Twitterdaten analysieren

mithilfe der blockbasierten Programmiersprache SNAP*!*

von Andreas Grillenberger

Die Förderung von Datenkompetenzen ist heute ein immer wichtigeres Ziel des Informatikunterrichts, das auch in immer mehr Lehrpläne Einzug hält. Beispielsweise sieht der aktuelle Lehrplan des Bayerischen Gymnasiums vor, dass Schülerinnen und Schüler „die Chancen und Risiken der automatischen Auswertung großer Datenmengen auch im Hinblick auf gesellschaftliche Auswirkungen [bewerten]“ (ISB, 2020); auch in anderen Lehrplänen sind oft ähnliche Formulierungen zu finden. Um solche Kompetenzen im Schulunterricht zu fördern, werden einerseits geeignete Werkzeuge, andererseits aber auch entsprechende Datenquellen benötigt, die solche Analysen sinnvoll möglich machen. Da, je nach Schulart und vorherigem Unterricht, nur sehr unterschiedliche Vorkenntnisse sowohl in der Arbeit mit Daten als auch bei der Programmierung vorausgesetzt werden können, wird in diesem Beitrag ein Ansatz präsentiert, der einen einfachen Einstieg in das Thema der Datenanalysen ermöglicht, keine spezifischen Vorkenntnisse erfordert, gleichzeitig aber einen reichen Datenschatz zugänglich macht. Entsprechend können auf dessen Basis sowohl im Unterricht als auch darüber hinaus spannende Erkenntnisse gewonnen werden und möglicherweise auch Fragestellungen anderer Fächer in interdisziplinären Projekten angegangen werden.

Twitter als Datenquelle

Eine wichtige Datenquelle – sowohl für Unternehmen als auch für wissenschaftliche Fragestellungen – stellen heute soziale Medien dar. Während viele dieser Portale eher auf spezifische Zielgruppen (oft alters- oder themenspezifisch) ausgerichtet sind, scheinen einige wenige wie Twitter heute einen guten Querschnitt der Bevölkerung zu erreichen, sodass anhand der im Rahmen der Nutzung dieser Portale anfallenden Daten interessante Fragestellungen mit relativ hoher Validität bearbeitet und somit spannende Aussagen gewonnen werden können. Gleichzeitig nehmen soziale Medien eine wichtige Rolle im Alltag der meisten Schülerinnen und Schüler ein, sodass sie auch hinsichtlich der Sensibilisierung für die möglichen Schlussfolgerungen, die aus der Nutzung dieser Systeme gezogen werden können, eine spannende Grundlage darstellen: Basierend auf diesen Daten werden heute Trends, Verkaufszahlen von Produkten oder sogar Wahlergebnisse vorhergesagt – teils mit höherer Genauigkeit als mit traditionellen und oft wesentlich aufwendigeren Methoden. Insbesondere Twitter kommt dabei eine zentrale Rolle zu: Aufgrund der eingeschränkten Länge der Tweets und der gleichzeitig großen Fülle an Metadaten sind diese meist relativ einfach analysierbar. Außerdem ist Twitter auch in völlig verschiedenen Bevölkerungsschichten und Altersgruppen verbreitet, sodass, im Vergleich mit anderen sozialen Medien, für viele Analysen eine relativ repräsentative Stichprobe vorliegt.

Auf Twitter wurden 2013 pro Sekunde im Durchschnitt 5700 Tweets veröffentlicht (vgl. Krikorian, 2013). Obwohl seitdem keine verlässlichen Zahlen veröffentlicht wurden, kann vermutet werden, dass diese Anzahl seitdem nicht abgenommen, sondern eher zugenommen hat. Auf die gesamte Datenmenge, die dabei erzeugt wird, kann über die entsprechende Programmierschnittstelle (API) in nahezu Echtzeit zugegriffen werden; es ist jedoch nur ein Teil dieses Datenstroms kostenfrei verfügbar, der ca. 30 bis 50 Tweets zufällig aus dem Datenstrom ausgewählten Tweets pro Sekunde entspricht. Neben dem eigentlichen Inhalt enthält jeder Tweet ca. 150 weitere Attribute (vgl. Dwoskin, 2014), die diesem als Metadaten mitgeliefert werden und auch in Datenanalysen einbezogen werden können: Neben einer eindeutigen ID liefert jeder Tweet beispielsweise die Sprache des Tweets mit, aber auch das Land aus dem dieser abgesetzt wurde und – wenn möglich – genauere Koordinaten des Orts, aber auch Informationen über den Autor (wie Benutzername, Follower, ID) und sein Profil auf Twitter (wie beispielsweise die gewählte Hintergrundfarbe).

Trotz der kleinen Größe eines einzelnen Tweets ist eine klassische Analyse der Daten aus mehreren Gründen nicht möglich: Einerseits können bei einer datenbankbasierten Analyse kaum Ergebnisse in Echtzeit produziert werden, da durch die gleichzeitigen Schreib- und Lesezugriffe auf die Datenbank durch die Erzeugung neuer Tweets und die Analyse Konkurrenzsituationen entstehen würden. Andererseits ist die Datenmenge höher als vielleicht erwartet, sodass auch Speicherplatzprobleme bei der Nutzung einer Datenbank nicht vernachlässigt werden können. Um die Datenmenge, die auf Twitter zur Verfügung steht bzw. kontinuierlich generiert wird, nach unten abzuschätzen, kann angenommen werden, dass der reine Tweettext (unter der Annahme, dass dieser im Durchschnitt nur ca. 70 Zeichen lang ist und UTF-8-kodiert gespeichert

bzw. übertragen wird) selbst ohne Metadaten mindestens 200 Bytes groß ist. Eine konservative Schätzung für den gesamten Tweet inklusive aller Metadaten und strukturierender Informationen des genutzten Datenformats beträgt daher sicherlich über 500 Bytes pro Tweet, sodass pro Stunde über 10 GB, pro Tag sogar über 250 GB an Daten entstehen – real wahrscheinlich deutlich mehr, wie Experimente des Autors zeigten. Selbst bei der für die Schule zugreifbaren Datenmenge von ca. 40 Tweets pro Sekunde kommen pro Stunde noch über 70 MB an Daten zusammen. Durch diese große Menge an zur Verfügung stehenden Daten mit umfangreichen Metadaten werden entsprechend interessante Datenanalysen möglich. Damit kann Twitter als spannende und für die Schule relativ einfach zugreifbare Datenquelle genutzt werden.

