Let‘s talk about AI: Machine Learning

An Overview

Whitepaper

js-soft

SUMMARY / TOC

Over the course of the last three decades, machine learning has evolved from a mere academic exercise with a few niche applications into a dominant topic whose advances are at the core of many recent technological innovations and breakthroughs. A vastly expanded volume of available development data, increasingly affordable computing power and betterments of algorithms and related software have made machine learning accessible and applicable to a wide spectrum of industries and businesses, in theory.

Notwithstanding its popularity and apparent pervasiveness, machine learning remains a technically challenging subject. Many businesses struggle with the practical aspects of interfacing their existing processes with machine learning applications and face considerable expenses for the endeavor to merge novel technologies with their existing tech-stack. To make matters worse, the highly convoluted technological landscape complicates orientation and introduces a substantial entry barrier.

Addressing these matters this whitepaper provides a set of guiding principles and best practices for businesses that seek to venture into the field of machine learning. We provide a synopsis of the most relevant machine learning technologies provided by SAP. Although the focus of our market overview lies in SAP-related technologies the predominant majority of the information presented in this document is universally valid and applicable.

Chapter I introduces basic concepts and gives an overview of the current state of affairs of machine learning. Chapter II details the anatomy of a fully-featured machine learning application and helps to gauge the overall effort involved in its realization. Chapter III lists SAP's most relevant technologies and tools related to machine learning and offers insights into possible application scenarios. Finally, Chapter IV presents the recommended best practices derived from the expositions of the preceding chapters and concludes this report.

© j&s-soft AG

MACHINE LEARNING FUNDAMENTALS

If and how well machine learning may be employed in an organization depends on the degree of understanding of what it is and where it is headed. This chapter provides a quick economic outlook and introduces basic terminology pertaining to machine learning and artificial intelligence and serves as a brief glossary for the notions most relevant to this report. We prefer conciseness and clarity over an attempt to provide a rigorous and exhaustive set of definitions which are bound to be complex and unwieldy. As this is a 'practical' report we are accordingly concerned with those aspects of terminology which turn out to be relevant for practical application.

Economic Outlook

In this section we consult several recent surveys and reports to establish a baseline of how machine learning is perceived by the industry and what expectations it elicits in its early adopters. While machine learning may seem to be pervasive and to be used across all industries, as its extensive publicity in the technologically oriented media may suggest, the reality is very different. Although surveys tend to exaggerate towards the positive many participants are willing to concede that their own experience paints a much more humbling picture.

McKinsey estimates that artificial intelligence techniques, which are currently mainly based on machine learning, have the potential to create between $3.5 trillion and $5.8 trillion in annual value across industries [1]. Similar estimates were published by Gartner [3]. Driven by such high expectations, many companies reach for new opportunities and, depending on the survey, between 33% and 63% of enterprises worldwide claim to have already adopted machine learning in production today [2, 5, 6].

In a survey of 1.100 IT and line-of-business executives from US-based companies, 63% state that their artificial intelligence adoption aims to catch up with competitors or to establish a lead [6]. 42% even conjecture that artificial intelligence technologies will become critical in the next one to two years [6], which is a comparatively short period of time to establish expertise given the complexity of these technologies.

In practice, any potential benefits of adoption are contrasted with a high amount of time and effort to build up expertise and to overcome machine learning specific limitations and obstacles [1]. Although two-thirds of enterprises believe in the importance of artificial intelligence and machine learning, at most one-third trust in their competence and experience in dealing with these new technologies [7]. Besides model building, a lack of experience in model deployment and consumption in an application poses a major problem. The result is an apparent gap between model experimentation and models that end up in production. Gartner predicts that most artificial intelligence projects will not make it to production until 2022 [4].

Based on these observations, Machine learning as a service providers increasingly supply enterprises with implemented or pre-configured functionality which serves to facilitate the adoption of machine learning and to narrow its technical entry barrier. New services continue to spawn and even though machine learning on an enterprise-scale is certainly still in its early stages the machine learning market is already very rich.

Among the providers of machine learning services is SAP, which is the focus of this report and to which Chapter III and Chapter IV are dedicated.

