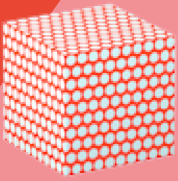
Artificial Intelligence

A Quick Start Guide



futurice

Artificial Intelligence is easily one of the most hyped subjects in the world right now. It is the repository of much hope as well as the anxiety we as a society feel about the future – will it prove to be the source of immeasurable wealth and progress for much of society, or will the few reap its rewards? Will it destroy more jobs than it creates? Or will it go all Skynet on us?!

Artificial Intelligence is here today and it's neither magic nor alchemy. Nor even a mystery. With a little effort, you can get a decent understanding on what it is, and more importantly, how you can apply it to your business.

As our Senior Data Scientist Daryl Weir highlights in his wonderful article, AI is always a computer program someone wrote. One of the goals in the work we do in this area is to demystify AI and increase people's understanding of what it is and what it can do. We believe this is the best way to make sure AI is used in a way that benefits society as a whole.

What is Artificial Intelligence? The dictionary definition is:

> The theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages

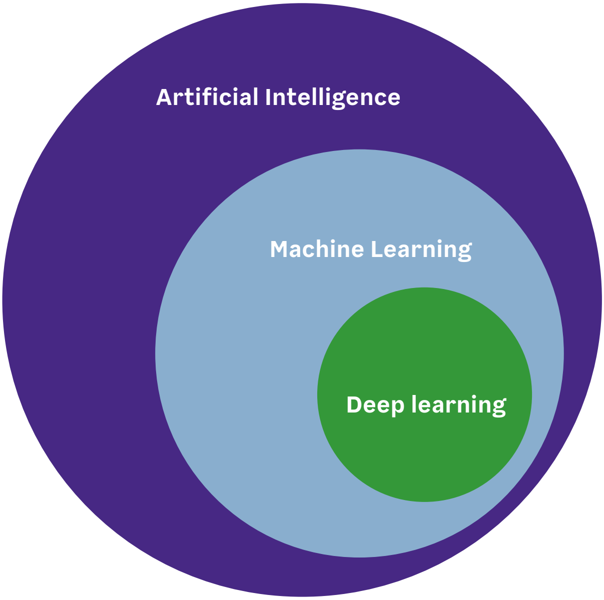
Why is AI so hot right now? It all boils down to the convergence of three distinct phenomena:

01 the amount of data is increasing exponentially,

02 computing power is doing the same and cloud computing is cheap, and

03 as a result of items 1 and 2, neural networks have become useful.

The AI hype cycle is cresting, buoyed by a lot of hot air. The balloon will soon lose altitude, just like the Internet balloon did after its initial stratospheric lift. And just like the Internet ended up



changing our lives in ways much more profound than envisioned during the first hype, AI will bounce back and have a transformative impact on us as a species. So it pays to stay on board.

Why does any of this matter to you or your business? How is it any different from other recent business buzzwords like 'digitalization' or from the meat-and-potatoes work of software development in general? These are some of the questions we look at in this collection of articles from our experts.

Over the last few years, we have worked on a wide variety of cases that involve AI to varying degrees, with clients like Supercell, Barona, ABB, Veikkaus, Sanoma, as well as a number of NGOs, etc. Our work includes solutions for recruitment, hate speech detection, customer support, personalisation, facial recognition, and many others.

In this little pamphlet we've collected some of the articles our employees have written about Machine Learning and AI. They delve into what ML and AI are, how machines actually learn, how AI/ML differ from software development – which many of you may already be quite familiar with – and, for inspiration, how data can be monetized. We hope you'll find it an enjoyable read.

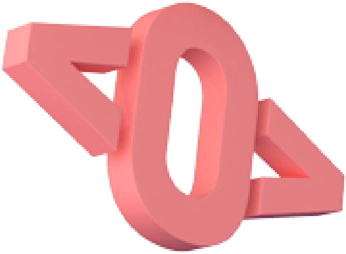
In addition to what you'll find on these pages, we also offer a concise AI Crash Course, especially created with the needs of corporate leaders in mind.

If you have any questions about AI or the work we do in the area, please contact:

Claes Kaarni
Vice President
claes.kaarni@futurice.com
+358 50 487 7255

Katja Metsola
Head of Sales, Data Science
katja.metsola@futurice.com
+358 40 745 3770

Contents



How do machines learn?

Antti Ajanki - Data Scientist

A brief and easy-to-understand look at what Machine Learning is, how a machine learns and what sorts of tasks it's suited for.

Recommended for anyone interested in, but not conversant in the basics of Machine Learning

Machine Learning is the secret sauce behind speech recognition, self-driving cars and other fascinating technologies. But what exactly is Machine Learning, and where can it be applied?

Learning by choosing an algorithm

The only thing that computers (that is the “machine” part of Machine Learning) do is execute algorithms. An algorithm is a fixed sequence of steps. Computers aren’t creative, nor can they come up with new ideas. How do they then learn anything at all?

For any given task there are countless conceivable algorithms, some better and some worse. If a computer is instructed to evaluate algorithms one by one and keep track of the best algorithm it has encountered so far, its performance in the task will increase over time as it stumbles upon better algorithms. It seems as if the computer is gradually learning to master the task. This is the core idea behind Machine Learning.

Let’s consider a spam filter, which was an early success case for Machine Learning. A spam filter takes an email message as an input, processes it (somehow), and finally outputs a label: “spam” or “not spam”. One could try to write a spam filtering algorithm by hand. Perhaps collect a list of spammy words (“casino”, “free”, “genuine”, etc.), count how many of these words appear in an email message, and say that a message is spam if the count exceed a certain threshold. However, there is no straightforward criterion of deciding which words should be on the list (“free” is likely to appear in a legitimate message, too) or the threshold value. Machine Learning, on the other hand, can diligently evaluate millions of word list variants and pick the one that most accurately detects spam.

Data as a yardstick for comparing algorithms

So far we have seen that Machine Learning boils down to a selection of a well-performing algorithm. But how do we measure the performance? A fair way to compare algorithms is to evaluate them on a data set representative of the task the computer is supposed to learn. Typically, this training data is a set of input examples and corresponding human-annotated expected outputs. For example, a set of email messages that a person has read and annotated either as “spam” or “not spam” would work as training data for learning a spam filter.

