Today I will talk about how AI frees people from paperwork in finance. I am Ping Luo, an associate professor in the institute of computing technology. And my general research area is data mining and machine learning. In this talk, Dr. Cao will also join us. Yixuan Cao is an assistant professor in the Institute of Computing Technology, Chinese Academy of Sciences. And his general research area is also data mining and machine learning, with a focus on natural language processing.

Let's begin this talk. First Let me ask you a question. Which position is the most hard and tedious in the financial industry? This is a very famous on my survey in Zhihu. Zhihu is the most popular Q & A platform in China. Do you have the answer? And the answer to this survey is that the number one tedious position in financial industry is investment bankers.

So although financial industry looks like very bright and shiny and the colleagues in this industry often have very good education background, however, there are also grunt works that are laborious and tedious. One of them is document writing and reviewing. The glorious, nice and disclosed prospectuses such as Bond Prospectuses and IPO documents are usually the products of countless all-nighters. We observe that finance industry is document intensive. Practitioners in this industry have to read, search, write, review documents every day. Documents in this industry include invoice and receipt, contracts, bond prospectuses, IPO prospectuses, Annual Report and Quarterly Report. Manually processing these documents is labor-intensive, time-consuming, and error prone.

However, in many areas, AI has helped people or even freed people from tedious work. For example, AI is applied in automatic customer service, credit assessment, and machine translation, so that people can devote to more creative works. Therefore, one question is that can AI free people from paperwork in financial industry. This is the main content we will talk about in this talk.

So this talk will be divided into two parts. In the first part, we will talk about what is Document Intelligence and present some real world applications in this area to show how AI can free you from tedious paperwork in your everyday work. Specifically, we will talk about the applications in the following three aspects, including information gathering, intelligent review, and intelligent writing. These tools may help you find information, review documents and write documents in a more efficient and effective way.

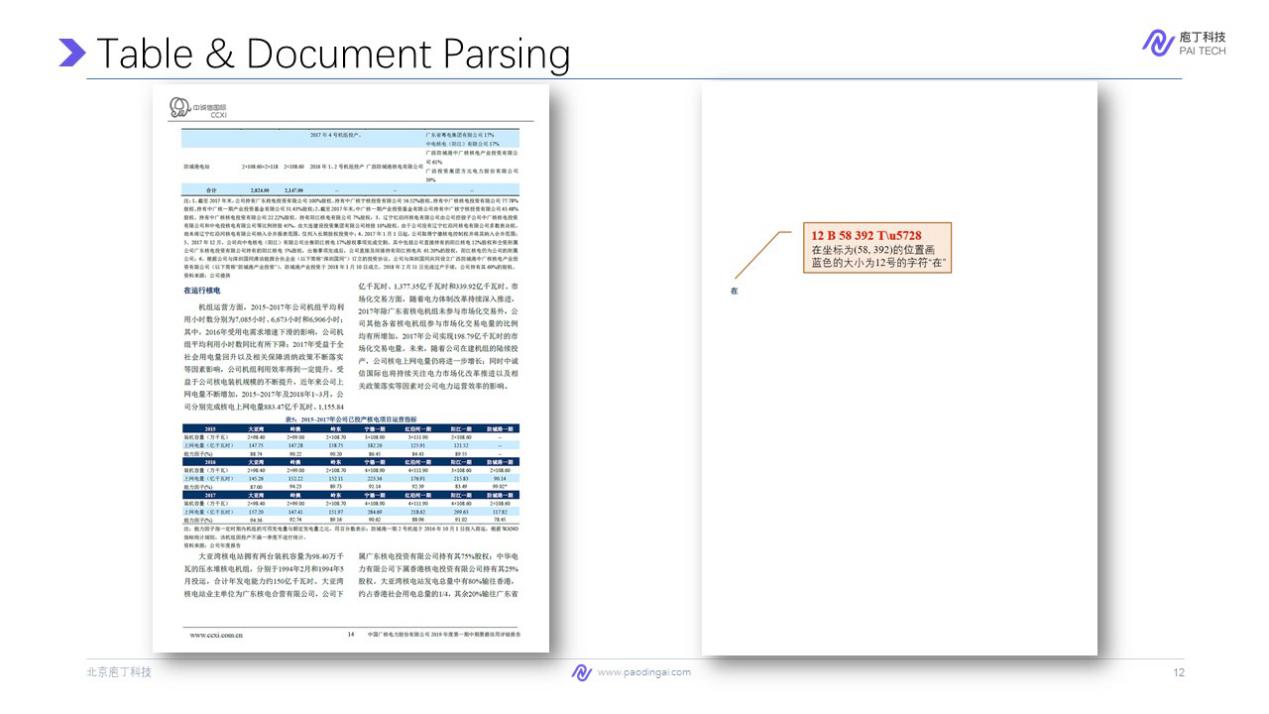
After the first part, Dr. Cao will help to talk about the key technologies behind these document intelligence applications. We will begin with some basic concepts in AI machinery and deep learning. Then we'll give you a simple example to show you how machine can learn from the data. Last we will move to some advanced topics in computer vision and natural language processing. This is the overview of this talk.

Let's move to the first part. First, what is Document Intelligence? In 2019, there hold a workshop on document intelligence. In this workshop, the meeting organizers try to give a clear definition on document intelligence. Document intelligence is collectively referred to as the ability to read, understand, and interpret business documents. Business documents include sales agreements, vendor contracts, no applications, purchase orders, invoices, financial statements, employment agreements, and many more. These documents often reflect complex legal agreements and reference, explicitly or implicitly regulations, case law and standard business practices. Thus they are central to the operation of business.

In this talk, we will introduce the following applications to show the latest progress in the field of Document Intelligence.

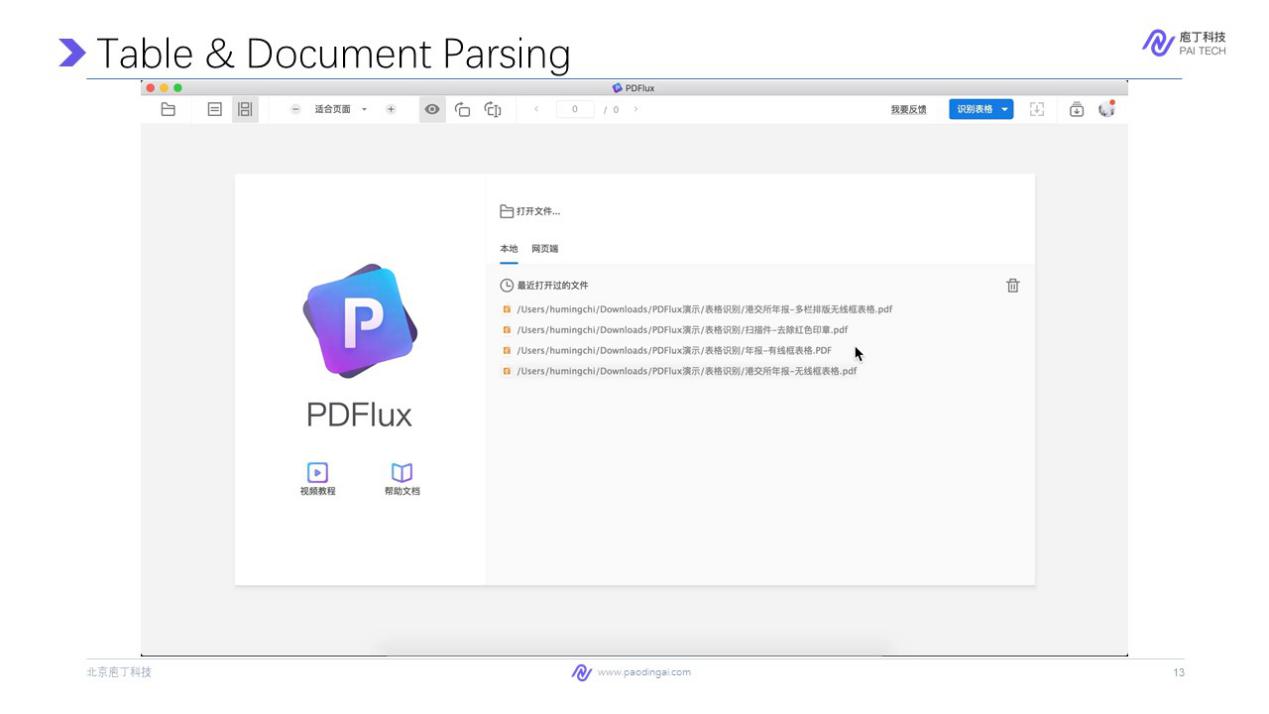
First, let's talk about intelligent information gathering. And in our daily work, correcting data is one of the most important and frequent tasks. However, in many cases, it is not easy. Here we show you an example. I'm copying a table from a PDF file. The result is a set of text, instead of a table, this means the information of the table structure is lost in the PDF.

Another example is that we want to copy the text of a paragraph. But when we copy the text into the word, we find that the text is broken into multiple lines. However, all these breaks are totally unnecessary. So why can't we get correct information we want from PDF files? This is because PDF is designed to ensure it looks the same everywhere, so that we can transfer, display, and print it. But the information of the document structure is lost in PDF.

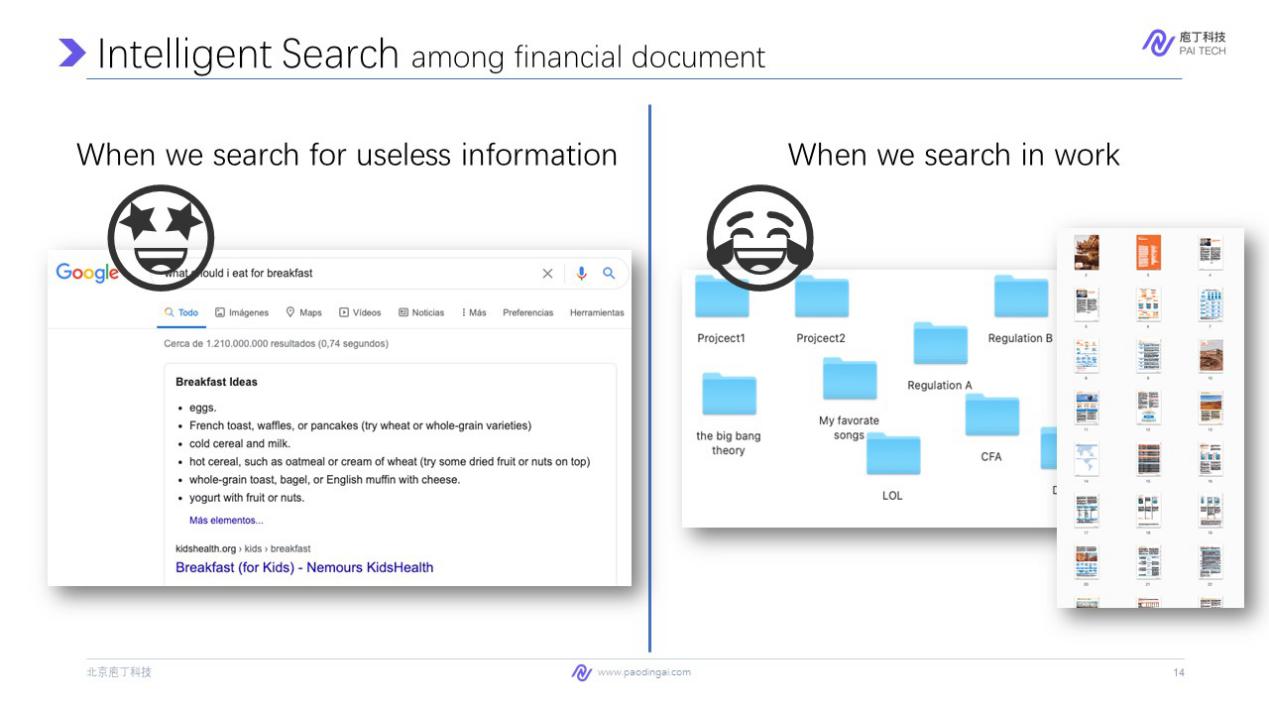


We will show you the details. What is stored inside PDF files? This is a screenshot of a PDF page. When we look into the PDF, it only stores the instructions about how to print each character in the page. For example, this code is used to draw a character in the position (58, 392), in a blue font with size 12. And when we open the PDF, the software or PDF reader follows these instructions to draw each character lines and figures. But in fact, it has no information about lines, about programs or tables.

Unfortunately, most company and financial institutes publish their documents, in PDF files. These files only used for home in reading. However, when we need to conduct data analysis, we have to waste most of time on copying cells out of table, and formatting the paragraphs. This is really tedious.



So now with the help of AI we can easily get the structure table with a click. Specifically, we invent a software of PDFlux. It has two versions for windows or MAC computers. You can download it for free with PDFlux. You can copy all the tables inside the PDF files to Word, Excel files to conduct data analysis easily. We will give you an example. So with this software, you can open a PDF file. Then in this software, you can copy this table directly into a Word files. So AI technology behind it will be introduced in the second part of this lecture.

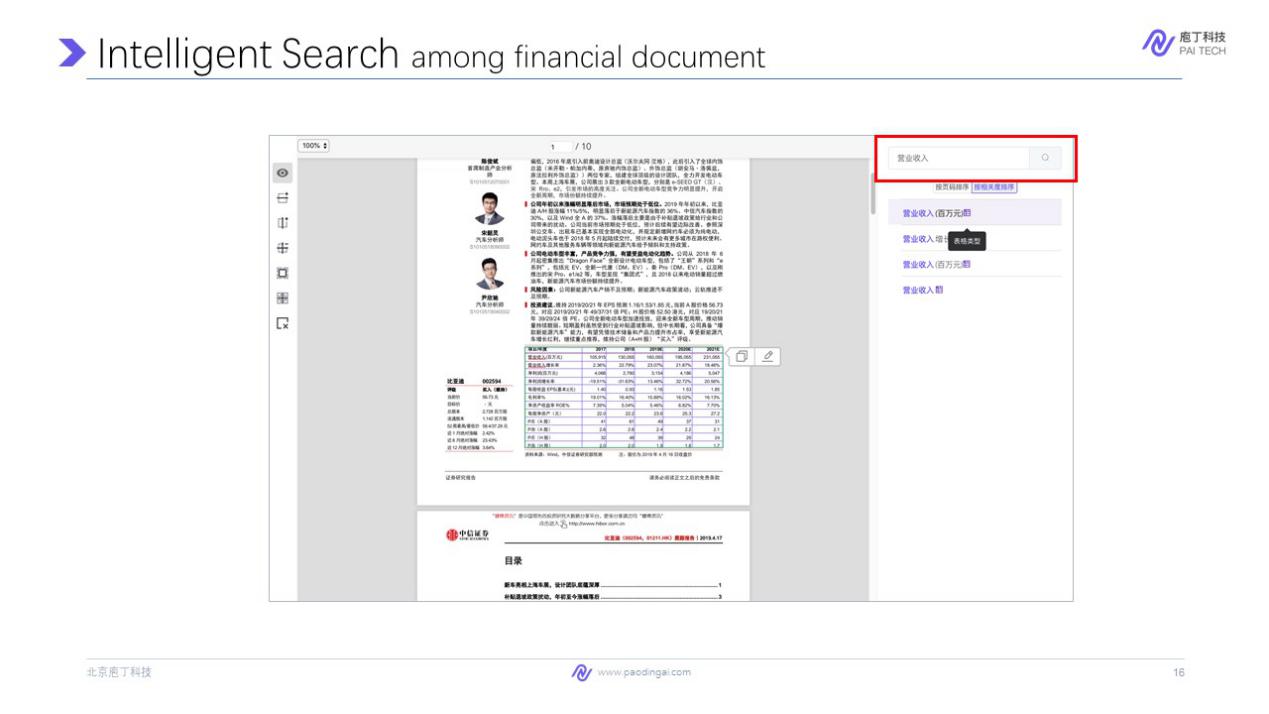


The second application in information gathering is intelligent search. Search engines like Baidu and Google make it easy for us to find information in our daily life. If the question you ask is not very important, like "what should I eat for breakfast"? It will give you lots of useful results. The results are mostly web pages. But during everyday work, especially in finance industry, most of the data we want stores in documents. They usually are stored in our computer loosely.