© j&s-soft AG

Areas in Artificial Intelligence

The terms artificial intelligence, machine learning and deep learning are often confused and are utilized interchangeably. The following paragraphs clarify and differentiate the three concepts and characterize their mutual relationship which is summarized by the symbolic diagram in Figure I.1.

Artificial Intelligence

Artificial intelligence denotes the concept of computers performing tasks which normally require human intelligence and has established itself as an indiscriminately used umbrella term encompassing everything a computer can be made to do that might be perceived as ‘smart’ by a human. We explicitly stress that the term is nebulous and that it is best to conceive of artificial intelligence in accordance with how it is contemporarily used and understand it as a vague notion in the above sense as opposed to attempting to find a rigorous definition for it.

n addition to the strict subset of machine learning techniques characterized in the next section artificial intelligence includes rule-based programming techniques which are sometimes referred to as classical programming techniques. Rule-based programming techniques can be used to construct sophisticated expert systems and rules engines which may justifiably be perceived as smart. The characteristic which distinguishes artificial intelligence from machine learning is that the ‘intelligence’ in classical AI is realized by the manual construction of a decision tree which maps the understanding of a human agent into the computer program whereas machine learning techniques are applied to construct the decision tree semi-autonomously.

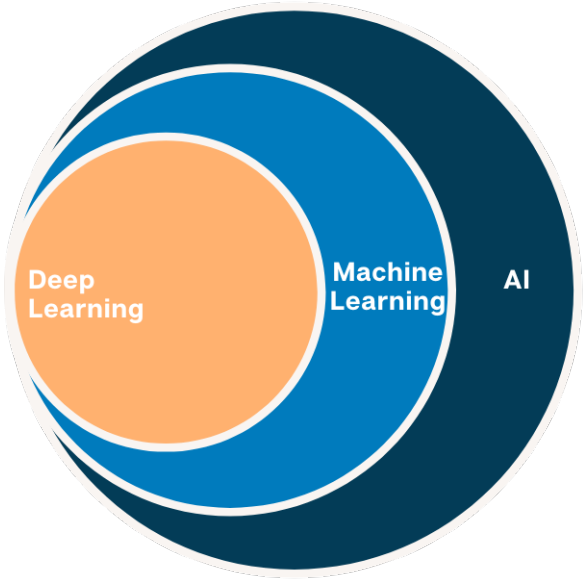

Figure I.1: Symbolic set diagram highlighting the relationship between artificial intelligence, ma-chine learning and deep learning. Artificial intelligence encompasses machine learning and classical, rule-based programming techniques. Deep learning is distinguished from machine learning by the application of multi-layered neural networks.

It has become a widespread custom to attach the AI-label to almost everything in an attempt to elicit a perception of grandeur and importance in the target audience. Artificial intelligence has a long and interesting history and its recent technological developments have ignited the imaginations of many which has brought the term back into vogue. This is very relevant in the corporate context of enterprise software with which this report is partly concerned. Also, note that whenever breakthroughs in or novel applications of artificial intelligence are reported they apply exclusively to machine learning or deep learning.

© j&s-soft AG

Machine Learning

Machine learning is the science of providing computers with the ability to learn and automatically im-prove from experience without being explicitly programmed. The focal point of this definition lies in its final part which stresses that, in contrast to classical programming techniques, there is no need to manu-ally construct the reasoning logic. Instead, machine learning techniques are used to autonomously adapt to a provided set of data and to ‘learn’ its characteristics. A machine learning program alters itself.

The learning procedure corresponds to an optimization problem in which the loss function, a task-specific metric, is minimized. During this process internal variables are continuously adjusted until a stable point is arrived at. The entirety of the internal variables and the applied algorithms are referred to as the machine learning model. After the conclusion of the training phase the model’s parameters reflect the internal state of the reasoning logic which has been extracted from the data.

By definition, machine learning applies where rule-based reasoning is unsuitable, be it due to the fact that the rules are unknown or because the rules are prohibitively copious and intricate. Many tasks which are performed based on human intuition fall into this category such as the ability of every child to distinguish daisies from sunflowers or to differentiate the sounds produced by different musical instruments.

