Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Neo4j

Martin Schön, Bc.

AIS ID: 103121

E-mail: xschon@stuba.sk

GitHub repozitár: https://github.com/FIIT-DBS/zadanie-pdt-xSchon

Predmet: Pokročilé databázové technológie

Zimný semester 2022/2023

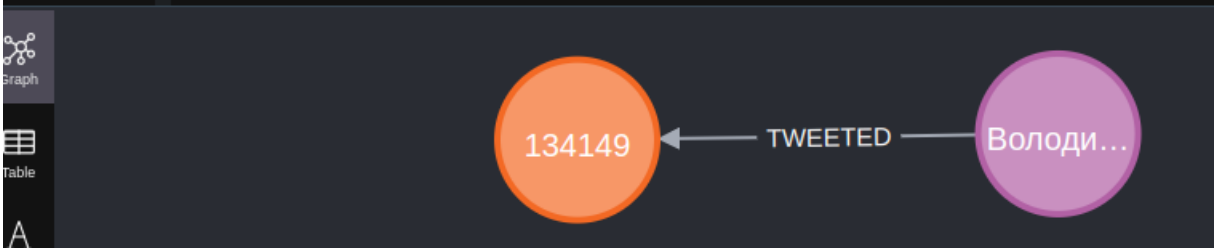
Úvod

Zadanie bolo vypracované s využitím Neo4j Desktop aplikácie, verzie 1.5.2, na OS Ubuntu 22.04 LTS. Neo4J databáza bola vo verzii 4.4.5, pričom bežala lokálne.
Queries použité a screenshotnuté v tomto dokumente sú priložené v súbore queries.txt. Priložil som aj výsledky - v priečinku results sa nachádzajú .json aj .png súbory jednotlivých zadaní. Meno súboru je vždy z04_<číso_úlohy>_schon.<json|png>.
Všetky tieto náležitosti, rovnako ako táto dokumentácia, sú dostupné na githube spomenutom na úvodnej strane tohto dokumentu. Zadanie som vypracoval samostatne, s využitím vlastných vedomostí nadobudnutých na predmete. Verím, že pri čítaní tejto dokumentácie budete mať príjemný čas.

1. Vloženie autora s tweetom

K tejto úlohe bolo potrebné pristúpiť postupne. Najskôr som zistil, ktorý tweet má najviac odpovedí spomedzi všetkých tweetov Zelenského. To sa mi podarilo docieliť nasledovnou query:

```
MATCH (author:Author {username: 'ZelenskyyUa'})-[rel:TWEETED]-
(tweet:Conversation)<-[rel2:REPLIED_TO]-(reply:Conversation)
RETURN author, tweet, COUNT(rel2) AS topt
ORDER BY topt DESC
LIMIT 1
```



Vyberiem všetky tweety, ktoré Zelensky tweetol, spoločne s odpoveďami na ne - pomocou vzťahu TWEETED zistím Zelenským napísané tweety, ktoré majú odpovede skrze reláciu REPLIED_TO. Nad touto reláciou viem vykonať COUNT, ktorý mi vráti počet odpovedí jednotlivých tweetov a potom už cez ORDE BY DESC a LIMIT dostanem najodpovedanejší.