And the way we search for information in work involves two steps. 1. search for documents from badly structured folders and 2. then search inside the PDF documents, which usually have hundreds of pages. This is time consuming. Is there a better way? Why can't we have a search engine for work?

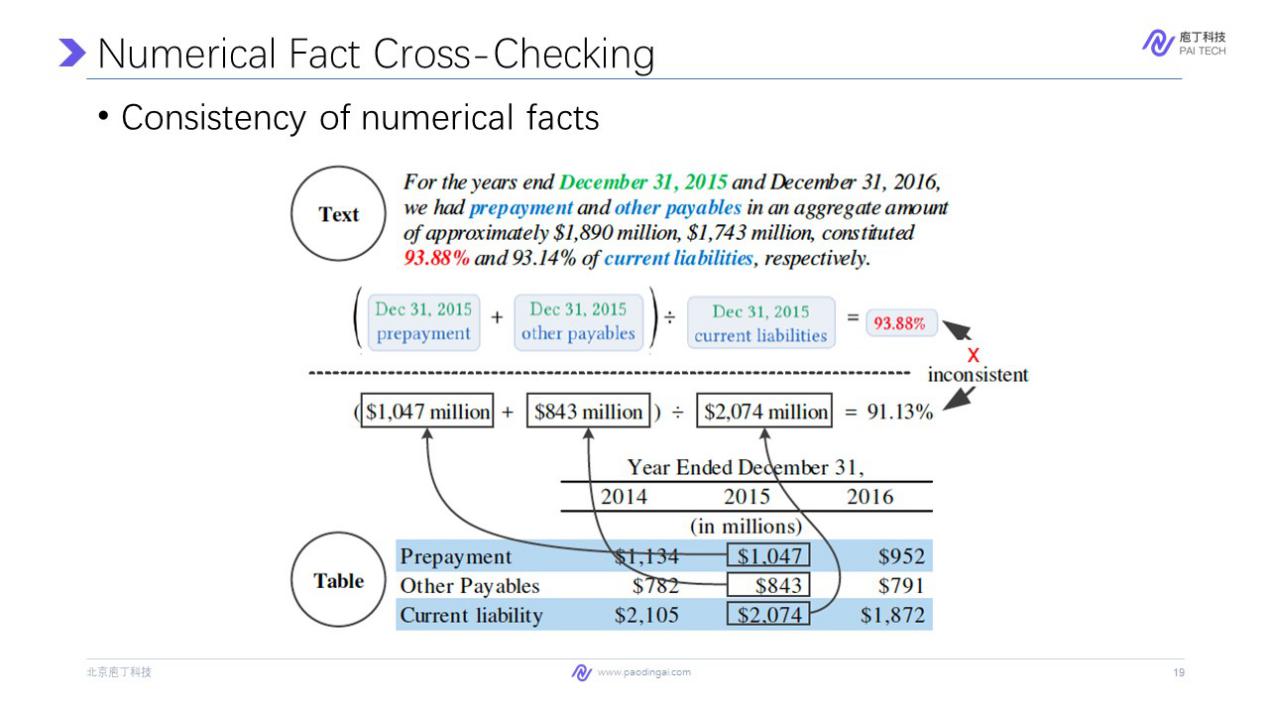
The most difficulty lies in document parsing. As we introduced in previous slides, comparing to web pages, PDFs are hard to parse. But this has been tackled by the PDFlux tool. So, we build an intelligent search engine for work upon a huge amount of professional documents.

Here we show a demo for you. For example, we search for the profit prediction of BYD. BYD is the company name from CITIC Securities. The result page shows a list of PDF files for you, each with some snippets. Since the profit prediction is usually a table, we click this snippet. And then the search engine will directly show you the table inside the document.



So the two-step search: search for document and search inside the document is combined into one. Moreover, this search engine also integrate the PDFlux function. You can copy the table with a click into a Word or Excel file. Such an integrated search engine is also powered by AI technologies. Then you can also search inside a document. Despite the online published documents, you can also search your personal document if you upload your own document to the locally deployed search engine. And in this way, the search engine becomes your personal knowledge base.

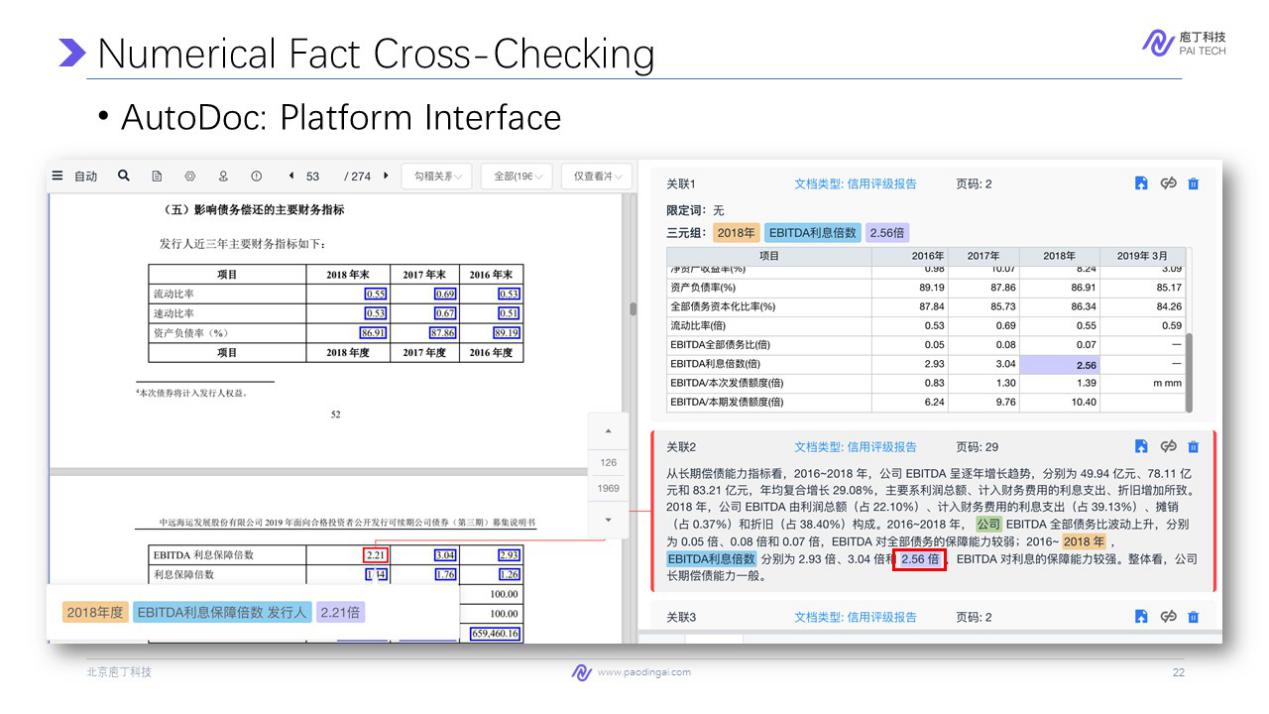
Next we talk about intelligent review from two aspects, including numerical cross-checking and Compliance Assessment against Listing Rules. First, what is numerical factor cross checking? Every published financial document, include annual report, IPO prospectus, and bond prospectus, has to comply with some common standards. One of them is the consistency of numerical facts among a document, as well as among several related documents.



For example, this sentence in this page: for a year and December 31, 2015 and December 31, 2016, we had prepayments and other payables in an aggregate amount of approximately this number and this number constituted this percentage and this percentage of current liabilities, respectively. It describes some summations and ratios of among financial indicators. One of the formula is show shown here. In the same document, there is the table disclosing the information of the basic indicators like prepayment. Now we have two representations of the same data, in text in the paragraph and in table. So we need to make sure that these two indicators are consistent.

A very famous news about this cross-checking is that in 2011, Goldman Sacks made a typo in its disclosure documents, which cost it $45 million. Since published financial documents are the most important way for investors to understand a company, even immaterial errors in the document could damage the reputation of companies. So the traditional way for cross-checking is to do it manually. In finance, there is a special job called authorized reading. On average, an experienced employee has to take a week to cross check a 500 page document. It is really time consuming and laborious.

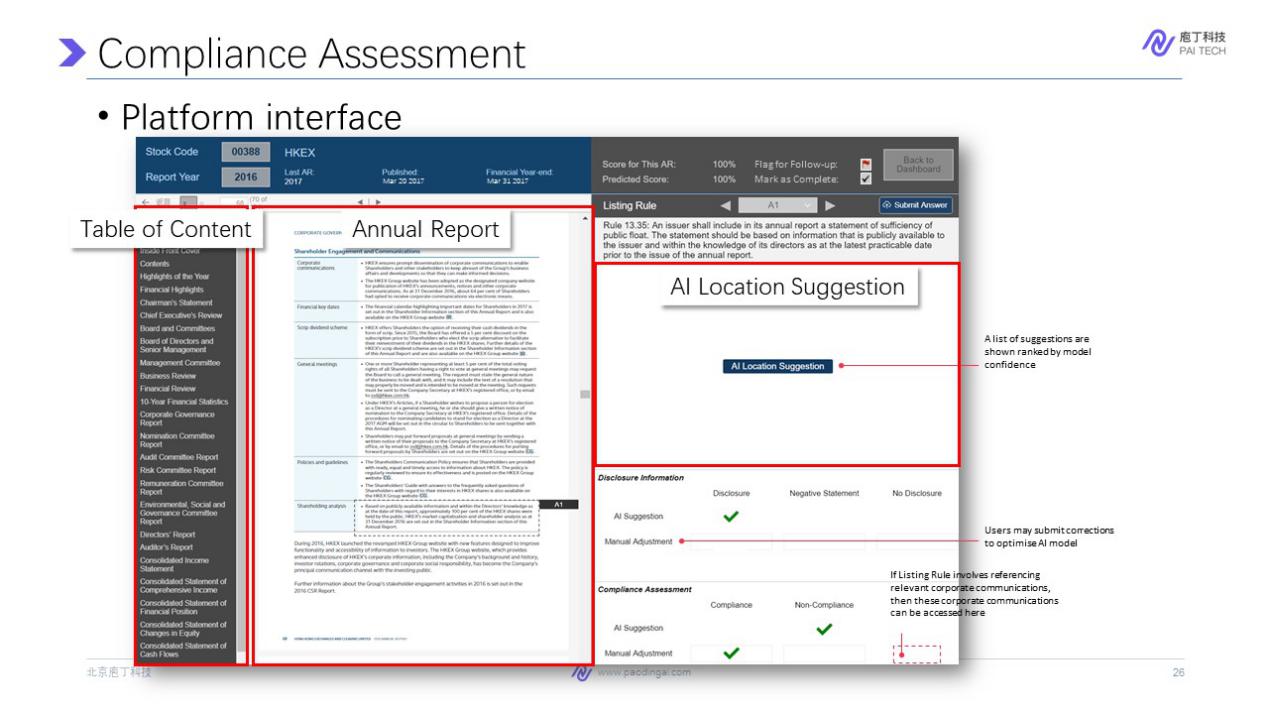
And more importantly, manual checking is error prone. This is obvious. Imagine you need to check a document of 500 pages, thousands of numbers and formulas in about one week, every human being will get tired. Then mistakes happen. We run our system on about a thousand of officially published Chinese prospects, found that nearly 70% of them have data inconsistency errors. With the help of AI, the system we built can do the same job in less than one hour. It means the cross checking work is done when you have a coffee break.



This system is called AutoDoc. Here is the interface of this system. The interface highlight every number that has been cross-checked. The inconsistencies are highlighted by red boxes. We also shows the semantics of the number, including the value of indicator, time and company name. It requires a clear understanding of the table and the paragraph for AI to correctly extract such semantic meaning of each value as the financial document is very long and complex.

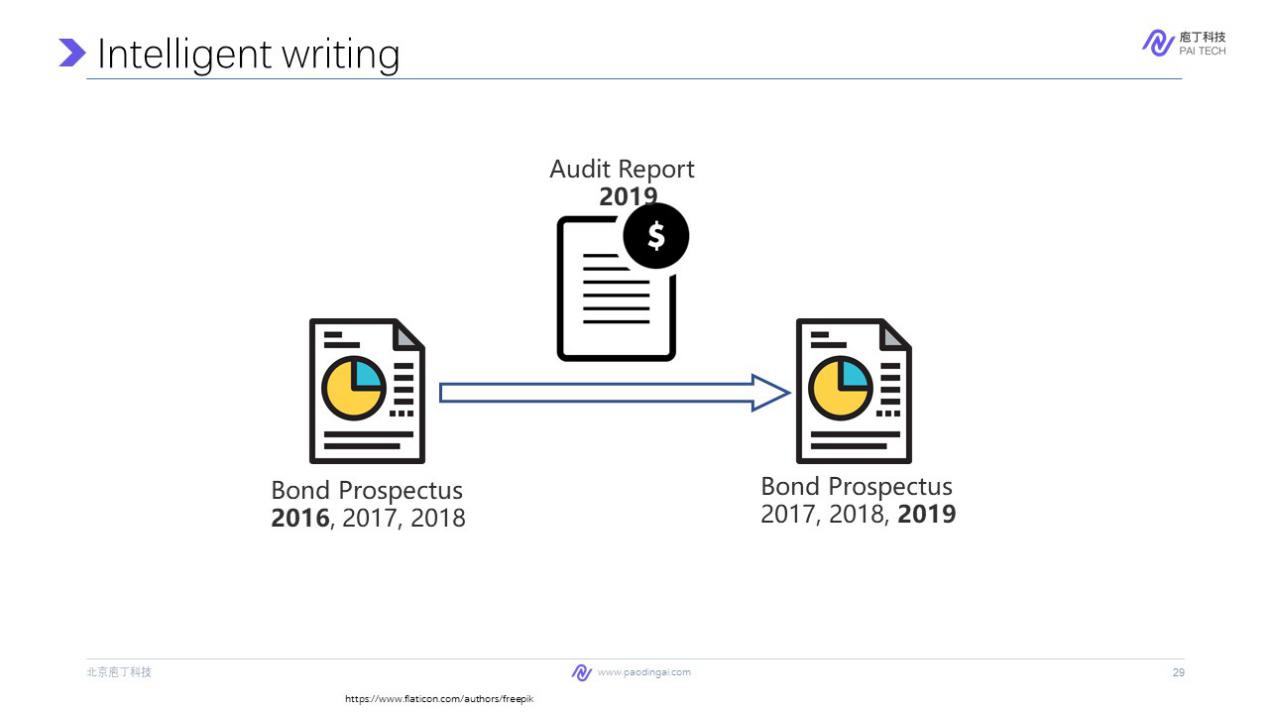
Beside the numerical consistency, what information to disclose and how to disclose them also has strict regulatory policies. Here, we introduce the compliance assessment of Hong Kong exchange. The IPO market in Hong Kong is one of the most active in the world. The number of companies listed on the Stock Exchange of Hong Kong (issuers) has more than tripled over the past 20 years, reaching 2,507 as at the end of July 2020. Each issuer must publish an annual report each year, which has 173 pages on average.

Hong Kong Exchanges and Clearing Limited (HKEX) pays close attention to the content of Annual Reports, monitoring them to check, among other things, that the issuers are disclosing all the relevant information that the Listing Rules of the Stock Exchange require of them. The number of listing rules is more than one hundred. For example, this listing rule asks to disclose the information about the five highest paid individuals, as well as how to disclose. To assess the compliance of this rule, a reviewer goes through hundreds of pages to find a relevant table, and check the details inside the table.