Candidate algorithms are ranked based on how accurately they replicate the expected outputs when they are fed input data instances. Of course, ultimately we are interested on how well an algorithm will perform on inputs which are not part of the training (after all, we already know the correct output for all the training instances). If the training data is representative and the evaluation is done carefully, there are guarantees that the performance generalizes to previously unseen data points.

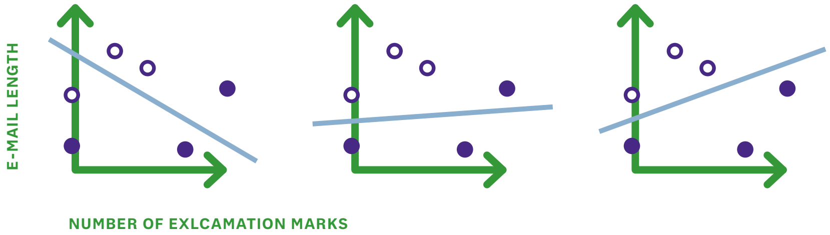
Choosing an algorithm: an exercise in mathematical optimization

Where do the candidate algorithms come from? A data scientist, who is setting up the learning task, has to decide which algorithms the computer should consider.

The choice of which algorithms to evaluate is affected by two, often contradictory, considerations: the set of candidate algorithms should be flexible enough to capture essential features of the task (such as information about which variables depend on each other), and, at the same time, the search over the candidates should be mathematically tractable. Often, the candidate set is selected so that a comparison between algorithms reduces to a mathematical optimization problem. Optimization is a heavily studied field, and plenty of tools exists for solving optimization problems effectively.

Example: learning a spam filter

The image below visualizes the task of learning a simple spam filter. It depicts emails that are collected for training purposes and three instances of competing algorithms. The training emails are shown as dots. The solid dots are spam and the open dots are non-spam messages. In this example, emails are described by just two variables that are assumed to be related to their degree of spamminess: the number of words in an email (vertical axis) and the number of exclamation mark in an email (horizontal).



I have decided to restrict the search space of algorithms to lines. A line defines a spam filter: emails that fall on one side of the line are classified as spam and the ones on the other side as non-spam. The three pictures show three examples out of infinitely many possible ways of drawing a line on the plane.

Machine Learning now becomes optimization: choose a line a line so that as many of the solid dots as possible fall on the one side of the line and as many open dots as possible on the other side. Obviously, the rightmost picture is the optimal solution, and hence the most accurate spam filter, because it cleanly separates the two types of dots.
Of course, beyond this simplified example, there are usually more than two variables, and the data might not separate so cleanly into two regions. These introduce some additional challenges, but the basic principles of learning stay the same.

Where Machine Learning can help?

Some tasks are more suited for Machine Learning that others. Machine Learning depends on the availability of training data. Therefore, tasks with many easily available training instances are particularly suitable for Machine Learning. In cases like natural language translation appropriate data is readily available because a huge number of texts have already been translated by human translators.

Machine Learning can conquer some tasks that are complicated for humans because machines are good at searching through huge spaces of algorithms. Computers learn to beat humans at games like Go by systematically exploring the games' vast state spaces.

An interesting class of tasks are those that humans can carry out instinctively, but attempting to program a computer to do the same is hard, because formulating the thought process as an explicit algorithm is next to impossible. These include driving a car or coloring black-and-white images. With Machine Learning, computers can learn to perform these tasks, too.

Challenges

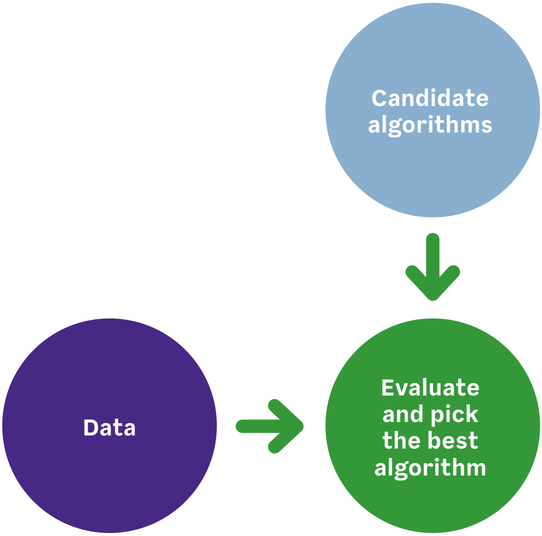
Computers don't learn by themselves. A human data scientist must express the learning goals in a format a computer understands. It can be tricky to formulate a learning task so that the computer is able to learn efficiently. Usually, a lot of time is spent on preprocessing the raw data so that a learning algorithm can take advantage of it.

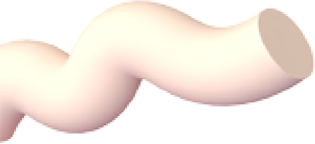
Trying to use Machine Learning for everything is not feasible; humans are still best at tasks that require real ingenuity and flexibility.

It is very easy to get fooled by spurious correlations. If a spam filter is trained on a small data set, it may happen by chance that some common English word, such as “you”, occurs in spam messages but not in any of the non-spam messages. A naive spam filter would learn to associate the word “you” with spam and therefore make foolish classifications. Guarding against such spurious patterns demands rigorous controls on the learning process.

Summary:
Let the computer find the algorithm

Instead of trying to formulate an algorithm for a particular task by hand, machine learners climb higher on the ladder of abstractions and introduce a new trick: they let the computer search for a well-performing algorithm. This approach can be taken if there is suitable data available, and if the search for a good algorithm can be formulated in a mathematically tractable way. When these conditions are met, Machine Learning enables us to find solutions to many difficult tasks where developers struggle to write algorithms directly.





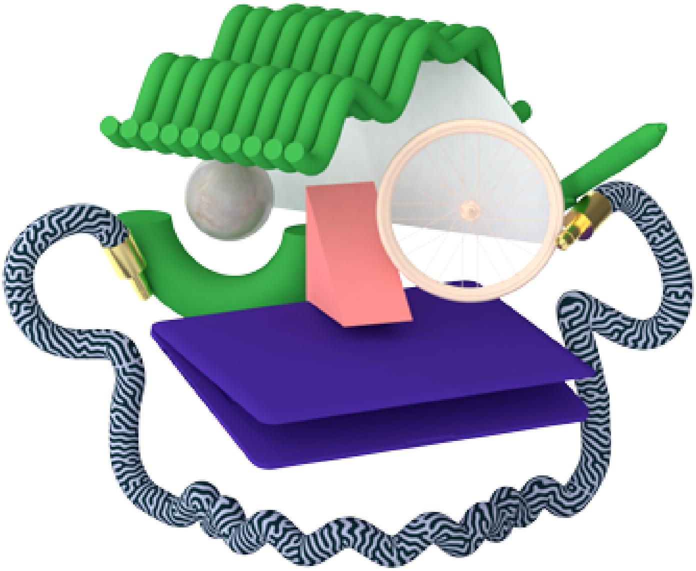
Differences between Machine Learning and software engineering

Antti Ajanki - Data Scientist

A brief look at how software development and Machine Learning projects are similar and different from each other.

