

Evaluación Redes Neuronales

Se tiene el proyecto dedicado a la poda de mapas de características en redes neuronales que se adjunta.

La clase “LenetPruner” es una clase especializada en la poda de la red neuronal Lenet dedicada a la clasificación de imágenes de dígitos que describe la clase “Lenet”. Esta red está compuesta por 2 capas de convolución con bias y 2 capas “fully connected”, FC de ahora en adelante. La primera convolución tiene 32 mapas de características, la segunda 64, la primera FC 1024 y la segunda FC 10.

La poda se realiza iterativamente, un mapa de característica por iteración y **sin re-entrenamiento** entre iteraciones.

La clase “LenetPruner” tiene como función virtual el método “select_fmap_to_prune(self, sess)”. Tal y como se documenta en el código, esta función es la responsable de seleccionar qué mapa de características se poda en cada iteración. A modo de ejemplo, se proveen dos criterios diferentes: una selección aleatoria en la clase “RandomPruner” y una selección en base a los pesos de cada convolución en “MinWeightPruner”.

Se pide:

Siguiendo el paper [Pruning Convolutional Neural Networks for Resource Efficient Inference](#):

- 1) Implemente el criterio de Taylor de la ecuación (8) con y sin normalización L2.
- 2) Implemente una poda guiada por los FLOPs de cada capa según la ecuación (9) para un valor de $\lambda = 0.001$.

Entregable:

Código fuente implementado.

Puede seguir la estructura de los ejemplos, es decir, heredando de “LenetPruner” y sobrecargando la función de selección de mapas.

Un ejemplo de la ejecución puede ser el siguiente:

11 de Enero de 2018

```
[4911] 2018-01-11 11:55:45,695 INFO Taylor| Test accuracy 0.98539972, pruned: 923
[4911] 2018-01-11 11:55:46,115 INFO Taylor| Test accuracy 0.98499972, pruned: 924
[4911] 2018-01-11 11:55:46,507 INFO Taylor| Test accuracy 0.98419976, pruned: 925
[4911] 2018-01-11 11:55:46,890 INFO Taylor| Test accuracy 0.98439974, pruned: 926
[4911] 2018-01-11 11:55:47,327 INFO Taylor| Test accuracy 0.98419970, pruned: 927
[4911] 2018-01-11 11:55:47,762 INFO Taylor| Test accuracy 0.98379970, pruned: 928
[4911] 2018-01-11 11:55:48,202 INFO Taylor| Test accuracy 0.98439980, pruned: 929
[4911] 2018-01-11 11:55:48,588 INFO Taylor| Test accuracy 0.98459971, pruned: 930
[4911] 2018-01-11 11:55:48,992 INFO Taylor| Test accuracy 0.98519969, pruned: 931
[4911] 2018-01-11 11:55:49,342 INFO Taylor| Test accuracy 0.98479974, pruned: 932
[4911] 2018-01-11 11:55:49,707 INFO Taylor| Test accuracy 0.98499978, pruned: 933
[4911] 2018-01-11 11:55:50,111 INFO Taylor| Test accuracy 0.98499972, pruned: 934
[4911] 2018-01-11 11:55:50,476 INFO Taylor| Test accuracy 0.98419976, pruned: 935
[4911] 2018-01-11 11:55:50,828 INFO Taylor| Test accuracy 0.98359966, pruned: 936
[4911] 2018-01-11 11:55:51,173 INFO Taylor| Test accuracy 0.98359972, pruned: 937
[4911] 2018-01-11 11:55:51,576 INFO Taylor| Test accuracy 0.98379976, pruned: 938
[4911] 2018-01-11 11:55:51,939 INFO Taylor| Test accuracy 0.98479968, pruned: 939
[4911] 2018-01-11 11:55:52,285 INFO Taylor| Test accuracy 0.98479974, pruned: 940
[4911] 2018-01-11 11:55:52,711 INFO Taylor| Test accuracy 0.98499966, pruned: 941
```

Tenga en cuenta que los resultados de precisión pueden diferir.

NOTAS:

- Junto con el código fuente también se proveen los pesos ya entrenados de la red.
- Dado que no es necesario entrenar una red neuronal para realizar la prueba, es recomendable que realice una instalación en CPU pues es más fácil y rápida que la instalación en GPU de Tensorflow. El siguiente enlace explica cómo lograrlo [\[enlace\]](#).
- El código está en python 2.
- En la implementación actual, se usa el set completo de validación para validar la precisión. Si fuera un inconveniente utilizar el set completo por problemas de rendimiento, se puede modificar la clase "FeedDict" para que utilice un subconjunto de las imágenes de validación.
- Siguiendo las instrucciones del "README", para ejecutar una poda aleatoria se ha de ejecutar el comando "python prune_procedure.py --logdir logdir/ --layer_to_prune all".