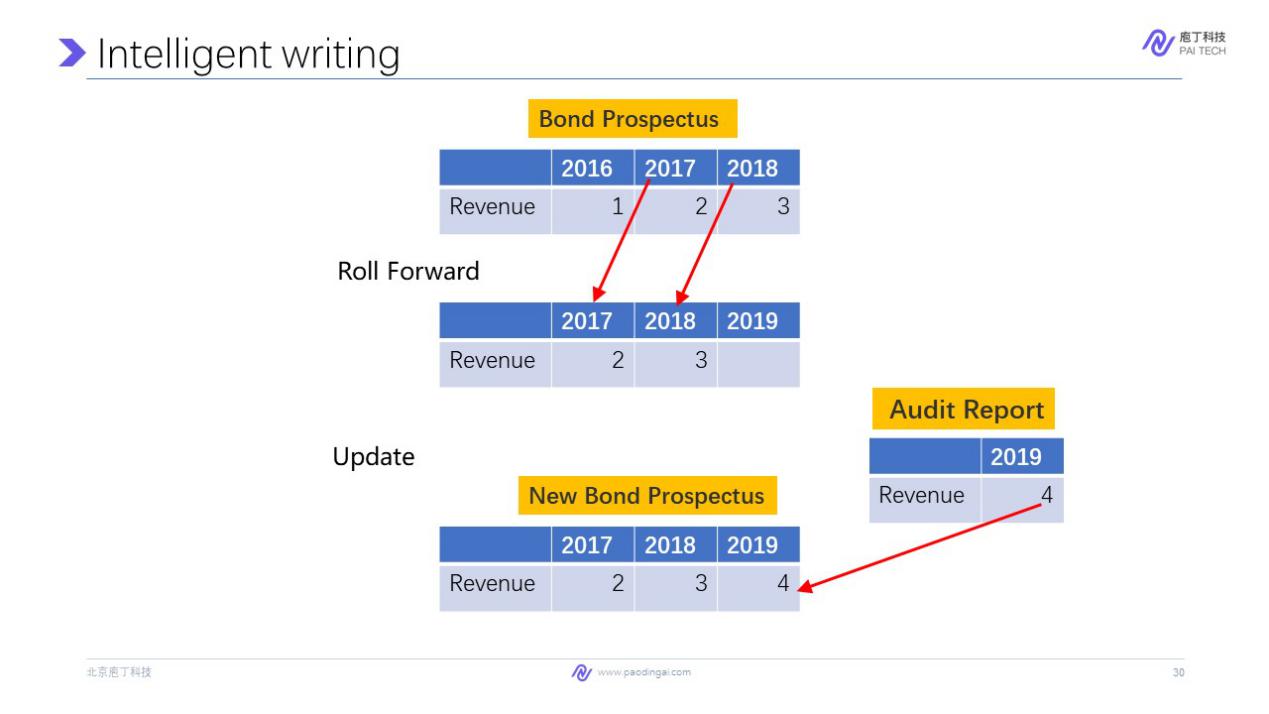


Here we find this table. And the bond of the salary is compliant to the rule. So we can see this document is compliant on this rule. Go through hundreds of rules, is really time consuming for human being. So with an assistant system, is developed based on AI. When reviewing another report for each listing rule, AI will automatically suggests several passages that are relevant to the rule, so that a reviewer doesn't have to search for them throughout hundreds of pages.

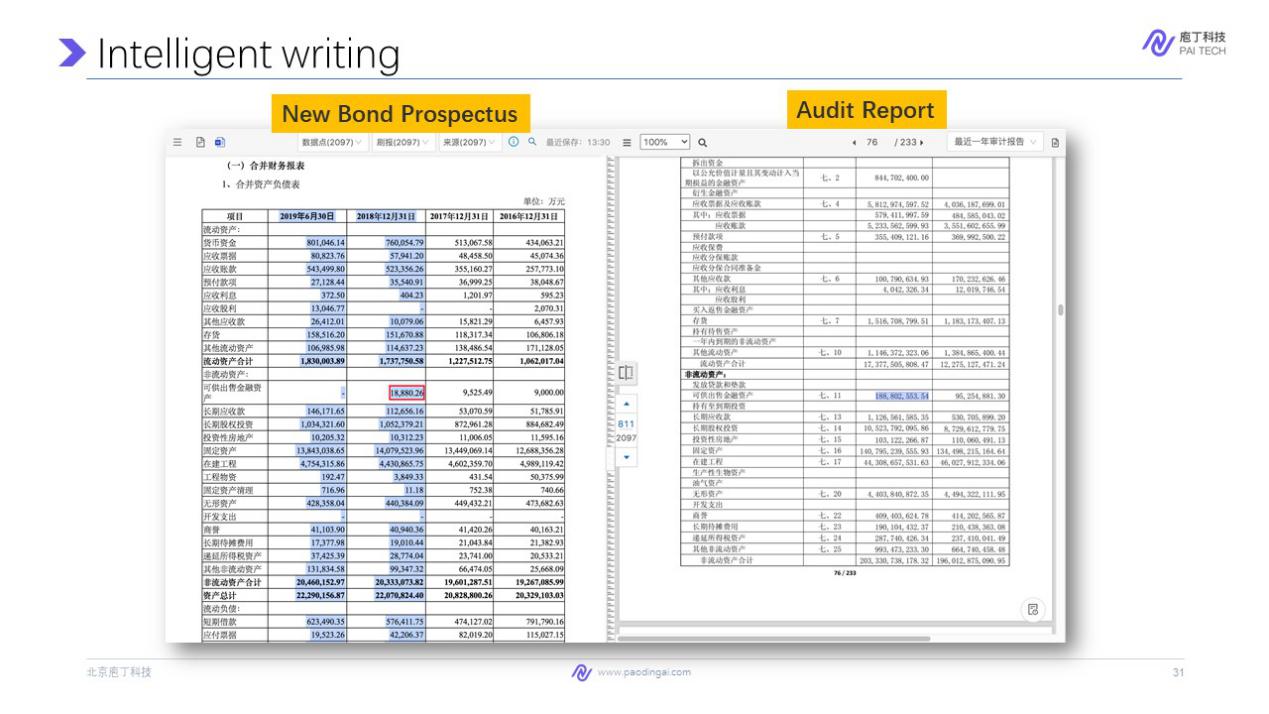
Meanwhile, the platform will automatically suggest whether the disclosure is compliant or not. Specialists in Hong Kong exchange have tasted the platform on new annual reports. And the result shows that 80% of time is saved compared to a manual assessment. This is all about the application of intelligent document reviewing.



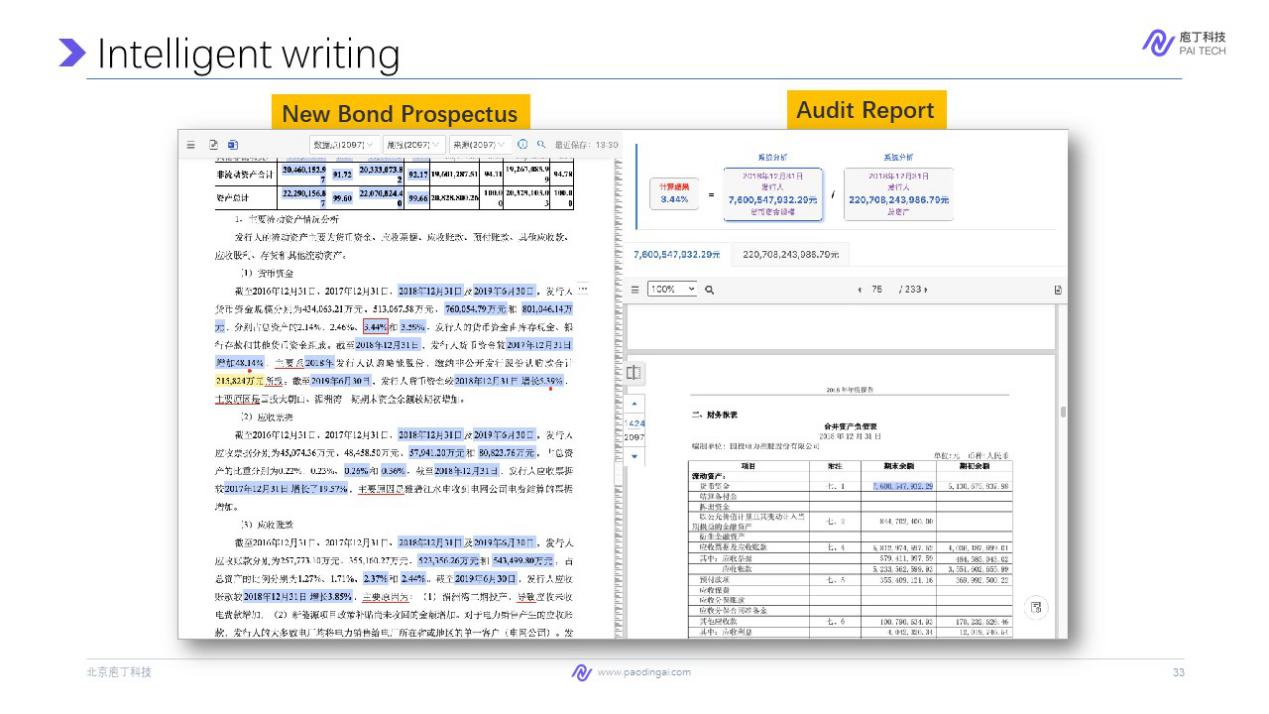
Finally, let's talk about intelligent writing. In finance industry, when we write a document, for example a bond prospectus, or a audit report, in most cases, we don't have to write it from scratch. For example, we have a bond prospectus whose reporting period is from 2016 to 2018. We update it with an audit report of 2019 to get a new bond prospectus for spectators.



Specifically, it involves two steps. Rolling forward, move the time period of the bond prospectus forward. On here, the value of the latest time period is empty. Then the update step fill this emptiness with the new values from the auditory report.



Here is a system interface, the updated data point, include time and values are highlighted, you can click on the value (for example the one in red box) to see its original place in the audit report, and also values. And the sentences in the paradox can also be updated.



It can even update values that are result of computation. Like the ratio of monetary capital of total assets, which never appear in any documents, and can only be computed by dividing the two values. This requires a deep understanding of the meaning of value in the sentence. On average, one bond prospectus has around 1000 of data points to be updated. And tested by professionals, with the help of AI, the time to update a document reduced from 3 hours to half an hour, that is 83% of time is saved.

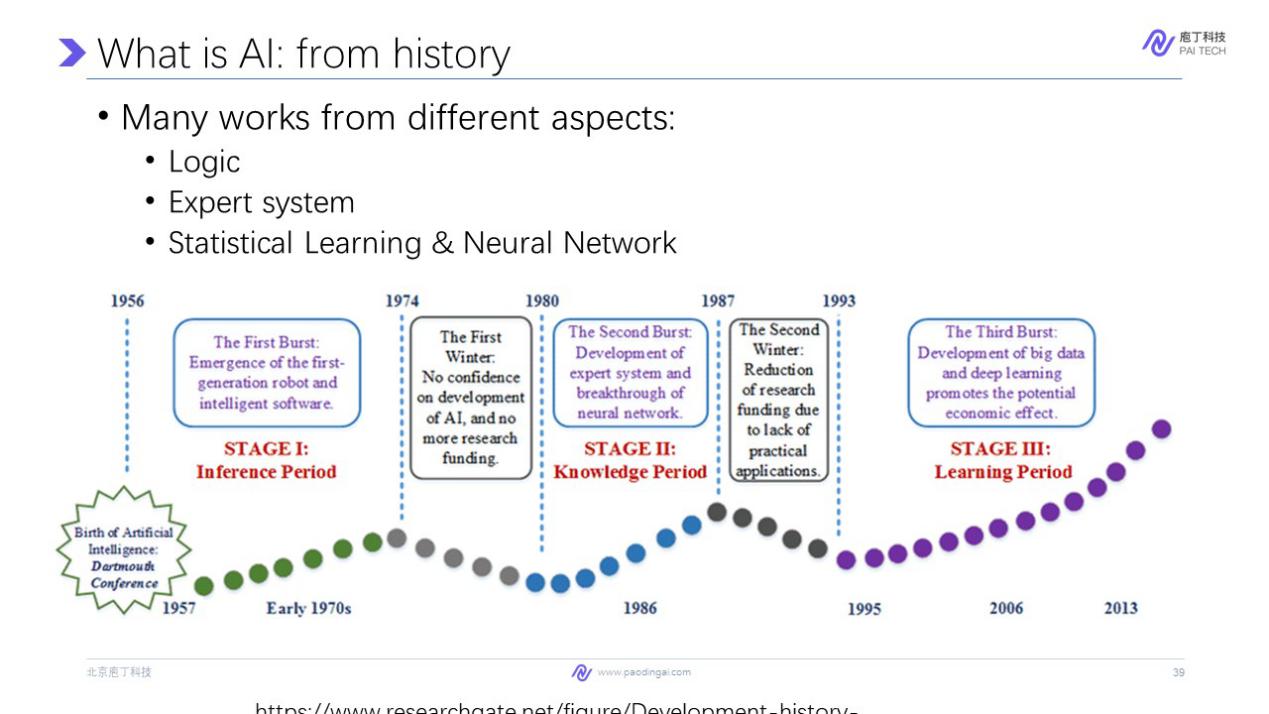
This is the application of intelligent document writing. Faced with so many applications, you might become interested in what is the technology behind them. And next, we move into the second part. And this is more technical and show you the technologies behind. Let's welcome Dr. Cao.

Nice to meet you guys. I am Yixuan Cao from the Institute of Computing Technology. I'm very happy to be here with you. And in this part, I want to give you an intuitive idea about what is AI, machine learning, and deep learning. I hope that with a simple example, you will realize that AI is not a mystery, but just some math functions, combined with a lot of data. Then I will introduce some technologies behind the applications introduced in the previous part, including how do we achieve document intelligence with computer vision and natural language processing.

So, first, what is AI? Most people heard about AI from news and media. like AlphaGo defeats top human Go player. Google launched a new round of auto drive test. You probably also have experienced AI when you wake up your Siri, or chat with customer service chatbots in Taobao. This is what we have heard and experienced from our view. But from academic view, AI is a very broad concept. It is more like a goal rather than a specific technology.

In the book, Artificial Intelligence: A Modern Approach. In a sense, AI describes machines (or computers) that mimic "cognitive" functions that humans associate with the human mind, such as "learning" and "problem solving". So it says AI is some machine act like human mind. But there is another interesting remark from Tesler's who said, AI is whatever hasn't been done yet. This is an interesting point.

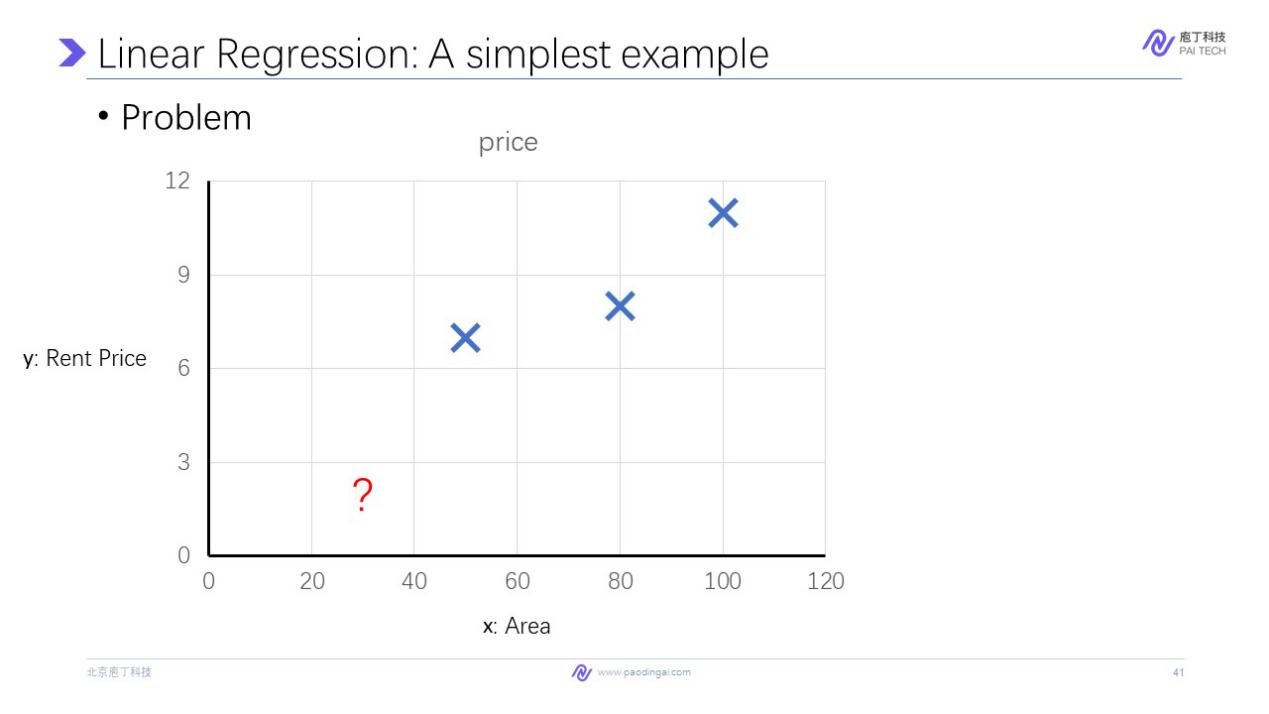
As we all can remember that when we use the calculator, it also looks like a human mind, that's it can enter numbers and can subtract numbers. But since we have already been familiar with that, we don't think it's something called artificial intelligence. But it actually can solve problem. And this kind of ability is quite related to human mind.



