Challenge 1 : Fire mapping

Authors :
M. Luca ORDRONNEAU
M. Louis REBERGA
M. Johan MONCOUTIE
M. Julian ETTARIAN

Version of
June 21, 2021

Contents

Introduction

We are 4 students who will enter our last year of engineering school with an artificial intelligence and data science option. After our first year on the basics of artificial intelligence theory. We challenged ourselves to do this challenge with no practical knowledge of image segmentation.We learnt a lot and were proud of the fact that we reached the semi-final.

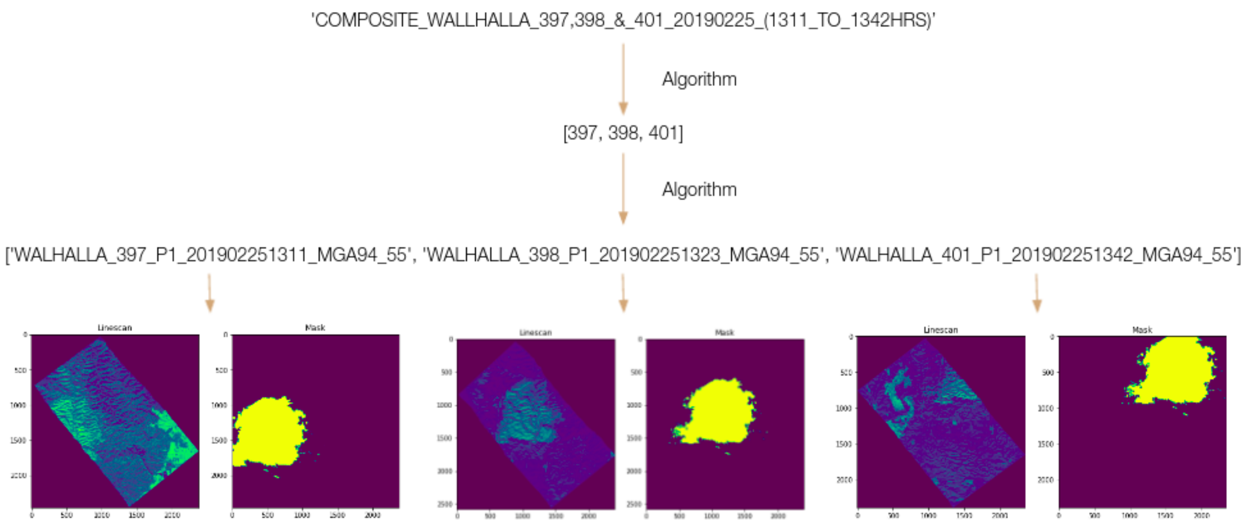
1 Key elements of our approach

There were three key points in the way we approached the challenge.

1.1 Data collection / selection

The first point was the collection, restructuring and analysis of the data provided. Indeed, to have access to all the data, we had to match polygons with linescans/composites. To do this, we made some algorithms, detecting a pattern on the linescan/composite name. Thus, we could correctly match them.

Here is an example with the different steps for match composite polygons :


'COMPOSITE_WALLHALLA_397,398_&_401_20190225_(1311_TO_1342HRS)'
Algorithm
[397, 398, 401]
Algorithm
['WALHALLA_397_P1_201902251311_MGA94_55', 'WALHALLA_398_P1_201902251323_MGA94_55', 'WALHALLA_401_P1_201902251342_MGA94_55']
Linescan
Mask
Linescan
Mask
Linescan
Mask