Deep Learning

Deep learning is a subset of machine learning which is concerned with neural networks. Neural net-works are a concept loosely inspired by biology in which artificial neurons are arranged in multiple interconnected layers as shown below in Figure I.2. The naming of deep learning simply stems from the circumstance that the tree graph spanned by multi-layered neural networks is deep, as opposed to architectures with a single or two layers which are referred to as shallow.

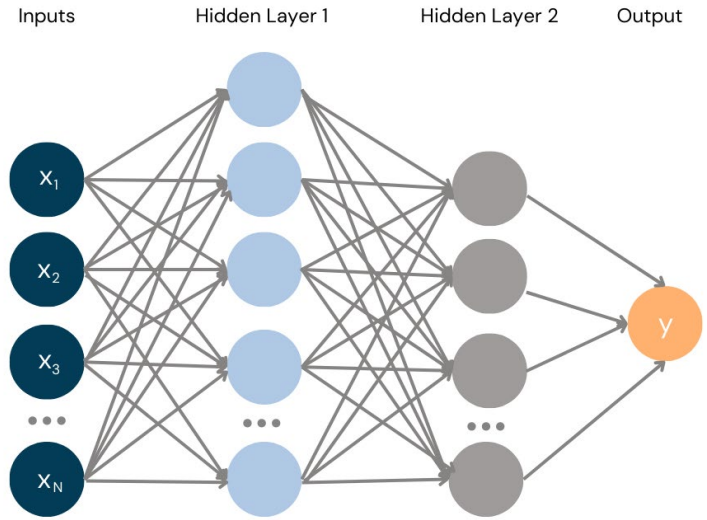

Figure I.2: Multi-layer Neural Network Multi-layer networks refer to architectures with multiple layers between the input and output layers, which are as referred to as hidden layers. The resulting tree diagram grows in depth with every layer which gives deep learning its name.

© j&s-soft AG

The class of models comprised of neural networks and the associated algorithms are at the heart of the many breakthroughs of recent years and have made it possible to surpass human performance levels in areas such as image and audio recognition and natural language processing, i.e. in the very fields of applications which are notoriously difficult to manage using rule-based systems and in which human intuition reigned supreme over computer agents.

An essential advantage of deep learning is that neural networks can be applied to unstructured data. Many statistical algorithms of machine learning require structured data, i.e data that conforms to a simple format such as tabular data which makes said algorithms inapplicable to fields where unstructured data is prevalent such as in the fields of application mentioned above.

An important disadvantage of neural networks is the requirement of very large amounts of data. Due to this high data volume and the complex network topology, training deep neural networks usually requires a lot of computing power and training time. Another drawback of deep learning is the 'black box' problem. The trained networks are rarely interpretable and it is nearly impossible to reason in human terms why a certain conclusion was drawn.

Basic Machine Learning Terminology

Model Constituents

As stated above, machine learning is based on the idea of automated parameter fitting to adapt an algorithm to the information provided by a given dataset. The overall aim is to arrive at an approximation of the distribution underlying the data. If the data is representative and the approximation is accurate it can be used to provide predictions on new, unseen data. This process involves constituents which we briefly describe here for later reference. See Figure I.3 for a symbolic overview.

The algorithm is a class of rules or functions that are parameterized during the training process. It determines the totality of all distributions which may be learned from the provided data set. Algorithms are chosen based on the characteristics of the data and assumptions about the problem at hand. An example of an algorithm is polynomial regression.

The hyperparameters are the static subset of an algorithm's variables, which is not estimated during training. Hyperparameters determine the shape and character of a model and are specified manually prior to training. Their choice selects a specific function from of the algorithm's class of functions to be trained on the data. The hyperparameters for a polynomial regression include the degree of the polynomial.

