Meta learning for scene-text-recognition on new languages

DL4CV final project by: Bar Karov ,Shiri Moshe and Yonatan Sverdlov

Introduction

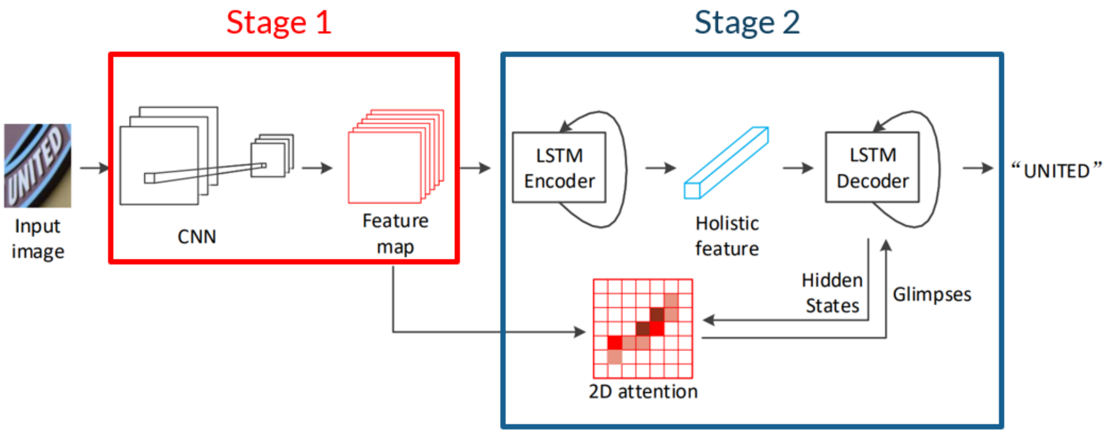
While text recognition from scanned documents is largely solved, text recognition in natural images (known as scene text recognition) remains a challenging problem with many practical applications. In natural images, text may appear over complex backgrounds, and in many different fonts and angles. Therefore, modern neural networks for STR (scene text recognition) often require millions of samples to achieve state-of-the-art results.

Due to this requirement for large datasets, most literature focuses on text recognition in widely spoken languages, mostly English and Mandarin. In our project, we attempt to utilize meta learning to adapt an existing STR neural network to the recognition of new languages, using a relatively small dataset. To perform this task, we take an existing STR neural network, adapt it to handle different languages, and attempt to find smart initial weights for the recognition of new languages. These smart initial weights would ideally require fewer weight updates (meaning fewer samples) to adapt to any new language. These initial weights are calculated using the REPTILE meta-learning algorithm[2], which calculates them by meta-training the network over a number of different languages.

The underlying assumption of this meta-learning algorithm is that STR networks for the recognition of different languages share some intrinsic shared knowledge in their weights. This intrinsic knowledge can be learned by a meta-learning algorithm, and embedded into smart initial weights for the base network.

Method

To find these smart initial weights, we need a meta-learning algorithm and a base network for which these weights would apply. Our chosen base network is called "Show, attend, and Read" [1], or SAR in short. This network is comprised of two stages. The first performs feature extraction, and for our purposes remains constant for the entire process (the weights for this part do not change). Extracting features for text recognition is assumed to be independent from the language of the text, making meta-learning for this stage redundant. The second stage begins with LSTM layers which encode the feature map outputted by stage 1. When this process is done, another LSTM layer decodes the sequence outputted by the first LSTM module into a sequence of characters. Additionally, an attention mechanism is used to focus the network on the location where the next character is likely to appear, while communicating with the LSTM based decoder:



SAR network architecture. Meta learning is performed only on stage 2.

To learn “smart” initial weights for the 2nd stage of this network, we used the REPTILE algorithm and STR datasets in multiple languages (English, Italian, French, German, Hindi, Bangla, Arabic and images that contain only punctuations). The REPTILE algorithm Initializes some weights θ, and trains the network from these θ iteratively, each time on a different language. The training on each language slightly moves θ in the direction of the ideal weights found for that language. Eventually, these θ are outputted as the meta-learned initial weights. Using these weights, fewer samples would be needed to train the network on new, previously unseen languages (languages which haven’t been used during meta-learning). A visual explanation of this algorithm is available in the attached video.

Results

To measure the success of the meta-learning process, we compare the performance of the base network (SAR) under two conditions- when training it on a new language from standard initialized weights, and when training from the meta learned initial weights. When meta-training to find these initial weights, we do not use the language that the network is eventually trained and tested on. In both cases (initializations), we train the network using the same dataset and same hyper-parameters.

The meta-learning process is considered successful if it leads to a significant improvement in performance over the new language, compared to the performance achieved when training from standard initialized weights. To measure the performance of the network, we measured the normalized edit-distance between the network’s prediction and the label over a test-set of images. Since fewer required edits between predictions and labels are an indication of better performance, a lower value here is better:

Initialization Language	Standard	Meta Learned
English	0.94	0.88
Italian	0.93	0.84

As the table shows, meta learning achieved a slight improvement in performance. However, it appears that with both initializations, performance was relatively poor. This is likely due to the complexity of the base network, which required millions of samples to reach state of the art results. In our experiments we used less than 100,000. With the hardware that’s available to us, training the base network with the dataset used in its original article would take approximately 2 days, even without meta learning. In most applications of the REPTILE algorithm for other problems, the meta-training stage takes significantly longer than the final training stage.

When further examining the network’s predictions, it appears that meta learning helped mostly with recognizing the length of the labels. This could be explained by the fact that predicting a word’s length from an image is mostly independent from the language at hand, making meta learning effective for this task.

In summary, it appears that meta learning helps achieve some very initial learning that wasn’t possible without meta learning. Despite this, due to the low level of learning in both initializations, it’s not immediately clear if this result could be generalized for larger datasets, which would be required to train the base network to a level where it could be practically useful.

Summary

In our project, we attempted to enable STR on new languages using smaller datasets by utilizing meta-learning. While it appears that meta-learning slightly improved performance, more resources (both data and hardware) are needed to conclude whether meta-learning could be effective for this task.

Bibliography

[1] - Li, Hui, et al. "Show, attend and read: A simple and strong baseline for irregular text recognition." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 33. No. 01. 2019.

[2] - Nichol, Alex, Joshua Achiam, and John Schulman. "On first-order meta-learning algorithms." arXiv preprint arXiv:1803.02999 (2018).