Entrega final Proyecto Integrador

Liceth Cristina Mosquera - lcmosquerg@eafit.edu.co
Juan Diego Estrada Perez - jestra15@eafit.edu.co
Johan Steward Rios Naranjo - jriosna1@eafit.edu.co
Juan Mauricio Cuscagua Lopez - jcuscagu@eafit.edu.co
Programa: Almacenamiento y Recuperación de Información
Docente:
Edwin Nelson Montoya Munera
24 de junio de 2019

***Resumen*—En la actualidad existen muchas librerías en programación para procesamiento de texto. Lo que hacen estas librerías es facilitar las implementaciones para aplicaciones de minería de texto. Consisten en trabajar con datos no estructurados, como lo son los documentos, y aplicar un proceso para darles estructura y así poder realizar diferentes aproximaciones y metodologías que permitan cumplir con un objetivo en específico.**

En este documento se busca establecer una serie de pasos y técnicas alrededor de la minería de texto que permita dar cuenta de las diferentes posibilidades existentes que se pueden aplicar según un objetivo específico, así como hablar de las librerias que se pueden usar sobre el lenguaje de programación python y de los servicios de computación en la nube disponibles que brindan los recursos necesarios para implementar dichos métodos.

Palabras Clave: Procesamiento de texto, datos no estructurados, minería de texto, python, librerias, conputación en la nube

I. Introducción

El procesamiento de texto, minería de texto o simplemente analítica de texto es un conjunto de técnicas estadísticas computacionales, que nace como respuesta a las nuevas tendencias y permite analizar, de una manera estructurada, datos almacenados de manera no estructurada como lo son los textos.

Lo anterior no quiere decir que la manera de representar e interpretar esta información no se haya estudiado en el pasado, de hecho, los primeros estudios relacionados al tema se referencian en la década de 1950 [1] sólo que dichas tendencias obedecen a los cambios que han generado los avances tecnológicos, que han permitido capturar información estructurada y no estructurada de manera novedosa y masiva. Las nuevas bases de datos que se generan están compuestas por ambos tipos de datos m las cuales requieren el desarrollo de nuevas técnicas para ser analizados e interpretados.

Hoy en día, como derivado de la hiper conectividad, las empresas y las personas están cada vez más interesadas en generar contenido web, contenido que se materializa a través de redes sociales, mensajes de texto, blogs, entre otras y dicho interés ha generado una conexión natural por el campo y por estas técnicas, buscando extraer información que logre apalancar a quienes toman decisiones [**?**].

El objetivo general del procesamiento de texto es entonces lograr que la información no estructurada almacenada como texto se transforme en información que pueda ser analizable para tomar decisiones, para esto, existen varias etapas de trabajo que se aprovechan de técnicas estadísticas y computacionales robustas, estas son:

1. Recuperación de la información
2. Procesamiento de lenguajes naturales (NLP) por sus siglas en inglés
3. Extracción de la información
4. Minería de datos

Al hablar de todo esto no se hace mención de algo tan importante como las técnias en sí, y es lo complejo de utilizar estas técnicas en conjuntos de información tan bastos como lo es el texto, en donde habitualmente se busca aplicar dichos procesos a miles de textos. Como una solución a esto, diferentes empresas, de las cuales nos centraremos en Amazon, han creado una serie de servicios orientados a solucionar estos problemas de arquitectura que podrían sufrir aquellas personas o empresas que desean hacer uso de estas técnicas.

Todo esto es el centro de este trabajo, profundizando en cómo se debe realizar la identificación de los documentos relevantes dada una búsqueda o consulta hecha por el usuario y como el análisis de tópicos sirve como un forma de establecer similitud entre documentos.

