Twitter Welt-Topic nach Geokoordinaten geclustert

Viele von uns verwenden Twitter täglich, um sich über die neusten Ereignisse und das aktuelle Weltgeschehen zu informieren. Darüber hinaus ist ein grosser Teil der Twitter-Nutzer selbst aktiv und schreibt täglich Beiträge in Form von Tweets. Twitter Inc. verwendet, die so generierten Daten, um trendige Topics zu erstellen. Wie wichtig die trendigen Topics sind, ist daran zu erkennen, dass Twitter mit den «Trending Topic Ads» viel Geld verdient. Mittels der trendigen Topics ist es möglich die Aufmerksamkeit auf einen Punkt zu richten. In dieser Arbeit wird zu Beginn aufgezeigt wie weltweit trendige Topics entstehen und wie diese bei Twitter abgerufen werden können. Im zweiten Teil wird ermittelt, wo auf der Welt das Topic verfasst wurde. Dabei stellt sich die Frage, ob sich bei den trendigen Topics von Twitter Cluster erkennen lassen und sind diese aussagekräftig.

Ein trendiges Topic entsteht, wenn viele Tweets das gleiche Schlagwort aufweisen. Doch die Masse alleine reicht nicht aus. Damit ein Schlagwort als trendiges Topic zählt, muss die Verbreitung des Schlagwortes ein exponentielles Wachstum aufweisen. Trendige Topics entstehen nicht aus dem nichts. Sie werden von Mainstream-Medien und Nachrichtensendern ausgelöst. Oftmals entsteht ein trendiges Topic nach einem schockierenden Medienbericht, da dann die Twitter-Nutzer ihre Meinung und Gedanken dazu mittels Twitter verbreiten möchten. Über die Twitter-API[1] ist es möglich die Top-Topics einer Region abzufragen. Diese Topics sind anhand von Standorten über die WOEID (Where on Earth ID) gruppiert. Die WOEID «1» repräsentiert die ganze Erde. Dadurch ist es möglich, Welt-Topics über diesen API-Endpunkt zu beziehen. In Snippet 1 ist aufgezeigt, wie Welt-Topics von Twitter API mittels der Python Bibliothek Tweepy[2] bezogen werden. Um eine Verbindung zur Twitter API herzustellen, wird zuerst eine Authentifizierung eingerichtet. Die Authentifizierung erfolgt über das OAuth 2.0 Protokoll[3]. Dazu muss bei Twitter die Applikation registriert sein (Matthew A. Russel, 2019).

```
import tweepy
WOEID_GLOBAL = 1
auth = tweepy.OAuthHandler(CONSUMER_KEY, CONSUMER_SECRET)
auth.set_access_token(OAUTH_TOKEN, OAUTH_TOKEN_SECRET)
api = tweepy.API(auth)
trends = api.trends_place(WOEID_GLOBAL)
world_popular_trend = trends[0]['trends'][0]
print('{}: {}'.format(world_popular_trend['name'], world_popular_trend['tweet_volume']))
```

Snippet 1: Trendige Topics von Twitter abfragen

```
Vindman: 251935
```

Abbildung 1: Output Snippet 1

1 Twitter Inc. (2019). *Twitter API.* Retrieved 11 19, 2019, from https://developer.twitter.com/en/docs/tweets/filter-realtime/overview

2 Roesslein, J. (2019). *Tweepy.* Retrieved 11 18, 2019, from https://tweepy.readthedocs.io/en/latest/streaming_how_to.html

3 IEFT OAuth Working Group. (2019). *OAuth 2.0.* Retrieved 11 18, 2019, from https://oauth.net/2/

Am 18. November 2019 war «Vindman» mit einem Tweetvolumen von über 250'000 das Welt Topic. Dies kam nicht überraschend, da an diesem Tag eine Impeachment-Anhörung in den USA stadtfand und Alexander Vindman ein wichtiger Zeuge war. Den Vindman kritisierte die Aussagen von Präsident Donald Trump bei einem Telefonat mit dessen ukrainischen Amtskollegen. An diesem Tag war diese Nachricht in allen Medien zu lesen. Dies bestätigt die Aussage, dass ein Twitter Topic nicht aus einem Tweet entsteht, sondern von den Medien ausgelöst wird.
Das geladene Topic kann nun als Schlagwort bei der Suche nach Tweets verwendet werden. Snippet 2 zeigt auf wie Tweets, die das aktuelle Welt-Topic beinhalten geladen werden. Dies erfolgt wie in Snippet 1 über die Python Bibliothek Tweepy (Matthew A. Russel, 2019).

```
import tweepy
import json
tweets = []
try:
   for tweet in tweepy.Cursor(api.search, q=world_popular_trend['name'], count=1000).items():
      tweets.append(tweet._json)
except Exception as e:
   pass
f = open('tweets.json', 'w')
f.write(json.dumps(tweets, indent=1))
f.close()
```