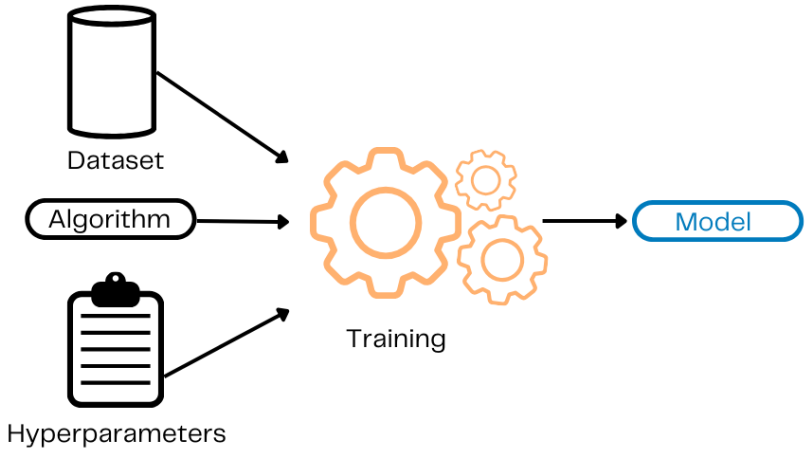

Figure I.3: Constituents of a machine learning model. The generic algorithm and a set of hyperparameters comprise the model whose internal parameters are adjusted according to a training data set.

© j&s-soft AG

The parameters are the internal variables of a specific function determined by a choice of hyperparameters which are estimated and incrementally adjusted during the training process. Their values capture the internalized representation of the data set which was utilized for training. The parameters for polynomial regression correspond to the weights of the polynomial.

A model is an umbrella term describing a selected algorithm and a concrete choice of hyperparameters. A model is said to have been trained if it has undergone the training- process and its parameters have been determined accordingly. The term is sometimes used more broadly referring to algorithms or whole classes of algorithms.

Learning Types

The broad field of machine learning can be subdivided into categories according to the type of training scheme which is applied to a model for a given task. Distinctions are made between supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning is utilized when the aim is to make predictions about input data which may either be assigned to a member of a set of categories or a continuous numerical value. Such tasks are referred to as classification tasks or regression tasks, respectively. Supervised learning uses labeled input data for the training process, i.e. each piece of input data on which the model is trained is accompanied by the 'true' value according to which the model's parameters are steered stepwise in the right direction.

Exemplary supervised learning tasks are given in Table I.1.

	Classification	Regression
Task description	The output variable is a category or an integer	The output variable is a real value or continuous quantity
Task examples	Music genre classification	Housing price prediction
	Spam email detection (spam vs. no spam)	Predicting how much a customer will Spend
	Predicting failure of machines (failure/ no failure)	Predicting the remaining lifetime of a Machine

Table I.1: Exemplary supervised learning tasks: Classification vs. regression tasks

Unsupervised learning is applied to autonomously uncover hidden structures in the data. In contrast to supervised learning, unsupervised learning attempts to recognize previously unknown patterns in the data without the explicit requirement for labels. Two important applications of unsupervised learning are clustering and dimensionality reduction. Clustering, or cluster analysis, divides the data into groups of similar data points, whereby the measure of similarity is determined by the choice of the model. Dimensionality reduction aims to remove portions of data which are redundant or may be expressed more concisely in order to find a minimal representation of one's data set.

References

[1] Michael Chui et al. Notes from the ai frontier - Insights from hundreds of use cases. Tech. rep. 2018.
[2] Louis Columbus. The State of AI and Machine Learning. Tech. rep. 2019. URL: Link.
[3] Gartner. Gartner Says Global Artificial Intelligence Business Value to Reach $1.2 Trillion in 2018. 2018. URL: Link.
[4] Gartner. Gartner Says Nearly Half of CIOs Are Planning to Deploy Artificial Intelligence. 2018. URL: Link.
[5] Ben Lorica and Paco Nathan. "The state of machine learning adoption in the enterprise". In: O'Reilly (2018).
[6] Jeff Loucks, Tom Davenport, and David Schatsky. "State of AI in the Enterprise, 2nd Edition". In: (2018), p. 26.
[7] Kevin Poskitt. SAP Data Intelligence: Enterprise AI Meets Intelligent Information Management | SAP Blogs. 2019. URL: Link.

© j&s-soft AG

THE MACHINE LEARNING LIFE CYCLE

Building a mature, real-world machine learning application involves many facets beyond the mere choice of a suitable model or algorithm and its training. Model deployment, evaluation and monitoring comprise essential components of a machine learning project's pipeline and require just as much thoughtful consideration. This chapter aims to provide a structured blueprint of a typical machine learning project. Its individual components are laid out sequentially as a series of logically consecutive steps which need to be implemented one way or another.

