Tendencias de Tecnología en Twitter y periódico El Tiempo

Nefrety Sanchez
Ingeniería de Sistemas
Universidad Tecnológica de Bolívar
Email: nefretysanchez@gmail.com

Dayana Rodriguez
Ingeniería de Sistemas
Universidad Tecnológica de Bolívar
Email: lorenarodriquez@gmail.com

Yulissa Restrepo
Ingeniería de Sistemas
Universidad Tecnológica de Bolívar
Email: yulissatatiana@gmail.com

Valentina Correa
Ingeniería de Sistemas
Universidad Tecnológica de Bolívar
Email: valencorreabarco@hotmail.com

***Abstract*—In this document, we explain the analysis of trends in the social network Twitter and in the newspaper "El Universal", where trends in the area of technology are searched. For this, an extraction of the data was carried out, for a subsequent creation of the corpus, then a pre-processing of the texts, transformation and finally some results and / or graphs and an analysis were produced. In addition to the above, the description of the problem, the objective with this analysis, its conclusions and the address of the repository on GitHub were made.**

Keywords: News, twitter, newspaper, trends, technology, Data Preprocessing, Unigrams, Bigramas, N-Grams

I. Descripción del problema

La información que se encuentra en páginas web junto con las redes sociales es muy diversa provocando que a partir de un comentario sobre algún tema se haga tan viral que genere una gran tendencia e incluso recomendarte temas de los cuales se habla como lo es twitter en su #tendencia en el cual te muestra de lo que recientemente se habla en los tweets de las personas alrededor del mundo y en páginas web el top de noticias más recientes y frecuentadas.

El objetivo de la predicción tanto en twitter como en la página web "El tiempo" es encontrar aquellos temas tendencias en el área de la tecnología para su posterior análisis.

II. Dirección del repositorio

El código fuente del proyecto a tratar en el siguiente artículo se encuentra en el siguiente repositorio de Github:

https://github.com/YuliRestrepo/ProyectoBigData

III. Extracción de datos (Web)

Para la extracción de datos fue utilizada la técnica conocida como Web scraping es una forma de copia al texto fuente de las páginas web, está escrito en el lenguaje de HTML (Hypertext Markup Language), estos códigos fuentes en HTML son información legibles para los humanos y para las máquinas, también llamados tags o etiquetas el cual nos sirve para el escarbado de información, para definir y extraer, los datos que son recopilados y copiados de la web, generalmente para recogerlos en una base de datos local central, para su posterior recuperación o análisis.

A. Extracción/Creación del corpus

Se llama corpus a un conjunto de documentos sin clasificar son textos de dominios pero no se encuentran determinados a que van hacer utilizados, los datos son provenientes de una fuente de datos HTML y están organizados con unos identificadores: texto, fuente,identificador correspondiente a valor único que diferencie a un registro de los demás.

Para extraer el texto usamos una técnica llamada web scraping dado que nuestro conjunto de documentos son artículos de una página de internet. Los datos extraídos se guardan en un csv. Además se importaron librerías como:

- Urllib.request: esta librería nos ayuda a abrir la página web para leer el contenido en html.
- Beautifulsoup: es una librería de python que permite analizar documentos HTML y extraer datos de ellos, compensando imperfecciones que puedan existir,por ejemplo, permite extraer los atributos href de las etiquetas de anclaje donde esta ayuda a evaluar los tag, attribute del link
- Selenium.webdriver: esta librería nos ayuda a mostrar el código fuente en el formato de html, la variable es un objeto que extrae las las etiquetas de anclaje, se guarda en la variable tags para que contenga todas las etiquetas que contengan hipervínculos y para cada una de estas se recorren por un bucle for y se imprime el contenido del atributo href o puede ser los ítem, etc.
- seaborn as sns: Esta librería nos ayuda a tener mejor visibilidad de los gráficos, está basada en matplotlib y proporciona una interfaz de alto nivel y nos proporciona las herramientas necesarias para la represenar las gráficas.

