TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Umweltwissenschaften Fachrichtung Geowissenschaften, Institut für Kartographie

Master-Arbeit

Konzeption und Entwicklung eines automatisierten Workflows zur geovisuellen Analyse von georeferenzierten Textdaten(strömen) / Microblogging Content

Mathias Gröbe
Geboren am: 1.1.1991 in Pirna
Matrikelnummer: 3659701

zur Erlangung des akademischen Grades

Master of Science (M. Sc.)

Betreuer
Prof. Dr.-Ing. Dirk Burghardt
Dr. Alessio Bertone

Eingereicht am: 30.9.2015

TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Umweltwissenschaften Fachrichtung Geowissenschaften, Institut für Kartographie

Aufgabenstellung für die Anfertigung einer Master-Arbeit

Studiengang:	Geoinformationstechnologien
Name:	**Mathias Gröbe**
Matrikelnummer:	3659701
Titel:	**Konzeption und Entwicklung eines automatisierten Workflows zur geovisuellen Analyse von georeferenzierten Textdaten(strömen) / Microblogging Content**

Ziele der Arbeit

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Schwerpunkte der Arbeit

- Recherche
- Analyse
- Entwicklung eines Konzeptes
- Anwendung der entwickelten Methodik
- Dokumentation und grafische Aufbereitung der Ergebnisse

Betreuer:	Prof. Dr.-Ing. Dirk Burghardt Dr. Alessio Bertone
Ausgehändigt am:	1.5.2015
Einzureichen am:	30.9.2015

Selbstständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Master-Arbeit mit dem Titel *Konzeption und Entwicklung eines automatisierten Workflows zur geovisuellen Analyse von georeferenzierten Textdaten(strömen) / Microblogging Content* selbstständig und ohne unzulässige Hilfe Dritter verfasst habe. Es wurden keine anderen als die in der Arbeit angegebenen Hilfsmittel und Quellen benutzt. Die wörtlichen und sinngemäß übernommenen Zitate habe ich als solche kenntlich gemacht. Es waren keine weiteren Personen an der geistigen Herstellung der vorliegenden Arbeit beteiligt. Mir ist bekannt, dass die Nichteinhaltung dieser Erklärung zum nachträglichen Entzug des Hochschulabschlusses führen kann.

Dresden, 30.9.2015

Mathias Gröbe

Zusammenfassung

Die vorliegende Masterarbeit behandelt den Entwurf und die exemplarische Umsetzung eines Arbeitsablaufs zur Aufbereitung von georeferenziertem *Microblogging Content*. Als beispielhafte Datenquelle wurde Twitter herangezogen. Darauf basierend, wurden Überlegungen angestellt, welche Arbeitsschritte nötig und mit welchen Mitteln sie am besten realisiert werden können.

Dabei zeigte sich, dass eine ganze Reihe von Bausteinen aus dem Bereich des Data Mining und des Text Mining für eine Pipeline bereits vorhanden sind und diese zum Teil nur noch mit dem richtigen Einstellungen aneinandergereiht werden müssen. Zwar kann eine logische Reihenfolge definiert werden, aber weitere Anpassungen auf die Fragestellung und die verwendeten Daten können notwendig sein.

Unterstützt wird dieser Prozess durch verschiedenen Visualisierungen mittels Histogrammen, Wortwolken und Kartendarstellungen. So kann neues Wissen entdeckt und nach und nach die Parametrisierung der Schritte gemäß des Prinzipien des *Geovisual Analytics* verfeinert werden. Für eine exemplarische Umsetzung wurde nach der Betrachtung verschiedener Softwareprodukte die für statistische Anwendungen optimierte Programmiersprache R ausgewählt. Abschließend wurden die Software mit Daten von Twitter und Flickr evaluiert.

Abstract

This Master's Thesis deals with the conception and exemplary implementation of a workflow for georeferenced *Microblogging Content*. Data from Twitter is used as an example and as a starting point to think about how to build that workflow.

In the field of Data Mining and Text Mining, there was found a whole range of useful software modules that already exist. Mostly, they only need to get lined up to a process pipeline using appropriate preferences. Although a logical order can be defined, further adjustments according to the research question and the data are required.

The process is supported by different forms of visualizations such as histograms, tag clouds and maps. This way new knowledge can be discovered and the options for the preparation can be improved. This way of knowledge discovery is already known as *Geovisual Analytics*. After a review of multiple existing software tools, the programming language R is used to implement the workflow as this language is optimized for solving statistical problems. Finally, the workflow has been tested using data from Twitter and Flickr.

Inhaltsverzeichnis

1. Einleitung **1**

2. Aktueller Forschungsstand **5**
2.1. Nutzergenerierte Daten . . . 5
2.1.1. Definition . . . 5
2.1.2. Arten . . . 5
2.1.3. Microblogging Content . . . 6
2.1.4. Bedeutung und Möglichkeiten . . . 6
2.2. Quellen für Nutzergenerierte Daten . . . 8
2.2.1. Twitter . . . 8
2.2.2. Flickr . . . 10
2.3. Beispielanwendungen . . . 10
2.3.1. Twitter #INTERACTIVE . . . 10
2.3.2. Onemilliontweetmap . . . 12
2.3.3. Tweetping . . . 13
2.3.4. Trendsmap . . . 13
2.3.5. followerwonk . . . 15
2.3.6. Zusammenfassung zu den Beispielanwendungen . . . 17
2.4. Data Mining . . . 17
2.4.1. Geovisual Analytics . . . 18
2.4.2. Text Mining . . . 18
2.4.3. Maschinelles Lernen . . . 18
2.4.4. Sentiment Analyse . . . 19
2.4.5. Georeferenzierung von Text auf der Basis der Inhalte . . . 19

3. Methoden **21**
3.1. Software zur Informationsgewinnung . . . 21
3.1.1. R . . . 22
3.1.2. Python . . . 23
3.1.3. KNIME . . . 24
3.1.4. RapidMiner . . . 25
3.1.5. Vergleich der Software . . . 26
3.2. Aufbereitung von Daten . . . 27
3.2.1. Probleme . . . 27
3.2.2. Abfolge der Aufbereitungsschritte . . . 28
3.3. Visualisierung . . . 30
3.3.1. Tabelle . . . 30

3.3.2. Histogramm . . . 32
3.3.3. Wortwolke . . . 33
3.3.4. Graphen . . . 34
3.3.5. Kartendarstellungen . . . 36

4. Gewinnung, Aufbereitung und Analyse von Microblogging Content 39
4.1. Anforderung an eine Anwendung . . . 40
4.1.1. Twitter als Beispiel . . . 40
4.1.2. Die Pipeline . . . 40
4.2. Entwurf einer Anwendung . . . 41
4.3. Umsetzung . . . 44
4.3.1. Prototyp . . . 46
4.3.2. Erweiterung . . . 47

5. Diskussion 49
5.1. Diskussion der Pipeline . . . 49
5.2. Diskussion des Prototypen . . . 50
5.3. Potentielle Anwendungsmöglichkeiten . . . 52

6. Fazit 53
6.1. Wen interessiert was in Rom? . . . 53
6.2. Zukünftige Arbeit . . . 55

Literatur 57

A. Hinweise zur Nutzung des Prototypen 61
A.1. Auswahl . . . 61
A.1.1. Beispieldatenbank . . . 61
A.1.2. MySQL . . . 61
A.1.3. Twitter . . . 62
A.2. Filtern . . . 62
A.3. Zeichen filtern . . . 62
A.4. Wörter filtern . . . 62
A.5. Analyse . . . 63
A.6. Export . . . 63
A.7. Einstellungen . . . 63

B. Abbildungen 65

C. Tabellen 71

Abbildungsverzeichnis

1.1. Darstellung der Sprachen von Tweets in Europa. (Fischer, 2011) 2

2.1. Das offizielle Profil von Barack Obama auf Twitter. 7
2.2. Wiedergabe aller Tweets mit einer Koordinatenangabe seit 2009 visualisiert im Jahr 2013. Ein farbiger Punkt steht jeweils für einen Tweet. Bei weiterer Überlagerung sind diese wieder heller dargestellt. (Ríos, 2013b) 9
2.3. Anhand von Fotometadaten rekonstruierte Reiserouten. Blau die Touren von Einheimischen, rot die von Touristen. (Fischer, 2010) 11
2.4. Verschiedene Visualisierungen von Twitter basierend auf eigenen Daten im Überblick. 12
2.5. Ein geöffneter Tweet auf der der *One million tweet map* in Dresden. 13
2.6. Die Startseite von Tweetping mit aufblitzenden Lichtern für Tweets. Einzelne Texte werden zum Teil ebenfalls kurzzeitig angezeigt. 14
2.7. Visualisierung von Tweetping für Starbucks. 14
2.8. Trends in Europa dargestellt auf Trendsmap am 10. September 2015. 15
2.9. Ein Teil der Auswertung der Follower auf `https://followerwonk.com/analyze/` für den Nutzer `@RDataMining` im sozialen Netzwerk Twitter. 16

3.1. Visualisierung von Tweets auf einer Karte, die zuvor über die Twitter-API abgefragt wurden im RStudio. 22
3.2. Abfrage einer *Usertimeline* über die Twitter-API mit Python. 23
3.3. Einblick in die Arbeitsumgebung von KNIME mit einem Modell für Text Mining. 24
3.4. Die Arbeitsoberfläche des RapidMiners. 25
3.5. Ablauf der Aufbereitung. 29
3.6. Anteile der Sprachen größer als 1 %, dargestellt als Histogramm. 33
3.7. Darstellung der 250 häufigsten Begriffe als Wortwolke. 34
3.8. Darstellung der Wörter und ihrer Korrelation ab einer Häufigkeit von 400 Vorkommen. 35
3.9. Verteilung der Tweets im April 2015. Je blauer eine Region ist, desto mehr Tweets wurden dort verortet. 36
3.10. Verteilung der Tweets in vier verschieden Sprachen im April 2015. 37
3.11. Verteilung der Tweets in den vier verschieden Sprachen im April 2015. Je blauer eine Region ist, desto mehr Tweets wurden dort verortet. 38

4.1. Übersicht über den Arbeitsablauf. 42
4.2. Entwurf der Arbeitsoberfläche. 43

4.3. Beispiel für den Ablauf der Aufbereitung und Auswirkungen der Schritte. Rot markierte Zeichen werden gelöscht. 45
4.4. Eine Ansicht des Prototypen. 47

5.1. Beispiel für HTML-Code und Sonderzeichencodierungen in den Flickr-Daten 51

6.1. Vergleich der Sprachverteilungen in Tweets: oben Englisch unten Italienisch, dazu sind die Hauptverkehrsstraßen dargestellt. Markiert sind folgende markanten Orte: ① Petersplatz, ② *Piazza del Popolo*, ③ Pantheon, ④ Kolosseum, ⑤ *Roma Termini* (Hauptbahnhof) 54

B.1. Auswahl 66
B.2. Filtern 66
B.3. Zeichen filtern 67
B.4. Wörter Filtern 67
B.5. Analyse 68
B.6. Export 68
B.7. Einstellungen 69

Tabellenverzeichnis

3.1. Übersicht über die verglichene Software. 26
3.2. Einblick in den Beispieldatensatz, es wird hier nur eine Auswahl der Felder wiedergegeben. 31
3.3. Auflistung aller Sprachen und ihrer Häufigkeit im Beispieldatensatz. 32

5.1. Einige Beispiele für häufig verwendete Abkürzungen auf Twitter. (Beal, 2014) 50

C.1. Tabelle mit allen in Prototypen verwendeten R-Pakten. Sie können unter `https://cran.r-project.org/` nachgeschlagen oder heruntergeladen werden. 71
C.2. Anforderungen an die zu entwickelnde Software und ihre Umsetzung. 73

Abkürzungen

API	Application Programming Interface
CSV	Comma-separated values
ETL	Extraction-Transformation-Loading
GIS	Geoinformationstystem
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
KDD	Knowledge Discovery in Databases
KNIME	Konstanz Information Miner
NLP	Natural Language Processing
OSM	Openstreetmap
POS	Part-of-Speech
REST	Representational State Transfer
SQL	Structured Query Language
UGC	User-generated content
URL	Uniform Resource Locator
UTF-8	8-Bit Universal Character Set Transformation Format
VGI	Volunteered Geographic Information
WGS84	World Geodetic System 1984

1. Einleitung

Das Thema der Masterarbeit war schon im Wesentlichen umrissen, als ich im März 2015 Rom besuchte. Dabei beobachtete ich, wie der bunte Besucherstrom fleißig seine Smartphones zur Erstellung von Erinnerungsfotos nutze, die sicher auf früher oder später in sozialen Netzwerken landen würden. Wenn man schon einmal in der „Ewigen Stadt“ zu Besuch ist, möchte man natürlich auch die Freunde und Follower daran teilhaben lassen. Dazu markiert man am besten noch den Ort in der geteilten Nachricht. Dadurch ist dem Leser klar, wo man sich befindet und ein eindeutiger Ortsbezug wird hergestellt.

Vor Ort beobachtete ich, dass in der Umgebung des Vatikans mehr Deutsch gesprochen wurde als anderswo in der Stadt. Sind etwa die Deutschen, oder präziser ausgedrückt die deutschsprachigen Besucher Roms besonders interessiert am Petersdom und den Vatikanischen Museen? Man könnte als Begründung anführen, dass ein besonderes Interesse herrscht, da bis vor kurzem der Pontifex ein Deutscher war. Wie überprüft man nun aber diese Vermutung über die Interessen der Touristen?

Bei der Recherche für die Masterarbeit hatte mich Herr Prof. Burghardt auf eine Karte mit der Visualisierung der Sprachen von Tweets aufmerksam gemacht. Die Karte ist in Abbildung 1.1 zu sehen. Deutlich sind viele Ländergrenzen erkennbar, abgebildet durch die unterschiedlichen Sprachen. Könnte man so etwas nicht auch für Rom erstellen? Dann wüsste man, welche Sprachgruppe sich für welche Attraktionen mehr interessiert. Das Ergebnis dürfte für Stadtführer, Redaktionen von Reiseführern und Verwaltungen durchaus interessant sein. Sie könnten dann besser ihre Angebote auf die Nachfrage der jeweiligen Zielgruppe anpassen.

Für die Untersuchung müsste zunächst erstmal eine Datengrundlage geschaffen werden. Daran kann sich dann die Visualisierung und Auswertung anschließen. Dazu ist es nötig die Daten zu sammeln und aufzubereiten. Da es dafür bisher noch kein generisches Vorgehensmodell gibt, beginnt hier bereits die Forschung und damit das wesentliche Thema dieser Masterarbeit. Es soll eine Pipeline ausgearbeitet werden, die es ermöglicht diese Daten aufzubereiten. Als explizites Beispiel soll dabei Twitter dienen.

Die einfachste und bekannteste Datenquelle für *Microblogging Content* ist die Plattform Twitter. Sich selbst bezeichnet das Soziale Netzwerk als den „globalen Marktplatz“. (Twitter, 2012) Gehandelt werden auf Twitter in der Regel Informationen und keine Waren. Es wird favorisiert, geteilt, retweetet und gefolgt – um die Plattform selbst hat sich schon ein Komplex an Begriffen gebildet, der die Interaktionen dort beschreibt.

Politiker vermelden hier in 140 Zeichen langen Nachrichten Neuigkeiten, Freunde teilen ihre Erlebnisse, Bots posten aktuelle Verkehrs- und Wetterdaten oder Unternehmen bieten Support an. Die Anzahl der Nutzungsmöglichkeiten sind vielfältig und nahezu weltweit verbreitet. Voraussetzung für die Teilnahme ist ein internetfähiges Gerät und schon ist man selbst ein Teil des großen Ganzen. Eine kostenlose Anmeldung ist natürlich nötig, wobei ein Nutzername und eine E-Mail-Adresse angegeben werden muss.

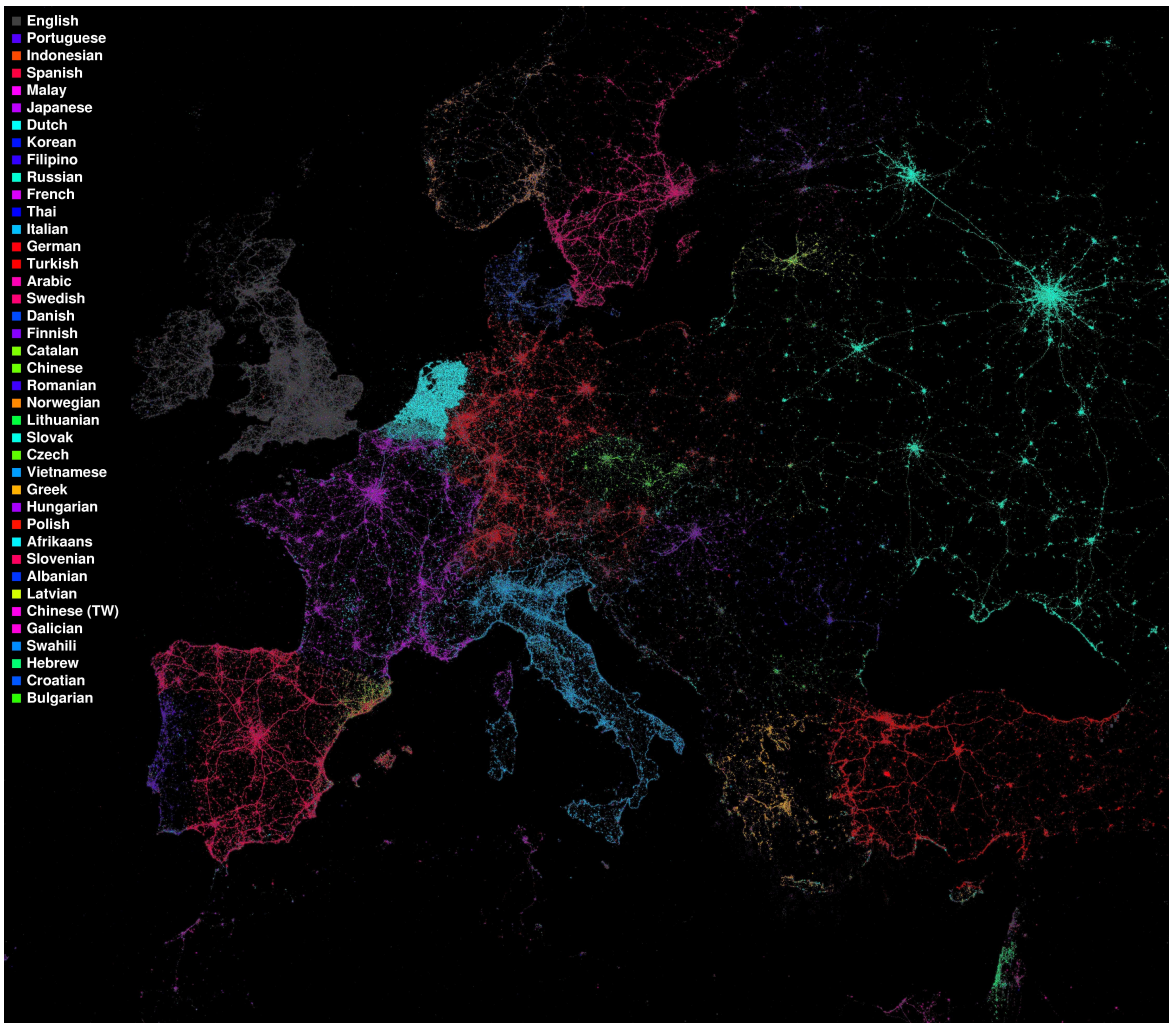

Abbildung 1.1.: Darstellung der Sprachen von Tweets in Europa. (Fischer, 2011)

Zur Aufbereitung der Daten soll eine exemplarische Software entwickelt werden, welche die Abläufe vereinfacht, vor allem im Hinblick auf die Arbeit von Daten mit Ortsbezug. Die eigentliche Analyse soll hierbei im Hintergrund stehen und nur vorbereitet werden. Darauf aufbauend können später interessantere Fragestellungen entworfen werden, die mehr Nutzwert als das oben angeführte Beispiel haben und auch kommerziell verwendbar sind.

Einige Beispiele sollen aber auch schon hier benannt werden: So untersucht Radzikowski, Hollen und Fuhrmann, 2015 beispielsweise auf der Basis von Twitter, Flickr und YouTube die Diskussion in den USA über Zusammenhang zwischen Sonnenbräunen und Hautkrebs. Aus Basis dieser Medien konnte die Intensität der Auseinandersetzung mit dem Themenkomplex in den verschiedenen Bundesstaaten über mehrere Monate beobachtet werden.

Beim Hochwasser 2013 organisierten freiwillige Helfer über soziale Netzwerke Hilfe für die Betroffenen und koordinierten ihr Vorgehen. Genauso informierten aber auch Städte und Organisationen auf Twitter über das aktuelle Geschehen, bedankten sich für Hilfe, baten um Hilfe oder kommentierten einfach nur das Geschehen. (Janssen, 2013) Die professionellen Helfer kamen teilweise gar nicht so recht hinterher. Könnte man also nicht versuchen, in Echtzeit Informationen zur Lage aus den sozialen Netzwerken zu gewinnen, um schneller auf solche Entwicklungen reagieren zu können? Diese könnten dann in bestehende Informations- und Koordinierungssysteme wie MobiKat[1] integriert werden.

[1] `http://www.mobikat.net/`

Genauso könnten Hersteller beobachten, wie Kunden in sozialen Netzwerken über ihre Produkte urteilen und so kostengünstig eine Rückmeldung bekommen, die unvoreingenommen ist und sogar noch einen regionalen Bezug hat. So kann es sein, dass eine neues Automodell in Süddeutschland gut ankommt, aber im Norden nicht so beliebt ist. Dabei sollte aber auch die Privatsphäre nicht außer Acht gelassen werden. Zwar ist Vieles möglich, aber nicht immer auch erlaubt. Es sollte nur öffentlich zugängliches Material genutzt werden und immer die Rechte und die Anonymität des Einzelnen gewahrt bleiben.

2. Aktueller Forschungsstand

2.1. Nutzergenerierte Daten

Der *Micoblogging Content* gehört mit zu den nutzergerierten Daten, welche im Englisch als *User-generated content (UGC)* bezeichnet werden. Dabei werden die Inhalte auf Webseiten nicht vom Seitenbetreiber erstellt, sondern von Nutzern der Webseite hinzugefügt. Andere Nutzer können dann auf die Inhalte zugreifen, sie betrachten oder auch selbst bearbeiten.

2.1.1. Definition

Ein Vorschlag einer Definition nach (Bauer, 2010, S. 5), lautet:

> „User Generated Content“ bezeichnet die Gesamtheit aller von Internetnutzern bewusst erzeugten wahrnehmbaren elektronischen Medieninhalte, die von diesen unmittelbar und unabhängig von einer vorherigen redaktionellen Auswahl über das Internet der Öffentlichkeit zugänglich gemacht werden, sofern es sich hierbei nicht um professionell erstellte und zu gewerblichen Zwecken veröffentlichte Inhalte handelt.

Die Definition ist absichtlich sehr allgemein gehalten, da die Ausprägungen sehr verschieden sein können. So wird zunächst einmal alles eingeschlossen, auch im Hinblick auf die schnelle Entwicklung in diesem Bereich.

2.1.2. Arten

Bauer, 2010 selbst unterscheidet nutzergenerierte Daten in Text, Bild, Audio und Video UGC. Diese Unterteilung ergibt aus seinem Blickwinkel mit Schwerpunkt auf die rechtliche Situation Sinn. Aber sie lässt zum Beispiel die geographischen Daten aus und legt den Fokus sehr auf den Medientyp. Dieser Typ von Daten wird daher auch als Spezialfall des UGC unter dem Namen *Volunteered Geographic Information (VGI)* beschrieben.

Das beste Beispiel für VGI ist das Openstreetmap (OSM) Projekt. Auf der Basis von Satellitenbildern können hier die Nutzer Daten erfassen und mit ihrem eigenen lokalen Wissen kombinieren. Durch die Plattform werden Hilfen zur Datenerfassung bereitgestellt, so dass auch fachfremde Nutzer mit den Werkzeugen arbeiten können. Problematisch ist dabei aber die Überprüfung von Informationen, wie bei allen nutzergenerierten Daten. Es ist einfach, absichtlich oder unbewusst, falsche Informationen beizutragen.

2.1.3. Microblogging Content

Das *Microblogging* (deutsch Mikroblogging) beschreibt eine spezielle Form des Bloggens. Dabei werden kurze Nachrichten veröffentlicht, die wenige Informationen enthalten, welche nicht in die Tiefe gehen und versuchen Aussagen auf den Punkt zu bringen. Beim gewöhnlichen Blogging nimmt sich der Autor hingegen mehr Zeit zum Schreiben und Ausformulieren seines *Posts* (deutsch für Beitrag).

Dabei können Texte, Links zu anderen Inhalten, Bilder oder Videos Teil des Beitrags sein. Diese Inhalte sind der *Content*, welcher die Informationen transportiert und auch einen Ortsbezug haben kann. Dies ist durch den Inhalt selbst möglich, der sich auf einen Ort auf der Erde bezieht oder durch eine Koordinatenangabe in den Metadaten.

Julia H. Grace, D. Zhao und Boyd, 2010 beschreibt das *Microblogging* wie folgt:

> *[...] microblogging is only mildly reminiscent of blogging; this communication medium sits somewhere between text messages, IM status messages, blogs and social network sites.*

Viele Prominente und Firmen nutzen diese Art der Kommunikation, um mit ihren Kunden in Verbindung zu bleiben und Neuigkeiten zu verbreiten. In den letzten Jahren kamen auch öffentliche Organisationen und Medien hinzu, die ebenfalls von den Möglichkeiten vielfältig Gebrauch machen. Das klassische Beispiel dafür ist Twitter und das chinesische Pendant mit dem Namen *Sina Weibo*, welches sich auf Grund der Sperrung von Twitter in China sehr großer Beliebtheit erfreut. Ähnliche Beiträge sind noch in sozialen Netzwerken wie Facebook oder Google+ möglich, wobei die Länge der Texte dort nicht so streng reglementiert ist.

An dieser Stelle noch zu erwähnen sind Spezialfälle wie Tumblr[1], das Nutzern die Veröffentlichung von Texten, Bildern, Zitaten, Chatlogs, Links, Video- sowie Audiodateien ermöglicht und Flickr als eine sehr bekannte Plattform, auf der Bilder mit Metadaten und Beschreibungen geteilt werden können. Eine weitere, ältere Plattform ist noch Reddit[2], welche das Teilen und Kategorisieren von Beiträgen ermöglicht.

2.1.4. Bedeutung und Möglichkeiten

Das nutzergenerierte Daten in den letzten Jahren so sehr in den Schwerpunkt der Forschung gerückt sind begründet Hahmann, 2014 damit, dass die Daten mit wenigen bis keinen Kosten verfügbar sind sie dabei Vielzahl von Entscheidungen widerspiegeln und oft ortsbezogen sind. Daraus ergeben sich neue Möglichkeiten: So hat sich in der Kartographie die Art der Datenerfassung mit OSM für Karten geändert. Neben der hoheitlichen Erfassung durch Behörden hat sich noch die Community als Datenquelle etabliert. Aber auch in anderen Gebieten gab es Verschiebungen. Flickr beispielsweise kann mit seinen Bilder und Bildbeschreibungen zum Erschließen von Gebieten genutzt werden und dadurch die klassischen Bildbände zum Teil ersetzen.

Es können aber auch die Auswirkungen von Naturkatastrophen und Großereignissen beobachtet (Schade u. a., 2013) oder auch Emotionen auf Karten dargestellt werden. (Hauthal, 2015) Genauso kann das Verhältnis von Kunden zu Produkten durch eine passende Software beobachtet und so die Marketing-Strategie verbessert werden. Die in Abschnitt 2.3 beschriebenen Anwendungen lassen sich vor allem zu diesem Zweck einsetzen.