Recommended for developers and anyone else who works on or is familiar with software development, and is interested in Machine Learning

.



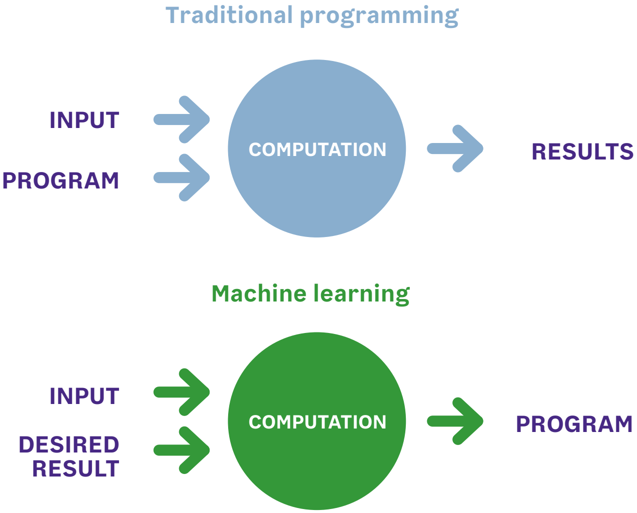
Software engineering is the art of automating a task by writing rules for a computer to follow. Machine Learning goes a step further: it automates the task of writing the rules. How do traditional software engineering and Machine Learning differ? Are there any similarities?

There are ways to alleviate your pain.

The developer perspective

The starting points for traditional software engineering and Machine Learning are quite similar. Both aim to solve problems and both start by becoming familiar with the problem domain: discussing the matter with people, exploring existing software and databases. The differences are in the execution.

Software engineers use their human ingenuity to come up with a solution and formulate it as a precise program a computer can execute. Data scientists, that is people who implement Machine Learning systems, don't try to write down a program by themselves. Instead, they collect input data (dashboard video and other sensor feeds of a car, for example) and desired target values (the throttle level and the angle of the steering wheel). Next, they instruct a computer to find a program that computes an output for each input value (a program that drives a car given the sensor inputs).



Traditionally programmers automate tasks by writing programs. In Machine Learning, a computer finds a program that fits to data.

A software engineer is concerned with the correctness in every corner case. Meanwhile, a data scientist has to be much more comfortable with uncertainty and variability. After all, Machine Learning is all about mining statistical patterns from data. Because of the inherently statistical nature of Machine Learning, it is more flexible on complex problems, but also more difficult to interpret and debug.

Developing a Machine Learning application is even more iterative and explorative process than software engineering. Machine Learning is applied on problems that are too complicated for humans to figure out (that is why we ask a computer to find a solution for us!). Therefore, a data scientist has to embrace experimental attitude and be prepared to test a few approaches before settling on a satisfying one.

From the outside, the modes of work looks very similar: both species of professionals spend a lot of time hunched over a laptop. Data scientists spend a lot of their time writing code in Python or other general-purpose programming language just like traditional programmers. The majority of time in a Machine Learning project is consumed by tasks that are best carried out by traditional programming: writing scripts for merging, cleaning up and visualizing data, and integrating the Machine Learning subsystem with the rest of the application. Certainly the toolkits do have their differences, too. Data scientists are familiar with linear regression and other statistical algorithms while traditional programmers known REST APIs and web frameworks inside out.

The product perspective

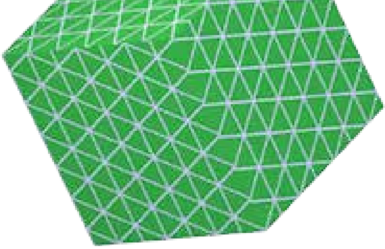
When does a product benefit from Machine Learning? Will there be any use for traditional software engineering in the future or will Machine Learning consume all of software development?

No, Machine Learning will not displace traditional software engineering. Most types of problems that are solved with software engineering today, will be carried out by traditional programming also in the future. Machine Learning, on the other hand, provides a way to tackle new kinds of problems, the kinds that have been unfeasible to solve previously. Tasks that humans perform with relative ease, but that can't be formulated as exact rules (detecting objects in images, driving a car, etc.) are prime candidates for Machine Learning solutions. Machine Learning might be the correct solution also if a software has to adapt to regular changes in its environment.

There are some limitations, however. Learning rules from data requires that you have a large data set of typical cases available. Furthermore, the data must be tagged with the desired outcome. Sometimes suitable data is generated as a side effect of some existing business process or is published as open data. If not, collecting and labeling data can require considerable effort, which might be expensive.

It's really a continuum

In this article, I have contrasted Machine Learning and traditional programming to better highlight their characteristics. This may make the distinction appear starker than it really is. It's really a continuum of how much the application functionality is affected by data as opposed to explicit decisions by a programmer.
Let's imagine we're building a search engine for Wikipedia. In the rigid programming extreme, a search engine might simply return



all documents which contain the exact search terms. Not all terms are equally descriptive, however. Terms that occur frequently in a cluster of documents, but are quite rare overall, are distinctive and should have greater influence on the ranking of the documents. In this case the functionality of the search engine depends partly on the data, namely term frequency.

Another step towards better utilization of data would be the PageRank algorithm which identifies important pages by analyzing the network of links between the pages. Google resides even further on the Machine Learning end of the spectrum. It uses a mix of signals to try to capture the semantic content of the search terms and aims to provides meaningful answers, not just matching search terms.

There is also another way how Machine Learning and traditional programming will approach each other. I believe that in the future it gets easier to experiment with intelligent features, such as recommendations or machine translation. More and more Machine Learning solutions will be published as handy services and reusable components.