Twitter trifft SNAP*!*

Um einfache Analysen des Twitter-Datenstroms im Informatikunterricht thematisieren zu können, ist – wie zuvor erwähnt – zwingend ein geeignetes Werkzeug notwendig, das im Gegensatz zu professionellen Werkzeugen eine für den Schulunterricht angemessene Komplexität besitzt, wobei insbesondere eine geringe Einstiegshürde angestrebt wurde, aber auch ausreichende Möglichkeiten zur Verfügung stehen sollten, um mehr als nur primitive Analysen zu ermöglichen. Um diese Ziele zu erreichen, wurde entschieden, kein komplett neues Werkzeug zu entwickeln. Stattdessen wird auf die blockbasierte Programmierumgebung SNAP*!* aufgesetzt und diese so erweitert, dass einfache Datenanalysen am Beispiel des Twitter-Datenstroms ermöglicht werden. Die Wahl fiel auf dieses konkrete Werkzeug, da einerseits blockbasierte Programmierung heute vielen Schülerinnen und Schülern bereits bekannt und auch ohne Vorkenntnisse relativ einfach beherrschbar ist, sodass die Einstiegshürde sinkt. Andererseits zeichnet sich SNAP*!* auch durch eine hohe Flexibilität und einfache Erweiterbarkeit aus, was zugleich die Werkzeugentwicklung unterstützt und vereinfacht, aber auch dafür sorgt, dass den späteren Nutzern vielfältige Möglichkeiten offenstehen. Prinzipiell kann das im Folgenden vorgestellte Konzept jedoch auch auf viele weitere blockbasierte Programmierumgebungen und auch auf die textuelle Programmierung übertragen werden.

Ein technisches Wissen, beispielsweise hinsichtlich des Zugriffs auf die Twitter-API, ist auf Seiten des Nutzers nicht nötig, da durch Implementierung eigener Blöcke für alle mit Twitter verwandten Aufgaben eine entsprechende Abstraktionsebene geschaffen wurde, die aber prinzipiell durchschaut werden kann. Hinsichtlich der genauen technischen Funktionsweise werden interessierte Leserinnen und Leser auf den Abschnitt „Technische Hintergründe“ verwiesen. Für die Schülerinnen und Schüler wird das Werkzeug komplett im Internet und nutzbar im Browser zur Verfügung gestellt, um eine möglichst einfache Nutzung zu ermöglichen. Um jedoch keinen völlig freien Zugriffspunkt auf die Twitter-API zu schaffen, findet bei der ersten Abfrage von Daten von Twitter eine Abfrage nach Benutzername und Passwort statt – diese können derzeit nur durch den Serverbetreiber festgelegt werden, aber gleichzeitig durch mehrere Nutzer verwendet werden. Alle Zugriffe auf Twitter-Daten erfolgen über Blöcke, die bei Aufruf von SNAP*!*Twitter automatisch in die SNAP*!*-Oberfläche integriert werden. Dabei stehen unter anderem die folgenden zur Verfügung:

- Empfang eines einzelnen Tweets (JSON-Format – ***J****AVA****S****CRIPT* ***O****bject* ***N****otation*),
- Interpretation von Tweets als Tabelle von Attributen und Attributwerten,
- Extraktion von Attributen des Tweets, wie beispielsweise Geodaten,
- Ausführen von Skripten für alle empfangenen Tweets.

Weiterhin stellt SNAP*!*Twitter auch Funktionen zur Verfügung, die die Visualisierung von Analyseergebnissen unterstützen:

- Möglichkeit zur Nutzung von Kartendarstellungen, z.B. um Tweets dort anzuzeigen (scroll- und zoombar, Umschaltbar auf Satellitendarstellung, zuschaltbares Clustering),
- eine flexibel einsetzbare Diagramm-Bibliothek (u.a. mit Unterstützung von Balken, Linien- und Tortendiagrammen),
- Im- und Exportieren von Daten als CSV-Dateien, um diese ggf. mit anderen Programmen weiterzuverarbeiten.

Zugriff auf SNAP*!*Twitter

Eine Installation von SNAP*!*Twitter steht Ihnen mit den folgenden Daten kostenfrei zur Verfügung:

URL: https://snaptwitter.dataliteracy.education
Benutzername: login2020
Passwort: snaptwitter

Benutzername und Passwort werden bei der ersten Datenabfrage von Twitter abgefragt.
Das Programmierwerkzeug funktioniert derzeit am besten mit *Chrome* und *Firefox*.
Weiterhin kann ein eigener Server betrieben oder Einsicht in den Quelltext genommen werden.
Informationen dazu finden sich auf GitHub unter
https://github.com/AGrillenberger/SnapTwitter2

Fachliche Grundlage: Datenstromanalysen

Der Analyse der Twitterdaten mittels SNAP*!*Twitter liegt ein anderes Prinzip zugrunde als bei der klassischen Datenanalyse: Während klassisch Daten in einer