II. Justificación

La minería de texto busca extraer información significativa y útil de los mismos con el fin de ser usada para un propósito específico. Estos propósitos pueden ir desde algo general como la recuperación de texto a través de un motor de búsqueda, como algo tan específico como el análisis de sentimientos con el fin de catalogar los textos vistos en redes sociales y determinar si las campañas publicitarias están surtiendo efecto o no. Es debido a estos diferentes enfoques y usos que la minería de texto se hace tan relevante hoy en día. El aprendizaje de las diferentes técnicas es necesario para poder enriquecer la información que se encuentra disponible y en tan grandes cantidades como lo es el texto. El uso de un solo método o la combinación de varios hará que una tarea en específico sea mejor entendida y aprovechada. Como dicha información es tan abundante y tan compleja, se requiere de una buena máquina para correr dichos métodos y generar la solución buscada. Como solución a los altos costos que implica construir un computador con la capacidad necesaria

para desarrollar y ejecutar dichos métodos, existe el cloud computing, el cual es un servicio que diferentes empresas ofrecen para que dichos procesos sean ejecutados en servidores destinados de esas compañias y que permiten configurar que tan potente es el ambiente que necesitas para desarrollar el trabajo que buscas. Adicionalmente dichas empresas, han incluido en sus serviciso una serie de métodos ya optimizados para que sea más fácil aplicar estos al conjunto de datos que tengas. Por esta razón, el uso de dichos servicios es hoy en día una labor del día a día de las personas que buscan analizar información.

III. Marco teórico

Citando el libro ”Deep Text”de Tom Reamy, se pretende mostrar la importancia del porqué hacer analítica de texto. .[El] análisis de texto puede ahorrarle decenas de millones de dólares, abrir nuevas dimensiones de inteligencia de clientes y comunicación y, de hecho, permitirle utilizar una pila gigante de lo que actualmente se considera en su mayoría cosas inútiles: texto no estructurado.” [2].

Es por esta importancia que muchas personas hablan acerca de qué es la analítica de texto y cómo usarla. Sin embargo, no comienzan por el porqué es importante, y son los mismos usuarios los que deben encontrarle una utilidad y un sentido. Según el artículo [3] se puede definir la analítica de texto como una extensión de la minería de datos, cuyo propósito es encontrar patrones de texto en grandes fuentes no estructuradas de información.

La utilidad de la analítica de texto y el procesamiento del lenguaje natural a menudo se presentan como funciones de ciencia de datos muy difíciles de aprender y usar y que sólo pueden ser entendidas y manejadas por científicos de datos entrenados en el tema. Sin embargo, los principios teóricos sobre los que se fundamenta la analítica de texto son fáciles de entender y en general, es lo que pasa con muchos problemas de analítica en ciencia de datos. Por ejemplo, un motor de análisis de texto debe dividir las oraciones y las frases antes de poder entrar a analizar cualquier cosa, dividiendo documentos de texto no estructurados en sus principales componentes. Este es el primer paso en casi todas las funciones de procesamiento del lenguaje natural, que incluye la recomendación de textos [4], la extracción de temáticas [5], entre otras.

En [6] se puede encontrar una revisión de diferentes técnicas utilizadas con ayuda de minería de texto, para diferentes campos de investigación y los problemas que pueden surgir al momento de implementar alguna de estas metodologías.

El obejtivo, indiferente de la elección de la forma de generar la bolsa de palabras, reside en generar un índice invertido el cual es una estructura de datos que reune en un solo lugar, palbras, documentos, frecuencia de palabra en el documento, posición en el documento, y en general cualquier información que se considere relevante para el rank del documento [7]. Para este rank existen muchas formas de calcularlo, pero para este caso se usa el Okapi BM25, el cual es una función de ranking utilizada en recuperación de información para la asignación de relevancia a los documentos en un buscador, dicho de otra forma, es una función que nos permite ordenar por relevancia los documentos que contienen las palabras que el usuario ha introducido en la caja de búsqueda de un buscador [8].

Diferentes usos como el análisis de opiniones [9] o el análisis de sentimientos [10] son utilizados con gran medida para direccionar campañas publicitarias en las empresas, tomando como fuente de datos diferentes medios como las redes sociales o los blogs de internet. Se busca reconocer patrones en palabras clave como los numerales (#) o el arroba (@) y captar las temáticas tendencia del momento así como los adjetivos subyacentes a dichas temáticas con el fin de generar una respuesta que mejore o corrija lo que se encontró en dicho análisis.