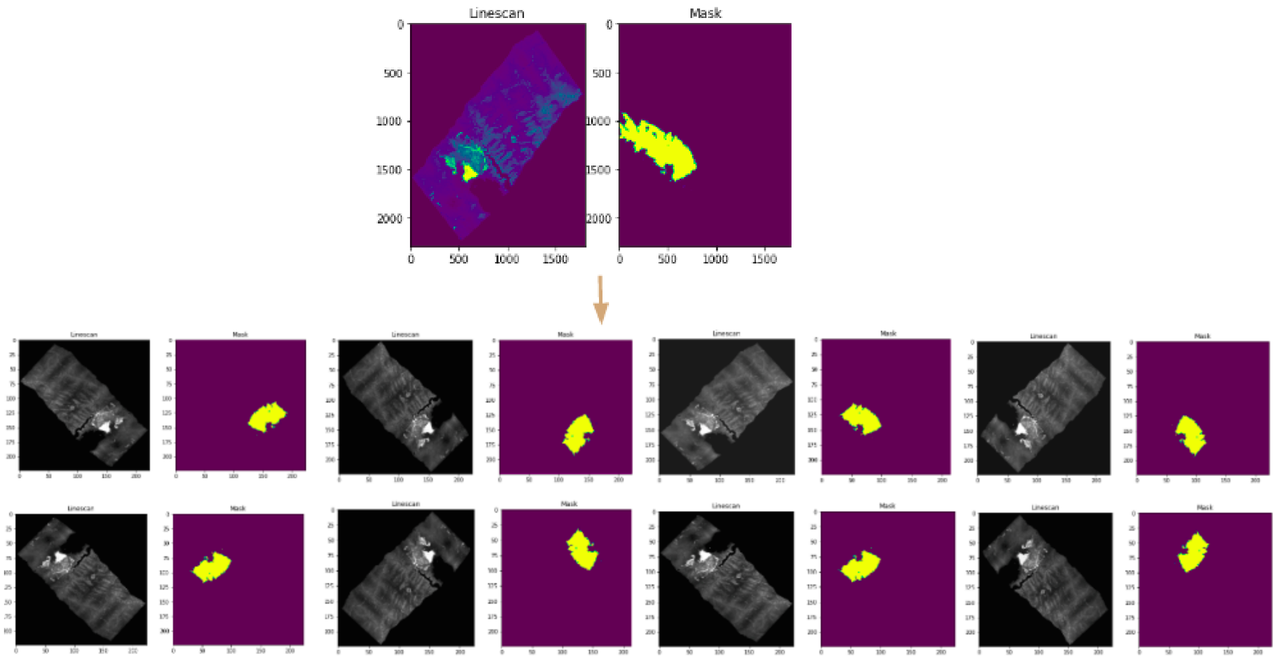
Moreover, we had to analyse the linescans with **the mean**, **the median** and **the histogram of the pixels** in order to keep only those that could improve our score, model. We have mainly removed linescans that contain clouds.

1.2 Data pre-processing

The second point and not the least was the pre-processing of the linescans so that they could be trained in our neural network. The objective was to bring out the fire zones as well as possible through several techniques :

- Resize (224x224)
- Gaussian Filter, to reduce image noise and reduce detail.
- Normalisation
- Image augmentation (Variation of Brightness, Contrast and Saturation. Horizontal, Vertical Flip, Rotate, Transpose)
- Modification of the mask, to avoid the model predicting fires outside the linescan.
- Application of contrast limited adaptive histogram equalization (CLAHE). It is variant of adaptive histogram equalization (AHE) in which the contrast amplification is limited. Indeed, AHE creates noise to be too amplified in near-constant regions. This is why we used CLAHE.

Here is an example of preprocessing for a linescan :



1.3 Model

The third point and final point was the choice of the model and the adjustment of these hyperparameters. Indeed, throughout the challenge we went from the **Unet** model to the **Unet with Pretrained ResNet50 Encoder** where the difference in improvement was substantial and then to the **Unet with Pretrained Inception ResNetV2 Encoder** where we have slightly improved our score. Moreover, we used the following metrics dice coefficient, IoU, that best simulated the F1 score and best improved our predictions.

2 Training Methods

2.1 Model

Throughout the challenge, we conducted extensive research into the different models for segmentation. We went through the Unet model which gave us an F1 score of around 0.5 and then we went through the Unet with pretrained encoder models such as ResNet50 and InceptionResNetV2 which gave us our best F1 score of around 0.73.