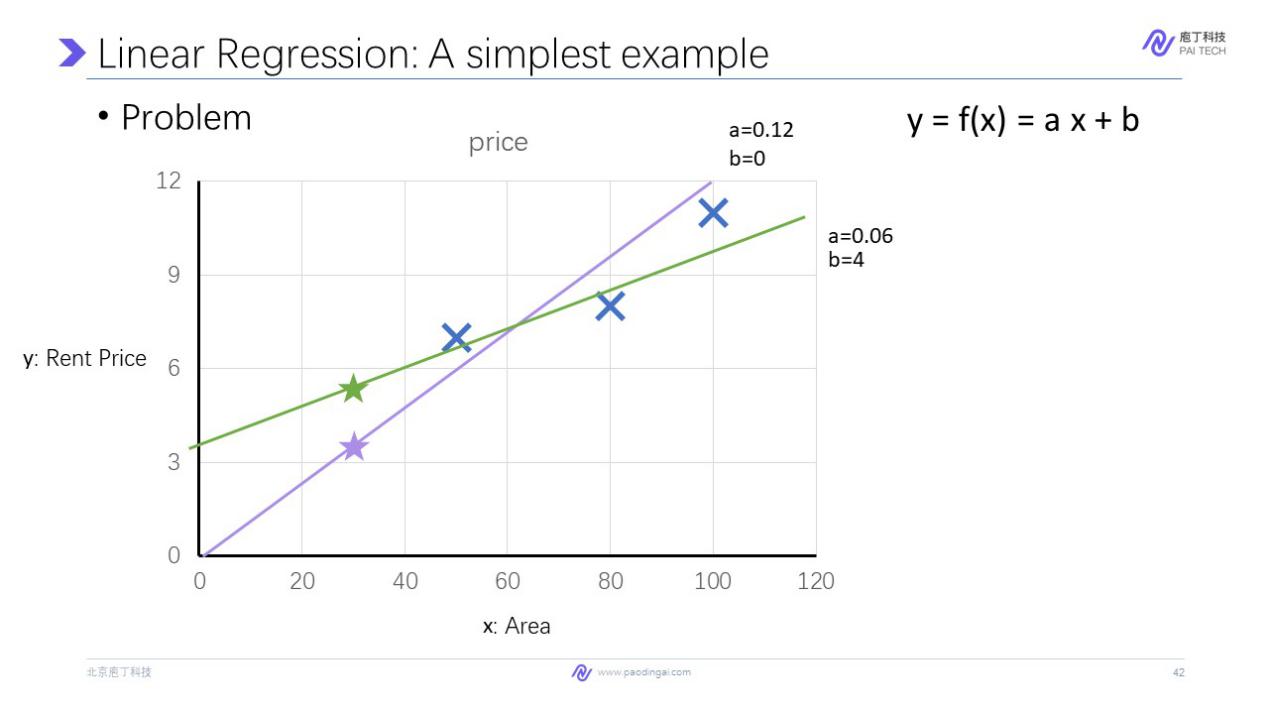
Although AI is a very hot topic currently, it experienced several ups and downs. Back in 1957, AI is born in Dartmouth Workshop in America. The first wave of artificial intelligence focused on inference and logic. But soon, the first winter came. Later in the 1980s came the second burst, which is characterized by expert system, and the start of neural network Then, the second winter came, due to the lack of practical applications and thus the reduction of research funding. Finally, the third weave is about learning, start from statistical learning, like SVM, and currently the deep learning.

So whether will there be the 3rd wave of winter? From my point of view, as AI is applied in real world applications, the technique is kind of mature, although the heat will fade, it will not enter winter in a short period.

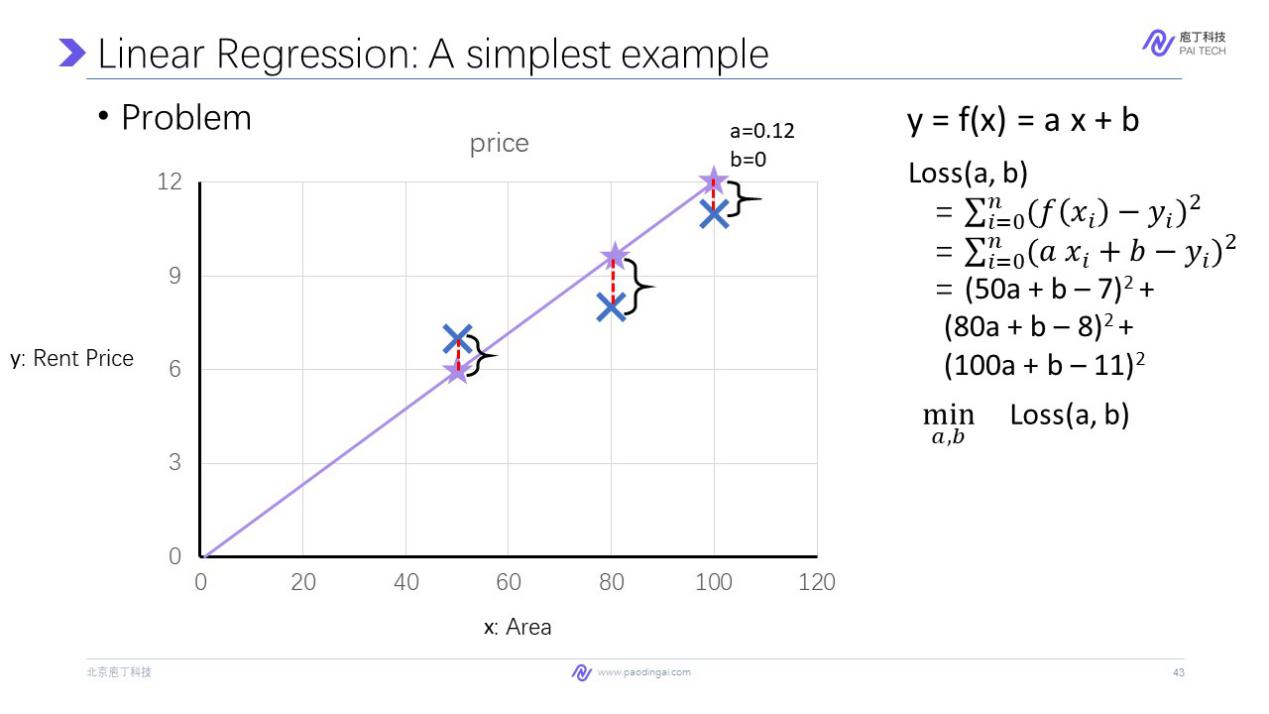
Currently, when we talk about AI, we probably are talking about the last burst, which is machine learning. So what is machine learning? This is a technical definition of machine learning. A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E. This is kind of obscure. So we will give you a very simple example to understand this.



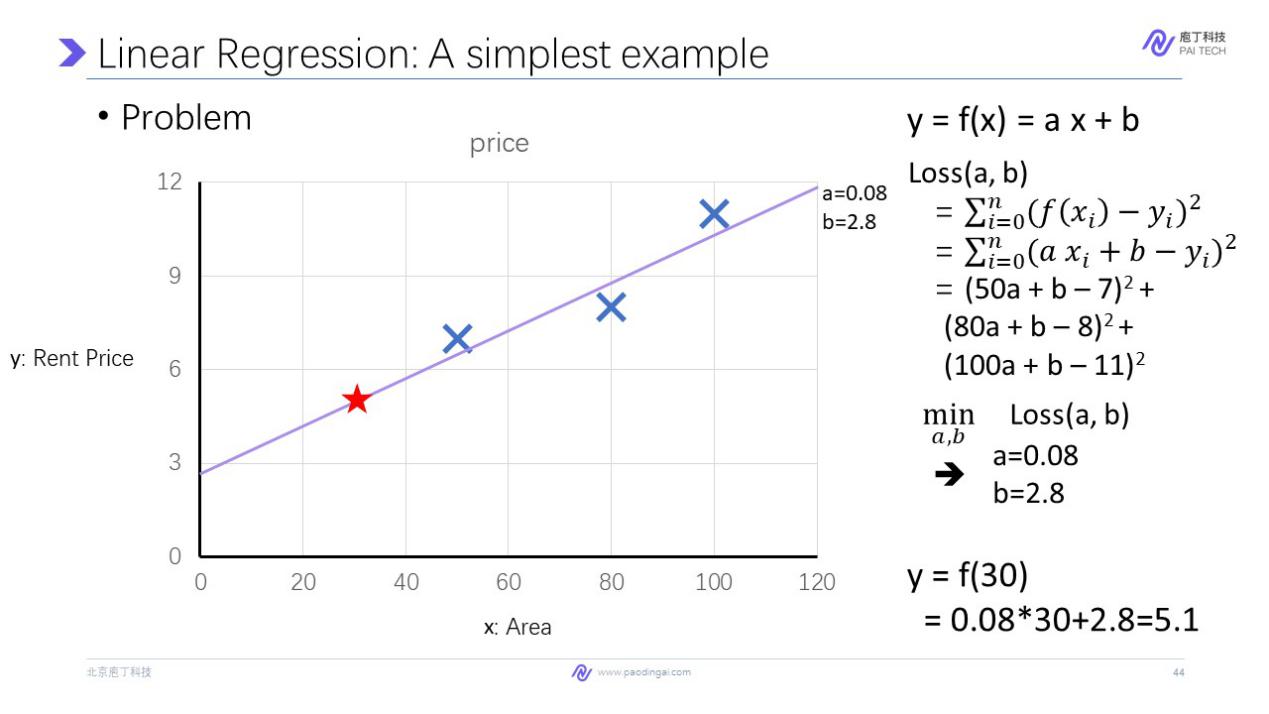
Suppose we are going to rent a room. And we get the prices of three rooms with different areas. Can we predict what the price will be for a room of 30 square meters? Can a machine do it? Actually, this is a typical regression task that you may have learn in high school. We can assume that y(rent price) is a linear function of x: y=ax+b, we have many choices on a and b. For example, here a equals 0.12, and b equals 0. And with this choice of a and b, the prediction of the function when the area is 30, is about 3000 Yuan.



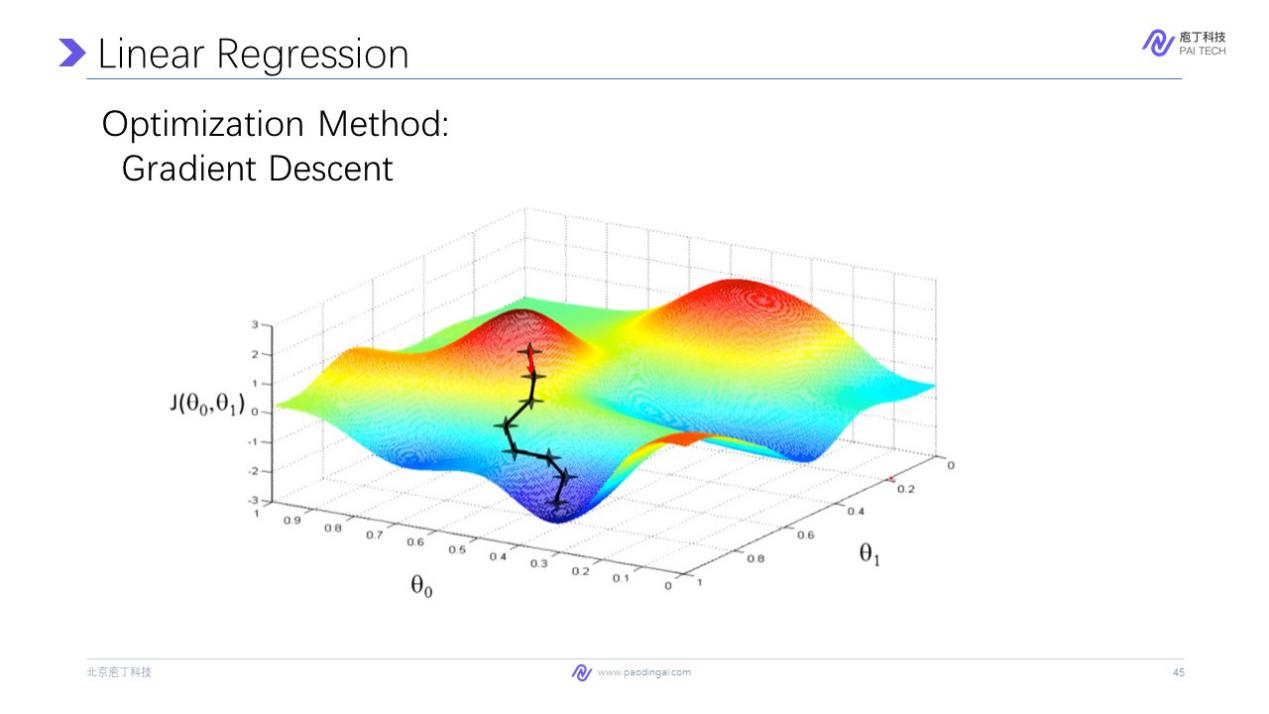
There is another choice of a and b where a is equal to 0.06 and b is equal to 4. And with this a choice of a and b, the function will predict that the price will be around 6000 yuan.



But which one is the best? We need an indicator to measure which function is the best at predicting the price. We measure the error on each point as the square of the difference between prediction and real data. For example, with this choice of a and b when the area is 15, here is the difference between the prediction and real data. We take the square of this loss. So the overall error on this function made on the three data is the sum on 0. We call this loss function. Here we look at the loss function. It's sums all that point from zero to three, the square of the prediction and the real data. We can see that the loss function is actually a function of the model parameters a and b, and the data points like x1 and y1 are the coefficients of the function.



Minimizing the loss function, by changing a and b, we can get the best prediction. Here we minimize the loss function by changing a and b and we find the best choice of a and b that minimize a loss function in this one, a equals 0.08, and b is 2.8. So this is the best function we can find. It predicts that the area of the room is 30.



We minimize this loss function, but we didn't mention how to do it. In most cases, in machine learning, we optimize or minimize loss function by gradient descent. So what is gradient descent? We can imagine that the loss function that is parameterized by a and b is a mountain like this, where the combination of a and b is the x y location of the mountain.

And the cost is the height of the mountain. So we can start from a random choice of the parameters. So we start from anywhere in the mountain. We compute the gradient of the loss function. Remember that the gradient of a data point in a function shows us the direction where the function decent the fastest. So we can take a step along the gradient direction and get a set of new production. Then we can compute the gradient descent at this point. Take a step forward to this direction and this progress repeat here. And until we reach the bottom of the valley, where the gradient is close to zero, we can stop.

And the parameter here in this position is the best choice which minimizes the loss function. This example shows us how to make a machine learning algorithm that can predict on unseen inputs. We summarize it and give you the basic elements of machine learning. First, we need a hypothesis space. That is how the prediction looks like. In this example, we choose f to be a linear function with two parameters, a and b.

