

Experimento 1:

Propuesta de mejoras del estudio “*A Preliminary Study on Deep Transfer Learning Applied to Image Classification for Small Datasets*”. Parte 1: Hiper-parámetros

Índice

Experimento 1	3
Introducción	3
Análisis de tamaño de lotes	3
Análisis del número de épocas	5
Aplicación del conocimiento	7
Conclusión	9
Bibliografía	10

Experimento 1

Introducción

En el experimento 0, trabajamos con la red SOCO, esta red se basa en la arquitectura propuesta en el paper “A Preliminary Study on Deep Transfer Learning Applied to Image Classification for Small Datasets”[1].

Durante el experimento, se detectaron algunos puntos dudosos como la selección de hiper-parámetros. En este experimento pretendemos hacer una crítica formal a esta.

Como comentamos en el experimento, los principales hiper-parámetros a criticar son:

- Tamaño del lote.
- Número de épocas.
- Tamaño de imágenes.

Análisis de tamaño de lotes

Para esta parte del experimento nos basamos en estudios previos sobre el tamaño del lote en relación con el ratio de aprendizaje [2]. Este estudio recomienda trabajar con ratios de aprendizaje pequeños (entorno a 0.0001) y tamaños de lotes pequeños (recomiendan 32 como valor por defecto).

En el estudio a criticar, el tamaño de lote seleccionado es de 128. Este valor cumple una de las recomendaciones generales cuando se trabaja con CPU y es usar un valor en base 2. No obstante, desoye recomendaciones sobre la relación de ratio de aprendizaje de optimizadores y tamaño de lotes.

Se plantea por tanto la siguiente hipótesis : “Reducir el batch-size puede incrementar la velocidad de convergencia y la precisión del modelo”

Para comprobar si se cumple esta hipótesis y saber cual es el tamaño del lote óptimo, realizaremos cuatro pruebas con distintos tamaños. Se usará la misma arquitectura y optimizador. Se realizarán 10 iteraciones con cada tamaño de forma que los resultados sean robustos.

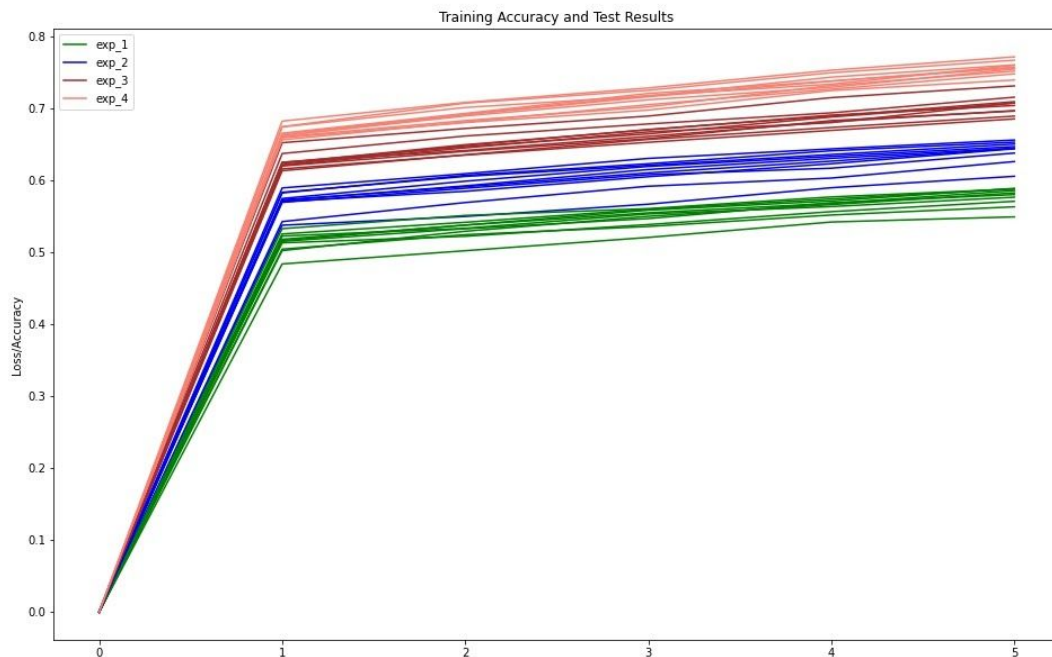
El esquema de pruebas será el siguiente:

