Coursework: Participating in a AICrowd challenge
Crowdsourcing and AI

Michael Mosimann

Université de Genève

Faculté des Sciences

Département d'informatique

The work done for the AI blitz challenge 8:

For this coursework, the received task was to participate in a AICrowd challenge and also observe how the participants behave. The chosen challenge was the 8 AI Blitz challenge: https://www.aicrowd.com/challenges/ai-blitz-8.
I chose to participate in the smoke elimination: https://www.aicrowd.com/challenges/ai-blitz-8/problems/f1-smoke-elimination.

At the beginning, i posted only the baseline as submissions to test the submission process and to see what score i need to improve on. The baseline just submits
I had been working on the code and looking around the web to have some ideas, and test some codes to see how it might improve the AI.
I had some problems with installing some libraries, and had some memory problems as the program took a lot of memory.
I had some ram memory problems due to the large number of images in smoke elimination. I have used Colab and my machine to test some part of program, like training the model, in order to make it train faster than Colab.
Colab is quite useful to run parts of the code in different cells, which avoids to run all the code multiple times, if some part of the code has some errors, and just fixing the cell who had errors.

I tried to work on smoke elimination, but also tried to do the speed detection, as I had been having troubles with the smoke elimination challenge, and I wanted to at least manage to improve a challenge. In the end, both challenges were improved compared to the baseline submission.

For the smoke elimination, a GAN(Generative Adversarial Network) seems a good idea to transform the images with some smoke into images without any smoke, with the generator part doing a synthetic image, close to the clear image.

It might gain a better score than the given baseline, since the baseline does not change the test images, it returns them unchanged.

The main idea is to train an AI to transform an image with smoke into a clear image, by learning to associate a smoke image to the same image without the smoke, and trying to generate a clear image from an image with smoke.
I took the code from that site: https://machinelearningmastery.com/how-to-develop-a-pix2pix-gan-for-image-to-image-translation/ for the smoke elimination, and it does work, unfortunately the training part takes a very long time, thankfully the program saves a model after 10 epochs, which gave me the possibility to test the various models, and find one that would posses a better score. Also I took a small part of the training data as it takes a lot of ram memory, and the increase of training data makes the training and epoch loop last a very long time.

In fact, on the 22th of May, the training loop has been going on for about more 48 hours, and it was not yet finished. Thankfully a combination of using my computer for training, and of using Colab online for predicting the test set, allows me to try the models made after each 10 epochs of training. After testing a few models, the model made after 40 epochs finally begins to improve the score a little bit, and it seems that the stabilises around 70 epochs.

This result shows that the model did seem to require more training time/loop to better perform the task of removing the smoke in the picture, and then generating a similar image without the smoke. After testing a few more models, it seems the score is improving very slowly, due to the lack of training date.
A possible way for score improvement is to load more training data and add it into the model, which would make the loading of data heavier for the ram memory, and would make the training part even longer.

After using more of the training data, a quarter of the data,(10000), only after one epoch, the result was much better with a MSE of 36.882, indicating that an increase of the data might help improve the score, but not always, as one model who had 5 epoch worsened the score.

Also the training and the testing part were much quicker, as I managed to use the GPU instead of the CPU, by following a guide showing what I needed to installed allowing the recognition and the use of GPU. The guide: https://www.tensorflow.org/install/gpu.

For the speed detection challenge, https://www.aicrowd.com/challenges/ai-blitz-8/problems/f1-speed-recognition, at first, i tried to grayscale and it seems that grayscaling does not improve the score, but worsens it.

I tested difference models to see if it impacts the performance of the training and might help improve the score.
The squeezenet1_1 gives very bad results for the speed recognition, whereas the resnet18 improves the score a lot.
It shows how some models can solve more efficiently some problems compared to others models, whereas with another problem, the same model could do poorly.
The resnet34 improves a little bit the score, but the major downside is that it's take a lot of time to train/compute it, due to the complexity of the model.
So the choice of the models depends on the improving the score while having an acceptable training time, not to long.

I looked most of the time for online information, while trying some parts of code and trying to understand how it works, in order to see if I could use it to improve the program and have a better score.