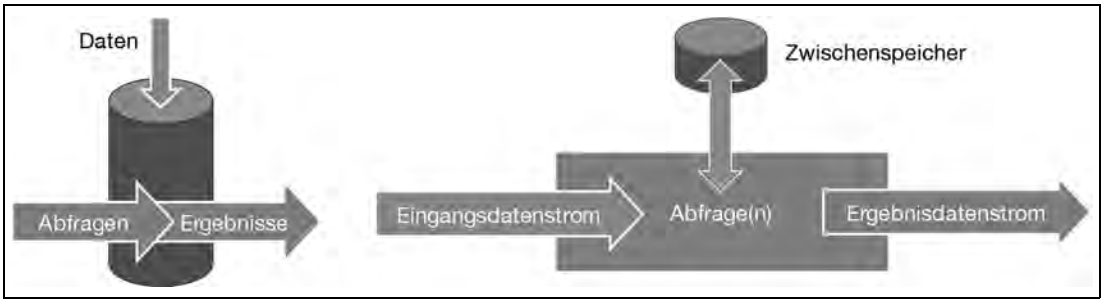


Bild 1: Vergleich der Funktionsweise von Datenbanken (links) und Datenstromsystemen (rechts).

Datenbank oder einem anderen Datenspeicher wie in strukturierten Textdateien gesammelt und im Nachgang ausgewertet werden, müssen Daten von Datenquellen wie Twitter oder auch Sensoren oft mehr oder weniger direkt ausgewertet werden, um valide Analyseergebnisse zu generieren. Es handelt sich dabei um eine *Datenstromanalyse*, bei der keine vorherige (und für die Analyse auch überhaupt keine dauerhafte) Speicherung der Daten nötig ist, sondern Daten direkt, nachdem sie empfangen wurden, verarbeitet werden (siehe Bild 1). Datenstromsysteme sind im Gegensatz zu Datenbanken nicht auf eine dauerhafte Speicherung, sondern auf eine schnelle Analyse von Daten, idealerweise in Echtzeit, optimiert. Sie agieren als Filter für Datenströme und generieren so kontinuierlich Ergebnisse auf anhand bereits vorher definierter Abfragen. Weitere Informationen sind in Grillenberger/Romeike (2019) zu finden.

Da SNAP/Twitter im Hintergrund immer einige Tweets in einem Puffer zwischenspeichert, wird das Datenstromprinzip hier geringfügig verletzt, jedoch eine deutlich geringere Latenz erreicht, sodass dies für Bildungszwecke als adäquate Anpassung erscheint. An allen weiteren Stellen wird das Datenstromprinzip jedoch bestmöglich umgesetzt, sodass diese Anpassung kaum für den Nutzer erkennbar ist.

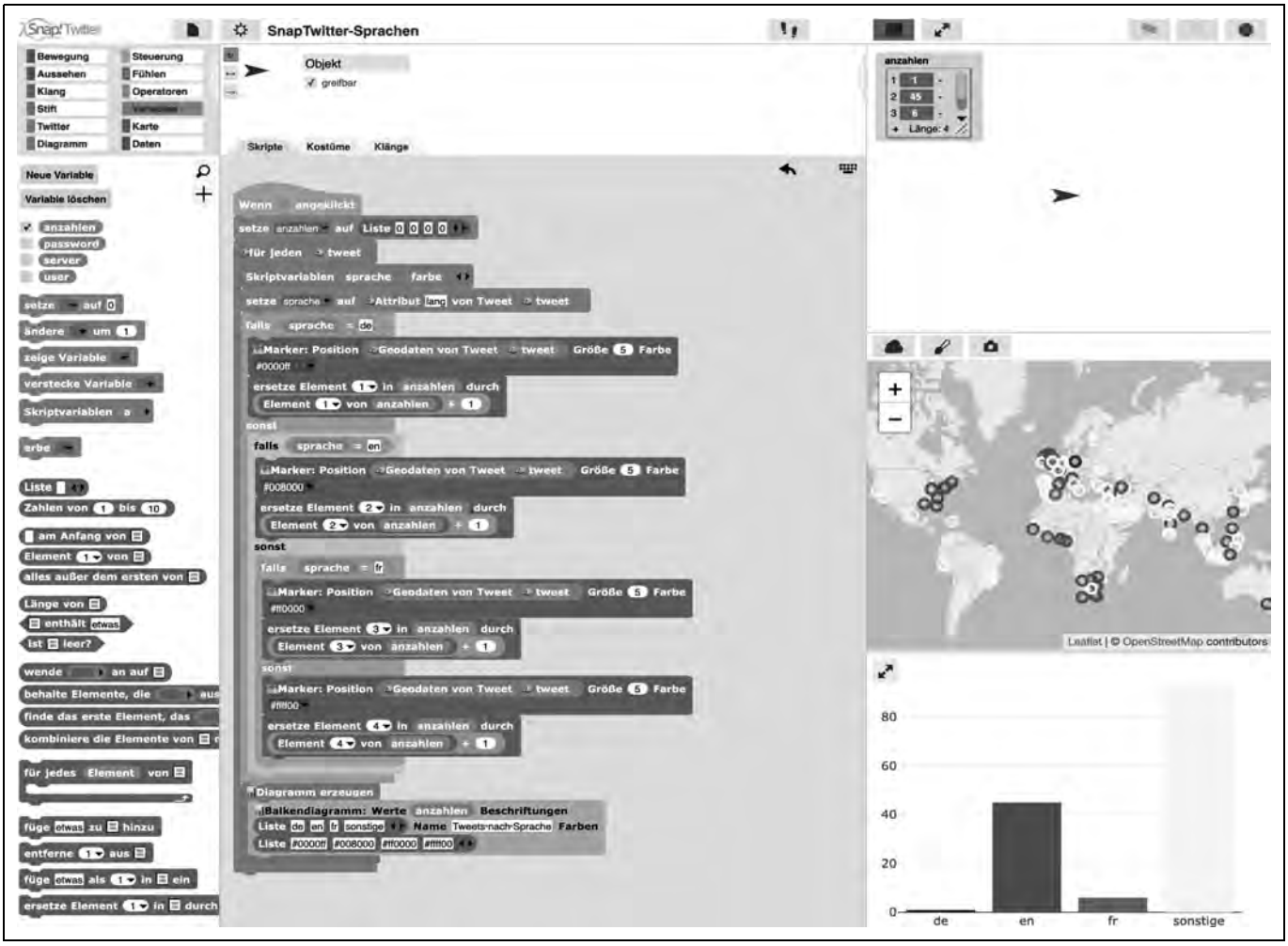
Einsatzmöglichkeiten im Informatikunterricht