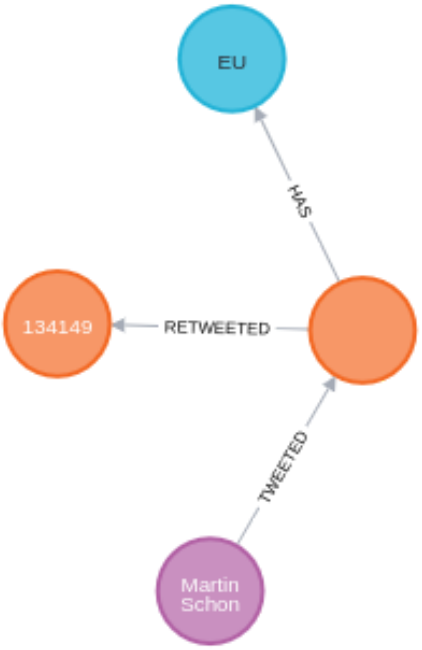
Takto získaný tweet použijem v hlavnej query tejto úlohy s využitím CALLu, ktorý zavolám ako prvý a teda dostanem tento tweet jednoducho do väčšej query. Hlavná query vyzerá nasledovne:

```
CALL {
MATCH (author:Author {username: 'ZelenskyyUa'})-[rel:TWEETED]-(tweet:Conversation)<-[rel2:REPLIED_TO]-(reply:Conversation)
RETURN tweet, COUNT(rel2) AS topt
ORDER BY topt DESC
LIMIT 1
}
MATCH (h:Hashtag {tag: 'EU'})
CREATE (me:Author {username : 'xschon', name: 'Martin Schon'})-[:TWEETED]->
(myTweet:Conversation {content : 'I wonder what data we would use if there was no war!', language : 'en'})-[:HAS]->(h)
CREATE (myTweet)-[:RETWEETED]->(tweet)
RETURN me, myTweet, h, tweet
```

Po získaní Zelenského tweetu získam aj mnou vybraný hashtag - v mojom prípade “EU” pomocou MATCHu. Takto mám prichystané náležitosti môjho tweetu.

V CREATE vytvorím môjho užívateľa s danými atribútmi. Ten má následne pridelenú

reláciu TWEETED na *novo vytvorený* tweet, ktorý je vytvorený v rámci tejto CREATE inštancie. Tweetu ešte pomocou HAS pridelím EU hashtag. Tweet má taktiež za úlohu retweetovať tweet, na to potrebujem druhý krát zavolať funkciu CREATE, ktorá mi umožní pridať RETWEETED vzťah medzi môj a Zelenského tweet.
Nakoniec ešte returnem novovytvorené hodnoty, aby som si vizuálne skontroloval, čo sa mi podarilo vytvoriť - nový užívateľ, nový tweet, ktorí referencujú už existujúce hodnoty.



2. Zlyhania influencerov

Táto úloha sa taktiež skladá z dvoch častí. V prvej z nich potrebujem získať 10 používateľov, ktorí sú najviac retweetovaní (majú najväčší celkový počet retweetov svojich tweetov).

To dosiahnem nasledovnou query:

```
MATCH (topa:Author)-[twtd:TWEETED]→(t:Conversation)←[rt:RETWEETED]-(c)
RETURN topa, COUNT(rt) AS ct
ORDER BY ct DESC
LIMIT 10
```

Táto podúloha je možná s využitím vzťahov TWEETED a RETWEETED naviazanými na konkrétne konverzácie. Skontrolujem všetky tweety každého autora a následne na ne naviažem všetky retweety. Následne vykonám agregačnú funkciu COUNT nad retweetmi. Zo všetkých údajov potom vytiahnem len takto vykonané COUNTy a autorov - automaticky mi prebehne zhluknutie všetkých ostatných stĺpcov spolu, čím sa mi spočíta počet všetkých retweetov každého autora (všetky COUNTy jedného autora sa spočítajú dokopy). Vo finále mi tak zostane iba autor a celkový počet jeho retweetovania. Takto získané dáta potom dosadím do hlavnej query s využitím funkcie CALL.

V hlavnej query potom zoberiem už **iba** týchto užívateľov a podobným štýlom opäť dostanem počty retweetov ich tweetov - tento krát ich ale zoradím od najmenšieho - teda dostanem najmenej retweetované tweety inak populárnych ľudí.

```
CALL {MATCH (topa:Author)-[twtd:TWEETED]→(t:Conversation)←[rt:RETWEETED]-(c)
RETURN topa, COUNT(rt) AS ct
ORDER BY ct DESC
LIMIT 10}
MATCH (topa)-[:TWEETED]→(twts:Conversation)←[r:RETWEETED]-(:Conversation)
RETURN topa, twts, COUNT(r) AS rts
ORDER BY rts
LIMIT 20
```

Výsledky sú nasledovné (vykresľujem si aj autorov, aj keď nie je treba):

