Hello every one, I'm Ou Liechuan. My internship in Hitachi is almost finished. It started from July and will end on this Thursday. Next I will tell you what I have done and what I have learned during the three month.

All of my work is about wells. There are many geological reports, amounting to about 10,000. They contain rich information but can not be read directly because every page is a image and the information is embedded in the variant layouts. Before my supervisor has designed a rule-based method to extract them while recently I have tried using a machine learning method, deep neural network to improve it. What I implemented can be seen as a tool. The input is pdf files and the output is a json file containing the information we are interested. To make the tool available to other people, I made a website where users can upload pdf files and download result files. The tool and website are deployed at the GPU server, where many applications are running. I use docker to make my applications isolated from others. In addition, I designed and implemented a website to visualize the production distribution and well logs.

Let me recall the problem. What's we need is the rock descriptions in the reports. But different operators use different formats for depth range and rock descriptions. Here are three examples. Current solution uses rule-based extraction. That is, for each format, we define a extraction rule. However the page layouts have many variants that rules can not handle.

Our new solution is based on deep neural network. Specifically, we use CNN. It has a characteristic of translation invariant, which means for two images with same content but different looks, the CNN will output similar features. Our solution can be divided into two parts. First, we need to find out those pages containing descriptions. Every page is classified one of the five classes. They are descriptions, form4, form6, form6 page3 and others. The network structure is VGGNet, which is the most popular cnn model and is developed by Oxford. As our training data is about 50,000 images, it's not enough to train a cnn from scratch so we do fine-tuning on a pre-trained model. After finding out description pages, we need to extract description blocks. We treat description blocks as a kind of object then we can use object detection method in computer vision to solve the problem. Again VGGNet is used. On the top of it, we train a binary classier to judge whether a image patch contains a description and a regressor to output the bounding box.

Here is the result. You can see that for description and form classes, recall are higher than 95% so we will not lost information and the precisions are also good. For class 'others', the recall is low comparing to description and form, it's not a problem because it means some other pages will be misclassified but the following detection or extraction step will ignore such invalid page. As for detection step, the network will output a confidence value, which indicates how likely it's a target object. We can set a threshold. Obviously higher threshold results in higher accuracy but lower recall. Here we compute a roc value, 0.86. It means most description blocks are detected.

Next I want to make a comparison between the previous rule-based method and the current deep learning method. About 1,400,000 descriptions are extracted before but 1,530,000 currently. Totally 5000 pdf files can be extracted valid descriptions but 5300 currently. So we can say deep learning approach can handle more cases. In general, rule-based methods are vulnerable to noise content and layout variance while cnn is robust thanks to its property of translation invariance. However, machine learning methods are not flexible to handle special cases. I will show you some examples.

Here are two bad cases. This page is typically two column type. If we input it to Tesseract, the output is some text blocks. The depth range and description are in separate blocks. For each depth range ,we need find the corresponding description. A simple rule is they should has same or very close value in Y axis. It's effective in many pages but not in this page. You can see this digit is very near to the Y axis but the digit here has a distance to the Y axis. It's because the descriptions are rotated a little, which results in the difference between Y depth and Y description is bigger than the matching threshold. For the right page, I don't know why the rule-based method doesn't get any description but I guess it's because these noise content.

Next let's see a bad case in machine learning approach. They are description pages but very different from the most usual case. It's easy add a rule to cover this special case but difficult for deep learning. We can not train the model again because there are not enough amount of data available.

In addition to the rock description, form information is also extracted. This is the page3 of form6. It contains stimulation properties, like date, stage and formation. This module is developed by Shinjo. A template is used. Totally there are about 6000 among 10,000 pdf files in which we can get such information.

Next I'll share you what I have learned from the work above. First If the rule is known, just use it. It's simple and also effective. Then if the rule is too complex to express and plenty of data is available, machine learning can be used. Finally if the goal is to cover as many cases as possible, both approaches are needed. Machine learning is responsible for the general cases and rules can be added to handle special cases.

The tool I implemented is deployed on the GPU server. Many users are using it and many applications are running on it. How to get a isolated environment? The solution is docked which can achieve system level abstraction and isolation. It's like a independent system and all the dependent libraries can be installed in the docker. Also it's easy to migrate to other machines.

What's more? My another main task is the visualization of wells data. The goal of visualization is to show the well information in an intuitive way. I think both the global and local information should be presented. So a heat map is used to express the production distribution and colorful markers are used to tell how much the production is. As for implementation, the back-end involves flask and mongoDB, the font-end uses googleMap and javascript.

Let me show you the webpage.
Once opening the page, you can see a heat map. Hot area means there are more wells and more production. Using this dropdown menu, we can switch between oil and water. Let's zoom in. Many markers are shown and the heat map is gone. Warm colors like red indicate high production while cold color like blue say low production. If we click a marker, a side window will be shown. Here are some information about this well. And the sensor data is plotted in this chart. Click the popup button, we can see a big chart. More detail is present.
Let's zoom out, the markers are gone and the heat map is back.
My report is done. Do you have any question?