

Enunciado 3C (Jugando con los hiperparámetros y visualizando el error para guiar nuestro aprendizaje)

En esta tarea emplearéis Colab. En Drive cargaréis tanto el notebook como los datos. El path podría ser [/content/drive/MyDrive/ColabNotebooksHapLab/TIA-2024-2025/data/trees/](#)

Debería de haber 3 ficheros:

train.txt, dev.txt y test.txt

La tarea que tendréis que desempeñar es la de análisis de sentimientos. Primero es importante entender la entrada y la salida (las dimensiones de las mismas).

Váis a jugar con los hiperparámetros y se os solicitará que añadáis al informe el código final al que hayais llegado después de jugar y modificar los hiperparámetros y que expliquéis qué aporta vuestra solución y razonéis vuestra propuesta y sus mejoras.

Estrategias para evitar el overfitting

Drop out:

Desactivación de un porcentaje de neuronas al azar

Regularización:

Añadir a la función de error (loss function):

$$L1: \lambda \sum ||w_i||$$

$$L2: \lambda \sum ||w_i||^2$$

De forma que al aplicar el gradiente en la actualización de pesos, la corrección para todos será mayor, eso quiere decir que para aquellos pesos que sean pequeños la actualización con el gradiente los dejará en 0 y solo aquellos que hayan aportado mucho serán los que continúen teniendo un valor >0 . Actúa como una especie a feature selection.

Supongamos que w_1 sea 0.003 y su gradiente sea 0.0002 y que el L1 $\lambda \sum ||w_i||$ sea muy grande porque hay pesos muy grandes, pongamos que es 0.0002 eso anularía al peso w_1

Si por el contrario el peso w_1 hubiese sido 0.1, actualizarlo con su gradiente añadiéndole ese 0.0002 no le hubiese afectado tanto y seguirá actualizándose