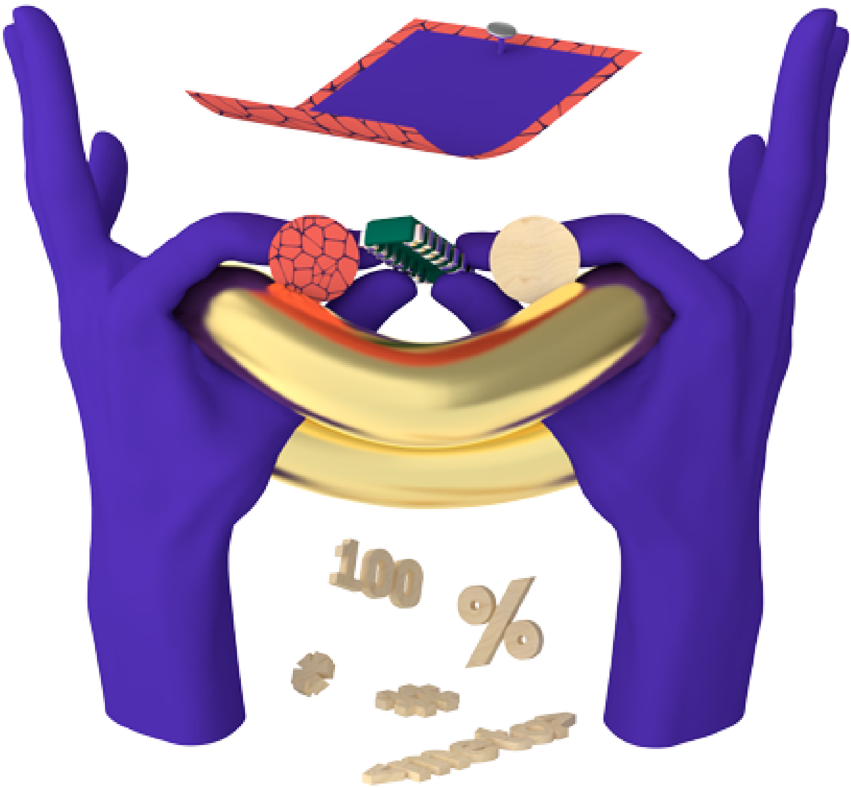
Machine Learning complements traditional programming by mining rules from data. It is useful in complicated cases where writing the rules by hand is unfeasible. Traditional programming and Machine Learning have their distinctions but also share a close kinship.

Fantastic Problems and Where to Find Them

Daryl Weir - Senior Data Scientist

A very nice primer on Artificial Intelligence, Machine Learning and related concepts. Daryl's article attempts to sidestep the hype and take a realistic look at the implementation potential, now and in the future.

Recommended for everyone.



100
%

This article is a written companion to a talk of the same title I've given in a few different tech meetups. The slides for that talk can be found on Github and Slideshare.

Artificial Intelligence, Machine Learning, neural networks, deep learning... If you follow technology news, chances are you've seen these terms. There's a lot of hype around Machine Learning these days, and it's hard to know how it applies to your own business or project. Here, I'll cut through some of that hype, identify some real use cases, and give some pointers for how to identify problems that Machine Learning can solve.

Artificial Intelligence (AI) is not a very useful term in this discussion. It's a powerful, evocative phrase, conjuring up images of HAL from 2001, Skynet from The Terminator, or maybe GLaDOS from Portal, depending on your pop culture tastes. Science fiction often portrays AI as a system that thinks, reasons, and learns in a human-like way, but, frankly, that's pretty far from where we are.

When you read a headline like "AI learns to do task X", it actually means that a researcher or data scientist wrote a mathematical description of the task, collected a bunch of example data, and used techniques from Machine Learning to learn a set of rules from that data. It's a laborious process that requires a lot of human oversight. Allison Parrish put it pretty well in a Tweet on March 26, 2016 at 3:26 PM:

> a cool thing to remember is that whenever someone says “A.I.” what they’re really talking about is “a computer program someone wrote”
> - @aparrish

This kind of “AI” has a narrow focus on answering a single question, and can’t generalise to function in totally new contexts as a human can. If I show a chess board to my self-driving car, it’s not going to learn how to play.

However, all this is not to say that these efforts are useless. Within their own narrow areas of focus, Machine Learning systems can often achieve super-human performance. Amazing things can be achieved by choosing the right questions to answer

Machine Learning

Machine Learning is the study of algorithms that learn from data. Rather than explicitly writing a series of steps to solve a problem, as in traditional programming, you input a description of the goal and a large number of past examples. The system then uses mathematical and statistical techniques to learn to accomplish the goal as well as it can. In some sense, it’s like having the computer “program itself”.

Machine Learning is a toolbox of techniques that let us learn from examples. Deep learning is one of the tools in that toolbox and has been successfully used in many different areas in recent years. Deep learning is based on the artificial neural network, a technique that's been around since the 1950s. Recent advances in both computational power and the training process for these networks have allowed deep learning to flourish.

The most common type of Machine Learning is what's known as supervised learning. This covers tasks of the form "given a situation A, predict an outcome B". Each piece of data is a historical situation, labelled by its outcome. Based on these examples, the machine learner tries to learn to predict the outcome for new, unseen situations. The trick to creating a useful system lies in choosing the right A and B.

Self-driving cars are a good example. One of the many problems they have to solve is how to adjust the steering to stay on the road. The simplest way to do this is to take a picture of the road in front of the car, and predict the correct steering direction from that. Our situation (A) is an image from a camera, and our outcome (B) is a steering wheel angle.

Navlab 5, as driven by RALPH.

Carnegie Mellon University created a neural network based system called RALPH that handled the steering wheel of a mini-van as it drove nearly 3000 miles from Pittsburgh to San Diego. RALPH remained in control 98% of the time on this cross-country drive. The human operating the pedals had to take over the wheel for trickier intersections. The coolest part? CMU did this in 1995. The technology has been around a long time, but is only now becoming mainstream.

What is it good for?

The obvious next question is: when should we use Machine Learning? What kind of problems is Machine Learning good at solving? Andrew Ng, a professor at Stanford University and prominent Machine Learning expert, has a pretty far-ranging answer for this. Writing in Harvard Business Review, he said:

> “If a typical person can do a mental task with less than one second of thought, we can probably automate it using AI either now or in the near future”

That’s a bold statement, and I’d argue there are plenty of things humans can do in less than a second of thought that are not good candidates for Machine Learning. As a simple counterexample, think about the problem of parity: determining whether a given number is odd or even. I hope you’ll agree most people can do this in less than a second. Plus, it’s trivial to collect labelled examples

- 322 is even, 343 is odd, 42 is even, and so on - that could be used to train a Machine Learning system.

