Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava

Pokročilé databázové technológie

Zadanie č. 1 – Analýza tweetov

Cvičenie: Piatok 11:00 Ak. rok: 2021/22
Vedúci: Ing. Rastislav Bencel, PhD.
Odovzdáva: Bc. Denis Krajčirik

Zadanie 1 – analýza tweetov

Odovzdanie do 10.10.2021 23:59 – máte na to presne 2 týždne – dostanete za to 7,5 boda.

Prvé zadanie je zamerané na roztriedenie tweetov medzi rôzne konšpiračné teórie uvedené pod zadaním. Programovať môžete v hocijakom jazyku a váš zdroják sa odovzdáva to IS, no rovnako MUSÍ byť zavesený na vašom githube – v dokumente na začiatku uveďte vždy linku na projekt. Ak nebude GitHub, nebudú body. Okrem zdrojáku odovzdávate aj dokument kde budú screenshoty vašich výsledkov, výsledky ako text a grafy. Môžete napísať aj nejaké teplé slovko k tomu, nech sa pobavíme.

Vašou úlohou je teda:

1. Importovať dataset do PostgreSQL 13+: https://drive.google.com/drive/folders/1_crPerzWU2Nzc4mR5k6xuGqbp0fJDvsY?usp=sharing

2. Vypočítať sentiment pomocou VADER lexikónu https://towardsdatascience.com/sentimental-analysis-using-vader-a3415fef7664 pre tweety, ktoré obsahujú nasledovné hashtagy: #DeepstateVirus #DeepStateVaccine #DeepStateFauci #QAnon #Agenda21 #CCPVirus #ClimateChangeHoax #GlobalWarmingHoax #ChinaLiedPeopleDied #SorosVirus #5GCoronavirus #MAGA #WWG1WGA #Chemtrails #flatEarth #MoonLandingHoax #moonhoax #illuminati #pizzaGateIsReal #PedoGateIsReal #911truth #911insidejob #reptilians Pri výpočte nezohľadňujte (ignorujte) emotikony ani hashtagy ani mentiony v texte.

3. Roztriediť vyfiltrované tweety z predošlého zadania medzi konšpiračné teórie – spravte si na to novú tabuľku a mapovanie – nech je jasné ktorý tweet patrí ktorej konšpiračnej teórii.

4. Vypočítajte pomer extrémnych a neutrálnych sentimentov tweetov pre konšpiračné teórie po týždňoch a zistite, či daná konšpiračná teória rastie alebo upadá v čase. Výstup vizualizujte v grafe. Rovnako uveďte aj absolútne čísla: tweet_count, tweet_extreme_count, tweet_neutral_count v tabuľke pre každý týždeň. Za extrém považujeme keď je compound väčší ako 0,5 alebo menší ako -0,5.

5. Nájdite TOP10 account-ov ktoré sú najaktívnejšie v každej konšpiračnej teórii s extrémnym sentimentom a ukážte ich v tabuľke: id, name, screen_name a tweet_count.

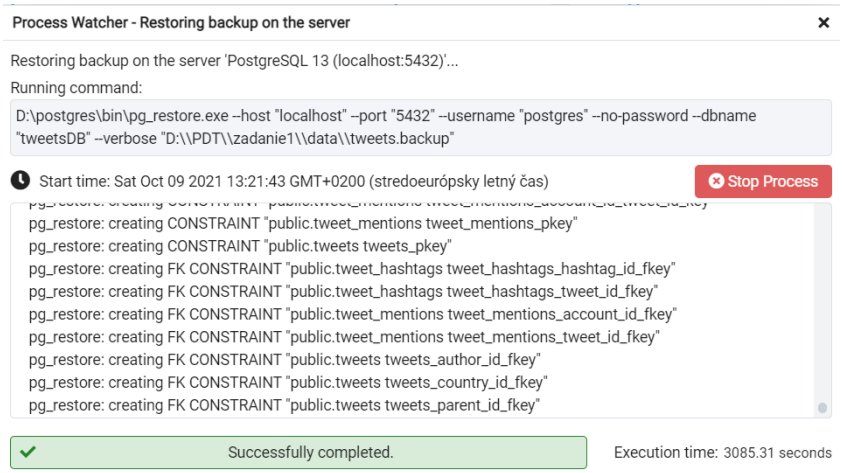
6. Nájdite TOP10 najčastejšie používaných hashtagov pre každú konšpiračnú teóriu z tweetov s extrémnym sentimentom, vypíšte aj počet.

