Report

Classifying images with and without tomato(es)

# Metric

I used F1-score as the metric. The reason is, both train and test dataset are extremly imbalanced (ratio ~ 95%:5%). F1-score is indeed better than accuracy score in these datasets.

# Algorithm

I used ConvNet (or CNN, depends on your style) to classify the images.

It is difficult to not know about CNN during last years, especially after the milestone paper in CVPR2012 which presented CNN to classify Google Image. It seems to me that today CNN is a de-facto in computer vision.

On the other hand, the dicussing problem is a binary – classification. It is a traditional one, hence there are so many tutorials, libraries, and even ready-code for them.

# Parameters

Parameters-tuning is always a painful task in deep learing. As far as I can concern, it should be considered as a “dark-art”. The best way we can do (in Jan 2017) is to try different combinations to see what works best[[1]](#footnote-1).

However, some things might help:

* Increase the number of layers. On the other hand, it will eat all of your computational resouces. Prepare a (or multi) good GPU(s).
* Increase each layer size. In my experience, it does not help as much as the above trick.
* Dropout. Very simple. Very handy. The recent study [1] stated that, deep models are very redundant, and dropout can indeed reduce the over-fitting.
* Add regulizations.
* Other hyper-parameters are more or less able to be random-choiced.

# Result

# Reference:

[1] Molchanov, D., Ashukha, A. and Vetrov, D., 2017. Variational Dropout Sparsifies Deep Neural Networks. arXiv preprint arXiv:1701.05369.

1. One can argue that, there are again, different ways to try different combinations, such as using GridSearch or RandomSearch. However, to me their ideas are the same: you just try, and wish for your luck. There is no guarantee that you will find the best parameter sets, or even a set which is close to the best. [↑](#footnote-ref-1)