Diplomatura en Ciencia de Datos, Aprendizaje Automático y sus Aplicaciones 2020

Detección de Sarcasmo

Materia: Análisis y Visualización de datos

Introducción

En este primer trabajo nos enfocaremos en algunos de los conceptos abarcados por la materia de análisis y visualización pero con algunas particularidades especiales, ya que a diferencia de un dataset con datos numéricos trabajaremos como ya lo hemos hablado previamente con lenguaje natural, para el cual las técnicas de análisis y visualización son un poco diferentes.

Por ejemplo, algo muy utilizado dentro del lenguaje natural para tener una noción rápida de las palabras que estamos tratando es la técnica de clusterización, pero la misma quedará para más adelante cuando tratemos el práctico de aprendizaje no supervisado. Si tienen algún interés al respecto les dejo el link de wikipedia al tema: https://en.wikipedia.org/wiki/Document_clustering

La idea principal de este práctico es entender el dataset que vamos a trabajar, visualizar las features disponibles y empezar a pensar en qué dificultades nos podemos llegar a encontrar cuando tengamos que realizar la curación del dataset.

Tareas a realizar

La entrega del práctico tendrá que tener el formato de una jupyter notebook (.ipynb) en donde se esperarán encontrar los siguientes análisis:

- Analizar y visualizar la frecuencia de palabras, para esto pueden usar cualquiera de las librerías de lenguaje natural, la idea es que investiguen cuáles son y una vez que decidan cuál o cuáles usar lo charlemos y las veamos en detalle antes de la entrega final. El análisis de la frecuencia de palabras es importante porque nos permite dar una noción rápida de los tópicos principales del dataset, acá podríamos empezar a pensar quizás si hay algún “tipo” de sarcasmo más frecuente.
  En este punto podrían mostrar gráficos diferentes para las palabras más frecuentes, las intermedias y las de baja ocurrencia.

- ¿Existe alguna correlación entre las palabras más frecuentes y menos frecuentes? Algo que se puede probar es inspeccionar las palabras cercanas en cuanto a distancia de una palabra frecuente y ver si existe alguna relación. Lo mismo para las palabras menos frecuentes. Esto tiene que ver con un problema de NLP que se llama “mutual information”, pueden averiguar al respecto pero para este práctico no es necesario tanto detalle.

- Investigar acerca de los n-gramas, librerías existentes y cómo aplicarían este concepto a nuestro corpus

Visualizaciones esperadas

- Histograma con la frecuencia de las palabras

- Mostrar en un heatmap las frecuencias observadas, por ejemplo pueden seleccionar un grupo de las más frecuentes, de las promedio y de las menos frecuentes y hacer heat maps comparando (serían 3 heat maps)

- Visualización de n-gramas encontrados

- Visualizar las palabras con mayor grado de correlación (opcional)

Una vez realizadas estas tareas sacar conclusiones acerca de lo observado, algunas preguntas que podrían hacerse:
¿Es posible encontrar algún término que nos indique la presencia de un posible sarcasmo?
¿De qué otra manera podríamos analizar las palabras?
¿Sería útil crear por ejemplo distintos conjuntos de palabras y analizarlos por separado?
¿Qué pasa con los signos de puntuación, interrogación, nos sirven?

Entrega

La entrega será por mail donde se adjuntará la notebook con la que hayan resuelto los enunciados, deberá tener el nombre “grupoN_lab1” donde N pertenece al conjunto {1, 2} según el grupo al cual pertenezcan :)
Como asunto del mail poner “entrega_mentoria”.

Las consultas que tengan las vamos resolviendo por los canales que tenemos habilitados: Slack, Whatsapp, mail, etc.

Cuando hayan leído el práctico vamos a coordinar una reunión para resolver cuestiones puntuales de los enunciados. Dejo una cronología tentativa de reuniones:

Reunión orientativa: 20/06/2020
Reunión de seguimiento: 24/06/2020
Reunión pre-entrega: 28/06/2020
Entrega: 29/06/2020

A continuación les dejo algunos links donde pueden sacar información sobre los temas necesarios, son sólo de carácter orientativo, no se pretende que los usen como metodología de trabajo:

https://stackabuse.com/python-for-nlp-developing-an-automatic-text-filler-using-n-grams/

https://stackoverflow.com/questions/40669141/python-nltk-counting-word-and-phrase-frequency

https://pythonprogramming.net/tokenizing-words-sentences-nltk-tutorial/

https://stackoverflow.com/questions/35596128/how-to-generate-a-word-frequency-histogram-where-bars-are-ordered-according-to