In spite of the sequential structure, many components exhibit mutual dependencies that require incremental fine-tuning of previously implemented components, going back and forth multiple times. In practice, a machine learning project is not carried out in a top-down fashion but instead is realized by iterating over the individual steps of implementation multiple times, incorporating acquired information and understanding which oftentimes calls for readjustments to the project's original goals. We thus refer to this iterative and cyclical process as the machine learning life cycle.

The schema of the machine learning life cycle presented herein shall provide a cognitive aid rather than a rigorous set of principles. We aim to capture the crucial aspects and commonalities of real-life machine learning applications. The machine learning life cycle's steps are introduced as an overview in Figure II.1 with further detail being provided in the subsequent paragraphs.

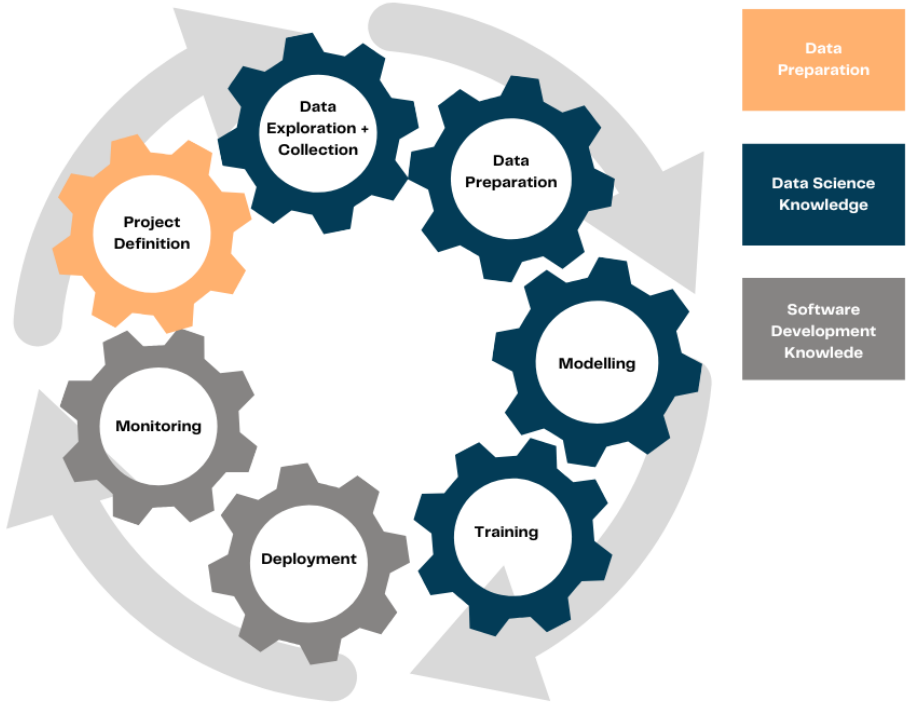

Figure II.1: ***A machine learning project's life cycle.*** *A project's individual steps are coarsely colorcode*

© j&s-soft AG

Project Definition

The pivotal question for any machine learning task is whether it is advisable to use machine learning in the first place. As the contents of this chapter will demonstrate a machine learning application entails unique aspects of great complexity which may be elided altogether by the usage of classical programming techniques if the problem at hand so allows.

Problems that may qualify for machine learning are distinguished from classical problems of statistics or programming by the impossibility to formulate a deterministic and simple rule-based solution or where the exhaustive set of such rules would be unfeasibly complex and large. The rise of machine learning is not the least due to having made practical to systematically address this class of problems at all. Where sufficient quantities of data can be provided a problem's characteristics may be deduced by the application of the techniques of machine learning.

Given a task has been determined to be solved using machine learning, the subsequent project definition phase should establish which subset of components of the machine learning lifecycle will have to be implemented manually and which are available as more or less ready-to-use modules or services from the ever-growing machine learning marketplace. Many frameworks and tools are available which encapsulate interrelated functionality of a machine learning project and provide control via well-defined interfaces which reduce the exposure to technical details to the degree that is actually required for the problem to be solved.

