Project Management and Quality Assessment in label creation process for Machine Learning projects

A case study for rooftop detection, under the supervision of Occupy AI

AHMED AL DADA

DEPARTMENT OF MANAGEMENT AND PRODUCTION ENGINEERING

MSc in ENGINEERING AND MANAGEMENT

Supervisors: Prof. Giulia Bruno, PhD Emiliano Traini

Academic Year 2022-2023

ACKNOWLEDGMENTS

I would like to thank everyone who helped and motivated me in this Master's Degree. It was a hard beginning, especially in the Covid- 19 times. I had to take more time than I thought I needed, but I made it through.

I would like to thank God, my parents, my 2 siblings and my friends whether here or in my country, Lebanon, for their forever support and motivation. I would like to thank my cousin: without him, I wouldn't have come and studied in Italy. I want to thank my supervisor, Prof. Giulia Bruno, who was the only person that replied to my email about a thesis proposal and helped me to do it with a company. For what concerns the company members, I will be forever grateful for your full time support and help. Thank you to Emiliano Traini, Alessandro Ciociola, and Clémence Maquet for your great supervision.

ABSTRACT

Projects in machine learning are becoming more common, and proper project management methods and techniques are required to assure their success. To guarantee a good end, these initiatives must be well managed and the success criterias must be understood. Project management (PM) is a profession that has arisen precisely to guarantee that project goals are realized by the use of project-specific knowledge, tools, and procedures.

The purpose of this paper is to analyze the project management approach for a project in rooftop detections using satellite images, under the supervision of Occupy AI, the company responsible for this project. The methodologies used in the paper to conduct the ML project are SCRUM and Lean software development, which are presented in depth before being applied through the comprehensive case study.

TABLE OF CONTENTS

CHAPTER 1 - INTRODUCTION

1.1 Project management in machine learning project

Project management (PM) is the practice of guiding a team to achieve the goals of a project within specified constraints. These constraints are typically documented at the start of the development process, and include factors such as scope, time, and budget. In addition to meeting these constraints, project management aims to optimize the distribution of required inputs to achieve predefined goals.

Projects have a life cycle that includes phases of initiation, planning, and execution. Project management provides tools and techniques to manage this life cycle and ensure success. For example, project management can improve the process of developing software. From an operational standpoint, the main responsibilities of project management include establishing milestones, coordinating operations, communicating with clients, evaluating and developing personnel, and ensuring effective project implementation.

The goal of project management is to provide inputs efficiently, but it is transitioning from a purely operational function to one that also delivers strategy. Project management is essential to corporate innovation processes, as projects are an effective way to organize and promote innovation. It is necessary because of the environment's instability, complexity, and change. Every project involves changes in demand and requires team adaptation.

As organizations are impacted by the growth of AI technologies, the role of the project manager for artificial intelligence projects is becoming increasingly important. Machine learning (ML) project management is of growing importance, as a team working in this sector can benefit from having a clear, agile framework. However, the difficulty of PM in machine learning is real. It is challenging to predict the length of ML tasks, ensure that the team follows a repeatable process, and set defined goals. Despite the painstaking efforts of practitioners, stakeholders are generally dissatisfied with these tools and techniques brought by the PM so far.

1.2 Labeling data in ML systems

Annotation is the process of labeling data on images in machine learning, and more broadly, in artificial intelligence (AI) applications. The images may include individuals, automobiles, or other entities, making them machine-readable. Annotations come in various forms and can be used to teach machines about the existence of various items in the world (Minds, 2019). To summarize, annotation is the procedure of labeling or annotating an object of interest so that it can be detected or recognized by algorithms. Depending on the project's needs, various methodologies and styles of data labeling are used and developed (Bisen, 2019).

Data labeling or annotation is used in many research domains and typically requires numerous specialists working in collaborative processes to ensure the reliability of the final guidelines. Many cycles of data analysis, label debate, and labeling guidelines improvement may occur to achieve consistent final labels with a good level of assertiveness (Lima and Lighthill, 2021).

Labeling data may seem like a straightforward job necessary for training various machine learning systems, but it is actually loaded with complications. So far, this work has been unavoidably arduous, particularly when supplying a large volume of labeled data to more advanced techniques, including deep learning algorithms.

Domain experts, who have the unique knowledge to create precise annotations, can label data for the highest number of machine learning projects. However, these experts often lack expertise in labeling software. As a result, the labeling procedure must be performed through an artificial intelligence (AI) service provider or an external third-party, which can prove to be costly or ineffective in terms of cost-efficiency. Thus, it is necessary to develop a comprehensive and user-friendly labeling solution to reduce the expenses of this activity (Lima and Lighthill, 2021).

1.3 Computer vision for satellite images

Computer vision is a field of artificial intelligence (AI) that enables machines to extract useful information from digital images and videos, and then take action or make recommendations based on that information. While AI allows computers to comprehend, computer vision allows them to see and interpret visual data. Although computer vision works similarly to human vision, humans have an advantage due to a lifetime of experience and knowledge to learn how to differentiate images, determine distance, movement, and detect visual anomalies.

Computer Vision (CV), deep learning, and AI for remote sensing applications can help address challenges posed by vast satellite image datasets through the use of characterized-based models that can identify and classify features with pinpoint accuracy and speed. By incorporating CV and deep learning techniques, end-users can better understand the data and improve the reliability of feature categorization, as well as assess aspects of image data quality. Additionally, CV algorithms can remove noise, improve satellite and aerial picture data, and facilitate analysis of large areas, such as categorizing objects, features, detecting changes, performing data fusion, cloud removal, and spectrum analysis from imagery.

To identify a building's roof, the space must first be divided by a road. The structure is spotted in the footage because it is split by roadways. Roads and paths are defined as places in the image where space is separated by either a constant or unique reference point, as they are created in a straight line. This is a critical step in rooftop detection using satellite images, which is the focus of a project carried out by Occupy AI, a consulting and teaching company, and one of their corporate clients. The goal of the project is to use a precise methodology and platform to detect rooftops in satellite images received from the client.

The project utilized lean and scrum agile methods, which will be discussed in detail in this paper. The general project will be briefly described, followed by a clear presentation of the methodology and results. The paper will conclude with an overview of the outcome of the project, as well as the implications of the results for the field of machine learning project management.

CHAPTER 2 - STATE OF THE ART

The "State of the Art" section of our paper will have several subsections that contribute to the ultimate goal of presenting the latest developments in the field we are studying. We will begin with a review of other relevant research papers, followed by a discussion of project management in machine learning systems based on current literature. Then, we will provide an overview of computer vision for roof detection, labeling services, and business examples of labeling services.

2.1 Research papers review

In this section, we will present a novel approach to the literature review of existing research papers, with the guidance of my supervisor. We will summarize the key contents of the research papers and focus on the main aspects relevant to our study. The information included will be categorized by the article's title, scope, methods used, and results.

Articles	**Scope**	**Method**	**Results**
Big Data analytics in Agile software development: A systematic mapping study Katarzyna Biesialska, Xavier Franch, Victor Muntés-Mulero 2020	The goal of this endeavor is to advance by combining agile system development (ASD) with Big Data analytics (BDA)	Manually searching and snowballing publications released between 2011 and 2019.	The findings suggest that Big Data analytics (BDA) is used throughout the agile system development (ASD)lifespan. According to the findings, data-driven software development is concentrated in the following areas: code source analytics, defect/bug repair, validation, project management analytics, and application use analytics.
Data Labeling: An Empirical Investigation	Identify industry issues in the labeling process and	Present and explain Active Learning and	Supervised machine learning algorithms

into Industrial Challenges and Mitigation Strategies Teodor Fredriksson, David Issa Mattos, Jan Bosch, Helena Holmstrom Olsson 2020	provide solutions to these challenges.	Semi-supervised Learning approaches in terms of how they might be used to label difficulties in practice.	account for 95% of all machine learning algorithms used in the industry. As a result, it is critical that each dataset has tagged instances. Alternatively, the data would've been inadequate, making supervised learning impossible.
Understanding rooftop PV panel semantic segmentation of satellite and aerial images for better-using machine learning Peiran Li , Haoran Zhang, Zhiling Guoa, Suxing Lyua, Jinyu Chena, Wenjing Li, Xuan Song, Ryosuke Shibasaki, Jinyue Yanb 2021	From the standpoint of computer vision, this work explored the semantic-segmentation properties of photovoltaic (PV) panels.	photovoltaic (PV) installed capacity provides a novel solution that is both cost-effective and data-consistent. Previous research looked into the potential of segmenting photovoltaic (PV) panels from photos using machine learning techniques.	The results show that photovoltaic (PV) panel image data contains numerous distinct properties, including a highly class-imbalanced and non-concentrated distribution, homogenous texture and heterogeneous color features, and a significant resolution threshold for efficient semantic segmentation.
Aerial Imagery for Roof Segmentation: A Large-Scale Dataset towards Automatic Mapping of Buildings Qi Chen, Lei Wang, Yifan Wu, Guangming Wu, Zhiling Guo, Steven L. Waslander 2018	The study introduces Aerial Imagery for Roof Segmentation, a newly designed benchmark dataset (AIRS). This collection covers around 220,000 structures and has a large coverage of aerial photography with a resolution of 7.5 cm.	As baseline approaches for benchmarking, several cutting-edge deep learning models can be used. ResNet is a popular object recognition architecture because it allows for the successful training of really deep neural networks.	The study validates the deep learning models' outstanding performance on roof segmentation while also highlighting areas for further development.
Empirical studies of agile software development: A systematic review	The purpose of this systematic review is to assess, synthesize, and	Methods in agile software development are a collection of	The investigations were divided into four categories: introduction

Tore Dyba, Torgeir Dingsøyr 200	provide the empirical results about agile software development to date, as well as to offer an overview of themes examined, their findings, the robustness of the findings, and implications for practice and research.	software development strategies developed by experienced practitioners. These strategies might be considered as a reaction against traditional or plan-based tactics.	and implementation, human and societal variables, agile method perceptions, and comparative studies. Within each of these topics, they discovered a number of documented benefits and limits of agile development.
A Taxonomy of Software Engineering Challenges for Machine Learning Systems: An Empirical Investigation Lucy Ellen Lwakatare, Aiswarya Raj, Jan Bosch, Helena Holmstr¨om Olsson, and Ivica Crnkovic 2019	The goal of this research is to identify and categorize software engineering issues that different firms confront while designing software-intensive systems with machine learning components.	They investigated the creation of machine learning systems across six different firms across diverse areas using a case study technique and found key software engineering problems.	The difficulties are represented by a suggested taxonomy that represents the evolution of the adoption of machine learning (ML) components in software-intensive systems used in industrial contexts.
Machine Learning Project Management CHRISTOFFER BRASJÖ & MARTIN LINDOVSKY 2019	The goal of this thesis is to address a knowledge vacuum in the literature by providing project management assistance for machine learning initiatives.	This study concentrated on applied machine learning initiatives in the Gothenburg area of Sweden, in which data from 13 interview participants from ten firms and organizations were collected.	According to the report, firms often fail to invest in developing a data culture through strategic data collecting before attempting to implement machine learning (ML) to automate operations and/or improve customer experience.
Project Management In Industrial Applications of Machine Learning Andreas Trokildsen Hjertaker, Irnis Besirovic	The goal of this thesis is to study the performance targets for Project Management within Industrial Machine Learning Applications.	This paper is based on a case study using qualitative data collected through eight semi-structured interviews. Interviews	The results reveal both variety and correlations among the informants when compared to the thesis's accessible used literature.

2022		involving Project Managers (PM) were utilized to gain first-hand knowledge of critical elements in ML projects in the industrial sphere.	
CNN Algorithm for Roof Detection and Material Classification in Satellite Images Jonguk Kim, Hyansu Bae, Hyunwoo Kang and Suk Gyu Lee 2021	This work proposes an algorithm for determining a building's location from satellite photos and utilizing that knowledge to change roof content.	They present a method for detecting roofs and categorizing materials in satellite pictures. Based on roads, satellite imagery identifies locations where buildings are expected to exist.	When compared to GoogleNet, the suggested model's learning outcomes revealed an approximately 9% improvement in material categorization accuracy.

Table 1: Literature review of the main articles used

2.2 Project management approaches

Agile software development methods are a set of practices that were developed as a response to plan-based or conventional methods that emphasize a "rationalized, engineering-based approach". Classical methods view problems as specifiable, and solutions as optimum, predictable, and universally applicable for every problem (Tore and Dingsøyr 2008). Traditionalists promote detailed planning, standardized methods, and strict reuse to create a reliable and predictable development process. In contrast, Agile methodologies acknowledge the reality of an unpredictable environment by relying on people and their ingenuity, rather than processes alone.

