**Modified**

*TOO LONG*

**Summarise**

*TOO LONG TO SUMMARISE CORRECTLY*

***Important***

**MACHINE LEARNING**

We will look at different types of machine learning. supervised learning, inductive learning, unsupervised learning. And then there is this kind of reinforcement learning where the system operates in an environment. Let's go into more technical details.

**CLASSIFICATION OF CANCER WITH SUPERVISED LEARNING AND UNSUPERVISED LEARNING**

A lot of the time the learning task involve input with multiple dimension. We don't look at the tumor size only. We probably look also at the age of the patient and many other attributes. When you actually have multiple dimensions then the supervised learning will be more challenging.

**MACHINE LEARNING 6, THE PROCESS OF LEARNING**

All right, so we already go through different types of machine learning. We talk about supervised learning, unsupervised learning, semi supervised learning and also reinforcement learning. After identifying the target function, the next thing is to apply the appropriate learning algorithms. And this knowledge will be sent to the performance elements that the agents is using to make the decision.

**TRAINING VS. TESTING**

The best way in order to obtain the training and the test data is that they are independently drawn from the same overall distribution of data. If the test distribution is different, then the technique that use typically is called transfer learning.

**MACHINE LEARNING**

There are many machine learning algorithms. All of them required to contain these three components representations. Choosing the target function to learn and choosing the right representation for that function is part of the representations in every machine learning algorithm. After that evaluate how good or how bad that model is against some test data.

**WHAT IS MODEL TUNING?**

The concept of model tuning involves avoiding both overfitting and underfitting. You may have to consider the hyperparameters that the machine learning algorithm use when training the model. And then this is again easily confused with the evaluation with test data.

**MACHINE LEARNING: THE PROCESS**

Machine learning practitioners need to understand the domain that the machine learning system need to be developed. Learning can be viewed as using either direct or indirect experience. I'm planning to produce another videos to give more details about some advanced topics related to machine learning.