- exp_1: Tamaño del lote = 128
- exp_2: Tamaño del lote = 64
- exp_3: Tamaño del lote = 32
- exp_4: Tamaño del lote = 16

El esquema de entrenamiento que ha seguido ha sido el propuesto en el estudio criticado:

- El conjunto de entrenamiento = source (60% del total del dataset),
- El conjunto de reentrenamiento = train (70% del 40% restante del dataset),
- El conjunto de test = test (30% del 40% restante del dataset)

En la siguiente imagen podemos observar el proceso de entrenamiento de la red y cómo afecta el tamaño del lote.



Si nos fijamos en la imagen superior vemos que la rapidez con la que converge el modelo es inversamente proporcional al tamaño del lote.

En la siguiente tabla podemos ver los resultados una vez han sido evaluados con el subconjunto de test.

Iteración	Caso 1	Caso 2	Caso 3	Caso 4
1	0.478565	0.496063	0.556430	0.720618
2	0.496063	0.462234	0.605716	0.762613
3	0.512394	0.619131	0.579178	0.624380
4	0.429571	0.440653	0.652960	0.736658
5	0.419073	0.552056	0.714494	0.748906
6	0.468650	0.543015	0.686206	0.695246
7	0.403325	0.543015	0.513561	0.569554
8	0.482065	0.596967	0.643628	0.682415
9	0.426947	0.559055	0.689122	0.710994
10	0.417614	0.547098	0.668416	0.685331
Media	0.453427	0.535929	0.630971	0.693672

Como podemos ver en la tabla anterior los resultados son abrumadores. Para cada iteración el mejor resultado lo obtiene la red entrenada con un tamaño de lote de 16. La mejora con respecto a la configuración propuesta en el paper es de un 52%.

Análisis del número de épocas

La hipótesis de esta parte del experimento es : “El número de épocas recomendadas en el artículo puede ser insuficiente para que el modelo tenga un desempeño óptimo al ser nuestro conjunto de datos más complejo”

Se han planteado un solo experimento. Los parámetros que usaremos serán:

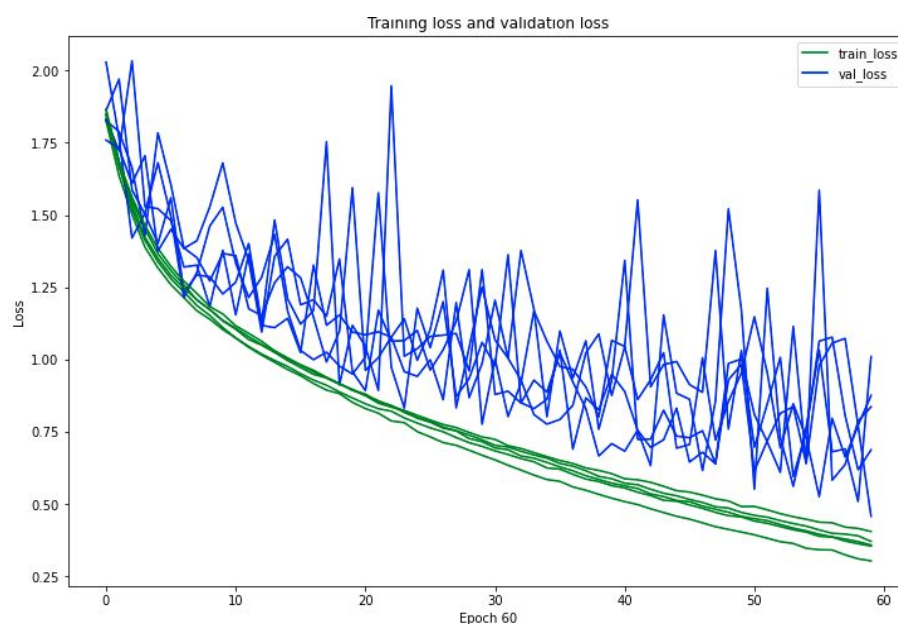
- Tanto el optimizador como el ratio de aprendizaje serán el recomendado en el estudio. (RSMProp con lr de 0.0001).
- El tamaño de lote será 128 (el recomendado por el estudio)
- El número de épocas será 30 tanto en el primer entrenamiento del modelo como en la fase de transfer learning.