Second, we need a data set to learn. Here we have this data point in this data set, is consist of x and y at where in this case, x is the area of the rule. And y is the price of the rule. Here we have three data points in this data set.

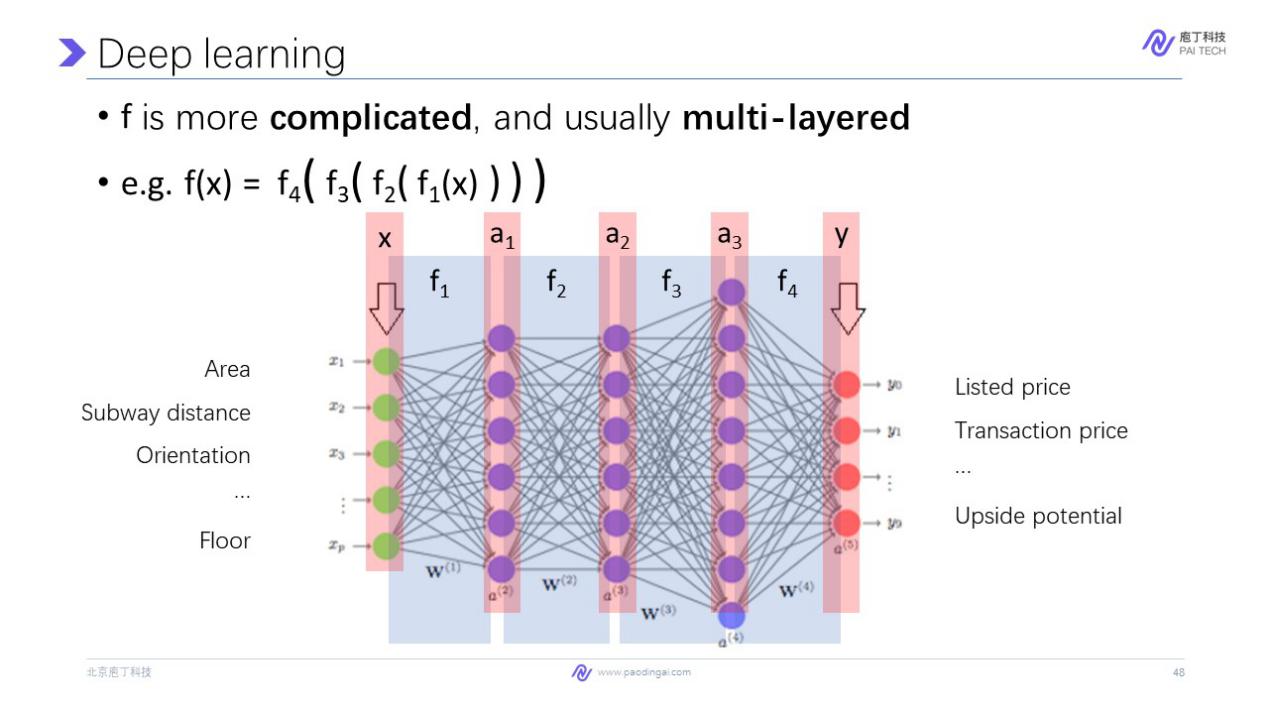
And third, we need to set up a loss function to minimize, which is the goal we want to achieve, that is to minimize the difference between the prediction and the labelled data. Then we optimize a function that is to move the parameters towards the best function we can find. And in this case, we find the best choice of parameters a and b by gradient descent.

Finally, we get the best model parameters, we can predict on unseen samples. The example above is the simplest case we can find. And we may extend it from many different aspects so that it can be used in real cases. First, the input can be multi-dimensional. That is, we can input a vector, always say that function is a multi-variable function.

For example, in the task, we introduced before, we can input the area of the rule as well as distance to the subway station, which is the price of the room. And the input will be a vector like (50, 1). And in reality, in most applications, the input to the function is usually very large. usually is bigger than 100. And sometimes it can reach like tens of thousands of dimensions. Then we can also see that the output can be multi-dimensional. Or we can see f function is a vector-valued function like this. We also take the example above that. We can predict not only the transition price, we can also predict the listed price, but the transaction prices really happened during the transaction.

Moreover, f is not limited to be linear. This is related to the hypothesis space, as we introduced in the last slide. For example, if you want to predict this task, the input is the diameter of a pizza. The output is its price. We know that the prices usually linearly related to the area of the size of a pizza. So f would better be a quadratic function that will perform better than a linear function. Of course, a function is not limited to linear, a quadratic, it can be any kind of function. We need any kind of function. We need to model the relation between the input and output.

So we have introduced the the basic element of machine learning, as well as how we can extend it and unified that machine learning is all about math, vector and functions.



So besides machine learning, we might have also heard deep learning from media. And you may hearabout deep learning more frequently than machine learning. Indeed, deep learning is a special kind of machine learning, whose hypothesis space is more complicated and usually multi-layered. That is to say, f function is a compound function of many sub functions, which in graph looks like different layers here.

Here we visualize this f function where the input is a vector, and the output is also a vector. The f function here is very complicated, and that can be partitioned into several layers of functions, f 1 up to f 3 and f 4. Thiscorresponds to these functions in this formula. And unlike the function we introduced before, which directly convert a vector of input into the output. The functions in deep learning usually have multiple intermediate representation of the data. Here this function converts the input x, a vector into a 1, a 2, a 3, 3 different intermediate vectors.

And finally, predict the output based on a3 We usually say that a 1, a 2, a 3 is the representation of the input at different levels. So machine learning is also related or referred to as representation learning.

And the characteristics of deep learning are from two aspects. First, the number of parameters in a deep learning model is usually very large. Second, the training data needed to train up deep learning model is also very large. For example, the currently best model on text generation, which is called GPT-3, has 175 billion parameters. And it is trained on 45 terabytes of data, which is really affordable for personal computer.

So in summary, AI is a very broad term, research on AI started from a half century ago and has broad research concepts. Machine learning is the sub-field of AI that studies how machine can learn from data. And deep learning is a current hot research direction of machine learning, whose f function is usually very complicated and multi-layered. **Now, I hope you have a feeling that deep learning, or we usually refer it as AI, is about learn a function from data.**

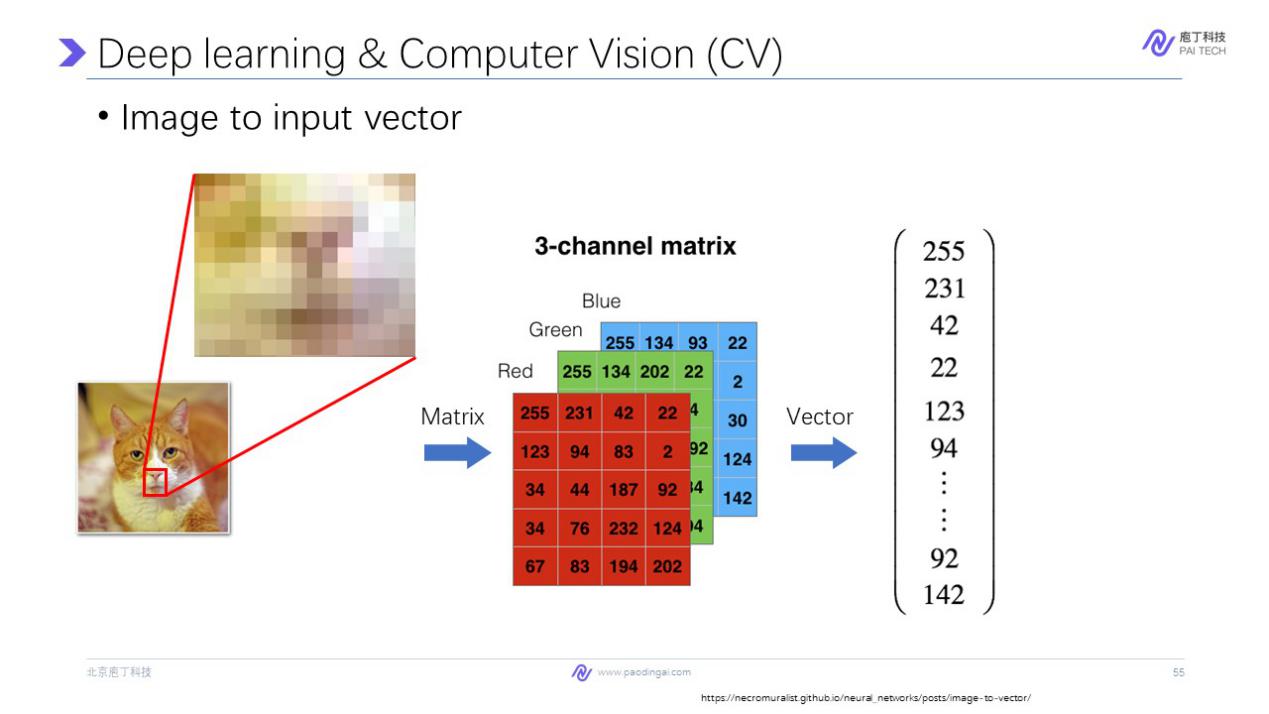
Applying AI on business document is called document intelligence, which as introduced by Dr. Luo, and involves the ability to read, understand, and interpret business documents. Most of the technologies behind AI is deep learning-based. And the applications professor Luo has introduced, like information gathering, intelligent review, and intelligent writing is all based on deep learning.

So next we will give you a tour on technologies behind this document intelligence applications. We will first learn computer vision, which is about how to deal with image. And the next, we will dive into natural language processing, which is about how to deal with text.

Now let's first talk about computer vision from two aspects: Document Parsing ,and Table Parsing. In this section, we want to answer this question. How can PDFlux introduced by professor Luo, copy a structured table from PDF? As we introduced, PDF does not store the structured information. So how can PDFlux output it from nowhere?

In fact, AI looks at the PDF page, regarding it as an image, and understand the table from its layout of lines and characters, which is the same as we human understand a table. The technology behind it is come related to computer vision, which studies how machine handles with images and videos. We introduce computer vision with the simplest problem, which is image classification.

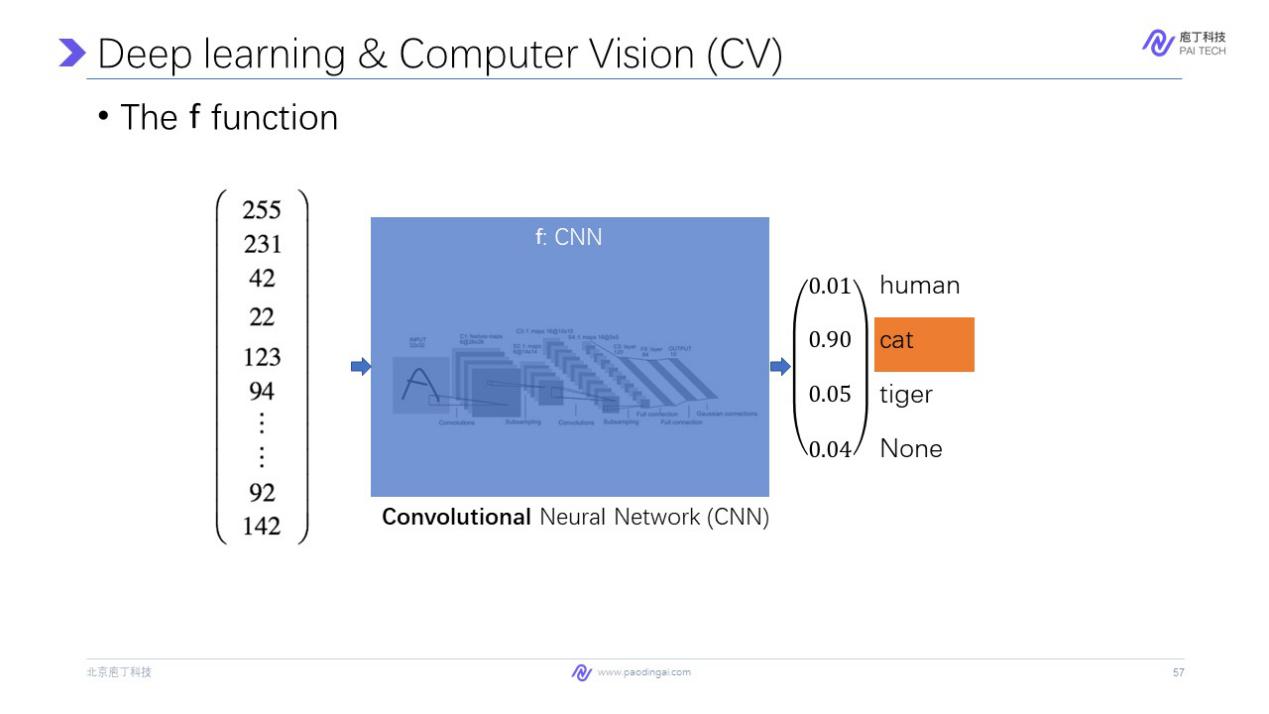
So how can the machine classify whether a picture is about a cat, a tiger, or a human? Recall that deep learning model is a function that accepts as input a vector and output a vector. So how can we connect the image classification problem with machine learning?



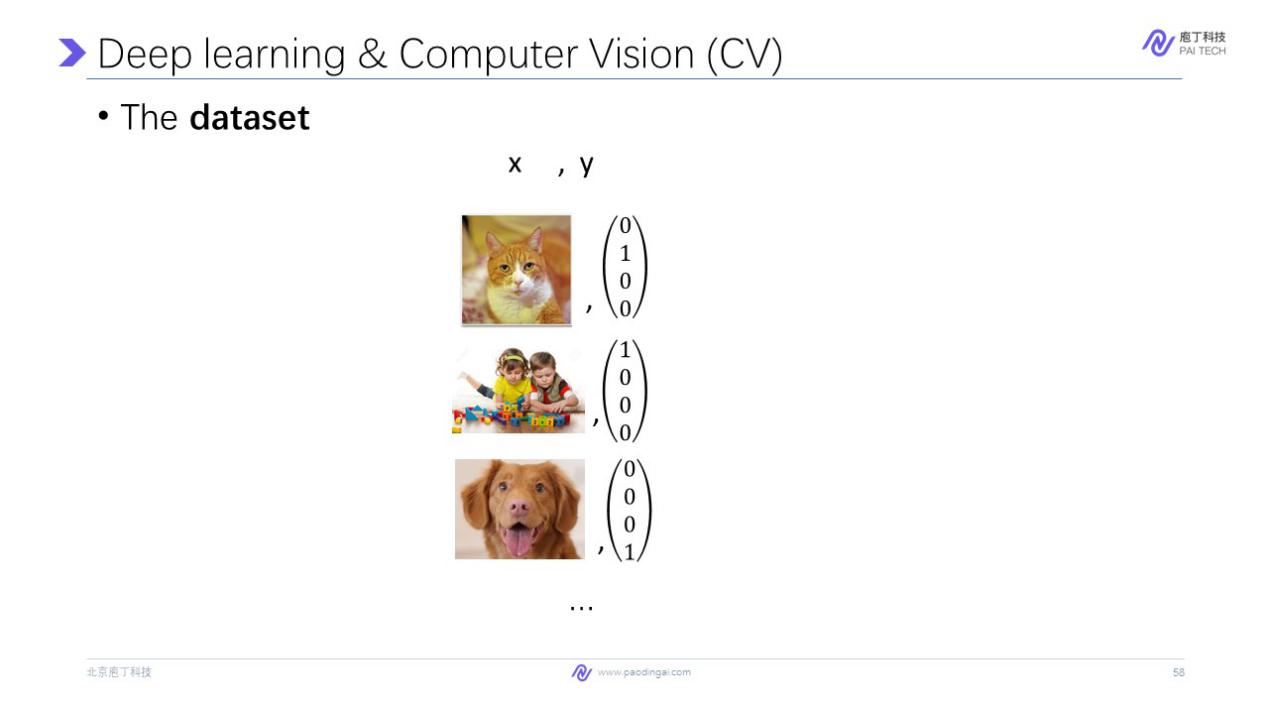
We will introduce this on how image is converted to input vector. First image is actually composed of pixels. Here each box is a pixel in the the picture, so a picture is indeed a matrix of picture. And each matrix have three values that record the red, green, and blue value on this pixel. A image can be directly converted into three matrices, red, green, and blue. Here the value in a matrix of right value is the value that this pixel have on how bright the pixel is. Here a larger value is like 255. This pixel is very red. However, on the contrary, low value like 0 or 2 means this pixel is very dark.