**Original**

Hello. Okay, now we have looked at what machine learning is and also we know about the reason why people, companies, businesses may want to use machine learning to add value to their business. Let's go into more technical details. So the first is we would like to look at different types of machine learning. We start with the most popular one, that is supervised learning, also called inductive learning. So in supervised learning, the machine learning system will be given training data and the design output for each piece of training data called the labels for that piece of training data. So for instance, if we want to classify the image to be whether that is a number 12345 and so on and so forth, then for each of the input image we have the label that is the image associated with the digit five, that image is associated with the number seven and so on and so forth. Now, with those labels for the training data, then the system will be able to train a model, a machine learning models, so that after it's trained with thousands or millions of pieces of input data, then it will have a model to recognize the image and then classify them into the digit, the corresponding digit. And when you give it a new image, it will be able to recognize this a number one or this a number nine. For instance. Another type of machine learning is unsupervised learning. So in this case, the system will be given training data and there is no labels for the data. Typically because we have no such label, we have a lot of data and then we need to make sense of the data, but we don't have the label for the data. And in general, getting this label may be quite time consuming and then a lot of effort involved. And so this is another type of learning that we want to consider as well. So unsupervised learning typically allow you to group the data into certain clusters, core clustering, because they have some similarity, or you want to associate different groups of data points to certain proofs or to certain kind of insight that you say that if they belong to this group, then they review this type of insight or they will satisfy this kind of rules. And the third kind of learning is a mixture of supervised and unsupervised learning. In this case we have tuning data, but only a few of pieces of this tuning data and there are potentially a lot of them. So you have millions or billion pieces of such tuning data, but only a few of them have the labels. And so the typical example is emails. So we would like to classify the email to see whether an email is a spam mail or a legitimate mail. But even with legitimate mail, some of them belong to some social networks, emails from LinkedIn, from Facebook. Some of them may be, even though they are not exactly spam because the user does want to see it from time to time, but it does come from companies that provide promotional material about their product or their business. And so even though it's not entirely spark, but it's probably considered to be promotional emails. And so not all of these emails has a label because the users only select a few emails that they consider to be really best emails. They are spam and then they explicitly say to the email providers, email service providers such as Gmail or other service providers email service provider, that this particular email is a spam mail. But a lot of the time they don't really put any label. So even though the companies incorrectly recognize an email being promotional and then it is not promotional, but then the user never really bothered to correct it and then telling the email service provider that these promotional emails are not a promotional email. So the system, the machine learning system will have quite a lot of such data and only some of them have label. So again, how to deal with this kind of data and then what insight can the machine learning system get from the data is the task of semi supervised learning. And then there is this kind of reinforcement learning where the system actually operates in an environment. This system does have its sensors, it has the actuator to actually make changes to the environment. And so the system has some decision making mechanism to allow it to upon receive certain observations from the environment, make a decision to perform some action. And because the system is still learning, so it does not always perform the optimal actions. And so when it performs some action, it observed that, well, my action doesn't really give me the best outcome. And then again, I try another actions in another state and then this is really good somehow, even though I didn't know that it's so good. But then after I perform it, and then it actually give me a very good outcome and so on and so forth. So a lot of these experience and the system really want to learn from each experience rather than just doing random things like million of times, collecting all the data from all those a million experience and start learning, like in the case of supervised learning or unsupervised learning. Now, I really want to learn from every piece of experience in order to improve my decision making mechanism. And so that is a reinforcement learning because every sequence of action lead me to a state, I observe that state and then I want to learn from seeing that my sequence of action bring me to that state. All right, so let's go through these on a very high level. Okay, the first one is supervised learning. So remember that in supervised learning you have data and for each such data you have the label. So the data will be in the form of, for instance, the input is s one and then the label for that data will be y one. But if the input is s two, then you have the label y two and so on and so forth. If the input is n, you have the label being y n and n could be small like 1000, but could be very large like 1 million, 1 billion. So now you have all this training data that you can learn from. Then one of the supervised learning tasks is called regression. So regression basically say that what the machine learning system need to do is to learn a function called a target function f. This target function f maps every possible input x and then try to map it into an output y. So essentially after you learn this target function s, then you will be able to predict for a given x the appropriate output y. If you do that for every x and then give a real value for y, then we have a regression task. Okay? And so this particular example is a good example for a regression task. So in this case you can see that given a year in the last, let's say that few decades, and clearly you also want to predict into the future as well. The question is we would like to see the extent of the Arctic sea ice area in September. So the Arctic is in the North Pole and then now we are very concerned that because of global warming, these areas of the CIS tent is becoming smaller and smaller because of global warming. And so in September so that is in the northern hemisphere, that is at the end of the