The problem is that given data like this, many popular Machine Learning algorithms will completely fail at predicting the parity of an unseen number - they'll be no better than random guessing. This problem can be overcome by giving the algorithms more inputs to work with, but this requires problem-specific knowledge. The point is, there are problems where a blind application of Machine Learning will yield nothing.

Here's an alternative rule of thumb. You might have a Machine Learning problem if:

01 It's difficult or impossible to write down a set of rules, but

02 It's easy to collect historical examples

The parity problem fails on the first point. The rules are very simple: even if the number is divisible by 2, odd otherwise. There's no need to use Machine Learning when the solution can be simply expressed as a regular program.

An example of a problem more suited to Machine Learning: “is there a cat in this image?”

Is this a cat?

Photo by Mohamed Nohassi on Unsplash

This passes the Ng criteria: it takes less than a second to judge an image. It also meets our two additional conditions. It's very hard to write down a set of rules that takes an arbitrary image and determines the presence or absence of a cat. However, luckily for us, the Internet is an elaborate machine designed to produce cat pictures, so it's easy to get example data.

Object recognition in images is a classic Machine Learning problem. Various techniques, especially those based on neural networks or deep learning, have had great success at solving it. Being able to accurately recognise cats is technically impressive, but not necessarily very useful from a business point of view. This was parodied in the show Silicon Valley with the app Not Hotdog (a real app they actually implemented). That gives us another important rule: before using Machine Learning, make sure the outcome you're predicting is actionable and creates value for your business.

Use Cases

Having rules for when to apply Machine Learning is good, but it's also useful to see how other people have already successfully applied these techniques. There's a huge number of Machine Learning algorithms, but many popular applications fall under three main categories.

PREDICTING THE FUTURE

The task of supervised learning described above is perhaps the most common application of Machine Learning. Essentially, the goal here is to predict the best future actions given the available historical information. Such predictions can be further broken down into two classes, depending on the type of quantity to be predicted.

The first class is regression, where the goal is to predict a number.

Demand forecasting for supermarkets is one example: given a product, how much should I order for my next stock delivery? Store owners want to make sure there's enough product on the shelves to satisfy demand, while avoiding overstock taking up expensive space in warehouses. It's especially important not to order too much of perishable goods that will have to be thrown out if not sold in time. Machine Learning models to make these predictions take into account past sales, similarity between products, demographics of each store's customers, and many other factors.

The stock market is another example: what's the optimal price to buy or sell a given share? Human traders are increasingly supported - or even replaced - by algorithmic systems that re-estimate prices and make transactions hundreds of times per second. Such systems buying and selling from each other account for an unnerving share of the world economy. Their pricing models bring in historical trends, real-time transaction data, tips from human advisors, and even sentiment analysis scraped from social media.

The other class of problem is classification, where the prediction is one of a finite set of options.

This might be a binary yes/no choice, or a selection from a bigger set of options, depending on the exact application. Think about face recognition: Snapchat might be interested in whether or not a face is in an image, so they can decide to apply filters; while Facebook might be more interested in whose specific face is in the image, so they can make tag suggestions. Futurice have employed face recognition at Helsinki Airport as a prototype for a hands-in-pocket travel experience.

Classification is also important in the medical domain. Image recognition techniques can process thousands of past scan results to learn the characteristic patterns of many diseases, resulting in diagnostic accuracy that is approaching and in some cases exceeding that of human doctors. Other algorithms can blend diverse data about symptoms, test results, and live vital readings to make accurate diagnoses. In bio-medical research, huge datasets on gene expressions and chemical signatures allow new links between diseases, genetics, and treatments to be discovered.

RECOMMENDING CONTENT

The second major category of Machine Learning application is content recommendation. Lots of businesses have content of some kind. This might be traditional media - written articles, videos, and so on - or products sold in a store. Google's main content and source of revenue is ads. Whatever the format of the content, not all items will appeal to all people equally, so companies have a vested interest in personalising the content shown to each user. As a Machine Learning task, this amounts to predicting how much a given person will like each piece of content, so that recommendations can be made.

One major strategy is content-based recommendation, which is exactly what it sounds like: suggesting items similar to content a user has already liked. This can be as simple as recommending articles by the same authors or videos from the same channels users have looked at before. Alternatively, it could involve analysis of the text in an article to extract the main topics, then recommending other articles on the same topics.

There are a couple of problems with this approach. First, it's not always easy, or even possible, to define similarity between items. This is especially true for companies with diverse offerings, like Amazon. How similar is a vacuum cleaner to the latest PS4 game? Secondly, there's the problem of what to recommend for new customers who haven't consumed any content yet. We don't know their tastes, so we can't recommend similar items.

These problems motivate a second approach, called collaborative filtering. The idea here is to recommend items other users have liked. When a user is new, you can just recommend the most popular items across all users. As they consume content, you can personalise the recommendations based on the habits of similar users.

This is how Spotify’s Discover Weekly works: first, your listening history and created playlists are analysed to create a profile. Then, Spotify finds other users who have similar habits to you, and creates a playlist from songs that those users have listened to, but you haven’t heard yet.

Netflix is another well-known user of collaborative filtering. Giving customers a steady stream of fresh content tailored to their tastes is an important part of keeping them as subscribers. Famously, in 2007 they announced the Netflix Prize, a competition offering $1,000,000 to anyone who created a recommendation algorithm that was at least 10% more accurate than Netflix’s own. Teams of researchers competed for 2 years before one group broke the 10% barrier and won the money. Interestingly, Netflix never used the winning solution in production, partly because of engineering restrictions and partly due to a business model shift from DVD mail order to streaming.