- sklearn.feature_extraction.tex: nos ayuda a preparar el texto.La biblioteca scikit-learn ofrece herramientas fáciles de usar para realizar tanto la tokenización como la extracción de características de sus datos de texto.
- TfidfVectorizer: nos ayuda a convertir el texto a vectores de frecuencia de palabras

Al final se obtienen los sublinks de las páginas de noticias y su respectivo contenido y serán guardados en un documento de texto.

B. Preprocesamiento de los textos

Pre-procesar el texto es limpiar nuestro texto de ruido y estandarizar nuestro texto para que sea fácil de comparar. Algunos de estos pasos incluyen quitar las puntuaciones, transformar el texto a minúscula, o eliminar palabras frecuentes usadas en el lenguaje como preposiciones, artículos, etc. que por sí mismas no aportan mucha información, además de usar la raíz de las palabras.

C. Transformación

Para el análisis, se le agregó la librería pandas encargada de enviar a un dataframe para que se almacene y se pueda hacer un mejor uso de la información obtenida.

- import pandas: Pandas es una herramienta de manipulación de datos
- import numpy: Trae integradas muchas funciones de cálculo matricial de N dimensiones, así como la transformada de Fourier, múltiples funciones de álgebra lineal y varias funciones de aleatoriedad.
- import matplotlib.pyplot as plt: es una librería para generar gráficas a partir de datos contenidos en listas, vectores, en el lenguaje de programación Python y en su extensión matemática NumPy.

Para el análisis de noticias nuestra fuente primaria fue el tiempo, se definen las variables que se quieren extraer que son las subpáginas de la página del tiempo que es un periodico de libre uso, toda la información que se depositada es accesible en cualquier momento, la técnica utilizada fue RSS (Really Simple Syndication) es un emisor de información que se utiliza y fue definido para masificar información de manera estándar la cual se emite por medio de los formatos rss o xml, este formato nos permite ver el top de noticias de forma plana, la forma de recolectar toda la información es conocida como corpus, este se define como la colección de documentos de un dominio particular, en este caso nuestro corpus es específicamente de noticias, porque solo estamos extrayendo fuentes de un periodico en la subpágina de tecnologia, para esto se definió una clase rss donde nos recibe la url y nos retorna un conjunto de artículos que se convierte en nuestro corpus, importamos las librerías y creamos las funciones para tener una comunicación vía web, se buscan los artículos que son las fuentes y los enlaces de donde se va a traer la información, que busque titulo, link, descripción y la fecha de publicación, dentro de esa búsqueda luego se busca los otros ítems que son como los hijos de esa noticia del articulo principal eso se almacena en un diccionario específico para la extracción, se almacena en una lista.

Para esto también es necesario definir los links de las páginas que queremos extraer en el formato xml, para obtener las páginas hija de esta, luego usamos el Tfid para preparar todo el texto extraído y hacer una bolsa de palabras, para que nos ayude con el recuento, para hallar la frecuencia inversa o de los términos que seria la relevancia de la palabra dentro de la web, luego para encontrar el tópico de tendencias con los resultados que nos arroja el Tfid sacamos la media, la moda, mediana, y tomamos de los datos que están más cerca de la media que son los datos que son más frecuentes que nos representa nuestro tópico, luego analizamos los trigrams, bigrams y unigrams, para obtener el top de tendencia de la web y los que estén más cercano a la media son los nuevos tópicos posibles.

Seguido, se agrupó los resultados en un dataframe con las palabras y los resultados del TF-IDF, para calcular su media, moda y mediana. Al calcular la media, sacamos un rango de las palabras que está más o menos 0.00033, por encima o por debajo de la media, para poder sacar el top 10 de las palabras.

D. Resultados y análisis

Para los resultados de la web en la página ”El Tiempo”, se obtuvo el top 10 de los unigrams, bigrams y trigrams encontrados en el dataset, los cuales manejaron los siguientes datos estadísticos para cada uno de los n-grams que se estan analizando en este artículo:

TABLE I
DATOS ESTADÍSTICOS WEB

	Unigrams	Bigrams	Trigrams
Mediana	0.001120	0.003699	0.006020
Moda	0.00064	0.0037	0.00602
Media	0.002963	0.006178	0.007913

En el top 10 de los unigrams, tenemos:

TABLE II
UNIGRAMS WEB

ID	Palabra
2888	marco
2972	país
2968	bancos
3048	reemplazar
3081	laboral
3050	indican
3078	universidad
2962	cambien
3044	oficios
2960	dejen

Dentro del las palabras obtenidas en el unigrams podemos notar que hay una gran tendecia hablar de temas como marco, país, bancos y hasta de universidad.