Agile software development (ASD) is often seen as a solution to the many shortcomings of plan-driven techniques (Biesialska, Franch, and Muntés-Mulero 2020). The key tenets of Agile software development include responding quickly to changing system requirements, close collaboration between developers and customers, and iterative development delivered in stages. In this approach, each iteration (known as a cycle in SCRUM) is required to produce and deliver a functional software product as quickly as possible. Consequently, a typical iteration includes stages such as demand, development and design testing, delivery, and assessment. Since multiple iterations may be required to produce a component or a new feature, all of these stages create a loop.

Although Agile methodologies were initially designed for small businesses, large enterprises have adopted their scaled-up equivalents such as the Scaled Agile Framework (SAFe), Large-Scale Scrum (LeSS), Scrum-of-Scrums, Nexus, Scrum at Scale, or Disciplined Agile Delivery (DAD) (Biesialska, Franch, and Muntés-Mulero 2020). Communication and coordination between geographically dispersed development teams or organizational units is a challenge for large enterprises. Therefore, data analytics in software engineering may help organizations learn how to improve their software development processes.

Design quality is critical in agile software development. Since design is done continuously, in smaller parts, rather than all at once and upfront, Agile approaches are sometimes confused with ad hoc and cowboy coding. Each Agile methodology addresses design quality in its unique way (Highsmith and Cockburn 2001). For example, the Dynamic Systems Development Methodology (DSDM) recommends a series of prototypes to target unstable or uncertain areas such as new technology, business rules, and user interface design.

Scrum employs intensive 15-minute daily meetings and thorough iteration evaluations at the conclusion of each 30-day iteration. Indeed, if the feedback loop between clients and management lasts six months or more, the team is not agile. Agile methodologies advocate for short iterations of two to six weeks in length, during which the team makes continual trade-off decisions and responds to new knowledge. The cycles of Extreme Programming (XP) and Scrum are more focused, lasting two to three weeks for XP and 30 days for Scrum, while other software development methodologies such as Crystal and Agile (ASD) allow for additional flexibility (Highsmith and Cockburn 2001).

Agile method	Description
Crystal methodologies	Yellow, Clear, Red, Orange, and Blue are a family of approaches for co-located teams of varying sizes and criticality. Crystal Clear, the most agile technique, concentrates on interaction in small teams producing non-life-critical software. A clear development is distinguished by seven characteristics: regular delivery, reflective growth, fluid communication, personal safety, concentration, simple

	access to expert users, and technological environment needs. (Cockburn 2005)
Dynamic software development method (DSDM)	Projects are divided into three stages: pre-project, life cycle of a project, and post-project. Dynamic Systems Development Methodology (DSDM) is founded on nine principles: user involvement, project team empowerment, frequent distribution, attempting to address business requirements, incremental and iterative development, allowing for modifying changes, high-level scope becoming fixed before the project begins, evaluating all through the lifecycle, and effective and efficient communication. (Stapleton 2003)
Feature-driven development	Manages to combine model-driven and agile development, with an emphasis on the initial object model, feature division, and iterative design for every feature. Claims to be ideal for critical system development. A feature iteration has two phases: development and design. (Palmer and Felsing 2002)
Lean software development	An application of lean manufacturing ideas, namely the Toyota production method, to software development. The seven principles are as follows: remove waste, amplify learning, choose as late as feasible, deliver as quickly as possible, empower the entire team, develop integrity, and see the big picture (Poppendieck, Poppendieck, and Poppendieck 2003)
Scrum	Focus on project management in contexts when planning ahead is difficult, including systems for "empirical process control"; feedback loops are a key component. A self-organizing team creates software in increments (called "sprints"), beginning with planning and finishing with a review. A backlog of features to be incorporated into the system is maintained. The product owner then determines which backlog items will be developed in the next sprint. A regular stand-up meeting allows team members to coordinate their work. The scrum master, a team member, is in control of resolving issues that prevent the team from functioning successfully. (Tore and Dingsøyr 2008)
Kanban	Kanban, which means "billboard" in Japanese, was established for lean manufacturing but has now spread to other fields, including software development. To eliminate bottlenecks, the Kanban board illustrates the workflow in the various stages and restricts the work-in-process tasks for the relevant phases. The team must prioritize user stories in this Kanban pipeline management system. Kanban, unlike Scrum, does not have defined meetings, time boxes, or roles. Kanban may give some structure to data science teams while still allowing for flexible execution (BRASJÖ and LINDOVSKY 2019)
Extreme	The emphasis is on optimal practices for growth. The planning game, modest

programming (XP; XP2)	updates, metaphor, simple design, validation, refactoring, pair programming, communal ownership, continuous integration, 40-hour week, on-site clients, and coding standards are among the twelve practices. The new "XP2" includes the following "primary practices": sit together, full team, informative workspace, enthusiastic work, system testing, stories, weekly cycles, quarterly cycle, slack, 10-minute build, integration, test-first coding, and incremental design. There are 11 corollary practices as well."(Beck 2000)

Table 2: Agile methodology description (Tore and Dingsøyr 2008)

2.3 Introduction to computer vision for roof detection

In recent years, there has been a significant increase in the use of GIS (Geographic Information System) technology for estimating rooftop photovoltaic (PV) capacity. One approach is to estimate PV capacity using a statistically representative sample size based on freely accessible data, as was done by Izquierdo, Rodrigues, and Fueyo (2008). Other researchers have attempted to estimate PV capacity by estimating the sun brightness of rooftops, which is crucial for deriving rooftop morphologies, using Light Detection and Ranging (LiDAR) data for rooftop extraction (Li, Zhang, and Guo 2021).

As machine learning technology has advanced, some researchers have started applying machine learning methods to estimate PV capacity. For instance, using vector GIS data and the Support Vector Machine (SVM) method, Assouline estimated Switzerland's possible rooftop PV capacity. Similarly, Joshi researched Abu Dhabi roofs for solar photovoltaic development using satellite/aerial photos and the Support Vector Machine technique (Li, Zhang, and Guo 2021). Additionally, some researchers used LiDAR data with a deep neural segmentation algorithm to identify the rooftop area for 3D model development, then finished the PV potential calculation. Other researchers used satellite pictures and deep learning algorithms, such as DeepRoof, for potential solar capacity evaluation.

In the field of computer vision, the ImageNet dataset has enabled early successes in the use of deep learning algorithms for object detection. The dataset contains over 14 million annotated photos divided into 22,000 classifications, and has since been used for recognition tasks (Wang, Wu, and Guo 2018).

Building detection using visual remote sensing (RS) images has also been extensively studied. Prior to the advent of deep learning, much of this research was done by extracting characteristics such as strong edge, form design, roof color, shadow data, local circumstances, or their combination. To perfect roof recognition, approaches like pattern recognition, mathematical morphology, dynamic contours, graph-based algorithms, random forests, and support vector machines (SVM) were applied (Wang, Wu, and Guo 2018).

In conclusion, machine learning methods have been successfully applied to the estimation of rooftop photovoltaic capacity and building detection. Researchers have used a variety of techniques, including deep learning, LiDAR, and SVM, to achieve accurate results. These methods have the potential to significantly improve project management for machine learning systems.

2.4 Existing labeling services

To perform supervised machine learning, labels of adequate quality are essential. The reliability of the dataset directly affects the model's performance in operations. Crowdsourcing has become a typical technique to gain quality labels under human guidance to overcome a dearth of labels in both number and quality, particularly for natural language and computer vision processing applications. An industry has developed around recruiting crowdsourced data annotators on platforms such as Amazon Mechanical Turk and Upwork.

Using a third-party firm to label your data provides advantages, such as eliminating the need to create your own annotation techniques and labeling infrastructure. In-house labeling also requires investing time in educating annotators, which may not be ideal if you lack the necessary time and money. Instead, third-party annotators are already familiar with their tasks. However, there are some disadvantages to using external data labelers. Annotations can be tossed carelessly to preserve the labelers' performance, and private sensitive corporate information has to be provided to crowdsourcing sites for labeling (Federiksson et al., 2020).

To improve label quality and, consequently, better the machine learning model, repeated labeling should be employed. This method appears to perform effectively when executed carefully, taking into account label and machine learning model variabilities. However, this strategy does not guarantee quality improvement. Empirical assessment studies have evaluated several techniques for computing crowd acceptance on benchmark crowdsourcing datasets using Statistical Quality Assurance Robustness Evaluation (SQUARE) (Fredriksson et al., 2020).

Pricing Examples in Labeling Services

In this section, we provide some examples of top platforms for labeling services and prices associated with them. The platforms considered are Google Cloud and Amazon Marketplace. For example, Google's AI Platform Data Labeling Service allows users to obtain human labeling for data that they want to use to train a bespoke machine learning model. The pricing for this service is determined by the number of human labels assigned to each data item and the total number of annotation units produced.

The table below shows the price per 1,000 units for a human labeler based on the tier listed for each target. The Tier 1 price applies to the first 50,000 monthly units in any Google Cloud project, while the Tier 2 price applies to the next 950,000 units every month, up to 1 million units.

LABELING SERVICES COSTS OF GOOGLE CLOUD

	OBJECTIVE	UNIT	TIERS 1	TIERS 2
IMAGE	classification	image	35$	25$
IMAGE	bounding box	bounding box	63$	49$
IMAGE	segmentation	segment	870$	850$
IMAGE	rotated box	bounding box	86$	60$
IMAGE	polygon, polyline	polygon, polyline	257$	180$
VIDEO	classification	5-sec video	86$	60$
VIDEO	object tracking	bounding box	86$	60$
VIDEO	event	event in 30-sec video	214$	150$
TEXT	classification	50 words	129$	90$
TEXT	entity extraction	entity	86$	60$

Table 3: Labeling services costs for Google Cloud

In terms of pricing, we provide some examples of top platforms for labeling services in the following section, namely Google Cloud and Amazon Marketplace.

Amazon Marketplace collaborates with Cogito on data labeling services, and Cogito is a leader in human-in-the-loop workforce solutions. With over 1,000 experienced specialists in-house, they provide data labeling and classification operations for computer vision and natural language processing applications. They have over 8 years of expertise in enriching various types of data, such as text, image, audio, and video.

Their primary competencies include:

- **Image**: item detection and tracking, image moderation, image classification, semantic segmentation, and image-to-text transcription.
- **Video**: video classification, video moderation, event-based timestamp labeling, frame-by-frame item detection and tracking, and live video stream monitoring.
- **3D point clouds:** item detection and tracking, semantic segmentation, and photogrammetric point clouds.
- **Text**: classification, recognizing named entities, and extracting relationships.
- **Documents**: data extraction, classification, and optical character recognition (OCR) for invoices, receipts, identification cards, paystubs, credit card statements, and bank statements.

Cogito software charges based on consumption, with the corporation charging $5.40 per hour or $0.015 per unit, starting from 10-second increments.

CHAPTER 3 - PROPOSED METHOD

In this paper, we are introducing 2 major methods for the project, which we will detail further in chapter 4. The proposed PM approach for the project uniting Occupy AI and its client is a hybrid methodology, taking inspiration from Lean software development and SCRUM. The project management framework of this paper thus applies to creating labels powered by humans in a supervised machine learning project. In this chapter, we will recall the theoretical foundations of project management, and especially our two methods. Then, we will analyze how PM methods can provide an applied framework for organizing ML projects, acquiring and assessing data quality, creating and validating labels, and tuning and validating the model, diving in each of these terms and explaining their meaning.

At the risk of repeating the fundamentals of PM, let us first recall the concept. Project management delivers a project that fulfills the client's needs. In many cases, project management involves shaping or refining the client's requirements to effectively meet their objectives. Once the client's objectives are clear, they should guide all decisions made by project participants, such as project managers, contractors, subcontractors, and designers. Poorly stated or overly restrictive project management goals can hinder decision-making and create a lack of direction.

In the context of this thesis, PM comes hand in hand with ML and supervised learning. The latter typically involves teaching a machine learning algorithm to learn a function that maps an input to an output based on sampled input-output pairs. Supervised learning is used when specific goals are identified to be achieved from a certain set of inputs. The most common supervised tasks are "classification," which separates data, and "regression," which fits the data. For example, predicting the class label or sentiment of a piece of text, such as a tweet or a product review, is an example of supervised learning.

3.1 Fundamentals of the PM methodology

For this project, we will focus on two aspects of agile methodologies: Lean Software Development and SCRUM Agile methodology. In the software development area, Lean Software Development combines Lean Manufacturing and Lean IT practices. Over the years, a pro-lean subculture quickly formed within the Agile movement, taking inspiration from techniques derived from the Toyota Production System.

Lean software development

Lean development can be summarized in seven concepts that are theoretically similar to those of Lean Manufacturing.

