrojová forma práce ve formátu LaTeX
- `text` ..... text práce
  - `thesis.pdf` ..... text práce ve formátu PDF
  - `thesis.ps` ..... text práce ve formátu PS