Using a pre-trained encoder helps the model to **converge much faster** in comparison to the non-pretrained model. A pre-trained encoder helps the model to achieve **high performance** as compared to a non pre-trained model. Furthermore, we used a pre-trained encoder because our dataset was not large enough and not diverse enough. Finally, with our limited computer vision experience it would have been difficult to build a model from scratch as the architecture would be too complex and would require a lot of knowledge and resources.

Here is a summary of our results according to the model :

Model	**F1** average score according to our submissions
Unet	0.487
Unet with pretrained ResNet50 encoder	0.665
Unet with pretrained InceptionResNetV2 encoder	0.723

2.2 Metrics / Loss Function

We looked for the most suitable hyperparameters for image segmentation. Indeed, at the beginning of the challenge, we used the loss function: **BinaryCrossentropy** which computes the cross-entropy loss between true labels and predicted labels with as metric the **accuracy** which can give a very good score without saying that the prediction is good (class imbalance). As these last hyperparameters were not conclusive, we did deeper research and used as loss function: **dice loss** and as metrics **dice coefficient**, **IoU**, **Recall** and **Precision** very powerful to evaluate our image segmentation model.

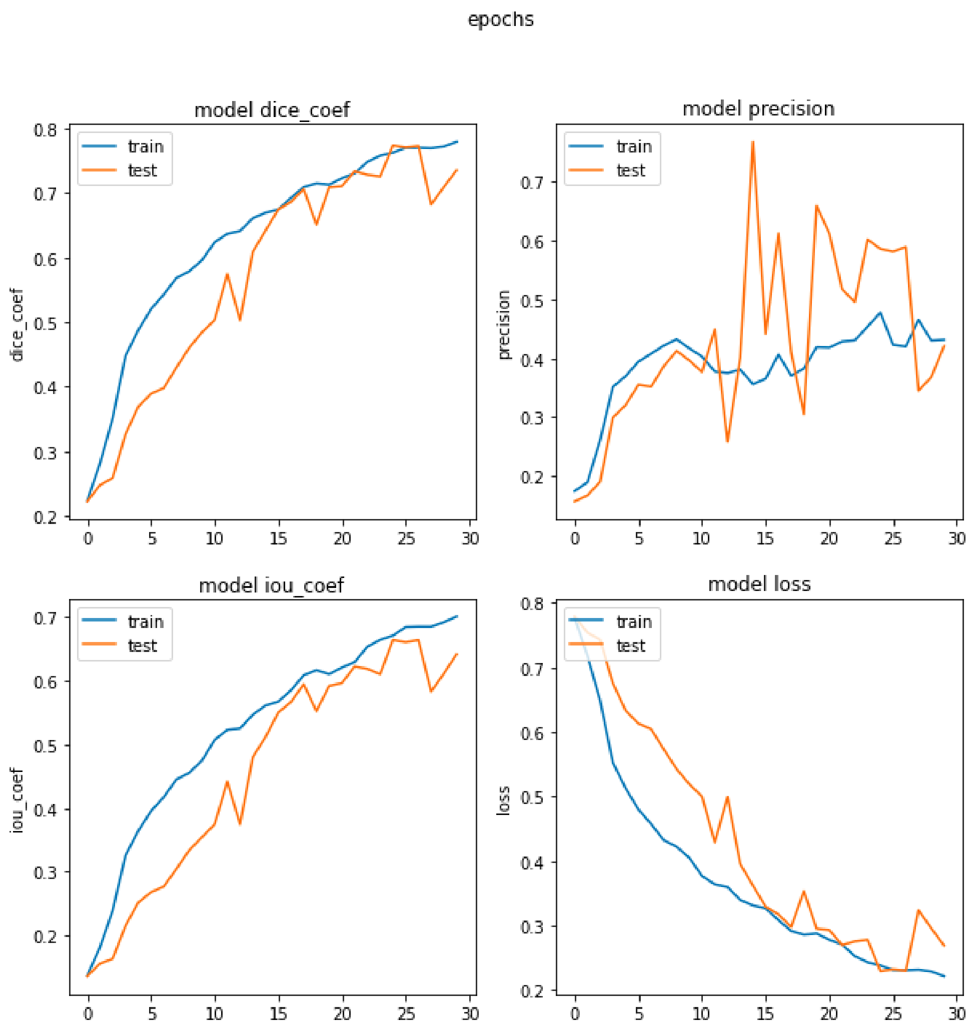
- **IoU** : Coefficient representing the area of overlap between the predicted segmentation and the ground truth divided by the area of union between the predicted segmentation and the ground truth. This metric ranges from 0–1 with 0 signifying no overlap and 1 signifying perfectly overlapping segmentation.