Remove waste

Everything that does not bring value to the client should be discarded and avoided as much as possible. This includes unused code and functionality of the software, software development process delays, unsafe and uncertain requirements, bureaucracy and heavy formalities, and slow and inefficient internal communication.

Enhance learning

Software development is a never-ending learning process. Developers often face challenges due to the complexity of the final product. It is crucial to maximize learning opportunities to ensure improvements in software development on every occasion.

Latest possible decision

As software development is inherently unpredictable, the best outcomes can be achieved through an option-based approach. This approach consists of postponing choices as much as possible until they can be made based on facts, rather than uncertain projections and assumptions.

Fastest possible delivery

In a rapidly changing technological landscape, the quickest delivery, without significant flaws, is the key to success. Faster delivery means faster feedback, which can be integrated into the following iteration. The fewer iterations, the better the team training and communication.

Empowering the team

In the past, managers advised employees on how to perform their jobs. With the work-out approach, managers are now trained to hear from developers to better describe the activities that the team can perform and to help make better ideas a reality.

Construction integrity

The client must have a complete experience of the system, which is referred to as perceived integrity. The client must have a sense of how it is promoted, distributed, disseminated, and accessible, as well as how easy it is to use, the pricing, and how well it solves issues.

See the whole

Today's software systems are more than the sum of their components. They are the result of software interactions and must be considered as such, not just as an assembled entity.

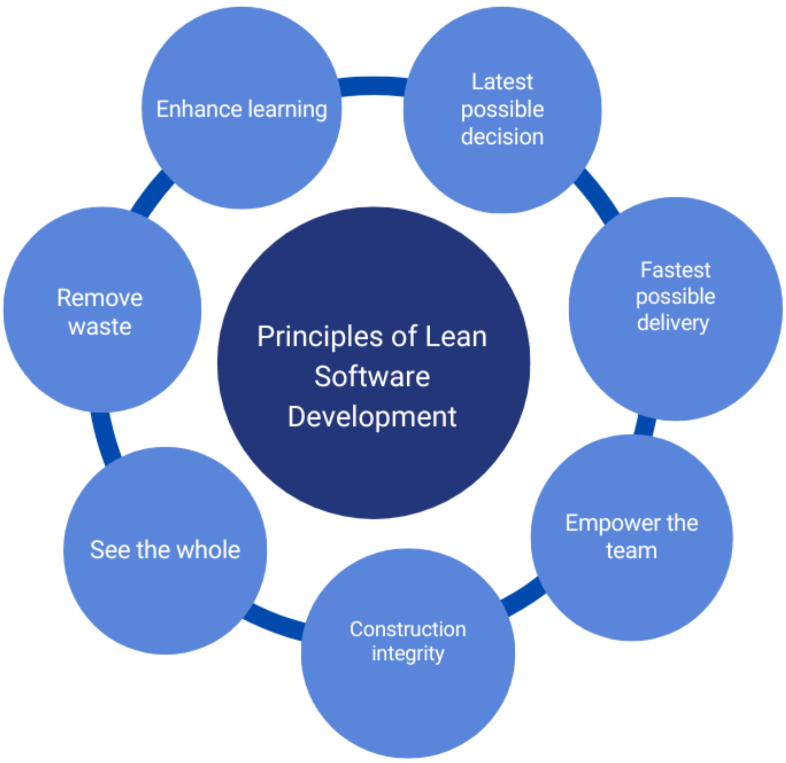

Figure 1: Principles of Lean Software Development

SCRUM Agile methodology

The SCRUM Agile methodology is another widely-used process framework for implementing Agile Project Management in machine learning system projects. The main idea behind SCRUM is that system development is subject to a variety of environmental and technical project variables that are likely to change during the process.

SCRUM has well-defined roles, responsibilities, and frequently scheduled meetings, and is based on the concept of developing the product in iterative cycles called "sprints." Each sprint includes all the traditional product development phases, and results in the building of a releasable increment (Brasjö and Lindovsky, 2019).

The duration of each sprint should be between 1-4 weeks, with each sprint lasting the same amount of time throughout the process. Stakeholder input is included at the end of each sprint in the form of feedback and review, but intermediate outcomes are not released before the completion of a sprint.

SCRUM emphasizes generating client value while empowering independent teams, but some corporations may face cultural hurdles when adopting it (Brasjö and Lindovsky, 2019). For example, insufficient time allocation to the planning phase is a common issue that can lead to problems later in the development process.

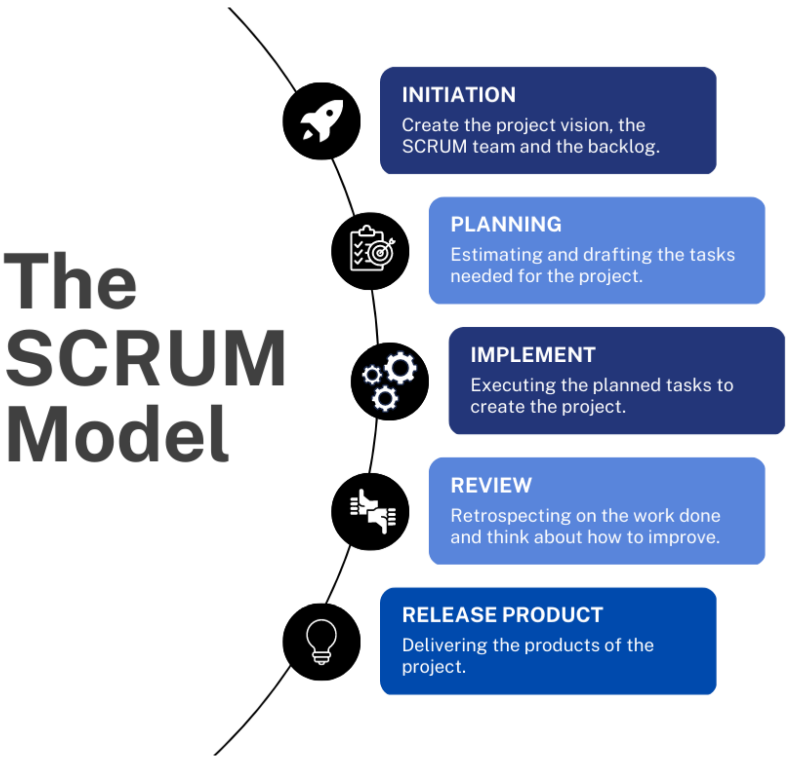

Figure 2: The SCRUM model

Using SCRUM, a software can be produced in increments that incorporate the functionality that the customer needs. At every increment, the client may alter the needs of every iteration, and SCRUM makes it extremely simple to address the new change requirements. It is used to manage the development process by using principles from the organizational process control hypothesis, such as adaptability, proficiency, and leanness. SCRUM emphasizes cooperation and the quality of products in a flexible setting (Hayat et al. 2019).

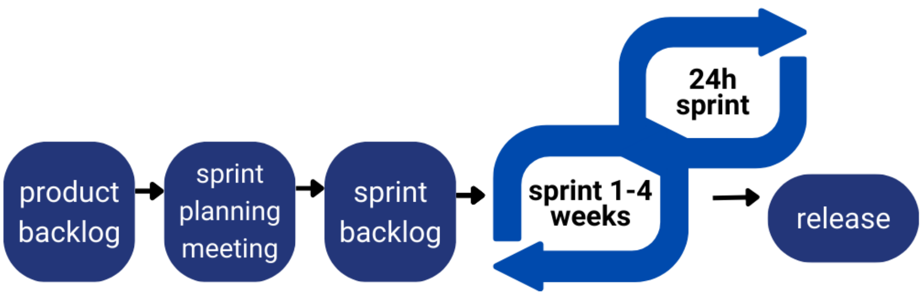

Figure 3: The SCRUM Methodology

The Six Principles of SCRUM

The SCRUM framework is based on six basic principles that must be followed throughout every project. These principles are non-negotiable and must be adhered to, especially when the team loses focus or the project encounters setbacks. The six principles are as follows:

Control over the experimental procedure

The SCRUM framework relies on hard evidence and experience of reality. The empirical process, which is the progress in project development based mostly on the observation of reality and experimentation, includes transparency, inspection, and adaptation.

Self-organization

SCRUM involves many individuals and teams, and self-organization is a core aspect of the framework. The people involved should be encouraged to work as independently as possible, with the aim of assessing individual performance.

Collaboration

SCRUM is a collaborative process, with many individuals performing different functions within teams, all working towards the same objective of completing the project. Values of awareness, clarity, and task distribution must be followed while working on each release.

Prioritization according to value

Tasks must be prioritized according to their value and importance, especially from the end-user's point of view. The value that tasks represent for the company must also be taken into account.

Time-boxing

Timeboxing refers to planning the time allocated for each task during each sprint in advance. The planning occurs during the sprint planning meeting, which usually lasts 1 to 2 hours, and is monitored during daily meetings. The review of time allocation happens at the end of the sprint, during the sprint review.

Iterative development

The requirements of a project are subject to constant adaptation and review. During the development process, software development activities are reworked and refined, an approach known as "iterative development," which makes the team more adaptable.

Figure 4: The principles of SCRUM

3.2 General description of the project

Occupy AI is collaborating with a corporate client on a single supervised model project called "Rooftop Detection from Satellite Images." The project requires the client to have expertise in the business area, which will allow Occupy AI to tailor its AI-based solutions to the customer's needs. The client team possesses a high level of specialization in the required technical discipline to deliver the end-user service, making it highly competent in integrating and using the solution provided by Occupy AI.

In this case, the customer has more precisely a high level of expertise in IT and data science methodologies, providing them with a considerable level of autonomy in using and integrating the service into their platforms.

The project description: a Flow Chart conceptualization

In this section, we will describe the different phases of the machine learning system for rooftop detection from satellite images using a flowchart.

To increase transparency in the project, we start by classifying it. Project classification helps in describing the project in a clear way. From this point on, we will define our methods. In this project, we focused on two main methods, Lean Software Development and SCRUM, and tested both approaches to determine which one would provide more reliability, a clear view, and ease of use.

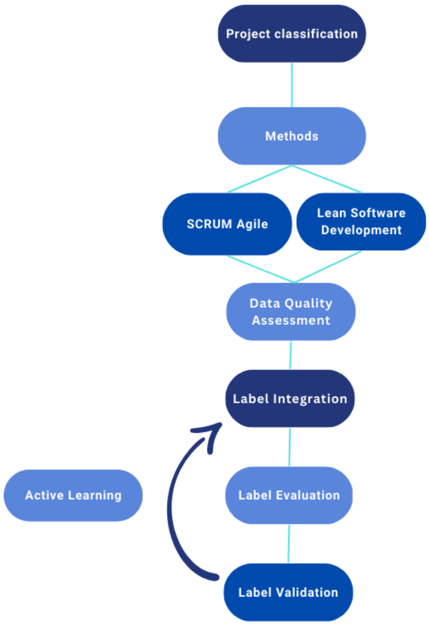

Figure 5: A flow chart representing the project description

Once the methods are defined, the data labeling phase is initiated, which is one of the crucial steps of the project. In this project, we work with raw data (images) provided by the client and label the items for further detection.

Data quality assessment is conducted to evaluate whether the data matches the quality requirements of the project. This assessment helps determine whether the data is reliable, accurate, and complete. The reliability of the data is essential for the next step, which is label integration.

Label integration involves analyzing, collecting, and reporting on the labeled data provided. Before validating the data, label evaluation is carried out to ensure that it has the correct type, quantity, and quality required. Once all these steps are completed, we move to the final step, which is label validation. It verifies all content and shows that all items are being applied to the label and that the information is correct.

The project classification: the ATTA method

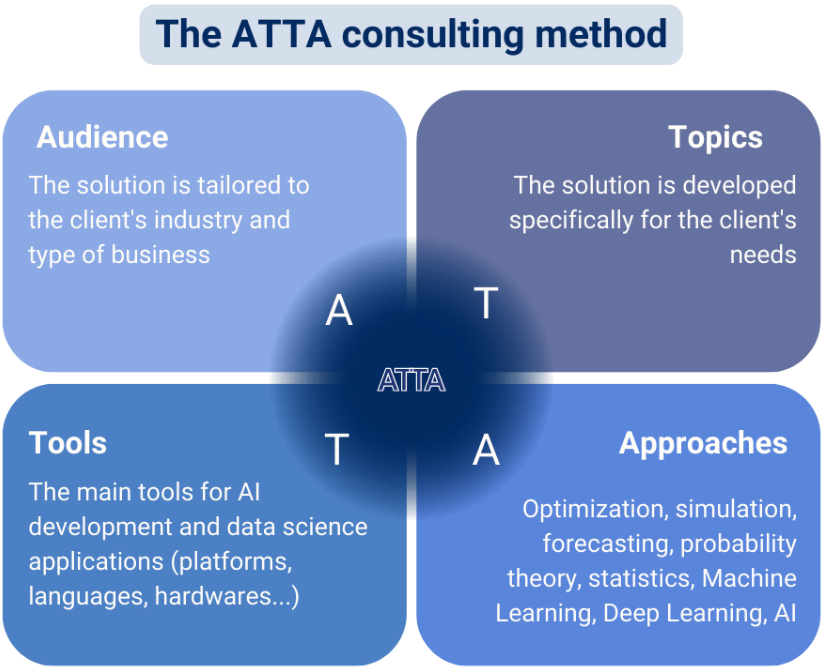

Figure 6: The Occupy AI method.

