Freie Universität Berlin
Institut für Informatik
Datenbanksysteme
Sommersemester 2017
Bearbeiter: Felix Binder, Nicolas Höcker, Armin Weber

Projektdokumentation

1. Projektziel

Ziel des Projekts ist es, eine „Projekt-Web-Anwendung“ zu erstellen, mit deren Hilfe der Datensatz „american-election-tweets“ analysiert werden kann. Analysieren heißt hier, dass die Anwendung in der Lage sein soll, das Beziehungsgeflecht zwischen den in den Tweets verwendeten Hashtags grafisch darzustellen und mittels benutzerdefinierter Anfragen weitere Informationen aus den Daten zu gewinnen. So soll es unter anderem möglich sein, die am meisten verwendeten Hashtags zu identifizieren, Aussagen über deren paarweise gemeinsames Auftreten zu treffen und die Wichtigkeit einzelner Tweets zu bewerten. Auch Auswertungen dessen, wie sich die einzelnen Punkte über die Zeit hinweg verändert haben, sollen möglich sein.

2. Team

Felix Binder: Studiert Philosophie und Informatik (60 LP) gegen Ende seines Bachelorstudiums.

Nicolas Höcker: Studiert Informatik im 4. Bachelorsemester.

Armin Weber: Studiert Informatik im vierten Semester.

3. Explorative Datenanalyse

Der Datensatz besteht aus 6126 Tweets, die zwischen dem 5. Januar 2016 und dem 28. September 2016 über die Twitter-Konten realDonaldTrump und HillaryClinton abgesetzt, d. h. von diesen selbst verfasst oder retweetet wurden. Das jeweilige Twitter-Konto findet sich dabei in der ersten Spalte des Datensatzes („handle“); die Information, ob es sich um einen Retweet handelt, in der dritten Spalte („is_retweet“), in der vierten bei Retweets der ursprüngliche Autor („original_author“). Der Text selbst – und mit ihm alle Hashtags – befindet sich in der zweiten Spalte.

Darüber hinaus enthält der Datensatz Informationen darüber, ob der Text ein Zitat ist und also auf eine andere Quelle verweist (siebte Spalte, „is_quote_status“, mit der Quelle in der neunten Spalte, „source_url“) sowie dabei evtl. abgekürzt wurde (zehnte Spalte, „truncated“), ob der jeweilige Tweet eine Antwort auf einen anderen Tweet darstellt (sechste Spalte, „in_reply_to_screen_name“), wie oft er retweetet wurde und wie oft favorisiert (Spalten 7 und 8).

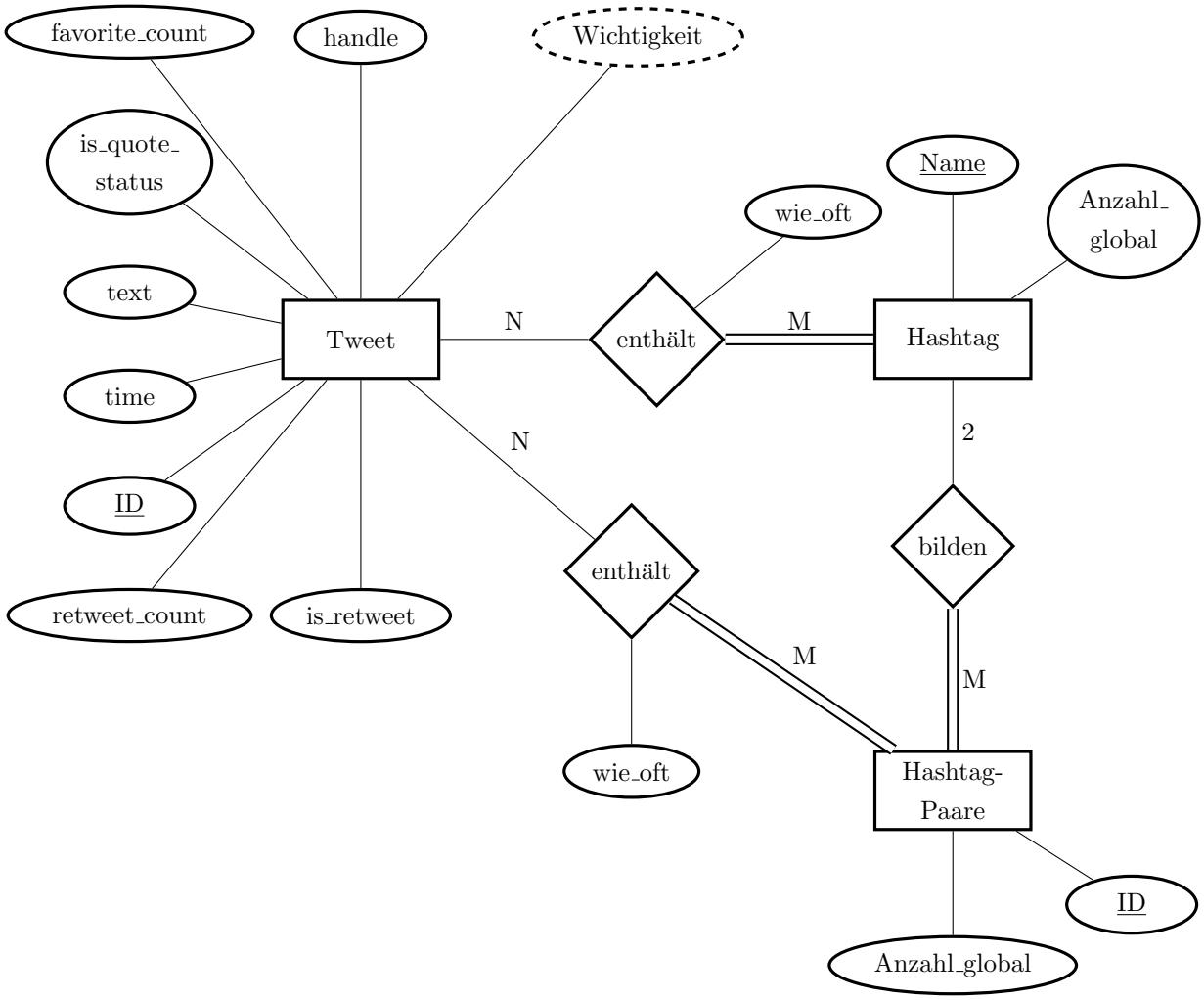
Dem ersten Augenschein nach sind für uns von Bedeutung:

a) die Spalte Text, um daraus die Hashtags zu extrahieren;
b) die Spalte mit dem Datum und der Uhrzeit, um das unterschiedlich starke Vorkommen von Hashtags zu verschiedenen Zeitpunkten nachvollziehen zu können;
c) die Spalten zu Retweets und Favorisierungen, weil sie Indikatoren für die Wichtigkeit von Tweets sind.

Die weiteren Spalten scheinen auf den ersten Blick für unsere Zwecke vernachlässigbar zu sein.

4. ER-Modellierung

Das ER-Modell für das Projekt sieht wie folgt aus:



Dieses Modell enthält augenblicklich noch mehr Informationen, als für die Beantwortung der in Abschnitt 1 genannten Fragen nötig wäre. So wäre es eigentlich unnötig abzuspeichern, ob ein Tweet ein Retweet oder ein Zitat ist. Allerdings wollen wir die Option offenhalten, in der fertigen Anwendung zu untersuchen, ob es signifikante Unterschiede hinsichtlich der Wichtigkeit der Tweets der verschiedenen Arten gibt. Da wir solche Unterschiede nicht ausschließen können, scheint uns das Weglassen solcher Informationen zu diesem frühen Zeitpunkt nicht angeraten.

Davon abgesehen war unser Anliegen, immer die naheliegenden Entscheidungen zu treffen: etwa Hashtag-Paare aus Hashtags zusammenzusetzen und in den Relationen alle Informationen zu speichern, die später abrufbar sein sollen, etwa wie oft Hashtags in einzelnen Tweets vorkommen. Dort, wo es sinnvoll erschien, haben wir als Schlüssel außerdem eigenständige IDs ergänzt, um keine zu komplexen Schlüssel verwenden zu müssen (wie es bei den Tweets sonst etwa der Fall wäre).

Außerdem haben wir uns nach einigem Hin und Her dazu entschlossen, den Autor eines

Tweets als Attribut handle in das Modell zu integrieren statt als eigene Entität. Hätten wir ihn als eigene Entität modelliert, die zu Tweets in den Relationen „schreibt“ und „retweetet“ steht, so hätten wir eine Tabelle mit nur einem einzigen Attribut (eben „handle“), wohingegen alle Attribute der Relationen zu „Tweet“ gezogen hätten werden können, da es sich um 1:N-Relationen gehandelt hätte. Damit hätte man für Abfragen nach dem Autor eines Tweets eigens mehrere Tabellen mittels JOIN verbinden müssen, was gerade bei Abfragen von den Hashtags her reichlich komplex geworden wäre, ohne dass das Ausgliedern des Autors als Entität irgendeinen Vorteil zu versprechen schien.

5. Relationales Modell

TWEET(ID, handle, text, time, is_retweet, is_quote_status, retweet_count, favorite_count)
HASHTAG(Name, Anzahl_global)
T_ENTH_H(T_ID, H_Name, wie_oft)
HASHTAG_PAARE(ID, Anzahl_global)
H_BILDEN_HP(H_Name, HP_ID)
T_ENTH_HP(T_ID, HP_ID, wie_oft)