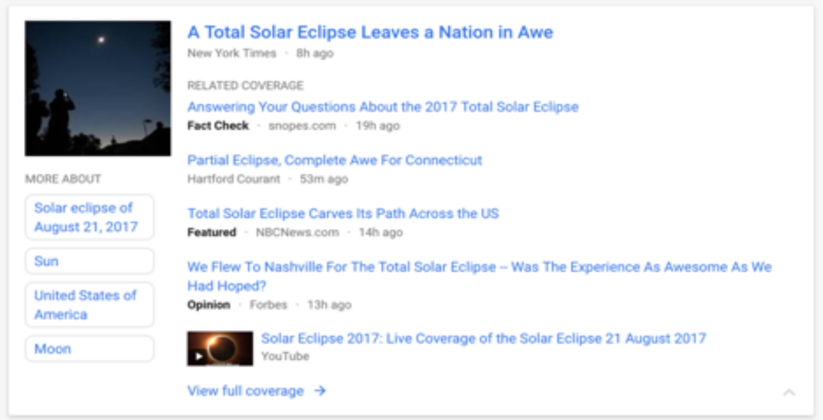
UNCOVERING HIDDEN STRUCTURE

The last big category of use cases is a bit different. In regression, classification, and recommendation, we were trying to predict some output for a given situation. What if we don't have something specific to predict? What if we just have lots of data about past situations, and want to know what's interesting about it? This is the problem of unsupervised learning. This is a much harder problem than the others, because there's no "truth" to compare against. The results are generally subject to human interpretation.

Unsupervised approaches allow you to answer different kinds of questions, like: What kind of groups exist in the data? Is some part of the data unusual compared to the rest? What are the recurring patterns in the data? The answers to such questions can be useful on their own, or they can be used as a preprocessing step before applying supervised learning techniques.

User segmentation is one of the most common applications of unsupervised methods. Consider a company who operate a mobile game, and maintain an extensive log of the actions players take in the game. By analysing these logs, the company can identify groups who act similarly. Maybe one group logs in regularly, and spends money on the in-game store. Another logs in every other week, and plays for a few minutes without spending money. These groups can form the basis for further analysis: the company might look for the kind of levels the first group plays the most, so they can create similar content and keep them engaged; or they might target offers to increase the playtime of the second group and convert them to paying customers.

Financial institutions make heavy use of unsupervised techniques to combat fraud. When processing credit card transactions, they are on the lookout for purchases that don't seem to fit established patterns. It can be hard to get good data on which transactions are truly fraudulent, but by considering user history, unusual items can be flagged. Systems might consider factors like time since the last transaction, number of transactions in the recent past, difference from the average transaction amount, and physical location of the vendor compared to historical purchases in order to assign a novelty score to each transaction. When the score is high, purchases might be automatically declined or flagged for manual review.



A Total Solar Eclipse Leaves a Nation in Awe
New York Times · 8h ago

RELATED COVERAGE

Answering Your Questions About the 2017 Total Solar Eclipse
Fact Check · snopes.com · 19h ago

Partial Eclipse, Complete Awe For Connecticut
Hartford Courant · 53m ago

Total Solar Eclipse Carves Its Path Across the US
Featured · NBCNews.com · 14h ago

We Flew To Nashville For The Total Solar Eclipse – Was The Experience As Awesome As We Had Hoped?
Opinion · Forbes · 13h ago

Solar Eclipse 2017: Live Coverage of the Solar Eclipse 21 August 2017
YouTube

View full coverage →

MORE ABOUT

Solar eclipse of August 21, 2017

Sun

United States of America

Moon

Google News automatically gathers related stories together and lists the key topics.

PUTTING THINGS TOGETHER

Some companies are starting to use supervised and unsupervised techniques together. Google News is a great example. The site aggregates stories from many media outlets, and unsupervised learning automatically groups different versions of the same story from different authors. The key topics of the top stories are also extracted automatically and shown in the sidebar. Meanwhile, classification is used to assign stories to the appropriate categories (Business, Health, World News, and so on). There's also content recommendation, as the service takes into account user preferences and offers a personalised list of articles. All this is done with minimal input from human editors.

Summing up

Machine Learning is here to stay. Businesses are turning to these tools as a way to make sense of ever-growing datasets and seek competitive advantages. Big companies like Google, Facebook, and Amazon have talked about the transformation of their engineering and business to focus on data driven approaches. Academic publications and open source libraries like Google's TensorFlow are enabling wider access to these techniques. Even the notoriously insular Apple have started allowing their employees to publish Machine Learning papers, just so they can compete for the pool of talented researchers.

Despite this, Machine Learning remains shrouded in hype and harried by fear-mongering around the perils of Artificial Intelligence. If you made it this far, you should have a better idea of where we really are with this technology, and how people have found success with it so far.

If you skipped to the end, here are some key points to take away:

01 Machine Learning systems aren't general purpose: they excel at providing answers to narrow, well-defined questions

02 Look for business problems where it's impossible to write down the rules, but easy to gather examples

03 Whatever you set out to predict, make sure it's actionable - if you can't change something in response to the prediction, it's useless

Daryl Weir was part of the team who designed our Intelligence Augmentation Design Toolkit. It's a great resource for designing intelligence into services. Check it out at http://iadesignkit.com/

Six inspirational ways to make money with data

Mika Ruokonen Business Director

A rumination on the many ways we can make money with the data we own or have access to.

Recommended for anyone interested in exploring the business potential in AI/ML

Photo by Markus Spiske on Unsplash

Companies have woken up to the importance of digital data in their business, as well as the new business opportunities that data gathering and analytics present. They've started to understand and identify what valuable data that they possess. They're collecting it, analyzing it and using it to improve their business and operations. They've also started to explore machine-learning applications that help them solve complex problems that can be answered through insights gained from large volumes of data.

The two most obvious ways for companies to commercialize data are:

01 Data is collected and analyzed for product development purposes, used to create better products, which are then to customers. This results in increased sales, products with higher added value or more closed deals.

02 Data is used to identify problems and bottlenecks in internal processes, which are then eliminated to improve business efficiency and profitability.

We help companies understand the commercial opportunities digital data provides and believe that these two ways are necessary, but just not enough. Companies must go beyond the obvious and apply creative thinking to their data commercialization efforts. They typically master the structured engineering-like process of data collection and analysis, but fail in the innovative and commercial side of things. They fail to make use of and commercialize all the data they have.

We want to inject fresh, new thinking into data commercialization and provide perspectives that companies tend to neglect. Following are six inspirational ways for companies to make money with their data, with real-life cases to illustrate how they work in practice.