Najmenej retweetovaný tweet (1x):

topa


```
{
  "identity": 2150224,
  "labels": [
    "Author"
  ],
  "properties": {
"tweet_count": 2056,
"following_count": 310,
"listed_count": 2879,
"followers_count": 306162,
"name": "Lesia Vasylenko",
"description": "Ukrainian MP, @goloszmin, working mom of 3 lovely humans,
lover of freedom, travel and all things green #Ukraine #geopolitics",
"id": 1219232377605644289,
"username": "lesiavasylenko"
  }
}
```

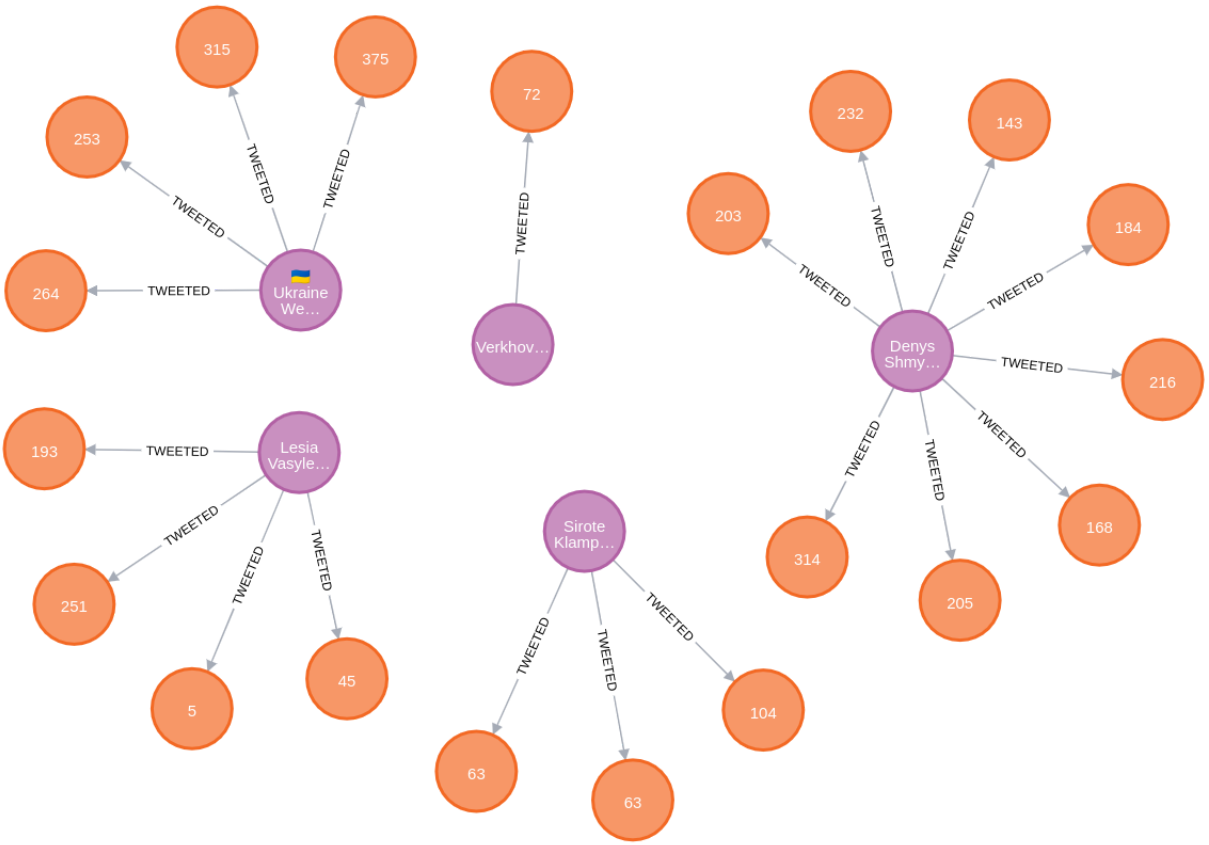

twts


```
{
  "identity": 8357005,
  "labels": [
    "Conversation"
  ],
  "properties": {
"like_count": 5,
"created_at": "2022-03-01T00:24:21",
"language": "en",
"source": "Twitter for iPhone",
"id": 1498438954592387073,
"reply_count": 0,
"author_id": 1219232377605644289,
"quote_count": 0,
"content": "@UEFA and @FIFAWorldCup ban #Russia from ALL football matches and break their sponsorship deal
with @GazpromFootball 🔥in January @GeorgeFoulkes had his resolution on football and values adopted in the
@PACE_News. International law in action💪",
"retweet_count": 1
  }
}
