Laboratorio - 2024

Introducción al Procesamiento de Lenguaje Natural

Para el laboratorio de este año les proponemos una serie de ejercicios relacionados a los modelos de lenguaje de n-gramas y a la clasificación de textos según su autoría.

Guiándose por el notebook publicado en el EVA, van a ir resolviendo los ejercicios propuestos, señalizados por números del (0) al (7). En estos ejercicios van a tener que implementar algoritmos, calcular métricas, hacer varios experimentos, compararlos y extraer conclusiones. Todo eso lo van a reportar en el informe a entregar. Más adelante les dejamos una guía sobre qué esperamos de ese informe.

¿Qué hay que entregar y hasta cuándo hay tiempo?

Teniendo como fecha límite el **lunes 11 de noviembre a las 23:59 hs.**, lo que tienen que entregar en el receptor de EVA es:

- el notebook completo con la solución, con **todas las celdas ejecutadas**
- un informe de un máximo **6 páginas en total** que describa su trabajo

⚠️ No consideraremos entregas que no cumplan con **estas restricciones**.
⚠️ Tanto el informe como el notebook deben tener el nombre y apellido de todos los integrantes del equipo.

¿Qué se espera del informe?

En el informe tienen que reportar, de manera ordenada, el trabajo que hicieron y las conclusiones que extraen de cada parte. Considerando que el notebook tiene cierto hilo conductor, esperamos que el informe que ustedes redacten también lo tenga (p. ej.: cómo aprovechan una parte para resolver otra).

Si les resulta más intuitivo, pueden dividir el informe en dos partes, correspondientes a las partes del laboratorio. A continuación les damos un punteo de detalles que esperamos encontrar en él, siguiendo esa separación por partes:

- Primera parte: Modelos de lenguaje de n-gramas
  - (1)→ Si bien en el ejercicio (0) ya se hace un preprocesamiento básico, pueden probar con pipelines que agreguen pasos complementarios. En ese caso, describan brevemente cada uno de ellos.
  - (1)→ Incluyan una tabla que compare los resultados obtenidos al inferir. Recuerden incluir el valor de *n* y qué pipeline usaron.
  - (1)→ Escriban algunas observaciones sobre cada uno de los textos generados que aparecen en esa tabla. ¿Son sintácticamente correctos? ¿Tienen sentido?

- (2)→ ¿Qué similitudes encuentran entre cada texto generado y su correspondiente top 3 de oraciones originales del libro con las distancias más bajas?
- (3)→ Sigan un formato similar al que usaron para el ejercicio anterior. Comenten qué similitudes encuentran entre cada texto generado y su correspondiente top 3 de oraciones originales del libro.
- (3)→ Incluyan breves observaciones comparando los top 3 obtenidos mediante mínima distancia de edición y los top 3 obtenidos mediante *sentence transformers*. ¿Los resultados son similares? ¿En qué se diferencian?
- Para cerrar esta primera parte, sería interesante leer algunas observaciones finales sobre el funcionamiento de los modelos de n-gramas y cómo el texto que generan se relaciona con el texto usado para entrenarlos. ¿Qué tan bien esos modelos generalizan el texto usado? ¿Repiten muchos fragmentos del texto original?

- Segunda parte: Clasificación de textos por autor
  - (5)→ Incluyan una tabla que muestre los tamaños de los subconjuntos (y la cantidad de instancias de cada clase) del corpus.
  - (6)→ Tal y como dice el notebook, esperamos que entrenen varios modelos y exploren diferentes direcciones. Asegúrense de probar **al menos 4 enfoques**.
  - (6)→ Más allá de los enfoques estadísticos o neuronales, también pueden usar enfoques por reglas (p. ej.: usando análisis sintáctico o conteo de palabras). En caso de usarlos, asegúrense de describir cuál es la idea general de su funcionamiento.
  - (6)→ Incluyan una tabla que resuma las características de cada modelo para facilitar su comparación: qué método usaron para representar numéricamente a los textos (p. ej.: bag of words, sentence transformers), qué algoritmo usaron para entrenar el modelo (p. ej.: Support Vector Machines, Naive Bayes, Multilayer Perceptron), cuáles son los valores de sus hiperparámetros, y qué resultado obtuvieron sobre *dev* según *precision*, *recall* y F1. En el caso de que usen *Large Language Models* con *few-shot* (o, incluso, *zero-shot*), indiquen en esa tabla el modelo (p. ej.: Gemini, Mistral, GPT-4) y qué prompt usaron como entrada. Si usaron pipelines adicionales de preprocesamiento como complemento al preprocesamiento básico del ejercicio (4), no olviden indicarlo también en la tabla, comentando brevemente en qué consisten.
  - (6)→ Analicen los resultados obtenidos ¿Qué modelos funcionaron mejor? ¿Hay algún componente que parezca determinante en los resultados obtenidos (p. ej.: el *pipeline1* de preprocesamiento complementario mejora siempre el puntaje F1)?
  - (7)→ Incluyan una tabla análoga a la del ejercicio anterior, pero evaluando sobre *test* los tres mejores modelos al evaluarlos en *dev*. ¿El mejor modelo en *test* es el mismo que el que funcionó mejor en *dev*?
  - (7)→ Es importante que esta parte la hagan al final, como último paso. A partir de esta evaluación, ningún modelo puede ser mejorado, ya que podría haber riesgo de caer en un sobreajuste (*overfitting*). Es por esa razón que usamos *dev* para evaluar durante el desarrollo del modelo.