[1] `https://www.tumblr.com/`
[2] `https://www.reddit.com/`

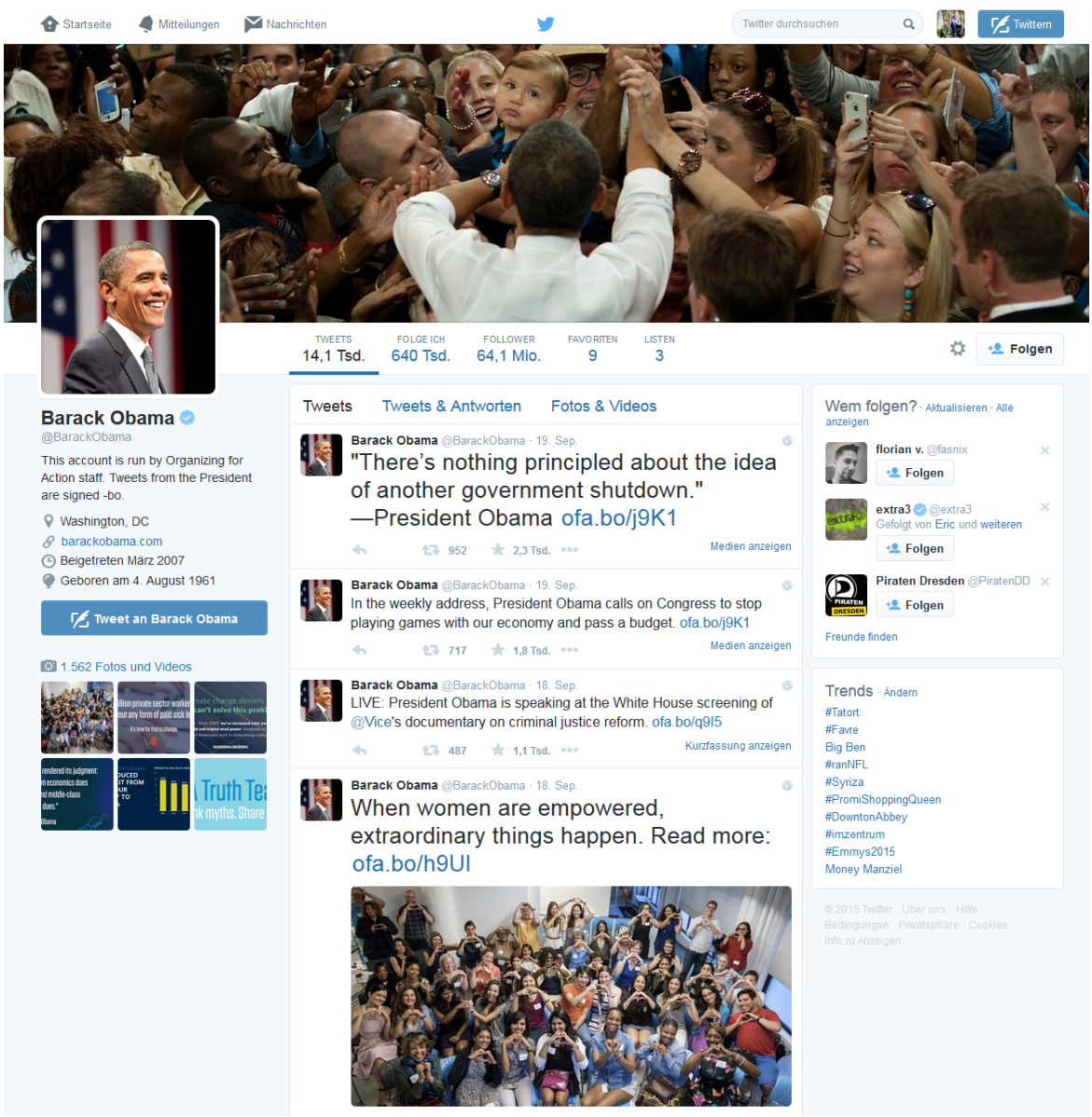

Abbildung 2.1.: Das offizielle Profil von Barack Obama auf Twitter.

2.2. Quellen für Nutzergenerierte Daten

Im folgenden Abschnitt sollen mit Twitter und Flickr zwei Plattformen als Datenquellen für *Microblogging Content* vorgestellt werden, die sehr bekannt sind und viele Nutzer haben. Auf Basis ihrer Datenbestände wird gerne geforscht, weil sie viele Daten kostenlos über Ihre APIs anbieten. Da der Schwerpunkt dieser Arbeit auf Textdaten mit Ortsbezug liegt, geht es folglich nur um Texte und nicht wie im Fall von Flickr um die Bilder, auch wenn diese durchaus eine lohnenswerte Informationsquelle darstellen können.

2.2.1. Twitter

Das soziale Netzwerk Twitter[3] ermöglicht es Nachrichten mit maximal 140 Zeichen, genannt „Tweets“, zu verbreiten und zu erhalten. Dadurch können schnell aktuelle Informationen zu vom Nutzer festgelegten Themen bezogen werden. Die Kommunikation erfolgt dabei in Echtzeit. Hierbei können Nutzer die Nachrichten anderer Nutzer kommentieren oder weiter verbreiten – *retweeten*. In den Nachrichten können neben URLs auch Bilder enthalten sein. (Twitter, 2015d)

Rund um Twitter haben sich Vokabeln und Konventionen entwickelt, um die verschiedenen Funktionen zu beschreiben. So werden Nutzernamen mit einem vorangestellten At-Zeichen `@` gekennzeichnet. Daraus wird automatisch ein Link auf das jeweilige Profil erzeugt. Ein mit einer Raute `#` vorangestelltes Wort wird als ein sogenannter *Hashtag* gekennzeichnet und stellt eine Verbindung zu einem Thema her. Klickt man auf diesen, wird nach anderen Tweets gesucht, die diesen Tag ebenfalls enthalten. (Twitter, 2015c)

Zurzeit hat Twitter 302 Millionen aktive Nutzer, die 500 Millionen Tweets am Tag verbreiten. 80 % davon verwenden dazu ein mobiles Endgerät. Die Plattform selbst unterstützt 33 Sprachen und ist weltweit verbreitet. Die Twitter Inc. besteht seit dem 19. April 2007 und hat etwa 3.900 Mitarbeiter rund um den Globus. (Twitter, 2015a)

In den letzten Jahren hat sich Twitter zu einer sehr beliebten Datenquelle für Sozialforschung entwickelt. Dazu passt gut, dass sich die Plattform selbst als *globaler Marktplatz* (*global town square*) bezeichnet und dies ausführlich auf ihren Seiten dokumentiert. Egal ob es um Politik, Sport, Naturkatastrophen oder Technik jeglicher Art geht, all diese Themen sind auf Twitter präsent. (Twitter, 2012) Neben der Aktualität und der Zugriffsmöglichkeiten durch eine frei zugänglichen API, ist ebenfalls der Standort interessant. Betrachtet man Abbildung 2.2, sieht es so aus, als würde Europa von einem Netz von Tweets umspannt. Diese lassen Städte und Verkehrswege erkennen. Wo Menschen leben scheinen sie auch zu twittern. Die Grafiken wurden auf Basis von georeferenzierten Kurznachrichten hergestellt. Die Punkte auf dem Meer lassen sich durch Fähren und Schiffe erklären. Rund um den Erdball könnte man ähnliche Bilder auf der Basis von Billionen von Tweets produzieren. (Ríos, 2013a)

Standardmäßig haben Tweets keinen direkten Ortsbezug. Erst durch die Aktivierung des Standortdienstes kann eine Markierung hinzugefügt werden. Dabei handelt es sich zunächst nicht um eine Koordinate, sondern um eine Ortsangabe. So wäre *Berlin* hierfür ein Beispiel. Eine Koordinate wird erst dann verfügbar, wenn ein „Genauer Standort“ explizit vom Nutzer geteilt wird. Twitter macht selbst auf Risiken, wie die Preisgabe der Privatanschrift, aufmerksam und gibt Empfehlungen, wann ein Ortsbezug passend wäre. Die Standortangaben können auch noch nachträglich gelöscht werden. (Twitter, 2014)

Während der *Twitter Decahouse*, siehe Leetaru u. a., 2013, enthielten 2,02 % aller Tweets geographische Metadaten. Wobei 1,6 % eine exakte Koordinate hatten und 1,8 % einen Standort enthielten. Beide Angaben waren bei 1,4 % aller untersuchten Kurznachrichten enthalten. Zu beachten ist, dass Tweets eine Koordinate und einen Standort enthalten können.

Ein interessanter Vergleich lässt sich zum *Earth City Lights Image* (NASA, 2000) ziehen.

[3] `https://twitter.com/`

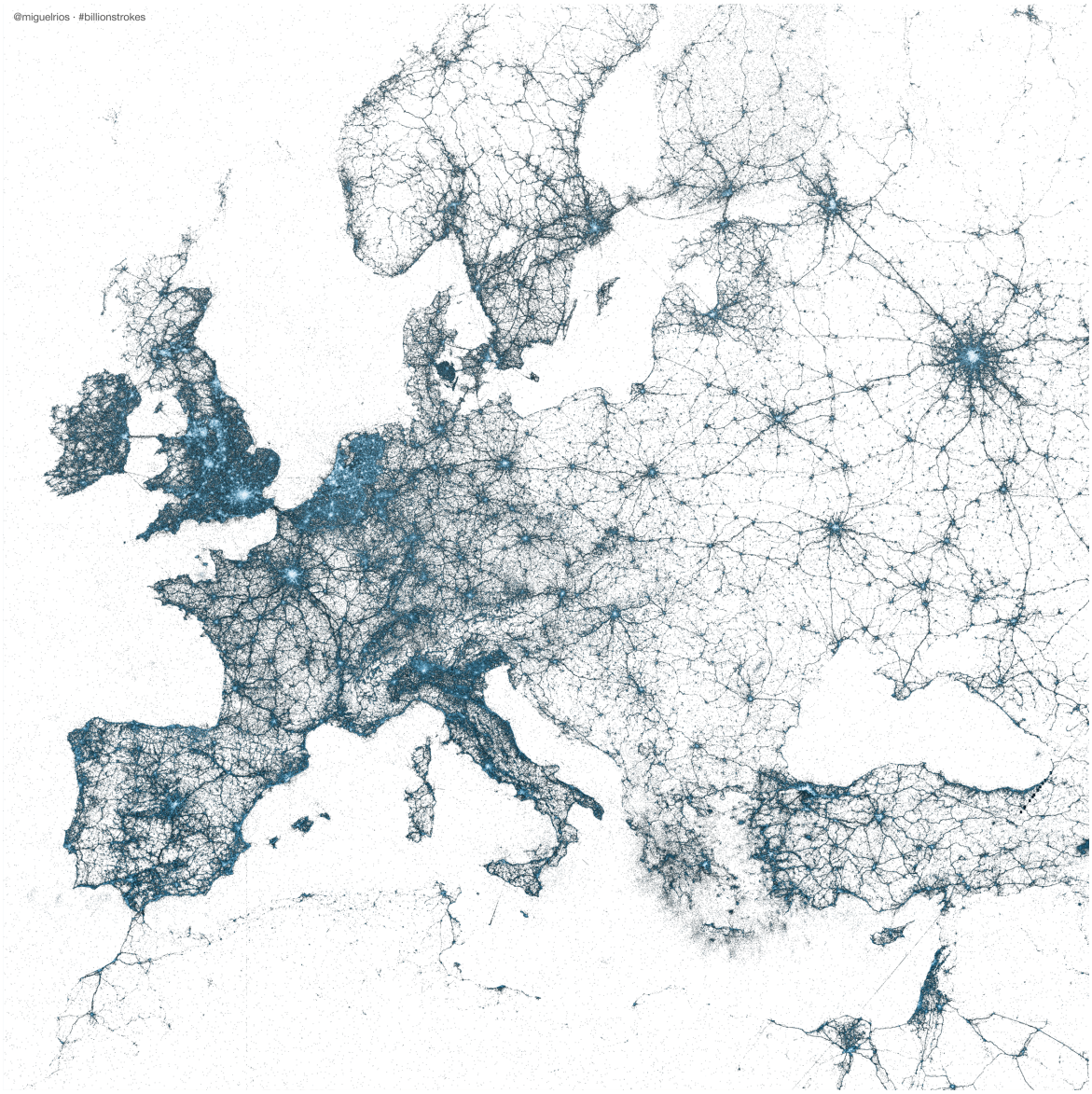

Abbildung 2.2.: Wiedergabe aller Tweets mit einer Koordinatenangabe seit 2009 visualisiert im Jahr 2013. Ein farbiger Punkt steht jeweils für einen Tweet. Bei weiterer Überlagerung sind diese wieder heller dargestellt. (Ríos, 2013b)

Auf dieser Karte wird gezeigt, wo nachts elektrisches Licht von Satelliten erfasst wurde. Die Bilder ähneln sich nicht nur optisch, auch mathematisch konnte der Zusammenhang von Leetaru u. a., 2013 nachgewiesen werden. Es ergab sich eine Übereinstimmung von fast 80 %. Twitter ist also sehr wahrscheinlich überall ein Teil des alltäglichen Lebens, wo elektrische Energie verfügbar ist. So ist die Plattform durchaus weltweit im hohen Maße repräsentativ, wenn auch in ländlichen Räumen Abstriche hingenommen werden müssen.

2.2.2. Flickr

Bei Flickr[4] handelt es sich um ein Webportal, das den Upload digitaler Bilder und kurzer Filme ermöglicht. Diese können zusätzlich zu ihren fotografischen Metadaten nach belieben noch mit einer Koordinate, einem Titel und einer Beschreibung versehen werden. Sie können dann als Alben oder einzelne Bilder mit anderen Nutzern geteilt werden, welche dann die Bilder bewerten und kommentieren können.

Die Plattform wurde zum ersten Mal im Jahr 2004 vorgestellt und 2005 von Yahoo aufgekauft. Im November 2007 wurden dann bereits über 2 Milliarden Fotos hochgeladen, 2009 folgte die erste App, zunächst für iOS, später auch für Android. Im August 2011 waren 6 Milliarden Bilder bei Flickr gespeichert. Im Jahr 2013 erfolge eine umfassende Erneuerung der Webseite. Seitdem verfügt jeder Nutzer über 1 TB kostenlosen Speicherplatz. Im Februar 2014 wurden laut eigenen Angaben jeden Tag circa 1 Million Bilder geteilt. (Kremerskothen, 2014)

Über die API von Twitter sind die Metadaten zu den öffentlichen Bildern verfügbar und lassen sich abrufen und visualisieren. So entstand unter anderem *The Geotaggers' World Atlas*[5] von Eric Fischer. Abbildung 2.3 zeigt eine Karte aus der Sammlung. Es wurden Linien zwischen den verschiedenen Orten der Fotos von einem Nutzer gezogen und so entstanden Routen, die die meistfotografierten Sehenswürdigkeiten verbinden. (Roncero, 2015)

Basierend auf den Texten zu den Bildern lassen sich auch Informationen zu den Orten ableiten und auf Karten darstellen. Dazu wurden die Texte aufbereitet und anschließend einer Skala von Emotionen zugeordnet. Zwar lassen sich so keine aktuellen Ereignisse beobachten, aber dennoch sind im Rückblick bestimmte Momente und die damit verbunden Gefühle identifizierbar. (Hauthal, 2015)

2.3. Beispielanwendungen

Im folgenden Abschnitt sollen einige Zusammenstellung, auch *Mashups* genannt, vorgestellt werden, die mit *Microblogging Content* arbeiten. Dies soll einen Überblick über die verschiedenen Varianten der Visualisierung, den aktuellen Stand der Forschung und weiterhin möglicher Anwendungen schaffen.

2.3.1. Twitter #INTERACTIVE

Als erstes Beispiel sollen Visualisierungen von Twitter selbst, basierend auf eigenen Daten, vorgestellt werden. Interessant ist hierbei nicht nur das es sich dabei um die bekannteste Plattform handelt, sondern auch die Vielfalt an erzeugten Darstellungen. Twitter sammelt sie unter dem Hashtag `#INTERACTIVE` und stellt sie ebenfalls auf einer Webseite[6] zur Verfügung.

In Abbildung 2.4 sieht man unterschiedliche Vorschaubilder auf einzelne Beiträge: Karten, welche zeigen, wie die Follower eines Fußballclub verteilt sind, einmal für die USA und einmal weltweit. Des Weiteren ist noch ein *Themeriver* zu sehen, auf dem die Menge der Follower der brasilianischen Präsidentschaftswahlkandidaten in ihrer zeitlichen Entwicklung

[4] `https://www.flickr.com/`
[5] `https://www.flickr.com/photos/walkingsf/sets/72157623971287575/`
[6] `https://interactive.twitter.com/`

Abbildung 2.3.: Anhand von Fotometadaten rekonstruierte Reiserouten. Blau die Touren von Einheimischen, rot die von Touristen. (Fischer, 2010)

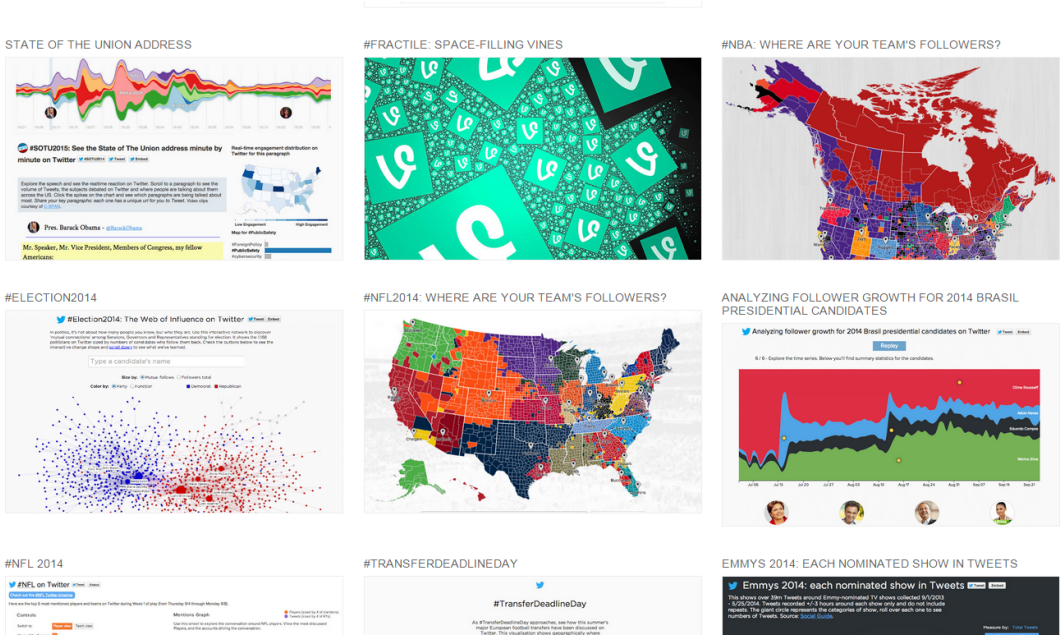

Abbildung 2.4.: Verschiedene Visualisierungen von Twitter basierend auf eigenen Daten im Überblick.

dargestellt werden. Einmal mit Mehr-Fenstertechniken dargestellt, andere mittels Graphen, die Verbindungen zwischen Personen darstellen.

Aus der Übersicht heraus sind jeweils einzelne Seiten verlinkt, die ein mal für sich selber stehen, Blogeinträge sind, Videos auf Youtube oder Bildbeiträge auf Flickr – insgesamt also ein bunter Überblick über Möglichkeiten und Themen für Visualisierungen. Einem erkennbaren Zweck dienen diese nicht.

2.3.2. Onemilliontweetmap

Bei der *One million tweet map*[7] handelt sich um eine Visualisierung für das soziale Netzwerk Twitter. Sie ermöglicht die Suche nach Begriffen oder Hashtags. Das Suchergebnis wird auf der Karte dargestellt. Entweder durch *Clustering* zu Kreisen zusammengefasst oder als *Heatmap*. Durch die Karte kann die Suche räumlich eingegrenzt werden. Eine Auswahlliste ermöglicht dies ebenfalls für den Suchzeitraum, wobei dieser maximal 6 Stunden betragen kann. Dadurch eignet sich die *One million tweet map* eher für die Beobachtung des aktuellen Geschehens.

Mit dem Laden der Seite beginnt ein Zähler zu laufen, der die Anzahl der Tweets aufaddiert. Neben den schon genannten Möglichkeiten der Filterung werden noch die häufigsten Hashtags ausgegeben. In Abbildung 2.5 ist die Karte für Dresden mit einem geöffneten Tweet zu sehen. Die Tweets sind zusammengefasst zu Kreisen, wobei die Anzahl in der Mitte steht. Diese Ansicht bietet scheinbar auch eine gewisse thematische Gruppierung, die aber auch zufällig so entstanden sein kann. Sie bietet sich für kleine Maßstäbe an, für größere Maßstäbe ist die *Heatmap* besser geeignet. Letztere verdeutlicht besser die Verteilung der Tweets. Die Entscheidung über die Darstellung bleibt aber dem Nutzer überlassen.

Insgesamt ist es ein interessantes Werkzeug zum Ausprobieren und Beobachten von Entwicklungen auf Twitter. Es ist ohne eine Dokumentation unklar, ob nur Nachrichten mit Koordinate angezeigt werden, oder versucht wurde den Ort zu detektieren. Zur besseren Übersicht bei großen Maßstäben wäre ein automatischer Wechsel der Darstellungsweise der Tweets beim Zoomen sicher praktischer.

[7] `http://onemilliontweetmap.com/`

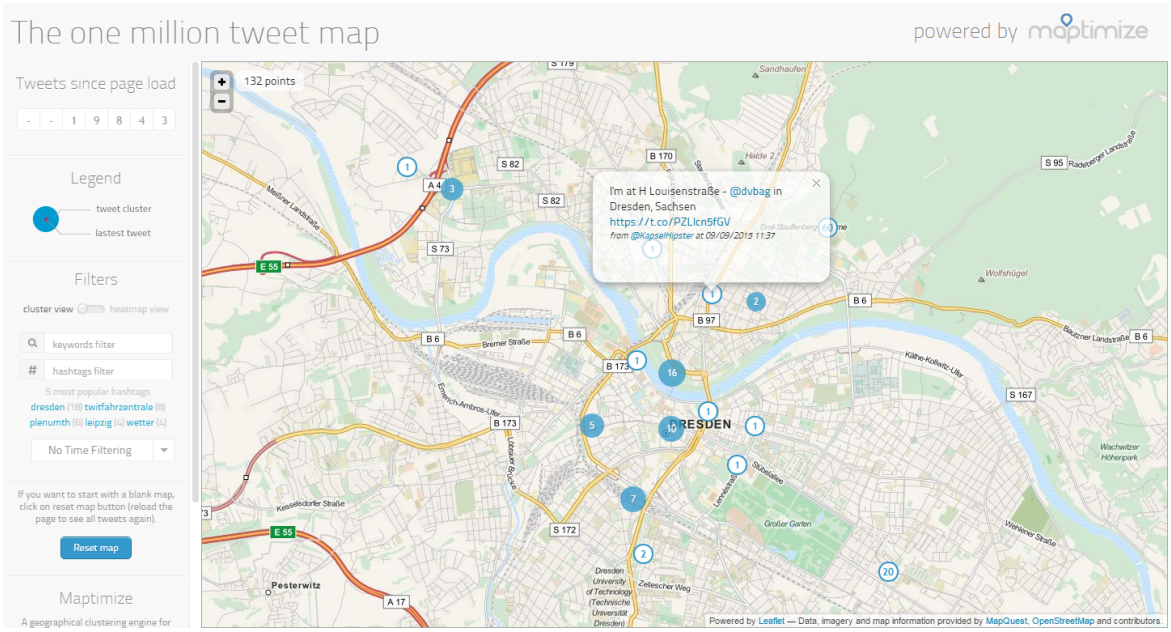

Abbildung 2.5.: Ein geöffneter Tweet auf der der *One million tweet map* in Dresden.

2.3.3. Tweetping

Die Visualisierung von Tweetping[8] fasziniert den Besucher zunächst. Hier werden Nachrichten an ihrem Erstellungsort dargestellt: als kleine Lichtblitze und bei einigen erscheint auch der Text. Abbildung 2.6 vermittelt einen Eindruck von der Webseite, welche eine Demonstration der Möglichkeiten ist und den Besucher zum verweilen auf der Seite ermuntern soll.

Besonders interessant sind die *Live Streams* zu verschiedenen Themen. Beispielhaft dafür st Starbucks in Abbildung 2.7 zu sehen. Die Tweets wurden vermutlich anhand des Hashtags `#starbucks`, der Ortsangabe `@starbucks` oder einfach dadurch, dass das Wort „starbucks“ in der Nachricht enthalten ist, identifiziert. Eine verlässliche Angabe dazu fehlt. In einem Dashboard werden die Tweets mit Text und Nutzerbildern dargestellt (rechts), angehängte Bilder angezeigt (rechts unten) und häufig enthaltene Begriffe ausgegeben (unten). Zu diesen ist ein zeitliche Verlauf verfügbar und die jeweiligen Anteile in den Tweets werden durch ein Kreissektordiagramm (links unten) visualisiert.

Tweetping bietet seinen Dienst zur Analyse explizit auf der Webseite an und wirbt mit bekannten Firmen wie Nutella, National Geographic und Xerox als Kunden. (Tweetping, 2015) Die Möglichkeiten der bereitgestellten Software wird offenbar zur Bobachtung von Kunden und ihrer Interaktion auf Twitter mit dem hergestellt Produkt genutzt. Damit handelt es sich um eine kommerzielle Anwendung mit einer definierten Zielgruppe.

2.3.4. Trendsmap

Auf der Trendsmap[9] werden aktuell häufig verwendete Begriffe auf einer Karte dargestellt, an den Orten, wo sie gepostet wurden. Als Datenbasis dafür dient Twitter. Ab einer bestimmten Zoomstufe muss dann bezahlt werden, um die Trends zu sehen. In Abbildung 2.8 ist ein Überblick für Europa zu sehen. Zu erkennen ist, dass in Deutschland offenbar zur Zeit die Flüchtlingsproblematik viel diskutiert wird.

Nach einer Registrierung ist es möglich gegen eine jährliche oder monatliche Gebühr weitere Funktionen zu nutzen. So können in der *Basic*-Version die wichtigsten Nutzer, Videos, Bilder

[8] `http://tweetping.net/`
[9] `http://trendsmap.com/`

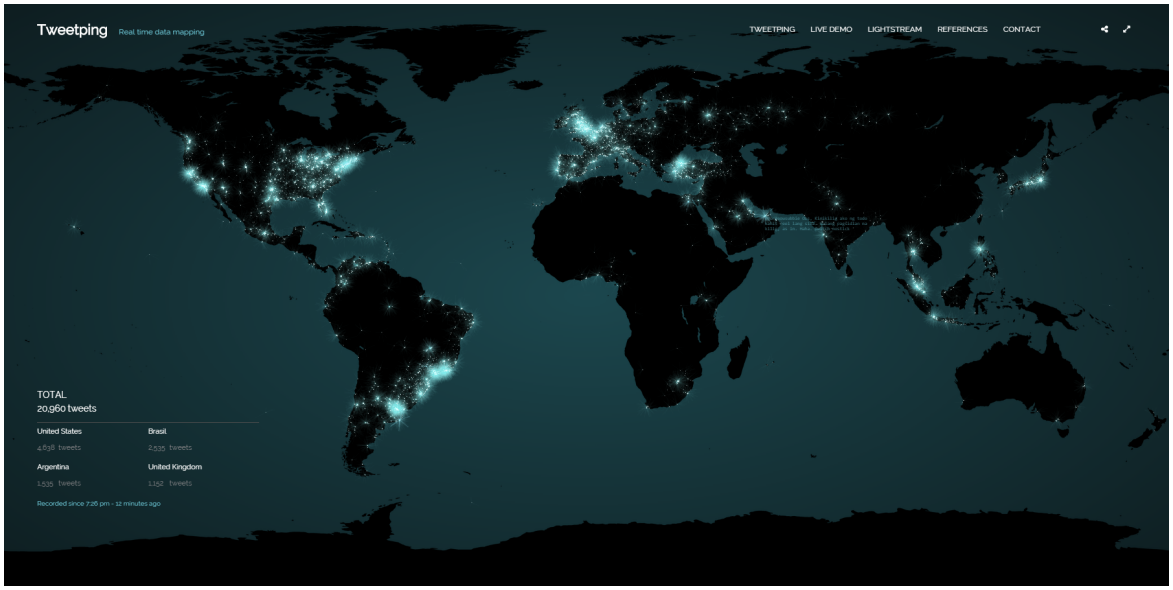

Abbildung 2.6.: Die Startseite von Tweetping mit aufblitzenden Lichtern für Tweets. Einzelne Texte werden zum Teil ebenfalls kurzzeitig angezeigt.