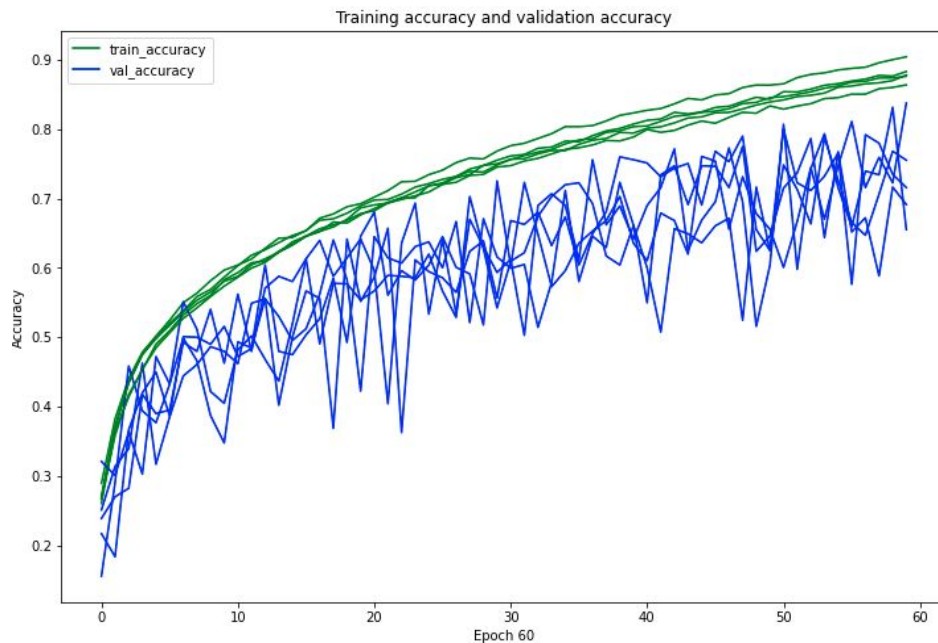
Se realizarán 5 iteraciones y el esquema de entrenamiento que se seguirá es el que mejor ha funcionado en el estudio anterior, es decir el caso 4. Tan solo introduciremos una pequeña variación y es que cojeremos el 30 % del conjunto de test como conjunto de validación. Hacemos esto para poder graficar el momento en el que la red comienza a sobre ajustarse.

El esquema de entrenamiento sería el siguiente:

- El conjunto de entrenamiento = source (60% del total del dataset),
- El conjunto de reentrenamiento = train (70% del 40% restante del dataset),
- El conjunto de test = test (21% del 40% restante del dataset)
- El conjunto de validación = validación (9% del 40% restante del dataset)

A continuación mostraremos los resultados obtenidos.





En la primera imagen podemos ver la pérdida, mientras que en la segunda podemos ver la precisión tanto en validación como en entrenamiento.

Estas dos imágenes nos dan una visión global sobre el comportamiento del modelo a lo largo de muchas iteraciones. Como podemos observar el modelo comienza a sobre ajustarse en torno a la época 30, pero tampoco encontramos unos resultados catastróficos.

Por otra parte, a continuación se muestra una tabla con los resultados después de la evaluación.