summer and that is when the sea ice area becomes smaller. It's the smallest during the year. And clearly we worry that this area is become smaller and smaller. And if you look at the data points in the past, you can see this trend, right? So back in the 1980s the area is still quite big. So it's around eight or 7.5 million km². But then as we progress over the time to 1990, it's become smaller, it go to around 7 million km², go to 2000, it's already dropped out to around 6.2 million, then go into 2010. Then you can see that it's now staying at around 4 million km². So clearly there is a trend from 1980 to 2010 that this area has been decreasing quite quickly. Now the question is, can we predict into the future? So in 2025, in 2030, what could be this CI expense in the north pole then? Based on these data points that we collected in the past years, since 1980s, is there any prediction that we can make? So this does become the regression because for each year, for instance, 20, 15, 20, 20, 20, 25, 20, 28, can we actually get a numbers of this CI extent for the Arctic? And so this number y that we would like to predict based on the input being s, being the year that we want to predict, then this real value for y become a regression problem for the machine learning task. Okay? And so you can see that the machine learner could output a few possible models. The model in the red lie, which is a linear lie, so is trying to approximate the data points that it learned from. And then you can see that this red lie is probably not so accurate because as it's moving through the year, then you can see that in the recent year it becomes less and less accurate because it doesn't really match these data points very well, right? On the other hand, a nonlinear curve, the green one over here. So this one is another machine learning model that has been trained using these data points. Now, it seemed to be matching these data points especially during the later years in the little bit better than the red line. So for these two models, perhaps the right answer would come, the better answer would come, the more accurate answer would come from the green line models rather than the red line models. So you can see that in this case, the machine learning path is a supervised learning because we do have the label, the value y for each of these data points. So the data points telling us that in a particular year, then we can have the size of the Arctic sea ice extent, okay? And then based on these labels, we are able to construct a few models. In this case we have two, the red line and the green curves. And then both of them allow us to predict the Arctic CI extent given the year x. Another supervised learning is classification. So classifications in term of the input is quite similar. So it also have the input s, one s, two SN. The label now is not a drill value, is a categorical value. So it can be categorized into, let's say that two categories. For instance, in the case of this example, so it's classified a tumor being benign or malignant, right? So two categories. But then of course it might have more than two categories. It can have 310 categories for instance. But the idea is that these are discrete categories. And then our aim is to learn a target function f. And after this target function F can be learned, we have a machine learning model to allow us to given any input f, then we will be able to predict this category for our output value y. Okay? So is it belong to the category zero, the tumor being benign, or does it belong to the output category one that indicates that the tumor being malignant? And so for the classification problems, then the input being the tumor site. And so you can see that now this classification problem can be visualized by this lie where we actually put our data points on this lie, right? And so we have all these data points for the blue dot and the red dot on this data, on this line for the tumor size. And then after that the classifications probably will split the data points into two size. And so when the tumor size is smaller than this value here, indicated by this very scat here, then it will be classified as being denied. So that means that small tumor size and smaller than this particular value for the size of the tumor, then the tumor will be classified as being benign. And if it is the big tumor, then in particular it is greater than this particular value that is specified by this line here. By this vertical line here, then we predict that the tumor being malignant. Okay? So you can see that by doing this now we are able to classify a tumor by looking at each side and then compare the tumor size to this value that's specified by this vertical line. And then if it's more than this value, then we will predict that e is benign while it is predicted to be malignant if it is greater than this size. Now the supervised learning that we have seen so far are simplistic because we look only at one single dimension for the input value x. Okay? So in the case of the CI extent we only have the year being the input value. In the case of the tumor being malignant or benign, we only look at the tumor size. But a lot of the time the learning task involve input with multiple dimension. So we don't look at the tumor size only. We probably look also at the age of the patient and many other attributes. Well, so for instance, the history of the person, whether their family have a history of having cancer or not and so on and so forth. So when you actually have multiple dimensions then the supervised learning will be more challenging. And for instance, in this case you can see that we can no longer look at the input being single dimensions with the tumor size only, but now it becomes two dimensions or three or four or n dimension. So in this example we have the two dimensions where the inputs are having the two attributes being having the tumor size and having the age of the patients. And then now the training data will be placed on this two dimensional plane. And so the classifications now can use this particular linear lie in order to classify them into either vinay for the blue line for the blue points or malignant for the red points. As I say, the number of dimensions can be just one or two, making the problem much simpler. Or maybe it can be quite large. For instance, you also have the dimensions for the thickness of the clam or the uniformity of the cell size, the uniformity of the cell shape and so on and so forth. So that is a very quick overview of supervised learning with the two main tasks of all the, whether it is an agriculture learning or a classification learning for unsupervised learning. So I just give you a simple example. In this case, you have the data set containing a lot of data points, so all these gray points and then let's say that they are described by two attributes, so attribute x and attributes are Y, okay? And so with all these data points specified by these two dimensions, then we would like to make sense of the