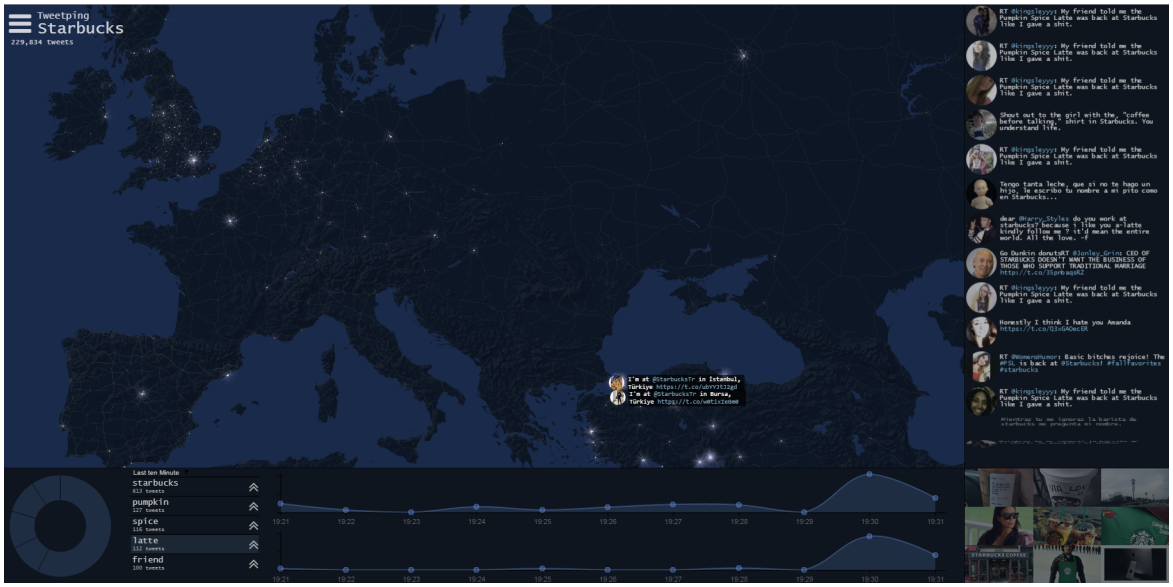

Abbildung 2.7.: Visualisierung von Tweetping für Starbucks.

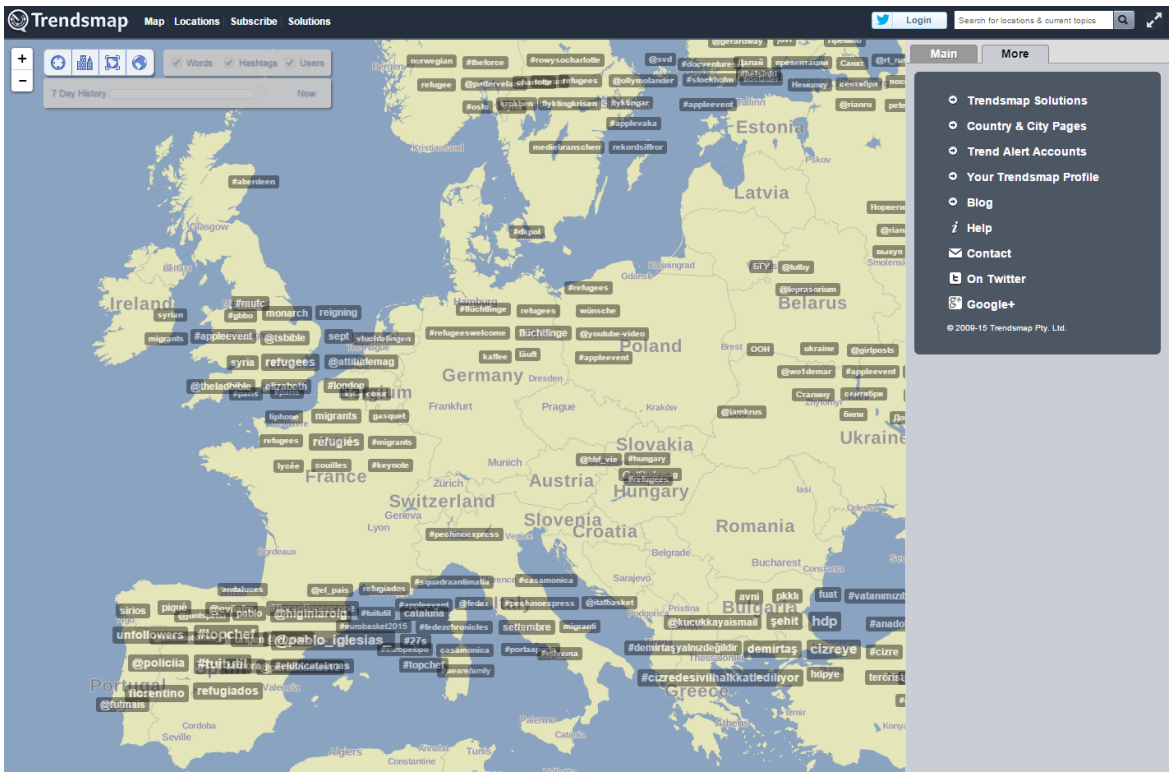

Abbildung 2.8.: Trends in Europa dargestellt auf Trendsmap am 10. September 2015.

und Links angezeigt werden. Die *Plus*-Version bietet noch wesentlich mehr Zoom auf der Karte, einen Rückblick auf die letzten sieben Tage und die Fähigkeit, nach Wörtern, Nutzern und Hashtags zu filtern.

Weitere Werkzeuge zur Visualisierung werden ebenfalls dort angeboten und bieten die Möglichkeit eigener Zusammenstellungen. Diese ermöglichen dann Zeitleisten, das Anzeigen von Tweets und die Option auf Animationen. (Solutions, 2015) Die Möglichkeiten entsprechen in etwa denen von Tweetping im Abschnitt 2.3.3.

2.3.5. followerwonk

Einen anderen Ansatz verfolgt die Plattform followerwonk[10], hier wird ausführlich auf einen einzelnen Nutzer eingegangen und nicht von einem Suchbegriff ausgegangen. Es können Nutzer verglichen, analysiert, die Follower verfolgt oder sortiert werden. Dazu muss ein Nutzername auf Twitter eingegeben werden. Bis zu einer bestimmten Anzahl an Ergebnissen sind die Funktionen frei und ohne Registrierung nutzbar.

In Abbildung 2.9 ist ein Ausschnitt aus der Analyse der Follower von `@RDataMining` zu sehen. Nach der Eingabe des Namens kann ausgewählt werden, ob die Follower des Nutzers oder die Nutzer, denen gefolgt wird analysiert werden sollen. Neben der gezeigten Karte und der Aktivitätsübersicht sind noch die Retweets, die Schlagwörter, das Geschlecht der Follower sowie die Sprachen der Follower und vieles mehr dargestellt. So erhält man einen Eindruck, wer sich für das Geschehen auf dem Twitter Profil interessiert und folgt.

Bei followerwonk gibt es noch weitere kostenpflichtige Funktionen. Es können Nutzer nach Beziehungen gefiltert, auf Graphen gesucht, sortiert oder überlagert und die Daten heruntergeladen werden. Dadurch ist eine umfassende Analyse der Follower möglich. Es wird damit geworben, ein gutes Marketing-Werkzeug zur Verfügung zu haben. (followerwonk, 2015)

[10] `https://followerwonk.com`

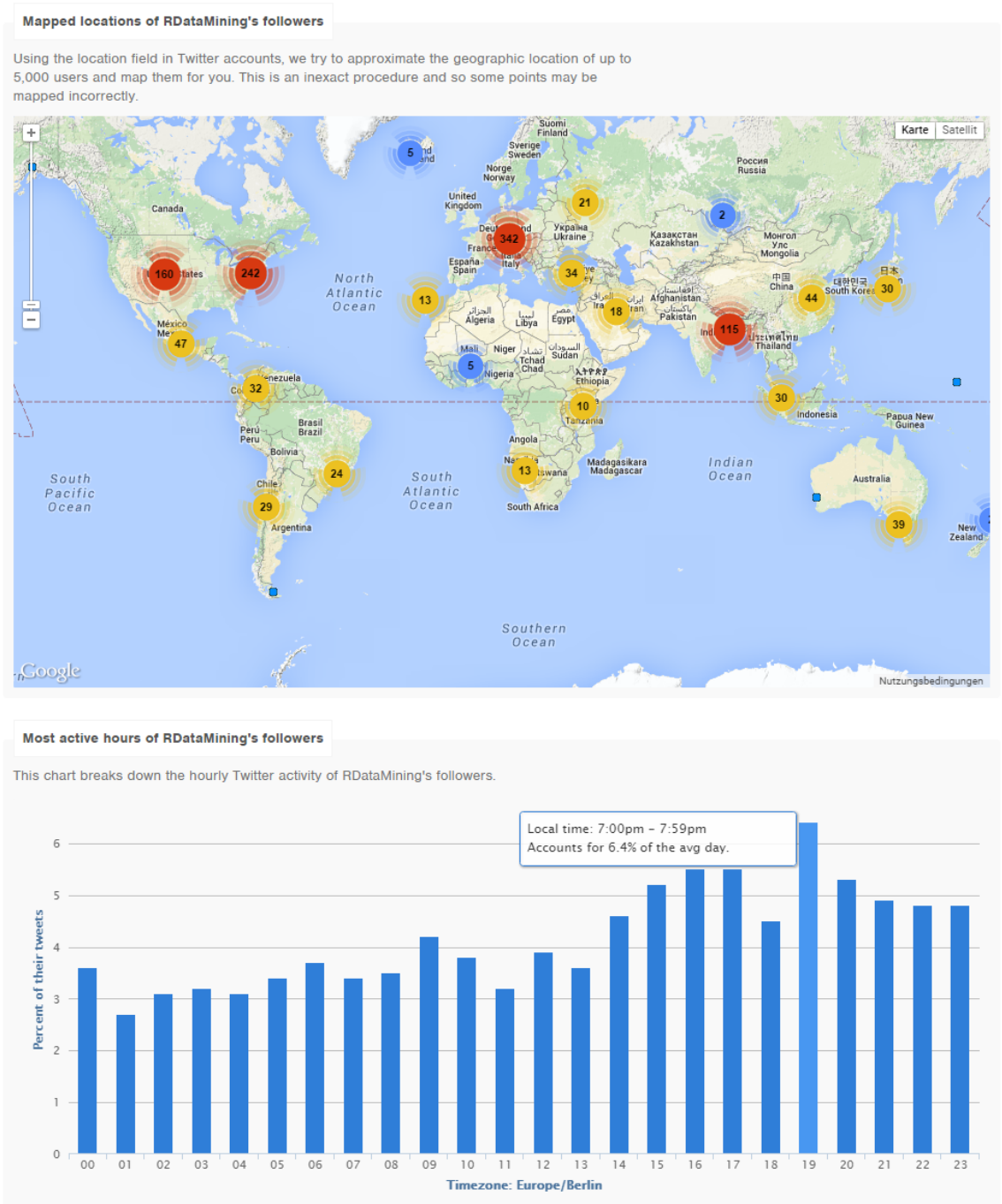

Abbildung 2.9.: Ein Teil der Auswertung der Follower auf `https://followerwonk.com/analyze/` für den Nutzer `@RDataMining` im sozialen Netzwerk Twitter.

2.3.6. Zusammenfassung zu den Beispielanwendungen

Insgesamt lässt sich also sagen, dass es schon einige ausgereifte Werkzeuge zur Analyse von Entwicklungen auf Basis von *Mircoblogging Content* gibt, wobei der Fokus auf Twitter als Datenquelle gelegt wurde. Andere Plattformen werden weniger genutzt, auf Grund stärkerer Restriktion der APIs. So ist es auch nicht weiter verwunderlich, dass Twitter gerne als soziales Netzwerk in der Wissenschaft genutzt wird.

Die Funktionen der hier vorgestellten Anwendungen sind meist vorgefertigt und eine Erweiterung nicht möglich. Unklar ist, was genau wie passiert und wie die Daten aufbereitet werden. Viele Prozesse finden in einer *Black Box* statt und bleiben dem Nutzer verborgen. Das schafft zum einen Unsicherheit und ist aus wissenschaftlicher Sicht nicht befriedigend.

Daher scheint Dokumentation und Forschung betreffs der Aufbereitung nötig, um diese Schritte fest zuhalten und zu verfeinern. Werkzeuge dafür sind bereits vorhanden, nur ist nicht immer klar wie sie angewandt werden müssen, mit welchen Parametern und in welcher Reihenfolge. Im folgenden Abschnitt sollen die bereits vorhanden Methoden vorgestellt werden.

2.4. Data Mining

Wie schon bei den Beispielanwendungen verdeutlicht, lassen sich aus großen Datenmengen Informationen gewinnen, die sich nicht immer auf den ersten Blick erschließen. Die Ansätze dazu werden unter dem Begriff des Data Mining zusammengefasst, welches als das Anwenden von Methoden und Algorithmen für eine möglichst automatische Extraktion von empirischen Zusammenhängen aus einer dafür aufgebauten Datenbasis beschrieben wird. (Lackes, 2009)

Das Data Mining ist der zentrale Schritt in einem umfassenden Datenanalyseprozess. Das gesamte Verfahren wird als *Knowledge Discovery in Databases (KDD)* bezeichnet. Dabei werden die Schritte nach Lackes und Siepermann, 2009 wie folgt unterteilt:

1. Problemabgrenzung
2. Definition der Data-Mining-Aufgabe
3. Datenvorverarbeitung
4. Codierung
5. Data Mining
6. Modellvalidierung
7. Decodierung
8. Filterung
9. Präsentation der Ergebnisse

Ausgehend von einer Fragestellung werden Aufgaben definiert, Daten gesammelt, Aufbereitet und für die Analyse die Informationen codiert. Anschließend folgt der eigentliche Prozess des Data Minings: das Finden einer Antwort. Diese wird überprüft, die Information wieder decodiert, gefiltert und abschließend das Ergebnis präsentiert.

Zum Bereich des Data Mining gehören noch einige Begriffe und Methode, die im folgenden Abschnitt kurz vorgestellt und erläutert werden. Diese sollen später zur Aufbereitung des *Microblogging Content* angewandt werden und den Rahmen zum Aufbau des Arbeitsablaufes beziehungsweise der Pipeline bilden.

2.4.1. Geovisual Analytics

Bei *Geovisual Analytics* handelt es sich um den Prozess des graphikbasierten, analytischen Schlussfolgerns. Dieser wird in einer Daten- und Informationerfassungsphase mit anschließender visueller Darstellung und dem Erkennen des Erwarteten oder Entdecken des Unbekannten eingeteilt. Auf Basis der gewonnenen Erkenntnisse kann die Auswertung abgebrochen oder fortgesetzt werden. Gegebenenfalls wird dann der Prozess erneut durchlaufen.

Als Erweiterung dazu bringt Luo und MacEachren, 2014 noch die soziale Komponente ins Spiel und bezieht sich auf das erste Gesetz der Geographie von Tobler:

„*Everything is related to everything else, but near things are more related than distant things.*“ (Tobler, 1970)

Darauf aufbauend wird die These formuliert, dass soziale Kontakte oft auch mit räumlicher Nähe zusammenhängen, welche sich auch in sozialen Netzwerken widerspiegeln und daher bei der Analyse berücksichtigt werden sollten. Auf Grund dessen wird der Begriff des *Geo-social visual analytics* eingeführt. Bestätigt wird angeführte These unter anderem durch Takhteyev, Gruzd und Wellman, 2012. Es leben 39 % der Follower eines Twitter-Nutzers weniger als 100 km entfernt. Zwar umspannt das gesamte Netzwerk die Erde, aber die Nutzer an sich sind meist nicht weltweit sondern nur regional vernetzt.

2.4.2. Text Mining

Das Text Mining wird als spezielle Form des Data Mining betrachtet, bei der Text als besondere Datenform analysiert wird. Texte sind im Gegensatz zur klassischen Variante des Data Mining nicht klar strukturiert und die Daten liegen nicht atomar in einzelnen Datenfeldern vor. Genauer betrachtet besitzen sie aber auch eine Struktur in Form von Überschriften und Grammatik, welche aber nicht für eine Maschine verständlich ist. Die Herausforderung liegt daher im Erschließen der Texte für eine maschinelle Analyse. (Hippner und Rentzmann, 2015)

Dabei werden verschiedene Techniken wie das *Natural Language Processing (NLP)* als computer-linguistische Methode, das maschinelle Lernen zur Klassifizierung, verschiedene statistische Methoden und natürlich auch Visualisierungen eingesetzt, um den gesamten Prozess zu unterstützen. Dieser läuft wieder in den gleichen Schrittfolgen ab, wie anfangs schon in Abschnitt 2.4 beschrieben.

2.4.3. Maschinelles Lernen

Die automatische Textklassifikation auf Basis von maschinellem Lernen ist heute weit verbreitet. Das wohl bekannteste Beispiel dafür stellt der Spam-Filter eines E-Mail-Postfaches da. Er entscheidet automatisch anhand von verschiedenen Merkmalen einer ankommenden E-Mail, ob dies eine unerwünscht übertragene Nachricht ist oder nicht. Ähnlich kann auch mit vielen anderen Texten verfahren werden, so dass diese automatisch in verschiedene Kategorien einsortiert werden können.

Bei einer überwachten Klassifikation wird zu nächst mit manuell ausgewählten Dokumenten trainiert, also überwacht gelernt, in dem daraus mit statistischen Verfahren Entscheidungskriterien abgeleitet werden. Dabei sind die zu erwartenden Kategorien bekannt. Im Gegensatz dazu gibt es noch das nicht überwachte Lernen, bei dem aufgrund von statistischen Zusammenhängen Gruppen gebildet werden.

2.4.4. Sentiment Analyse

Bei der Sentiment Analyse handelt sich um einen Teilaspekt des Text Mining, bei dem es um die automatische Erkennung einer positiven oder negativen Haltung gegenüber eines Objektes geht. Dies erfolgt wieder auf der Basis von statistischen Methoden in Verbindungen mit dem NLP und dem maschinellen Lernen.

2.4.5. Georeferenzierung von Text auf der Basis der Inhalte

Nicht alle Daten haben von Haus aus eine Koordinate, aber viele Texte haben einen Ortsbezug. Dieser wird oft hergestellt durch die Nennung von Ortsnamen im Text. Extrahiert man die Toponyme aus dem unstrukturierten Text und georeferenziert diese anschließend, erhält man eine Koordinate für das Dokument. Durch diesen Ansatz ist es möglich, noch eine Vielzahl an weiteren Textdaten zu lokalisieren und damit die Datenbasis für die Analyse zu vergrößern. Der Webservice *GeoTxt*[11] bietet genau dafür eine Schnittstelle an. Per Hypertext Transfer Protocol (HTTP) werden die Texte übertragen und die Ergebnisse als GeoJSON zurück gesendet. Intern wird *GeoNames* als Datenbank für geographische Namen verwendet und ein Ranking erstellt, falls mehrere Orte im Text erkannt werden. Auf der Webseite ist ebenfalls eine graphische Schnittstelle zugänglich, die zum Testen der API genutzt werden kann. (Karimzadeh u. a., 2013)

[11] `http://www.geotxt.org/`

3. Methoden

3.1. Software zur Informationsgewinnung

Im folgenden Abschnitt soll eine Auswahl an Softwareanwendungen vorgestellt werden, welche Werkzeuge zur Informationsgewinnung mittels Text Mining bereitstellen. Diese sollen anhand von Kriterien verglichen und auf die Verwendbarkeit für Daten mit Ortsbezug getestet werden. Für den Vergleich wurden die folgenden Kriterien formuliert:

- **Lizenz und Kosten:** Durch die Einsparung von Lizenzkosten fällt eine wesentliche Hürde für den weiteren und vor allem vielfältigen Einsatz weg. Die Software kann einfach getestet und weitergegeben werden.
- **Anbindung zu Twitter:** Eine direkte Einbindung der Twitter-API ermöglicht einen besseren Arbeitsablauf. Bei Bedarf können einfach noch Daten nachgeladen oder die Suchkriterien geändert werden. So kann spontaner geforscht werden und es muss nicht erst ein Datenbestand beschafft werden.
- **Datenhaltung (Dateiformate und Datenbanksysteme):** Auf Grund des zu erwartenden Datenvolumens und der Datenstruktur bietet sich die Verwendung von Datenbanken zur Datenspeicherung an. Diese können sehr flexibel, einheitlich und schnell mittels Structured Query Language (SQL) abgefragt werden und bieten sehr gute Performance.
- **Text Mining Tools:** Für die Analyse der Daten wurden bereits diverse Werkzeuge entwickelt, welche möglichst auch schon in der Programmumgebung für ein unkompliziertes Arbeiten vorhanden sein sollten.
- **Maschinelles Lernen:** Der Arbeitsschritt der Datenaufbereitung könnte perspektivisch vereinfacht werden, wenn auf der Basis vom maschinellen Lernen noch die Anzahl der manuellen Eingriffe verringert würde.
- **Zeitreihenanalyse:** Zur Beurteilung der Entwicklung in der Datensammlung kann die Zeitkomponente analysiert werden. So kann aus der Vergangenheit unter bestimmten Annahmen auf die Zukunft geschlossen werden.
- **Visualisierung:** Zum besseren Verständnis der Analyseergebnisse oder zum Zweck der Analyse sind verschiedene Darstellungen sehr hilfreich. So können Trends, Entwicklungen und Zusammenhänge in den Daten besser erkannt werden. Darauf aufbauend kann die Forschung vertieft und die Präsentation der Ergebnisse vereinfacht werden.
- **GIS-Funktionen:** Einfache Funktionen zur Darstellung der Daten auf Karten sollten vorhanden sein, ebenso die Möglichkeit einer räumlichen Auswahl.
- **Programmierkenntnisse erforderlich:** Das Arbeiten sollte intuitiv möglich sein. Wenn dafür erst Code geschrieben werden muss, ist dies hinderlich und erfordert mehr Zeit für die Einarbeitung oder stellt eine Hürde für den Einstieg dar.

- **Erweiterbarkeit:** Für die Anpassung an die speziellen Fragestellungen sollte es gegebenenfalls möglich, sein noch ergänzende Funktionalitäten mit einzubinden und in den Arbeitsablauf zu integrieren.

3.1.1. R

Die freie opensource Programmiersprache R[1] wurde für statistische Berechnungen und zum Erstellen von Grafiken entwickelt. Sie ist für Windows, MacOS und Linux verfügbar und sehr weit verbreitet im kommerziellen wie auch wissenschaftlichen Bereich. Der Funktionsumfang kann mit vielen frei verfügbaren Paketen erweitert werden, die es erlauben auf eine Vielzahl von unterschiedlichen Datenquellen zuzugreifen.

Als grafische Benutzeroberfläche und integrierte Entwicklungsumgebung wird sehr häufig das freie RStudio[2] verwendet. Dokumentation, Shell, Variablenübersicht und Editor können gleichzeitig betrachtet werden und schaffen so eine praktische Arbeitsumgebung. Die Erstellung von Dokumentationen mittels LaTeX ist ebenfalls möglich.

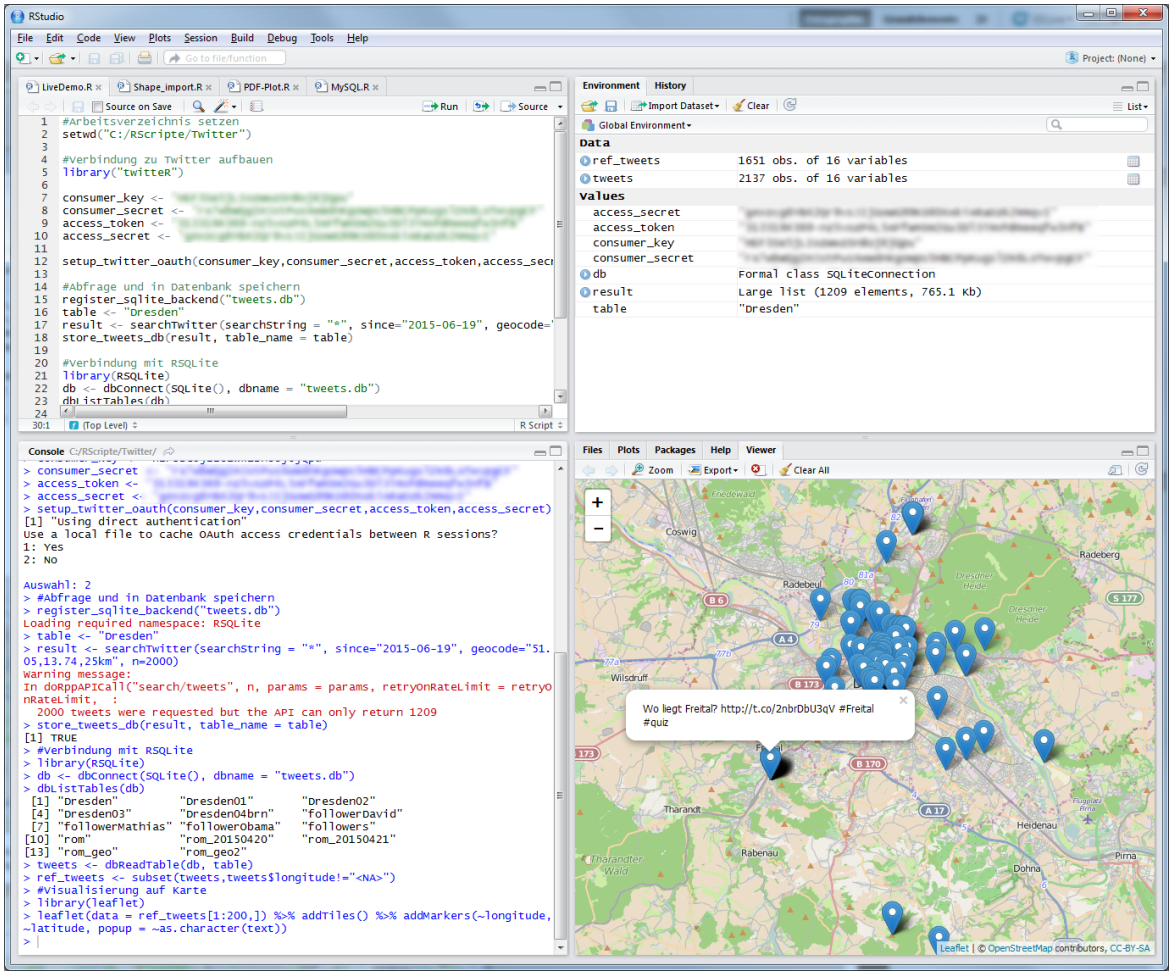

Abbildung 3.1.: Visualisierung von Tweets auf einer Karte, die zuvor über die Twitter-API abgefragt wurden im RStudio.

