Estructuras de Datos

Trabajo Práctico Nº 2: Elecciones 2017

Fechas de entrega

turno martes: 24/10/2017

turno jueves: 26/10/2017

En este TP vamos a tratar de explotar los datos que se pueden obtener en Twitter.

La primera parte consiste en recopilar una buena cantidad de tweets para armar nuestro corpus para analizar.

Nos vamos a enfocar en el último tramo antes de las elecciones y vamos a buscar los tweets en tiempo real que hacen referencia a alguno de los candidatos a Senadores por la provincia de Buenos Aires durante los últimos días antes de votar (hasta el sábado 21/10 inclusive), no hace falta recopilar todos los tweets pero si una buena parte de ellos (no menos de 100.000 entre todos los candidatos) para poder analizarlos. Se requiere almacenar en archivos (se puede usar json, shelve o pickle), se debe almacenar solo el texto de los tweets y la fecha y hora en la que se obtuvieron.

Los candidatos y sus usuarios en twitter son:

Candidato	Twitter	Nombre de archivo
Cristina Kirchner	@CFKArgentina	cfkargentina
Esteban Bullrich	@estebanbullrich	estebanbullrich
Florencio Randazzo	@RandazzoF	randazzof
Luis Drozd	sin cuenta	-
Néstor Pitrola	@nestorpitrola	nestorpitrola
Pino Solanas	@fernandosolanas	fernandosolanas
Sergio Massa	@SergioMassa	sergiomassa
Víctor de Gennaro	@victordegennaro	victordegennaro
Vilma Ripoll	@vilma_ripoll	vilma_ripoll

Donde nombre de archivo, es el nombre con el cual deberán guardarse los tweets referentes a cada candidato.

Si un tweet menciona a más de un candidato se debe almacenar en todos los archivos correspondientes.

El programa que registra los tweets deberá leer los tokens de acceso de twitter del archivo config.py que se encuentra en el mismo directorio o carpeta que el código fuente. (No es preciso que entreguen los tokens correspondientes a las cuentas de twitter usadas en este trabajo). Se adjunta el archivo config.py

La segunda parte del TP consiste en analizar el corpus obtenido bajo las siguientes consignas:

- Ranking de candidatos más twitteados

- Ranking de candidatos más apreciados (al menos en twitter), para lo cual vamos a realizar un ensayo

usando un diccionario de afectos en español[i].

El diccionario de afecto contiene 7 campos: palabra, media_agrado, media_activacion, media_imaginabilidad, stdev_agrado, stdev_activacion, stdev_imaginabilidad

donde palabra es una palabra en castellano seguida por un guion bajo y una letra que sirve para clasificarla:

N: sustantivo (noun)

V: verbo

R: adverbio

A: adjetivo

media_agrado es un valor entre 10.000 y 30.000, 10.000 corresponde a una connotación negativa o desagradable y 30.000 positiva o agradable

Se cuenta con otro listado de palabras, STOP_WORDS, que no se deben analizar. Tampoco se deben analizar palabras de longitud menor a 3.

El algoritmo para determinar la puntuación de cada tweet de cada candidato es el siguiente:

para cada tweet t:

    score ← 0

    cuenta ← 0

    para cada palabra w en t:

        si longitud(w)<LONGITUD_MINIMA o w está en STOP_WORDS:

            continuar

        si w está en el diccionario:

            score ← score + media_agrado(w)

            cuenta ← cuenta + 1

devolver (score/cuenta)

El score de apreciación de cada candidato será el promedio del score de todos los tweets que lo menciona.

Antes de calcular el score se deberá preprocesar cada tweet de la siguiente forma (usando expresiones regulares):

- Pasar todo a minúscula

- Reemplazar las vocales acentuadas por las correspondientes vocales sin acentuar

- Reemplazar las direcciones web y de correo (por ejemplo: www.google.com.ar, http://www.google.com.ar o usuario@dominio, etc.) por la palabra en mayúscula URL (STOP_WORD)

- Reemplazar las menciones a usuarios (@xxxxx) por la palabra en mayúscula USER (STOP_WORD)

Respecto a los diccionarios se deberán normalizar todas las palabras reemplazando las vocales acentuadas por la correspondiente vocal sin acentuar, tanto en el diccionario de afectos como en el de STOP_WORDS. En el diccionario de afecto también se deben eliminar el guión y letra adicional que se utiliza para clasificar la palabra.

Se debe presentar un informe con lo siguiente:

- Decisiones de diseño
- Instrucciones de uso

- Corpus en crudo (archivos con los tweets de cada candidato, tal como se registraron de twitter)
- Corpus con tweets preprocesados según las consignas
- STOP_WORDS normalizadas (con vocales sin acentuar)
- Diccionario de afectos normalizado (con vocales sin acentuar)
- Código fuente

El informe se debe entregar impreso (decisiones, instrucciones y código fuente) y enviar (comprimido) por mail a rositaw@gmail.com y mfranzo@gmail.com

[i] Agustín Gravano & Matías Dell' Amerlina Ríos, "Spanish DAL: A Spanish Dictionary of Affect in Language", Reporte Técnico, Departamento de Computación, FCEyN-UBA, Febrero 2014 - http://www.aclweb.org/anthology/W13-1604 (Descarga: http://habla.dc.uba.ar/gravano/sdal.php?lang=eng).