data, trying to get some insight of this data set. And so one way to do that is to see the similarity of the data points by clustering them into cluster. And after we cluster them, and then we see that these five points seem to be quite close each other, so we put them into one cluster and these five points also quite close each other. So we put them into another cluster and then these four points are close to each other, we put them into another cluster. So we have the cluster of blue points, green points and red points, and then after that we may find some rules associated with each of this cluster. And so for instance, we wouldn't try to cluster them like this, one cluster like this, and then another cluster like this for instance, because it doesn't make a lot of sense where this data point is quite far away from these tree and so that is probably not a good clustering there. Then we have reinforcement learning. With reinforcement learning. I already explained that if we have a sequence of states and actions, so clearly the agents perform a sequence of actions starting with some state. And so it starts from the initial state s zero and then it performs some actions and then bring it to another state s one and then perform another actions and then bring it to another state s two and then perform another action, bring it to another state s three and so on and so forth. Given this sequence of actions and state, the agents then trying to find an optimal policy so that it say that in whatever the state I'm in, doesn't matter whether there is s zero, s one, s two, s three, s, will be able to determine that the actions to perform in that state being actions, blah, okay? So that means that reinforcement learning will learn and output a policy which is mapping from the state to the actions that is going to take in that particular state. And there are many example of such reinforcement learning problems, for instance, how to actually assign the credit to a particular action, right? So when I'm performing a sequence of actions and then that sequence of actions bring me from the initial state to a very good end state after those ten actions, then which credits should I give to each of the actions? Is that the fifth actions that really bring me to the eventual good state? Or was it the seven action? And so this credit assignment problem is one of the problem that can be solved by using reinforcement learning and game playing as well. So you have a game of chess, a game of Go. You would like to learn to play the games now different sequence of actions and then against different sequence that the opponent is make will give you different outcomes. You can win that game, you can lose that game. Now again, in a particular state, when you are having a particular position of the board, of the chessboard, of the goal board, what will be the next action that you should perform? Or the robots try to navigate within a maze so the robots can be in this place in the maze or in that place in the maze, what would be the action to perform in each of those states and so on and so forth. So to capture the essence of reinforcement learning, so this is the kind of setting that reinforcement learning system or agents will be dealing with. So our systems called the agents will be operate within this environment. And so using the agent sensor, the agent will be able to observe the current state fi for the current time, I so at the current time I I observe the current state si and then because I have the policy to decide that at each state, what would be the action that I perform? According to my policy, I will choose the action AI. And then after I perform AI, then I will typically receive the reward by getting into a new state and then that new state is a very good one. Or maybe by performing the actions, then there will be a teacher who is going to give me feedback and say that you did well, wow. And therefore you get a mark of ten or you did not so well and therefore you only receive a mark of two. Okay? And so the rework will be used by the agent in order to decide on whether the action is a good one or a bad one in that particular state. And of course after I make the action AI, the environment change state and then in the new state, st plus one will go into the agents and then the whole process repeat. Okay? So the idea is that the agents observe the state of the environment. Based on its current policy, it choose the action and thanks to the reward it receives, it's now adjusting its policy. And so in the future maybe it again choose the same action because the reward is really high and therefore it's a good action. Or maybe because reward is so low it become a penalty. And then in the future the agents will no longer perform the action but choose a different action. And so example of reinforcement learning is that after applying reinforcement learning this unmanned aerial vehicles, which is a helicopter in this case was able to learn the policy so that it can actually choose the optimal actions and it can perform a lot of very interesting and difficult maneuver when it's flying in the air. Okay? So you can see that it can fly upside down, it can fly sideway like this one and so on and so forth, okay? And so all this showing that the agents that operate this helicopter has been able to learn using the reinforcement learning in order to choose the optimal policy to control the helicopters with different difficult maneuver. All right, so we already go through different types of machine learning, we talk about supervised learning, unsupervised learning, semi supervised learning and also reinforcement learning as well. Now, to just look at the idea of how we as the machine learning practitioner go in to design a learning system the first is that we will need to identify the kind of training experience that we have. So if the experience require the system to operate in the environment and every time it encounter a new experience use the experience to update its policy, then that clearly belong to the reinforcement learning. Right? But if the experience is the kind of data that the agents will be able to collect over time because this is about a large customer base with millions of customers using the reward card from coast, the flybys card and then now I have the data about all these 1 million customer and I would like to get some insights about my customer. So the data will become the training data with a huge set of data points after we got the training data. Could be the training experience for every step that I make if I use reinforcement learning. Or it could be a huge set of data with a lot of data points. In the case that I use supervised learning or unsupervised learning, then the next thing is that we need to choose what need to be learned. So that means that we need to identify the target function. So these target functions you probably still remember when I talk about supervised learning. So the function x that we would like to learn because it is a