Uno de los campos más usados en análisis de textos es el modelado de tópicos, la razón es simple, este permite generar una lista corta de palabras que representen en su mayoría a los textos y así poder clasificar textos comunes según estos tópicos. Adicionalmente para llegar a esta forma final, es necesario una limpieza de la información y además generar una reducción de dimensión del espacio vectorial resultante del proceso de indexación. La importancia de esta técnica ha llevado al desarrollo de métodos estadísticos y computacionales que permitan realizar de manera eficiente la generación de estos tópicos. Una revisión del modelado de tópicos y una comparativa matemática y formal de algunos métodos puede ser vista en el artículo ”Topic Modeling: A Complete Introductory Guide” [11].

IV. Desarrollo

IV-A. *Análisis Descriptivo*

Para esta parte se realiza una breve descripción del contenido de los textos, como que palabras son más frecuentes o que diario es el que escribe en su mayoría las noticas.

IV-B. *Almacenamiento y Cluster de Procesamiento*

Para hablar de este tema se refiere a una imágen proveída por AWS en el siguiente enlace:

https://aws.amazon.com/blogs/big-data/secure-amazon-emr-with-encryption/.

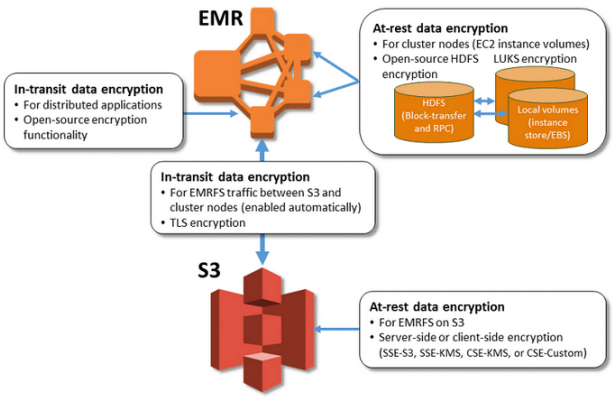

Figura 1: Arquitectura usada de AWS

IV-B1. Almacenamiento de Información: Para el almacenamiento de la información se utiliza S3, un servicio de AWS donde se pueden crear buckets o carpetas donde se aloja la información. Con la posibilidad de hacerlos públicos para que no sea necesarion un login para su consulta, o permitiendo acceder a él con la librería boto3 de python.

IV-B2. Configuración de Ambiente EMR: Pasos para configuración ambiente AWS:

- Acceder a la cuenta de aws educate por medio del siguiente link: https://www.awseducate.com/signin/SiteLogin Ingregsar a la opción AWS Starter Account
- Ingresar a la consola AWS
- Para almacenar en la nube archivos con AWS utilizamos S3. Creamos el Bucket lo hacemos público y se sube el archivo mediante un drag and drop
- Para configurar el cluster y poder procesar en paralelo las noticas, se utiliza EMR (Elastic Map Reduce) el cual es un servicio ofrecido por AWS para creacion de clusters en la nube
- Se ingresa a EMR, se crea el cluster con la siguiente confguración:
  - Nombre del cluster: Almacenamiento
  - Loggin: Verdadero
  - S3 folder: s3://aws-logs-895339990044-us-east-1/elasticmapreduce/
  - Release: emr-5.24.0
  - Application: Spark: Spark 2.4.0 on Hadoop 2.8.5 YARN with Ganglia 3.7.2 and Zeppelin 0.8.1
  - Instance type: m3.xlarge
  - Número de instancias: 3
  - EC2 Key Pair: almacenamiento-dos
- Una vez hecha la conexión a AWS y se ha instalado Jupyter Hub desde esa conexión. Se trae toda la información que se venía trabajando en Jupyter. Se abre una terminal, se escribe git clone y la URL del repositorio de donde se venía trabajando (text mining).

IV-B3. Instalación de Paquetes: Para instalar librerias adicionales, como nltk para el preprocesamiento, ingresamos a cada maquina por ssh utilizando el EC2 Key Pair configurado en la creación del Cluster.

