VADOT Dylan

Presentation Semestral Project

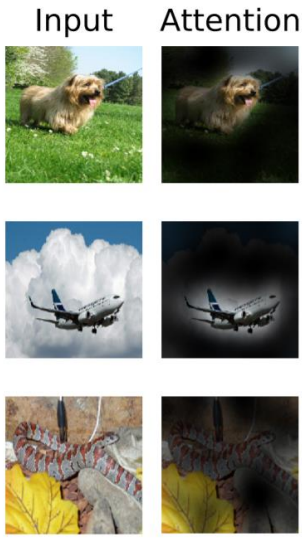
Image search and understanding

The purpose of the subject is to get a better understanding of images. It consists for example in determining important objects, ignoring background etc.

There are many application examples. It can be used in fashion for example, someone can take a picture of somebody wearing given clothes and find similar pieces of clothing in the catalog.

Machine learning offers a lot of new possibilities in image processing thanks to neural networks based on convolution (Convolutional Neuron Networks) or thanks to the principle of attention that allow to enhance some parts of the input data while diminishing other parts (Recurrent Neural Network, Transformers…).

We decided to concentrate on the second option. Attention mechanism has the benefit to add context to elements.



It is currently used a lot in Natural Language Processing (translation, auto-prediction, question answering…) and represent the state of the art in this field. In image it is not used a lot yet. We used a transformers encoder. Generally, in NLP the input of the transformers is a sequence of words that constitute a phrase. Here, we split an image into patches of the same size (like we split a phrase into each word).

We worked on 32x32 images and split the images into a sequence of 2x2 pixels. As the image is treated as a sequence, the order of the patches doesn't have any impact. To remediate to this, we had a position embedding so the position can matter.

Because of a memory limit, we limit our project to a classification between 10 classes.

We used the pre-existent database cifar10 composed by 50000 training images and 10000 test images divided into these 10 classes: airplane automobile bird cat deer dog frog horse ship truck.

As input of the transformer we put the sequence of patches with the position embedding. Several layers (Self-Attention Layer, Normalization Layer, Add Layer) are applied several times to these inputs. Then we use a Multi-Layer Perceptron to classify the input image in one of the 10 classes.

The results obtained were not very good. In the best case, we got an accuracy rate of 27.3%. These low results can be explained by the fact we were not able to train the model on many images because of computational time. Moreover, the parameters used (especially the size of the stack of transformer encoders) were not big enough but higher parameters were not supported by the device memory.