Das Paket `twitteR` ermöglicht es direkt aus R heraus Daten über die Twitter REST-API zu beziehen und weiter zu verarbeiten. Die Fähigkeiten zum Text Mining werden mit Paket `tm`

[1] `http://www.r-project.org/`
[2] `http://www.rstudio.com/`

realisiert. Des Weiteren existiert eine Vielzahl von kostenlosen Paketen für diverse Aufgaben. Diese können direkt aus dem RStudio heraus installiert werden.

Die Darstellung von Daten auf Karten ist interaktiv, wie auch statisch möglich. Ein Beispiel dafür ist in Abbildung 3.1 zu sehen. Ebenso kann problemlos mit diversen Geodatenformaten gearbeitet werden, wofür auch unterschiedliche Analysefunktionen zur Verfügung stehen. Nachteilig ist dabei die Arbeitsweise. Es wird nicht immer sofort sichtbar, was erledigt wurde und die einzelnen Befehle müssen erst nachgeschlagen werden. Dafür können Arbeitsabläufe aber mit einem Skript einfach automatisiert werden.

Die Erweiterung *Shiny*[3] bietet sich für wiederholte und interaktive Analysen an. Auf Basis von R-Skripten wird dabei eine Weboberfläche erstellt, in die die Ausgaben von Plots integriert und Parameter über Eingabefelder verändert werden können. Auf diese Weise können R-Funktionalitäten genutzt werden, ohne dass Kenntnisse in Programmierung nötig sind und die Analysetools als Service zur Verfügung gestellt werden.

3.1.2. Python

Python[4] ist eine universell einsetzbare freie Programmiersprache, welche meist nur interpretiert und nicht kompiliert wird. Sie wurde auf einfache Erlernbarkeit optimiert und fällt dadurch auf, dass der Programmcode nicht durch geschweifte Klammern, sondern durch Einrückungen strukturiert wird. Eine weitere Besonderheit ist, dass zwei Versionen schon längere Zeit nebeneinander existieren und genutzt werden. Oft wird noch der Zweig 2.7 verwendet, obwohl die Version 3.4 die aktuellste ist.

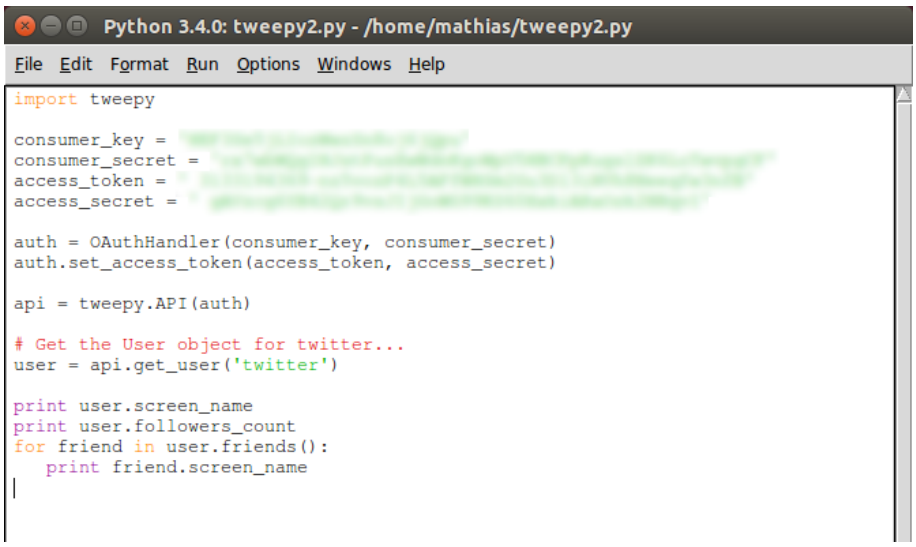

Abbildung 3.2.: Abfrage einer *Usertimeline* über die Twitter-API mit Python.

Für den vereinfachten Zugriff auf die API von Twitter stehen mehre Python-Module zur Verfügung. Wie in Abbildung 3.2 zu sehen ist, fiel die Wahl auf `tweepy`[5]. Dieses Modul kommt mit einer guten Dokumentation und Tutorials daher, wodurch der Einstieg sehr erleichtert wird. Unter Windows-Betriebssystemen kann die Installation von Modulen zum Teil problematisch sein und die Einarbeitung erschweren.

Die Arbeit mit Python kann interaktiv über eine Shell erfolgen, es können aber auch Skripte ausgeführt werden. So ist es möglich, Arbeitsgänge zu automatisieren ober bei Bedarf manuell mit angepassten Werten zu wiederholen. Erweiterungen zur Anbindung

[3] `http://shiny.rstudio.com/`
[4] `https://www.python.org/`
[5] `http://www.tweepy.org/`

an diverse Datenbanken sind genauso vorhanden, wie Module zum Text Mining und zur Sprachprozessierung. Visualisierungen lassen sich mit Python erstellen und werden als Datei abgespeichert und im Bedarfsfall neu erstellt.

Bedingung für die Nutzung bleiben aber Programmierkenntnisse. Im Bereich der Verarbeitung von und mit Geodaten ist Python weit verbreitet und bietet sehr gute Funktionalitäten. Auch ist es in bekannter GIS-Software wie ArcGIS und Qgis schon enthalten und könnte von dort aus gleich angesprochen werden.

3.1.3. KNIME

Hierbei handelt es sich um eine freie Software zur Datenanalyse auf der Basis von Java. KNIME[6] steht dabei für Konstanz Information Miner. Es handelt sich um ein Extraction-Transformation-Loading (ETL)-Programm, so dass sich einzelne Schritte von der Datenaufbereitung, Prozessierung bis hin zur Visualisierung beliebig aneinander reihen lassen. Es ist möglich, Python oder R einzubinden und für die Analyse zu nutzen. Für die einzelnen Anwendungsfälle existieren Erweiterungen, die entweder einzeln oder auch in eine Gesamtpaket installiert werden können.

Ein Modul zum Text Mining ist vorhanden und wird auch auf der Webseite erklärt. Allerdings befindet sich diese noch im Entwicklungsstadium. Eine Anbindung zur Twitter-API ist vorhanden. Allerdings sind nur einfache Suchanfragen nach Begriffen möglich – Koordinaten erhält man hier nicht. Ein Modul zur Visualisierung geografischer Koordinaten existiert. Ansonsten sind keine weiteren GIS-Funktionalitäten verfügbar. Eine Realisierung wäre dennoch über die Schnittstellen zu Python und R denkbar.

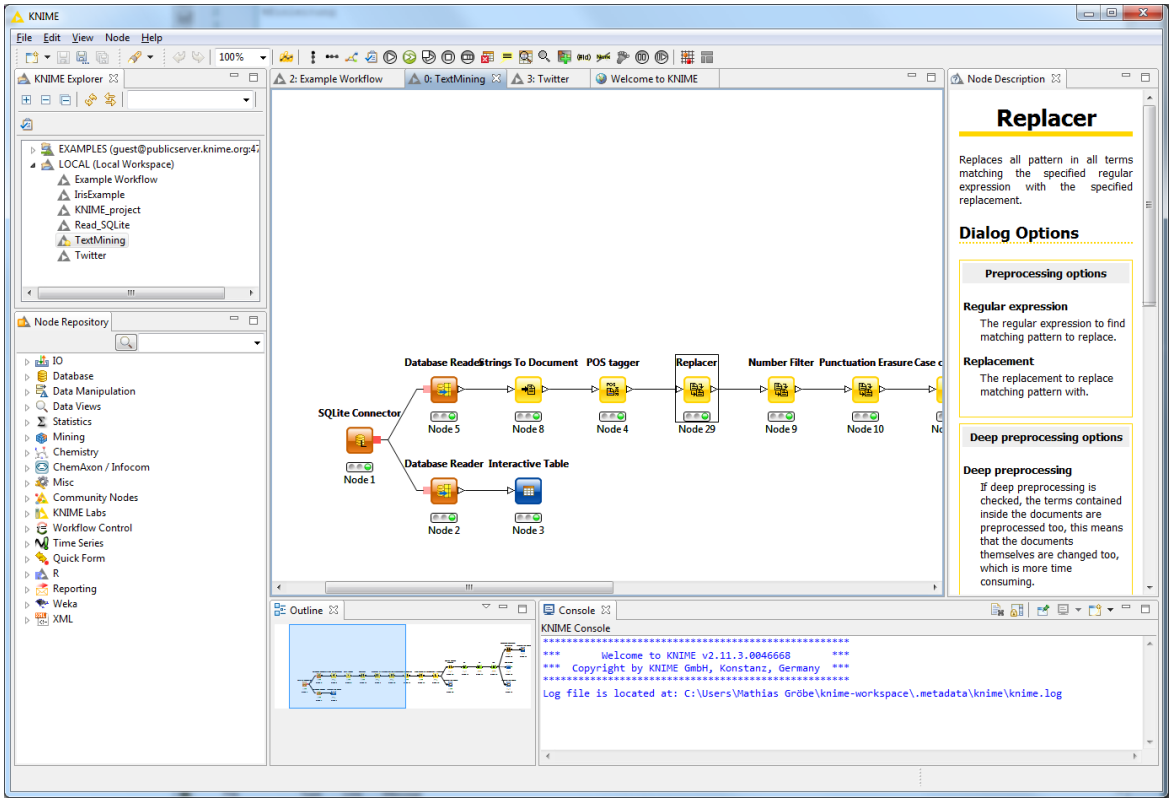

Abbildung 3.3.: Einblick in die Arbeitsumgebung von KNIME mit einem Modell für Text Mining.

[6] http://knime.org/

In Abbildung 3.3 ist ein Beispiel für Text Mining in KNIME zu sehen. Die verschiedenen Werkzeuge werden nacheinander auf eine Datenmenge angewandt. Verzweigungen, Bedingung und Schleifen können hier eingebaut werden. Die Arbeitsschritte werden dadurch visuell sichtbar. Im linken unteren Fenster sind die verschiedenen Werkzeuge „Nodes“ in Kategorien eingeordnet zu sehen. Von dort können sie auf die Arbeitsfläche gezogenen werden. Auf der rechten Seite ist eine Beschreibung der Werkzeuge verfügbar mit Hinweisen zu den Optionen.

Insgesamt ist die Nutzung intuitiv und nach einer kurzen Einarbeitung zu bewerkstelligen. Die Erweiterung der Funktionen ist prinzipiell möglich, wenn auch über den Umweg, andere Software einzubinden oder eigene Erweiterungen zu entwickeln.

3.1.4. RapidMiner

Die auf Java basierende kommerzielle Anwendung RapidMiner[7] wird laut der Webseite von bekannten Firmen wie Miele, ebay, CISCO, PayPal und VW eingesetzt. Sie ist in einer eingeschränkten Variante auch frei erhältlich. Es werden mehre Varianten der Software angeboten, die auf bestimmte Einsatzbereich ausgerichtet sind. Neben dem RapidMiner Studio gibt es noch unter anderem Versionen für Server und ein Cloud-Computing-Angebot. Im Folgenden soll die Variante Studio besprochen werden, welche als Testversion mit erweiterten Funktionen 30 Tage genutzt werden kann.

Zum Download der Software ist zunächst eine Registrierung erforderlich. Die Benutzeroberfläche ist mit KNIME vergleichbar, aber deutlich übersichtlicher und reagiert schneller auf Interaktionen. Im rapidminer Marketplace[8] befinden sich noch Erweiterungen, welche direkt aus der Software heraus installiert werden können. Über diese ist ebenfalls eine Anbindung zu Python und R möglich. Die Features für Text Mining sind in der Erweiterung *Text Processing* enthalten, welche erst noch nachinstalliert werden muss.

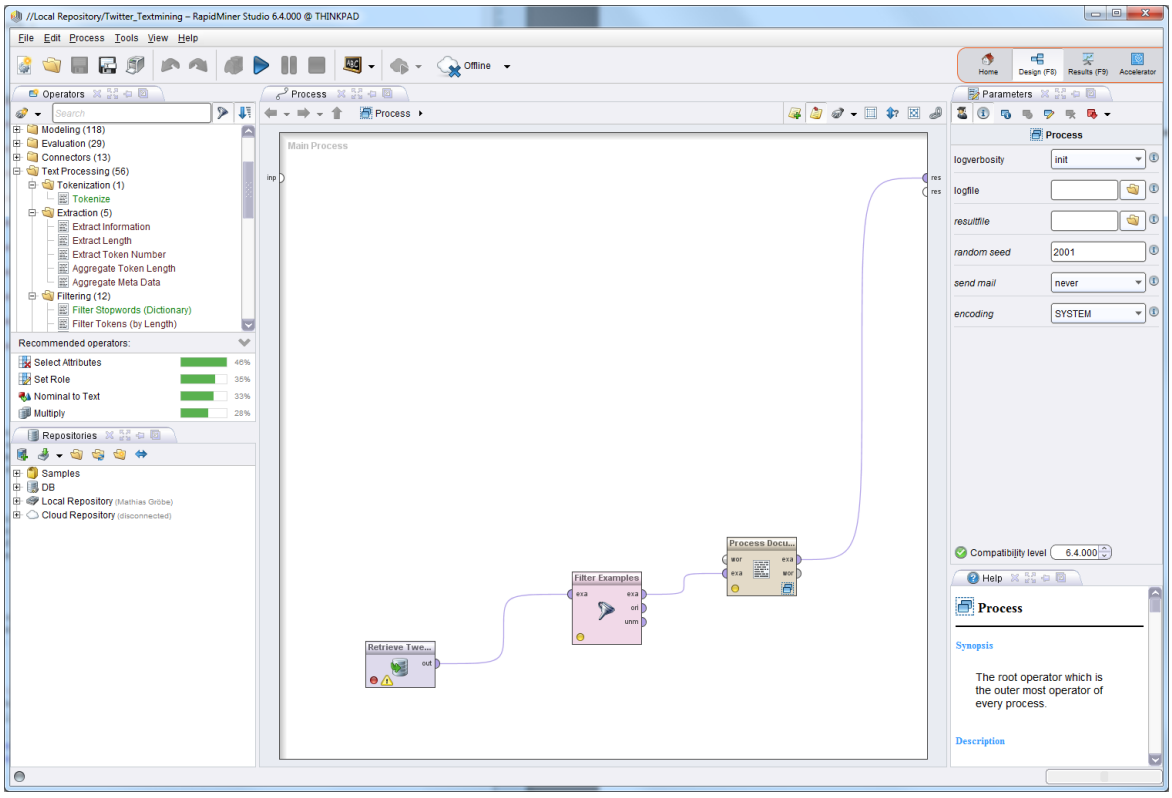

Abbildung 3.4.: Die Arbeitsoberfläche des RapidMiners.

[7] https://rapidminer.com/
[8] https://marketplace.rapid-i.com/UpdateServer/faces/index.xhtml

Neben dem Text Mining bietet der RapidMiner noch zahlreiche Möglichkeiten zur Sentiment Analyse, im Unterschied zu den anderen Tools. Hier könnte die Meinungsforschung noch weiter vertieft werden. In Abbildung 3.4 ist die Arbeitsoberfäche zu sehen. Diese ähnelt vom Aufbau her in der Design-Ansicht KNIME. Das angewandte Konzept ermöglicht die Trennung von Datenaufbereitung und Visualisierung, welche dann in der Result-Ansicht erfolgt.

Insgesamt macht die Software einen soliden Eindruck und ermöglicht einen praktischen Arbeitsablauf. Die Fähigkeiten zur Analyse sind sehr gut ausgebaut. Eine Schnittstelle zu Twitter wird zwar in der Dokumentation erwähnt, scheint aber momentan nicht verfügbar zu sein. GIS-Funktionalitäten sind nicht verfügbar, wären aber wieder über eine Anbindung zu R umsetzbar. Die Einarbeitung wird durch Tutorials erleichtert und stellt keine Hürde da. Es sind keine Programmierkenntnisse erforderlich.

3.1.5. Vergleich der Software

Zum besseren Vergleich der getesteten Software unterteilt man diese am besten in zwei Gruppen: Die mit einer grafischen Benutzeroberfläche, also KNIME und RapidMiner, welche keine Programmierkenntnisse erfordern und eine Gruppe, die direkt mit den Skriptsprachen arbeitet, wie R und Python. In Tabelle 3.1 ist eine Übersicht über die Software und die Erfüllung der aufgestellten Kriterien zu finden.

Python ist eine vielseitige Programmiersprache, die alle Anforderungen direkt mit Erweiterungen durch frei verfügbare Module erfüllt. R ist als erweiterbare Sprache für statistische Analysen entwickelt worden und kommt hier genau dafür zum Einsatz und kann dadurch voll seine Funktionalitäten ausspielen. Zurzeit ist die Sprache in starker Entwicklung begriffen und wird vielfältig eingesetzt.

Nachteilig an den Skriptsprachen sind ihre geringe Performance und eine nicht immer einheitliche Dokumentation. Sofern der User nicht um die Funktionen der Software weiß oder aktiv danach sucht, gibt es keinen roten Faden, der ihn leitet und Möglichkeiten aufzeigt. Hier hilft oft eine Benutzeroberfläche bei der Arbeit.

Fertige Software nach dem Baukastenprinzip, wie KNIME und RapidMiner als ETLs, bieten deutlich sichtbar Funktionen an, die nur noch aneinandergereiht werden müssen. Eine Dokumentation ist ebenfalls auf Abruf verfügbar. Die Einarbeitung ist einfacher und intuitiver. Bei den Funktionalitäten ist man aber erst einmal auf den Softwarehersteller angewiesen, auch wenn es prinzipiell immer möglich, ist noch eigene Erweiterungen zu entwickeln.

Tabelle 3.1.: Übersicht über die verglichene Software.

Kriterium	R	Python	KNIME	Rapidminer
kostenlose Software	✓	✓	✓	-
Verbindung zur Twitter API	✓	✓	✓	*
Datenbankanbindung	✓	✓	✓	✓
Text Mining Tools	✓	✓	✓	✓
Maschinelles Lernen	✓	✓	✓	✓
Zeitreihenanalyse	✓	✓	✓	✓
Visualisierungen	✓	✓	✓	✓
GIS-Funktionalitäten	✓	✓	✓*	✓*
Programmierkenntnisse	✓	✓	-	-
Erweiterbarkeit	✓	✓	✓	✓

Python bietet sehr gute Werkzeuge, um mit Twitter-Daten zu arbeiten, die einfach nur installiert werden müssen. Nachteilig ist dagegen, dass mehr Programmieraufwand erforderlich ist für die Auswertung. Zwar kann genauso in einem interaktiven Modus gearbeitet werden wie bei R, allerdings sind bei komplexeren Datentypen wesentlich mehr Arbeitsschritte nötig als in R. Dort sind Funktionen schon abstrahiert und mehr auf die Analyse ausgerichtet. Zur Datengewinnung bietet sich Python aber durchaus an. Die Option damit auch richtige Programme oder Webanwendungen damit zu erstellen, hat viel Potenzial, sowie auch die Möglichkeit einer Einbindung in ein GIS.

R bietet mit der Erweiterung `Shiny` eine sehr ansprechende Lösung, einfach und schnell eine Benutzeroberfläche zu schaffen, welche auch als Webservice zur Nutzung bereitgestellt werden kann. Mit Python lässt sich zwar ebenfalls etwas Vergleichbares schaffen, nur mit wesentlich mehr Aufwand. Die Installation von zusätzlichen Paketen erfolgt direkt aus R heraus. Die integrierte Entwicklungsumgebung RStudio vereinfacht die Entwicklung noch einmal merklich. Dazu bietet R die Möglichkeit direkt Visualisierungen zu erstellen, sei es als Karte, Diagramm oder Graph, und einen Einblick in die Daten, welche in der Regel als Tabellen *Dataframe* gespeichert werden, ist sehr einfach möglich.

Aufgrund von positiven Erfahrungen und der einfachen Erweiterbarkeit ist R das Mittel der Wahl für die Datenaufbereitung geworden. Es ist kostenlos, sehr einfach erweiterbar und funktioniert auf allen gängigen Plattformen. Zwar bieten Werkzeuge wie KNIME und Rapidminer auch sehr gute Möglichkeiten. Die Einarbeitung in diese Werkzeuge ist zwar wesentlich einfacher, sie sind aber nicht so gut erweiterbar und greifen letztlich auch teilweise auf Funktionalitäten von R und Python zurück. So ist es wesentlich stringenter, direkt mit diesen Tools zu arbeiten, anstatt sie aus dem ETL heraus anzusprechen.

3.2. Aufbereitung von Daten

Auf Basis der Arbeiten von Nicola, 2013, Hahmann, 2014 und Y. Zhao, 2013, S. 105-122, wurden unter anderem die folgende Probleme und Lösungsansätze zur Aufbereitung des *Micoblogging Content* zusammengetragen und sollen hier zu einem Arbeitsablauf, auch „Pipeline“ genannt, zusammengeführt werden.

3.2.1. Probleme

Um die Daten analysieren zu können, sind verschiedene Probleme zu lösen. Zum einen muss die Sprache des Textes festgestellt werden, um ihn Aufbereiten zu können. Dazu sind bereits verschiedene Algorithmen definiert, die Sprachen detektieren können. Eine Vermischung der Sprachen wäre ungünstig, da sich Begriffe gleichen Wortlauts mit unterschiedlicher Bedeutung überlagern könnten und so zu einem falschen Ergebnis führen würden. Beispielsweise meint man im Deutschen mit *Handy* ein Mobiltelefon. Im Englischen ist *handy* ein Adjektiv, dass sich am besten mit „praktisch“ übersetzen lässt. Auf Grund dessen sollte also die Sprache bei der Analyse von *Microblogging Content* berücksichtigt und in der Pipeline bei der weiteren Aufbereitung jede Sprache am besten für sich behandelt werden. Was sich auch allein dadurch begründen lässt, dass der Bearbeiter die Sprache verstehen sollte, mit der er arbeitet, und Sprachen unter anderem es ermöglichen, bestimmte Gruppen zu bilden. So können zum Teil Touristen anhand ihrer Sprache identifiziert werden.

Aus den Sprachen ergibt sich noch ein weiteres Problem – Orte haben schließlich in verschiedenen Sprachen unterschiedliche Namen. So heißt *Rom* im Italienischen *Roma* und im Englischen *Rome*. Fachsprachlich handelt es sich dabei um Toponyme. Wann welcher Name verwendet wird, ist nicht unbedingt von der Sprache des Textes abhängig, sondern oftmals von persönlichen Gewohnheiten. Daher ist es nötig, wenn Ortsnamen von Interesse sind, die verschiedenen Namen auf eine Variante zurückgeführt werden.

Des Weiteren gibt es noch Tippfehler in den Nachrichten, oder Wörter sind schlicht falsch geschrieben. Da es sich dabei nicht um systematische Fehler handelt, können diese nicht korrigiert werden. Des Weiteren sind nicht immer alle Zeichen bei der Erstellung der Texte verfügbar. So kann unter den beschrieben Umständen aus dem „Großen römischen Kaiser“ der „grosse roemische Kaiser“ werden. Als Lösung des Problems mit den Umlauten könnten diese in ihre Umschreibungen umgewandelt werden.

Ein weiteres Problem sind sogenannte *Bots*. Dabei handelt es sich um Nutzer, die automatisiert Nachrichten posten. Ein praktisches Beispiel hierfür wären Wetterdaten. Hier ändern sich nur die Werte, ansonsten bleibt der Inhalt der Nachrichten immer gleich. Ähnlich verhält es sich mit Infostatusmeldungen. Die Meldungen sind meist nicht automatisiert, behandeln aber stets ein Thema. So kann es das lokale Verkehrsunternehmen sein, dass über Umleitungen von Bussen und Straßenbahnen informiert, oder auch der Radiosender, der auf Staus und Unfälle hinweist. Sofern man sich nicht genau dafür interessiert, sollten die Nachrichten dieser Nutzer aussortiert werden. Ansonsten überlagern diese Nachrichten auf Grund ihrer großen Menge alle weiteren Konversationen.

3.2.2. Abfolge der Aufbereitungsschritte

Um die Daten für das eigentliche Data Mining aufzubereiten sind die oben genannten Probleme zu lösen. Dazu wird hier ein Arbeitsablauf konstruiert, der unter anderem auf die Methode *Geovisual Analytics* zurückgreift, siehe dazu Abschnitt 2.4.1. Dadurch kann schnell und einfach eine Verfeinerung des Ablaufs vorgenommen werden. Der gesamte Ablauf der Aufbereitung ist in Abbildung 3.5 visualisiert und wird im Folgenden Beschrieben. Die schon hier genannten Visualisierungen werden im folgenden Abschnitt 3.3 genauer mit Beispielen beschrieben. Ausgangspunkt für die Aufbereitung sind georeferenzierte Kurztexte mit Metadaten in strukturiert Form, mit bekannter Sprache, wie sie in Tabelle 3.2 exemplarisch zu sehen sind.

Zunächst wird dazu erst einmal auf Basis der Fragestellung eine Eingrenzung vorgenommen, die eine Auswahl aus den Daten oder dem Datenstrom darstellt. Dabei werden zeitliche, räumliche und thematische Kriterien berücksichtigt. Genauso kann aber auch auf die Daten eines Nutzers zugegriffen werden und nach den genannten Kriterien ausgewählt werden.

Auf der geschaffenen Datenbasis werden immer wieder Visualisierungen erstellt, um die Daten zu entdecken und zu erforschen. Dafür bieten sich Tabellen, Karten, Histogramme und Wortwolken an. Anhand des daraus gewonnenen Wissens kann der Aufbereitungsprozess verbessert werden. So kann unter anderem das Entfernen von Duplikaten sinnvoll sein. Aber auch Nutzer, die unerwünschte Statusmeldungen beitragen, bedürfen einer Filterung. Das Ergebnis kann dann immer wieder anhand der Visualisierungen überprüft werden.

Anschließend soll die Auswahl einer Sprache erfolgen, was voraussetzt, dass zu den Texten schon die Sprache bekannt ist. Damit erhält man eine homogenere Datenmenge, aus der dann URLs, Zahlen, Satzzeichen, Interpunktionszeichen und unerwünschte Zeichen entfernt werden können und abschließend alle Buchstaben in Kleinbuchstaben umgewandelt werden. Dadurch sind die Wörter besser miteinander vergleichbar. Auch das angesprochene Problem mit den Umlauten kann in diesem Schritt gelöst werden.

Das darauffolgende Entfernen der Stoppwörter und das Stemming ist spezifisch für jede Sprache. Stoppwörter sind Wörter, die vor allem grammatikalische oder syntaktische Funktionen übernehmen und keinen Informationen transportieren. Durch ihr häufiges Vorkommen überlagern sie ansonsten die relevanten Wörter. Es existieren bereits Listen mit Stoppwörtern für viele Sprachen, allerdings müssen diese noch auf die jeweilige Fragestellung erweitert werden, was hier wieder durch die Visualisierungen frequenzbasiert geschehen sollte. Als Stemming wird das Zurückführen eines Wortes auf einen Wortstamm bezeichnet. Dies geschieht mit einem Algorithmus. Alternativ kann eine Lemmatisierung erfolgen, wobei dies auf Basis eines Wörterbuches erfolgt. Abschließend erhält man aufbereiteten *Microblogging Content* für eine weitergehende Analyse.

