PRACTICA 2: Clasificación de páginas web

La **clasificación automática de textos** es una de las tareas clásicas en el área del Procesamiento del Lenguaje Natural (PLN). Consiste en agrupar documentos de manera automática en función de una característica común. En el día a día, nos encontramos múltiples ejemplos de esta clasificación automática como, por ejemplo, la detección de spam que hacen nuestros clientes de correo.

En esta práctica se llevará a cabo la clasificación de un corpus de páginas web de acuerdo con su temática. La práctica se realizará **en grupos de hasta 3 personas**. Cada grupo se identificará por un nombre a vuestra elección.

Pasos a seguir:

1) En esta práctica, os proporcionamos un corpus de páginas web pertenecientes a universidades. Cada página web pertenece a una categoría temática diferente (*course, department, faculty, other, project, staff,* etc.). La colección se encuentra dividida en conjunto de entrenamiento y test. De este último, no disponéis de las etiquetas, ya que se usará para la evaluación.
Podéis descargaros la colección del siguiente enlace:
https://nubeusc-my.sharepoint.com/:u:/g/personal/marcosfernandez_pichel_usc_es/EcdOwLXoi-pIjOyaXPlhMBABeatlEuD0Y8us2DlgyHRcvw?e=LSBKu3.
Sugerencia: podéis crear un split de validación a partir del conjunto de entrenamiento para comprobar cómo funcionan vuestras diferentes soluciones. Podéis usar el paquete de Python *split-folders* para ello.

2) El principal objetivo de la práctica se resume en que propongáis diferentes soluciones al problema de clasificación de las páginas web. Esta parte es libre y cada grupo presentará una o varias soluciones diferentes al problema de clasificación.
A continuación, os presentamos una lista de posibles ideas, pero no son las únicas y se valorará la originalidad de cada grupo y la búsqueda de soluciones alternativas al problema:

- Utilización de algoritmos tradicionales de clasificación (p.e., una SVM, un Naïve Bayes, KNN, etc), de modelos de Deep NLP (e.g. como BERT, RoBERTa, XLNet, etc.) o utilización de estrategias de prompt engineering con Large Language Models como ChatGPT.
  Sugerencia: Para los modelos de DeepNLP podeís usar la librería de HuggingFace (https://huggingface.co/docs/transformers/training) o Ernie (https://github.com/labteral/ernie). Tened en cuenta que estos modelos requieren de GPU para hacer inferencia, podéis usar herramientas como Google Colab, para ello.
- Utilización de diferentes estrategias de preprocesado (e.g. eliminar o no los tags HTML) y de extracción de características de texto.
- Probar diferentes librerías de limpieza de texto (p.e. Pyplexity: https://github.com/citiususc/pyplexity).
- Probar estrategias de "document expansión" para generar queries artificiales a partir de los documentos y expandir los textos con esas queries (e.g. docTTTTTquery: https://github.com/castorini/docTTTTTquery).
- Ampliar el training data (p.e. crawleando páginas de departamentos de universidades).
- Aplicar "transfer learning" usando clasificadores disponibles de alguna otra dimensión y que puedan generalizar bien a esta tarea (p.e. aplicar un clasificador de readability como este https://huggingface.co/valurank/en_readability con la hipótesis de que, por ejemplo, las páginas de los departamentos son más legibles).

3) Cada equipo deberá mandar varias variantes o soluciones (“runs”) en un csv con el siguiente formato:

doc-id-test categoría predecida (p.e. “department”)
doc-id-test categoría predecida (p.e. “department”)
….

4) Estas runs serán evaluadas automáticamente por los profesores de la materia y se presentará un “**leaderboard”** con las soluciones ordenadas por la métrica F1 macro. El objetivo no es ganar la competición y no se penalizará un rendimiento bajo. El principal objetivo del leaderboard es que todos/as aprendamos y veamos qué estrategias de clasificación funcionan mejor para este problema.

Entregables:

(en un único archivo .ZIP)

1) Guión python (.py)
2) Python Notebook (.pynb) (sed particularmente cautos/as en detallar los resultados de los experimentos, etc) con las diferentes soluciones implementadas
3) Ficheros .csv con las variantes

Es fundamental que el Notebook sea autoexplicativo de todos los pasos (con celdas textuales acompañando a celdas con código y que contenga explícitamente los resultados -sin tener que ejecutar las celdas de nuevo-). Comprobad esto antes de enviar el Notebook. Cualquier proyecto de Analítica de Datos debe ser autodocumentado y sus experimentos fáciles de reproducir. **Un aspecto clave en la evaluación de esta práctica reside en la calidad de las explicaciones y documentación que acompañéis al código dentro del Notebook**.

Dado que se trata de un trabajo en grupo el notebook debe tener un apartado en el que se explique cómo se ha organizado el trabajo, cómo se han dividido las tareas o retos a afrontar entre las personas del grupo y cómo se ha coordinado la comunicación, escritura de resultados, análisis, programación, etc.

- **Valoración y Fecha de Entrega:**

  Esta práctica tiene una valoración de 4 puntos (sobre el total de 7 puntos de la parte práctica de la materia)

  Fecha entrega: Habrá **tres entregas**. Una primera entrega de variantes el **3 de noviembre, a las 23:59h**. Esta entrega servirá para generar el primer leaderboard y que podáis ver cómo funcionan vuestras distintas soluciones y mejorarlas. Una segunda entrega de variantes será el **24 de noviembre, a las 23:59h**.

  La entrega definitiva será el **15 de diciembre, a las 23.59h.**

  Cada grupo en cada ronda de entrega de variantes sólo podrá enviar 5 ficheros de estimaciones (por ejemplo, los 5 modelos o aproximaciones que entiende son más prometedoras). Cada fichero de variante debe tener un nombre de archivo que identifique claramente qué grupo lo envía y de qué variante se trata (por ejemplo, MOZOS_DE_AROUSA_variante_SVM_kernel RBF.csv).

  El fichero ZIP con el código, notebook, etc. sólo se entregará en la ronda final (el 3 y el 15 de noviembre sólo se entregarán archivos CSVs con estimaciones).

Habrá una sesión final de defensa/presentación de resultados a realizar el 18 de diciembre a las 16:00h. Cada grupo tendrá 10 min para exponer su solución.

Se permiten entregas retrasadas del ZIP final pero se reducirá la puntuación del siguiente modo:

> Cada día tarde reduce en un 10% de la máxima nota alcanzable (es decir, cada día tarde resta un 0.4 puntos de la nota que se os asigne al valorar la práctica)