**UNIVERSIDAD EAFIT**

**MAESTRÍA EN CIENCIA DE DATOS Y ANALÍTICA**

**ST1800 ALMACENAMIENTO Y RECUPERACIÓN DE INFORMACIÓN, 2021-1**

**PROFESOR: EDWIN MONTOYA –** [**emontoya@eafit.edu.co**](mailto:emontoya@eafit.edu.co)

Trabajo 2 – Preparación de texto, representación de características y representación de documentos.

Apache Solr

Fecha de entrega: 8 de marzo de 2021.

# 

**Hallar el BoW más optimizado posible para un corpus, Realizar la representación de características y de documentos en diferentes librerías**

Objetivos:

Aprender y aplicar diferentes librerías para tokenización y preparación de datos, de tal manera que el estudiante sea consciente de los procesos de preparación de textos como tokenización, remoción de stopwords, remoción/reemplazo de caracteres especiales, procesos de de stemming y lemmatization.

Aprender y aplicar diferentes técnicas de representación de características como one-bit encoding, TF, TF-IDF, BM25.

Aprender y aplicar diferentes técnicas de representación de documentos, principalmente en Modelo de Espacio Vectorial (VSM) y opcionalmente doc2vec.

Aprender diferentes librerías en Python para realizar preparación de texto como NLTK, Spark y META.

Descripción:

Corpus:

Tomar como base el conjunto de documentos \*.txt así como el archivo papers-dc.xml que se encuentran en el github:

<https://github.com/st1800eafit/st1800_20211/tree/main/datasets/papers_sample_pdf>

Librerias a utilizar:

El proyecto lo deberá realizar con:

* Python, nltk, scikit y/o gensim.
* PySpark, con Spark y las diferentes librerías de nlp (algunas funciones las podrá complementar de nltk o similar sino se encuentra implementada en pyspark)
* Metapy, usar la librería <https://meta-toolkit.org/> y particularmente la implementación en Python (metapy), en el sitio se encuentra numerosos ejemplos. Mirar los ejemplo del github: <https://github.com/st1800eafit/st1800_20211/tree/main/meta>

Todos los ejemplos deberán ser realizados en google colab y compartir la carpeta con los datos, notebooks e instrucciones a la cuenta [edwin.montoya@gmail.com](mailto:edwin.montoya@gmail.com) para la revisión y calificación. Opcionalmente podrá utilizar otros ambientes de desarrollo y ejecución de Python y pyspark como AWS EMR/Notebooks, AWS Sagemaker, etc con datos en AWS S3.

Requerimientos:

Text prep:

* Remoción de caracteres especiales y tokens que considere irrelevantes para una consulta.
* Realice diferentes opciones de tokenización (.split() y las propias de cada librería)
* Remoción de stopwords
* Stemming
* Lemmatization
* Realizar tokenización adicional en ngrams (bigram y trigrams)
* Realizar en cada paso la impresión del tamaño del BoW parcial, los 20 top tokens en TF (Term Frequency) y la visualización en un histograma.
* Hasta acá, el objetivo es obtener el BoW más óptimo (reducido) para pasar a la fase de representación de característicos y de documentos.

Representación de características y de documentos:

* Investigar en cada una de las librerías (nltk/spacy/scikit-learn/gensim, pyspark, metapy) diferentes opciones de representación de características como one-bit, TF, TF-IDF, BM25, word2vec, doc2vec.
  + Ej: en pyspark utilizar HastingTF, CountVectorizer, IDF, etc. librerías similares existen en las otras librerías.
  + Nota: en las librerías se fusiona en el mismo ‘paso’ la representación de características y la representación de documentos.
* Al final del proceso, deberá almacenar en disco los modelos de representación de características y de documentos, igualmente una utilidad para cargar de disco dichos modelos.

Otros ejemplos y demos pueden ser encontrados en el github de la materia: <https://github.com/st1800eafit/st1800_20211.git> y muchos foros de comunidad:

* Data Science Central
* Towards data science
* Kaggle.com
* Medium Daily Digest NLP

# Criterios de evaluación:

Se tendrán en cuenta los siguientes porcentajes de evaluación de cada una de las partes:

|  |  |
| --- | --- |
| ITEM | Porcentaje |
| Text Prep, Rep características y docs en Python y librerías relacionadas | 40% |
| Text Prep, Rep características y docs en PySpark con las librerías de SparkML de NLP y Text Mining | 40% |
| Text Prep, Rep características y docs en metapy | 20% |