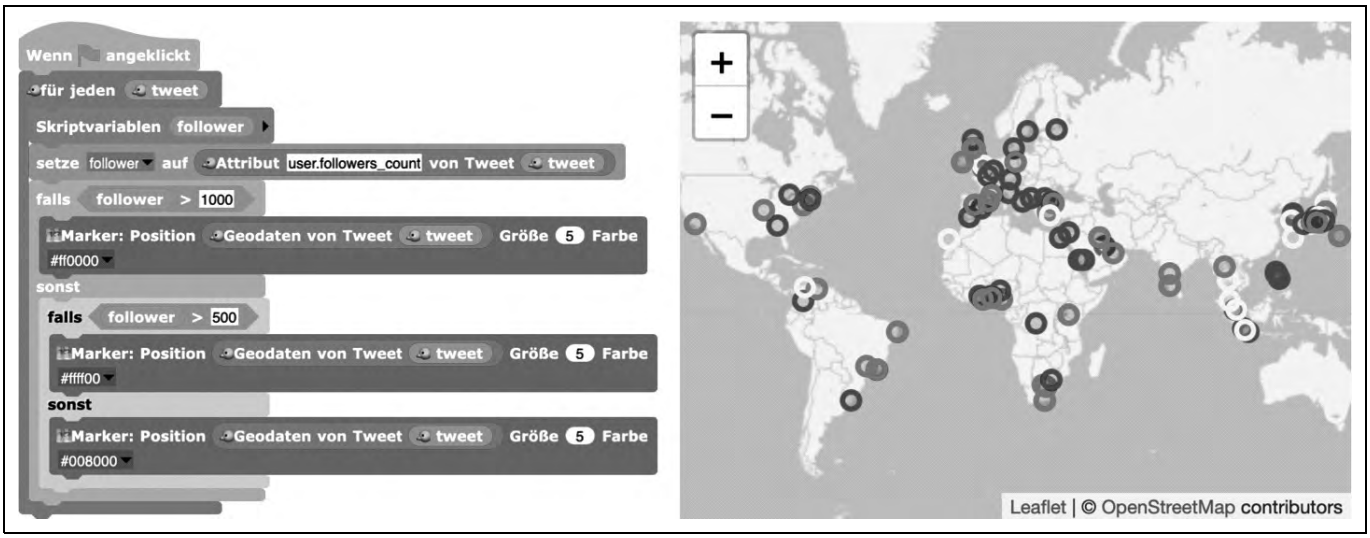
Das entwickelte Werkzeug kann auf unterschiedliche Weise und mit völlig verschiedenen Zielen im Informatikunterricht eingesetzt werden. An dieser Stelle wird exemplarisch eine Möglichkeit vorgestellt, bei der es darum gehen soll, dass Schülerinnen und Schüler Grundzüge von Datenstromanalysen kennenlernen und verstehen. Das Ziel im Unterricht ist insbesondere, eigene einfache Datenanalysen durchführen und Schlüsse aus den gewonnenen Ergebnissen ziehen zu können. Dies kann im Unterricht vielfältig kontextualisiert werden: Insbesondere sind als Kontext Datenanalysen gut geeignet, denen Schülerinnen und Schüler heute bereits im Alltag begegnen, beispielsweise bei Verwendung sozialer Medien oder beim Einkauf im Online-Versandhandel. Andauernd werden unter anderem Personen hinsichtlich verschiedener Merkmale untersucht, anhand dieser in Klassen eingeordnet bzw. zu solchen zusammengefasst und basierend auf dieser Einordnung Schlüsse gezogen. Beispielsweise werden in Onlineshops Personen anhand ihrer zuletzt gekauften Produkte klassifiziert und dadurch wird versucht vorherzusagen, welche Produkte sie als Nächstes kaufen, sodass gezielt Werbung platziert werden kann. Andererseits werden aber auch Standortdaten verwendet, um z.B. Preisanpassungen für bestimmte Kundengruppen durchzuführen. Um zu verstehen, wie solche Analysen und Vorhersagen überhaupt funktionieren, welches Potenzial und welche Risiken diese bergen und um fundiert zu entscheiden, ob und in welchem Umfang man Anbietern, die solche Analysen nutzen, vertraut, sind grundlegende Kenntnisse und Erfahrungen in der Datenanalyse nötig. Dabei sind insbesondere die drei grundlegenden Datenanalysemethoden *Klassifikation*, *Clusterbildung* und *Assoziation* zentral, die bereits mit dem vorgestellten Werkzeug einfach und zielführend thematisiert werden können:

Anhand einfacher *Klassifikationsaufgaben* können Lernende das der Klassifikation zugrunde liegende Prinzip erkennen: die Einteilung von vorliegenden Daten in zuvor definierte Kategorien. Beispielsweise kann dafür die Aufgabe gestellt werden, mit dem zur Verfügung gestellten Programmierwerkzeug alle eingehenden Tweets anhand bestimmter Stichworte oder anhand der Sprache, in der sie verfasst wurden, zu klassifizieren. Als Ergebnis kann beispielsweise ein Balkendiagramm (z.B. wie in Bild 2, nächste Seite) erstellt werden.