```


rts

1

Graficky znázornené:

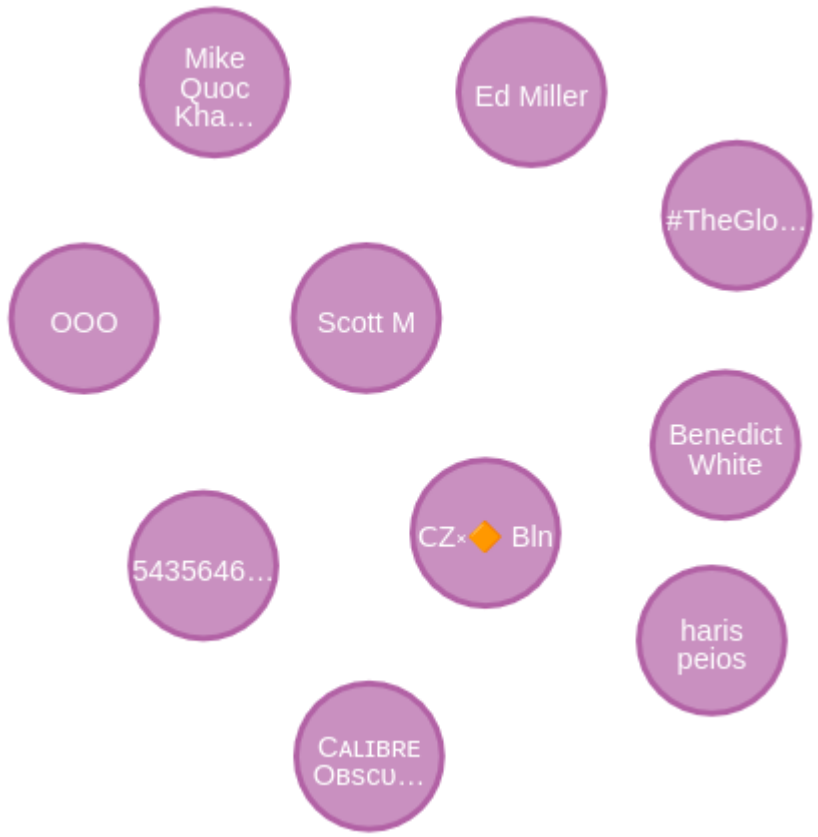


3. Odporučenie Mariosovi

V tejto úlohe najskôr získam všetky tweety, ktoré Marios retweetol - pomocou relácií :TWEETED získam všetky jeho tweety - z nich vyberiem tie, ktoré pomocou :RETWEETED odkazujú na iné konverzácie - tak získam pôvodné tweety retweetované Mariosom (referované ako twts). Pre tieto twts konverzácie potom nájdem ich retweety, ktoré môžu byť až do hĺbky 2. Toto je znázornené pomocou notácie :VZŤAH*min…max, v mojom prípade :RETWEETED*..2. To umožňuje grafovej databáze pozerať sa až do hĺbky 2 - teda na retweet retweetu tweetu twts. Potom mi už stačí zistiť iba autorov týchto retweetov - pomocou parametru :TWEETED namapujem autorov, ktorí ich retweetli (a následne vyhodím Mariosa).

```
MATCH (a: Author {username:"Marios59885699"})-[:TWEETED]→(twts:Conversation)-[:RETWEETED]→
(:Conversation)←[:RETWEETED*..2]-(shared:Conversation)←[:TWEETED]-(other:Author)
WHERE other.username <> "Marios59885699"
RETURN other, COUNT(shared) AS sha
ORDER BY sha DESC
LIMIT 10
```