Snippet 2: Tweets laden zu Topics

Twitter bietet für eine Applikation mehrere Möglichkeiten an, Daten zu laden. Je nach Umfang wird jedoch eine Gebühr für die Applikation verlangt. Durch das Snippet 2 konnten über 10'000 Tweets geladen werden, danach wurde der kostenlose Applikationsschlüssel für 24 Stunden gesperrt. Da sämtliche Tweets keine Geokoordinaten enthalten, wird der Standorts des Benutzers in Geokoordinaten umgewandelt und verwendet. Für die Umwandlung wurde LocationIQ[4] eingesetzt. LocationIQ bietet ein kostenlosen API-Endpunkt an, um ein Standort in Geokoordinaten umzuwandeln. Dabei gibt es jedoch die Restriktion, dass nur zwei Standorte in der Sekunde umgewandelt werden dürfen. Deshalb ist in Snippet 3 ein Timeout eingebaut. Die Anbindung an die API von LocationIQ wird mit dem Python Bibliothek Geocoder[5] realisiert. Geocoder bietet alle benötigten Hilfsmethoden, um eine Verbindung zum Endpunkt herzustellen.

4 Unwired Labs. (2019). *LocationIQ*. Retrieved 11 18, 2019, from https://locationiq.com/
5 Carriere, D. (2013). *Geocoder*. Retrieved 11 18, 2019, from https://geocoder.readthedocs.io/

```
import json
import geocoder
import time
locations = []
with open('tweets.json', 'r') as json_file:
    data = json.load(json_file)
    tweets_with_locations = [tweet for tweet in data if tweet['user']['location'] != '']

    for tweet in tweets_with_locations:
        try:
            geo = geocoder.locationiq(tweet['user']['location'], key=FREE_KEY)
            if geo.json != None:
                locations.append(geo.json)
                time.sleep(1) # API-Limite nicht überschreiten.
        except Exception as e:
            pass
f = open('location.json', 'w')
f.write(json.dumps(data, indent=1))
f.close()
```

Snippet 3: Standorte in Geokoordinaten umwandeln

Die in Snippet 3 gesammelten Geokoordinaten werden verwendet, um diese als Punkteauf einer Weltkarte zu platzieren. Der dazu verwendete Code befindet sich im Anhang im Snippet 4. Um Geokoordinaten auf einer Landkarte zu zeichnen wurde die Python Bibliothek Matplotlib[6] verwendet. Diese beinhaltet die Klasse Basemap, die genau für diese Aufgabe erstellt wurde. Aus den gesammelten Geokoordinaten aus Snippet 3 und Snippet 4 entstand die Abbildung 2. Auf den ersten Blick sticht ein Gebiet stark heraus: die USA. Bei einem trendigen Topic, das mit der politischen Situation des Landes zusammenhängt, war dies zu erwarten, dadie Bevölkerung des Landes seine Meinung über dieses Thema teilen möchte. Doch bei einer genaueren Betrachtung gibt es eine zweite starke Ansammlung in Europa. Diese Ansammlung ist nicht so ausgeprägt wie in USA, jedoch immer noch grösser als in anderen Kontinenten. Da es aufgrund API-Schnittstellen nicht möglich war, alle Tweets zu diesem Welt Topic zu erfassen, wäre es vage eine Aussage darüber zu machen, dass sich nur die USA und Europa für dieses Thema interessiert.