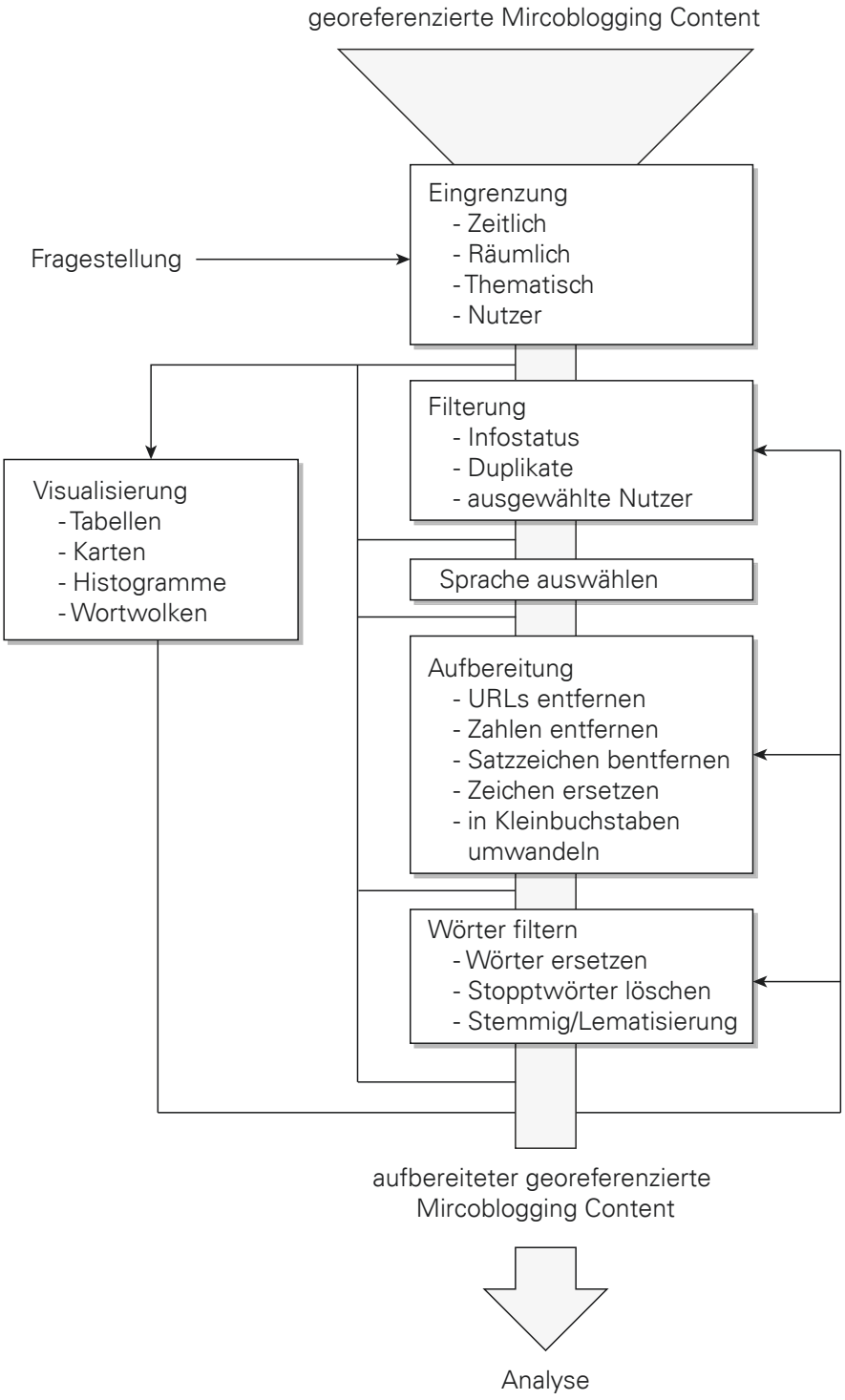

Abbildung 3.5.: Ablauf der Aufbereitung.

3.3. Visualisierung

Im folgenden Abschnitt sollen die Möglichkeiten zur Visualisierung von Ergebnissen im Text Mining Prozess aufgezeigt werden. Die verschiedenen Darstellungen können einen Einblick in die Daten geben, die Aufbereitung erleichtern und abschließend auch der Analyse dienen. Die jeweiligen Methoden haben ihre Vor- und Nachteile, welche kurz beschrieben werden, wobei auch auf die Besonderheiten der jeweiligen Visualisierungen eingegangen werden soll.

Als Beispieldatensatz für die Visualisierungen dienen Tweets mit exakter Koordinate aus der Umgebung von Rom vom 1. bis 30. April 2015. Die Daten wurden vom Institut für Kartographie der TU Dresden erfasst und wöchentlich als Shapefile abgelegt. Dabei wird die ID eines Tweets, eine eventuelle Retweet-ID oder Reply-ID, sowie der Nutzername, der Anzeigenamen, die Sprache des Nutzers, seine Zeitzone, seine Ortsangabe, die Quelle des Tweets, die Hashtags, die Koordinaten im Koordinatensystem World Geodetic System 1984 (WGS84) und der Zeitstempel gespeichert. Die Sprache der Kurznachrichten wurde mit dem `n-gram` Algorithmus der Apache Tike Bibliothek detektiert. Zu beachten ist aber, dass dieser bei sehr kurzen Texten fehleranfällig ist. (Insitut für Kartographie TU Dresden, 2015)

Für die Auswertung wurden die entsprechenden Wochen in eine MySQL Datenbank importiert und von doppelten Einträgen bereinigt. Im Anschluss erfolgte die zeitliche und räumliche Auswahl. Als Basis für den Ausschnitt wurde das umgebende Rechteck um die Region Latinum in Italien, in der Rom liegt, verwendet. So können auch wichtige Verkehrsknoten, wie die Flughäfen mit erfasst werden.

Die Daten wurden dann in R weiter ausgewertet und damit die folgenden Grafiken erstellt. Dabei wurde im wesentlichen das Paket `ggplot2` verwendet. Als Vorlage dienten dabei die Folien zu einem Vortrag von Yanchang Zhao, der dabei ausführlich mit Beispielen und Code die Möglichkeiten der Analyse von Twitter Daten vorstellt. (Y. Zhao, 2015) Die Kartendarstellungen basieren auf einem Tutorial von Ross, 2014.

3.3.1. Tabelle

Für den ersten Einblick in die Daten eignet sich oft eine Tabelle. Dabei handelt es sich um eine geordnete Zusammenstellung von Texten und Daten. Die Inhalte werden dabei in Zeilen und Spalten gegliedert. Die erste Zeile heißt Kopfzeile und beschreibt die folgenden Werte. Im Falle der Spalten spricht man von einer Vorspalte. Im wissenschaftlichen Umfeld steht in der Regel eine Bobachtung in einer Zeile, während die Spalten die Eigenschaften darstellen. Die Tabelle sollte eine Über- oder eine Unterschrift haben, welche den Inhalt erklärt. Eine logische Sortierung erleichtert dabei noch die Wahrnehmung der Inhalte.

Zwar kann mit einer Tabelle nicht der gesamte Datenbestand überblickt werden, aber die Struktur der Daten wird deutlich und liefert Ansätze für weitergehende Analysen. Tabelle 3.2 soll hier als Beispiel dafür stehen. Eine interaktive Tabelle mit Sortierfunktionalitäten ermöglicht es noch, Extremwerte zu finden und zu betrachten. Schon durch „scharfes Hinsehen“ kann hier zum Beispiel ein Nutzer als Bot identifiziert werden.

Basierend auf dem Wissen um die verschiedenen Sprachen in den Tweets, lässt sich eine Zusammenfassung erstellen. So wurden die Tweets mit einem Sprachcode gezählt und anschließend die relative Häufigkeit errechnet. Sortiert man die Ergebnisse, kann ein klarer Trend wie in Tabelle 3.3 erkennbar werden. Dieses Wissen kann anschließend in einem Diagramm für eine visuelle Analyse dargestellt werden. Eine sofortige Visualisierung ist zwar möglich, allerdings wäre etwa die Hälfte der Datensätze nicht wirklich wahrnehmbar. Hier sollte eine Vorauswahl getroffen werden, um der Grafik mehr Aussagekraft zu geben.

Tabelle 3.2.: Einblick in den Beispieldatensatz, es wird hier nur eine Auswahl der Felder wiedergegeben.

TWEETID	TWEET	USERNAME	LANGUAGE	LATITUDE	LONGITUDE	TIME
260328	@ale_dibattista vince facile con la domande ridicole di floris #dimartedi	manuela	it	41.86	12.64	2015-04-01 00:00:04
260344	Pressione 1012,2 hPa, Diminuzione lenta. Temp 11,1 C. Pioggia 0,0 mm. Umidità 85% UV 0 http://t.co/POvaqDyZpb	Stazione Meteo	it	41.86	13.03	2015-04-01 00:00:08
260356	@orkalorka @ftk33333 a parte che a breve si accasa, ma a me non dice niente. Anche perché non lo conosco...	Cristiana N	it	41.86	12.55	2015-04-01 00:00:11
260355	@donnavventura la 1ª menzione del 'Capo Verde' appare nel tuo TL. Adesso è Tendenza in Italy! #trndnl	Trendinalia Italia	it	41.90	12.50	2015-04-01 00:00:11
260364	6. Tea Falco 7. Joe Jonas 8. Jace 9. Wells 10. #portaaporta 2015/3/31 23:53 CEST #trndnl2015 http://t.co/1z2nx2qb8e	Trendinalia Italia	et	41.90	12.50	2015-04-01 00:00:13
260367	1. #zayninpuglia 2. #dimartedi 3. #ItaliaInghilterra 4. #madeinsud 5. #Ballarò 2015/3/31 23:53 CEST #trndnl2015 http://t.co/1z2nx2qb8e	Trendinalia Italia	lt	41.90	12.50	2015-04-01 00:00:13

Tabelle 3.3.: Auflistung aller Sprachen und ihrer Häufigkeit im Beispieldatensatz.

Prozent	Häufigkeit	Sprach Code	Name
56.40	129444	it	Italienisch
10.14	23268	lt	Litauisch
6.74	15480	en	Englisch
4.07	9343	et	Estnisch
3.71	8517	gl	Galicisch
3.05	6995	sk	Slowakisch
3.02	6929	no	Norwegisch
2.88	6615	ro	Rumänisch
1.41	3234	eo	Esperanto
1.39	3197	sl	Slowenisch
1.36	3121	hu	Ungarisch
1.22	2790	pl	Polnisch
1.03	2359	fi	Finnisch
0.60	1377	pt	Portugiesisch
0.56	1296	ca	Katalanisch
0.55	1252	es	Spanisch
0.53	1222	is	Isländisch
0.53	1208	fr	Französisch
0.21	477	nl	Niederländisch
0.12	271	sv	Schwedisch
0.11	243	ru	Russisch
0.10	237	uk	Ukrainisch
0.09	214	de	Deutsch
0.06	140	be	Weißrussisch
0.05	117	da	Dänisch
0.05	110	th	Thailändisch
0.02	49	el	Neugriechisch

3.3.2. Histogramm

Für die Darstellung von Verteilungen bietet sich das Histogramm an. Dabei handelt es sich um die graphische Darstellung der Häufigkeitsverteilung bezogenen auf ein quantitatives Merkmal. Basierend darauf können Klassen gebildet werden. Häufig wird für die Darstellung ein Balkendiagramm verwendet, bei dem auf der Abszissenachse das Merkmal und auf der Ordinatenachse die Häufigkeit abgebildet wird. Neben der absoluten Häufigkeit kann genauso aber auch die relative Häufigkeit dargestellt werden, was einen besseren Überblick über die gesamte Verteilung ermöglicht.

Die Abbildung 3.6 zeigt ein Histogramm für die verschiedenen detektierten Sprachen aus Tabelle 3.3. Die Häufigkeit der einzelnen Sprachen ist leicht zu erschließen und zu vergleichen, wobei allerdings die Sprachen in den Hintergrund rücken gegenüber dem darstellenden Balken. In Verbindung mit der aufsteigenden Sortierungen ließen sich hier auch auf einfache Weise Klassen bilden. Nachteilig ist, dass es sich nicht um eine vollständige Wiedergabe der Werte aus Tabelle 3.3 handelt. Es wurde ein Schwellwert bei 1 % gesetzt, um die Übersichtlichkeit zu wahren. Dieses Vorgehen erhöht die Lesbarkeit und sollte, wann immer nötig, angewandt werden. Bei der Abbildung muss dann aber unbedingt darauf hingewiesen werden.

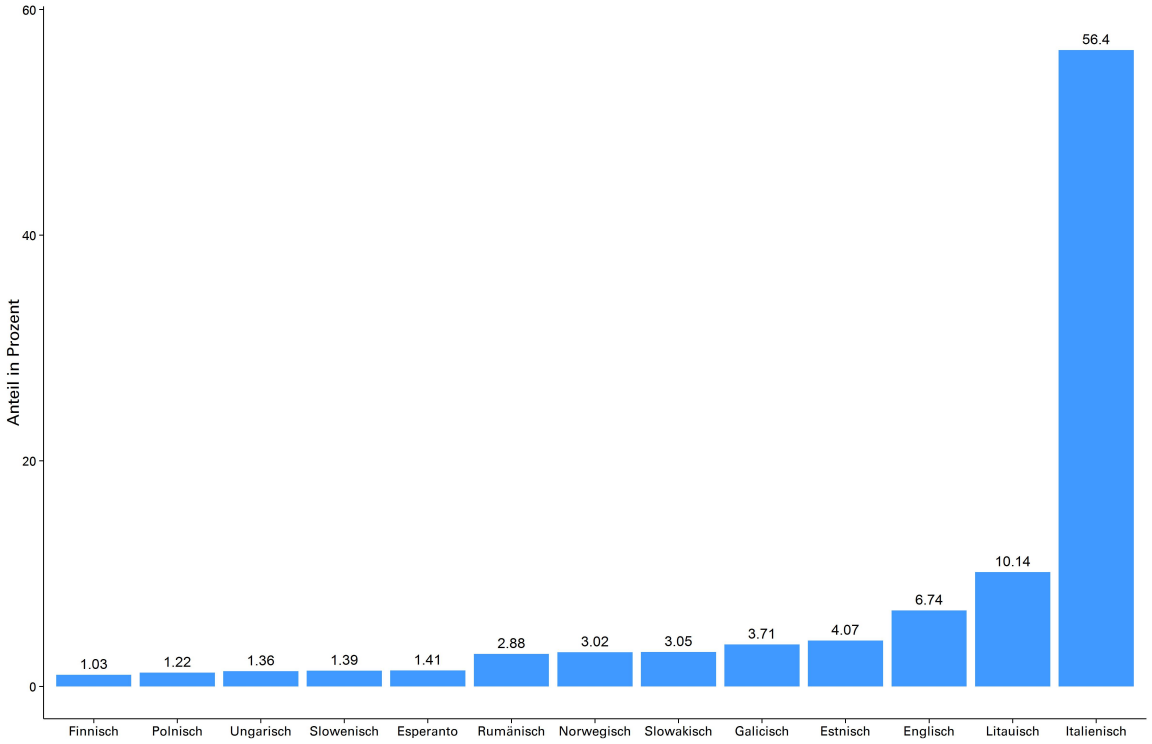

Abbildung 3.6.: Anteile der Sprachen größer als 1 %, dargestellt als Histogramm.

3.3.3. Wortwolke

Die Wortwolke, englisch *word cloud* oder *tag cloud*, ist ein sehr beliebtes Mittel zur visuellen Analyse von Texten. Sie lässt sich schnell und einfach aus Texten mit Hilfe von Programmen oder Webdiensten erzeugen. Beispiele dafür sind Wordle[9], Tagul[10] und Tagxedo[11]. Diese Dienste ermöglichen es, Wortwolken aus eingefügten Texten, URLs, Feeds oder Suchergebnissen zu erstellen. Dass sich damit auch eine anspruchsvolle Textanalyse umsetzten lässt, zeigt Heimerl u. a., 2014 mit dem *Word Cloud Explorer*.

Nach Rivadeneira u. a., 2007 können Wortwolken dazu genutzt werden, bestimmte Begriffe oder alternative Begriffe zu finden. Sie ermöglichen das Erkunden von Inhalten und vermitteln für diese einen bildhaften Eindruck und Überblick. Auch lässt sich durch eine Wortwolke der thematische Inhalt eines Textes erschließen.

Mit der Wortwolke wird die Häufigkeit eines Wortes dokumentiert. Auf den Inhalt bezogen ist es eine Darstellung der Popularität. Das Aussehen der Wortwolke wird von der Schriftgröße, der Schriftstärke, der Farbe, der Intensität, der Anzahl der Pixel, der Wortlänge, der Anzahl der Buchstaben, der Verteilung sowie der Anordnung der Wörter bestimmt. Wichtig für die Wahrnehmung ist die Schriftgröße und -stärke sowie die Intensität. Mit Vorsicht sollten Farben eingesetzt werden, da bestimmte Farben gegebenenfalls mehr die Aufmerksamkeit des Nutzers auf sich ziehen als andere. Ähnlich verhält es sich mit der Position von Wörtern. (Bateman, Gutwin und Nacenta, 2008)

Abbildung 3.7 zeigt die 250 häufigsten Wörter, extrahiert aus den als englischsprachig erkannten Tweets des Beispieldatensatzes. Zunächst wurden alle Buchstaben in Kleinbuchstaben umgewandelt, URLs, Zahlen, Satzzeichen und englische Stoppwörter entfernt. Anschließend wurde die Worthäufigkeit berechnet und basierend auf dieser die Wortwolke erstellt.

[9] `http://www.wordle.net/`
[10] `https://tagul.com/`
[11] `http://www.tagxedo.com/`

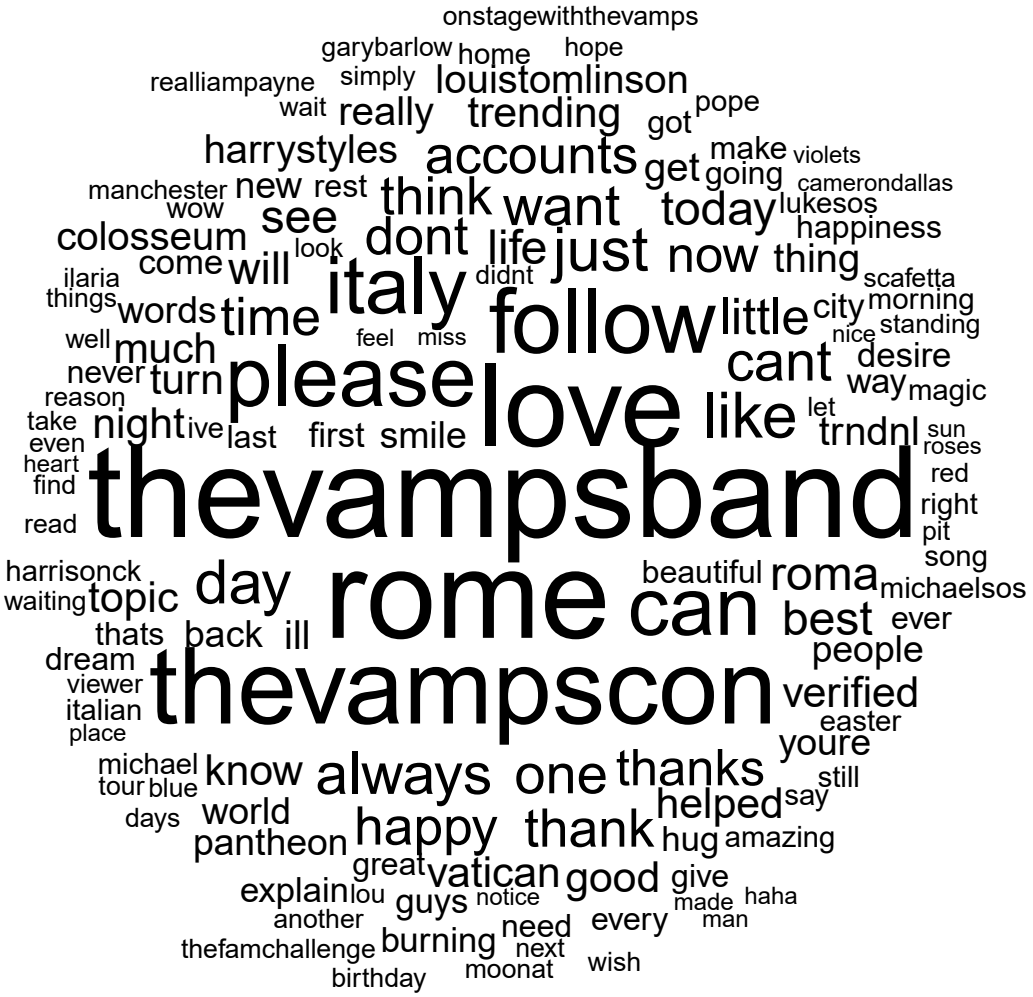

Abbildung 3.7.: Darstellung der 250 häufigsten Begriffe als Wortwolke.

Die Wortwolke ermöglicht einen sehr guten Überblick über die im Texte verwendeten Wörter. Das Verstehen ist intuitiv möglich und wichtige Begriffe sind auf einen Blick sichtbar. So können mehr Wörter auf geringeren Raum, visualisiert werden als in einem Histogramm oder in einer Tabelle. Nachteilig ist, dass kein Wert abgelesen werden kann, wie häufig ein Wort verwendet wurde. So kann es zu falschen Schlussfolgerungen kommen. Eine mögliche Erweiterung schlägt Lee u. a., 2010 mit *SparkClouds* vor, bei denen die Wörter noch um ein Diagramm zum Vorkommen in einem bestimmten Zeitabschnitt ergänzt werden.

3.3.4. Graphen

Ein Graph ist eine abstrakte Struktur, die eine Menge von Objekten beziehungsweise „Knoten“ mit Verbindungen, auch „Kanten“ genannt, untereinander darstellt. Graphen können gerichtet oder ungerichtet sein. Ebenfalls können Kanten noch gewichtet und auch mit anderen Bedingung verknüpft werden.

Ein bekanntes Beispiel ist der Netzplan einer U-Bahn. Hier sind die Bahnhöfe die Knoten und die Strecken die Kanten. Neben der Information der einzelnen Verbindungen ist auch die relative geographische Lage zueinander mit dargestellt. Genauso werden Attribute, wie die Linie, der Name der Bahnhöfe und deren Merkmale abgebildet. Hierbei wird eine größere Menge an Informationen überschaubar visualisiert.

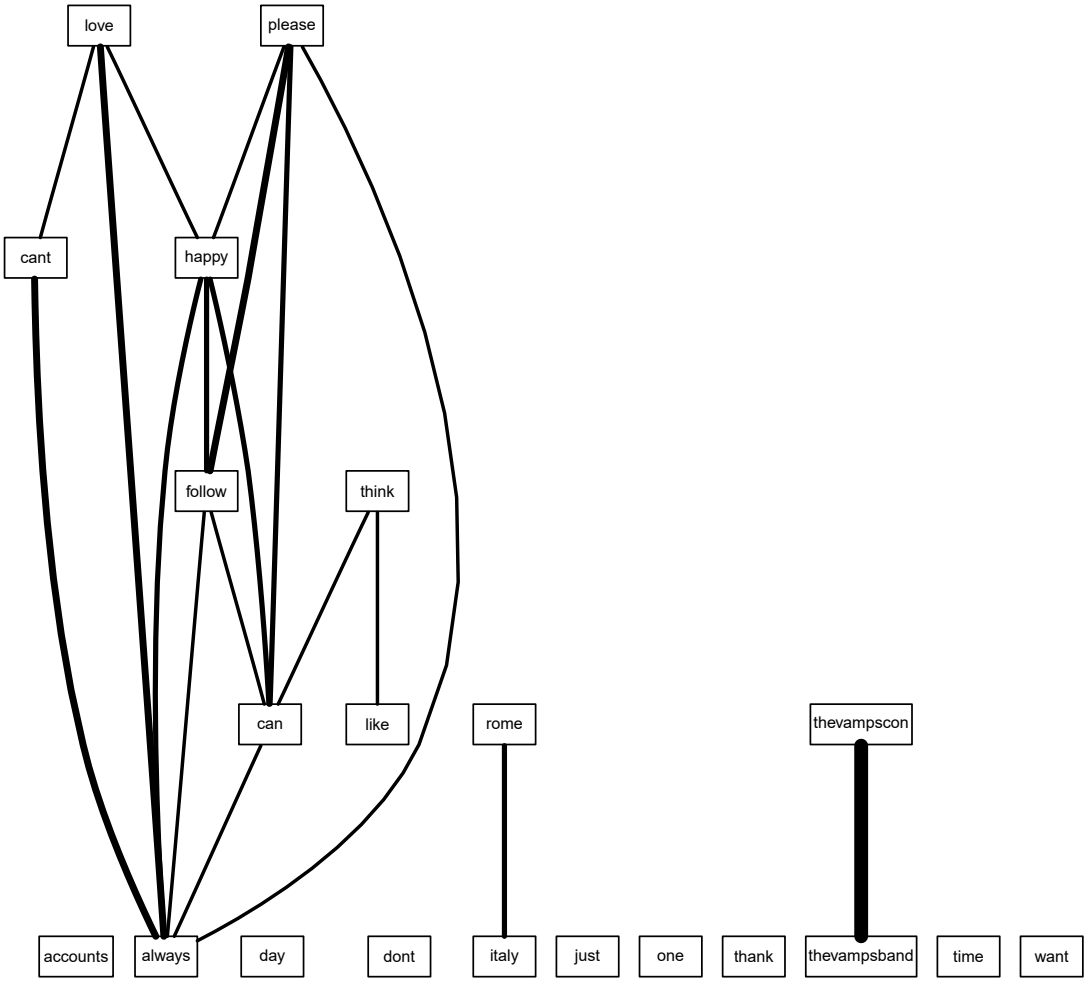

Abbildung 3.8.: Darstellung der Wörter und ihrer Korrelation ab einer Häufigkeit von 400 Vorkommen.

Im gegebenen Anwendungsfall geht es darum, Zusammenhänge, die durch Korrelation gefunden wurden, gewichtet abzubilden. Für den Graph in Abbildung 3.8 wurden alle Wörter aus dem Corpus ausgewählt, die mehr als 400-mal vorkommen. Als Datengrundlage diente die *Term-Document-Matrix*, aus der sich Korrelationen zwischen den einzelnen Begriffen errechnen lassen. Als Grenzwert für eine Korrelation wurde ein Schwellwert von 0.20 festgelegt. Dieser entspricht 20 % der Menge aller Begriffe. Liegt der Wert darüber, wird einen Linie zwischen den Wörtern gezeichnet. Je dicker diese Verbindungslinie, desto höher ist der Wert für die Korrelation. Diese Begriffen wurden damit offensichtlich häufiger in einem Text erwähnt.

Durch den Graphen wird so klar, wie stark ein Begriff mit einem anderen korreliert. Es werden hier auf einen Blick Verbindungen deutlich, die sonst erst abgefragt werden müssten und so einfach entdeckt werden können.

3.3.5. Kartendarstellungen

Zur Visualisierung der räumlichen Verteilung bieten sich Karten an. Diese sind wie folgt definiert: „Eine Karte [...] ist eine grundrissbezogene graphische Repräsentation georäumlichen Wissens auf der Basis kartographischer Abbildungsbedingungen.“ (Bollmann, 2001) Nicht zu verachten ist bei der räumlichen Darstellung die besondere Natur der Daten und ihre Visualisierung. Es können nicht immer die klassischen Methoden angewandt werden, zum einen, weil eine Generalisierung viel zu aufwändig wäre, nur um die Daten zu erforschen, aber auch, weil die Aussage noch keineswegs feststeht.

Eine recht verbreitete Variante zur Verortung von Inhalten auf Webkarten ist der Einsatz von Markern, wie er unter anderem in Abbildung 3.1 zu sehen ist. Die Herstellung einer solchen Karte kann schnell erfolgen und bietet eine gute Möglichkeit, die Verbreitung und die Inhalte zu erkunden. Allerdings stößt man schnell an die Grenzen, wenn die Karte zu voll mit Markern ist. Dann kann die Verbreitung nur noch bedingt erfasst werden und die Auswahl eines einzelnen Markers wird erschwert. Dazu kommen noch die Performance-Einbußen durch eine solche Vielzahl an Objekten. Mit einer begrenzten Anzahl von Objekten ist diese Methode aber sehr effektiv. Zusätzlich könnten die Marker gegebenenfalls unterschiedlich eingefärbt und dadurch Kategorien repräsentiert werden.