Iteración	Precisión obtenida al evaluar el modelo
1	68.7372 %
2	75.2114 %
3	65.5585 %
4	69.2038 %
5	85.0685 %
Media	72.7559 %
Desviación estándar	0.4763

La media de la precisión es 72.76 %. El resultado medio obtenido con los parámetros indicados en el estudio son 43.53%. Lo que implica una mejora de un 29.3% o lo que es lo mismo ha aumentado en un 67% con respecto a la configuración propuesta.

Aplicación del conocimiento

En esta sección pretendemos medir el rendimiento de aplicar el conocimiento adquirido. Por un lado usaremos un tamaño de lote más bajo y usaremos un número de épocas mayor.

La hipótesis inicial es : ¿Al aplicar ambas técnicas (reducción de tamaño de lote e incremento de épocas) mejorará el modelo más que al aplicar una sola de ellas?

Para ello vamos a realizar un experimento que involucre ambas variables.

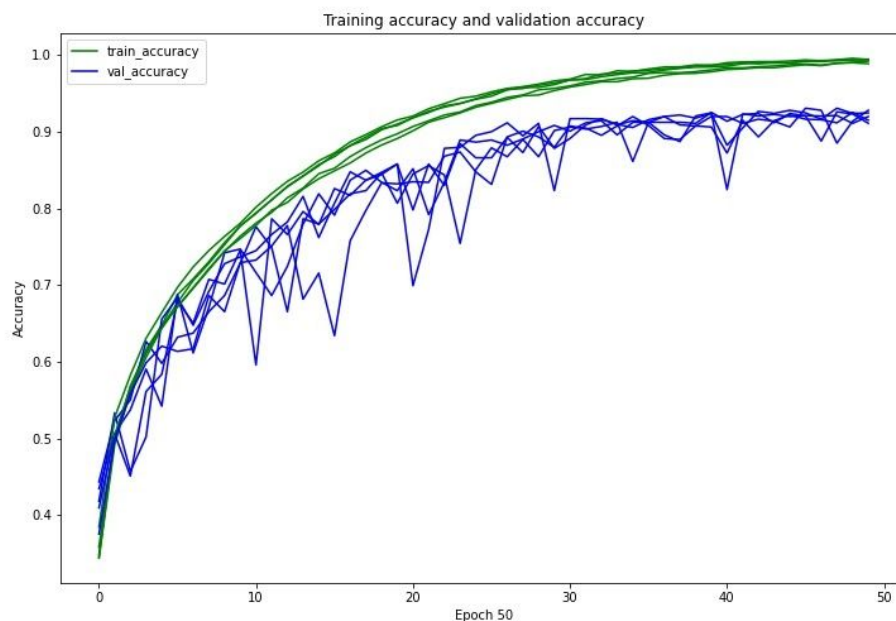
El esquema de entrenamiento será el mismo que en el apartado anterior:

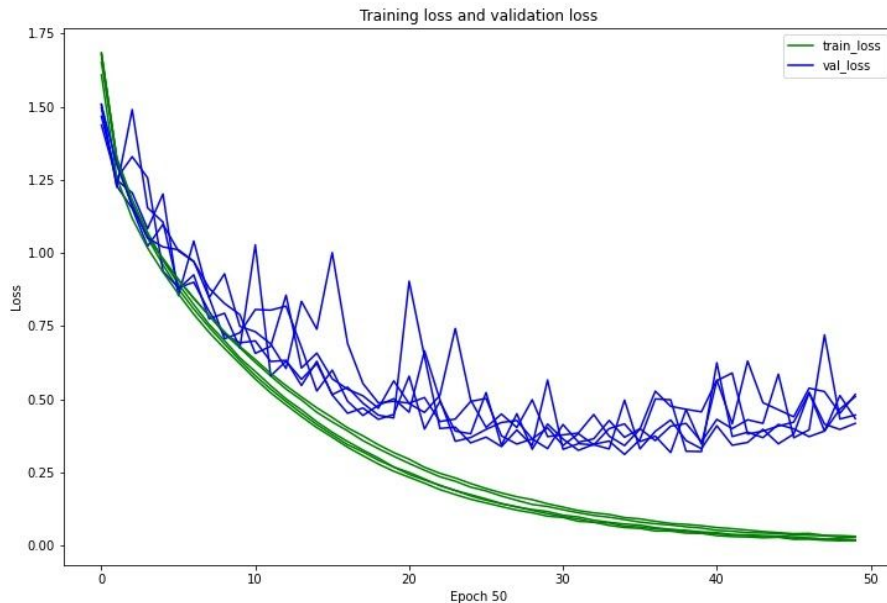
- El conjunto de entrenamiento = source (60% del total del dataset),
- El conjunto de reentrenamiento = train (70% del 40% restante del dataset),
- El conjunto de test = test (21% del 40% restante del dataset)
- El conjunto de validación = validación (9% del 40% restante del dataset)