The ATTA matrix is a tool developed by Occupy AI to outline the business value the company provides. Occupy AI evaluates the customer's maturity in the field of AI and collaborates with them to define shared goals. The company offers a scalable, modular solution that fits the customer's unique needs.

The audience includes all stakeholders, including clients, shareholders, and students, whose services are addressed by Occupy AI. Topics refer to real-world sectors such as energy, transportation, health, and others, in which Occupy AI provides its services. Tools include software, programming languages, and platforms used for project development. Approaches encompass methods drawn from theoretical and applied sciences within mathematics, computer science, and engineering.

3.3 Data Acquisition and Data Quality Assessment

Data quality assessment (DQA) is the scientific and statistical evaluation of data to ensure that it meets the quality requirements for projects or business operations and is of the appropriate type and amount to support its intended purpose. DQA consists of a set of rules and strategies to define data in an application context, along with a list of procedures to analyze and enhance data quality.

Data collection involves obtaining and analyzing information on specific variables in an existing system, allowing one to answer pertinent questions and assess outcomes. Data gathering is an essential component of research in all disciplines of study, including social and physical sciences, arts, and business. Regardless of the field, the focus is on ensuring accurate and truthful data collection. The purpose of any data collection is to gather high-quality information that can be analyzed to produce convincing and trustworthy answers to the questions at hand.

Whether data is defined quantitatively or qualitatively, reliable data collection is critical to maintaining study integrity. Using appropriate data-gathering instruments (current, modified, or newly invented) and well-defined instructions for their proper usage decreases the likelihood of errors.

A formal data-gathering procedure is necessary to ensure that the obtained information is both specific and accurate. The results of such a procedure form the basis for making informed decisions. It also provides a baseline against which to evaluate and, in certain cases, an indication of what needs improvement.

DQA identifies flaws in business and technical data, enabling the company to prepare for data analysis and enrichment initiatives more effectively. This is often done to protect system integrity, quality assurance requirements, and compliance issues. Technical quality concerns, such as inconsistent formatting, standard problems, incomplete or missing data, and inaccuracies in data fields, are often easy to detect and fix, while more complex issues require more specific approaches.

DQA is typically used to resolve subjective concerns with business processes, such as the development of accurate reports, and to ensure that data-driven and data-dependent procedures are functioning as planned.

API integration in IT architecture to improve the source of data

API architects play a high-level project management role within a software development team or company. Their responsibilities are broad and varied, and a successful API architect must have advanced technical abilities, business expertise, and a strong emphasis on communication and teamwork. Multiple API projects may be running simultaneously, and the API architect should oversee the entire portfolio.

API architects are more planners than developers. They develop and maintain technology roadmaps that align with business requirements. For example, an API architect should create a reference model for the organization's service options, detailing each of them and describing how they function (Bigelow 2019).

The architect defines the API's functionalities, security setup, scalability, and monetization. The API architect also establishes best practices, standards, and analytics for API use. These rules should be updated if mistakes are identified, and better alternatives emerge. The API architect should take the lead in enterprise administration, overseeing large, enterprise-wide initiatives that promote innovation and give the company's products or services a competitive advantage (Bigelow 2019).

3.4 The IT infrastructure and data organization

In today's world, technology is the backbone of almost every organization, from individual employee work to product and service delivery. When technology is properly integrated, it can improve communication, boost efficiency, and increase productivity.

Having an adaptable, dependable, and secure IT infrastructure can help an organization achieve its goals and gain a competitive advantage in the market. Conversely, an incorrectly built IT infrastructure can result in connectivity, productivity, and security issues, such as system interruptions and breaches. Ultimately, a well-integrated infrastructure can impact the profitability of a company.

A robust IT infrastructure can help a company:

- Provide a great customer experience by offering easy accessibility to its website and online store
- Develop and market solutions rapidly
- Make speedy decisions by collecting data in real-time
- Increase staff productivity

As we process and analyze information for extended periods, it becomes challenging to recall, particularly when the material presented is vast and complex. Our brains tend to visualize objects to make them more memorable. In essence, this is what data handling accomplishes. Data handling involves collecting and presenting data in a way that our brains can comprehend.

To handle data effectively, companies must first determine what data is necessary and relevant to their business goals. This requires careful planning, as gathering and storing data that is not needed can be a waste of resources. Once relevant data has been identified, it must be organized in a way that makes it easily accessible and usable for analysis. This can involve creating a data infrastructure that includes databases, data warehouses, and data lakes.

Additionally, data handling must take into account data privacy and security concerns. Companies must ensure that sensitive data is protected from unauthorized access and that they are in compliance with data protection regulations. This includes developing and implementing data governance policies that outline how data is collected, stored, and used.

The advantages of organizing data

Data organization provides various benefits to organizations, including time-saving, eliminating potential errors, and making it easier to understand and remember. Structured data is more visually attractive, which makes it easier to interpret and recall.

- **Saving time:** With properly indexed and labeled data, employees can find the information they need more quickly and easily, reducing the time and effort required for data-related tasks.

- **Eliminating any potential mistakes:** With disorganized data, the risk of error is not zero; errors can occur either during data collection or during data representation; nevertheless, in organized data, it is ensured that the data presented is totally correct and free of errors.

- **Simple to grasp and remember:** Structured data is more visually attractive and easier to remember than raw data.

A platform for project management: the need of a time-dependent process

Monday is a powerful project management platform that provides various functionalities. Project team members can access pertinent information organized in the most effective way possible using different project perspectives. There are also various Monday time tracking integration and automation options available.

The site can interface with hundreds of different platforms and apps, which makes it easy to perform dozens of additional operations, including repetitive, conditional, but critical daily chores that will benefit all teams. Agile teams can also use the software straight away, as a template library contains a variety of processes for many businesses, ranging from software development to customer support.

3.5 Label creation and human-based validation

Data labeling or annotation is a common practice in scientific fields where raw data is identified, and one or more relevant and useful labels are applied to provide context for a machine learning algorithm to learn from. As stated by Chang, Amershi, and Kamar (2017), behavioral and social science researchers also annotate data to form hypotheses about the collected data and analyze the annotated findings to identify significant occurrences. In computer vision, image labeling is the process of applying specified tags to raw data such as movies and photographs, with each tag indicating an object class that is associated with this data. These labels are used by supervised machine learning (ML) methods to learn to recognize a particular object class from unclassified data.

However, labeling data is not a simple task, and various factors affect the quality of the labels, such as labeler expertise, the complexity of the information, and printing quality. Every piece of content must go through label validation to ensure that all intended elements are put on the label, and the information is valid. Ensuring the quality of the printing process is essential to ensure that the labels are legible, whether scanned in a distribution environment or a medical setting.

Effective label creation and validation processes are crucial for companies to build accurate and reliable machine learning models, enabling them to generate meaningful insights and drive business success. As Maayan (2022) suggests, by developing effective processes for label creation and validation, organizations can ensure that their machine learning models are built on accurate and reliable data, improving their ability to generate meaningful insights and drive business success. Therefore, organizations should carefully consider the complexities of data labeling and establish effective processes for creating and validating labels to enhance their machine learning and AI capabilities.

Roles in Machine Learning Project Management

Product manager

Product managers play a vital role in the development of machine learning (ML) systems, bringing their traditional responsibilities and skills to bear while also possessing a solid understanding of math and statistics. ML product managers must remain user-centric, focusing on customer needs and using ML only when it is well-suited to address specific problems. The ability to map problems is essential, especially given that some data problems may not be effectively solved by ML due to data structure or other factors. The presence of data engineers and data scientists on the team means that ML product managers must have a strong foundation in data literacy, able to provide ML-literate specifications, ask the right questions, and understand the feasibility of the available data.

Communication is a key to success for any product manager, and this is especially true for ML products and teams. Bridging the gap between product development and data science is crucial, and product managers must be skilled at explaining complex technical concepts in a way that is understandable to less technical stakeholders. Explainability is also essential, as it helps to build trust with customers and other stakeholders. ML product managers must ensure that the ML product is clear, coherent, and understandable, despite its inherently technical and abstract nature.

One of the most challenging aspects of managing ML systems is defining rigorous acceptance criteria for the outputs. Given the complexity of ML models, the only reliable way to assess whether the system is working as intended is to define and implement rigorous quality control measures. This includes establishing standards for data processing and output results, and regularly checking for open bugs, output precision issues, result incompleteness and inconsistency, missing data, and other issues. ML product managers should pay particular attention to acceptance criteria, which is crucial to ensure the overall quality of the ML product.

Project manager

Project managers play a crucial role in the success of any project, especially in the context of machine learning systems. They are responsible for ensuring that tasks are clearly defined, planning the work schedule in advance, allocating resources, and delegating responsibilities to different team members. The project manager is essential in monitoring the progress of the project and evaluating the team's performance to ensure the overall quality of the end product. As project management is an evolving field, project managers typically have significant experience dealing with both successful and unsuccessful projects (Lavagnon, 2009).

The project manager also has a key role in establishing the team culture and promoting effective communication channels. They are responsible for leading the team and facilitating their training and development. Balancing the needs of the group with individual needs and recognizing exceptional work is a critical component of their role. Additionally, the project manager is instrumental in addressing personal issues that may arise within the team.

In summary, the project manager is responsible for overseeing the entire project, from planning and execution to monitoring and evaluation, and for establishing a positive team culture that promotes effective communication and personal growth. With their extensive experience and expertise, project managers are critical in ensuring the success of machine learning projects (Hjertaker and Besirovic 2022).

Label analyst

Data labeling is a critical component of machine learning systems, and label analysts play a key role in the process. They are responsible for providing labeled data to train models, helping the algorithms to recognize and classify images. Using a labeling platform, analysts create bounding boxes around specific areas of images and label them accordingly. This can involve labeling both still photos and video data, requiring the analyst to track an object's movement throughout the footage.

Data labeling for image recognition demands a high level of skill and attention to detail. For instance, analysts must ensure that the bounding box is precisely around the region of the image where the model is to identify the features specified in the label, such as "tree," "bicycle," or "cat." Including too much or too little of the image in the bounding box can lead to inaccurate outputs. Additionally, label analysts must maintain consistency and accuracy in their work, providing high-quality labeled data to support the development of the machine learning model.

3.6 Introduction to model validation

Label validation

Different projects require varying measures of data quality. Nevertheless, many data researchers and scientists generally agree on certain aspects of high-quality datasets that they evaluate for big data initiatives.

First and foremost, the dataset itself is crucial. The diversity and balance of data points within it indicate how well the algorithm will anticipate future comparable points and patterns. For instance, consider an autonomous vehicle-trained model that is designed to teach AI how to distinguish between moving and stationary cars. If the dataset comprises 90% photographs of moving cars but only 10% of images of stationary cars, it is considered uneven, which increases the possibility of inaccuracy. To address this issue, methods such as oversampling, downsampling, and weight balancing are used.

Second, the precision with which labels and categories are assigned to each data point is frequently used to define the quality of datasets for model training. However, it is not just the quality of data labeling that is important, but also the frequency of its accuracy. Throughout the assurance process, both data correctness and consistency are measured using different phases that can be conducted manually or automatically. Several procedures are typically combined to cross-check and ensure the final faultlessness of the provided dataset.

Active learning phase

During the active learning phase of machine learning, model validation plays a critical role. Its proper use helps in determining the effectiveness of the machine learning model to respond to new data. This approach provides two benefits: it helps identify the appropriate method and parameters to use, and it prevents overfitting during the training phase.

To address a problem with a dataset, selecting the best machine learning technique to build the model is crucial. Each design has its advantages and disadvantages, with some algorithms being more tolerant of small datasets while others thrive with vast amounts of high-dimensional data. Hence, two models using the same data can produce different outcomes and varying degrees of accuracy.

Determining the best model for a given dataset is a collaborative effort that involves experimenting with different strategies to minimize model error. Hyperparameters, which are the parameters that govern the behavior of a machine learning algorithm, can significantly impact the final model. By adjusting the values of hyperparameters, you may discover other and potentially better models.

