Freie Universität Berlin

Datenbanksysteme SS17: Projekt
3. Iteration

Dozentin: Agnes Voisard

Bernadeta Chișărău, Dor Cohen, Mihai Renea

26. Juli 2017

---

1 Clusteranalyse

Für die Clusteranalyse haben wir den K-Means Algorithmus mithilfe der java-ml Library eingesetzt. Der Algorithmus partitioniert die Menge der Hashtags in 6 Clusters, wo jedes Hashtag ein 2-dimensionales Vektor mit den folgenden Metriken ist:

- Hashtag-Wichtigkeit – als Durchschnitt der Wichtigkeitswerten aller Tweets, die Hashtag h enthalten.
  Wichtigkeit $W_T(t)$ eines Tweets t:

$$W_T(t) = \sqrt[4]{\frac{t_{favorites\ count} + t_{retweet\ count}}{2}}$$

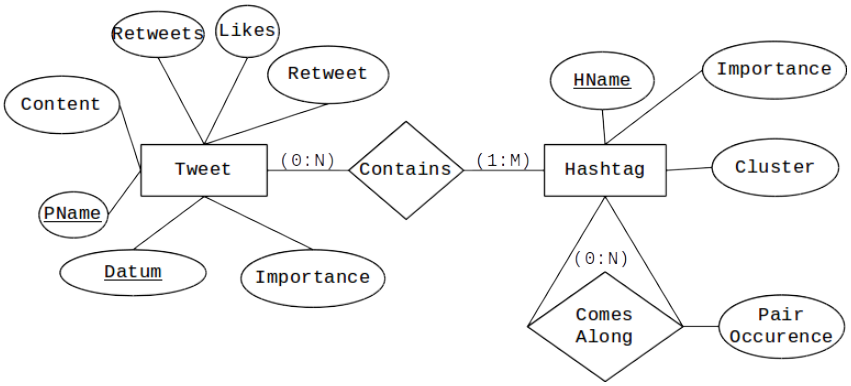
  Wichtigkeit $W_H(h)$ eines Hashtags h:

$$W_H(h) = \frac{\sum_{t \in T} W_T(t)}{|T|}$$

  Wo T die Menge der Tweets, die Hashtag h enthalten.

- Hashtag-Occurence – wie oft ein Hashtag insgesamt auftaucht.

Anschließend speichern wir die neu-erzeugten Informationen in einer neuen Tabelle, *hashtag*. Damit kann man für die Visualisierung die gebrauchten Werte einfach ablesen. Dadurch entsteht die aktuelle DB-Schema:



2 Datenvisualisierung

Um die Datenvisualisierung zu vereinfachen haben wir ein Programm geschrieben (Node_data_creator.java), das die Informationen aus der Datenbank in JSON-Dateien bereitstellt, und zwar:

- *plots.json* - Informationen für die Visualisierung des Hashtagnetzwerkes - Knotenpositionen (nodes-Array) und Verbindungen (edges-Array):

```
{
   "nodes": [
     {
       "color": "rgb(r,g,b)",
       "size": 100,
       "x": "x",
       "y": "y",
       "id": "hname",
       "label": "hname",
       "type": "tweetegy"
     },
     .
     .
     .
   ],
   "edges": [
     {
       "id": "i",
       "source": "hname1",
       "target": "hname2"
     },
     .
     .
     .
    ]
}
```

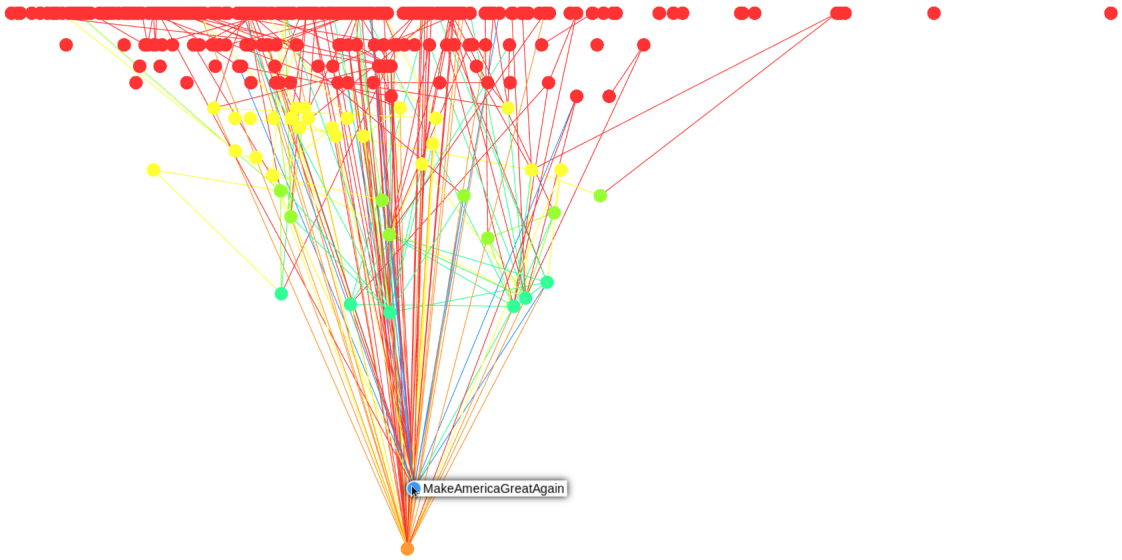
- *days.json* - Liste aller Tagen, mit der Anzahl der verschiedenen Hashtags, für die Zeitanalyse:

```
[
   {
     "x": x,
     "y": sum over "htags",
     "label": yy-mm-dd,
     "htags": [
       {
         "y": y1,
         "hname": "hname1"
       },
       {
         "y": y2,
         "hname": "hname2"
       },
       .
       .
       .
     ]
   },
   .
   .
   .
]
```

