

# Sistemas Inteligentes para la Gestión en la Empresa

## Práctica 2: Deep Learning para clasificación

Curso 2020-2021

### Objetivos y evaluación

En esta segunda práctica de la asignatura Sistemas Inteligentes para la Gestión en la Empresa estudiaremos cómo desarrollar un modelo de clasificación basado en redes neuronales profundas capaz de trabajar con datos multimodales.

La práctica consistirá en la resolución de un problema de aprendizaje utilizando técnicas de Deep Learning.

La práctica se desarrollará en grupos de dos personas (preferiblemente) o individualmente (previa consulta al profesor).

La calificación constituirá el 25% de la nota final de la asignatura (2.5 puntos).

Se evaluará la calidad de la memoria presentada y la elaboración de las soluciones propuestas. Se valorará especialmente la claridad en la redacción y el apoyo con materiales gráficos: diagramas, gráficas, etc.

La entrega se realizará a través de PRADO, en el enlace que se habilitará al efecto en la página de la asignatura.

### Descripción del problema

Se trabajará con el conjunto Fakeddit<sup>1</sup>, que contiene más de un millón de registros sobre publicaciones en Reddit que incluyen desinformación.

El problema de clasificación con Fakeddit consiste en predecir si una publicación es una "noticia falsa". Existen varias versiones del problema según las categorías de "falsedad", desde clasificación binaria a clasificación con seis clases.

Respecto a los datos para realizar la predicción, disponemos de tres fuentes: metadatos sobre la publicación, imágenes asociadas a la publicación y comentarios de la publicación. En esta práctica, se deberán utilizar al menos las imágenes para la clasificación.

---

<sup>1</sup> <https://github.com/entitize/Fakeddit>

## Departamento de Ciencias de la Computación e Inteligencia Artificial

### Datos

Los datos originales se encuentran disponibles en la web de los autores de Fakeddit, aunque se recomienda utilizar las copias publicadas en esta carpeta de Google Drive:

<https://drive.google.com/drive/folders/1qYWWdfdp-OAxKNXbKgAMh2x3p04X55TO>

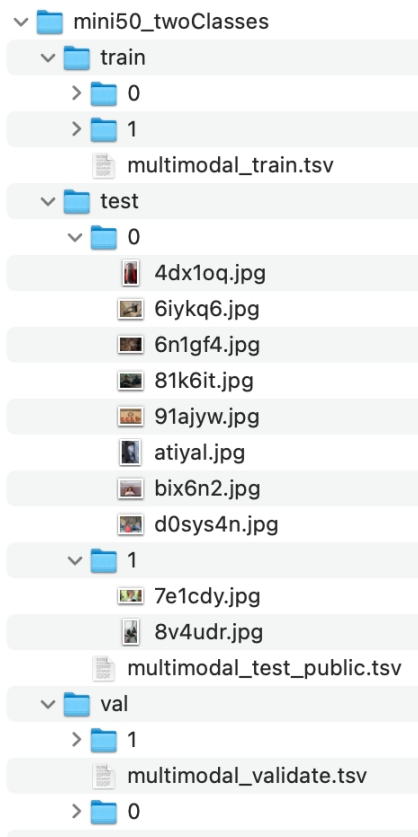
Estas copias incluyen solamente las noticias que tienen imágenes asociadas.

Se han realizado tres muestras de datos {mini, medium, all}, que contienen 50, 10.000 y todas las imágenes, respectivamente.

Para cada muestra de datos, se ha construido una versión del problema con dos clases (twoClasses) y seis clases (sixClasses).

En cada caso, se han tomado las imágenes y se han dividido en diferentes subcarpetas para entrenamiento, validación y test. El tamaño de validación y test para los problemas reducidos  $n = \{50, 10.000\}$  es  $0.2 \times n$ . Cada una de estas carpetas incluye un fichero .csv con los metadatos de todas las imágenes de esa carpeta. Las imágenes ya aparecen divididas según la etiqueta de salida.

Por ejemplo, el fichero *mini50\_twoClasses.zip* contiene la versión del problema con dos clases seleccionando 50 imágenes al azar para entrenamiento, 10 para validación y 10 para test, organizadas como se ve en la siguiente figura:



Los ficheros de metadatos .tsv contienen, entre otros campos, el identificador (*id*) de cada publicación, que se corresponde con el nombre de fichero de la imagen asociada.

## Departamento de Ciencias de la Computación e Inteligencia Artificial

Los comentarios de las imágenes no están segmentados: disponemos de un único fichero `all_comments.tsv.zip`. Los comentarios se pueden combinar con las publicaciones a través del campo `submission_id` ( $\rightarrow$  `id` de metadatos).

## Material

Se proporcionan varios *scripts* para comenzar el trabajo:

- Particionamiento de los datos (*split.R*): Sólo sería necesario en caso de querer realizar una partición diferente de los datos del problema.
- Exploración de metadatos, imágenes y comentarios (*intro-eda.Rmd*): Ejemplo simple de carga de imágenes y combinación de metadatos-comentarios.
- Clasificador inicial con CNNs (*intro-cnn.Rmd*): Ejemplo simple de clasificación con Keras R y redes convolutivas.

Todos estos ficheros están disponibles en el repositorio Github de la asignatura (`practicas/p2/starting-package.zip`).

## Tareas y evaluación

**Se entregará la solución del problema (código R) y una memoria descriptiva de las tareas realizadas.**

La memoria explicará qué tareas para construir el modelo de clasificación y mejorar su calidad se han llevado a cabo y con qué objetivo, así como los resultados obtenidos – en términos de la calidad del clasificador y del tiempo requerido para su entrenamiento.

Las tareas que se pueden llevar a cabo y su puntuación asociada (sobre 10) son las siguientes:

7 puntos: Resolución del problema de clasificación binaria con imágenes

- Clasificación binaria
- Utilización de subconjunto de 10.000 imágenes
- Análisis exploratorio
- Ajuste de hiperparámetros, topología de la red, función de coste y optimizador
- Aplicación y estudio de 2 técnicas para mejora del aprendizaje

+1 punto: Extensiones de la solución básica para multiclase

+1 punto: Uso de MLflow para gestión del proceso de aprendizaje

+1 punto: Por cada una de las siguientes ampliaciones:

- Análisis exploratorio avanzado
- Aprendizaje multimodal (imágenes + metadatos, imágenes + texto)
- Estrategias de binarización
- Ajuste automático de hiperparámetros (AutoML)
- Métodos de tipo *ensemble* e híbridos
- Transferencia de aprendizaje o *fine tuning*

La puntuación por encima de los 10 puntos de esta práctica se sumará a la nota global de la asignatura, hasta un máximo de 1.25 sobre la calificación final.



## Entrega

**Límite:** 8 de junio de 2020 a las 23:55

**Contenidos:** Un fichero .zip, incluyendo:

- Código en R (.R, .Rmd)
- Memoria
  - Portada: nombre de los autores, título
  - Índice
  - Contenidos
    - Fundamentos teóricos
    - Descripción de las técnicas empleadas
    - Discusión de resultados
    - Conclusiones
  - Bibliografía