Riešenie:

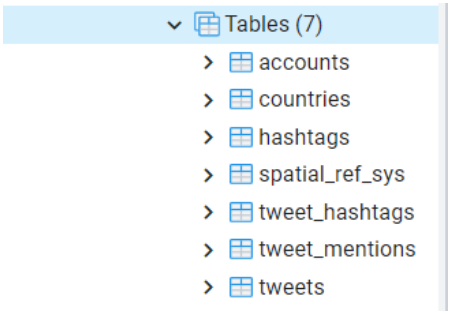
Riešenie zadania sme sa rozhodli implementovať v jazyku Python v Jupyter notebooku. Na interakciu s databázou sme používali knižnicu sqlalchemy.

Úloha 1:

Dáta sme importovali cez rozhranie pgAdmin4, kde sme si vytvorili najprv novú databázu, ktorú sme pomenovali tweetsDB. Následne sme cez *Restore* naimportovali dáta z dumpu do db. Dáta sme importovali na HDD disk. Import v tomto prípade trval 51 min.



Po importovaní máme nasledujúce tabuľky:



Úloha 2:

Pri riešení úlohy 2 sme postupovali tak, že sme si najprv pridali potrebné stĺpce do ktorých sme ukladali jednotlivé hodnoty pre sentiment. Potom pomocou filtra ktorý nám „matchuje“ cez select query všetky záznamy ktoré v sebe obsahujú hashtag uvedený v zadaní úlohy 2. Pri riešení ďalších úloh najmä mapovaniu hashtagov na konkrétnu konšpiračnú teóriu sme si všimli, že pri jednej konšpiračnej teórií je uvedený hashtag, ktorý nebol spomenutý v zadaní úlohy 2 (COVID19 and microchipping - #BillGAtes), rozhodli sme sa preto ďalej s týmto hashtagom nepracovať ani v úlohe 2 ani v ostatných úlohách. Akonáhle sme získali všetky tweety, ktoré v sebe obsahovali daný hashtag, tak sme parsli content tweetu podla

nasledujúceho regexu a emoji patternu, ktorý nám odstránil všetky emoji a aj iné metacharaktery z contentu.

Regex

```
"[#@]+[\w.-]*"
```

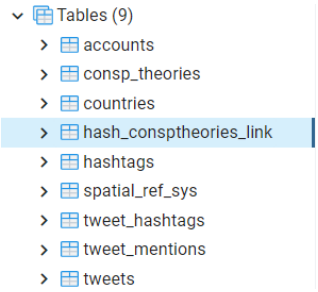
Emoji pattern

```
u"\U0001F600-\U0001F64F"  # emoticons
u"\U0001F300-\U0001F5FF"  # symbols & pictographs
u"\U0001F680-\U0001F6FF"  # transport & map symbols
u"\U0001F1E0-\U0001F1FF"  # flags (iOS)
# u"\U00002500-\U00002BEF"  # chinese char
u"\U00002702-\U000027B0"
# u"\U00002702-\U000027B0"
# u"\U000024C2-\U0001F251"  # mandarinian chars
u"\U00010000-\U0010ffff"
u"\U0001f926-\U0001f937"
u"\u2600-\u2B55"
u"\u2640-\u2642"
u"\u200d"
u"\u231a"
u"\u23e9"
u"\u23cf"
u"\ufe0f"  # dingbats
u"\u3030"
```

Po parsovaní contentu z tweetu sme použili *SentimentIntensityAnalyzer()* z **nltk.sentiment.vader** knižnice na výpočet sentimentu. Ako posledné sme daný sentiment zapísali do príslušných stĺpcov **neg, neu, pos** a **compound** na základe id daného tweetu.