En el bigrams y su top 10 tenemos:

TABLE III
BIGRAMS WEB

ID	Palabras
9	tiempo las
1	tecnosfera cuáles
3	contenido las
5	contenido los
7	tecnosfera medellín
2	tiempo firmas
4	información firmas
6	tiempo los
5525	tecnosfera los
8	tecnosfera las

Dentro de los bigramas se puede observar, muchos temas tienen tendencia pero podemos ver que los más hablados son información sobre firmas, tecnosfera, tiempo entre otros.

Y por último encontramos el top 10 de los trigrams, el cual está dato por:

TABLE IV
BIGRAMS WEB

ID	Palabras
10	disponibletecnósferatwitter tecnósferaet los
17	ios tecnósferaet la
16	inteltecnósferatwitter tecnosferaet medellín
15	tecnosferaet medellín presenta
12	mejor contenido los
11	contenido los señalamientos
20	tiempo las profesiones
21	el tiempo las
6	contenido las claves
3	iphonetecnósferatwitter tecnosferaet cuáles

Dentro de los trigramas, se habla de presentación de tecnosfera en Medellín, contenido de las claves, iphone, twitter, tecnosfera, entre otros temas.

IV. EXTRACCIÓN DE DATOS(TWITTER)

A. Extracción/Creación del corpus

Dentro de la extracción de los tweets en el #tecnologia se usa en tweepy, es una libreria de la API de twitter para python. Para esto se necesita tener unas keys y tokens correspondientes, autenticando estas con la cuenta de desarrollador para tener el acceso a los tweets con OAuthHandler, usando tweepy.Cursor(), la cual lleva unos parámetros específicos para hacer su uso como api.search, el buscador específico del hashtag, el idioma y la ubicación para que no agarre tweets de diferentes lugares del mundo. Los tweets extraídos se guardan en un csv. Además de librerías como:

- import sys: este es un módulo que se encarga de proveer variables y funcionalidades, directamente relacionadas con el intérprete.
- import pandas: es una librería destinada al análisis de datos, que proporciona unas estructuras de datos flexibles y permiten trabajar con ellos de forma eficiente.
- import unicodedata: Este módulo proporciona acceso a la base de datos de caracteres Unicode (UCD), que define las propiedades de todos los caracteres Unicode.
- import re: Este módulo proporciona operaciones de coincidencia de expresiones regulares similares a las encontradas en Perl.Tanto los patrones como las cadenas de texto a buscar pueden ser cadenas de Unicode (str) así como cadenas de 8 bits (bytes).
- import preprocessor: es una biblioteca de preprocesamiento para datos de tweets escritos en Python.
- import nltk: módulo de Python que contiene muchas funciones diseñadas para su uso en el análisis lingüístico de documentos y en el procesamiento de lenguaje natural

B. Preprocesamiento de los textos

Los diferentes tweets llegan con muchos aspectos que obstaculizan el análisis limpio del texto, por ello hacemos uso de algunas funciones que nos ayudan a limpiar todo el texto como las etiquetas, condiciones con “utf-8” pero aun asi las etiquetas también se eliminan para que esté más limpio el texto final.

C. Transformación

En el análisis se usó las librerías:

- from sklearn.feature_extraction.text import TfidfVectorizer: Tfidfvectorizer convierte una colección de documentos sin procesar en una matriz de funciones TF-IDF.

Para el análisis, el texto que fue mejorado en el preprocesamiento se le agregó la librería pandas encargada enviar a un dataframe para que se almacene y se pueda hacer un mejor uso de la información de los tweets suministrados por tweepy. Para poder encontrar y agrupar las palabras frecuentes necesitadas, se tuvo que usar unigrams, bigrams y trigrams.