Los parámetros que usaremos serán:

- Tanto el optimizador como el ratio de aprendizaje serán el recomendado en el estudio. (RSMProp con lr de 0.0001).
- El tamaño de lote será 16 (el recomendado por el estudio)
- El número de épocas será 25 tanto en el primer entrenamiento del modelo como en la fase de transfer learning.

Los resultados obtenidos son los siguientes:





Como podemos ver en la segunda imagen, la que hace referencia al valor de pérdida, se produce un sobreajuste serio. Aún así, en la primera imagen vemos que los resultados tienden a estabilizarse en torno al 90 por ciento de precisión sobre el set de validación.

A continuación trataremos los datos obtenidos al evaluar sobre el set de datos de test:

Iteración	Precisión obtenida al evaluar el modelo
1	93.4675 %
2	92.3010 %
3	92.7676 %
4	92.8259 %
5	91.5719 %
Media	92.5868 %
Desviación estándar	0.6288

Podemos observar que el resultado obtenido es considerablemente mejor que de cada mejora individual. El mejor resultado medio obtenido en la replicación del estudio fue de 43.53%. La precisión media de este experimento ha sido del 92.59%.

El porcentaje de mejora es del 112,7%, es decir, nuestra configuración de hiper-parámetros ha duplicado la precisión del modelo.

Conclusión

A lo largo de este estudio se han validado dos hipótesis:

1. Un tamaño de lote pequeño acelera la convergencia del modelo. Lo ideal es el uso de un valor entorno a 16, y es recomendable acompañarlo de un modelo con un ratio de aprendizaje bajo (0.0001).
2. El estudio del número de épocas que se va a estar entrenando el modelo es de importancia capital, y a menudo dependerá del conjunto de imágenes el tamaño necesario. En nuestro caso 25 épocas (recordemos que son 25 épocas para entrenar el modelo por scratching y 25 más aplicando transfer learning) parece generar cierto sobre ajuste.

Además de la hipótesis validadas, también se ha dado con un modelo con una precisión media del 92.59 % con una variación de 0.62 a lo largo de 5 ejemplos, lo que invita a pensar la robustez de este. El gran contra de este modelo es la tendencia al sobre ajuste, profundizaremos en algunas técnicas para evitar esto en la segunda parte del estudio.

Bibliografía

[1] - Molina M.Á., Asencio-Cortés G., Riquelme J.C., Martínez-Álvarez F. (2021) A Preliminary Study on Deep Transfer Learning Applied to Image Classification for Small Datasets. In: Herrero Á., Cambra C., Urda D., Sedano J., Quintián H., Corchado E. (eds) 15th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2020). SOCO 2020. Advances in Intelligent Systems and Computing, vol 1268. Springer, Cham. https://doi.org/10.1007/978-3-030-57802-2_71

[2] - The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. Ibrahem Kandel, Mauro Castelli. Nova Information Management School (NOVA IMS), Universidade Nova de Lisboa, Campus de Campolide, 1070-312, Lisbon, Portugal.