Úloha 3:

V úlohe č. 3 sme sa najprv vytvorili v pythone dataframe tabuľku, ktorá reprezentovala našu tabuľku konšpiračných teórií, ktorú sme neskôr vytvorili v databáze a zapisovali do nej konkrétne konšpiračné teórie. Taktiež sme spolu s každým záznamom konšpiračnej teórie insertly aj jeho id, ktoré sme si vlastne vygenerovali a takéto dvojice sme pridávali to tabuľky. Potom ako sme mali tabuľku konšpirač. teórií pridanú vytvorili sme si mapovaciu tabuľku, ktorá nám mapovala daný hashtag na konšp. teóriu. Do mapovacej tabuľky sme pridávali trojice (PK: id, FK: hashtagID a FK: conspTheoryID). Po pridaní tabuliek vyzerali naše tabuľky nasledovne:



Príklady obsahu dát v pridaných tabuľkách:

Tabuľka konšp. teórií

	id [PK] integer	theory_name character varying (255)
1	1	Pizzagate conspiracy theory
2	2	FlatEarth
3	3	9/11 was inside job
4	4	Reptilian conspiracy theory
5	5	Qanon
6	6	The virus escaped from a Chinese lab
7	7	New world order
8	8	COVID19 is preaded by 5G
9	9	GLobal Warming is HOAX
10	10	Illuminati
11	11	Moon landing is fake

Mapovacia tabuľka

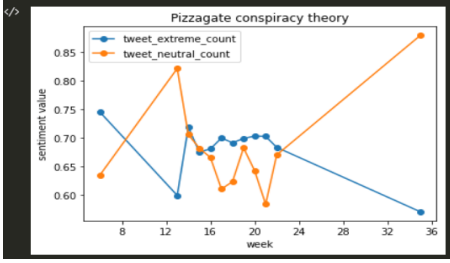
	id [PK] integer	hashtag_id integer	consp_theory_id integer
1	1	14415	5
2	2	18287	5
3	3	20506	5
4	4	24818	5
5	5	25089	5
6	6	25091	5
7	7	27195	5
8	8	33792	12
9	9	59789	5
10	10	29024	5
11	11	66445	7

Úloha 4:

Úlohu č.4 sme riešili tak, že sme si selectly najprv tweety s extrémnym sentimentom a následne, ktoré boli prepojené s tabuľkami tweet_hashtags, hashtags s našou mapovaciou tabuľkou aby sme namapovali tweety na konšp. teóriu. Následne sme si extrahovali týždeň z timestampu kedy tweet vznikol a sním sme si selectli meno konšp. teórie, týždeň, compound a neutral daného tweetu. Toto tvoril náš subselect. Z toho čo nám subselect vrátil sme vyberali meno teórie, týždeň, tweet_count (počet tweetov za týždeň) a ešte hodnoty zodpovedajúce priemeru compoundu podľa daného týždňa a priemeru neutralu podľa daného týždňa . Toto sme ešte grouply podľa týždňa a názvu konšp. teórie. Následne sme tieto výsledky usporiadali a vizualizovali.

Ukážka vizualizácie

```
#######
---  Pizzagate conspiracy theory  ---
######
 week  tweet_count  tweet_extreme_count  tweet_neutral_count
 6           1            0.745600            0.635000
13           2            0.599400            0.822000
14          35            0.718940            0.706000
15         115            0.675477            0.682313
16         148            0.680988            0.665919
17          47            0.700043            0.610596
18         186            0.691783            0.623608
19          28            0.699218            0.682714
20          87            0.703439            0.642908
21          65            0.703023            0.584108
22          71            0.683690            0.670394
35           1            0.570700            0.880000
```



Úloha 5:

Pri úlohe 5 sme zozačiatku postupovali rovnako ako pri 4, avšak okrem pôvodných joinov sme najoinovali tweety aj s tweet_mentions a accounts. Potom sme si už len grouply dáta nad menom konšp. teórie, idčko accountu, account.name, account.screen_name, výsledky sme si zoradili podľa mena teórie a ako posledné sme si selectly atribúty ako meno konšp. teórie, id accountu, account.name, account.screen_name a tweet_count, čo je počet tweetov zodpovedajúcich danému accountu.