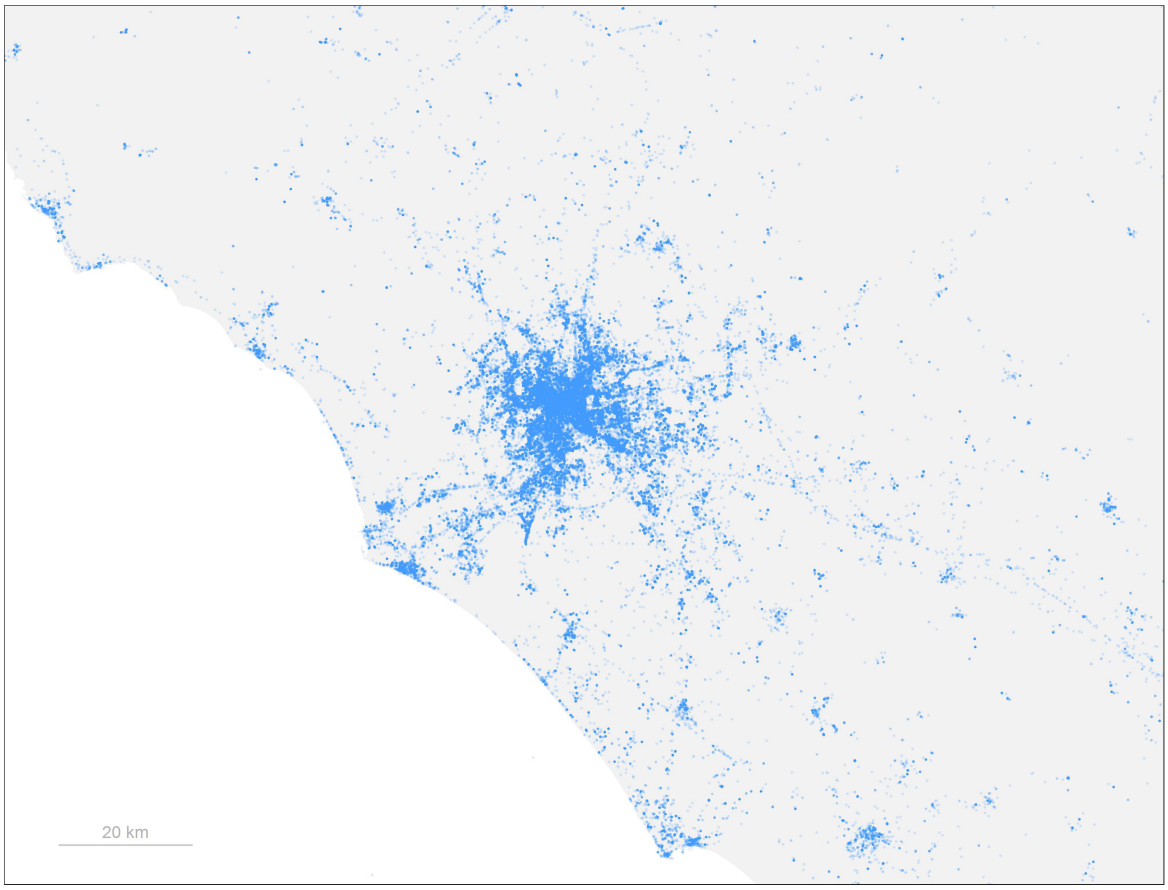

Abbildung 3.9.: Verteilung der Tweets im April 2015. Je blauer eine Region ist, desto mehr Tweets wurden dort verortet.

Eine eher klassische Variante ist die Anwendung der Punktmethode. Hier steht meist ein Punkt für eine definierte Anzahl an Einheiten. Um in einem Punkt mehrere Tweets darzustellen, wäre zunächst eine Generalisierung nötig, die hier aber zu lange dauern würde, nur um eine Vorschau zu erzeugen. Als Lösungsweg bietet sich hier das *Overplotting* an. Dabei werden die einzelnen Punkte mit Transparenz versehen und übereinander dargestellt,

so dass sich durch die Überlagerung mehr oder weniger intensive Bereiche ergeben, welche wiederum Rückschlüsse auf die Anzahl zulassen. Ein Beispiel dafür ist in Abbildung 3.9 zu sehen. Jeder Punkt steht für einen Tweet. Der Transparenzwert wurde auf 20 % eingestellt. Dadurch sind Regionen mit vielen Tweets deutlich zu erkennen.

Die Tweets wurden in verschiedenen Sprach verfasst, welche wiederum detektiert wurden. Es wäre also von Interesse, die Verteilung der unterschiedlichen Sprachen zu betrachten. Die Vermutung liegt nahe, dass Menschen, die diese Sprache sprechen, die Orte auch aufgesucht haben, wo die Kurznachrichten verortet sind.

Stellt man nur eine Auswahl der in Tabelle 3.3 angegeben Sprachen auf einer Karte dar, so entsteht ein bunter Teppich von Punkten, wie er in Abbildung 3.10 zu sehen ist. Eine Aussage lässt sich davon aber schwer ableiten. Die Überlagerung der Punkte ergibt hier kein Bild. Es verändert sich dadurch die Farbe, unter anderem bedingt durch die Transparenz. Dadurch kann eine richtige Zuordnung in der Legende nicht mehr erfolgen.

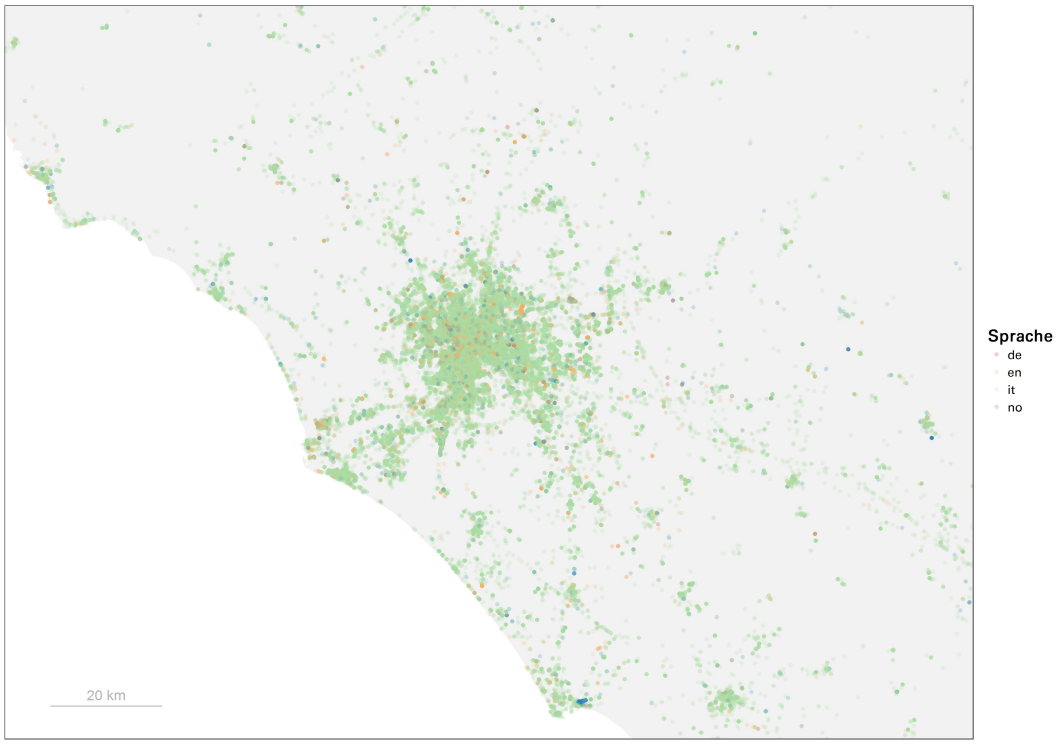

Abbildung 3.10.: Verteilung der Tweets in vier verschieden Sprachen im April 2015.

Als Lösung sind verschiedene Karten direkt nebeneinander denkbar, wie in Abbildung 3.11 zu sehen. Dadurch wird zum einen die unterschiedliche Größe der Datensätze deutlich sichtbar, aber auch sehr klar die Verteilung. Zur besseren Vergleichbarkeit wurden immer die gleichen graphischen Merkmale für die Punkte eingesetzt. Unterschiedliche Farben wären aber auch denkbar und würden die Unterschiede stärker hervorheben.

Es sind noch weitere Kartendarstellungen denkbar, wie zum Beispiel die Visualisierung der Hashtags, von Nutzern und deren Aktivitäten, Kurznachrichten, die Themen zugeordnet wurden, die Zuordnung der Tweets zu Regionen, welche dann als Chloroplethen Karte visualisiert werden oder die Visualisierung der Emoticons oder Emojis.[12] Für den hier verfolgten Zweck der Aufbereitung der Texte für eine weitere Analyse erscheinen diese Möglichkeiten zu aufwändig und sollen daher nicht weiter verfolgt werden.

[12] Hinweiszu besseren Abgrenzung: Emoticons können mit alphanumerischen Zeichen dargestellt werden, Emojis werden als eigenen Zeichen in einem Zeichensatz definiert.

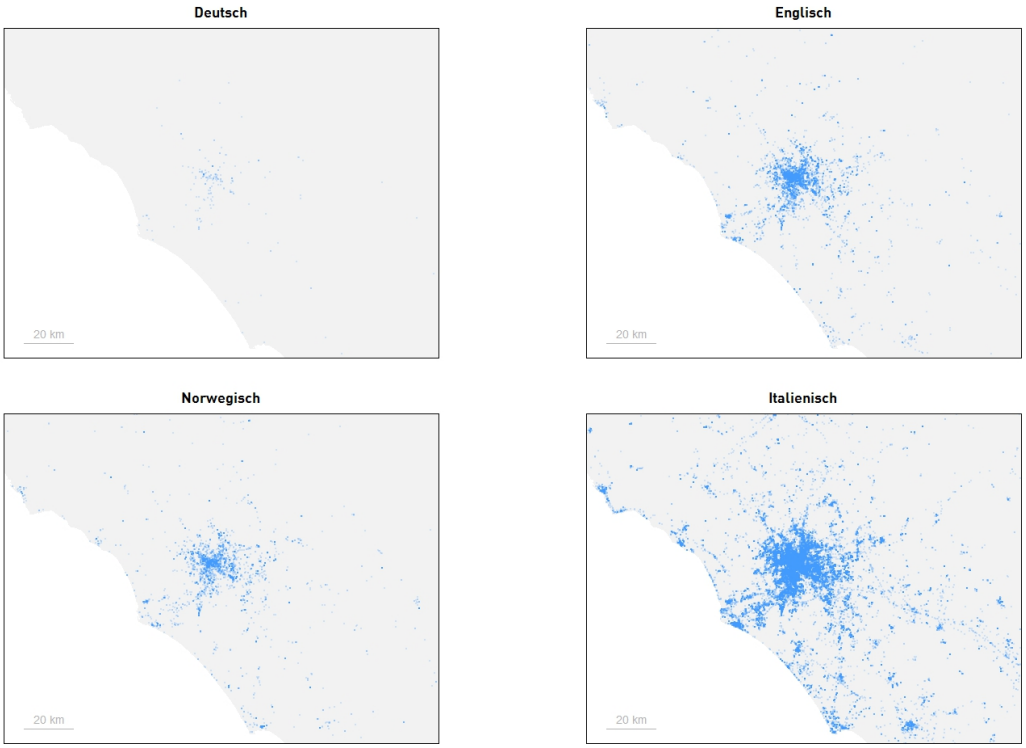

Abbildung 3.11.: Verteilung der Tweets in den vier verschieden Sprachen im April 2015. Je blauer eine Region ist, desto mehr Tweets wurden dort verortet.

4. Gewinnung, Aufbereitung und Analyse von Microblogging Content

Betrachtet man die Datenstruktur und die Eigenheiten der Plattformen für *Microblogging Content*, ergeben sich daraus drei verschiedene Ansätze für eine Analyse. Die einfachste Variante ist den Inhalt zu sammeln und dann auszuwerten. Dadurch kann aus einer Vielzahl von Texten erkannt werden, wo welches Thema zurzeit im Fokus steht, was die Nutzer damit assoziieren und ob sie positiv oder negative Emotionen damit verbinden. Im Kontext mit den Metadaten Zeit und Raum lassen sich dadurch Trends und Entwicklungen erkennen.

Ein anderer Ansatz zielt eher auf die Nutzer einer Plattform und ihre Vernetzung untereinander ab. Ob es nun Verlinkungen, Freundschaften oder Follower sind, es ist immer eine soziale Komponente mit in den Daten enthalten. Diese lässt sich mit ihren Verknüpfungen untereinander sehr gut als Graph darstellen. Georeferenziert man die Nutzerprofile sind Rückschlüsse auf die räumliche Vernetzung der Nutzer und die Reichweite des Inhalt möglich.

Verbindet man Inhalte mit den möglichen Reaktionen darauf lassen ebenfalls Schlüsse auf die Relevanz eines Beitrages zu. So kann ein Tweet bei Twitter Retweetet oder Favorisiert werden. Die reine Anzahl dieser Reaktionen ermöglicht bereits eine Einschätzung der Reichweite im jeweiligen Netzwerk. Zusätzlich kann noch die räumliche Reichweite untersucht werden, wenn man die einzelnen Nutzer und ihren Standort untersucht, welche die genannten Aktionen unternommen haben. Darüber hinaus kann man herausfinden ob der Inhalt der Ausschlag gebenden Nachricht, ein Thema mit räumlich begrenzter oder weltweiter Bedeutung behandelt.

Im Folgenden Teil wird ein Werkzeug Entworfen und Entwickelt, dass sich auf die Vorbereitung der Auswertung der Inhalte konzentriert. Dazu werden zu nächst die Anforderungen formuliert, um anschließend einen Prototypen zu entwerfen. Abschließend soll eine Software entwickelt und getestet werden. Die oben genannten Möglichkeiten für weitere Analyse sollen dabei vorerst außen vor bleiben.

Der Fokus liegt auf dem Text Mining in Verbindung mit Inhalten, die eine Koordinate besitzen. Für Texte, die ihren Ortsbezug über Inhalte herstellen, könnte dieser vorher automatisch ermittelt werden. Beispielsweise mit der in Abschnitt 2.4.5 beschriebenen API.

Um die Informationen mittels Text Mining zu finden, muss zunächst die Datengrundlage bereinigt werden. Wie in der digitalen Bildverarbeitung das Rauschen aus einem Bild entfernt wird, um die Aussagekraft zu erhöhen, so verhält es sich auch hier. Wetterinformationen, Mitfahrangebote und Verkehrsmeldungen, sind nicht immer von Interesse und sollen daher mit passenden Methoden herausgefiltert werden. Die Probleme dabei und ein möglicher Arbeitsablauf zu deren Lösung wurde bereits im Abschnitt 3.2 beschrieben.

4.1. Anforderung an eine Anwendung

4.1.1. Twitter als Beispiel

Als Beispiel soll hier Twitter dienen, da es sich um die bekannteste Plattform handelt auf die sich auch die meisten Publikationen beziehen. Auf Grund dessen ist die Nutzung der API auch sehr einfach möglich und in diversen Programmen bereits implementiert.

In einem weiteren Schritt soll dann noch untersucht werden, wie sich die Pipeline beziehungsweise der entworfene Arbeitsablauf, auf andere Daten übertragen lässt. Als Beispieldatensatz soll dazu neben dem für Twitter aus Abschnitt 3.3 verwendeten Daten auch noch ein Flickr-Datensatz genutzt werden. Dieser entspricht dem in Hauthal, 2015 verwendetem. Es wurden noch die Sprachen mittels des R-Paktes `textcat` detektiert und anschließend die Daten in einer MySQL-Datenbank gespeichert.

Zusätzlich dazu soll die Twitter-API mit eingebunden werden, um direkt Daten abgreifen zu können. Genutzt werden soll dafür die REST-API, welche aber gewissen Beschränkung unterliegt. Die Einschränkungen auf maximal 180 Suchen in 15 Minuten sollte nicht das Hindernis darstellen. (Twitter, 2015b) Eher ist da schon der Hinweis auf die Unvollständigkeit der Suchergebnisse zu beachten. Der Fokus liegt auf Relevanz und nicht Vollständigkeit, dafür soll die Streaming-API genutzt werden. (Twitter, 2015e) Es werden also nicht alle Ergebnisse geliefert, was sich zum Beispiel auch daran zeigt, dass meist nur Suchergebnisse aus den letzten zwei Monaten zurückgeliefert werden.

4.1.2. Die Pipeline

Die hier formulierten Anforderungen an eine Arbeitsablauf sind auf *Mircoblogging-Content* mit Ortsbezug zugeschnitten und können nicht beliebig auf andere Problemfälle übertragen werden. Es wird von kurzen unstrukturierten Texten in einer Sprache ausgegangen, welche sich nur auf ein Thema beziehen.

Bevor die Datenaufbereitung begingen kann, muss sichergestellt werden, dass die Daten auch Ortsbezug haben und die korrekte Kodierung verwendet wird. Dies sollte im Regelfall UTF-8 sein, so dass alle verwendeten Schriftzeichen auch richtig angezeigt werden können.

Diese Anforderungen beziehen sich mehr auf die Datenquelle und soll Rahmenbedingungen darstellen. Die Daten können entweder aus einer Datenbank abgefragt oder direkt über die API bezogen werden. Ob so auf einem kontinuierlichen Datenstrom gearbeitet wird oder auf einen gleichbleibenden Datenbestand zurückgegriffen wird, soll hierbei offenbleiben und dem Nutzer überlassen werden.

Der gesamte Ablauf ist in Abbildung 4.1 visuelle dargestellt und basiert auf dem Entwurf in Abschnitt 3.2. Eine Übersicht der Anforderungen an die zu einwickelnde Anwendung und ihrer Gewichtung befindet sich in Tabelle C.2.

Als erster Schritt für die Analyse muss eine Auswahl erfolgen, wobei Zeit, Ort, Thema oder auch ein Nutzer, ein Kriterium sein können. Das Thema kann durch einen oder mehrere Suchbegriff beschrieben werden, aber genauso auch ganz weggelassen werden. So soll es möglich sein, alle Nachrichten an einem bestimmten Ort zu erforschen. Dazu soll eine Methode zur räumlichen Auswahl zur Verfügung stehen. Komplexere Anfragen können aber immer noch bei Bedarf über die Datenquelle vorbereitet werden und dieses Kriterium außen vor gelassen werden. Eine zeitliche Eingrenzung wird immer gegeben sein, entweder durch eine API oder um das Datenvolumen auf ein prozessierbares Volumen zu beschränken. Die Möglichkeit sich einen Nutzer zur Analyse heraus zunehmen ist für Persönlichkeiten des öffentlichen Lebens oder Medienaccounts gedacht. Passend zur jeweiligen Auswahl sollten die Daten angezeigt werden können, beispielsweise als Karte oder Tabelle.

Die Möglichkeit einen bestimmten Nutzer der Plattform auszuwählen ist zwar hinsichtlich der Privatsphäre bedenklich, wenn es sich um den Account einer realen Person handelt. Als

Ausgangspunkt das Profil einer Organisation oder beispielsweise einen Aktivisten auszuwählen, erscheint hingegen unbedenklich und bietet Möglichkeiten die Wahrnehmung, die Aktivität und die Beteiligung an Geschehnissen zu verfolgen.

Im nächsten Schritt erfolgt dann die Aufbereitung. Diese soll, wie die Auswahl bereits, durch unterschiedliche Visualisierungen, wie Histogramme, Tabellen, Wortwolken und Kartendarstellungen unterstützt werden. Dafür muss in einem ersten Schritt noch die Sprache ausgewählt werden. Dies ist nötig, weil bestimmte Funktionen, wie Stemming und das Entfernen von Stoppwörtern Sprachspezifisch sind. Im Anschluss sollten unter anderem die URLs entfernt werden, da sie keine direkten Informationen enthalten und durch die nächsten Maßnahmen unter Umständen nicht mehr eindeutig zu identifizieren wären.

Anschließend sollen alle vorhanden Großbuchstaben in Kleinbuchstaben umgewandelt, sowie Zahlen, Sonderzeichen und Satzzeichen entfernt werden. Dabei soll aber dem Nutzer noch freie Hand gelassen werden, was er entfernen möchte. So könnte ein Dollar oder Euro-Zeichen durchaus von Belang sein für eine spätere Analyse. Danach können die Stoppwörter entfernt werden und das Stemming erfolgen. Der hier beschriebene Prozess soll wiederholbar sein und es ermöglichen einzelne Nutzer aus den Ergebnissen zu entfernen, wenn seine Nachrichten unerwünscht sind und in der Aufbereitung störend auffallen.

Insgesamt soll die Anwendung weitgehend selbsterklärend sein und Hinweise zum Vorgehen liefern, was eine graphische Benutzeroberfläche voraussetzt. Abschließend soll es möglich sein die bereinigten Daten in einem Format mit Ortsbezug zu exportieren, um eine spätere Analyse durchführen zu können.

4.2. Entwurf einer Anwendung

Die Anwendung soll mit Programmiersprache R umgesetzt werden, wobei die Funktionalitäten der Erweiterung `Shiny` zum Erstellen einer graphischen Oberfläche genutzt werden sollen. Die Oberfläche kann direkt aus R heraus aufgerufen werden und läuft dann in einem eigenen Fenster. Dabei handelt es sich nur um das Interface eines Browsers. Genauso kann auch jeder beliebiger Browser auf dem genutzten Computer verwendet werden.

Ebenfalls kann die Anwendung als Dienst über einen Server zur Verfügung gestellt werden. In diesem Falle entfällt die Installation von R sowie den nötigen Erweiterungen und es reicht ein Browser, egal auf welchem Gerät, mit einer Verbindung zum Server zur Verwendung aus. Der Server kann selbst betrieben werden, beispielsweise in einem Intranet, um das Datenvolumen besser zu verarbeiten und die Privatsphäre zu wahren. Aber auch die Nutzung der Plattform shinyapps.io[1] bietet sich an. Hier entfällt die Einrichtung und Wartung vollständig.

In Abbildung 4.2 sind zwei Entwürfe für die Benutzeroberfläche zu sehen. Der Nutzer soll immer auf der linken Seite die Optionen sehen können und auf der rechten Seite die verschiedene Ansichten, die daraus erzeugt werden. Zum einen sollen Registerkarten, auch *Tabsets* genannt eingesetzt werden, zum anderen vier Kacheln mit jeweils einer Ansicht, so dass diese direkt verglichen werden können und kein hin und herschalten nötig ist. Insgesamt soll der Benutzer Oben die einzelnen Schritte anwählen können, die aufeinander aufbauen. Der erste Schritt müsste somit die Auswahl sein, der nächste die Aufbereitung und der letzte Schritt der Export der Daten.

Zur Visualisierung der Daten sollen die im Abschnitt 3.3 vorgestellten Möglichkeiten angewendet werden, um die Analyse zu unterstützen. Bei den Kartendarstellungen sollen interaktive Markerkarten verwendet werden, die noch einen Zugriffe auf den Inhalt ermöglichen und dadurch der Bezug zu Orten durch die Hintergrundkarte leichter hergestellt werden kann. Es können so aber nicht alle Inhalte repräsentiert werden, da die maximale Anzahl an Markern durch die Performance beschränkt ist. Des Weiteren sollen Wortwolken und Histogramme die Aufbereitung noch unterstützen.

[1] `http://www.shinyapps.io/`

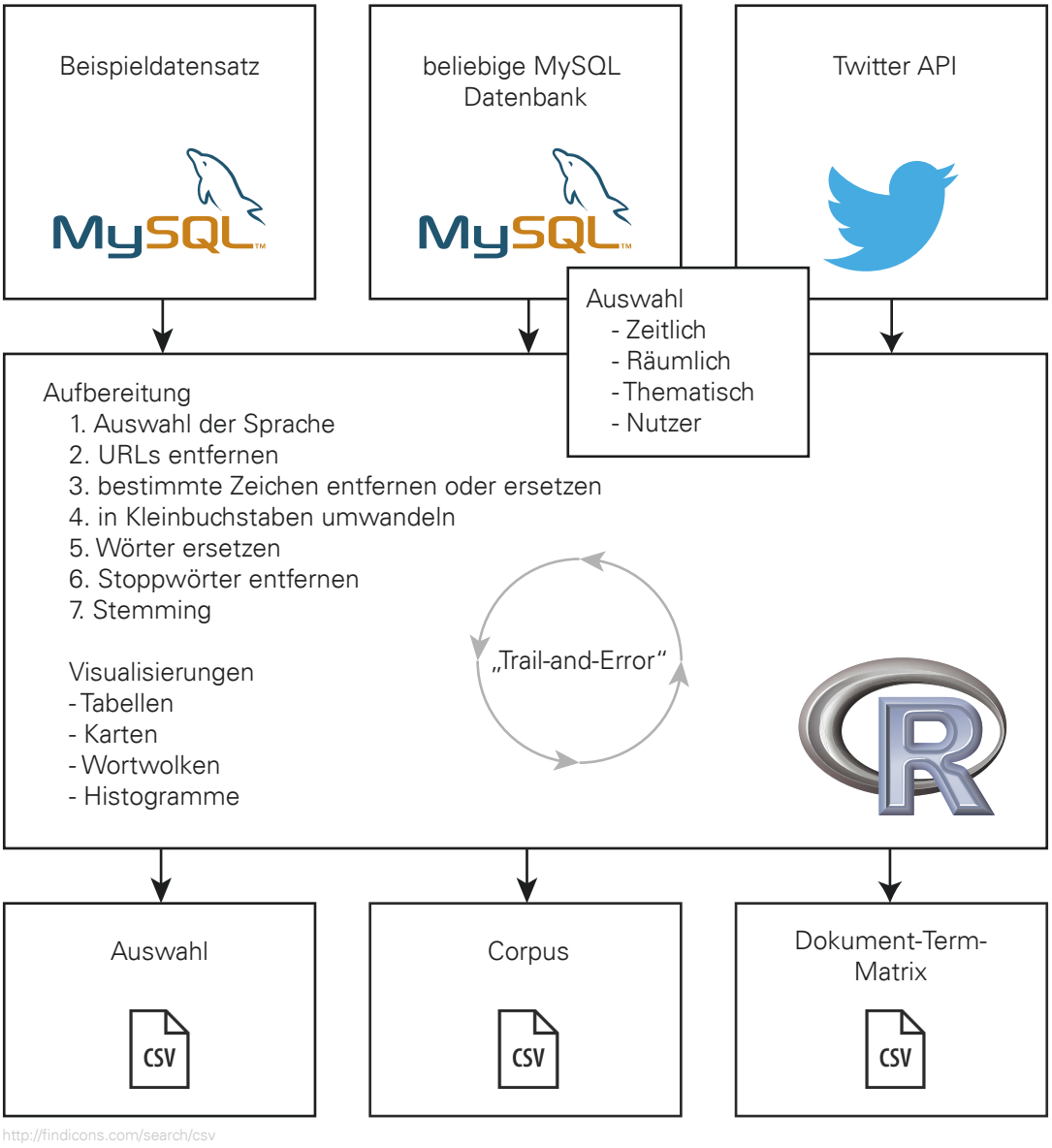

Abbildung 4.1.: Übersicht über den Arbeitsablauf.