Without model validation, it is easy to adjust the model to the point where it starts overfitting, unbeknownst to the user. While the training method modifies parameters to reduce a loss function, it can sometimes go too far, resulting in an overfit model that is highly complex and unable to perform well with new data.

One way to verify how well a model will perform with new data is by using model validation. This involves splitting the dataset into two parts, one to train the algorithm and the other to test it. Model validation is a widely used approach to prevent overfitting during the training phase, as it does not use all the data to develop the model.

CHAPTER 4 - CASE STUDY: NOTATION SERVICE FOR SATELLITE IMAGES

4.1 Project description

The goal of the project between Occupy AI and its client is to automate rooftop evaluations using Artificial Intelligence (AI) to drastically reduce the cost of the standard rooftop assessment process. This process is commonly time-consuming and expensive, often taking up to 2 full days to calculate the potential of each rooftop and resulting in high costs. For instance, cost of sales can take up to 30–40% of total project costs.

By utilizing machine learning and computer vision models, the primary objective is to detect rooftops in a given image and determine their type and structure, such as flat, hip-roof, shed-roof, and others, which presents an instance segmentation problem. To achieve this, the project aims to combine multiple models to automatically identify rooftops and detect rooftop features like obstacles, material, slopes, and area from high-resolution satellite imagery using machine learning.

Satellite images (also called earth observation imagery, spaceborne photography, or simply satellite photos) are images of the earth collected by imaging satellites operated by governments and businesses around the world.

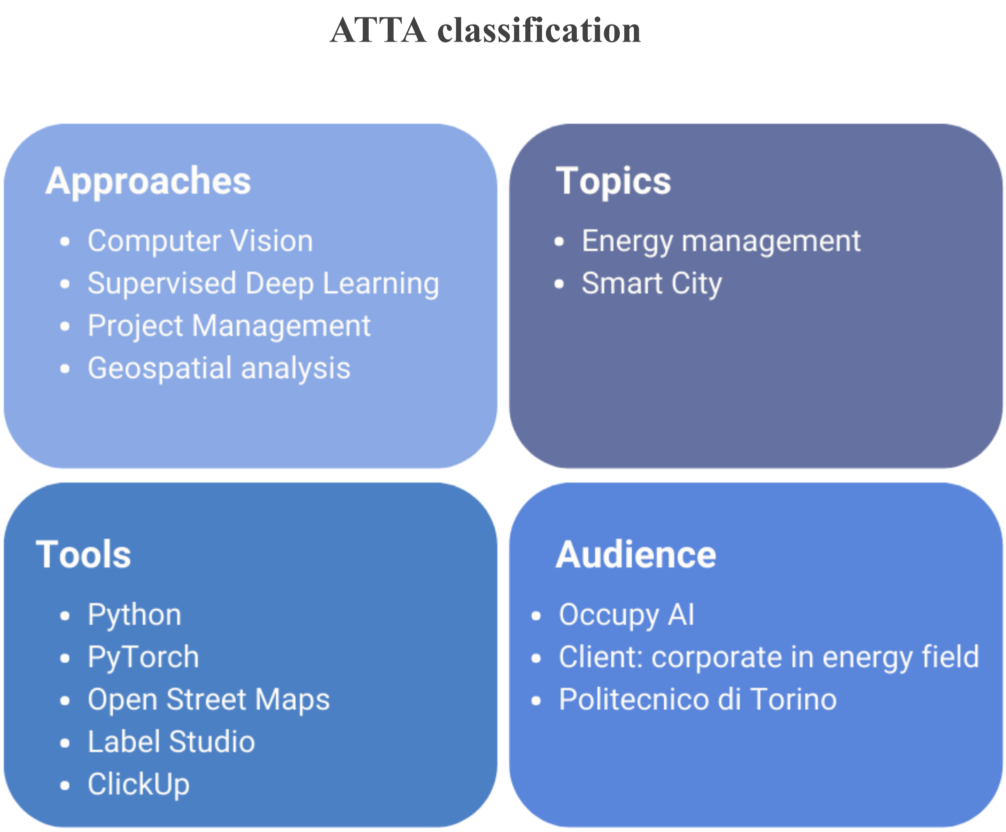

Figure 7: The ATTA classification of Occupy AI project

In our project, as previously mentioned in the project overview, we utilize the ATTA matrix, focusing on four main approaches: computer vision, supervised deep learning, project management, and geospatial analysis. The topics being the sectors for which the projects can apply to, are here considered to be smart cities and energy management.

In terms of tools, we require a programming language and machine learning library, which are respectively Python and Pytorch. We utilize Open Street Maps as a metadata repository to provide information on the satellite images provided by the client.

Label Studio is the primary platform used for data labeling, while Click Up is utilized for managing and tracking the workflow. There are three main stakeholders involved in this project: the university, the company Occupy AI, and the client.

Supervised learning applied to rooftop detection

In the project undertaken by Occupy AI, the primary goal is to detect rooftops in satellite images and determine their type and structure using machine learning and computer vision models. To achieve this objective, one of the key techniques employed is supervised learning. In this paragraph, we will discuss the application of instance segmentation, a type of supervised learning, to the problem of rooftop detection in satellite images. Specifically, we will explore how instance segmentation is used to identify different types of rooftops and their specific features.

Supervised learning is a widely-used family of techniques in machine learning and artificial intelligence that allows models to learn from labeled datasets and accurately classify data or predict outcomes. In our project, we utilize supervised learning to detect rooftops in a given satellite image and determine their type and structure, such as flat, hip-roof, shed-roof, and others, presenting an instance segmentation problem. To achieve this, we rely on human labelers to annotate the data, providing a labeled dataset to train our model.

The supervised learning process involves using a training set to educate the model to produce the desired output. The training dataset contains both correct and incorrect outputs, allowing the model to evolve over time. During the crossover validation process, the weights of the model are adjusted until it is well-fitted to the data. The algorithm evaluates its accuracy using a loss function and makes adjustments until the error is sufficiently reduced.

In supervised learning, there are two types of issues: classification and regression. Classification is used to allocate test results to specific groups correctly. It identifies particular entities inside the dataset and tries to derive inferences about how those items must be labeled or described. Common classification techniques include proposed systems, support vector machines (SVM), selection trees, k-nearest neighbors, and random forest. Regression, on the other hand, is a statistical method used to determine the relationship between independent and dependent variables. It is widely used to produce forecasts, such as those for a company's sales revenue. While not directly applicable to our project, regression may be useful for other aspects of sustainable energy management.

In our project, we find ourselves in the computer vision domain, as image data are used to train a model. In particular, we use instance segmentation to identify the type and structure of each rooftop, which is a classification problem. As a classification technique, we use convolutional neural networks, which we will detail further in this chapter.

4.2 Data Acquisition and Data Quality Assessment

In any machine learning project, the quality of the acquired data plays a crucial role in determining the accuracy and reliability of the resulting model: we placed a great emphasis on the quality assessment of the acquired images. In this section, we discuss data acquisition and quality assessment for our rooftop detection project.

Acquisition and enhancing of the data

The image acquisition process involves several steps. Usually, for single-shot images (one image representing a bounding box over a geographical location), we rely on the download process set by the image supplier. For images that are a mosaic of many satellite or aerial shots captured over a period of time, we are provided with a time range to illustrate the periods the images were captured between. In cases where there is no or limited date information available, we can use a start and finish date to estimate the period within which the photograph was shot.

In the first part of the project, our client provided us with approximately 250,000 satellite images on a hard disk. In a second phase, we were given a Cloud storage engine connecting private storage of our client with a shared server running Label Studio. To enhance the data, we used Open Street Maps to add geographical metadata to the images. For labeling the images, we utilized Label Studio, an open source labeling software.

Data Quality Assessment

As part of the data quality assessment, we performed a review of the images to ensure that they were of good quality and free of errors or artifacts that could negatively impact our machine learning models. Data quality assessment is a crucial step in any data-driven project, especially those involving machine learning and computer vision models. In our project, we employed a thorough approach to assess the quality of the acquired satellite images, and conducted a visual inspection of the images. Specifically, we checked for image resolution, contrast, brightness, and color balance. We also checked for any missing or incomplete data in the images, such as areas that were obscured by clouds or other obstructions. In cases where the quality of the images was not satisfactory, we either replaced them with better quality images or excluded them from our dataset.

Data quality generally needs to meet six dimensions: accuracy, completeness, consistency, timeliness, validity, and uniqueness. We are not allowed to share the precise DQA report, as it corresponds to a product of the company protected by intellectual property.

Data cleaning and preprocessing

We also performed data cleaning and preprocessing to ensure that the images were in a suitable format for our machine learning algorithms. We used various techniques to remove noise and outliers from the images, such as thresholding, filtering, and morphological operations. We also normalized the images to ensure consistency in their size, orientation, and scale. Overall, our data quality assessment process ensured that the acquired satellite images were of high quality and suitable for our machine learning and computer vision models.

Meta-data Assessment for Rooftop Detection and Classification Project

The dataset for our project consists of satellite photos of rooftops and related metadata, with the metadata stored in a MySQL database and the images stored on a hard disk.

The MySQL dataset comprises multiple tables, but for the purposes of this research, we will focus on only three tables: "positions," "position images," and "obstacles." A data model was provided for these tables, highlighting the key elements to be examined.

The dataset includes three data frames, which are linked to the following three dictionaries (with description and size given by “entry x fields or features”):

- "positions" (identified rooftops, 309,381 entries)
- "position_images" (satellite photos and metadata, 288,086 entries)
- "obstacles" (349,725 entries)

The "positions" table specifies the rooftops identified in satellite photos provided by the client, which have been subsequently tagged. Importantly, only some of the roofs in each image have been classified, resulting in photographs with both labeled and unlabeled rooftops.

The features describe the type of roof, as well as its properties such as height, length, breadth, area, azimuth, cover, and sheet materials. Unfortunately, a significant amount of data is missing in many of these aspects. For the purposes of this report, we will focus only on the columns included in the data description.

Analyzed metadata features	
Available by dictionary “positions”	Description
id	ID of each roof
project_id .	ID of the project where the roof was installed
roof_type	Type of roof.
height	Height of the roof (i.e. depth), in meters.
length	Length of the roof, in meters.
width	Width of the rooftops, in meters.
area .	Product of length and width, in square meters
slope	Slope of the roof, in degrees.
roof_cover	The material of which the roof is made.
obstacle_count	The number of obstacles that have been labeled on one roof.
pixelCoordinates	The pixels that define the boundary of the roof.

The project's database contains a table called "position_images," where each row represents a satellite picture that includes rooftops. The table contains metadata for each image, such as zoom level (up to 22), longitude and latitude of the image center, and the filename that it links to ("imageURL"). It is important to note that this table does not include all the satellite photos collected for the project.

In fact, there are over 500,000 photographs, and only slightly more than half are included in "position_images." The majority of the roof photos are from continental Europe, but the project has at least one dataset from each continent.

Analyzed metadata features	
Available by dictionary “position_images”	Description
position_id	ID of the roof to which the image belongs to (foreign key to the "positions" table).
id	(redundant) row data identifier.
imageURL	Linking pictures in "imageURL" to the rooftops identified in the "positions" table (they can repeat themselves, as there can be more than one roof in an image).
scale	Empty and dropped.
rotation	Empty and dropped.
type	Constant ("satelliteImage"), can be dropped.

The last table, "obstacles", describes the obstacles that have been tagged on the rooftops.

Analyzed metadata features	
Available by dictionary “obstacles”	Description
id	ID of the obstacle
position id	ID of the roof the obstacle belongs to.
pixelCoordinates	The pixels that bound the obstacle.

The scientific literature uses datasets of hundreds of thousands of photos to develop a classification method: the data appears sufficient from this perspective. To be more specific, 288,841 roofs have been classified in 198,452 photos. In 64,408 satellite photos, at least one barrier has been found.

The overall number of photos is larger than 500,000; information is accessible for nearly half of them, and around 40% of the satellite images have at least one roof tagged. In other terms, not even all rooftops and barriers have been labeled in every photo: this will undoubtedly affect model training, albeit it is impossible to predict how much the model will be impacted.

To summarize

To summarize, the data for our project comprises satellite photos of rooftops, with metadata stored in a MySQL database and images stored. The MySQL dataset contains three tables, with the "positions" and "position_images" tables being the most relevant for the project. The dataset includes data on over 288,000 roofs, and over 64,000 satellite photos have at least one barrier. However, not all rooftops and obstacles have been labeled in every photo, which may impact model training. Yet the DQA has shown that the information available in the client database is not enough to start a ML pipeline: for this reason, we had to enrich this data with more geospatial information and human based labeling to create a more reliable database.