Para ingresar al master:

- ssh -i almacenamiento-dos.pem hadoop@ec2-18-205-245-164.compute-1.amazonaws.com
- ssh -i almacenamiento-dos.pem hadoop@ec2-18-205-245-164.compute-1.amazonaws.com
- sudo su - pip install pandas pip install matplotlib pip install nltk
- python -m nltk.downloader stopwords
- Para ingresar al slave 1: ssh -i almacenamiento-dos.pem hadoop@ec2-18-207-150-157.compute-1.amazonaws.com
- Para ingresar al slave 2: ssh -i almacenamiento-dos.pem hadoop@ec2-54-235-5-64.compute-1.amazonaws.com
- Nota: Si no es posible ingresar por ssh a las maquinas es necesario configurar que puerto y desde que ip se puede acceder a estas maquinas. Esto es posible hacerlo, ingresando a cada maquina y configurando por la opción: Servicios -¿EC2 -¿Network Security -¿Security Groups -¿Inbound: Aca nos paramos en cada maquina en editamos la tabla para agregar la siguiente configuración
- Type: All TCP Protocol: TCP Port Range: 0 - 65535 Source: Custom Ip: 0.0.0.0/0

IV-B4. Manejo y ejecución de notebooks: Para crear los notebooks y trabajar con Spark, ingresamos a la opción Notebooks, luego Çreate NoteBook", en ingresamos la siguiente información:

- Notebook name
- Cluster: Seleccionamos el cluster previamente creado
- Notebook location: Seleccionamos el lugar donde quedara almacenado el notebook

IV-C. Indexación, Búsqueda y Recomendación con Meta

META es un kit de herramientas que se enfoca en proporcionar una amplia funcionalidad para aplicaciones de minería de texto y une el aprendizaje automático, la recuperación de información y el procesamiento de lenguaje natural enfocado en la indexación, permitiendo obtener los datos relevantes de un gran conjunto de textos [12], además permite interpretar los patrones descubiertos en los datos. META integra las capacidades de búsqueda con funciones de análisis de texto, lo que permite construir una potente aplicación de análisis de texto.

IV-C1. Carga de Información: Dado que AWS maneja los datos en el formato S3 se deben manejar de igual forma en el Jupyter Hub. Normalmente el acceso a S3 a través de python lo hacemos con la librería boto3 en caso que el bucket no sea público.

IV-C2. Paquetes: Los paquetes que se requieren para trabajar son la parte del motor de búsqueda con metapy y boto3 para trabajar con archivos S3.

IV-C3. Índice Invertido: El motor de búsqueda META puede almacenar vectores de documentos en un índice invertido y puntuarlos con respecto a una consulta. Puede ser usado para crear un motor de búsqueda o para hacer clasificación con el método KNN. Almacena información con los términos relevantes en un corpus indexado por identificación de términos $terms_{ids}$, los cuales están asociados con una frecuencia por documento doc_{ids}. Este forma de guardar la información permite disminuir tiempo de búsqueda.

IV-C4. Ranking: La función utilizada para ranquear fue Okapi BM25, la cual permite ordenar por relevancia los documentos que contienen las palabras del vector de búsqueda (query). Esta expresión comprende cuatro partes, la componente de la frecuencia inversa del documento (IDF), la componente de frecuencia de término (TF), dos funciones de saturación controladas por los parámetros suavizado del termino de documentos k1 = 1.2 y suavizado del término de consulta k3 = 500, los cuales controlan la saturación de las funciones y el factor de normalización de longitud de documentos b = 0.75. La formula en donde se aplican se muestra a continuación.

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{f(q_i,D) \cdot (k_1+1)}{f(q_i,D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})}$$

IV-C5. Recuperación de Documentos: Se hace a través de una consulta o query. Para la regresión de documentos se utiliza el Descenso de Gradientes Estocástico (SGD) predice respuestas de valores reales de documentos caracterizados. La métrica de evaluación es el error medio cuadrático que permite la comparación del modelo.