Data Exploration and Collection

The first hands-on step in a machine learning project is to explore the available data which might be potentially relevant to the task at hand. Establishing a proper overview of one's data set is an essential prerequisite and sets the stage for the rest of the project as the choice of a model or algorithm is intimately linked to the available data.

Generally speaking, the available corpus of data must be sufficiently representative of the problem domain to allow an algorithm or a model to internalize or 'learn' its characteristics. It is hence useful to think in terms of quality vs. quantity of input data during data collection. The less relevant information any given piece of gathered data contains the greater the required total volume of data will be for the trained model to properly generalize to previously unseen information. While most machine learning tasks involving unstructured data such as imagery or human speech require prohibitively copious amounts of input data for a model to be trained from scratch, transfer learning can drastically reduce this required volume if a suitable pre-trained model is available.

After anonymizing potentially sensitive information the definitive corpus of data should be stored in a consistent format and should be readily accessible and expandable. Viable storage options include relational database systems and structured directory trees in a local file system which are often used when data is present in the form of individual files.

© j&s-soft AG

Data Preparation

The gathered data is then subjected to multiple steps of data preparation. Redundant or irrelevant information is removed and the data may be expanded by augmenting it with domain-specific knowledge and mathematical transformations that present the data in a way that is more ‘understandable’ by the model. These collective measures are referred to as feature engineering and aim to produce a representation of one’s data which is optimal with respect to the set of algorithmic tools and models to be used for solving the problem at hand. Feature engineering is best performed utilizing a mixture of common sense and domain expertise and involves manipulations as simple as removing the color channels from images for object classification tasks and more sophisticated transformations such as dimensionality reduction.

Conditioning refers to transformations of one’s numerical data such that rounding errors inherent to calculations performed by computer hardware do not build up. A processor’s floating-point arithmetic exhibits unavoidable rounding errors which, although minute individually, can accumulate and produce erroneous computations and can prevent the convergence of iterative methods unless special care is taken to control their propagation. Note that conditioning is sometimes considered to be a part of the learning algorithm itself.

Adding to the above, supervised learning tasks entail additional work. An accurate set of labels must accompany the data which, if absent, will have to be prepared by manual labor. If the data corpus is sufficiently large to allow for hold-out validation a small portion of the data volume is put aside as a test set in order to periodically estimate the progress in model training by gauging the performance on the test set. If possible, another small subset of data referred to as the validation set should be reserved in order to compare different settings of model hyperparameters. Special care needs to be taken to ensure sufficiently similar statistical distributions of the data subsets, i.e. the subsets must exhibit the same characteristics. A careless sampling of the total data volume will introduce hidden biases.

Modeling

The choice of a machine learning model must be made in consideration of the data set’s volume and degree of intricacy. The informal notion of capacity, sometimes called a model’s complexity, describes a model’s ability to internalize and express the complex relationships contained in data. An overly simplistic model, i.e. a model whose capacity is too small, is unable to grasp the characteristics of a complicated data set and will thus perform poorly which is referred to as underfitting. On the flip side, an exceedingly complex model requires much more data to work properly or it will otherwise exhibit overfitting and will internalize the data’s noise.

Where the hypotheses of machine learning models are used in safety-relevant or delicate matters the introspectability of a model becomes relevant which refers to the possibility to comprehend why a model arrived at a prediction or an estimation. While it’s usually of no concern how an image classifier reasons a model’s decision about a company’s employee leading to termination must be understood by a human. In such cases, a black-box model should be considered unacceptable regardless of how well the model ostensibly performed on historical data during its training process.

The introspectability loosely correlates with a models complexity. While simpler models such as linear or polynomial regressors readily expose their internal perception of the importance of each feature, models on the opposite end of the complexity spectrum such as the family of neural networks are virtually opaque.

© j&s-soft AG

Training

The training process of a machine learning model follows a simple scheme: After initializing the model with a random set of its parameters the training data is used to produce optimum parameter values by solving an optimization task based on the minimization of a domain-specific metric. For supervised learning tasks the metric correlates strongly with how well the model performs on the training set while for unsupervised learning tasks the metric is of a more abstract nature.