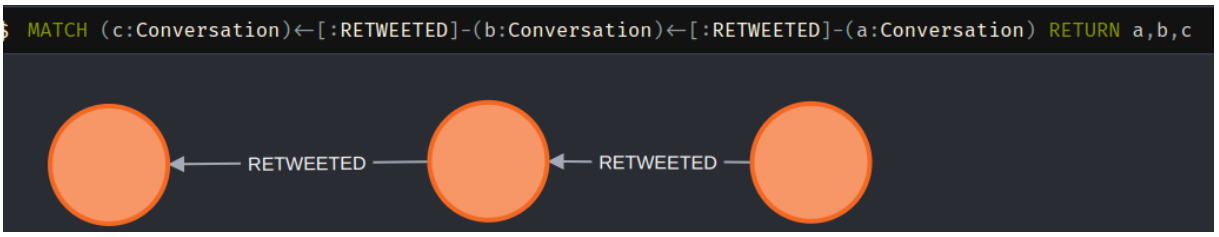
Grafické znázornenie výsledku 10 užívateľov.



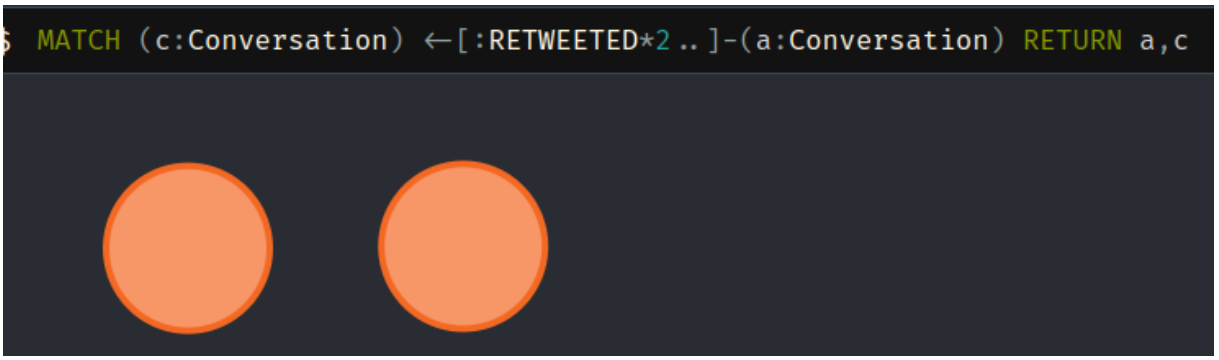
V tabuľke výsledkov pod parametrom sha vidíme, aj koľkokrát retweetovali rovnaký tweet, ako Marios - najviac to bolo 41 krát pre užívateľa CalibreObscura.

Poznámka k hĺbkovému hľadaniu:

V tomto datasete sa naturálne nenachádza žiadny tweet, ktorý by bol retweet retweetu. Ja som si jeden takýto prípad ručne vložil, aby som mohol otestovať funkčnosť notácie *min..max:



Ako vidno, neexistuje žiadny prirodzený retweet retweetu, ALE ak by sa v datasete taký nachádzal, tak moja query je pripravená rátať aj ten do odporúčaní pre Mariosa.