Die Diskussion möglicher aus der Klassifikation gewinnbarer Aussagen offenbart den Schülerinnen und Schülern die Grenzen dieser Methode: Beispielsweise kann durch Klassifikation problemlos analysiert werden, welche Produktgruppe beliebter ist als eine andere, woher die meisten Käufer kommen und Ähnliches. Die zuvor beschriebenen Schlussfolgerungen, die heute aus Daten gewonnen werden können, gehen jedoch weit über dieses Beispiel hinaus: Es geht nicht nur darum zu entdecken, welches Produkt am beliebtesten ist, sondern darum, welches Produkt in welcher Region bevorzugt wird – es wird eine weitere Dimension eingeführt. Eine Kategorisierung nach mehreren Dimensionen wäre zwar möglich, die Anzahl der Kategorien explodiert dabei jedoch, da jede Kategorie mit jeder anderen kombiniert werden kann. Zusätzlich können in solchen Fällen nicht immer schon vor der Analyse die Kategorien festgelegt werden; beispielsweise muss die Region hier nicht zwingend administrativen Regionen entsprechen. Somit wird in solchen Fällen eine Klassifikation beliebig komplex beziehungsweise sogar unmöglich.

An dieser Stelle setzt die *Clusterbildung* an: Daten werden dort anhand ähnlicher Merkmalsausprägungen zu nicht vorher definierten Gruppen zusammengefasst.





Trotz der komplexen mathematischen Grundlagen vieler Clusterverfahren kann eine einfache Clusteranalyse bereits mit dem hier entwickelten Werkzeug bewältigt werden: Nach Visualisierung einer bestimmten Eigenschaft auf der Karte können die Cluster intuitiv optisch bestimmt und so erkannt werden, welche der Ausprägungen dieser Eigenschaft in verschiedenen Regionen vorherrscht (siehe Bild 3). Obwohl insbesondere die Mächtigkeit dieses Vorgehens nicht der automatischen

Bild 2 (oben): Beispiel einer Datenanalyse mit Ergebnisdarstellung als Karte und Diagramm.

Bild 3 (unten): Code und Kartendarstellung einer Analyse mit SNAP!/Twitter. Die Farben visualisieren die Followerzahl des Autors (rot/mittleres grau: über 1000, gelb/hellgrau: über 500, grün/dunkelgrau: maximal 500).

Clusterbildung entspricht, hilft diese intuitive Herangehensweise, das Prinzip der Clusterbildung nachzuvollziehen, ohne die mathematischen Grundlagen verstehen zu müssen. Bereits dieses einfache Beispiel zeigt dabei einige grundlegende Fragestellungen der Clusterbildung:

- ▷ Sollen eher große und damit ungenauere Cluster erzeugt oder kleinere und genauere?
- ▷ Ab wann wird eine Ausprägung einer Eigenschaft als in einem Cluster vorherrschend charakterisiert?
- ▷ Welchen Fehlergrad bin ich bereit einzugehen?

Wenn anhand von Daten Entscheidungen oder Vorhersagen getroffen werden sollen, müssen entsprechende Regeln definiert bzw. gefunden werden. Diese werden in Form von Assoziationen aufgestellt, die Zusammenhänge zwischen Attributen beschreiben. Eine Assoziationsanalyse kann sowohl manuell als auch automatisch stattfinden, wobei im letztgenannten Fall verschiedene Algorithmen eingesetzt werden können. Diese sind jedoch stark mathematisch geprägt und aufgrund ihrer Komplexität für Schülerinnen und Schüler nur eingeschränkt nachvollziehbar. Trotzdem kann auch die Assoziation mit den Lernenden thematisiert werden, indem potenzielle Zusammenhänge bzw. Regeln in Form von Assoziationen, die sich aus den obigen Betrachtungen ergeben, mit den Schülerinnen und Schülern auf ihre Aussagekraft hin untersucht werden: Mit Blick auf eine entsprechende Kartendarstellung kann beispielsweise die Assoziation „Wer Twitter in Westeuropa, Südostasien oder den USA nutzt, hat mindestens 1000 Follower“ mit relativ geringem Fehler zutreffend zu sein. Diese Aussage vernachlässigt jegliche Kausalität (was im Sinne von korrelationsbasierten Analysen zulässig und üblich ist): Es wird nicht weiter überlegt, warum beispielsweise in Afrika, Australien oder Russland Twitter anscheinend kaum genutzt wird. Da es sich bei der Auswertung jedoch um eine nicht-repräsentative Stichprobe handelt (es wurden nur solche Tweets betrachtet, die Geodaten offenbaren) und diese auch sehr klein ist und einen Ausschnitt zu einer bestimmten Tageszeit zeigt, kann die Gültigkeit dieser Regel stark angezweifelt werden. Andere Assoziationen – beispielsweise ein Schluss von der Sprache des Tweets auf dessen Herkunftsland – scheint jedoch in vielen Fällen auch bei längerer Überprüfung mit wenigen Ausnahmen (z.B. Weltsprachen wie Englisch) zutreffend zu sein und kann damit eine Analyse bereichern, indem diese Assoziation dabei hilft, das Herkunftsland auch in solchen Fällen berücksichtigen zu können, in denen diese Information ansonsten fehlen würde. Diese Grenzen der Aussagekraft von Datenanalysen müssen in diesem Zusammenhang den Schülerinnen und Schülern unbedingt bewusst werden, um falsche Schlüsse und somit auch das Vorurteil der Allwissenheit und Unfehlbarkeit von Big Data zu vermeiden, aber auch um die Bedeutung eines möglichst vollständigen Datensatzes bei Big-Data-Analysen nachvollziehen zu können.