Open Street Maps as a metadata repository for the project

In the context of our project management for machine learning systems, OpenStreetMap (OSM) was thus used as a metadata repository to provide geographic information for the rooftop detection project. A geographic information system (GIS) is a database that includes geographic data, as well as software tools for organizing, analyzing, and displaying such data. It allows for the manipulation and analysis of geographical data, including information on land use, natural resources, and human demographics.

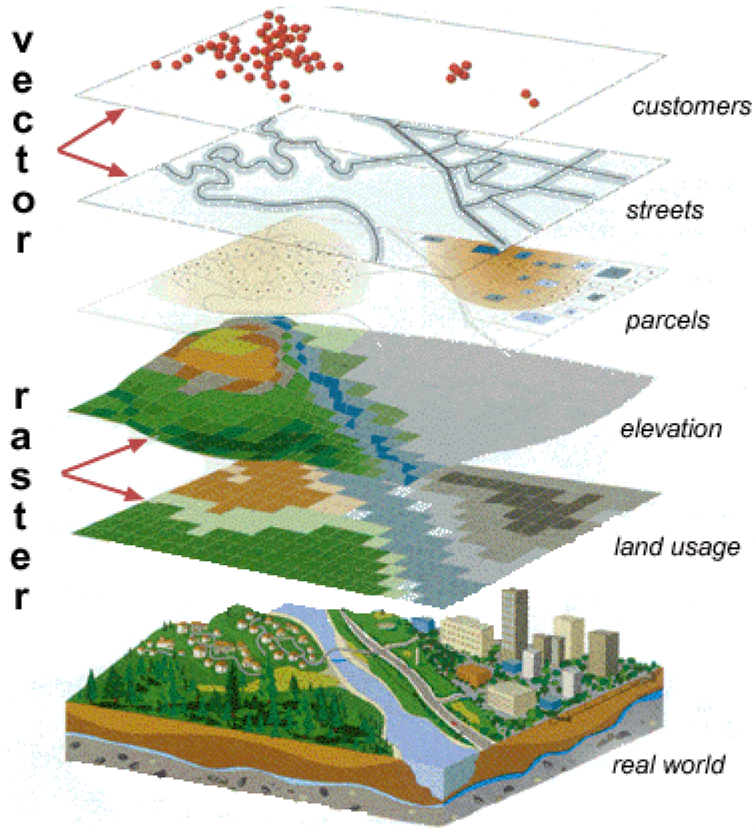

Figure 8: GIS Concept

In figure 8, we represent the basics of GIS. The vector layers use specific lines and point segments to identify location on earth, and they are the customers and streets. Then there is the parcels layer which is related to the boundaries and demographics. Elevation, land usage, landmarks, and water features are all under the raster layer's name where they use a series of cells to identify location on earth. Lastly, the real world layer which is everything we see in our eyes.

OpenStreetMap is a free, open-source geographic database maintained by a community of volunteers who collect data from surveys, trace from aerial imagery, and import from other freely licensed geodata sources. It is licensed under the Open Database License and commonly used to make electronic maps, inform turn-by-turn navigation, assist in humanitarian aid, and data visualization. This information is then used by researchers, developers, and other stakeholders to build tools and applications that utilize spatial data.

In addition, OSM's website serves as an online map, geodata search engine, and editor. Geospatial metadata, or geographic metadata, contains information about geographic data or information, such as the content of a message or a picture. This metadata is maintained in a geographic information system (GIS) and is useful for storing papers, data sets, photographs, or other objects, services, or similar items that occur in another native context but whose characteristics may be useful to represent in a (geographic) metadata catalog.

In our project, we used OpenStreetMap to add geographical metadata to the satellite images provided by the client. This metadata was essential in identifying and classifying rooftops and their structures accurately. Through OSM, we were able to obtain information on the height, size, and shape of buildings, as well as their locations and surrounding land use. The use of OSM allowed us to enhance the accuracy and completeness of our data by incorporating additional information beyond what was available in the satellite images alone.

With OSM, we were able to add location information to the images, enabling us to better organize and analyze the data. The use of OSM allowed us to tap into its own topology for storing geographical features, which can be exported into other GIS file formats. However, the metadata of OSM alone are not sufficient for the scope of building a reliable database, and needs to be integrated with human-labeled data.

4.3 IT infrastructure and data organization

The IT infrastructure and data organization play a critical role in the success of this project. As the project deals with large volumes of satellite images and related metadata, it is essential to have a robust IT infrastructure that can handle the storage, processing, and analysis of these data. Effective data organization is equally important, as it enables the project team to access, manage, and analyze the data efficiently. With proper data organization, the team can easily identify the relevant data and attributes required for analysis, ensuring accuracy, completeness, and consistency. A well-designed IT infrastructure and data organization system also promotes collaboration among team members and improves the project's overall efficiency. Additionally, effective data management practices reduce the risk of data loss or corruption, which is critical for a project of this magnitude. Ultimately, an efficient IT infrastructure and data organization system can significantly contribute to the project's success by enabling the team to achieve its goals effectively and efficiently.

Data storage

A hard disk was used to store satellite pictures and metadata. In order to complete out the prototype efforts, OAI is giving resources and creating solutions for the maintenance and usage of this database. Throughout the next phases, the final storage option will be determined with the customer.

The SQL database discard takes up around 1.6 GB of storage space, however, the actual data required is significantly less, as there are only three tables of importance. Moreover, after the aspects have been selected, the design matrix could be kept in a simple parquet file or solutions.

But dealing with the entire collection of images locally is more difficult: the entire dataset requires 400 GB of storage, which must be handled in stages. Each batch of the dataset must be downloaded locally, code must be run on it, and then the batch must be removed to handle the storage requirements. For this reason, in agreement with the client, we set up a connection between the private cloud storage of this same client and the server which we use to run any analysis and which contains the Label Studio instance we describe in next sections.

The workflow management tools

Label Studio

In addition to the IT infrastructure and data organization, we utilized several tools to manage and streamline our project workflow.

For our project on a-priori labeling, we utilized Label Studio, an open-source data labeling application that can be customized to suit various data types and processes. Label Studio provides a variety of features to improve labeling efficiency, including customizable layouts and templates, integration with ML/AI pipelines via webhooks, Python SDK, and API, and the ability to incorporate ML backend predictions into the labeling process.

In addition, Label Studio offers cloud storage options with the main providers (Amazon S3, Microsoft Azure, Google Cloud) to streamline data labeling. It also includes a powerful Data Manager tool, which enables users to create and manage data using various filters.

Label Studio supports numerous projects, use cases, and data types on a single platform, and can be used for any form of data. In our case, we focused on computer vision and used Label Studio for image classification, object recognition with support for boxes, polygons, circles, and keypoints, and semantic segmentation. We employed machine learning models to pre-label and enhance the labeling process.

Overall, Label Studio's configurable and adaptable features, integration with ML/AI pipelines, cloud storage options, and powerful data management capabilities make it an effective tool for data labeling, particularly in the context of machine learning projects.

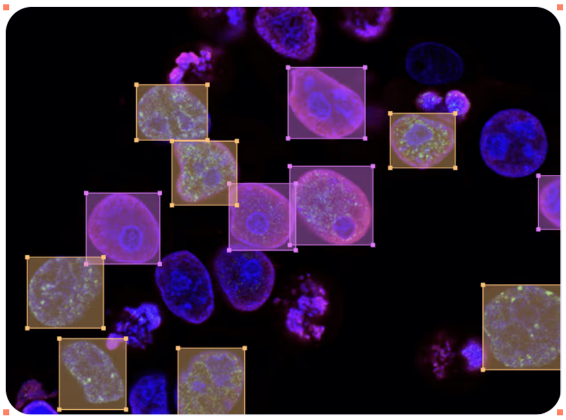

Figure 9: image showing how Label Studio works on computer vision

ClickUp

Another tool we used was ClickUp, a project management software that helped us keep track of tasks, deadlines, and team communication. This platform allowed us to easily assign tasks, collaborate on projects, and monitor progress.

What is ClickUp?

ClickUp is indeed a cloud-based project management application that strikes a balancing act between offering a wide range of capabilities while remaining cost-effective and simple to use.It enables teams to communicate and collaborate on common tasks through postings and project updates, leading to better alignment of processes with objectives.

As the primary hub for team collaboration, ClickUp provides a range of tools such as tasks, documents, chat, objectives, whiteboards, and more. Let's take a closer look at some of the key features of ClickUp:

1. **Time Management:**

Effective time management starts with knowing what needs to be done. ClickUp helps you prioritize tasks by organizing them and prioritizing those that must be completed first. ClickUp's time management features are user-friendly for project managers and business owners, enabling streamlined processes through more seamless communication.

2. **Mobile Compatibility:**

ClickUp mobile enables users to stay up-to-date on projects from anywhere. The mobile experience provides the same value to users as the desktop experience, without compromising on quality or usability.

3. **Tagged Comments:**

When someone is tagged in a comment, their name appears highlighted in their notification stream. They are also automatically designated as a "watcher," receiving alerts whenever the task changes. This feature helps team members prioritize the most critical action items and improve monitoring of vital project activities.

4. **Perspectives and Collaboration:**

ClickUp offers nine distinct view types, including list, board, table, box, calendar perspective, Gantt, activity, timeline, and workload view. Users can also create free-form thought maps by selecting the mind map view. ClickUp promotes collaboration through team monitoring, real-time editing, real-time synchronization, comment editing, mentions, and multiple assignees. The program also supports emojis, snapshot editing, comment assignment, debates, comment reminders, sharing, and quoting.

ClickUp as a project management tracker and organizer

ClickUp was used as a project management tool to keep track of tasks, deadlines, and progress. The platform allowed the team to create and organize tasks in various formats such as lists, boards, and timelines. Each task was assigned to team members responsible for its completion, with due dates and priorities set to ensure timely delivery. ClickUp was also used to track the progress of each task, with regular updates from team members on their status.

In addition, ClickUp was used to store important project documentation, such as meeting notes and technical specifications, making it easily accessible to the entire team. The platform's collaboration features allowed team members to comment on tasks, ask questions, and provide feedback to each other.

ClickUp's integration with other tools such as Slack and Google Drive allowed for seamless communication and file sharing among team members. Overall, ClickUp provided a centralized platform for project management, helping the team to stay organized and on track throughout the project lifecycle.

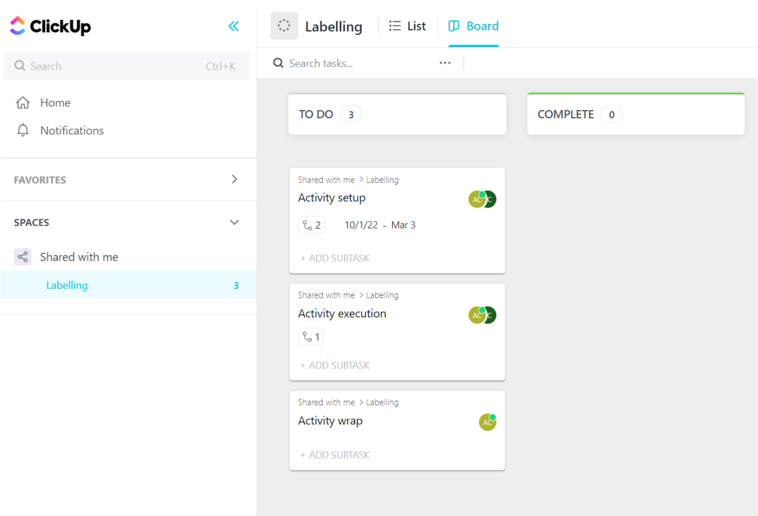

Figure 10: example of the ClickUp dashboard monitoring and organizing labeling tasks

Slack

Finally, we used Slack as our primary communication tool, allowing us to quickly and efficiently communicate with each other and stay up to date on project updates and progress.

Slack is a professional communication app that enables you to send messages and share files with your colleagues. Its core feature is to connect coworkers through direct messaging or channels, which are group chats. While there are add-ons available, they are not necessary for using Slack.

Slack can help your team communicate more effectively by combining email, texting, and instant messaging into a single application. Its mobile and desktop versions enable your team to stay connected and manage their tasks from anywhere, whether they are in the field office, at home, or on the go.
However, Slack may not be suitable for every team's working style, and security is also a consideration. With proper setup and maintenance of your Slack workspace, it can provide your team with a sense of community, regardless of when or where they work.

Slack is often used for the following purposes:

- Keeping vital information in a centralized location rather than scattered across individual email accounts.
- Sharing files, including photos, text, and videos, with colleagues.
- Mentioning colleagues' names to send reminders to their smartphone or computer and attract their attention.
- Starting a video or voice call with your Slack team members.
- Setting reminders for yourself and others, with built-in notifications, Google Calendar, and multiple to-do lists to help volunteers stay on top of deadlines.
- Establishing a fun community that your volunteers and staff can access regardless of their location.
- Supporting rapid inquiries and decision-making, real-time collaboration, spontaneous audio or video conferences, capturing someone's attention quickly, quick voting or polling, keeping everyone engaged, and instant onboarding of new team contributors and members.