Then we can reshape these matrices by screening them into a flatten vector, which is usually very large. Now we convert an image into a vector. For example, here the matrix 255 is the value converted here. This is the corresponding of the value in the matrix and the vector. Then now we have converted the input into a vector. When we consider how to converse output into a vector, the output of the function on the model should be choose a class of the image like weather is a human, a cat, a tiger, or none of above. And we convert a decision of what the class of on the model output with the vector of four dimensions where this dimension is related to a class.

For example, the value in the second dimension, 0.9 shows the probability that all the likelihood that the model thing, the image is about a cat. We have converted image into a vector and converted output into a vector, then we introduce the f function here.



Usually we use a special kind of function that is called Convolutional Neural Network, or CNN for short, that takes as input a vector of a image and output a vector. And we don't dive into what is inside a CNN model. If you're interested, you can find a lot about CNN on the web. But here I want to tell you some of the characteristic about CNN. First, it is layer lives as most of different model hymns.



And second, it does a convolutional function on the image, that means it computes a feature of local context of each pixel. When we introduce the data set, we need for image classification. For each image, human labels, what classes it belongs to has a 4 dimensional vector.

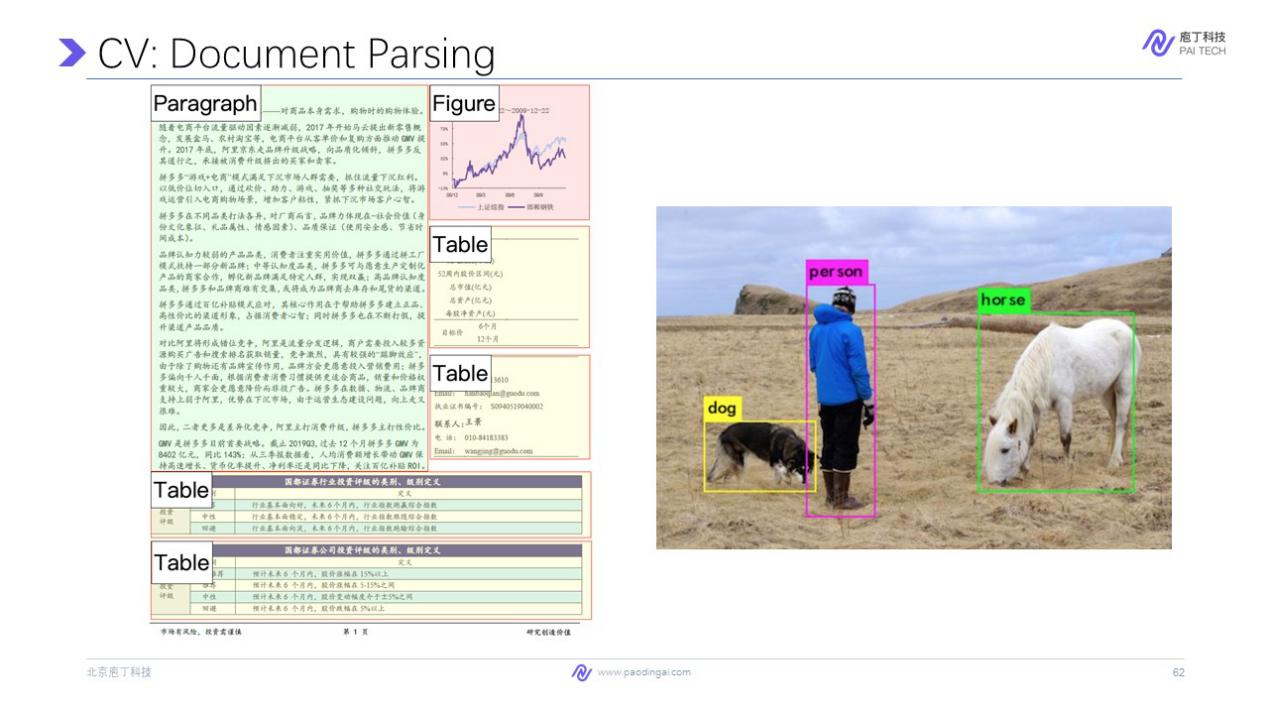
For example, for this image, human label 100% is about a cat. And for this image, human person is certainly about human. And for this image, the human label is not human tiger cats. So the training data set would be like this. For each image, human labels what class it belongs to as a 4 dimensional vector.

And then we will describe what is the loss function. Given an image, there would be a ground truth of this image. And there would be a prediction from the model. And we can see there are differences between them. We might use the square loss as we used in rent price prediction. But however, the standard loss function for classification problem is cross entropy loss. Here we use cross entropy loss instead of square loss is because the prediction of the model is usually very secure to zero or one. And a prediction of 0.8 is usually not regarded as a good result.

And however, for square loss function, if the difference between these two values is very close, the gradient will be very small. And when we optimize the loss function using gradient descent, the small gradient usually will make the optimization process very slow. And here, I don't want to dive into this formula as you might not interested in this math details.

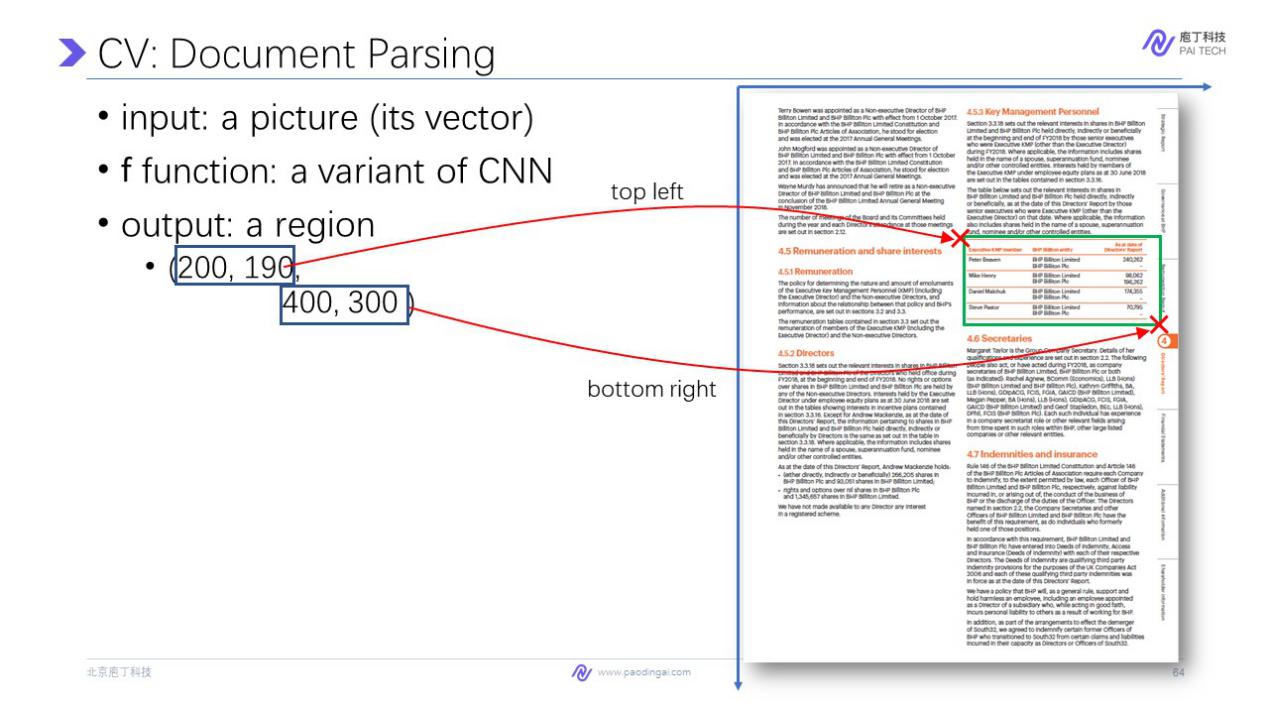
In summary, we have introduced all the basic elements of machine learning used in image classification. Here we summarize it as follows. For the hypothesis space, it uses CNN architecture. And the data set is composed of image and human labeled factors. And loss function is cross entropy for classification problems. And we use gradient descent to optimize the function. So we can get a function like we did in rent price prediction for image classification.

This is the introduction about computer vision. Next we introduce the answer to the question, how can PDFlux copy a structured table from PDF? It involves two steps. First, document parsing, that is to detect the region of table from a PDF page. This is necessary as PDF only have idea of characters. It has no information about where is the table. There is a figure of paradise. And after we detect the region of table, we need to recognize the inner structure of a table.

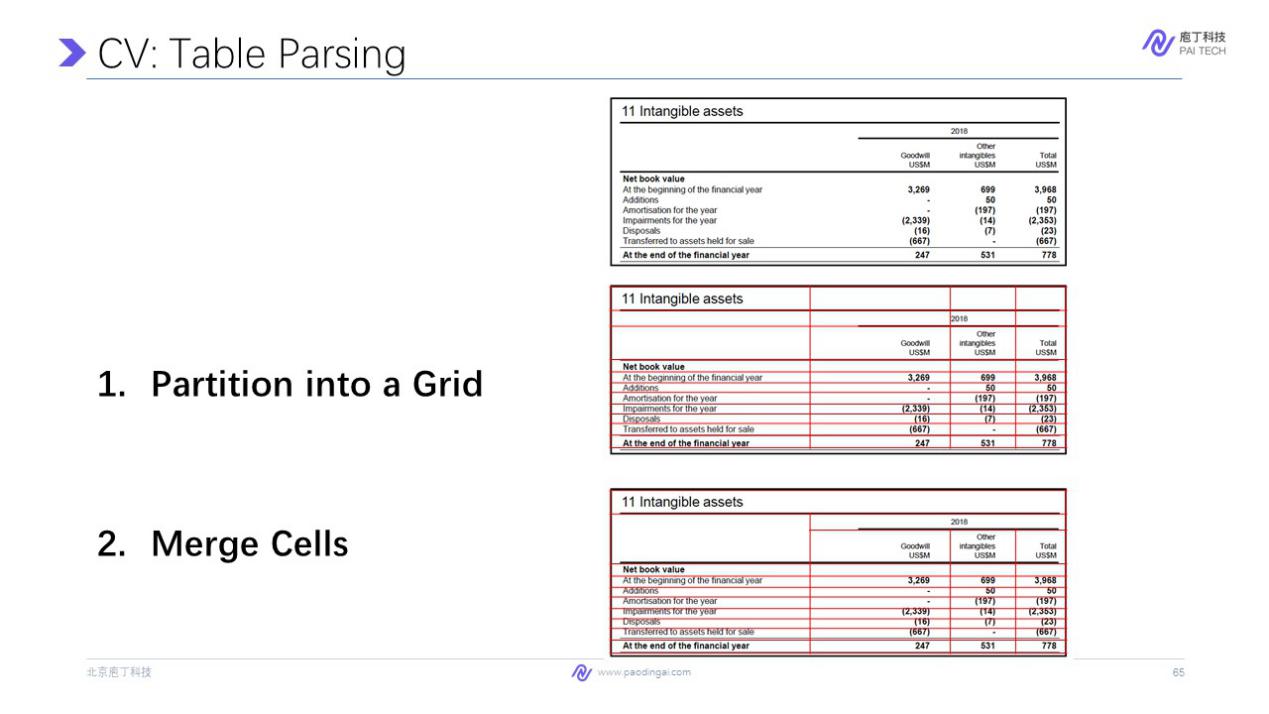


Now as for introducing the first step, the document parsing step, document parsing aims to separate a PDF page into regions of paragraphs, tables, and figures etc. This is similar to object detection problem in computer vision, where the goal is to detect things like dogs, persons horses in our image. And we also regard a page of PDF file as the image.

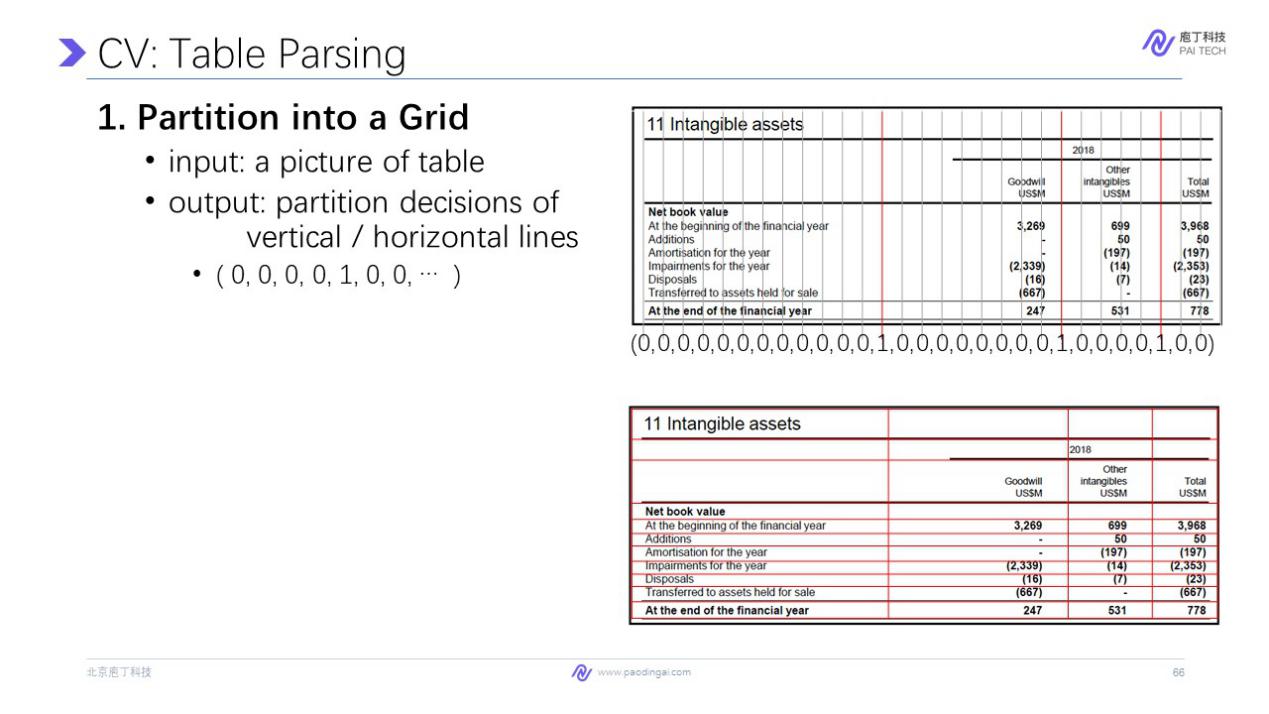
Here we consider a simpler version. Document aims to detect all the elements in a document. Here suppose a page has exactly one table, and we want to locate it. So the image, the input of this algorithm is a picture of a page in a document. And we'll convert it into a vector as we introduced before. And the function is a variant of CNN. The output of this function is a region action to how we represent the region.



More specifically, the first two dimensions, the region vector is a 4 dimensional vector. The first two vector is the (x, y) position of the top left position corner of the table, and the following two dimensions is about the bottom right corner of the table. So with this four parameters we can draw a rectangle from this top left to bottom right. So this task takes as inputs an image and outputs a vector, which is we have already seen before in a cat classification. So after positioning the table into different regions and detect the region of table in a page, we now parsing the table so that we can get the inner structure of it.



Given an image of a table, we parse it into steps. First, partition it into a grid like this, where each cell in this grid is the smallest partition we should get in this table. And then we merge cells. For example, 2018 should be merged cells of these three sub-cells. And also this is a blank cell, which should not be partitioned into two cells. So this partition and merge process is very similar to how we create a table with excel, that we first fill the data into the grid on excel. And then we merge the cells so that it looks better.