Ukážka výstupu

```
    #######
    ---  Pizzagate conspiracy theory  ---
    ######
           id            name    screen_name  tweet_count
1071777608    John Podesta    johnpodesta            1
1339835893 Hillary Clinton HillaryClinton            1

    #######
    ---  FlatEarth  ---
    ######
                  id                name     screen_name  tweet_count
929387229880946688        Jack William   Jackszooquest            4
          25073877     Donald J. Trump realDonaldTrump            2
          10228272             YouTube         YouTube            1
          17471979 National Geographic          NatGeo            1
          38190348              Daniel crimescenevegas            1
          40053694    Flavio Bolsonaro FlavioBolsonaro            1
          68712576    Carlos Bolsonaro CarlosBolsonaro            1
          74756085 Eduardo BolsonaroBR     BolsonaroSP            1
         120910874     Hidden Mountain HiddenMountain7            1
         128372940   Jair M. Bolsonaro   jairbolsonaro            1
```

Úloha 6:

V úlohe 6 sme použili rovnaký select ako v úlohe 4 no tu sme groupovaly podľa mena konšp. teórie a id hashtagu. Vybrali sme si atribúty meno konšp. teórie, hashtagID, hashtag hodnota, tweet_count, čo je počet rôznych tweetov pre daný hashtag. Ako posledné sme výsledky usporiadali podľa mena teórie a tweet_countu v zostupnom poradí.

Ukážka výstupu

```
    #######
    ---  Pizzagate conspiracy theory ---
    ######
        hashtag_val  tweet_count
PIZZAGATEISREAL          356
PizzaGateIsReal          279
pizzagateisreal           43
 PEDOGATEISREAL           38
PizzagateIsReal           37
Pizzagateisreal           16
 pedogateisreal           14
PizzagateisReal            2
 PedogateIsReal            1

    #######
    ---  FlatEarth ---
    ######
          hashtag_val  tweet_count
         FlatEarth           47
         flatearth           23
   researchflatearth         11
         FLATEARTH            6
        FlatEarther           3
    flatearthsociety          3
```

Zhodnotenie:

Riešenie zadania za seba hodnotím v celku OK, čo sa týka správnosti riešenia tak podľa vecí aké som overoval mi vychádza, že moje riešenie by malo byť správne. Čo sa týka behu programu tak ak nerátam import cez pgAdmin, tak vcelku ok dokopy cca hodina ale to je spôsobené najmä poslednou query v ktorej bol zložitý subselect ten som nahradil jednoduchým selectom, ktorý robil to isté čiže ak odrátam čas vykonávania toho poopraveného selectu, tak celý beh programu trval cca okolo 30-40 min. Ak sa pozrieme na pamäť, tak čo som si všimol tak na celý beh programu v jupyteri s veľa výpismi mi „odkuslo“ cca okolo 4-5 GB pamäte max. Celkovo viac som sa snažil spraviť zadanie správne ako optimalizovať dobu behu programu. V riešení úloh som sa snažil, čo možno najviac používať sql, keďže som myslel, že určité veci ako filtrovanie a zoraďovanie dát bude rýchlejšie v sql ako v pythone. Zadanie ako také bolo celkom zaujímavé no miestami mi vadili niektoré nejasnosti ohľadom úloh. Pozitívne hodnotím aj to, že som si po dlhej dobe zopakoval sql, negatívne max tak dlhý import dát a čakanie na výsledky určitých querín.