Slack, the project’s primary communication channel

Slack was used as the primary communication tool for the project. The team created channels to discuss various topics related to the project, such as data collection, data labeling, model development, and project management. Channels were also created for specific tasks or issues that needed attention, such as bug fixes or feature requests. This allowed team members to communicate and collaborate effectively in real-time. Additionally, Slack was integrated with other tools such as GitHub, ClickUp, and Label Studio, providing a centralized platform for the team to access all project-related information and updates. Slack also allowed for direct messaging between team members for one-on-one communication or to discuss sensitive or confidential information.

4.4 Label creation and human-based validation

In the project with the company Occupy AI, the task of detecting rooftops on satellite images was accomplished through a supervised machine learning approach. To ensure the accuracy of the model, it was essential to have high-quality labels. The process of label creation involved employing data labelers to annotate the images provided by the client. While manual labeling can be time-consuming and expensive, it remains the gold standard for labeling complex data such as satellite images. The labeling task was performed on a platform called Label Studio, which allowed the labelers to draw bounding boxes around the rooftops accurately, according to the roof type they could observe. Below, you can see an example of an annotated image: boxes are drawn joining points together, and represent either obstacles, hip roof, gable roof, monopitch or flat roof, as well as unknown surfaces.

It was equally crucial to have human-based validation to ensure the correctness of the labels. Therefore, the team of data labelers worked closely with the Occupy AI team, which was coordinating their activities. The company itself worked with the client to develop a set of guidelines to ensure consistency in the labeling process, and make the process as smooth as possible for labelers.

After the labels were created, a team of validators reviewed them to ensure that the bounding boxes accurately captured the rooftops. This human-based validation process ensured that the model was trained on high-quality data and that it would have a high degree of accuracy when deployed. This project was especially critical for a company in the energy sector, where rooftop detection is essential for optimizing the activities -the reveal of the precise scope of the project is strictly forbidden for preserving corporate activity. The process of data validation is further explained in the following parts.

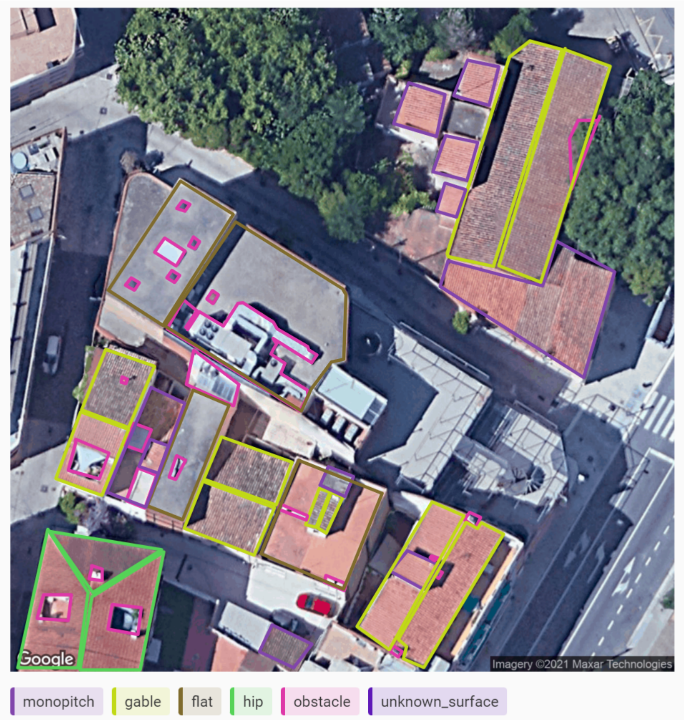

Figure 11: annotated image on Label Studio

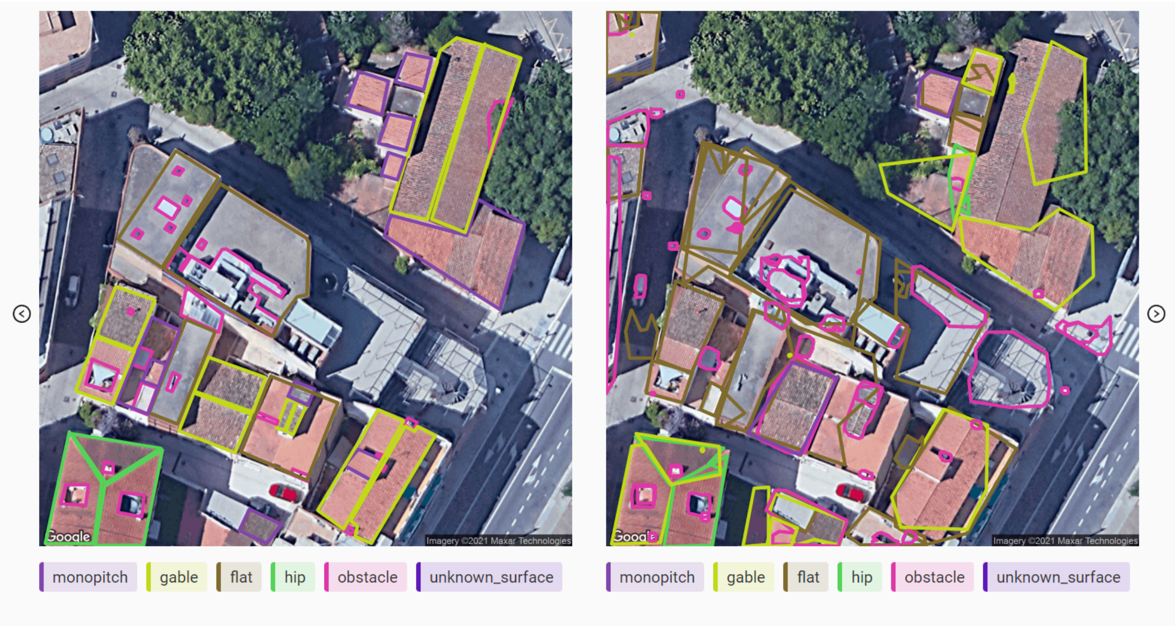

Figure 12: comparison between algorithm-labeled image and human-labeled image on Label Studio

4.5 Tuning and validation of the model

Tuning and validation of a machine learning model is a crucial step in the development of any computer vision project. The effectiveness of a model's ability to accurately detect and classify objects is directly related to the quality and quantity of the data used for training and validation. In the case of roof detection on satellite images, the availability of high-quality and properly labeled data is particularly important. This supervised machine learning project requires human label creation and validation to ensure the accuracy of the model's predictions.

To achieve accurate roof detection, and as we already explained, the project with Occupy AI involved employing data labelers for annotating satellite images provided by a client on the platform called Label Studio. The annotated images were then used to train a computer vision model, which was then fine-tuned using various optimization techniques such as data augmentation and hyperparameter tuning. The goal was to create a model that could accurately identify rooftops on satellite images with a high degree of intersection over union, the quality metric used.

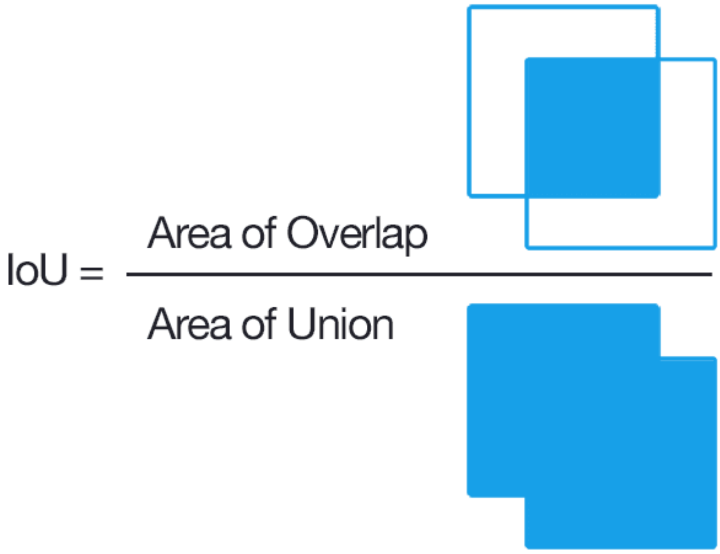

Figure 13: Intersection over Union metric explanation. Credits: PyImageSearch, Adrian Rosebrock

One of the major challenges faced by the project was the identification of small and complex rooftops that were difficult to detect using traditional computer vision algorithms. Additionally, obstacles such as trees, windows and chimneys were an issue in identifying. To address this, the team employed a deep learning approach using transfer learning on convolutional neural networks (CNNs), which have shown to be highly effective in object detection tasks. Transfer learning involves fine-tuning an existing pre-trained CNN on a new task, to achieve better performance with a limited amount of labeled data, as explained in the dedicated part of this thesis.

In the project with Occupy AI, convolutional neural networks (CNNs) played a central role in the development of the roof detection model. The CNN architecture is designed to automatically learn relevant features from images, which can be used for classification or detection tasks. CNNs consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers. In the convolutional layer, a set of learnable filters are applied to the input image to extract features at different spatial scales. The pooling layer then downsamples the feature maps to reduce computational complexity and increase translation invariance. Finally, the fully connected layers use the extracted features to perform classification or regression tasks.

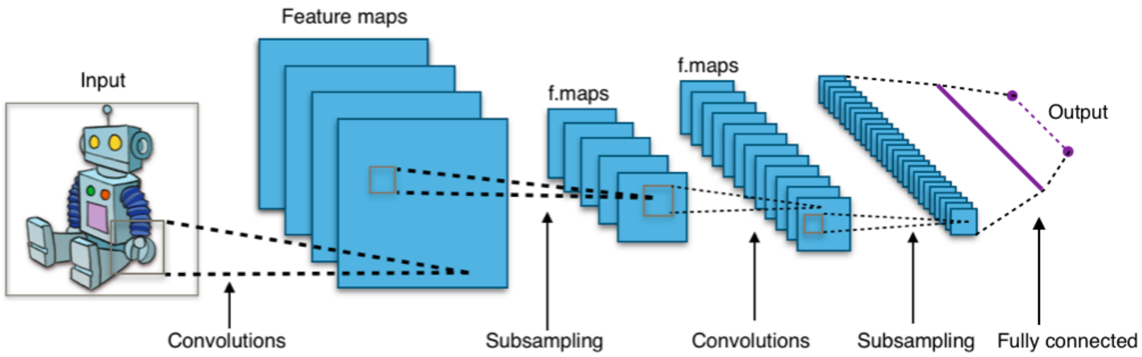

Figure 14: a CNN algorithm. Credits: Wikipedia Commons

The importance of human-based validation cannot be overstated in this project. The process of data labeling and validation was a critical component in ensuring the accuracy of the model's predictions. Human labelers were employed to manually annotate images and validate the accuracy of the model's predictions. The iterative process of model tuning and validation allowed the team to continually improve the accuracy of the model's predictions, ultimately leading to a highly effective roof detection model.

Overall, the project with Occupy AI demonstrates the importance of proper data labeling and validation in the development of effective machine learning models for computer vision tasks. The use of deep learning and human validation allowed for the development of a highly accurate model for rooftop detection, with potential applications in the sustainable energy sector for identifying suitable locations for installations proper to the demands of the client.

4.6 Label validation for quality of the service

Validating all the labeled data can be time-consuming, expensive, and may not be practical in many situations, especially when the labeled data is voluminous. In contrast, validating only a small percentage of the labeled data may not be enough to capture potential errors or inconsistencies in the labeling process.

In the project, label validation played a crucial role in ensuring the accuracy and quality of the labeled data used to train and validate the roof detection model. Human labelers were employed, and a process of quality control was implemented to ensure the accuracy and consistency of the labels.

Random sample checks of labeled data

One approach to ensuring label quality was to randomly sample a percentage of the labeled data for a manual check of the labels. For example, the team could randomly select 10% of the labeled images and manually review the annotations to ensure that they were accurate and consistent with the project requirements. The sample check ratio can vary based on the project requirements and can provide an estimate of the overall label quality: if the labeled data is particularly complex, or if the labeling task requires a high degree of accuracy, the sample check ratio may need to be increased. Conversely, if the labeled data is relatively simple, or if the labeling task requires less accuracy, the sample check ratio may be decreased. In the project, the 10% were chosen based on a balance between the need to validate a representative sample of the labeled data and the practicality of manual validation.
The whole process of data validation could be repeated periodically throughout the labeling process to ensure ongoing quality control and maintain the accuracy of the labeled data.