regression function that map each of the input value to a real value output y or it could be a classification because for each input value I would like to classify into one of a number of finite categories. After I determine on the target functions then the next thing is that we have to be able to represent the target function. That means that even with the task of regression or classifications there are many ways to represent it. So I can use a linear function but maybe the linear function is not good enough. For instance, in the case of CSPM, the linear function doesn't seem to be giving us a very good prediction into the future, right? So maybe a nonlinear function is necessary or maybe it can be just a tree like representations if you choose to use decision tree in order to represent that target function. And so there may be different representations for you to consider to represent this target function. Now, after you already settle on the training data and then you know about the target function that you want to learn, you know how to represent it using a linear functions, using a nonlinear functions, using a decision tree, then the next thing is to try to apply the appropriate learning algorithms. So if you use decision tree, that would be the decision tree learning algorithms. If that is the linear regressions, then that would be a different learning algorithms and so on and so forth in order to actually view the model for these target functions. So you will have a model for F and that model F hat will be the one that produced from your learning algorithms by training over the training data here TD here using these learning algorithms on the representation of the target functions. Okay, so to summarize, you can see that the environment or the experience will produce this calculating data. The learner is able to determine on the target functions and the representation for that target functions. Then it will be applying the learning algorithm Li in order to work on the training data TD and then producing the model. And then that become the knowledge that the learner produced. And this knowledge will be sent to the performance elements that the agents is using in order to make the decision. You probably still remember our definitions of intelligent agents. The intelligent agents will take the input being the observations from the sensors fit it into the performance element. And within the performance elements, the agents have certain kind of knowledge, in this case the knowledge being learned from the training data. And then it's in the form of this model S hat. And then it will be used within the performance element to choose the next action to perform. Send that next action to perform. Now this upper part is the learning using the training data. But we haven't been able to say how good or how bad this model F hat is. In order to determine how good or how bad this model S hat is, then we typically also need to test it as well. And so we have the testing data as part of the data that the agents have available. And then using this testing data, the agents will be able to evaluate these models at heart in order to determine how good or how bad this model is. And so this is the process of designing a learning system. And so this training versus testing now bring us to another issues. So let's say that if we actually have a set of data, okay? Now, the best way in order to obtain the training and the test data is that they are independently drawn from the same overall distribution of data. And we call this IID, which is independent and identically distributed. Okay? So why is it why is important that they are identically distributed? Because they need to be drawn from the same overall distribution of data. Just imagine that you get the training data. Let's say that about the customer and all of them with the training data are the young customer. So the age of those customer are between 20 and 35. But then when you test your models, then the test data come from another distribution of the data, which is the old customer. So the test data is all about the customer whose age between 50 to 70. Now, you actually learn your models from the young customer, the one whose age is between 20 and 35. But then when you test, when you evaluate your models, and then you evaluate against the data from the old customer whose age between 55 and 17, then clearly your model may not be very accurate because the spending behavior of the young people are very different from the spending behavior of the old people. And so that means that you are running into the risk of getting a draw models. And then it's not wrong because the algorithm is not good, the representation is not good. It's just because you actually train it with a very biased set of data and then test it against node bias sets of data. So that's the reason why it's important that the training, the test examples draw from the same overall distribution of data. So they need to be identically distributed and they are independent. Okay? And so if the example are not independent, then you probably require some kind of collective classification. So that means that you try to do the classifications for a set of data that there are some correlation between the data pieces. And that means that when the test show that there is some high accuracy, it may mean because of the correlations within the data. And if the test distribution is different, then the technique that use typically is called transfer learning because you want to transfer the learning about the young populations to the populations of the old people. All right? And so after we learned about designing the learning system and then testing the learning system to evaluate its performance, now we are at the point that we are able to talk about machine learning in general. So currently there are many machine learning algorithms. There exists tens of thousands of such algorithms. Not only that there are many such algorithms, but new algorithms are being invented as well. So hundreds of new algorithms are being invented every year. And so you can see that there are a lot of machine learning algorithms, however, all of them required to contain these three components representations. So you already remember that we talk about when we design a learning systems. We really need to choose the target function that we want to learn and then we need to choose the right representation for that target function. So choosing the target function to learn and choosing the right representation for that function is part of the representations in every machine learning algorithm. Okay? The second one is that when we actually learn a target functions then we really need to find the right models among many models. So for instance, if we want to have these linear regressions then clearly we would like to have the parameters of the linear regressions to be optimal so that the light will approximate the training data as much as possible. So that means that that mass