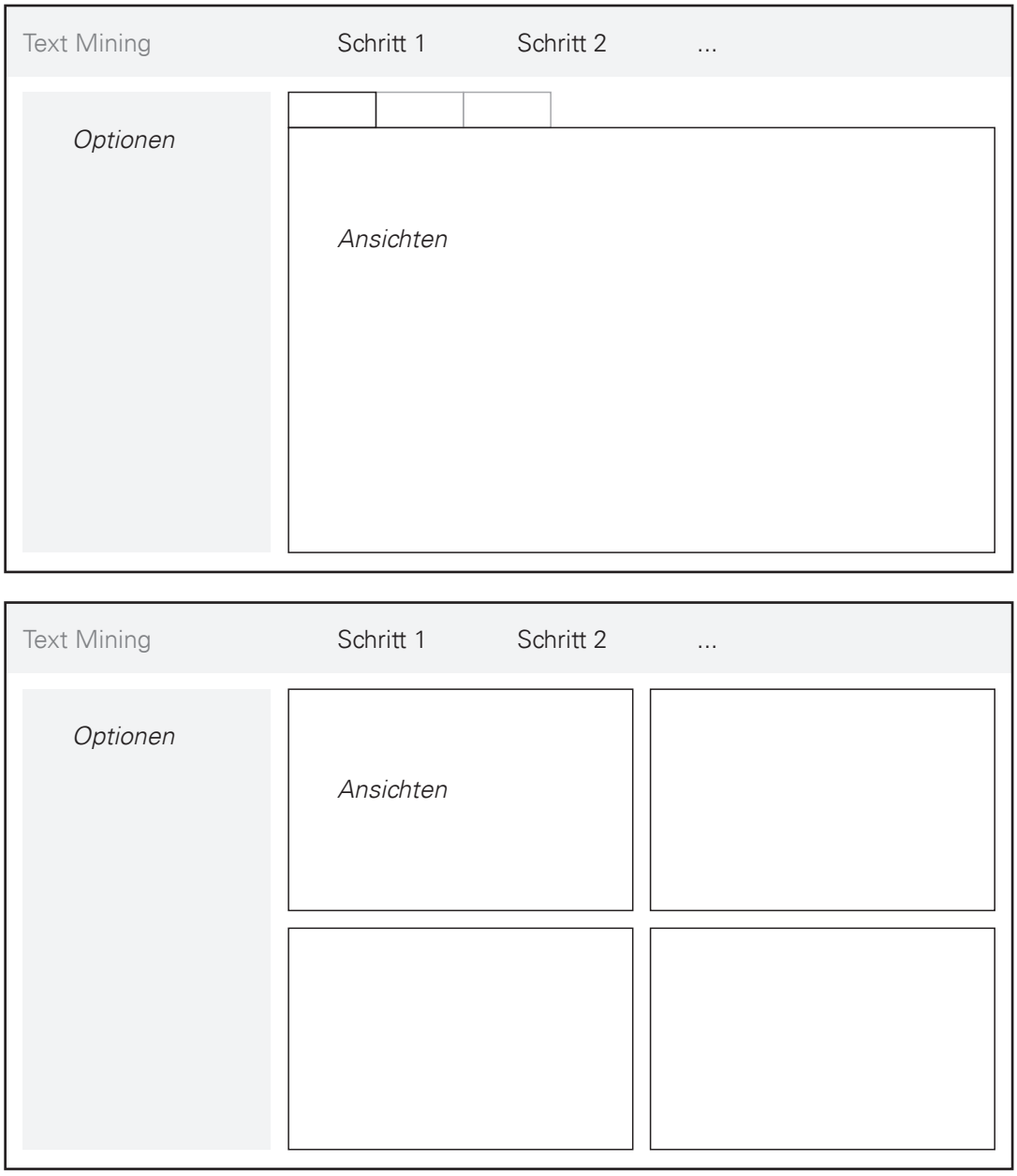

Abbildung 4.2.: Entwurf der Arbeitsoberfläche.

Um die Performance zu erhalten, erscheint es daher sinnvoll zwei Limits zu implementieren. Einmal für die Karte mit den Markern und zum anderen für die Suchergebnisse. So kann verhindert werden, dass es von Anfang an zu Problemen bei der Nutzung kommt. Diese Einschränkungen sollen vom Nutzer angepasst werden können, je nach Leistungsfähigkeit der Umgebung und dem Bedarf des Arbeitsablaufes.

Zum Testen der Anwendung soll der bereits verwendete Beispieldatensatz für Rom im April 2015 dienen und direkt mit eingebunden werden. Damit kann experimentiert werden und es muss nicht jedes Mal auf die Twitter-API zugegriffen werden, was zum einen Wartezeit verursacht und zum andern auch Limitiert ist. Die Beispieldaten werden dafür in einer MySQL-Datenbank abgelegt und bei Bedarf geladen. Als weitere Datenquellen kann neben Twitter, noch eine eigene MySQL-Datenbank angesprochen und abgefragt werden. Auf diesem Weg können beliebige Daten analysiert werden.

Die räumliche Abfrage soll als Umkreissuche umgesetzt werden, da sich diese am einfachsten Parametrisieren und Implementieren lässt. Zum einen bietet die genutzte Twitter-API keine andere Möglichkeit, zum anderen muss beachtet werden, dass in MySQL die Funktionen zur Verarbeitung und Speicherung von Geometrien noch sehr frisch sind und noch nicht von allen verwendeten Systemen unterstützt werden. Aufgrund dessen soll eine SQL-Funktion für die Umkreissuche verwendet werden und keine MySQL eigene Funktion. Da dies ein häufiger auftretendes Problem ist, steht auf GitHub[2] eine fertige frei nutzbare Funktion zur Verfügung.

Der abschließende Export von Daten soll ins CSV-Format erfolgen. Diese einfache, textbasierte Format kann von diversen Anwendungen zur Datenanaylse sowie Tabellenkalkulation eingelesen werden und ist auch mit GIS-Anwendungen kompatibel. Verfügbar sein sollen hier die ausgewählten Daten, die bereinigten Daten und die *Dokument-Term-Matrix*.

Die einzelnen Einstellungen zu speichern und zu laden scheint bei der noch überschaubaren Anzahl noch nicht nötig. Für den Fall, dass die gewünscht wäre, könnten diese ebenfalls als Tabelle ausgegeben werden. Von besonderem Interesse wären dabei die Stoppwörter, da diese sehr von den Inhalten abhängen und sehr spezifisch sein können. Das automatische wiederherstellen der Einstellungen erscheint als vorerst nicht umsetzbar und wird daher nicht in Betracht gezogen.

Als Zugabe könnten noch Zusammenhänge auf der Basis von Korrelation dargestellt werden und einen kleinen Einblick in mögliche Zusammenhänge liefern. Welche Wörter gemeinsam verwendet werden, lässt sich beispielsweise in einem Graphen gut darstellen und einen ersten Schritt in Richtung Analyse gehen, die mit der Pipeline vorbereitet werden soll.

4.3. Umsetzung

Auf Basis des Entwurfs wurde mit der Entwicklung einer *Shiny-App* in R begonnen. Dazu ist das R-Paket *shiny* erforderlich und die Verwendung des RStudios ist sinnvoll. Die Anwendung ist grundsätzlich aus zwei Dateien aufgebaut: Der `ui.R` für die Benutzeroberfläche und der `server.R` für die Umsetzung der Funktionalitäten. Zusätzlich können in der Datei `global.R` Variablen, Funktionen und Pakete definiert werden, die dann in den beiden anderen Dateien zur Verfügung stehen.

Auf Basis des Entwurfs im Abschnitt 4.2 wurde mit der Umsetzung der Benutzeroberfläche begonnen. Dabei stellte sich heraus, dass die vier gleichzeitigen Ansichten, wie in Abbildung 4.2, nur schwer umsetzen lassen und dann auf diesen durch die geringe Fläche zu wenig zu erkennen ist. Auf Grund dessen ist stets auf die Registerkartenansicht zurückgegriffen worden.

Bei der Entwicklung konnten einzelne Codeteile weiterverwendete werden, die schon beim Testen und beim Erstellen der Grafiken für Abschnitte 3.3 verwendet wurden. So lag der eigentliche Schwerpunkt auf der Anbindung verschiedener Datenquellen und dem schaffen einer Oberfläche für die verschiedenen Werkzeuge.

[2] `https://github.com/mattg888/MySQL-Distance-Calculation-Functions`

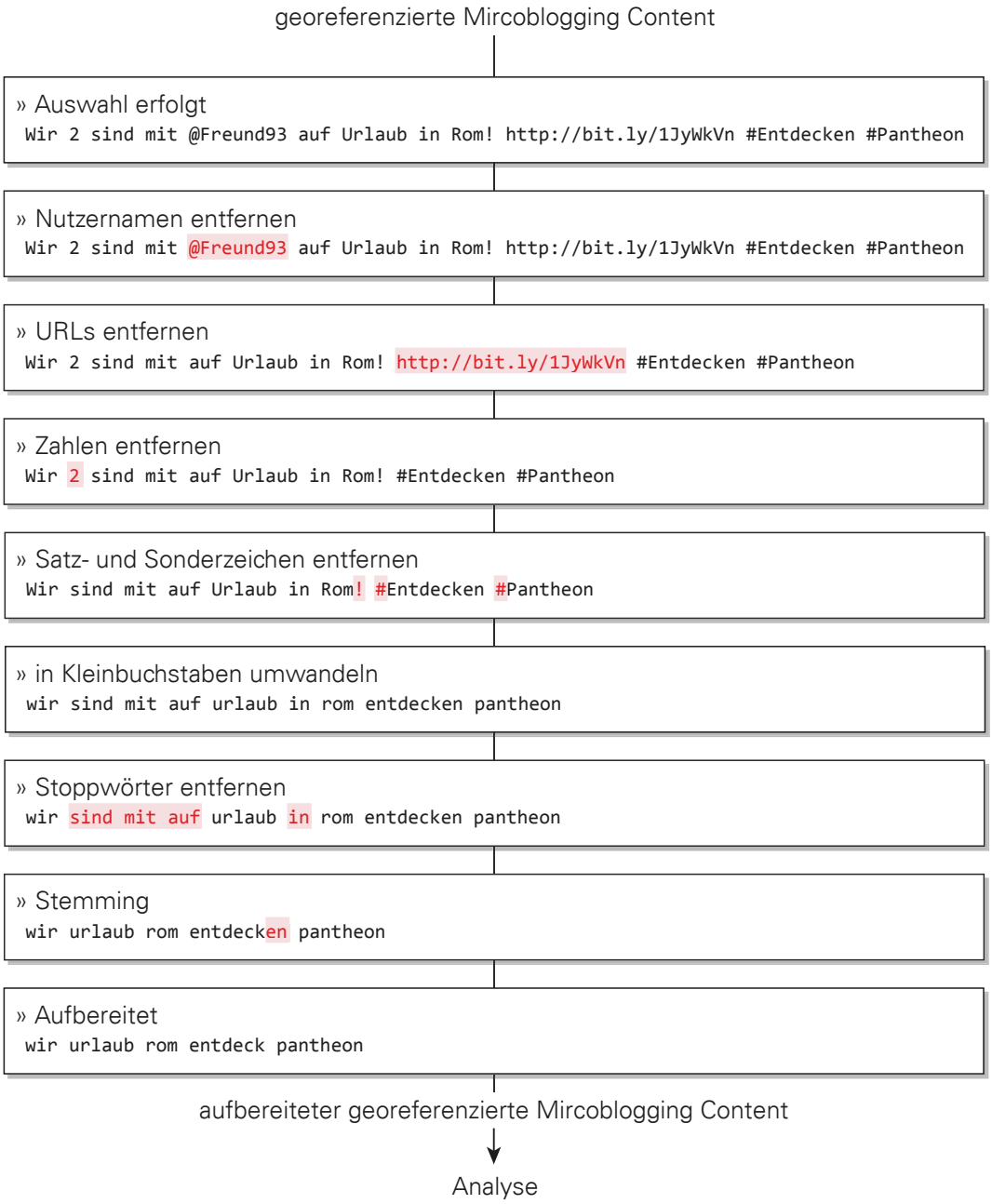

Abbildung 4.3.: Beispiel für den Ablauf der Aufbereitung und Auswirkungen der Schritte. Rot markierte Zeichen werden gelöscht.

In Abbildung 4.3 sind die einzelnen Arbeitsschritte zur Aufbereitung graphisch dargestellt und die Veränderungen hervorgehoben. Die einzelnen Funktion zur Aufbereitung stammen unter anderem aus dem R-Paket `tm` für das Text Mining oder wurden mit Hilfe von Regulären Ausrücken umgesetzt. Als Stemmer wurde über R auf die `Snowball`[3] Implementierung von Martin Porter zurückgegriffen.

4.3.1. Prototyp

Im folgenden Abschnitt ist der Prototyp beschrieben. Abbildungen mit der Ansicht aller Seiten sind im Anhang B abgedruckt. Ebenso findet sich dort eine Auflistung aller verwendeter R-Pakete in Tabelle C.1. Einen ersten Eindruck der Anwendung vermittelt Abbildung 4.4. Die einzelnen Schritte sind in gewohnter Leserichtung als Unterseiten von links nach rechts abzuarbeiten und sind immer gleich aufgebaut.

Zunächst ist die `Auswahl` einer Menge von Dokumenten aus dem vorhanden Datenbestand zu treffen. Dabei kann auf verschiedene Datenquellen zugegriffen werden, die unter dem Punkt `Datenquellen` zur Auswahl stehen. Anhand der einzustellenden Kriterien Zeitraum, Ort, Suchbegriff und Nutzer erfolgt eine Abfrage des Datenbestandes. Das Ergebnis wird dann auf der rechten Seite dargestellt. Dabei kann in der Tabelle in die Daten eingesehen werden, auf der Karte eine Überblick über die Verteilung gewonnen werden und in der Zusammenfassung statistische Kennziffern betrachtet werden. Für die Tabelle wurde das R-Paket `DT` verwendet, was interaktive Tabellen bereit stellt und eine angepasste Variante des Webmappingen Clients *Leaflet* als gleichnamiges R-Paket für die Kartendarstellung. Aus Performancegründen werden nicht alle Ergebnisse dargestellt, die genaue Anzahl kann in den `Einstellungen` angepasst werden. Die Zusammenfassung basiert auf der R-Funktion `summary()`.

Die nächste Seite `Filtern` ist im Abschnitt 4.3.2 genauer beschrieben, da der Inhalt nicht in den Anforderungen definiert wurde. Dieser Schritt steht vor allen anderen Aufbereitungsfunktionen und beinhaltet nur nicht Sprachspezifische Funktionen.

Unter `Zeichen filtern` muss dann eine Sprache ausgewählt werden. Anschließend beziehen sich alle Aktionen auf Texte mit dieser Sprache. Es können hier URLs, Zahlen, Satzzeichen und überflüssige Leerzeichen entfernt und alle Buchstaben in Kleinbuchstaben umgewandelt werden, so dass verschiedene Schreibweisen vereinheitlicht werden können. Die Option nur *Buchstaben behalten* kann in gewisser Weise als eine Zusammenfassung von mehreren einzelnen Aktionen betrachtet werden. Sollten diese generellen Funktionen zu unspezifisch sein, können auch drei eigenen Zeichen aus den Texten entfernt werden. Als Visualisierungen stehen eine Karte, ein Histogramm und eine Tabelle zur Verfügung. Die Schwellwerte bei dem Histogramm und der Wortwolke für die Anzeige sind direkt anpassbar. Allerdings wird von der verwendeten Funktionen teilweise die Untergrenze beim Erstellen der Wortwolke ignoriert.

Unter dem nächsten Punkt `Wörter filtern` kann eine Wort oder eine Zeichenkette durch eine beliebige andere Zeichenkette ersetzt werden. Es steht auch eine Menge von Stoppwörtern für eine Auswahl von Sprachen bereit. Diese können genutzt werden und bei Bedarf noch durch eigene Wörter ergänzt werden. Ebenfalls kann Stemming für einige Sprachen durchgeführt werden. Zur graphischen Analyse stehen wieder die Wortwolke, das Histogramm und die Tabelle zur Verfügung. Zusätzlich werden noch die standardmäßig verwendeten Stoppwörtern in einem eigenen Reiter ausgegeben.

Der Punkt `Analyse` ist ebenfalls eine Erweiterung, die noch die Funktionen der Pipeline abrunden soll und im Abschnitt 4.3.2 genauer beschrieben. Daher gleich zum Punkt `Export`: Hier kann die Auswahl, die aufbereiteten Texte und die *Dokument-Term-Matrix* heruntergeladen werden. Jeweils wird dafür eine Vorschau angeboten, um schon vorab einen in Einblick in die Daten und ihre Struktur zu gewähren.

[3] `http://snowball.tartarus.org/`

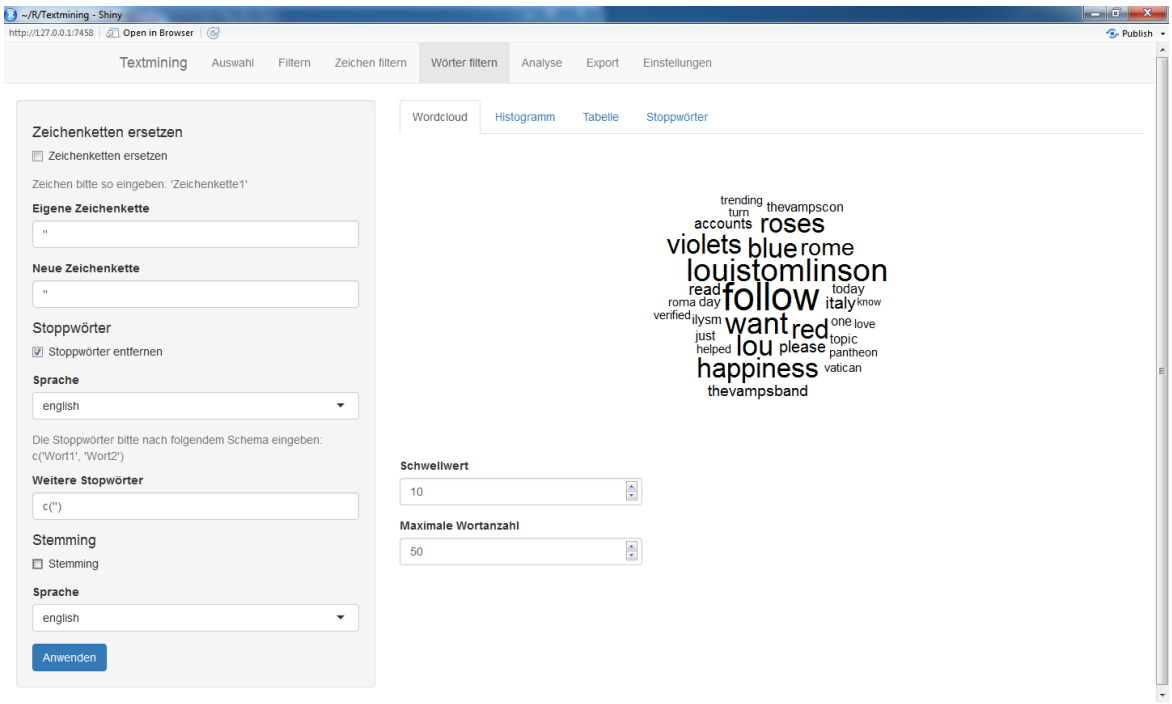

Abbildung 4.4.: Eine Ansicht des Prototypen.

In den `Einstellungen` können die Zugangsdaten zur einer MySQL-Datenbank oder für die Twitter-API eingegeben werden. Dazu sind jeweils noch Hinweise angegeben. Auf der linken Seite kann ausgewählt werden, welche Datenquelle gerade verwendet werden soll. Dazu sind noch zwei Orte mit Koordinaten vorbereitet, die hier angewählt werden können. Diese werden dann bei der Suche eingesetzt und verwendet, können aber auch vom Nutzer geändert werden. Des Weiteren kann noch das Limit für die Marker auf der *Leaflet*-Karten und das für die Suchergebnisse angepasst werden.

4.3.2. Erweiterung

Nach der Erfüllung der gesteckten Ziele erschien es noch zweckmäßig noch einige weitere Funktionen zu dem Prototypen hinzuzufügen. Diese wurden vor der eigentlichen Aufbereitung eingereiht und modifizieren schon zum Teil den Inhalt der Texte.

So können unter anderem nun Duplikate aus der ausgewählten Menge an Texten entfernt werden. Im Fall des Beispiels Twitter handelt es sich dabei meist um Retweets. Des Weiten können die Nutzernamen entfernt werden. Es kann ansonsten sein, dass häufig genannte Nutzer mit ihrem Namen als Begriff sichtbar werden. Sollte dies nicht gewünscht sein, können diese entfernt werden. Erkannt werden sie anhand des At-Zeichens @ zu Beginn des Wortes. Zu beachten ist dabei aber, dass so auch zum Teil Ortsangabe ausgedrückt werden. So wird in `in Roma` häufig als `@Roma` geschrieben. Genauso kann das Entfernen von Hashtags mit ihrer vorangestellten Raute # Sinn ergeben oder nicht.

Ebenfalls wurden zur Funktion des Nutzer Herausfilterns noch eine Tabelle mit der Anzahl der Text aller Nutzer erstellt, so dass auf einen Blick ersichtlich wird, wer viele Nachrichten erstellt hat. Nimmt man an, dass überdurchschnittlich aktive Nutzer Bots oder Statusmelder sind, so können sie hier schnell identifiziert werden. Über die Funktion zum Nutzer filtern können sie dann aus der Auswahl entfernt werden. Die Übersicht der zeitlichen Verteilung von Nachrichten ist noch eine gute Möglichkeit besondere Ereignisse zu erkennen und gegebenenfalls die Suche noch etwas zu verfeinern.

Die Seite `Analyse` wurden bereits eher vorgeschlagen. Hier kann nach einem Begriff gesucht werden und dessen Assoziationen werden ab einem Schwellwert ausgegeben. Der Schwellwert hat einen Wertebereich von 0.00 bis 1.0, wobei 1.0 den 100 % entspricht. Bei `Vorkommen` werden alle Wörter ausgegeben, deren Häufigkeit über der angegeben Anzahl liegt. Aus der Korrelation und dem Vorkommen zusammen kann auch ein Graph erstellt werden, der die Korrelation zwischen den einzelnen Begriffen darstellt. Für eine gute Lesbarkeit muss aber ein wenig mit den Werten gespielt werden.

5. Diskussion

Im folgendem Kapitel sollen die Ergebnisse der Masterarbeit diskutiert und hinterfragt werden. Dabei soll zwischen dem entworfenen Arbeitsablauf als allgemeine Abfolge an Aufbereitungsschritten und dem umgesetzten Prototypen unterschieden werden. Auf diesem Weg kann besser auf die konzeptionellen und praktischen Probleme eingegangen werden. Diese ergeben sich zum Teil auch erst durch die Umsetzung und deren Rahmenbedingungen.

5.1. Diskussion der Pipeline

Da es allgemein um die Aufbereitung von *Microblogging Conent* geht, ist der Arbeitsablauf sehr einfach gehalten und nicht auf die Erfordernisse von einzelnen Datenquellen angepasst. Dies hätte noch in der Pipeline im Schritt Eingrenzung berücksichtigt werden können. Da aber ein allgemein gefasster Ablauf das Ziel war, erschien dies nicht sinnvoll.

Die Pipeline wurden nach der Methode des *Geovisual Analytics* um Visualisierungen ergänzt. Diese gehören nicht direkt zu den Arbeitsschritten, sie sollen diese viel mehr unterstützen und vereinfachen. Aufgrund der inhomogene Datenstruktur können nicht immer die gleichen Parameter angewandt werden – diese müssen diese erst erschlossen werden. Dies kann am besten durch wiederholtes ausprobieren geschehen. Die verschiedenen vorgeschlagenen Darstellungen liefern hier Anhaltspunkte für eine Entscheidungsfindung.

Recht einfach ergibt sich, dass Anhand der Fragestellung eine Auswahl über Zeit, Raum, Thema und gegebenenfalls auch einen Nutzer getroffen werden kann. An dieser Stelle wäre es auch möglich fest zu legen, auf Basis welcher Sprachen vorgegangen werden soll, da diese dem Untersuchenden verständlich sein sollten. Geht man davon aus, dass die Sprache der Texte bekannt ist, wäre dies noch schlüssiger. Aber es lohnt sich durchaus auch in anderen Sprachen nach Schlagwörtern zu suchen, da diese zum Teil sprachübergreifend vorkommen. Teilweise lässt sich dies mit der verbreiteten Nutzung der englischen Sprache im Internet erklären. Ein weiteres Problem ist, dass Sprachen vermischt oder auch durch die verwendeten Algorithmen nicht richtig erkannt werden. So ergibt es zunächst einmal Sinn mehr Daten mit aufzubereiten und zumindest einen Blick hineinzuwerfen, anstatt sie gleich zu verwerfen.

Was die Aufbereitungsschritte der Texte mit der Entfernung von URLs und Satzzeichen anbelangt, herrscht weitestgehend ein Konsens vor, wie vorzugehen ist und dass alle Buchstaben kleingeschrieben werden. Dass man sich auf einzelne Nutzer konzentriert oder auch störende Nutzer entfernt ist abhängig von der Fragestellung.

Diskussionswürdig ist noch der Einsatz von *Part-of-Speech (POS)-Tagging* bei der Aufbereitung der Texte. Dabei werden einzelne Wörter auf ihre Aufgabe in einem Satz hin untersucht. Allerdings liegt hier schon das Problem: Es handelt sich nicht immer um ganze Sätze sowie die Verwendung von Hashtags und URLs stört diese Struktur nochmals zusätzlich.

Tabelle 5.1.: Einige Beispiele für häufig verwendete Abkürzungen auf Twitter. (Beal, 2014)

Abkürzung	Bedeutung
b/c	because
B	be
b4	before
BR	best regards
cld	could
cre8	create
fab	fabulous

Im Text Mining allgemein ist der Einsatz daher zu befürworten, bei *Microblogging Conent* hingegen erscheint die Verwendung nicht sinnvoll.

Möglich wäre es sich auch noch weitere Strukturen aus den Daten zu extrahieren. Es könnten wieder Verknüpfungen in den Texten über die Nutzernamen oder die Hashtags hergestellt werden. Ebenso könnten auch verschiedene Konversationen und Themen nachverfolgt werden. Dies ist aber durch die Konzentration auf georeferenzierte Tweets schwierig, da hier nur eine geringe Auswahl der gesamten Datenmenge genutzt wird. Es wären also große Lücken vorhanden, die die ganze Arbeit nicht sinnvoll erscheinen lassen. Ebenfalls würde dies nicht wieder auf alle Datenquelle anwendbar sein. Im Fall von Twitter könnte dies ebenso gut auf der Webseite durchgeführt werden, sofern alle Nachrichten noch zugänglich sind.

5.2. Diskussion des Prototypen

Die entwickelte Software ermöglicht mit verschiedene Verfahren die Aufbereitung von Texten für eine spätere Analyse. Dazu sind keine Programmierkenntnisse erforderlich. Die Verarbeitung erfolgt interaktiv und folgt dem Prinzip des *Visual Analytics*. Auch bei bereits reichlich vorhandener Erfahrung ermöglicht der Prototyp einen besseren Arbeitsablauf als über die Kommandozeile von R. Allerdings wird auf diese Weise ein Teil der Flexibilität eingebüßt. Es ist nicht möglich den Ablauf zu verändern oder zu erweitern. Die Möglichkeit, dass Werkzeug als Webservice anzubieten erleichtert den Einstieg und lässt ein komplexe Installation und Einrichtung wegfallen. Dadurch könnten bereits mehr Nutzer gewonnen werden als über eine klassische lokale Installation.

Die Entscheidung für R als Programmiersprache erscheint auf den ersten Blick ungewöhnlich. Es gibt viele Skriptsprachen, die sich zur Automatisierung von Aufgaben einsetzen lassen. Der Vorteil von R liegt aber in der Spezialisierung auf die Datenanalyse. Hier sind viele dafür benötigte Funktionen bereits vorhanden. Außerdem lassen sich sehr einfach Visualisierungen erstellen und anzeigen. In Verbindung mit dem Paket `shiny` konnte schnell und einfach eine Benutzeroberfläche geschaffen werden, welche auch optisch ansprechend strukturiert ist und aktuelle Technologien aufgreift. Nachteilig ist allerdings die Performance bei größeren Datenmengen. Hier kommt es zu merklichen Wartezeiten, die ein flüssiges Arbeiten erschweren.

In den Prototypen könnte noch weiteres Wissen eingebunden werden, was die Aufbereitung noch effektiver gestalten könnte. Oft werden bei Twitter häufig auf Grund der beschränkten Zeichenzahl Abkürzungen verwendet. Ersetzt man diese wieder durch ihre eigentlich Bedeutung, so kann der Text auch besser mit in die Analyse einbezogen werden. In Tabelle 5.1 sind einige Beispiele für Abkürzungen aufgeführt.