[6] John Hunter, D. D. E. F. M. D. (2019). *Matplotlib*. Retrieved 11 18, 2019, from https://matplotlib.org/

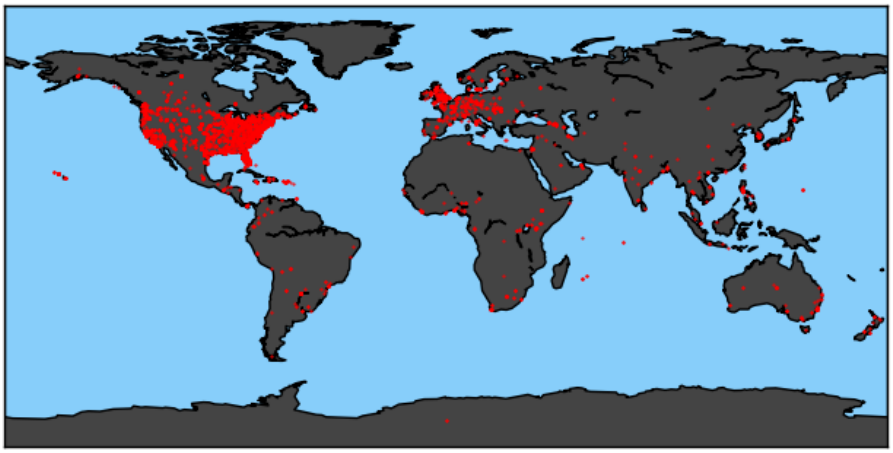

Abbildung 2: Tweets Geokoordinaten auf Weltkarte

Aufgrund der zwei starken Ansammlungen wurde mittels dem K-Means-Algorithmus zwei Cluster berechnet. Abbildung 3 zeigt das Resultat auf. Der Code befindet sich im Anhang im Snippet 5. Hierzu wurde die Python Bibliothek sklearn[7] verwendet, denn diese beinhaltet ein Modul «Cluster» für den K-Means-Algorithmus. Für die graphische Darstellung der Cluster wurde wiederum Matplotlib verwendet. Beide Zentern der Cluster werden mit einem roten Punkt dargestellt. USA und Europa sind die zwei Regionen, die am meisten zum Topic «Vindman» beitrugen. Da sich die roten Punkte in beiden Clustern nicht im Zentrum befinden, gibt es sicher noch weitere Gruppierungen, die auf der Landkarte in Abbildung 2 nicht herausstachen. Deshalb wurde Snippet 5 weitere Male durchgeführt, solange bis sich bei den gefundenen Clustern die roten Punkte im Zentrum befanden. Abbildung 4 ist das Resultat des K-Means-Algorithums von acht Cluster.
In Abbildung 4 stechen nun die Kontinente Südamerika, Europa, Afrika, Asien und Australien hervor. Nordamerika besteht aus drei Cluster: West-, Mittel und Ostamerika. Somit beteiligten sich bei diesem Topic Subjekte aus der ganzen Welt. Dies bestätigen wiederum Tech-Blogs und Fachzeitschriften. Für ein Welt-Topic braucht es eine Flächendeckende Beteiligung. Das Resultat würde mit sehr hoher Wahrscheinlichkeit bei einer noch grösseren Anzahl von Tweets und Geokoordinaten noch detaillierter ausfallen.

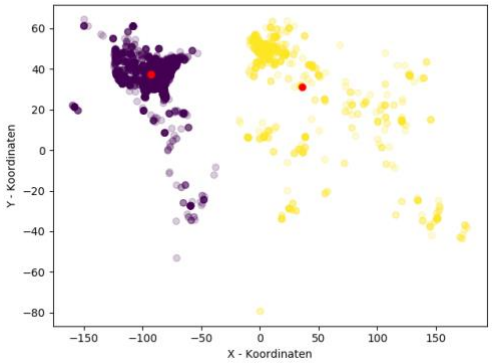

Abbildung 3: K-Means mit 2 Cluster

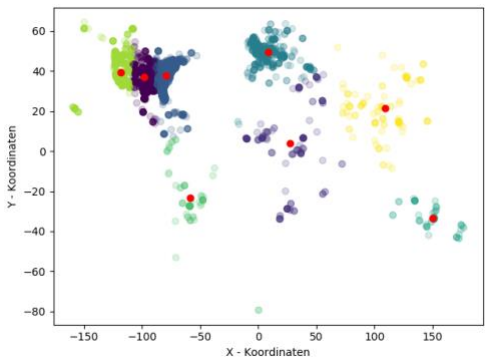

Abbildung 4: K-Means mit 8 Cluster

Nordamerika wurde durch den K-Means-Algorithmus in drei Gruppen aufgeteilt. Dies bedeutet, dass bei dem bearbeiteten Datensatz aus Snippet 2 der grösste Teil aus Nordamerika entstand,

