Twitter Sentiment Analysis

Programación Distribuida sobre
Grandes Volúmenes de Datos

Presentado por Marco Moresi

Diciembre 7, 2016

Contenido

Contenido

Introducción

Objetivo

- Familiarizarse con la libreria MLlib de Apache Spark
- Crear un pipeline de Machine Learning, el cual me permita clasificar Tweets según su sentimiento (Positivo y Negativo).

Contenido

Motivación

- Conocer lo que opinan los usuarios de Twitter, tiene varias aplicaciones tales como conocer la opinión sobre un producto recien lanzado al mercado, opinión sobre algún hecho en particular, entre otras.
- Además se puede utilizar para conocer la opinión pública sobre candidatos a las distintas elecciones (e.g. #Elections2016, #DonalTrump, #Hilary)

Contenido

Recolecion de Datos

- Se utilizó el dataset provisto por la cátedra el cual tiene las siguientes caracteristicas:
  - Idioma : Inglés
  - Tweets: 122.443
  - Tweets únicos: 38.055
  - Palabras en tweets únicos: 406.311
  - Palabras únicas: 72.792

Contenido

Preprocesamiento

- El dataset es cargado a un DataFrame, manteniendo toda la información que contiene cada Tweet.
- Con una Window Function, enumero los tweets para tener una referencia, en caso de ser necesario.
- Tokenizo los tweets con RegexTokenizer el cual me permite tokenizar a partir de una expresion regular, que me permite tomar las menciones(@usuario), hashstag(#tema) y las URL como tokens.
- Luego Remuevo Stop Words, las cuales no aportan al "sentimiento" del tweet.
- Vectorizo los tweets, y guardo el modelo que luego usaré mas adelante.

Contenido

Clustering

- Luego de vectorizar los Tweets, utilcé K-Means para generar clusters de Tweets tratando de dividir los tweets según el aspecto que tratan.

Contenido

Clasificador

- Se utilizaron dos datasets distintos para entrenar el clasificador Naive Bayes
  - Criticas de Cine
  - Lexicones
- En ambos casos se realiza la tokenización con Tokenizer, remuevo Stop Words y utilizo el modelo de CountVectorizer creado previamente para vectorizarlos.

Contenido

Configuracion de Parámetros

- Movie Reviews
  - Clustering: K=4, maxIter=10, seed=1
  - Visualización: Palabras que aparecen más de 50 veces
- Lexicon
  - Clustering: K=4, maxIter=10, seed=1
  - Visualización: Palabras que aparecen más de 50 veces

Contenido

Visualización

- Para la visualización utilicé D3.js, el cual me permite crear un "Bubble Chart" en cual se muestra la palabra (token), coloreada según la cantidad de ocurrencias en tweets clasificados de acuerdo al sentimiento.
  - Rojo en caso de aparecer mayormente en tweets Negativos
  - Verde en caso de aparecer mayormente en tweets Positivos

Demo

Conclusiones

- El funcionamiento del Clasificador para Tweets depende mucho del dataset con el que se lo entrene.
- No es tarea trivial procesar el lenguaje natural.
- Hay muchos aspectos que se dejaron fuera de las pruebas por cuestion de dificultad o tiempo.

- Posibles Mejoras
  - Mejorar la Clusterización
  - Esclarecer la Visualización
  - Utilizar otro Clasificador (SVM, Decision Tree)
  - Aumentar la cantidad de lexicons en el dataset
  - Parametrizar las frecuencias de los tokens tanto a nivel de tweet como de dataset

Gracias!!

Preguntas?