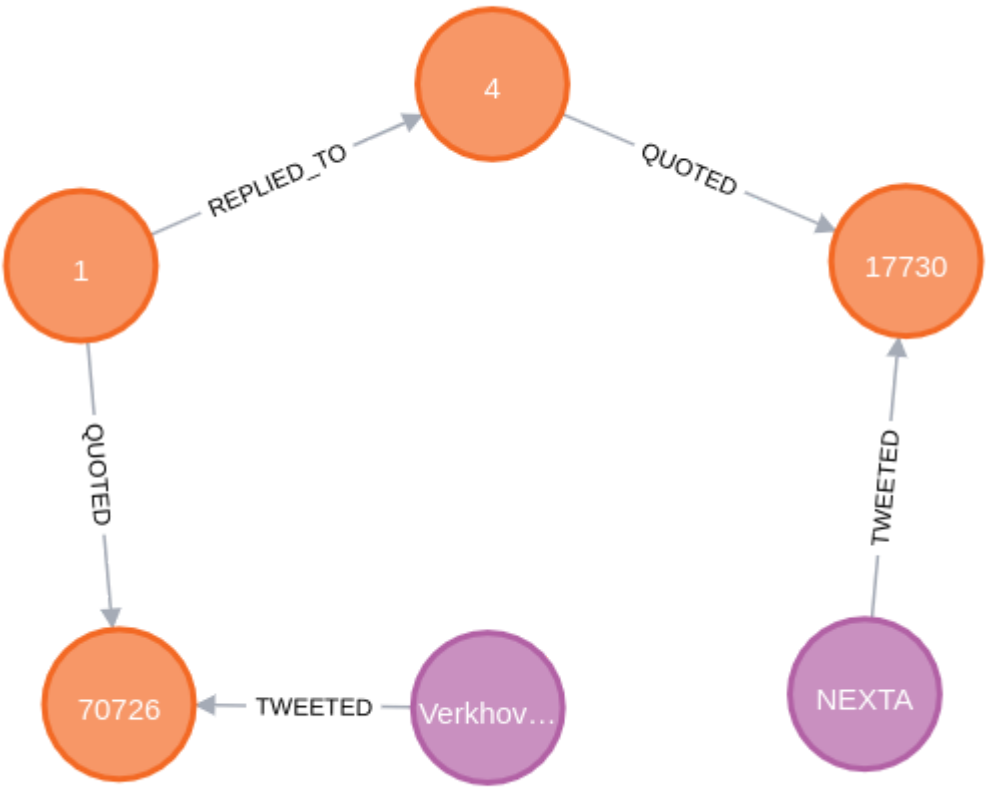


4. Najkratšia cesta ua parlamentu a NextaTV

Využil som len jeden MATCH príkaz s pomocou funkcie allShortestPaths, ktorá mi vráti najkratšie cesty (všetky cesty najkratšej možnej dĺžky) medzi dvoma nódmi spojenými definovanými reláciami. V mojom prípade sú nódy definované ako autori - jeden ua_parliament a druhý ako nexta_tv. Medzi nimi sa nachádzajú povolené konverzácie TWEETED, RETWEETED, REPLITED_TO a QUOTED. Na definovanie množiny vzťahov používam operátor | medzi rôznymi vzťahmi. Nakoniec definujem *..10, ktorá značí cestu maximálnej dĺžky 10 (konkrétne <1,10>). Podstatné je, aby relácie na nódy autorov mali obojstranné náväznosti, teda nepoužívam žiadne šípky (ako v prípade predchádzajúcich úloh), ale nechám obojstranný tok vzťahov.

```
MATCH p = allShortestPaths((:Author {username : "nexta_tv"})-
[:TWEETED|RETWEETED|REPLIED_TO|QUOTED*..10]-(:Author {username : "ua_parliament"}))
RETURN p
```

Výsledkom je jedna cesta dĺžky 5:

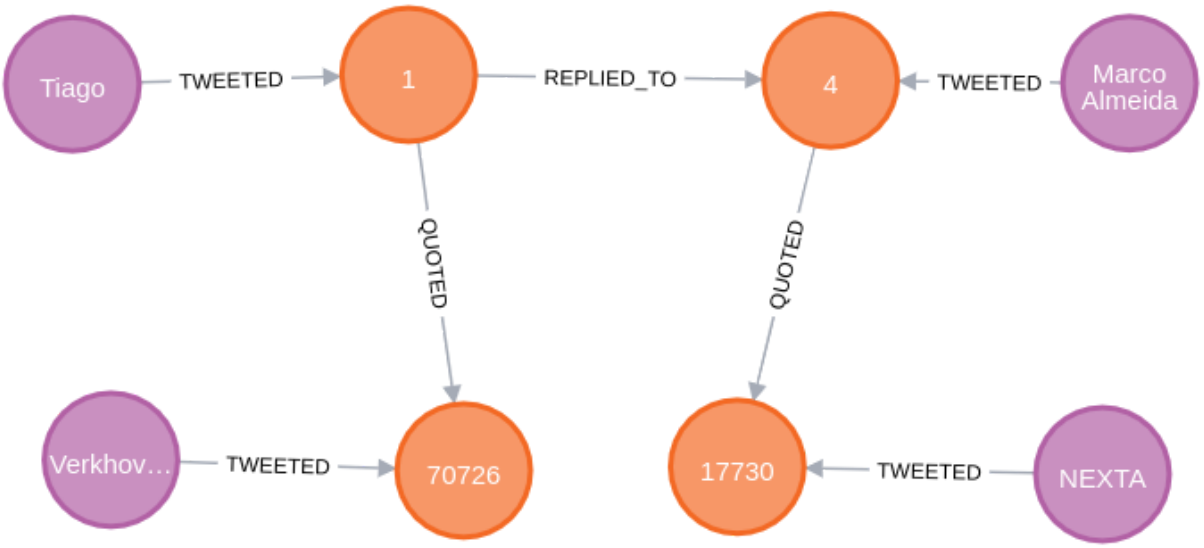


5. Autori v najkratšej ceste

V tejto časti použijem časť predchádzajúcej úlohy - vyberiem shortestPath (aby som dostal maximálne jednu cestu) a tú si CALLom dostanem do zvyšku query. Tam následne použijem UNWIND, ktorý mi pomôže transformovať path (ktorý dostanem zo shortestPathu) na jednotlivé hodnoty. Tie potom pomocou nodes ešte pretransformujem na samostatné nódy. Na tieto nódy potom dokážem pomocou :TWEETED namapovať ich autorov.

```
CALL {
MATCH sp = shortestPath((:Author {username : "nexta_tv"})-[:TWEETED|RETWEETED|REPLIED_TO|QUOTED*..10]-
(:Author {username : "ua_parliament"}))
return sp}
UNWIND nodes(sp) AS nodeLinks
MATCH (nodeLinks)<-[:TWEETED]-(a:Author)
RETURN nodeLinks, a
```

Obrázok cesty s autormi tweetov:



6. 10 najpoužívanejších hashtagov

V tejto úlohe sa vraciame k typu úloh, ktoré sú zložené z dvoch častí. V prvej z nich si nájdem 10 najpoužívanejších hashtagov celkovo - hashtagy sú na tweety naviazané cez vzťah HAS, spočítam teda počet týchto vzťahov jednotlivých hashtagov a vyberiem 10 najpoužívanejších z nich.

```
MATCH (h:Hashtag)<-[use:HAS]-()
RETURN h, COUNT(use) AS usage
ORDER BY usage DESC
LIMIT 10
```