IV-C6. Toml y Configuración de Meta: Cuando las dependencias están instaladas. Se configura el archivo config.toml para construir el índice. Meta procesa los datos con un sistema de análisis, tokenización y filtro antes de indexar. Analiza superficialmente los documentos como un conjunto de oraciones compuestas por tokens, tokeniza dividiendo los streams en tokens, a los cuales les aplica filtros de expresiones regulares, elimina stopwords que se han tomado del proyecto Lemur y reduce las palabras a su forma básica con Porter2English stemmer. En config también se especifica entre otras la ubicación del dataset y del Corpus que es la colección de documentos del índice, el uso estimado de la memoria RAM al hacer el indexador, el método para el análisis que es unigram, el maximo del resultado del query que son 10, el método y los parámetros para el ranquer, el método para la recuperación de texto que es SGD, los parámetros del modelo LDA, entre otros.

IV-D. Modelado de Tópicos

El modelado de tópicos es una de las técnicas más usadas en minería de texto debido a la facilidad de su implementación, de su interpretabilidad y que permite establecer un punto común entre documentos diferentes. Esta técnica consiste en extraer aquellas palabras que explican en su mayoría a un documento y así generar una lista de palabras asociadas a él.

IV-D1. Justificación del Modelo Seleccionado: Con el fin de desarrollar el análisis de tópicos, se optó por la implementación del algoritmo LDA (Latent Dirichlet Allocation) debido a la necesidad de tener un contacto con enfoques diferentes a los proporcionados por los proveedores como Google, AWS, Microsoft, etc; los cuales cuentan con metodologías predeterminadas para la extracción de tópicos. En primera instancia, se escogió LDA por ser un algoritmo no supervisado, esto se debe a que no hay un conocimiento a priori de los tópicos de cada texto en el dataset. El objetivo del algoritmo es tomar un conjunto de documentos, y extraer un cierto número K de tópicos, estos serán la salida del algoritmo. Con dicha lista de tópicos, se busca determinar qué porcentaje del documento trata sobre cada tópico encontrado.

Para profundizar más sobre LDA, refiérase al link https://towardsdatascience.com/dimensionality-reduction-with-latent-dirichlet-allocation-8d73c586738c

Las principales ventajas por las que se escogió este algoritmo son:

Eficiente: En general, el algoritmo es suficientemente rápido a la hora de extraer los tópicos en un conjunto de texto, sin embargo, como la mayoría de los algoritmos, empieza a ser un poco más lento a medida que la longitud de los documentos y el número de ellos en el conjunto de datos puede implicar una disminución de eficiencia en su tiempo de ejecución.

Intuitivo: A pesar de ser un algoritmo no supervisado, el output del algoritmo, al ser una lista de tópicos a la que se asocia un peso de acuerdo a cada documento, es bastante fácil de interpretar.

Predicción para nuevos documentos desconocidos: Siempre que la estructura general del documento nuevo sea como la que tenía el dataset con el que se entrenó el algoritmo, LDA será capaz de identificar qué porcentaje de cada tópico se trata en el nuevo documento. Es claro que esto no será así para documentos con estructuras diferentes.

IV-D2. Método de Reducción de Dimensiones: Uno de los grandes desafíos que se tuvo en el desarrollo de la práctica, fue lograr un equilibrio entre interpretabilidad y eficiencia. En este sentido, LDA es un algoritmo suficientemente robusto para tener un acercamiento al problema de extracción de tópicos y una de sus principales ventajas es la interpretabilidad de su output. Una reducción de dimensionalidad, como PCA, podría afectar dicha interpretabilidad en términos de que las palabras que conforman cada tópico, ya no sean palabras individuales sino combinaciones o modificaciones de las mismas.

IV-E. Análisis de Sentimientos

El análisis de sentimientos es una

IV-E1. Forma de Entrenamiento: Para clasificar la noticia en una de las tres categorías (positiva, neutra, negativa), se decide implementar un modelo ya entrenado, el cual se encuentra en a librería textblob de python. Una alternativa sería manualmente clasificar un conjunto de palabras y entrenar un modelo de clasificación que permita reconocer la "polaridad." el sentimiento que tiene esa palabra para posteriormente categorizar la noticia. Siendo ambas opciones supervisadas ya que se depende de una variable respuesta para entrenar el modelo.