For the first step, we partition the table by considering all possible vertical and horizontal position decisions. For example, in horizontal position, there are many lines at which we can partition the table. We need to decide on each line this is a correct position or not? So the output is a vector of decision on each line. Most of them are zero as these lines are not correct lines partitions, and value of one indicates that this is a good partition. So the input is a picture of table and output is the partition decision of vertical and horizontal lines. Here we only draw the vertical lines. And the result should be a vector, that is very long. And the result would be like this, with which we only consider the prediction that says this line is a good partition.

Then for each adjacent cells, they are adjacent to each other. We need to decide whether we should merge it or not. so the input of this problem is two cells or a picture of two cells, and crop this small area of this picture and converted it into a vector to be the input of the model. And output should be a scalar. That is the likelihood that we should merge these two cells, which ranges from zero to one.

And this is also achieved by CNN as we introduced before, which is specially designed for images. So after we merge the cells, the inner structure of a table is restored. So with this two step, that's parse the documents and detect the tables, and then parse the table and get in a structure. We can achieve the function that PDFlux readers to copy a table easily from a PDF file.

Then we move on to the second section. That is natural language processing. And in this section, I want to answer to this question, how can we understand the text in professional documents? For example, how can we understand the meaning of this sentence? The revenue of China Southern Airlines was some value in 2019. And answer to this question is very important for the task like intelligent review and intelligent writing. And the techniques behind this is related to natural language processing, or NLP for short.

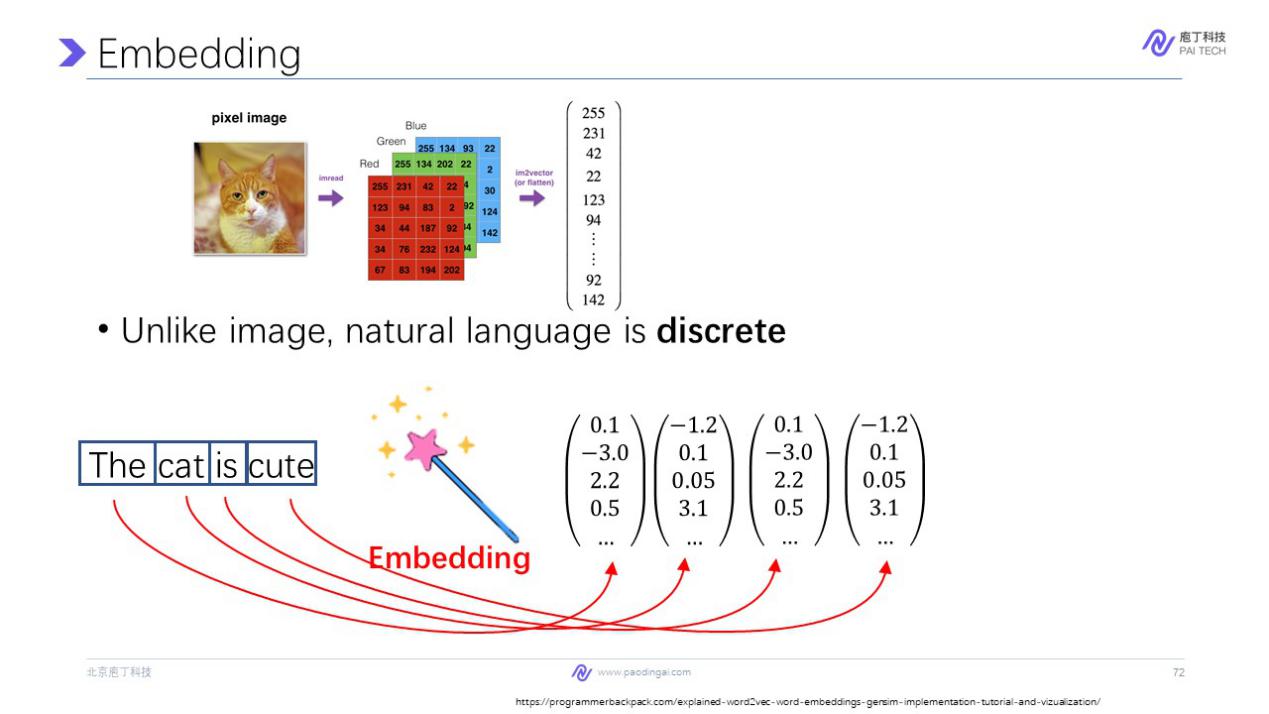
we will introduce two tasks on understanding the text. The first one is named entity recognition. It is to detect entities like revenue is a financial indicator. And China South Airlines is a company. The second step is relation extraction. That is to detect semantic relationships between or among entities and form a meaning, how can you used to express a fact.



And entity recognition is usually conducted as sequence labeling task. Here, the sequence labeling means we give each word a tag on whether or not it is a part of entity. Here, the cross mark means it is not a part of an entity. And the correct mark means this is a part of an entity. And by connecting the adjacent entity words. For example, this word is marked as part of entity. And it is adjacent. So we think this word is an entity.

So how can we give each word a tag using deep learning model so that it can be conducted in a more automatic way. Recall that deep learning is all about functions and vectors.

To handle image, we convert it into a vector where each dimension corresponds to a pixel in the image. But on the contrary, natural language is discrete. That means it is a sequence of words where each word is actually a discrete symbol. Unlike the values in image that continues, it seems very hard to represent a word as in a vector which is required by machine learning.

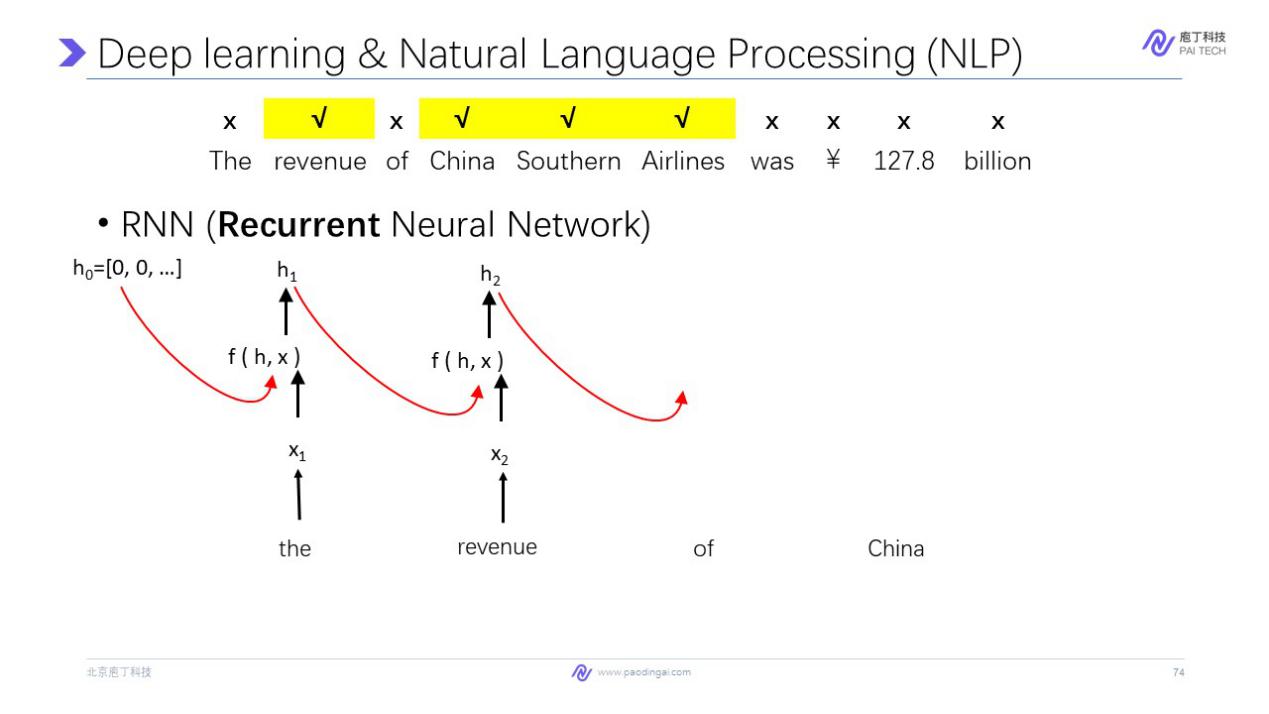


The way we handle natural language in deep learning is a technique called embedding, which means we embed a word into a high dimensional space so that a word is represented by a vector. For example, we convert the word "the" into the vector like this. And the "cat" is a vector like this. And a sentence as a sequence of word is converted into a sequence of embedding or vectors, as shown this picture.

In real world, the vector usually have hundreds of dimensions, so that it can capture the the meaning of our word, and distinguish among each other. The embedding technique here seems to be a magic. So I draw a magic bar. But in fact, it is just you can imagine it as a data base, where you input where each word is corresponding to a vector. And this vector is learned during the training process. You can regard this as a part of the f function as we mentioned before, however, by converting each word into a vector is not enough for entity recognition.

Because one important characteristic of language is context. For example, if we input the vector of the word southern, how can a deep learning model predict whether it is a part of an entity or not?

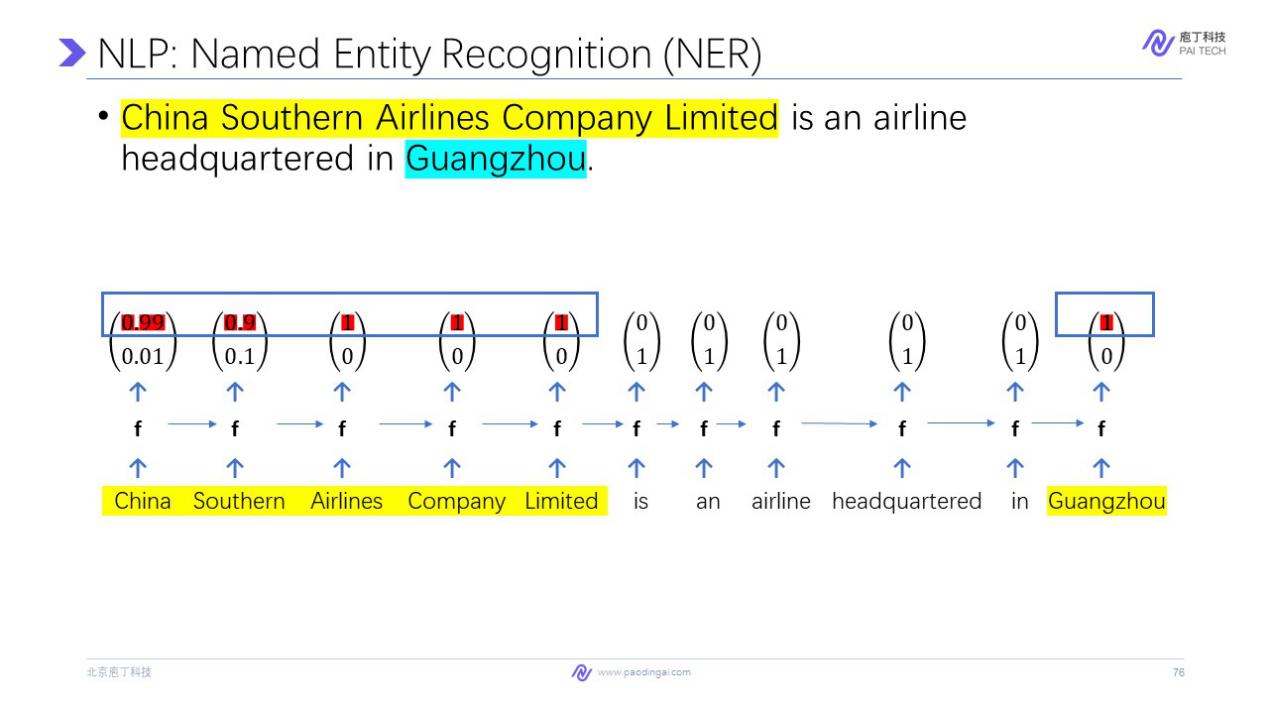
Actually, I don’t think function can decide this because even as human cannot decide, whether southern is a part of an entity if we only look at this word. We need to put this word into a sentence.



And by looking at each context, we can decide whether it's a part of an entity or not. This is achieved by a special kind of deep learning model called Recurrent Neural Network, or RNN. It is called recurrent because the same function is applied on the sequence of vectors repeatedly.

Here that function accepts as input two vectors. The first one is the input of the last word. And the second one is the embedding of current word. So the function connects the current word with previous words. And the input of this function h is a high dimensional vector that records the information of the preceding sequence. And this function is applied recurrently to every word in this sequence. As shown here, so that a hidden vector like h2 records not only the information of revenue, but also the information of the word "the", so now each vector here records the context information of the world.

But we still need a function that predict whether it is not a part of an entity. So we had a new function g here. It accepts the input h and output of two dimensional vectors. The first dimension is the probability that this word is part of an entity. And the second dimension is the probability that this word is not part of an energy. So with the help of embedding that we can convert each word into a vector. And with the help of recurrent neural network that connecting words in the sentence. And a record is context information. We can finally decide whether a word should be a part of an entity.

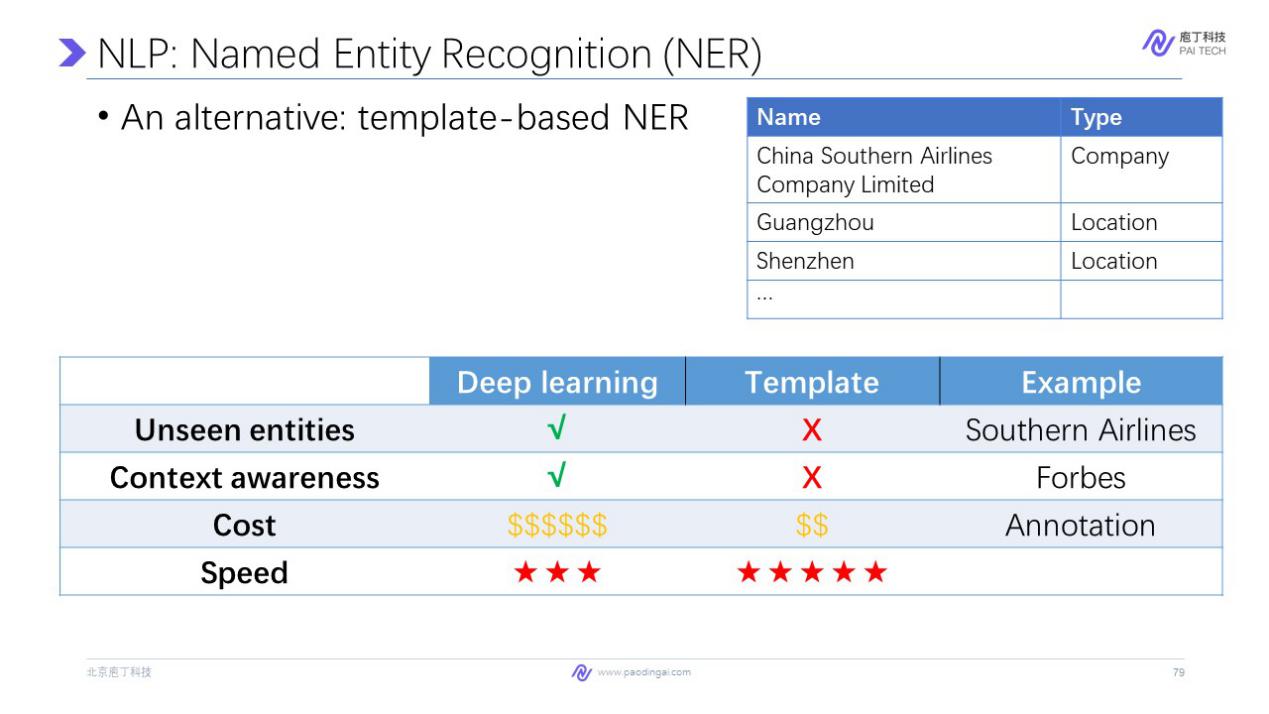


So the predicting on the whole sentence will produce a result like this, where each word has an output of two dimensional vector like this. where the first dimension is 0.99 and the second dimension is 0.01, and so on.