approximations of the linear line to the training data will be the optimization that you need to perform. And finally you already seen that I talk about. After we produce a model from the training data, we would like to assess, we would like to evaluate how good or how bad that model is against some test data. And so that step of evaluations against the test data is another critical part that you need to perform against your model. And so every machine learning algorithm need to specify these three components to allow the machine learning practitioner to represent the models, choose the best model with optimization and then after that evaluate to see how good or how bad that model can be used in the future data points. So there are several other important but they are a bit advanced and so I would like to talk about them in a separate video because it can be confusing if I talk about all of them in this short video, introductory video. And so we will have the concept of model tuning, okay, so even after we choose a target functions and the right representations, now we put that target function through training in order to find the optimal model. But then that model may not be the good one because of a number of reasons. For instance, you may overfit that model into the training data or maybe because of the training data is not sufficient and so it become underfit as well. And so that means that you will need to tune the model to ensure that it is avoiding both overfitting and underfitting. And the technique for model tuning typically look into some technique like regularizations and then you may have to consider the hyperparameters that the machine learning algorithm use when training the model, when learning the models. And so we need to deal with these hyperparameters through another process called meta reasoning in order to choose the right hyperparameters. And that process of choosing the right hyperparameters may require another step covalidations. And then this is again easily confused with the evaluation with test data. So a lot of the people could not really see the difference between validation and test. Okay? And so all this concept is a bit advanced and I'm going to cover them in a separate video. So to get back to the summary of the task that a machine learner machine learning practitioners need to do so this is the general process. The first is that the machine learning practitioner need to understand the domain that the machine learning system need to be developed. The machine learning practitioner needs to be aware of all the prior knowledge that can assist in order to view the optimal the best machine learning models and what is the goals that it need to achieve with this machine learning model. So after it has a clear understanding of this domain, including the prior knowledge, including the objectives of this machine learning system, then the next step is to obtain the data, integrate the data, clean the data, select the right data, preprocess the data, dealing with the data that available in the best possible way so that it can produce the best possible machine learning model at the end of the process, after it has the right data, then now we try to learn a model. Okay, so this the machine learners task, right? So that's involved with choosing the target functions, choosing the representations of data functions, then training the model for that particular selections of target functions and then validate that model, trying to make sure that the model has been tuned to avoid over 15, under 15 and so on and so forth. After that it has a model that this process produced. Okay, so all of this step allow you to produce a model so that is the resulting model. Now the machine learner may need to interpret these results and then say that based on the validations outcomes, it seemed that my model is still a little bit overfitting the training data. Therefore I would like to train some of these hyperparameters and after making that change I probably go back to do this process again and again and again. And then after that I have a model that I'm confident that this the best that I could get from the current data set and aiming to achieve these goals for the machine learning system. Now I'm going to give it to my customer so that they can consolidate these models and then deploy it in their business. And then again the models when it deployed in the real world may actually show that even though it worked quite well in 80% of the time, but 20% of the time it's not work so well. So maybe we have to go back to this process or the requirements from the customer change. So originally they only need to understand the spending behavior, but now they also want to understand what would be the best way to change that spending behavior to ensure that the customer will be more healthy conscious and they are going to buy the product that is healthy and not the unhealthy product. And so we need to go back to the starting point by understand the new goals with the new prior knowledge and repeat this process. So to summarize, learning can be viewed as using either direct or indirect experience. Direct experience is the concept related to reinforcement learning when the system encounter an experience and directly use that in order to change its behavior immediately with every piece of experience. Indirect experience is typically after you produce a large set of data and after that you say that with this last set of data, how can I learn some insight from this data? Now I'm going to try to approximate a chosen target functions in order to get that insight. Once a target function has been chosen and represented using the appropriate representation, then the machine learner will typically perform these approximations via a search process in order to find the optimal models, also called hypothesis. And this hypothesis is being selected amongst many possible hypothesis within a space of hypothesis in order to choose the best one that fit the training data without overfitting and without underfitting. And after you are able to produce that best possible hypothesis within this last space of hypothesis which we typically also call the learn models. Now you are able to evaluate and then trying to deploy it into the business. And clearly because there are many different learning methods and learning algorithms, so we are going to have to deal with many different hypothesis spaces because there will be different representations and then with each representations there may be a particular different search techniques will be used in order to find that optimal models to produce. So this is at the end of these high level introductions into machine learning. But I'm planning to produce another videos to give you some more details about some advanced topics related to machine learning. For instance, I mentioned about model tuning, I mentioned about overfitting, underfitting, I mentioned about regularization. So stay tuned for that video. Thank you for your attention.