Another approach to ensuring label quality is to use inter-rater reliability measures, such as Cohen's kappa coefficient or Fleiss' kappa coefficient. These measures compare the agreement between multiple human labelers and can provide an objective measure of label quality. A high kappa score indicates that the raters are in good agreement, while a low score indicates that the raters are not in agreement, and the labels may need to be reevaluated or the raters may need additional training. The labelers can indeed be retrained or replaced if their agreement falls below an acceptable threshold.

Feedback loops to improve labels quality

Furthermore, the project also employed a feedback loop to continually improve the quality of the labeled data. The feedback loop involved providing regular feedback to the labelers on the accuracy of their annotations and using this feedback to improve their performance. This iterative process helped to ensure ongoing label quality and improve the overall accuracy of the roof detection model.

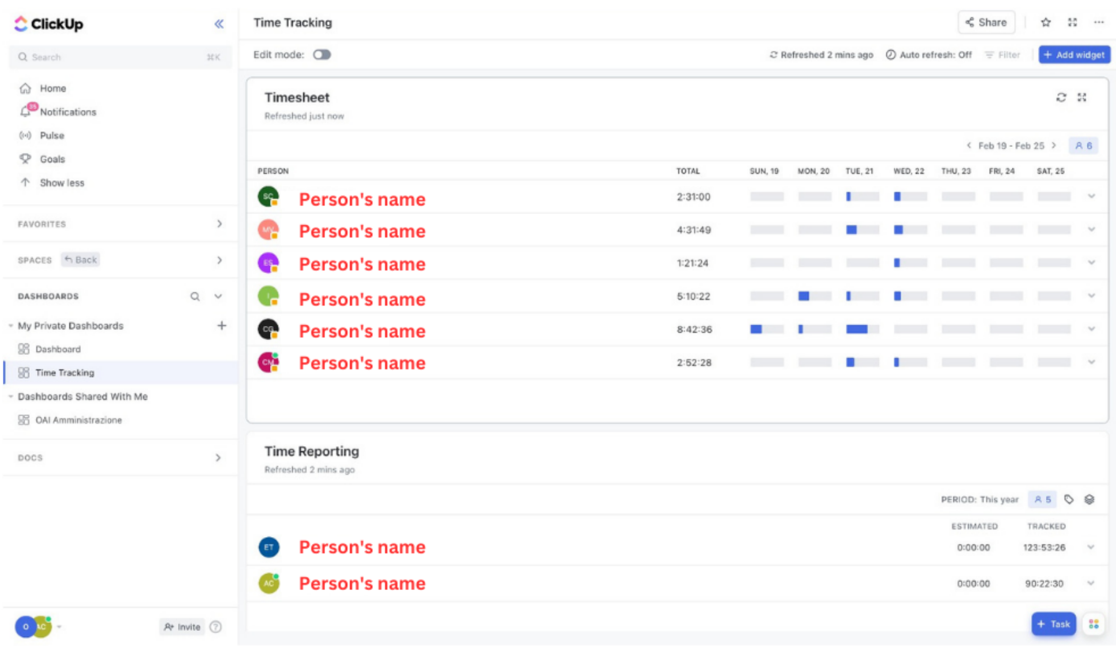

Figure 15: the time-tracking monitoring dashboard of ClickUp

This feedback loop was greatly supported by the use of ClickUp, a project management tool that can be used to track and analyze the performance of data labelers working on the Label Studio platform. ClickUp provides a dashboard that allows project managers to monitor the progress and accuracy of the labeling task, as well as to track the performance of individual labelers. project managers can set up tasks and assign them to individual labelers, track the time spent on each task, and monitor the accuracy of the labeled data. ClickUp also allows project managers to create custom fields and tags to track additional information, such as the type of labeling task or the expertise level of the labeler.

In the case of the roof detection project with Occupy AI, ClickUp was used to monitor the progress of the labeling task and to track the performance of individual labelers. The data collected through ClickUp was used to generate analytics and insights into the performance of the labeling task and the labelers. This allowed the project managers to identify areas for improvement and to provide feedback and guidance to the labelers as needed, ultimately leading to higher quality and more consistent labeled data.

User-friendly interface for data validation

Providing a user-friendly interface for data validation is essential to ensure the accuracy and consistency of labeled data. An interface for data validation can help to streamline the manual validation process and reduce the likelihood of errors and inconsistencies in the labeled data.

The web interface Label Studio provided by the client is an example of such an interface. It provides a user-friendly and intuitive interface for validating labeled data, making it easier for human labelers to review and validate the labeled data. It allows for the creation of different labeling tasks and the use of different types of labels and annotation tools, such as polygons or squares.The interface allows labelers to view the labeled data and the corresponding labels, and to check the accuracy of the labels by comparing them to the original data and the other annotations made by labelers.

The web interface also provided a dashboard to monitor the progress of the labeling task and to track the accuracy and consistency of the labeled data. The interface displayed the percentage of labeled data that had been validated and the percentage of validated data that was accurate. This allowed the project managers to monitor the quality of the labeled data and to adjust the sample check ratio or other aspects of the labeling process as needed.

The Label Studio interface also allows labelers to provide feedback on the accuracy of the labeled data and to suggest improvements or corrections, which can help to improve the overall quality of the labeled data. Additionally, the interface can be customized to meet the specific needs of the project, providing additional tools and features to facilitate the validation process.

In summary

Label validation and quality control are essential for ensuring the accuracy and consistency of the labeled data used in supervised machine learning. The use of inter-rater reliability measures, sample check ratios, great user-friendly interface and feedback loops can help ensure ongoing quality control and improve the overall accuracy of the labeled data, which can lead to improved model performance and better outcomes for the end-users. By leveraging the power of supervised machine learning and the expertise of human labelers and validators, the project team was able to create a robust model for rooftop detection, helping to drive sustainable energy practices forward.

CHAPTER 5 - CONCLUSION

In conclusion, this paper has explored the importance of project management in machine learning initiatives, specifically in the context of rooftop detection using satellite images. It mixes a strong theoretical foundation, particularly in chapters 1, 2, and 3, which provide a comprehensive overview of the topic of project management in machine learning, and a practical approach, brought by the case study of chapter 4.

Along the chapters, this thesis has presented a comprehensive approach for managing machine learning projects with a focus on labeling data and applying computer vision techniques for analyzing satellite images. Chapter 1 introduces the topic and provides an overview of the key components of project management in the context of machine learning, such as labeling data and computer vision for satellite images.

Chapter 2 delves deeper into the theoretical aspects of project management, with a thorough review of existing research papers and project management approaches. This chapter also introduces the concept of computer vision for roof detection and examines existing labeling services and their pricing structures.

Chapter 3 presents the proposed project management approach, which combines Lean software development and SCRUM Agile methodology. This chapter provides a detailed description of the project, including a flow chart conceptualization, the ATTA project classification method, and an overview of data acquisition and quality assessment. The chapter also covers IT infrastructure and data organization, label creation and human-based validation, and label validation to ensure quality of service.

The case study presented in chapter 4 illustrates how the proposed methodology can be applied to a specific project involving satellite images. The methodology presented in chapter

3 provides an applied framework for organizing the project, acquiring and assessing data quality, creating and validating labels, and tuning and validating the model.

By emphasizing the importance of project management, this paper highlights the need for proper planning, organization, and execution to ensure the success of machine learning initiatives. The use of SCRUM and lean software development methodologies, along with the Label Studio tool, ClickUp and Slack, further underscores the significance of agile and efficient project management techniques in the field of machine learning.

Overall, by uniting these theoretical aspects of project management and its practical application to a project in ML, this paper provides valuable insights and practical guidance for researchers and practitioners involved in machine learning initiatives, particularly in the field of image analysis and computer vision.

The purpose of this thesis paper was to illustrate and open the people's eyes on the importance of project management in machine learning projects, in our case rooftop detection. Using some of the main methods and methodologies in machine learning, we were able to do quality assessment and validate labeling data of rooftops using satellite images.

References

Beck, Kent. 2000. *Extreme programming explained*. Translated by Erich Gamma. N.p.: Addison-Wesley.

Biesialska, Katarzyna, Xavier Franch, and Victor Muntés-Mulero. 2020. "Big Data analytics in Agile software development: A systematic mapping study." *Information and software technology* 132, no. 106448 (september): 0950-5849. https://doi.org/10.1016/j.infsof.2020.106448.

Bigelow, Stephen J. 2019. "What does it mean to be an enterprise API management architect?" TechTarget. https://www.techtarget.com/searchapparchitecture/tip/What-does-it-mean-to-be-an-enterprise-API-management-architect.

Bisen, Vikram S. 2019. "Why Data Annotation is Important for Machine Learning and AI?" Medium. https://medium.com/vsinghbisen/why-data-annotation-is-important-for-machine-learning-and-ai-5e647637c621.

BRASJÖ, CHRISTOFFER, and MARTIN LINDOVSKY. 2019. "Machine Learning Project Management." Chalmers Publication Library. https://odr.chalmers.se/server/api/core/bitstreams/d060ba80-8b0f-4a10-a67b-cc03373fae52/content.

Chang, Joseph Chee, Saleema Amershi, and Ece Kamar. 2017. "Revolt: Collaborative Crowdsourcing for Labeling Machine Learning Datasets." In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2334-2346. https://doi.org/10.1145/3025453.3026044.

Cockburn, Alistair. 2005. *Crystal Clear: A Human-powered Methodology for Small Teams*. N.p.: Addison-Wesley.

Fredriksson, Teodor, David Issa Mattos, Jan Bosch, and Helena Holmstrom Olsson. 2020. “Data Labeling: An Empirical Investigation into Industrial Challenges and Mitigation Strategies.” *Research Gate* 12562 (November): 202 - 216. 10.1007/978-3-030-64148-1_13.

Gaba, Ishan. 2022. “What is Slack App & How to Use It to Improve Productivity?” Simplilearn. https://www.simplilearn.com/tutorials/slack-tutorial/what-is-slack-app-uses-to-improve-productivity#what_is_slack_used_for.

Hayat, Faisal, Ammar Ur Rehman, Khawaja Sarmad Arif, Kanwal Wahab, and Muhammad Abbas. 2019. *The Influence of Agile Methodology (Scrum) on Software Project Management*. N.p.: 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). 10.1109/SNPD.2019.8935813.

Highsmith, Jim, and Alistair Cockburn. 2001. “Agile Software Development: The Business of Innovation.” *SOFTWARE MANAGEMENT* 34, no. 9 (September): 120-127. 10.1109/2.947100.

Hjertaker, Andreas Torkildsen, and Irnis Besirovic. 2022. “Project Management in Idustrial Application of Machine Learning.” NTNU Open. https://ntnuopen.ntnu.no/ntnu-xmlui/bitstream/handle/11250/3004171/no.ntnu%3ainspera%3a111687478%3a111691937.pdf?sequence=1&isAllowed=y.

Izquierdo, Salvador, Marcos Rodrigues, and Norberto Fueyo. 2008. “A method for estimating the geographical distribution of the available roof surface area for large-scale photovoltaic energy-potential evaluations.” *Solar energy* 82, no. 10 (April): 929-939. 10.1016/j.solener.2008.03.007.

Li, Peiran, Haoran Zhang, and Zhiling Guo. 2021. “Understanding rooftop PV panel semantic segmentation of satellite and aerial images for better using machine learning.” *Advances in Applied Energy* 4, no. Elsevier (July): 100057. https://doi.org/10.1016/j.adapen.2021.100057.

Lima, Pedro M., and James Lighthill. 2021. “Data Labeling Tools for Computer Vision: a Review.” run@unl.pt. https://run.unl.pt/bitstream/10362/135873/1/TCDMAA0144.pdf.

Maayan, Gilad D. 2022. “Complete Guide to Image Labeling for Machine Learning - DATAVERSITY.” Dataversity. https://www.dataversity.net/complete-guide-to-image-labeling-for-machine-learning/.

Minds, Orenda. 2019. “What Is Annotation In AI / ML ?. Annotation in machine learning /… | by Orenda Minds.” Medium. https://medium.com/@orendaminds/what-is-annotation-in-ai-ml-df63af8599e8.

Palmer, Stephen R., and John M. Felsing. 2002. *A Practical Guide to Feature-driven Development*. N.p.: Prentice Hall PTR.

Poppendieck, Mary, Tom Poppendieck, and Thomas D. Poppendieck. 2003. *Lean Software Development: An Agile Toolkit*. N.p.: Addison-Wesley.

Stapleton, Jennifer, ed. 2003. *DSDM: Business Focused Development*. N.p.: Addison-Wesley.

Tore, and Torgeir Dingsøyr. 2008. “Empirical studies of agile software development: A systematic review.” *Information and Software Technology* 50 (9 - 10): 833 - 859. https://doi.org/10.1016/j.infsof.2008.01.006.