- **Dice coefficient** : This coefficient is used in statistics to determine the similarity between two samples, very similar to IoU. They are positively correlated.
- **Dice loss** : Very significant loss function in image segmentation from the dice coefficient.

3 Model validation / selection

To choose our model, we essentially based ourselves on the curves produced by the model for the different important metrics (dice_coef, dice_loss, IoU).
Indeed, as you can see below on the four graphs there are two curves, one on the training data (blue) and another on the test data, split of the training data about 15% (orange). We did several tests with different number of epochs (one training on all data) 15, 25, 30, 40, 70, 90. We concluded that with more than 50 epochs we were in **overfitting** (very performing model on training data but not on test data). Conversely, with a number of epochs below 20 we were **underfitting**. So a choice of the number of epochs between 30 and 40 allowed us to have an optimised model that could be adapted to all cases.

For 30 epochs we notice that the model converges well :



4 Predictions and morphological operations

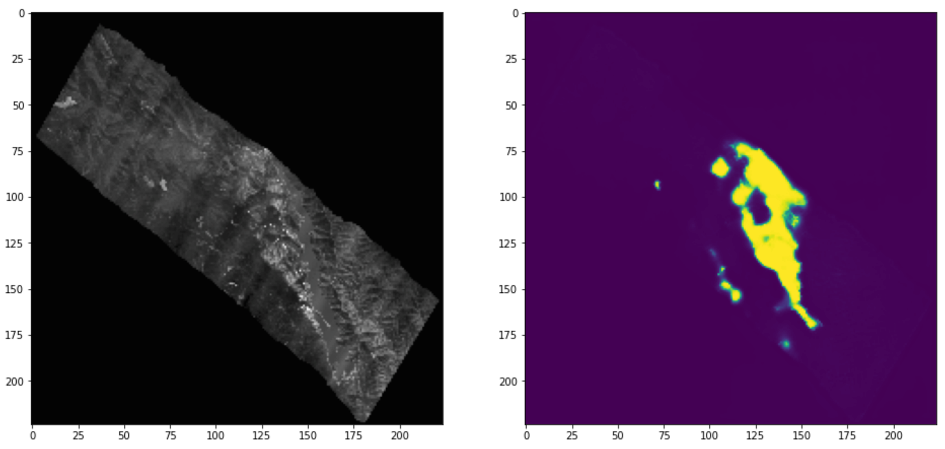
After selecting the model, the predictions generated by the latter must be reworked using **morphological operations**.

First, we applied a **threshold** to obtain a binary image. Then we used mathematical morphology with an **open kernel** to remove noise from the image and finally a **close kernel** to close small holes inside the foreground objects, or small black points on the object.

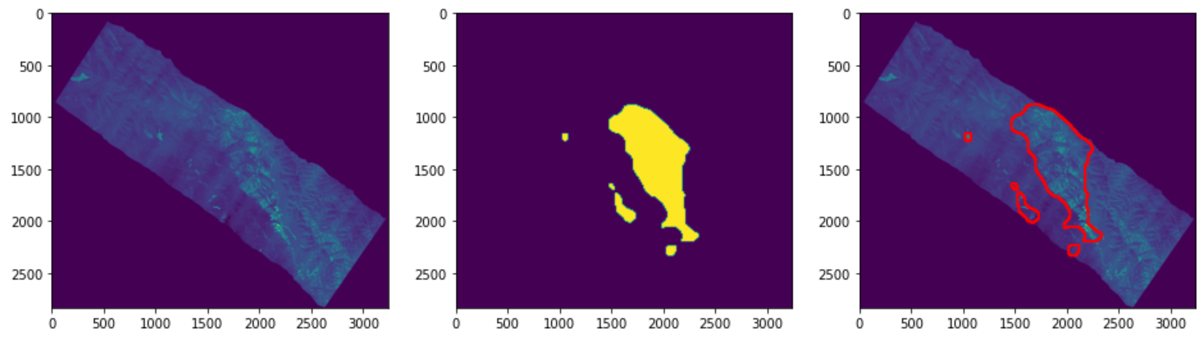
We used an **elliptic kernel** to have a more realistic prediction.

