Analysis on Human Protein Atlas Single Cell Classification Competition

Sadia Afrin

Goal of the Competition:

This competition asks for classification of single cell based on the pattern of protein location inside the cell. The pattern and location of protein is different in individual cells. These differences can help researchers to identify heterogeneity among cells. This is important as, proteins in human body are responsible for nearly every cellular task. They get signal from outside of the cells and provide response according to that signal. Therefore, identifying the pattern and location of protein can give rise to the understanding of how cells function, how disease develop and apparently better treatment for those disease.

Description on Dataset:

The training.csv file includes image id and label column which basically represents the class for each image. Now analyzing some sample submission, we can say that, for each image they have asked for three outputs; the class of each cell, confidence of the class prediction and segmentation mask for each cell. All three information will be included in the final prediction string. Beside this, for each image ID, there are 4 images in the training folder which corresponds to Red (Microtubules), Blue (Nucleus), Green (Protein of interest) and Yellow (Endoplasmic reticulum) channels. The channels can be blend into one and displayed in a single image for the simplification. In addition to these data, there are available images in Human Protein Atlas database. Previously, in a different competition this database was used to classify whole image [1].

Prediction Label:

There are 19 unique labels for images starting from 0 to 18. After making a chart on images with their respective labels, I found that the dataset is quite imbalanced. There is a description available in the public database that provides insights on which labels represents what. For example, 0 label represents Nucleoplasm and 1 represents nuclear membrane. No image category falls under label 18. That’s why it is defined as negative. The predicted labels apply to the cell where green color exists as green is the protein. For example, if there are 4 cells and 3 have green colors and image level-labels are Mitochondria, and Nucleoplasm. For cell 1, if green looks like to be Mitochondria the cell label is Mitochondria. For cell 2, if green looks like Nucleoplasm, then the label indicates Nucleoplasm. If cell 4 doesn’t have green, it is labeled as Negative.

Solution for this competition:

As input for this experiment, we need images of individual cells but for that we don’t need all the images. Therefore, one participant created a sample from the training set. The benefit of creating this sample dataset is, it is more balanced than the given dataset. Then he used fastai, a Layered API for Deep Learning, to prepare the data and to come up with default set of augmentation. One of the benefits of this API is that it easily splits the training and validation set. For submission, he pre-processed the public images the same way as prototyping dataset [2]. Another participant, obtaining best score, on the other hand used Multi-label stratification for splitting. Though, he used fastai API for item and batch transform [3]. Beside this, I found a Pytorch based approach. Unlike others, this participant used a subset of training data instead of creating any sample dataset. To train the dataset, he used resnet framework. One mentionable problem with this approach is its training time. It took almost 5 hours to train the model [4].

Paper or Article:

This is an ongoing competition which will end after two months. Therefore, the competitors and organizers have not published any work yet.

References:

[1] Ouyang, W., Winsnes, C.F., Hjelmare, M. *et al.* Analysis of the Human Protein Atlas Image Classification competition. *Nat Methods* **16,** 1254–1261 (2019). https://doi.org/10.1038/s41592-019-0658-6

[2] https://www.kaggle.com/thedrcat/fastai-quick-submission-template

[3] https://www.kaggle.com/dragonzhang/fastai-cell-tile-prototyping-training

[4] https://www.kaggle.com/ateplyuk/hpa-pytorch-starter-code