Visualisierung des Hashtagnetzwerkes

- haben die Javascript-Bibliothek sigmajs benutzt
- haben uns bewusst gegen eine kreisförmige Darstellung des Netzwerkes entschieden, weil diese die Lesbarkeit unseres Graphen und die Erkennung der verschiedenen Cluster erheblich erschwert hätte.
- Stattdessen haben wir uns für eine andere Art der Visualisierung entschieden, die die Visualisierung in die Breite ziehtünd das Problem der sehr ungleichmäßigen Verteilung löst
- Format unserer Input-Datei: .json

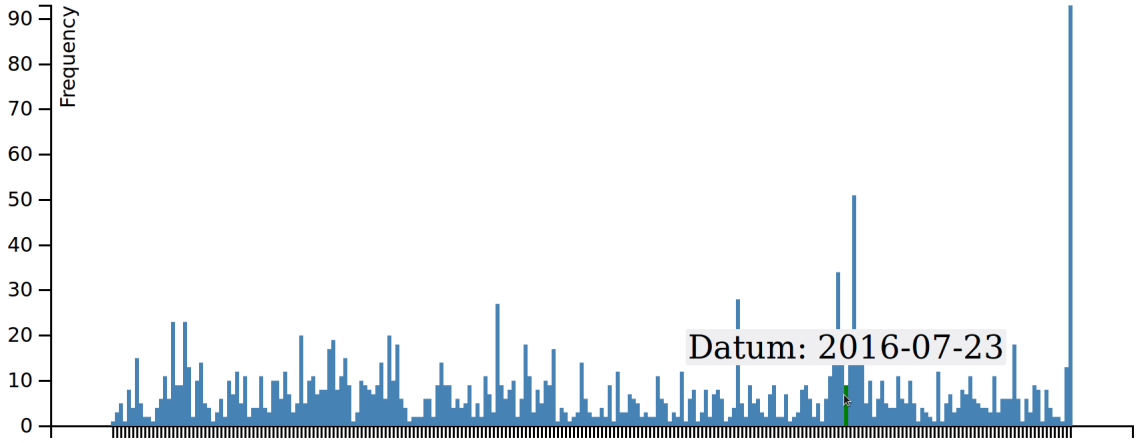
Abbildung 1: Hashtag-Netzwerk. x-Achse: Wichtigkeit; y-Achse: Häufigkeit



Visualisierung der Häufigkeit

- haben die Javascript-Bibliotheken d3js und Canvas benutzt
- auf der x-Achse tauchen alle im Datensatz aufgelisteten Tage auf vom 5.01.2016 bis zum 27.9.2016
- Die y-Achse geht von 0 bis 93, wobei 93 die höchste Anzahl an Hashtags darstellt, die an einem Tag getweetet wurden
- Da die Darstellung der Tage an der x-Achse aufgrund des großen Datensatzes nicht lesbar war, haben wir mit d3-tip Tooltips eingefügt, sodass wir beim Gleiten über jeden Balken das jeweilige Datum lesen koennen.

Abbildung 2: Gesamtanzahl aller Hashtags pro Tag



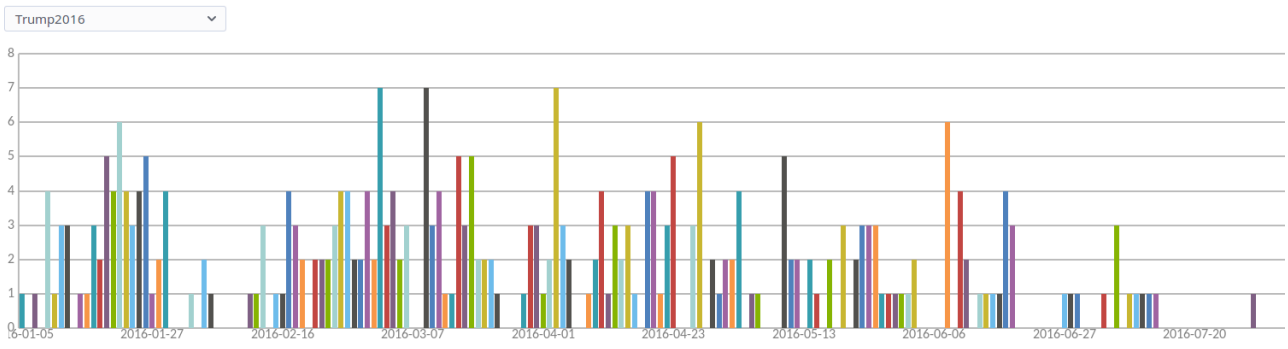
– fuer die Darstellung der Häufigkeit des Auftretens eines auswählbaren Hashtags haben wir aus den schon vorhandenen JSON-Dateien eine Liste mit Day-Objekten erstellt, die von der Canvas-Bibliothek für die Visualisierung als Input genutzt wird:

```
[
    {
     "x": 0,                    //X-Achse Index
     "y": 5,                    //Häufigkeit
     "label": "2016-05-09",     //Datum
    },
    .
    .
    .

]
```

was in diesem Fall bedeutet, dass der ausgewählte Hashtag an dem Tag 2016-05-09 fünf mal vorgekommen ist.

Abbildung 3: Häufigkeits des Hashtags *Trump2016*



Unser Projekt ist auf GitHub unter diesem Link zu finden: https://github.com/derMihai/DBS_Project