Machine Learning in the Cloud

Thomas Martin

21st April 2019

INM432 Big Data Coursework Part 2

https://github.com/tpgmartin/inm432_cw2

Introduction

This short coursework is designed as an exercise to learn how to use the Google Cloud Platform CloudML framework for machine learning tasks. The coursework is split into three main tasks,

- Task 1 - running with larger dataset: Running the provided source code using CloudML on two datasets
- Task 2 - modifying server & cluster configurations: Modifying the original source code to run with a custom GPU configuration
- Task 3 - dropout: Exploring the effect of dropout on training performance

The data examined in this coursework are from a flowers and coastlines dataset. Both datasets contain image files with classes. However, the number of rows is much larger in the coastlines dataset compared to the flowers dataset: 10,533 individual rows to 3,670 respectively. The number of classes also differs between the two datasets: The coastlines dataset contains 18 classes, compared to 5 in the flowers dataset.

To compare between the model performance on the two datasets, and between different configurations of the model, the following metrics will be considered,

- Accuracy - this is the proportion of true positives out of the total number of predictions for a given dataset
- Loss - this is calculated using Softmax function on the output layer
- Compute time - this is taken as relative time of each training/testing interval

Results for each task of the coursework are reproduced in tables below, as well as original files and downloads in the linked repo. Full results and trained models can be found at the bucket `gs://inm432-cw2-237509-ml/`, specific directories will be linked to below. All files used to train the models can be found in the directories `coastlines` and `flowers`. The contents of these directories are identical except for `sample.sh` and `test_train_split.py`.

Approach

This section outlines some preliminary steps required before training the model.

For each dataset and configuration, the Inception v3 model was used. In this approach, only the final layers are retrained for each novel configuration.

For the coastlines dataset a test train split was performed on the initial dataset taken from the publicly available bucket, using a ratio of 70:30 training to testing entries. The flowers dataset had already been split into training and testing sets.

Before training, preprocessing was performed on the image files, using the `./trainer/preprocess.py` file on both datasets. To save time, each dataset was only preprocessed the one time when trainig the initial model configuration. Subsequent runs then referenced the given bucket containing the preprocessed dataset as given by the `--eval_data_paths` and `--train_data_paths`. However this is not reflected in the script files given in this repo.

Task 1 - Running with Larger Dataset

In the repo `inm432_cw2`, the script run for each dataset can be found at `./coastlines/sample.sh` and `./flowers/sample.sh` for the coastlines and flowers dataset respectively. The full results are reproduced in the tables below. Full output can be found at,

- `gs://inm432-cw2-237509-ml/tpgmartin/flowers_tpgmartin_20190420_023029/`
- `gs://inm432-cw2-237509-ml/tpgmartin/coastlines_tpgmartin_20190413_212534/`

Dataset	Training Set Accuracy	Test Set Accuracy
Flowers	100%	92.33%
Coastlines	84.67%	72.67%

Dataset	Training Set Loss	Test Set Loss
Flowers	0.0054	0.33
Coastlines	0.5588	0.8442

Dataset	Training Set Time	Test Set Time
Flowers	17m 13s	17m 10s
Coastlines	3m 54s	3m 51s

Accuracy and loss are consistently higher for the flowers dataset compared to the coastlines data. This is likely a consequence of the much larger number of classes in the coastlines dataset compared to the flowers dataset, although it may also suggest that the former dataset is simply more difficult to classify due to the image files considered. In either case the model performance indicates that both models could train well without underfitting/overfitting.

It's likely that the compute time for the flowers dataset is erroneous for this specific run as all other runs are around the four minute mark.

Task 2 - Modifying Server & Cluster Configurations

For this task a config file was passed as a command line argument to the ml-engine task, following this guide. Both datasets use the same config file, with multiple workers, most of which are GPUs. Full output can be found at,

- `gs://inm432-cw2-237509-ml/tpgmartin/flowers_tpgmartin_20190420_025457/`
- `gs://inm432-cw2-237509-ml/tpgmartin/coastlines_tpgmartin_20190420_012604/`

Dataset	Training Set Accuracy	Test Set Accuracy
Flowers	100%	91.33%
Coastlines	79.33%	71.0%

Dataset	Training Set Loss	Test Set Loss
Flowers	0.0056	0.3546
Coastlines	0.7011	0.902

Dataset	Training Set Time	Test Set Time
Flowers	3m 48s	3m 45s
Coastlines	2m 52s	3m 25s

In both cases we observe a slight decrease in compute time, and slight decrease in overall accuracy. This is likely due to the use of multiple workers: The shorter compute time during training due to higher bandwidth, but lower accuracy due to pooling of workers. This effect is most noticeable for the coastlines dataset.

Task 3 - Dropout

For this final task, an additional dropout coefficient was passed to the the previous GPU configuration file, with all other parameters remaining the same. The dropout is set to 0.5, giving the probability that neurons in the final layer are switched on or off. Full output can be found at,

- `gs://inm432-cw2-237509-ml/tpgmartin/flowers_tpgmartin_20190420_030752/`
- `gs://inm432-cw2-237509-ml/tpgmartin/coastlines_tpgmartin_20190420_015004/`

Dataset	Training Set Accuracy	Test Set Accuracy
Flowers	100%	90.67%
Coastlines	81.33%	71.67%

Dataset	Training Set Loss	Test Set Loss
Flowers	0.0097	0.3623
Coastlines	0.6233	0.8708

Dataset	Training Set Time	Test Set Time
Flowers	4m 1s	3m 57s
Coastlines	3m 44s	3m 40s

Compared to task 2, we observe a decrease in overall performance against the flowers dataset, but an increase in performance against the coastlines dataset. However, in either case the changes are not dramatic.

Discussion & Conclusion

This project aimed to investigate the use of the Google Cloud Platform CloudML framework for machine learning tasks for two datasets. The two datasets contained image files and mapped to a classification task. The datasets differed in total number of entries, and more dramatically, in the number of classes. For all configurations the performance as given by the metrics of the models during training were pretty similar, although the initial model configuration produced the best performance overall. This would indicate that at least for the datasets considered, the model used was quite robust. It is likely that the change in performance would be more dramatic if using larger and more complex datasets.

For the flowers dataset, which demonstrated 100% training accuracy, it is likely that the remedy to improve testing performance is to use more data, regularisation, or to perform more hyperparamter tuning. For the coastlines dataset, which demonstrating a training accuracy of ~80% and lower testing accuracy, the next course of action would be to train a bigger model or use better optimisation algorithms.

No hyperparamter tuning was performed at the configuration file level, which would be an interesting extension to the project.