Ein erster Einstieg in die Datenanalyse kann beispielsweise durch das automatisch in SNAP*!*/Twitter angezeigte Tutorial oder einfache Fragestellungen erfolgen, wie etwa:

- ▷ In welchem Land / welcher Region sind die Twitternutzer am aktivsten?
- ▷ Über welchen Präsidentschaftskandidat in den USA wird am meisten getwittert?
- ▷ Fällt die Antwort auf Frage 2) regional unterschiedlich aus?
- ▷ Welche der Sprachen Deutsch, Englisch oder Französisch sind auf Twitter am beliebtesten?
- ▷ Vermutlich stellen Sie in Aufgabe 4 fest, dass nur wenige deutsche Tweets vorhanden sind. Eine Erklärung könnte sein, dass in Deutschland Twitter kaum genutzt wird. Überprüfen Sie diese Hypothese. Gibt es weitere mögliche Erklärungen?
- ▷ Sind Twitter-User mit mehr Followern auch selbst aktiver?
- ▷ Warum reicht ein Datenstromsystem alleine nicht aus, um die Trends auf der Startseite von Twitter zu generieren?

Später kann dann versucht werden, eine komplexere Fragestellung auf empirische Weise zu beantworten. Verschiedene Ideen hierzu sind die folgenden Fragen bzw. Thesen, die mithilfe von SNAP*!*/Twitter (z.T. eingeschränkt) überprüft werden können, sodass es sicherlich ein wichtiger Aspekt ist, die durchgeführte Analyse kritisch zu reflektieren:

- ▷ In eine Studie wird behauptet, dass Menschen in Japan wesentlich weniger soziale Kontakte im realen Leben pflegen als solche in den USA.
- ▷ Menschen in den USA haben oft die Lieblingsfarbe Blau – in Europa ist hingegen oft Grün.
- ▷ Im Internet sprechen die meisten Personen eher Englisch als ihre Muttersprache.
- ▷ Thema A ist derzeit mehr diskutiert als B.
- ▷ Personen aus England sind im Internet oft wesentlich beliebter.
- ▷ Die Nutzung von sozialen Medien wie Twitter ist regional geprägt – in welchen Ländern ist Twitter weniger beliebt?

Technische Hintergründe

Um die notwendigen Funktionalitäten zum Zugriff auf Twitter und zur Analyse der Daten in SNAP*!* zu implementieren, musste insbesondere ein Zugriff auf die Twitterdaten ermöglicht werden. Hierzu kann auf die gut dokumentierte Twitter-API https://developer.twitter.com/ direkt oder über diverse bereits vorgefertigte Bibliotheken zugegriffen werden. Dies ist jedoch nicht direkt aus SNAP*!* selbst möglich: Obwohl die Twitter-Bibliothek grundsätzlich auf Zugriffen über REST-Schnittstellen basiert, die direkt aus dem Browser in dem SNAP*!* läuft möglich wären, stellen die in allen bekannten Webbrowsern implementierten Sicherheitsmaßnahmen hier eine Schwierigkeit dar. Da es sich bei SNAP*!* um eine JAVASCRIPT-basierte Browseranwendung handelt, unterliegt diese u.a. Schutzmaßnahmen zur Verhinderung von Cross-Site-Scripting, sodass der JAVA-