1. Selling insights to customers

Taking existing data, aggregating and enriching, and then selling it to customers as new valuable insights makes sense. Reports, online dashboards and indexes can be standalone products bundled with the company's existing offering, helping increase the price of the bundle that is sold to customers. User interfaces can be augmented with Machine Learning applications to help customers get what they need or interact with the brand. New insights are created "on-the-go", as a part of the customer encounter.

An example: Oikotie.fi, a leading player in job portal business in Finland, has created a smart way to commercialize its data. Oikotie.fi created a payable product for its B2B customers: after paying a fee, recruiters can see how their job ads perform against the rest of the (anonymized) players of their industry.

This data helps the B2B customers improve their recruitment advertising and provides the job board with new revenues. (For the sake of full transparency, the writer of this blog used to work for Oikotie.fi earlier in his career)

2. Empowering the sales force with data

A sales organization’s role in a company is to maximize sales. Data can be a highly effective tool in reaching that target. Smart companies empower their sales force with rich customer data sets that help them easily identify customer problems, potentially churning customers and sales leads. With data, the sales force can give better product presentations, improve customer service and use more tailored sales argumentation when meeting customers. Smart companies position themselves as outstanding data leaders, and sales people play an important role in delivering that message to customers.

An example: The Finnish elevator and escalator company Kone is a good example of a firm that has given its sales force access to valuable data. When a sales rep meets a customer who has a Kone elevator or escalator installed, he/she is well informed on the condition of the device, including any potential problems, and can help the customer extract the most value out of the device. Often this leads to additional maintenance services and/or spare parts sales.

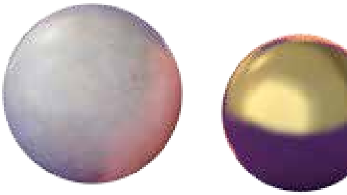
3. Using data in marketing and advertising

Data that tells us something about consumers and their interests can be used to create marketing and advertising solutions. You have two options: either the company uses its data to optimize its own marketing and advertising or sells its data to other companies, so they can do it.

An example: The search-and-discovery service app Foursquare sells its data to retailers so they can optimize their outdoor advertising and online marketing to match the routes people use to navigate the city. Media companies collect digital data on people's interests - e.g. sports, food, clothes, etc. - and sell it to online advertisers. Weather forecasting companies like Foreca can help clothing retailers optimize their advertising to match the weather conditions (e.g. sales of umbrellas, swimsuits and winter jackets). eCommerce companies like Zalando collect data on unfinished shopping carts in their services and target their own online advertising at the Facebook feeds and other channels of those potential buyers.

4. Selling data to players up and down the industry value chain

Companies often view their business as a "silo" and the data they have, they've derived from their own operations and own customer interactions. They use it for their ownpurposes only. The



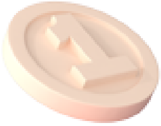
reality is that they've been operating in network environments and value chains where the final customer delivery is the result of the joint effort of several collaborative companies. In recent years many companies have woken up to the fact that these networked business environments create opportunities for sharing and capitalizing data from company to company. Data can be an important asset in optimizing the operations and cooperation of the players in the value chain. Companies can monetize their data by selling it to their suppliers and vendors down the value chain or by selling it to retailers, resellers and other sales-related partners up the value chain – or both.

An example: The provision of data from one player to another is typical in pharmaceutical value chains. Finnish pharmaceuticals distributor Tamro sells data on drug purchases by the Finnish people to local pharmacies, who can compare their own sales against their competitors and optimize their drug display and stock in stores. Tamro also sells data to drug manufacturers like GSK and Novartis, who then use the sales data to set prices for their products. Tamro very cleverly capitalizes its data both up (pharmacies) and down (drug manufacturers) the pharmaceuticals value chain.

5. Selling data to players outside your own industry

A not-so-obvious opportunity to monetize data is to look outside the company's industry or value chain. There might be a number of players interested in insights on economic activity, consumer behavior or other relevant topics. These players might be found in surprising fields. Companies should actively seek out these players and explore innovative joint opportunities.

An example: Due to a legislative changes in the EU, retail banks like Nordea and Danske Bank are required to open their accounts and payments data for use by others. This opens up lucrative, new business opportunities for fintech, information services, telecoms firms and others with the capabilities to create new digital services based on consumer banking data. Another potential example comes from the sportswear industry: companies like Nike have started to collect data generated by smart sports clothes. In the future they may start to offer that data to insurance companies, who can use the data to predict a customer's health insurance needs. A third example: export and import volumes of goods in harbors are a good early indicator of economic activity in a given country, so some harbor operators have started to sell that data to players who wish to forecast economic fluctuations, such as banks and financial institutions.



6. Using data to increase company valuation

The ultimate way to make money with data is to consider data as an asset in the company´s balance sheet and sell the entire company to a buyer who desperately needs the data. In this case, the data is considered valuable, because it can help the buyer to grow or improve its business. The data is sold as a full package, as part of the rest of the company assets.

An example: Would Facebook have paid $22 billion for the shares in WhatsApp in 2014 if the M&A target didn´t have data on 600 million consumers worldwide? Probably not. WhatsApp's revenues were very low at the time of the deal - only $10 million in 2013. The reason Microsoft paid $2,5 billion for Minecraft in 2014 had little to do with the actual revenues of the acquisition ($330 million in 2013). It was the data. A key element in the very high valuation of these acquired companies was the consumer data they possessed.

Making money with data is a gigantic, worldwide business opportunity and many smart companies are already moving ahead with their data commercialization plans. Partnerships with players in and outside industries are instigated with the aim of combining data from different sources to enrich it into valuable, new insights. Functions within companies, such as sales, marketing and product development, are benefiting from rich data and able to create new revenues. Some companies even approach their customers directly to offer data as a new packaged product - something that generates revenues immediately.