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \text{ 5x5 elliptic kernel}$$

Here is an example of a linescan predicted by our model:

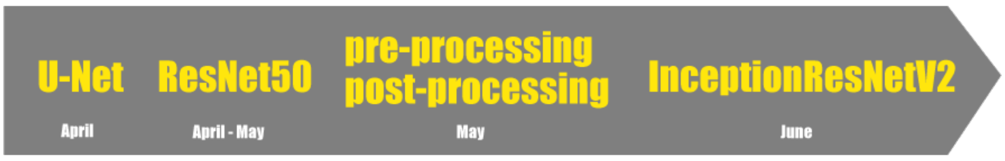


Here is the result after morphological operations:



5 Evolution of our approach

Here a simplified timeline of the evolution of our approach :



We first search the common machine learning approach used for image segmentation. We found the U-net algorithm. We all learned how the network is built (a succession of convolution, ReLU and pooling). We implemented the algorithm and tested it on the row data we retrieved.
Then, we found an improvement which is to use ResNet50 as an encoder. To increase the performance we decided to do pre and post-processing like image augmentation and filters.
Finally, we found a new way to improve our model : use InceptionResNetV2 instead of ResNet50 as an encoder.

6 Innovation and uniqueness of our approach

In order to solve the problem that is mapping fires in Australia, we choose to adapt the method to the data given.

- Because there are quite a lot of data, we choose to use a method that handles well a large amount of data : Machine learning.
- Because the data are pictures, we then choose what works the best with pictures treatment : Convolutional neural network.
- And finally, because we wanted to find the location of fire in the picture, we choose to use image segmentation CNNs.

What is innovative in our approach are the technologies themselves.
CNN and image segmentation are recent technologies that we believe have never been used to solve this matter.

What is unique about our approach is the model and results we ended up with.
What is unique is the combination of all the work we did. Indeed, the work we put in processing the data, the image segmentation model we choose and the post processing gave us results that are unique to our approach.

7 Mapping fires around the world

In the actual state of our model, you can use it to map fires around the world but the results obtained will depend on:

- The similarities between the environment of Australia and the environment used. If the two environments are really different, the result might be bad. The more the environments are similar, the more the model should work normally. To improve the reliability of the model in predicting the location of fires around the world would be to train the model with data taken around the world.
- The presence of clouds. Our model doesn't work well when there are clouds, so the less clouds, the better the results.

8 Most significant changes

During this experiment we tried many different strategies with more or less good ideas to improve the F1 score :

- Change models/algorithms
- Use of edge detection
- Image colorization
- Filters to remove clouds
- Reduce the mask according to the linescan

But it is thanks to the change of model from **Unet** then **Unet with pretrained ResNet50 encoder** and finally **Unet with pretrained InceptionResNetV2 encoder** that we realized the most important breakthrough that helped us to improve our score. We improved our F1 Score from 0.487 in average with Unet to 0.723 in average with pretrained InceptionResNetV2 encoder.

9 Hardest Part

Despite all these researches and various attempts, the hardest thing about solving this problem was the construction of a good dataset (creation of an algorithm to match the normal and composite linescans with the right polygons and the strategies to increase the dataset to best fit our model). Indeed, data is the key element of any machine learning algorithm and so this part of the challenge was a real important point on which we worked a lot to build the best train dataset to predict the best results.

Conclusion

Our model has the advantage of quickly predicting whether there is a forest fire or not by using a satellite image. Thus, we can help the fire brigade to react more quickly to predict forest fires in order to minimise the loss of life and property.