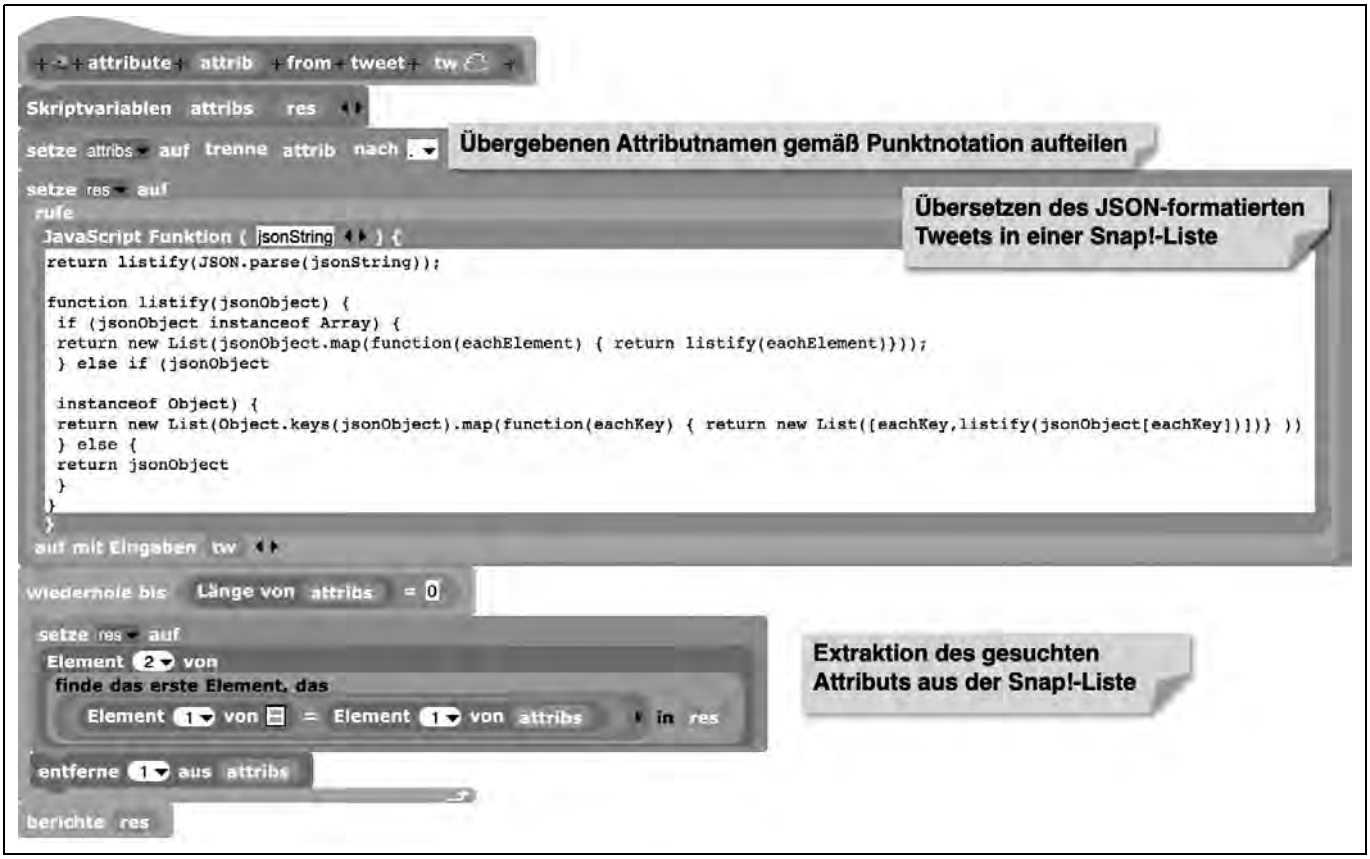


Bild 4: Code des Blocks zum Auslesen eines Attributs aus einem (noch in JSON-Format) übergebenen Tweets.

SCRIPT-basierte Zugriff auf andere Webseiten bzw. deren Daten entsprechend eingeschränkt ist. Insbesondere wird ein direkter Zugriff aus SNAP*!* auf die Twitter-API somit effizient unterbunden und ist innerhalb des Webbrowsers nicht direkt möglich. Um solche Zugriffe zu ermöglichen, müsste die angefragte Webseite (in diesem Fall Twitter) diese Zugriffe explizit erlauben, indem ein CORS-HTTP-Header gesetzt wird, der definiert, welche konkreten Domains/IPs zugreifen dürfen (*CORS* bezeichnet das ***C**ross-**O**rigin **R**esource **S**haring*, einen Mechanismus, der es durch Setzen eines HTTP-Headers auf einer Webseite anderen Webseiten erlaubt, trotz der *Same-Origin-Policy* auf diese bzw. ihre Daten zuzugreifen). Da dies praktisch nicht umsetzbar ist, muss der seitenübergreifende JAVASCRIPT-Zugriff daher vermieden werden, indem eine vorgeschaltete Anwendung genutzt wird, die als Proxy fungiert, indem sie die Daten von Twitter abfragt und diese über eine HTTP-basierte REST-Schnittstelle unter Nutzung des entsprechenden HTTP-Headers so zur Verfügung stellt, dass SNAP*!* darauf zugreifen kann. In einer ersten Version von SNAP*!*Twitter wurde hier eine lokale Anwendung genutzt, die auf den Client-PCs direkt zur Anwendung kam. Dies hatte jedoch insbesondere den Nachteil, dass diese auf allen PCs gestartet werden musste, möglicherweise Firewall-Probleme auftraten, da ein lokaler Port geöffnet werden musste, und insbesondere auch jeder PC als eigener Client gegenüber der Twitter-API auftrat, sodass entsprechende Zugangsdaten nötig waren. Um eine flexiblere Nutzung zu ermöglichen, wurde daher in der zweiten Version auf einen komplett serverbasierten Ansatz umgestiegen, sodass die Proxyanwendung nun am SNAP*!*Twitter-Server dauerhaft läuft und somit die Daten beliebig vielen Clients direkt zur Verfügung stellt, sodass für alle Nutzer nur noch die SNAP*!*Twitter-Webseite aufgerufen werden muss und somit technische Probleme effizient vermieden werden.

Im Hintergrund wird durch die Proxyanwendung gleichzeitig die Aufgabe des Webservers für die SNAP*!*Twitter-Bedienungsoberfläche selbst als auch die Abfrage der Daten von Twitter und das entsprechende Caching der Daten übernommen. Dazu wird beim Start der Serveranwendung eine Verbindung zur Twitter-API hergestellt und durch die API-Keys des Serverbetreibers die Verbindung authentifiziert, eine definierte Menge an Tweets heruntergeladen und nach einer bestimmten Leerlaufzeit die Verbindung zur API wieder getrennt und erst wieder hergestellt, wenn weitere Tweets benötigt werden (durch Nutzung von SNAP*!* Twitter). Alle Tweets werden durch die Serveranwendung – wie von Twitter empfangen – direkt in JSON-Notation gespeichert und auch SNAP*!* so zur Verfügung gestellt, sodass keine Manipulation stattfindet und eine weitere Verarbeitung somit direkt in SNAP*!* möglich ist. Der Austausch der Daten zwischen SNAP*!*Twitter und der Hintergrundanwendung findet über eine REST-Schnittstelle statt, wobei die Serveranwendung hier eben den zuvor angesprochenen CORS-Header setzt um Probleme beim Zugriff zu vermeiden. Diese Abfra-

gen wurden vor dem Nutzer „versteckt“, indem sie in SNAP*!*/Twitter in Blöcken gekapselt wurden; aufgrund der nativen Implementierung sind sie jedoch durch jeden Nutzer einseh- und bearbeitbar, wodurch Erweiterungen und Modifikationen einfach möglich werden. Die implementierten Blöcke basieren insbesondere auf der Abfrage der Daten mittels des SNAP*!*-HTTP-Blocks sowie der Interpretation der erhaltenen JSON-formatierten Tweets und der Extraktion relevanter Daten aus diesen. Dies ist am Beispiel des Blocks zum Auslesen eines Attributs aus einem Tweet in Bild 4 (vorige Seite) dargestellt.