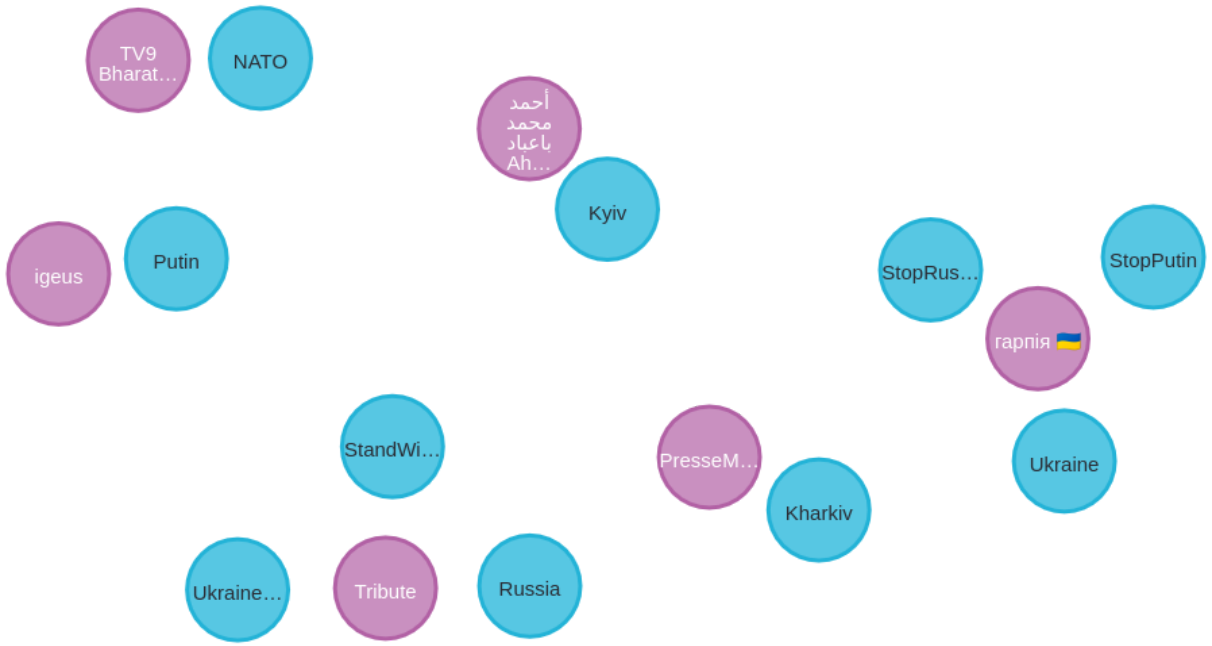
Tieto hashtagy potom cez CALL dostanem do hlavnej query. V nej nájdem všetky konverzácie, ktoré používajú tieto hashtagy a nájdem aj autorov týchto tweetov (cez vzťah TWEETED). Následne použijem WITH nad autormi, hashtagmi a počtom použití. Tak mi vznikne hUsag, v ktorej mám uložené informácie o tom, ktorý hashtag (z top 10) bol použitý ktorým autorom koľko ráz. Toto následne usporiadam podľa počtu od najväčšieho po najmenšie. Vo finále potom použijem výber hashtagu s pomocou collectu autora, odkiaľ vyberiem len prvého (najpočetnejšieho) autora, ako aj jeho počet použití hashtagu (cez max funkciu). Nakoniec ešte aj tieto výsledky zoradím podľa počtu použití, hoci to nebolo nevyhnutné.

```
CALL {MATCH (h:Hashtag)<-[use:HAS]-()
RETURN h, COUNT(use) AS usage
ORDER BY usage DESC
LIMIT 10}
MATCH (h)<-[husage:HAS]-(:Conversation)<-[:TWEETED]-(a:Author)
WITH a, h, count(husage) AS hUsag
ORDER BY hUsag DESC
RETURN h AS hashtag, collect(a)[0] AS author, max(hUsag) AS timesUsed
ORDER BY timesUsed DESC
```

Výstup potom vyzerá následovne:

hashtag	author	timesUsed
1 { "identity": 7, "labels": ["Hashtag"], "properties": { "tag": "UkraineRussiaWar" } }	{ "identity": 4294593, "labels": ["Author"], "properties": { "tweet_count": 5573, "following_count": 51, "listed_count": 0, "followers_count": 28, "name": "Tribute", "description": "", "id": 1221797851258163200, "username": "indiainstats1" } }	943

Na obrázku vyššie je možné vidieť najpoužívanejší hashtag jedným človekom - samozrejme, celý výsledok obsahuje top10 a nachádza sa v prílohe.
Grafické znázornenie potom neobsahuje vzťahy, pretože neexistuje priamy vzťah medzi userom a hashtagom, bolo by potrebné zobraziť konverzácie, alebo takýto vzťah vytvoriť. Snažil som sa posunúť hashtagy k ich autorom, znázornenie potom vyzerá takto:



To je k tomuto zadaniu všetko, keby boli nejaké časti nejasné, neváhajte ma kontaktovať.