Generally, the resulting parameters correspond to local optima, i.e. there may exist other sets of model parameters whose performance is better in terms of the metric. As it is unfeasible to scan for a global optimum the aim is thus to find a good enough approximation to serve as a solution to one's problem. Additionally, every choice of a metric entails a certain degree of arbitrariness in much the same way any attempt to precisely quantify beauty would be arbitrary. The final decision over the progress of the training process should thus be made by a human.

Where the amount of data is large the training procedure is very computationally heavy and ought to be executed on dedicated training hardware to achieve tolerable training durations. For many models and types of data the majority of involved computations consists of dense matrix arithmetic which is better dealt with by GPUs than by CPU compute nodes. The bottleneck for these operations is the memory bandwidth, i.e. the speed at which data can be read and written to and from main memory. A GPU's memory bandwidth exceeds that of a CPU at least tenfold while it also possesses advanced techniques to hide the latency of memory accesses. It has thus become standard practice to train on dedicated GPU hardware when working with large amounts of data or complex models, notably any flavor of neural network.

In stark contrast to the training itself the evaluation of a trained model is readily performed by standard commodity hardware. There exist many services that allow to perform the training step on remote hardware rented for the duration of training. Speaking in modern terms, one is able to train in the cloud while deploying a service based on the trained model independently.

Deployment

As with any other business application the ultimate objective is to create a service available for consumption. With regards to a machine learning project this stage of deployment refers to making the model available within its target environment. To this end, a wrapper application is set up which exposes the trained model's acquired wisdom through a well-defined interface that can be subsequently used to return predictions. As application deployment is generally well understood and not specific to machine learning we will keep this section accordingly brief and refer the reader to the existing literature for further reading.

A sizeable fraction of the plethora of technological novelties pertaining to machine learning is related to the deployment process. Many vendors of machine learning solutions provide custom tools and solutions which allow the user to deploy a machine learning model with little effort. Of the different strategies containerized deployments are the most popular due to their light weight, the ease with which the application can be moved around, and their inherent ability to scale.

Important aspects of a deployment solution to consider include the latency at which requests can be served, i.e. the delay between a query and the response containing the model's prediction. Auditability may be desirable where information about model accesses and its usage patterns are of interest. Where the user base of one's application is expected to grow over time scalable deployment solutions become relevant

© j&s-soft AG

Monitoring

Whenever the deployed model is intended to be used continuously for an extended period of time, as opposed to a one-shot usage scenario, it will necessarily have to be monitored. Unless working in a controlled environment changes in the input data's characteristics constitute a serious threat to the integrity of the whole application. Such changes, called concept drift or data drift, refer to a systematic shifting of the input data's distribution over time and may be produced by a multitude of causes, such as wear of mechanical or electronic components or changes to the environment, i.e. alteration of hidden variables whose variation had not been foreseen at the time of modeling.

Whichever circumstance may be the reason, concept drift causes the implicit and explicit assumptions made during modeling and training to no longer hold and will cause the model's performance to deteriorate over time with deleterious consequences for every dependent and user of the application.

A periodical sampling of the input data can be used to ascertain the degree of data drift and whether readjustments to the model or full retraining may be required. Where the model is continuously learning from newly gathered information, i.e. when dealing with online learning systems, the model may be probed with a stored set of labeled data to determine if it is still well-behaved or whether it has silently adjusted to rogue input data.

Do you want to find out how you can use AI profitably for your processes? Get in touch with us!

Involved in any project, our AI team, with their uniquely trained perspective, is capable of delivering surprising insights. They understand processes and identify tasks that AI can take over.
Our clients are amazed by how existing workflows are:

•streamlined,

•error rates significantly reduced,

•and quality measurably improved.

Book our AI team for your next project.
Include them early in the project planning phase. **Benefit from the AI perspective.**

Kontakt

Estelle Hounsa
+49 6221 43121-61
sales@js-soft.com

j&s-soft AG
Max-Jarecki-Str. 21
69115 Heidelberg

© j&s-soft AG

js-soft

Contact

j&s-soft AG
Estelle Hounsa
Max-Jarecki-Straße 21
69115 Heidelberg

+49 6221 43121-61

sales@js-soft.com

© j&s-soft AG

js-soft