IV-E2. Librerias vs AWS: El servicio EMR (Elastic Map Reduce) de Amazon Web Services, trae incluidos unos paquetes y unas herramientas que permiten implementar un conjunto de técnicas, tanto de procesamiento como de entrenamiento y clasificación. Sin embargo, es preferible el uso e instalación de librerías o paquetes externos, porque facilita la "portabilidad." la migración del notebook desarrollado a otro proveedor de servicios similares en caso de ser requerido.

IV-E3. Aplicación, Métodos y Justificación: Al ser un problema de clasificiación supervisada, el cual necesita ser más rápido que preciso por la cantidad enorme de datos y que su interpretabilidad no es la prioridad, existen dos opciones. La primera es el método de Naïve Bayes, el cual se utiliza cuando hay una cantidad inmesurable de datos, y el otro es un SVM normalmente lineal. Para el análisis de sentimientos en textblob se manejan 2 métodos, uno es el de NaiveBayesAnalizer, el cual es un clasificador de NLTK previamente entrenado, y el otro es PatternAnalizer, que se basa en un paquete llamado pattern. La documentación de textblob se encuentra en el siguiente enlace: https://textblob.readthedocs.io/en/dev/advanced_usage.html#sentiment-analyzers. Para este

trabajo se utiliza, se utziliza un modelo preentrenado con textblob. El parámetro del método de clasificación por defecto de textblob.sentiment, el cual es PatternAnalizer, la razón es que al no implicar un volumen de datos extremadamente largo, no se ve necesario el uso del método de Naïve Bayes.

V. Resultados

Los documentos son datos no estructurados, pero se les puede dar un componente descriptivo a través de sus metadatos como se observa a continuación.

	id	id_news	title	publication	author	date	year	month	url	content
0	0	17283	House Republicans Fret About Winning Their Hea...	New York Times	Carl Hulse	2016-12-31	2016.0	12.0	NaN	WASHINGTON — Congressional Republicans have...
1	1	17284	Rift Between Officers and Residents as Killing...	New York Times	Benjamin Mueller and Al Baker	2017-06-19	2017.0	6.0	NaN	After the bullet shells get counted, the blood...
2	2	17285	Tyrus Wong, 'Bambi' Artist Thwarted by Racial ...	New York Times	Margalit Fox	2017-01-06	2017.0	1.0	NaN	When Walt Disney's "Bambi" opened in 1942, cri...
3	3	17286	Among Deaths in 2016, a Heavy Toll in Pop Musi...	New York Times	William McDonald	2017-04-10	2017.0	4.0	NaN	Death may be the great equalizer, but it isn't...
4	4	17287	Kim Jong-un Says North Korea Is Preparing to T...	New York Times	Choe Sang-Hun	2017-01-02	2017.0	1.0	NaN	SEOUL, South Korea — North Korea's leader, ...

Figura 2: Contenido de Textos en un Dataframe

En el análisis descriptivo se obtiene que haciendo tokenización con NLTK, teniendo en cuenta solo caracteres alfabéticos, se tienen 102229203 Tokens con 314941 valores únicos y quitando stopwords se tienen 59204236 Tokens con 314796 valores únicos. Aplicando el Stemming de Porter, se tienen 59204236 Tokens con 251540 valores únicos. Aplicando el Stemming de Lancaster, se tienen 59204236 Tokens con 217835 valores únicos. Aplicando Lematizacion de NLTK, se tienen 59204236 Tokens con 296647 valores únicos. En la siguiente gráfica se puede observar las palabras que son más frecuentes.