[7] scikit-learn. (2019). *scikit-learn.* Retrieved 11 18, 2019 from https://scikit-learn.org/stable/

weil dort die Gruppendichte am grössten ist und somit der K-Means-Algorithmus mehrere Gruppen evaluieren konnte.
Etwas aus dem Topic «Vindman» und den Clustern zu interpretieren ist sehr schwierig und heikel, da die Datenbasis mit ca. 10'000 Tweets nur ein Bruchteil, der über 250'000 vorhanden Tweets ist. Jedoch zeigt die Methodik sehr gut auf was möglich wäre. Bei politischen Themen könnten nachdem die Tweets gefiltert wurden, politische Gruppierungen oder Regionen aufgedeckt werden, die zum Beispiel noch keine Meinung haben und für den Wahlkampf gewonnen werden könnten.

Twitter ist für viele Benutzer ein Tool um ihre Gedanken und Meinung mit der Öffentlichkeit zu teilen. Dazu gibt ein Benutzer sein Interesse an anderen Subjekten bekannt, indem er anderen Subjekten folgt. Auf den ersten Blick klingt dies harmlos. Doch bei genauerer Betrachtung und Untersuchung von Handlungen in Form von Tweets entpuppt sich Twitter als Schatzkammer von Informationen. Es handelt sich hier um eine Schatzkammer, die sich täglich weiter füllt.
Die Datenanalyse an sich ist sehr interessant und mächtig. Mit einem relativ kleinen Datensatz konnten Rückschlüsse auf politische Themen gemacht werden. Es wurde aufgezeigt, wie viel mehr Informationen ein einzelner Tweet beinhaltet als auf den ersten Blick ersichtlich ist. Die aufgezeigte Analyse kann beliebig auf mehrere Topics oder Schlagwörter angewendet werden. Wird die Analyse über eine längere Zeit mit grösseren Datensätzen durchgeführt, kann dies zu einem sehr mächtigen Tool werden. Dadurch kann die Frage zu Beginn mit gutem Gewissen bestätigt werden, dass aus den Clustern von Twitter-Daten sehr viel Hervorgesagt werden kann.

Weitere Snippets

```
import json
import matplotlib.pyplot as plt
from mpl_toolkits.basemap import Basemap
def getCoordinates(file):
    lat = []
    lon = []
    with open(file, 'r') as json_file:
        data = json.load(json_file)
        tweets_with_locations = [tweet for tweet in data if tweet['user']['location'] != '']
        for key, tweet in enumerate(tweets_with_locations):
            try:
                geo = tweet['user']['location']
                lat.append(float(geo['raw']['lat']))
                lon.append(float(geo['raw']['lon']))
            except Exception:
                pass
    x, y = map(lon, lat)
    return x, y
map = Basemap()
map.drawcoastlines()
map.fillcontinents(color='#444444', lake_color='#87CEFA')
map.drawmapboundary(fill_color='#87CEFA')
x, y = getCoordinates('location.json')
map.plot(x, y,'r.', markersize=1)
plt.show()
```

Snippet 4: Geokoordinaten auf Weltkarte zeichnen

```
from sklearn.cluster import KMeans
import numpy as np
import matplotlib.pyplot as plt
x, y = getCoordinates('location.json')
X = np.array(list(zip(x, y))).reshape(len(x), 2)
# clusters=2
clusters=8
kmeans = KMeans(n_clusters=clusters)
kmeans.fit(X)
y_kmeans = kmeans.predict(X)
centers = kmeans.cluster_centers_
plt.xlabel('X - Koordinaten')
plt.ylabel('Y - Koordinaten')
plt.scatter(X[:, 0], X[:, 1], c=y_kmeans, alpha=0.2)
plt.scatter(centers[:, 0], centers[:, 1], c='red')
plt.show()
```

Snippet 5: Geokoordinaten clustern mit K-Means

Literaturverzeichnis

Matthew A. Russel, M. K. (2019). *Mining the Social Web.* O'RELLY.

Abbildungsverzeichnis

Snippetverzeichnis