Neben den Möglichkeiten zum Zugriff auf Twitterdaten wurde SNAP*!* weiterhin so erweitert, dass Analyseergebnisse einfach auf Karten und in Diagrammen darstellbar sind, da diese wichtige Möglichkeiten zur Visualisierung von Ergebnissen darstellen. Dazu wurde jedoch keine eigene Implementierung vorgenommen, sondern auf bewährte JAVASCRIPT-Bibliotheken aufgebaut, die aufgrund des Aufbaus von SNAP*!* unter Nutzung der zur Verfügung stehenden JAVASCRIPT-Blöcke problemlos mit diesem verknüpft werden konnten. Zur Realisierung der Kartendarstellungen wurde die Bibliothek *leaflet.js* (https://leafletjs.com/) genutzt, während die Diagrammfunktionalitäten durch *plotly.js* (https://plotly.com/) bereitgestellt werden.

Ein Betrieb eines eigenen SNAP*!*/Twitter-Servers ist prinzipiell möglich, indem der Quellcode der in *node.js* geschriebenen Anwendung von GitHub verwendet wird:

https://github.com/AGrillenberger/SnapTwitter2

Schlussbemerkung

Das hier vorgestellte Werkzeug SNAP*!*/Twitter erlaubt es, eigene Datenanalysen basierend auf einer spannenden, alltagsnahen und auch für reale Analysen relevanten Datenquelle zu durchzuführen. Dabei können Kompetenzen bzw. Kompetenzbereiche erreicht werden, die im klassischen Datenbankunterricht weniger Bedeutung erreichen: Während diesem eine eher auf die Datenspeicherung fokussierte Betrachtung zugrunde liegt, kann anhand von SNAP*!*/Twitter die Analyse selbst und auch die Visualisierung der Daten in den Vordergrund rücken und die klassische Sichtweise entsprechend ergänzen. Dies ist auch im Datenlebenszyklus (siehe Bild 5) klar erkennbar.

Durch die Nutzung einer blockbasierten Programmiersprache wird die Einstieghürde möglichst niedrig gehalten, sodass erste Analysen ohne besondere Vorkenntnisse möglich sind. Gleichzeitig wird aber durch die Implementierung umfassender Möglichkeiten zum Zugriff auf die Twitterdaten und durch Bibliotheken zur Karten- und Diagrammdarstellung ein großer Umfang an Funktionalitäten bereitgestellt und somit werden auch komplexere Analysen ermöglicht.

Entsprechend eignet sich SNAP*!*/Twitter, wie auch anhand der zuvor skizzierten Beispielanalysen gezeigt, sehr gut für einen einfachen und motivierenden Einstieg in die Thematik der Datenanalysen und eröffnet einen Blick auf andere Bereiche des Datenlebenszyklus und des Datenmanagements als der klassische Datenbankunterricht.

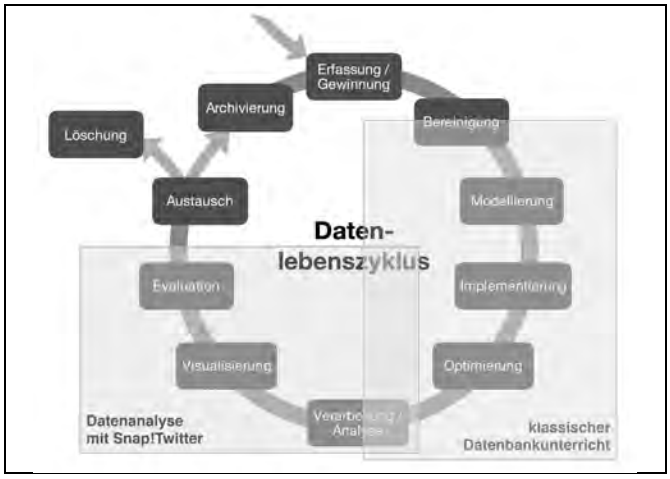


Bild 5:
Fokusbereiche des klassischen Datenbankunterrichts und des Unterrichts mit SNAP*!*/Twitter im Vergleich.

Dr. Andreas Grillenberger
Freie Universität Berlin
Didaktik der Informatik
Königin-Luise-Straße 24–26
14195 Berlin

E-Mail: andreas.grillenberger@fu-berlin.de

Literatur und Internetquellen

Dwoskin, E.: In a Single Tweet, as Many Pieces of Metadata as There Are Characters. 2014.
https://www.wsj.com/articles/BL-DGB-35668

Grillenberger, A.; Romeike, R.: Daten im Informatikunterricht – Schlüsselkonzepte des Datenmanagements als Grundlage für die Förderung von Datenkompetenzen im Unterricht. 2019.
https://datamanagement.education/pub/2019-Broschuere.pdf

ISB – Staatsinstitut für Schulqualität und Bildungsforschung München: LehrplanPLUS Bayern – Gymnasium – Jahrgangsstufe 9 – Grundlegende Kompetenzen (Jahrgangsstufenprofile) – Grundlegende Kompetenzen zum Ende der Jahrgangsstufe 9 (gültig ab Schuljahr 2021/22) – Informatik. 2020.
https://t1p.de/5vvn

Krikorian, R.: New Tweets per second record, and how! 2013.
https://t1p.de/0cuo

Alle Internetquellen wurden zuletzt am 18. Februar 2021 geprüft und können auch aus dem Service-Bereich des LOG IN Verlags (https://www.log-in-verlag.de/) heruntergeladen werden.