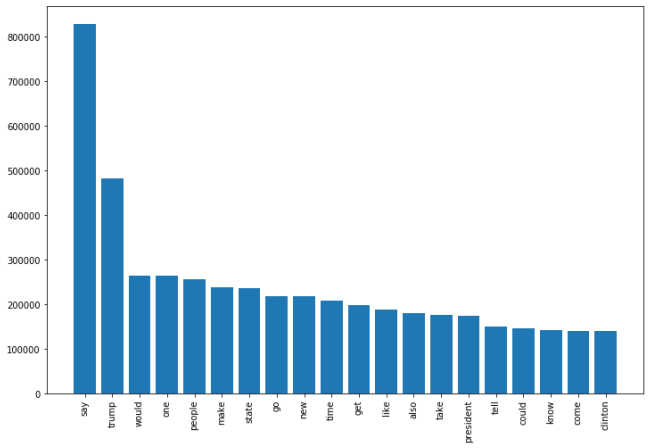

Figura 3: Frecuencia de palabra en los textos

Un análisis adicional muestra como la mayoría de las noticias provienen de un mismo periódico, el New York Times.

```
+---+------+--------------------+--------------------+--------------------+----------+------+-----+----+--------------------+
| id|id_news|               title|         publication|              author|      date|  year|month| url|             content|
+---+------+--------------------+--------------------+--------------------+----------+------+-----+----+--------------------+
|  0| 17283|House Republicans...|New York Times|         Carl Hulse|2016-12-31|2016.0| 12.0|null|WASHINGTON  —   C...|
|  1| 17284|Rift Between Offi...|New York Times|Benjamin Mueller ...|2017-06-19|2017.0|  6.0|null|After the bullet ...|
|  2| 17285|Tyrus Wong, 'Bamb...|New York Times|        Margalit Fox|2017-01-06|2017.0|  1.0|null|When Walt Disney'...|
|  3| 17286|Among Deaths in 2...|New York Times|     William McDonald|2017-04-10|2017.0|  4.0|null|Death may be the ...|
|  4| 17287|Kim Jong-un Says ...|New York Times|       Choe Sang-Hun|2017-01-02|2017.0|  1.0|null|SEOUL, South Kore...|
|  5| 17288|Sick With a Cold,...|New York Times|         Sewell Chan|2017-01-02|2017.0|  1.0|null|LONDON  —   Queen...|
|  6| 17289|Taiwan's Presiden...|New York Times| Javier C. Hernández|2017-01-02|2017.0|  1.0|null|BEIJING  —   Pres...|
|  7| 17290|After 'The Bigges...|New York Times|         Gina Kolata|2017-02-08|2017.0|  2.0|null|Danny Cahill stoo...|
|  8| 17291|First, a Mixtape....|New York Times|     Katherine Rosman|2016-12-31|2016.0| 12.0|null|Just how  is Hil...|
|  9| 17292|Calling on Angels...|New York Times|          Andy Newman|2016-12-31|2016.0| 12.0|null|Angels are everyw...|
+---+------+--------------------+--------------------+--------------------+----------+------+-----+----+--------------------+
only showing top 10 rows
```

Figura 4: Algunos Datos Importantes

Tras el proceso de modelado de tópicos, se traen los 10 tópicos más relevantes para los documentos así como el valor de la distribución asignada, recordando que el método usado para encontrar los tópicos es el Latent Dirichlet Allocation, el cual asigna probabilidades a los vectores de información y trae aquellos que explican en su mayoría los textos.

```
+---+--------------------+--------------------+--------------------+--------------------+--------------------+
|uid|                news|               words|         rawFeatures|            features|   topicDistribution|
+---+--------------------+--------------------+--------------------+--------------------+--------------------+
|  0|House Republicans...|[hous, republican...|(1000,[0,1,2,3,4,...|(1000,[0,1,2,3,4,...|[0.03091270065332...|
|  1|Rift Between Offi...|[rift, offic, res...|(1000,[0,2,3,4,5,...|(1000,[0,2,3,4,5,...|[0.642067193531 16...|
|  2|Tyrus Wong, 'Bamb...|[tyru, wong, bamb...|(1000,[0,2,4,5,6,...|(1000,[0,2,4,5,6,...|[0.10274094722929...|
|  3|Among Deaths in 2...|[among, death, 20...|(1000,[0,2,3,4,5,...|(1000,[0,2,3,4,5,...|[0.08045837977475...|
|  4|Kim Jong-un Says ...|[kim, jongun, say...|(1000,[0,1,3,5,6,...|(1000,[0,1,3,5,6,...|[0.03669554815878...|
|  5|Sick With a Cold,...|[sick, cold, quee...|(1000,[0,6,9,10,1...|(1000,[0,6,9,10,1...|[0.127183173874 19...|
|  6|Taiwan's Presiden...|[taiwan, presid, ...|(1000,[0,1,2,3,4,...|(1000,[0,1,2,3,4,...|[0.082849043227 99...|
|  7|After 'The Bigges...|[biggest, loser, ...|(1000,[0,2,3,4,5,...|(1000,[0,2,3,4,5,...|[0.058486103211 25...|
|  8|First, a Mixtape....|[first, mixtap, r...|(1000,[0,2,3,4,6,...|(1000,[0,2,3,4,6,...|[0.042232199933 56...|
|  9|Calling on Angels...|[call, angel, end...|(1000,[0,2,4,6,8,...|(1000,[0,2,4,6,8,...|[0.106583441637 00...|
+---+--------------------+--------------------+--------------------+--------------------+--------------------+
only showing top 10 rows
```