A este modelo de agrupación de palabras anteriormente dichas se les llama unigrams, bigrams y trigrams dependiendo de cuantas palabras se agrupe, por como su nombre lo indica una solo abarca una palabra o letra, la siguiente dos palabras o letras y la tercera usa tres palabras o letras. Dentro del proyecto se hizo uso de estas para la organización de las palabras del texto usando la librería sklearn y algunas funciones para el cálculo del TF-IDF, dependiendo de la organización de las palabras.

Seguido, se agrupó los resultados en un dataframe con las palabras y los resultados del TF-IDF, para calcular su media, moda y mediana. Al calcular la media, sacamos un rango de las palabras que está más o menos 0.00033, por encima o por debajo de la media, para poder sacar el top 10 de las palabras.

D. Resultados y análisis

Para los resultados de twitter, se obtuvo el top 10 de los unigrams, bigrams y trigrams, encontrados en el dataset. Se sustentan en las siguientes tablas:

En el top 10 de unigrams tenemos:

TABLE V
UNIGRAMS TWITTER

ID	Palabras
1	describiendo
2	list
3	oss
4	seguridad
5	circulación
6	appandroid
7	amber
8	subestimar
9	sobreestimar
10	zara

Dentro del unigrams podemos notar, se hablan de diferentes temas, como lo son seguridad, apps de Android, actriz, entre otras cosas.

Aún así hay otros temas tendencias como apple, whatsapp, google que aunque no tienen gran incidencia también están en el top 10 son temas que tienen relevancia.

En el top 10 de bigrams, las palabras encontradas se encuentran en la sigueinte tabla:

TABLE VI
BIGRAMS TWITTER

ID	Palabras
1	participara seminarios
2	autonomos cambiaran
3	cambiaran aspectos
4	aspectos importantes
5	importantes circulacion
6	circulacion trafico
7	trafico seran
8	cambios quieres
9	quieres wifi
10	casa atencion

Dentro de los bigramas se puede observar, muchos temas tienen tendencia pero podemos ver que los más hablados son participación en seminarios, circulación de trafico, wifi, entre otros.

Por último encontramos el Top 10 de trigrams:

Dentro de los trigramas que se obtuvieron en su mayoría se habla de seminarios de robótica, noticias de youtube, tik tok, construcciones de puentes, entre otros temas relevantes que se encuentran en el top 10.

TABLE VII
TRIGRAMS TWITTER

ID	Palabras
1	participara seminarios robotics
2	noticias youtube hara
3	tik tok via
4	tok via compañía
5	usuarios superen estudiantes
6	superen estudiantes construyen
7	estudiantes construyen puentes
8	construyen puentes tubos
9	puentes tubos papel
10	tubos papel asignatura

V. CONCLUSIONES

Se puede observar que los procesos para la extracción de la información es muy diferente debido a que usan librerías específicas, usa funciones de limpieza para que el texto final este bien y su análisis sea el más eficiente. Dentro de twitter cabe destacar que hay muchas palabras que están dentro de los tweets que por más filtros se le aplique para eliminar las stopwords y las etiquetas hay cosas que agarra de manera repetida como verbos, adverbios entre otros, siendo mostrados con frecuencia en las gráficas.

REFERENCES

[1] Clase 8: PLN con Python. Instituto de Ingeniería UNAM. Recuperado de: http://www.corpus.unam.mx/cursopln/plnPython/clase8.pdf
[2] Moya, R. (2015). Pandas en Python, con ejemplos -Parte I- Introducción. Jarroba. Recuperado de: https://jarroba.com/pandas-python-ejemplos-parte-i-introduccion/
[3] Python. Documentación: re — Operaciones con expresiones regulares. Recuperado de: https://docs.python.org/es/3/library/re.html
[4] Python. Documentación: unicodedata — Base de datos Unicode. Recuperado de: https://docs.python.org/es/3.9/library/unicodedata.html
[5] The Python Package Index. Project description: tweet-preprocessor. Recuperado de: https://pypi.org/project/tweet-preprocessor/
[6] Unipython. Matplotlib: Funciones principales. Recuperado de: https://unipython.com/matplotlib-funciones-principales/
[7] Uniwebsidad. (2013). Python: 10.1. Módulos de sistema. Recuperado de: https://uniwebsidad.com/libros/python/capitulo-10/modulos-de-sistema