And here we think that if the the probability of being part of an entity is larger than not being part of an entity, we think it is a part of an entity. So we connect. So here, these five words is regarded as part of an entity and we connect them to form a whole as an entity. That is China Southern Airlines Company Limited.

Moreover, notice that there may be different kind of entities. For example, this sentence, China Southern Airlines Company Limited is an airline headquartered in Guangzhou. Here China Southern Airlines Company Limited is an entity of type company. And Guangzhou is the entity of type location.

So how can we distinguish them? We can modify the g function here so that it can output a 3 dimensional vector. The first dimension is the probability of being part of a company entity. And second dimension is the probability of being part of a location entity. And the 3rd dimension is not above. And the result will look like this, where on each word we output a 3 dimensional vector. And we connect the result that have the same type of an entity here, we can see that this five words is an entity of the first type, which is a part of a company. And this word is predicted as the second class, which is part of a location.So the company and location are distinguished.



Despite the deep learning approach, another way to conduct entity recognition is by matching templates or template-based NER. We can maintain a large data set of entity using string matching to extract. For example, we see Guangzhou in a sentence, we directly extract it as it is matched to this record in the database.

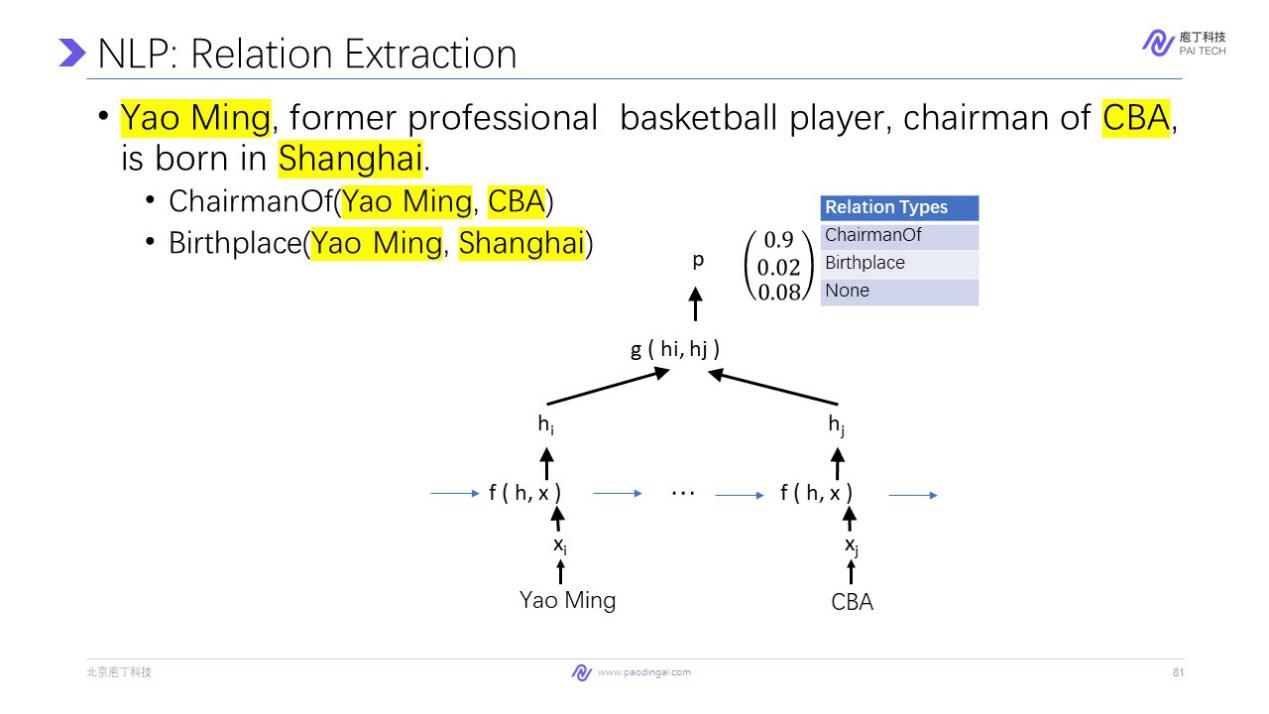
Comparing deep learning and template-based method, deep learning has two advantages. First, it can extract unseen entities. For example, in the training data, we know that China Southern Airlines Company Limited is a company. But when in prediction, it can be right in different kind of forms, for example, Southern Airlines.

But deep learning model usually can detect this kind of unseen entities when predicting on new text. But template-based method will not extract an entity, if it didn’t appear in the database. And the second advantage of deep learning is that it is context aware. In fact, as we mentioned before, in many cases, we can't decide whether a word is a part of an entity as we need the context information.

For example, when we look at the word Forbes. Sometimes it is a name of a person. And sometimes it might refer to the magazine. And this difference can be captured by contexts. For example, if we see Forbes sentences like Forbes was born in 1955. We can judge that this is a name of a person. But on the contrary, template-based method cannot distinguish a same word in different contexts.

However, deep learning has also some drawbacks compared to template-based method. First, deep learning model is usually more costing. This is because we need a lot of training data to train. And program of deep learning usually required some kind of special education on machine learning. On the contrary, template-based method is simpler. However, when the size of the database increases, it becomes more difficult to maintain the database. And the cost will increase. On the contrary, the cost of deep learning is usually decreasing as we have more and more data. Since the prediction is learned from data instead of hard programmed by human.

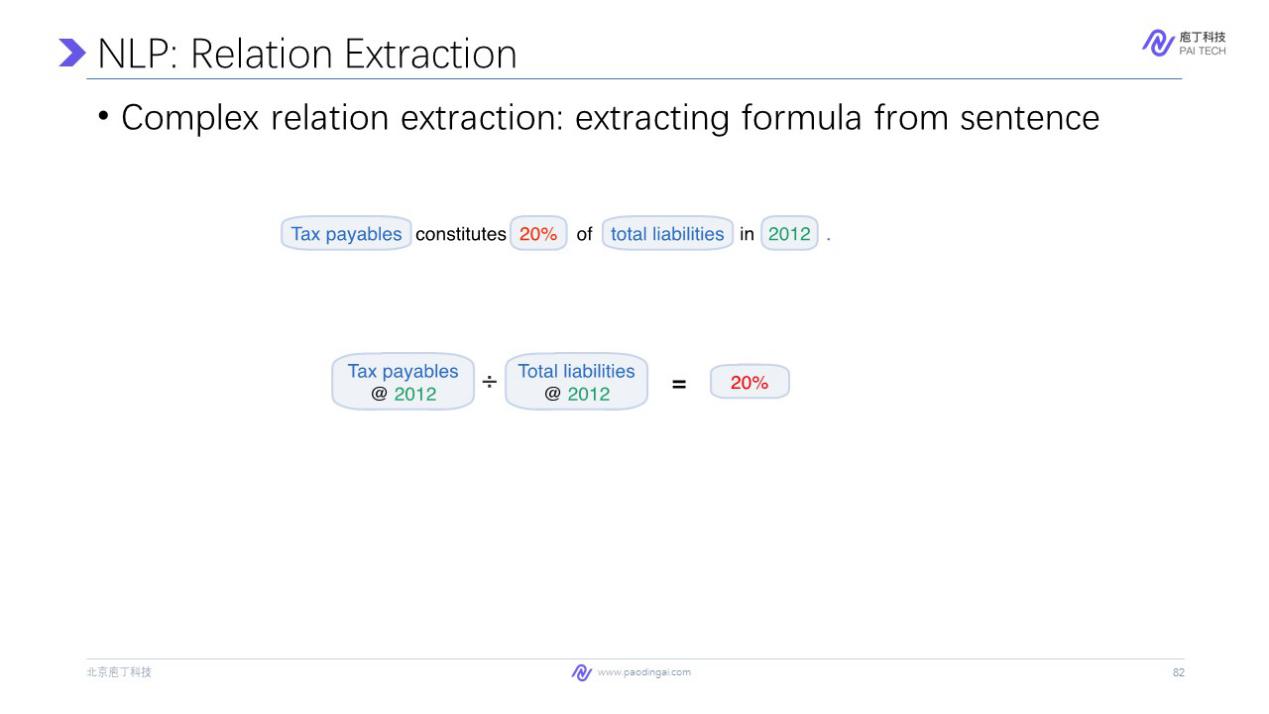
And the second drawback of deep learning is its speed. But here, as I indicated, the speed of deep learning is actually also acceptable in most cases. And with the development of devices that is intended for the deep learning models such as GPU. So in most cases, deep learning, we also output the result of a new entity recognition in a real tag, like template- based model do.



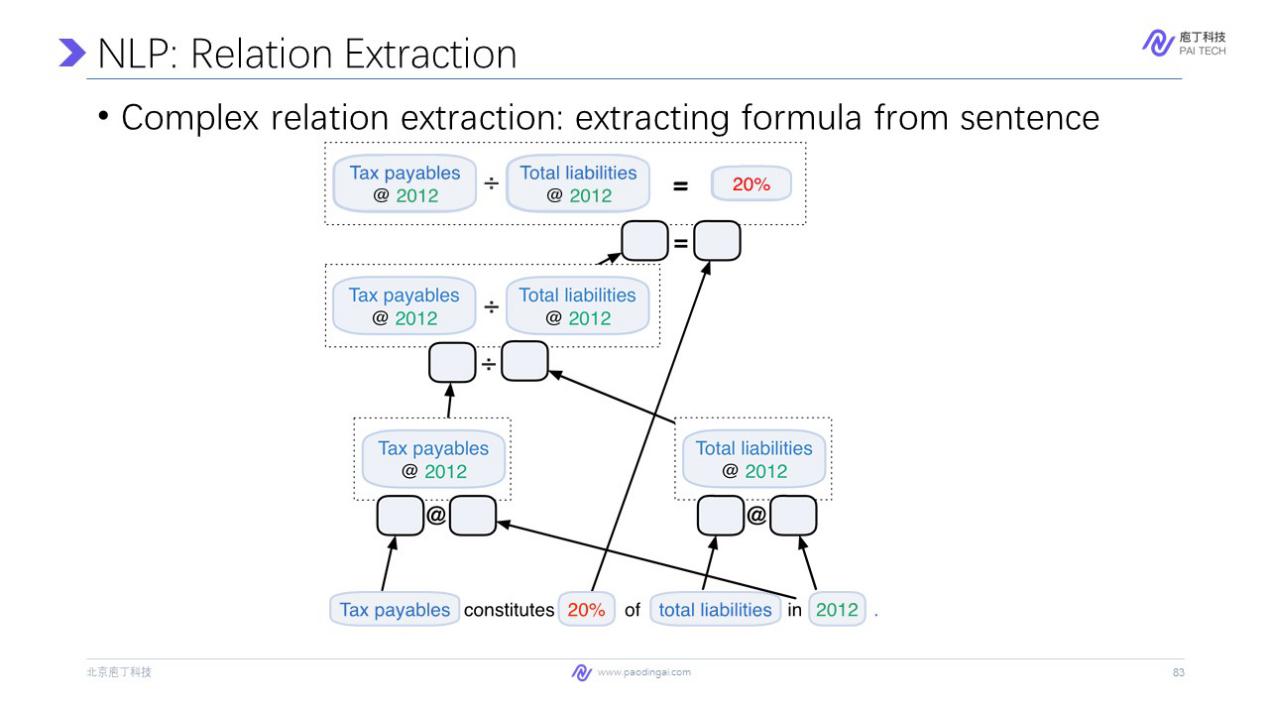
After we extract entities, the next step is to understand what this sentence is talking among them. For this sentence, Yao Ming, former professional basketball player, chairman of CBA, is born in Shanghai. For this sentence. Suppose these three entities are extracted from the previous step. we can understand that this thing has described two relations among them. First, Yao Ming and CBA has the relation of chairman of. And Yao Ming and Shanghai have the relation of birthplace. So extracting the relation among entities is a very important task. As entities usually have no meaning, you cannot express the fact with entities, but when you express or extract the relation between the entities, it usually have more concrete semantic meaning, as it can be used to express facts and events.

So we can regard this relation extraction for task as a pair-wise classification problem. That is, we classify the relation type between all possible entities pairs. Here we need to classify whether Yao Ming and CBA has some kind of relation, whether Yao Ming and Shanghai has some kind of relation and weather CBA and Shanghai has some kind of relation.

Here we suppose we only care about two kinds of relations, chairman of and birthplace. When we classify the relationship between two entities, for example, Yao Ming and CBA, here we use a RNN, as we introduced before, to represent each of them as a vector hi, hj, then we use another function g that takes as input two vectors, hi and hj and output a vector about probability that these two entities has the relation chairman of, Birthplace, or non of above. We can do this classification on every possible pairs of entities in this sentence, and get the results of which two entities have relations.



But to help us with paperwork in finance industry, usually we need more complex relation structure. For example, apply it in intelligent review and intelligent write. For example, in this sentence, tax payables constitute 20% of total liabilities in 2012, to check the number 200 to 20000 is correct or not. We need first to understand its meaning. It is a division of two indicators, tax payables of 2012 and total liability of 2012. But how can a machine, a deep learning model do it? Actually, this formula is a nested structure which consists many relations.



Let's discompose it from bottom. First, we need to extract entities in this sentence. For example, tax payable is a financial indicator, 20% is a value,total liabilities is a financial indicator and 2012 is a time. Then we extract the relation among these entities. At the first layer, we can understand what we need to extract. In this sentence, it describes the tax payables and 2012 and total liabilities in 2012, where each relation here is related to a financial number.

Then we extract that the sentence describes a division relation between these two sub-relations. And finally, we know that this division and this number has an equal relation. And this layered extraction of complex relation is very important to extract complex information expressed in financial documents, especially when we conduct automatic numerical cross-checking, as we introduced before. When we extract each relation here, the process is very similar to the simple relation extraction model as we introduced in this slide.

So in summary, in this part, we introduce technologies behind document intelligence. And now I want to give you some demo about the applications we introduced before and hope you have a more intuitive experience of this system.

First, go through the Hunter, which is the intelligent searching system we introduced before. For example, we want to search for the the newest prediction of profit of BYD company from the CITIC securities. Here is a list of result where each result is a PDF file. And it has several snippets where this snippet is clickable. We click this one and the browser, the search engine directs us to the this table, which is the same as we look. Here we click that table too.The prediction of profit, it directly gives us this table so that we don't have to search inside long document.

So with this, the search engine, usually the two step information gathering that search in your folder and find the document. And then search inside document is simplified and combined into one step. Then you can copy this table into a Word. You can see that the inner structure of the table is correctly recognized.

We can also edit it in the right side of this PDF. For example, this is a more complicated table. And the things like merge cells are correctly recognized by the AI model behind this search engine. We can copy it directly to the Word. And you can see the format is very neat. And we don't have to modify the merge and even the alliance of excel, this system combines the search engine and the PDFlux function we introduced.

The second part is about intelligent reviewing. Here we look at an example here. We can see that every value in this document is parsed into a structured format, for example, this value is about a current issue in 2018. And this value is connected to two values or three values in this document. And the first connected value is in this table. And we can see this table in the PDF and in this table, we also recognize the meaning of the value. You can see that every number here we post is the highlighted with an underline in tags so that you can check on each value whether the system has check it for you. And you can see whether there is conflict within your document.

Here is an example of the intelligent review system that we introduce on a compliant assessment of against listing rules. On the left is the PDF of the annual report, on the right is the AI suggestion, for example, for the first rule, AI suggests that you should look at this paragraph. And such suggestion is very convenient so that we don't have to search inside very long. Document, for example, this document has more than 200 pages. AI will give you a suggestion about whether this on this listing rule or whether the document disclosed it or it has a negative statement.

So in summary, we introduce the applications of AI and machine learning on paperwork, as well as the foundation of AI. I hope you enjoy the lecture, and we really encourage you to experience the power of AI so that you can be more productive and focus on more creative work in the future.