Figura 5: Topicos en Documentos

Para el análisis de sentimientos se tiene un histograma de la polaridad de las noticias, mostrando en su mayoría un sentimiento entre neutro y positivo.

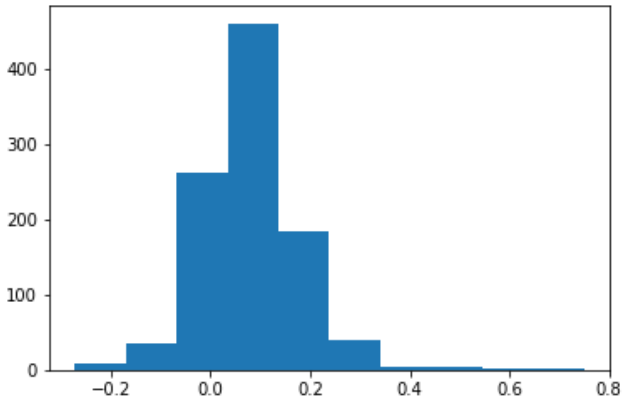

Figura 6: Polaridad de los sentimientos en documentos

VI. Conclusiones

AWS cuenta con una serie muy variada de servicios, desde ambientes altamente personalizados y con acceso a una gran cantidad de herramientas hasta una máquina muy sencilla que necesita ser configurada totalmente.

Existen circunstancias en las que algunas librerías deben crear archivos que deben ser almacenados. El EMR de AWS no permite almacenar datos como si fuera un ambiente local, por lo cual la instalación de dichos paquetes debe hacerse desde consola.

Meta es un set de herramientas muy completo que permite hacer todo lo relacionado con text mining de una manera muy rápida al correr en c++.

Los métodos de modelado de tópicos requieren de un buen preprocesamiento de datos previo para un correcto funcionamiento.

El método LDA es a su vez un método que reduce la dimensión de los datos, no solo funciona como clasificador. Aunque es recomendable complementar con otra técnica como un PCA o una descomposición en valores singulares.

Una forma de ahorrar mucho tiempo al momento de hacer

análisis de sentimiento es la de usar modelos preentrenados existentes en paquetes como nltk o textblob.

REFERENCIAS

[1] A. Turing, "Mind a quarterly review of psychology and philosophy," 1950.
[2] T. Reamy, *Deep text*. 2016.
[3] T. R. Antonio Moreno, "Text analytics: the convergence of big data and artificial intelligence," 2016.
[4] S. M. ChengXiang Zhai, *Text Data Managment and Analysis*. 2016.
[5] S. Li, "Topic modeling and latent dirichlet allocation (lda) in python," May 2018.
[6] S. A. Ramzan Talib, Muhammad Kashif Hanif† and F. Fatima, "Text mining: Techniques, applications and issues," 2016.
[7] S. B. Ajit Kumar Mahapatra, "Text mining: An introduction to theory and some applications," 2011.
[8] H. Z. Stephen Robertson, *The Probabilistic Relevance Framework: BM25 and Beyond*. Editorial Board, 2009.
[9] B. Liu, "Opinion mining," 2010.
[10] J. G. Carlos Henríquez Miranda, "Una revisión sobre el análisis de sentimientos en español," 2015.
[11] J. S. Wang, "Topic modeling: A complete introductory guide," 2017.
[12] Z. C. Massung S, Geicle C, "Meta: An unified tool-kit for text retrieval and analysis," 2016.