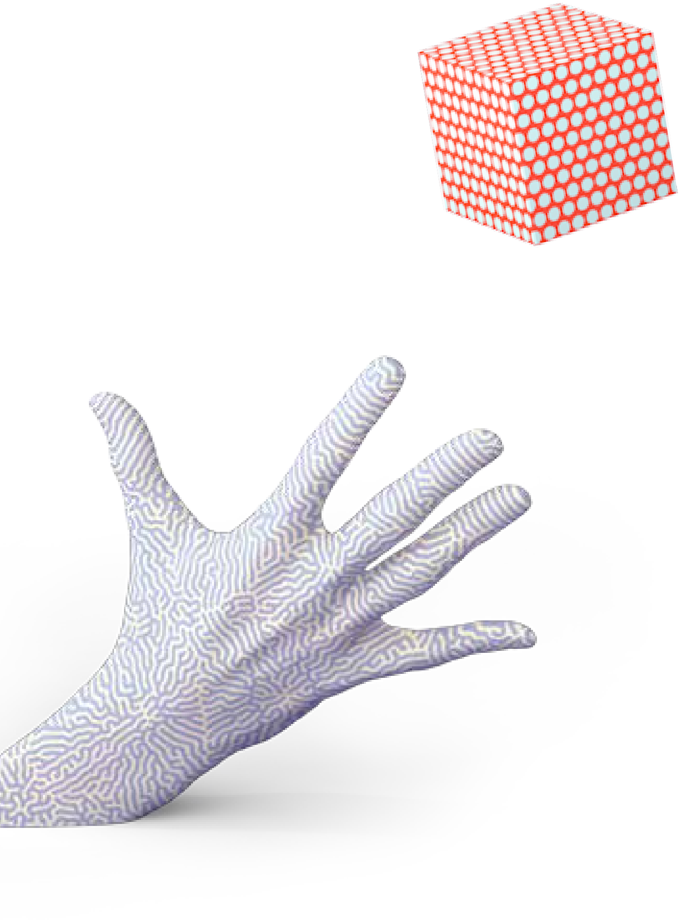
In the hands of skilled experts, data combined with the high processing power of computers enables the creation of Machine Learning applications with the potential of revolutionizing entire industries. These technologies will help companies move from the reporting of past events towards monitoring real-time events and, eventually, towards predicting what is likely to happen in the future. Given the recent advances in Machine Learning and the six innovative ways of making money with data mentioned above, I believe we're about to witness a real proliferation of companies whose business is heavily reliant on data exploitation.

There are legal issues related to data commercialization and consumers' privacy concerns must be taken seriously. Governmental players like the EU are actively and constantly shaping the way data should/can be made available and how it's stored, so companies should stay on top of legislative changes, now and

in the future. Whenever the law allows data analysis, disclosure, transfer and sales, there's plenty of room for innovative thinking, technology exploration and new business creation.

Mastering Machine Learning and data commercialization is serious business and requires management's full attention. Any company not doing its best at it is most likely leaving loads of money on the table.

AI Glossary



Artificial Intelligence

Systems capable of making decisions and/or solving problems, with or without gathered data. Though early chess AIs relied on nothing but written rules, an AI can't get very far with just manually written rules. This is why we need Machine Learning, which learns from data to make decisions. For example: using thousands of cat photos, it can analyze what the similarities are and then find even more photos of cats on the Internet – which, as we all know, is a challenging task.

Autonomous

An autonomous AI doesn't need people to help it fulfill its assigned tasks. At the time of writing, driverless cars are the most notable and visible example of AI autonomy. Autonomous is not yet sentient. It's not Skynet. AI sentience is still very much a theoretical concept.

Algorithm

The algorithm is the "brain" of AI. They're math formulas and/or programming commands that instruct a non-intelligent computer to solve problems using Artificial Intelligence. They are rules that teach computers how to learn and figure things out on their own.

Black box

Based on the applied rules, an AI does a lot of math that humans can't even begin to understand. Despite this fact, the information the AI outputs is often useful. This is called black box learning. We don't know what happened and how the computer made the decisions it did, but we know what rules were used, so it’s still under control.

Deep learning

Deep learning is a deep neural network with many hidden layers. Deep learning can be used to learn and extract good features directly from raw data such as images or audio. As it processes data, the AI gains an understanding of the issue at hand, be it recognizing pictures of cats or something less important.

Machine Learning

Machine Learning is the beating heart of AI and currently the most popular approach. They are closely connected, but not at all the same thing. In Machine Learning, the AI uses algorithms to perform Artificial Intelligence functions.

Supervised learning: You provide example data and target outputs for the examples, and the machine tries to learn a mapping from the examples to the targets. For example, you give a bunch of cat and dog images to a computer and tell it which consist of cats and which of dogs. If an AI has learned something, like how to tell if a given picture is of a cat or a hot dog, it can build on this knowledge without your instigation. It trains itself and the next time you check in to see what’s up, it’s even better at finding cats on the Internet than it was previously.

Unsupervised learning: this is the stuff in sci-fi movies. Machines are capable of learning on their own, using reams of data and their processing capability. Here the machine is not provided with an answer, so it finds whatever patterns it is able to. In practice, one can use unsupervised learning to find clusters of similar data points (e.g. customer segments) or structure in the data (e.g. topics).

Reinforcement learning: this is when we give an AI a task without specifying the wanted result. It will probably find a number of potential solutions to the problem we've presented. By cutting in and providing feedback on which solutions are optimal, the AI adjusts and provides even better results the next time around. For example, one can teach a computer to play chess just by limiting possible actions based on the rules of the game and providing feedback based on who won the game. A computer can be also trained to play chess by letting it play against itself.

Natural language processing

Human language is very complex, so it takes a very advanced neural network to work with it. Natural language processing is when an AI interprets human communication. It is used by everything from simple chat bots and basic translation solutions to assistants like Alexa and Siri.

Neural networks

Neural networks are one of many Machine Learning methods. Originally they were inspired by how the human brain works. A neural network provides an AI with the ability to solve even very complex problems by breaking them down into levels of data, starting from simple pixels and basic similarities between various cat pictures, and advancing to increasingly complex facets of the picture.

Thank you for your time!

Let´s stay in touch.

HELSINKI
Annankatu 34B
00100 Helsinki
Finland
Managing Director Finland
Teemu Moisala
teemu.moisala@futurice.com

TAMPERE
Kelloportinkatu 1 D
33100 Tampere
Finland
Vice President & Site Head
Riku Valtasola
riku.valtasola@futurice.com

LONDON
26 Underwood Street
N1 7JQ London
United Kingdom
Managing Director UK
Tom McQueen
tom.mcqueen@futurice.com

BERLIN
Schützenstraße 6
10117 Berlin
Germany
Managing Director
Gian Casanova
gian.casanova@futurice.com

MUNICH
Implerstraße 7
81371 München
Germany
Managing Director
Helmut Scherer
helmut.scherer@futurice.com

STOCKHOLM
Kungsgatan 5
111 43 Stockholm
Sweden
Managing Director
Henrik Edlund
henrik.edlund@futurice.com

futurice