For fun, i tried, submitted and modified the baseline for each challenges.
For F1 team classification and F1 car rotation, i also changed the model like in F1 speed recognition, using the resenet18 model.
For F1 car detection, I increased the iterations and decrease the base learning rate. The best submission was an iteration of 5000 and base learning rate of 0.00005.

All my submissions: https://www.aicrowd.com/challenges/ai-blitz-8/submissions?q%5Bparticipant_name_equals%5D=MichaelMosimann.

Observations on the participants of the AIcrowd.

At the beginnings, there was not much discussion between participants in the discussion tabs of the AI Blitz challenges. It was mostly bug reports, like about submission limits.

It seems that the participants weren't discussing with each others during the challenge, as only one has giving some help to possibly improve the score on the 14th of May. The hint was to sort by

index, since the resulting file isn't sorted, and after testing it, I saw that it had indeed improved the score of the file.
The user is called Victorkras2008, and here is the page: https://discourse.aicrowd.com/t/you-can-improve-your-results-if/5627. For this challenge, Victorkras2008 has been the most active in the discussions.

Looking at the leaderboard, it's interesting to note that the people holding the top places, hold the first place on most of all the challenges in the blitz, and in the case they don't hold the first place, they remain in the top of the leaderboard.
It shows that the challenge and AICrowd attract some good AI programmers, as some of the participants are really good at improving the score, and that some already managed to get the top score on the day the challenge began, the 4th of May.
With these observations, it leads me to think that most participants are more in a competitive state than a helpful mindset, trying to win the reward or have the best score.

The insight page shows that the curve of submissions during the challenge, and reveals there was at least some daily submissions. It's interesting to note that a participant tried to submitting even after the challenge ended, on the 26th and 27th of May.

I checked the AIcrowd discord to see if the participants used this to help each other. For this challenge, the AIBlitz room was used to complain about the baseline(the F1 Team Classification) giving a perfect score for the first challenge, and it seems the participants didn't help each other for this Blitz challenge on Discord, although the discussions in the some of the other chat rooms show that some participants do chat with each others, and help each other.
The other challenges may be more active, and more interesting for the others participants, as they may yield different rewards and challenges to work on.

The participants of AIcrowd can use the https://discourse.aicrowd.com/ page or discord to chat, https://discord.com/invite/5Q7Tfww3tk, either with the other participants or staff members. About 1411 members on Discord.

On the 16th of May 2021, there was 123 participants and 12 teams. On the 19th of May, there was 144 participants and 13 teams. On the 20th of May, at 10:49 am, there was 150 participants and 14 teams, and no new discussion and notebooks in the blitz page. On the 21st of May, at 10:25 am, the number of participants was 152. On the 22nd of May, there was 154 participants.

On the 24th of May, they were 158 participants in this blitz challenge.
Even after the challenge ended, a few people joined the challenge and on the 16th of June, there was 168 participants.

Some challengers made some notebooks about their codes, after the challenge has ended, expect for one who posted a few days before the challenge ended. Most of them is about the smoke elimination and one is for the speed recognition.

On the 28th of May, Aicrowd sent a mail announcing the winners of the AI blitz 8 of both categories, which were leaderboard topper and community contributor. The winner of leaderboard topper was Denis_tsaregorodtsev and the winning team was Mercedes_AMG_Petronas_F1_team. For the community contributors, the winners were derinformatiker and devesh_darshan.
Derinformatiker made a topics on the discussion tab of the challenge to indicate that his solutions are on Github.

The AICrowd members are planning a new AI blitz 9 about natural language processing, and they have also a survey about the AI blitz 8 challenge to help improve future AI blitz challenges.

Around the 8-9th of June, the new AI blitz has been launched, and as of the 16th of June, there is 162 participants and 16 teams, more than the start of the AI blitz 8, which shows the new challenge is more interesting for these participants.
There is also a few notebooks from participants, eager to help the other participants.

A big thank you for this page: https://machinelearningmastery.com/how-to-develop-a-pix2pix-gan-for-image-to-image-translation/.

Colab for smoke elimination:
https://colab.research.google.com/drive/1Ng0iGJMIAr6RKKgg4SojRYk4EsweozAv#scrollTo=vKvmjIFzuqBd.