Ein weiterer Punkt ist die Verwendung des Stemmings: die Zurückführung auf eine Stammform durch einen Algorithmus ist schnell und einfach, aber funktioniert bei weitem nicht

	id	text	user	language	latitude	longitude	date
41	24853	Frauenkirche und Martin Luther Denkmal Nikon D90 Tokina 11-16mm	2179	german	51.05158	13.74113	2010-04-11 17:34:00
42	24868	Blick von den Br	1209	german	51.05226	13.73739	2010-04-10 16:11:00
43	24871	Joerg-Seidel-Photography	1209	middle_frisian	51.15974	13.68020	2010-04-12 18:37:00
44	24873	Semperoper at night, a famous building in Dresden, Germany. Don't think about beer (there is a beer-promotion on TV)! I am sure that I am not the first one with this shot	3430	english	51.05376	13.73596	2010-04-03 19:33:00
45	24876	Inside the Zwinger in Dresden	79	english	51.05275	13.73375	2010-04-02 16:45:00
46	24878	Kamelie in Pilnitz, Ostern 2010	829	german	51.01171	13.86951	2010-04-05 14:59:00
47	24879	Kamelie in Pilnitz, Ostern 2010	829	german	51.01171	13.86951	2010-04-05 15:00:00
48	24880	---BEST SEEN LARGE --- Frauenkirche, destroyed 1945, reconstructed 1995-2005 HDR Panorama, 9 exp. D80, Tokina 11-16mm@11mm, f/6.3, ISO 100, 2-15s, Photomatix software	1418	latin	51.05153	13.74138	2010-04-13 21:09:00
49	24882	Kamelie in Pilnitz, Ostern 2010	829	german	51.01171	13.86951	2010-04-05 14:59:00
50	24888	i realize a new wallpainting part of the exhibition " generation x generation y" at galerie adlergasse / runde ecke - riesa efau - dresden	2055	english	51.05566	13.72127	2010-04-13 15:47:00

Abbildung 5.1.: Beispiel für HTML-Code und Sonderzeichencodierungen in den Flickr-Daten

so gut wie eine Lemmatisierung. Der Ansatz über ein Lexikon zu arbeiten ist wesentlich komplexer, aber liefert die besseren Ergebnisse: Wörter in Form des korrekten Infinitivs und keine „entstellten“ Wörter. Zwar ist es durchaus praktisch, wenn aus „Rome“ und „Roma“ dadurch „Rom“ wird, aber die Stammform „happi“ von „happy“ oder „Itali“ von „Italy“ können Verwirrung stiften. Genauso sind unregelmäßig gebildete Wörter ein Problem, da sie nicht durch eine Zerlegung zurückgeführt werden können.

Verwendet man den Datensatz von Flickr, welcher über die MySQL-Datenbank eingebunden werden kann, wird deutlich, dass der Prototyp doch eher auf die Arbeit mit Twitter-Daten ausgelegt ist. Es fehlt die Möglichkeit HTML-Quellcode oder die Kodierung von Sonderzeichen aus den Texten zu entfernen. Abbildung 5.1 zeigt einige Beispiele für das Auftreten dieser Sachverhalte. Das Entfernen ließe sich auch noch automatisieren. Im Moment müssten noch alle Schlüsselwörter manuell entfernt werden.

Verbesserungswürdig ist ebenfalls der Umgang mit Codierungen. So kommt es beim Erstellen der Korpus für die Textaufbereitung teilweise zur Umwandlung von Zeichen. Aus diesem Grund funktioniert zum Beispiel die Suchfunktion der Datentabelle nicht richtig und meldet Fehler. Hier müsste noch nachgebessert werden. Da hier viele verschiedene Softwarepakete zusammenarbeiten ist das nicht einfach zu realisieren und kann nicht direkt vom Anwender beeinflusst werden. Daher ist vielmehr der Einsatz von *Workarounds* nötig als die Behebung von Fehler. Eine Möglichkeit wäre noch problematische Zeichen bereits vor dem Laden der in den Prototypen herauszufiltern.

Eine Überlegung besteht noch darin vollständige Verteilungskarten der verorteten Text zu generieren, wie sie in Abschnitt 3.3.5 vorgestellt werden. So könnte noch ein anderer Überblick gewonnen und besser die Verteilung von verschiedenen Sprachen herausgestellt werden. Der Idee stehen aber der hohe Rechenaufwand und die damit verbundene Wartezeit entgegen. Auch müsste dazu noch eine passende Datenquelle für die Hintergrundkarten bereitgehalten werden. Daher sollen die Marker-Karten vorerst ausreichen.

5.3. Potentielle Anwendungsmöglichkeiten

Nach dem nun die Daten aufbereitet wurden, kann eine weitergehende Analyse folgen. Hier soll nicht besprochen werden, wie genau diese erfolgen soll, sondern welche Anwendungen auf geschaffenen Datenbasis möglich wären.

Im Arbeitsablauf der Aufbereitung wurden visuelle Analysen genutzt um Verteilungen zu erschließen. Der Zusammenhang zwischen elektrischen Licht wurde von Ríos, 2013a bereits erwiesen. Hahmann, Purves und Burghardt, 2014 kann eine Verbindung zwischen den Inhalten von *Microblogging Content* und der Position herstellen. Dabei wird auch eine Ähnlichkeit der Verteilung von Tweets und Bevölkerungsdichte in Deutschland hingewiesen.

Basierend darauf kann man davon ausgehen, dass sich die Reaktionen auf Twitter proportional zu den Reaktionen von Ort verhalten. Die Ausbreitung einer Neuigkeit, oder was auch immer, könnte nachverfolgt werden. Ein Beispiel, wo dies geschehen ist, war die Aktivierung des privaten Twitter Accounts des US-Präsidenten Barack Obama. Auf einer Weltkarte[1] wurde dargestellt, wo und wann Reaktionen darauf erfolgt sind. Daraus kann geschlossen werden, das in den USA und Westeuropa ein starkes öffentliches Interesse am Präsidenten vorhanden ist. Abstrahiert man aus dem Beispiel einen Anwendungsbereich, ist es möglich die Interaktion zwischen Politik und Gesellschaft zu beobachten.

Ein mögliches weiteres Anwendungsgebiet wäre das Gesundheitswesen. Es könnte die Ausbreitung von Krankheiten verfolgt oder die Einstellungen zum Impfen analysiert werden. Hier wäre Radzikowski, Hollen und Fuhrmann, 2015 mit Ihrer Untersuchung zum Sonnenbräunen und Hautkrebs als Beispiel einzuordnen. Konkret könnte eine Impfkampangen über Twitter oder die Ausbreitung der all winterlichen Grippewelle beobachtet werden. Auch die Unterstützung in Notfallsituationen ist vorstellbar. So könnten Staus automatisch erkannt werden, anhand der Nachrichten der Nutzer vor Ort. Aber auch der Einsatz im Katastrophenmanagment ist ein Beispiel. Ein Hilferuf in Verbindung mit einem Bild macht einen anderen Eindruck, als nur ein Anruf.

Auf diese Art könnte man auch Bewertungen und Emotionen zusammentragen. Sei es für den Tourismus, der hier als Rahmenbeispiel gewählt wurde oder die Reaktionen auf Produkte. „Wen interessiert etwas?“, wäre hier hier eine Beispielfragestellung. Dass Twitter schon Teil der Marketingstrategie bekannter Unternehmen ist, wurde unter anderem im Abschnitt 2.3 deutlich. Kunden können einfach beobachtet und dabei herausgefunden werden, was sie am aktuellen Sortiment besonders interessiert und was sie stört. Ebenfalls könnte hier wieder die Sentiment Analyse zum Einsatz kommen.

[1] `https://twitterdata.cartodb.com/viz/452f2280-fd96-11e4-b6e7-7054d21a95e5/embed_map`

6. Fazit

Zusammenfassend lässt sich sagen, dass sich eine allgemeine Schrittfolge für die Aufbereitung von *Microblogging Content* beschreiben lässt. Eine genaue Anpassung auf die jeweilige Fragestellung ist allerdings unerlässlich. Jede Plattform auf der gebloggt und kommuniziert wird, hat ihre eigene Besonderheiten, auf die eingegangen werden sollte. Der entwickelte Prototyp für die Aufbereitung ist vor allem auf Twitter angepasst, aber auch allgemein einsetzbar, wie ein Test mit Flickr-Daten zeigte.

Für die Umsetzung wurde auf eine Vielzahl von schon vorhandenen Funktionalitäten zurückgegriffen und diese verkettet und um Visualisierungen ergänzt. Dadurch ist ein Arbeitsablauf entstanden, der von der schnellen Datenbeschaffung über eine Twitter-API bis hin zur ersten Analyse reicht. Dazu sind keine Programmierkenntnisse nötig. Um mit größeren Datenmenge arbeiten zu können, sind dann aber Kenntnisse in der Arbeit mit Datenbanksystemen und von Programmiersprachen hilfreich. Diese sind zur Verwaltung der Daten und der Nutzung weitere Analysewerkzeuge unerlässlich. Ebenfalls ist der Transfer dieser Daten nicht immer einfach über die Standardwerkzeuge zu bewältigen.

6.1. Wen interessiert was in Rom?

In der Einleitung wurde die Frage aufgeworfen, für welche Sehenswürdigkeiten sich Touristen in Rom interessieren. Zur Beantwortung wurden unter der Nutzung der Funktionen des Prototypen aus den Beispieldaten 50 000 Tweets in einem Umkreise von 20 Kilometern um das Kolosseum in Rom ausgewählt. Um zunächst einmal den Fokus auf Touristen zu legen, wurde Englisch als Sprache für die Analyse ausgewählt. Es wurden die Nutzernamen, die Duplikate, die URLs, die Satzzeichen, die Sonderzeichen sowie die Stoppwörter entfernt und anschließend das Stemming durchgeführt.

Es ergab sich ab einem frei gewählten Schwellwert von 100 Vorkommen folgende Wörtern, welche am häufigsten vorkamen: *day, easter, follow, happi, itali, just, like, love, one, roma, rome, see thank, think, vatican, want.* Deutlich wird hier, dass noch mehr Wörter entfernt werden sollten für eine weitere Analyse. Die zeitliche Nähe zum Osterfest wird ebenfalls sichtbar, wie die verschiedenen Schreibweisen für Rom. Diese wurden noch zusammengeführt auf „Roma“. Anschließend wurden die Assoziationen mit einem Anteil von mehr als 10 % in dieser Rangfolge ermittelt: *itali, pantheon, italia, history, collosseum, beauti, cappuccino.* So entsteht der Eindruck, dass die englischsprachigen Besucher Roms sich wesentlich mehr für das Pantheon interessieren, als für das Kolosseum.

Der Versuch eine ähnliche Auswertung für die deutsche Sprache durchzuführen scheiterte an der zu geringen Anteil an deutschen Tweets im sich ergebendem Zeitraum von etwa 7 Tagen. Eine Ausdehnung des Zeitraumes auf den gesamten Aprils half ebenfalls nicht. Insgesamt ist

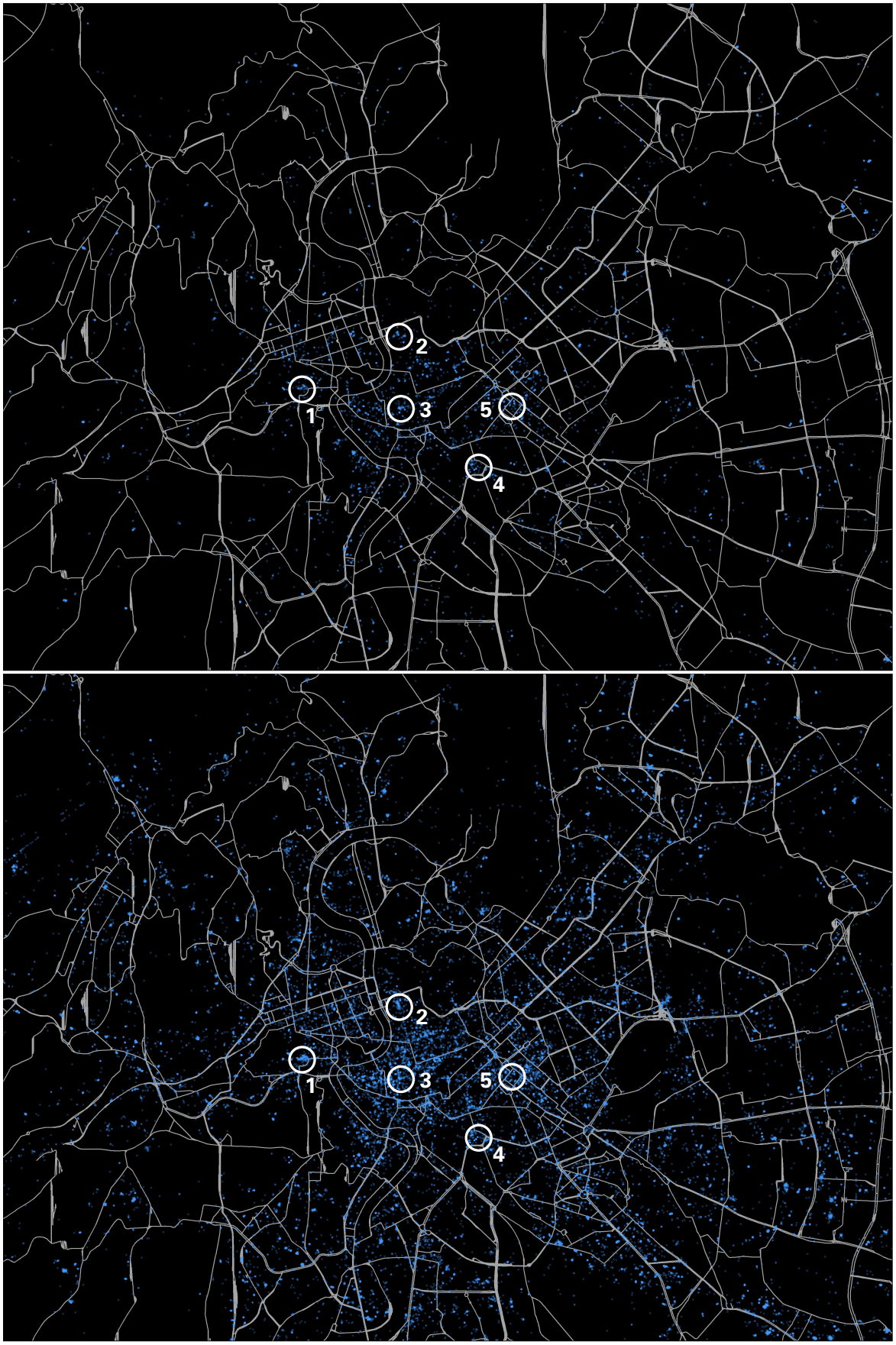

Abbildung 6.1.: Vergleich der Sprachverteilungen in Tweets: oben Englisch unten Italienisch, dazu sind die Hauptverkehrsstraßen dargestellt. Markiert sind folgende markanten Orte: ① Petersplatz, ② *Piazza del Popolo*, ③ Pantheon, ④ Kolosseum, ⑤ *Roma Termini* (Hauptbahnhof)

dafür die Datengrundlage auf der Basis von georeferenzierten Tweets zu gering. Die Frage, was die deutschsprachigen Besucher besonders interessiert lässt sich dadurch nicht beantworten.

Eine größere Datenmenge auszuwerten, ließ sich mit der Software nur bedingt interaktiv bewerkstelligen, da die Reaktionszeiten sehr lang werden. Problematisch war bereits die Arbeit mit den circa 25 000 italienischen Kurznachrichten, der hier verwendeten Auswahl. Ein interaktives Arbeiten ist also bei großen Datenmengen nicht möglich. Die genaue Grenze hängt natürlich von der verwendeten Computer-Hardware ab. Die circa 3 000 englischen Tweets waren kein Problem.

In der italienischen Sprache dominieren *rom, tendenz, trndnl* als häufigste Begriffe. Das *Roma* zu Rom geworden ist, ist auf das Stemming zurück zu führen. Die beiden anderen Worte sind wahrscheinlich durch Hashtags entstanden und können nicht zugeordnet werden. Hier dominiert offenbar das Rauschen und eine weitere Aufbereitung wäre nötig, was ohne genauere Kenntnis der Sprache nicht möglich ist.

Es bleibt letztlich nur die direkte Programmierarbeit für eine bessere Analyse mit mehr Daten und besseren Ergebnissen. So wurde dann auch eine ähnliche Abbildung 2.2 wie in der Einleitung direkt mit R erstellt. Die Abbildung 6.1 zeigt englische und italienische Tweets für April 2015. Die beiden Sprachen sind die mit den höchsten Anteilen. Sie wurden getrennt dargestellt, um Überlagerungen durch die unterschiedliche Häufigkeit zu vermeiden. Auffallend ist die gleichmäßigere Verteilung der italienischsprachigen Tweets, während bei den englischsprachigen Tweets die Sehenswürdigkeiten dominieren. Der Hauptbahnhof unterscheidet sich kaum in der Intensität, der Petersplatz hingegen stark. Hier ist offenbar das Interesse der italienischen Besucher deutlich stärker. So lässt sich zumindest auch optisch abschätzen, was englischsprachige Touristen interessiert. Der Eindruck passt insofern, dass die assoziierten Begriffe die Verteilung auf der Karte widerspiegeln. Der Vatikan scheint nicht so interessant zu sein für diese Besuchergruppe, das Pantheon und das Kolosseum deutlich mehr.

6.2. Zukünftige Arbeit

Im Laufe der Arbeit stellte sich mehr und mehr heraus, dass eine größere Zahl an georeferenzierte Tweets nötig sind. Nimmt man die Umgebung von Dresden als Beispiel, so sind zu wenige Tweets mit Koordinaten vorhanden um sinnvolle Analyse durchzuführen. Wenn dann noch ein bestimmtes Thema von Interesse ist, kann dieses dann kaum wahrgenommen werden. Daher sollten noch Tweets georeferenziet werden. Dazu wäre die Nutzung der unter Abschnitt 2.4.5 beschriebenen API GeoTxt sinnvoll. Dieser Arbeitsschritte könnte zu Beginn des beschriebenen Arbeitsablaufes eingefügt werden.

Verwendet man diese Herangehensweise, so ist eigentlich nur eine Eingrenzung über Begriffe oder Schlagwörter möglich in Verbindung mit einer zeitlichen Einschränkung. Eine räumliche Auswahl müsste dann in einem nächsten Schritt erfolgen. Denkt man noch weiter, dann ist das Auswahlkriterium über den Ort noch ein wenig kritisch zu hinterfragen. Wird zum Beispiel ein Tweet durch die Erwähnung des Bundeslandes Sachsen im Mittelpunkt der Fläche verortet, wurde aber in Leipzig abgesetzt wird der Ort stark verändert. Diese Ungenauigkeit kann die Analyse und damit die Ergebnisse verzerren. Eine Lösung wäre eine Gewichtung der verschiedenen Kurznachrichten, nach der Verortungsgenauigkeit, der Entfernung zu einem Punkt oder dem Vorkommen von bestimmten Schlagwörtern.

Überlegenswert wäre noch die Anwendung des maschinellen Lernens zum Erkennen von Statusmeldungen und Bots. Dadurch könnte noch einmal der Arbeitsaufwand in der Aufbereitung verringert werden. Bei der Wahl von R als Entwicklungsplattform wurde dies bereits berücksichtigt. Offen ist allerdings, wie das Anlernen des Algorithmus umgesetzt werden müsste und wie der höhere Rechenaufwand sich auf die Arbeit mit dem Prototypen auswirkt.

Literatur

Bateman, Scott, Carl Gutwin und Miguel Nacenta (2008). „Seeing Things in the Clouds: The Effect of Visual Features on Tag Cloud Selections“. In: *Proceedings of the Nineteenth ACM Conference on Hypertext and Hypermedia*. HT '08. Pittsburgh, PA, USA: ACM, S. 193–202. doi: http://doi.acm.org/10.1145/1379092.1379130. url: http://www.hci.usask.ca/publications/2008/fp039-bateman.pdf (besucht am 17. 06. 2015).

Bauer, Christian Alexander (2010). „User Generated Content – Urheberrechtliche Zulässigkeit nutzergenerierter Medieninhalte“. English. In: *Nutzergenerierte Inhalte als Gegenstand des Privatrechts*. Hrsg. von Henning Grosse Ruse-Khan, Nadine Klass und Silke von Lewinski. Bd. 15. MPI Studies on Intellectual Property, Competition and Tax Law. Springer Berlin Heidelberg, S. 1–42. doi: 10.1007/978-3-642-12411-2_1. url: http://dx.doi.org/10.1007/978-3-642-12411-2_1.

Beal, Vangie (2014). *Twitter Dictionary: A Guide to Understanding Twitter Lingo*. url: http://www.webopedia.com/quick_ref/Twitter_Dictionary_Guide.asp (besucht am 20. 09. 2015).

Bollmann, Jürgen [Hrsg.] (2001). *Lexikon der Kartographie und Geomatik in zwei Bänden*. Spektrum Akad. Verl.: Spektrum Akad. Verl. url: http://swbplus.bsz-bw.de/bsz089023811rez.htm.

Fischer, Eric (2010). *Locals and Tourists #13 (GTWA #5): Berlin. Blue pictures are by locals. Red pictures are by tourists. Yellow pictures might be by either. Base map © OpenStreetMap, CC-BY-SA*. Englisch. url: https://www.flickr.com/photos/walkingsf/4622375112/in/photostream/ (besucht am 26. 05. 2015).

– (2011). *Language communities of Twitter (European detail)*. url: https://www.flickr.com/photos/walkingsf/6276642489/in/album-72157631997324222/ (besucht am 07. 09. 2015).

followerwonk (2015). *Followerwonk: Twitter analytics, follower segmentation, social graph tracking, & more*. url: https://followerwonk.com/pro (besucht am 10. 09. 2015).

Hahmann, Stefan (2014). „Zur Beziehung von Raum und Inhalt nutzergenerierter geographischer Informationen“. Diss. url: http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-148835.

Hahmann, Stefan, Ross Purves und Dirk Burghardt (2014). „Twitter location (sometimes) matters: Exploring the relationship between georeferenced tweet content and nearby feature classes“. In: *Journal of Spatial Information Science*.

Hauthal, Eva (2015). „Detection, Modelling and Visualisation of Georeferenced Emotions from User-Generated Content; Detektion, Modellierung und Visualisierung ortsbezogener Emotionen aus nutzergenerierten Inhalten“. Englisch. Diss. Technische Universität Dresden, Fakultät Umweltwissenschaften. url: http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-163847.

Heimerl, F. u. a. (2014). „Word Cloud Explorer: Text Analytics Based on Word Clouds“. In: *System Sciences (HICSS), 2014 47th Hawaii International Conference on*, S. 1833–1842. doi: http://dx.doi.org/10.1109/HICSS.2014.231.

Hippner, Hajo und René Rentzmann (2015). *Text Mining*. url: https://www.gi.de/service/informatiklexikon/detailansicht/article/text-mining.html (besucht am 12. 09. 2015).

Insitut für Kartographie TU Dresden (2015). url: http://kartographie.geo.tu-dresden.de/downloads/twitter/readme.txt (besucht am 13. 07. 2015).

Janssen, Carolina (2013). *#Hochwasser – Die Jahrhundertflut auf Twitter*. url: https://blog.twitter.com/de/2013/hochwasser-die-jahrhundertflut-auf-twitter (besucht am 08. 09. 2015).

Julia H. Grace, Dejin Zhao und Danah Boyd, Hrsg. (2010). *Microblogging: What and How Can We Learn From It*. url: http://dmrussell.net/CHI2010/docs/p4517.pdf.

Karimzadeh, Morteza u. a. (2013). „GeoTxt: A Web API to Leverage Place References in Text“. In: *Proceedings of the 7th Workshop on Geographic Information Retrieval*. GIR '13. Orlando, Florida: ACM, S. 72–73. doi: 10.1145/2533888.2533942. url: http://doi.acm.org/10.1145/2533888.2533942.

Kremerskothen, Kay (2014). *Happy 10th Birthday, Flickr!* Englisch. url: http://blog.flickr.net/en/2014/02/10/happy-10th-birthday-flickr/ (besucht am 26. 05. 2015).

Lackes, Richard (2009). *Gabler Wirtschaftslexikon*. Hrsg. von Springer Gabler Verlag. url: http://wirtschaftslexikon.gabler.de/Archiv/57691/data-mining-v8.html (besucht am 28. 05. 2015).

Lackes, Richard und Markus Siepermann (2009). *Gabler Wirtschaftslexikon*. Hrsg. von Springer Gabler Verlag. url: http://wirtschaftslexikon.gabler.de/Archiv/75635/knowledge-discovery-in-databases-v8.html (besucht am 28. 05. 2015).

Lee, Bongshin u. a. (2010). „SparkClouds: Visualizing Trends in Tag Clouds“. In: *IEEE Transactions on Visualization and Computer Graphics* 16.6, S. 1182–1189. doi: 10.1109/TVCG.2010.194. url: http://dx.doi.org/10.1109/TVCG.2010.194.

Leetaru, Kalev u. a. (2013). „Mapping the global Twitter heartbeat: The geography of Twitter“. In: *First Monday* 18.5. url: http://journals.uic.edu/ojs/index.php/fm/article/view/4366 (besucht am 23. 05. 2015).

Luo, Wei und Alan M. MacEachren (2014). „Geo-social visual analysis“. In: *Journal of Spatial Information Science*. doi: 10.5311/JOSIS.2014.8.139. url: http://www.josis.org/index.php/josis/article/viewFile/139/103.

NASA (2000). *Eath's citiy lights*. Englisch. url: http://visibleearth.nasa.gov/view.php?id=55167 (besucht am 26. 05. 2015).

Nicola, Raluca Georgeta (2013). „Categorization and visualization of Twitter data“. Master Thesis. TU Dresden.

Radzikowski, Jacek Rafal, Heather Hollen und Sven Fuhrmann (2015). „Using Twitter Content to Crowdsource Opinions on Tanning in the United States“. In: *Kartogrpahische Nachrichten* (3/2015), S. 131–138. url: http://dgfk.net/download/openaccess/KN153_S_131-138.pdf.

Ríos, Miguel (2013a). *The geography of Tweets*. Englisch. url: https://blog.twitter.com/2013/the-geography-of-tweets (besucht am 19. 05. 2015).

– (2013b). *Visualization: Europe. The geography of Tweets: This image uses all of the geotagged Tweets since 2009 — billions of them. (Every dot is a Tweet, and the color is the Tweet count.) Copyright Twitter, Inc. (@twitter)*. url: https://www.flickr.com/photos/twitteroffice/8798022019/in/album-72157633647745984/ (besucht am 23. 05. 2015).

Rivadeneira, A. W. u. a. (2007). „Getting Our Head in the Clouds: Toward Evaluation Studies of Tagclouds“. In: *Proceedings of the SIGCHI Conference on Human Factors in*

Computing Systems. CHI '07. San Jose, California, USA: ACM, S. 995–998. DOI: 10.1145/1240624.1240775. URL: http://doi.acm.org/10.1145/1240624.1240775.

Roncero, Leticia (2015). *Eric Fischer's marvelous maps*. Englisch. URL: http://blog.flickr.net/en/2015/05/14/eric-fischers-marvelous-maps/ (besucht am 26. 05. 2015).

Ross, Zev (2014). *Mapping in R using the ggplot2 package*. URL: http://zevross.com/blog/2014/07/16/mapping-in-r-using-the-ggplot2-package/ (besucht am 24. 09. 2015).

Schade, Sven u. a. (2013). „Citizen-based sensing of crisis events: sensor web enablement for volunteered geographic information". English. In: *Applied Geomatics* 5.1, S. 3–18. DOI: 10.1007/s12518-011-0056-y. URL: http://dx.doi.org/10.1007/s12518-011-0056-y.

Solutions, Trendsmap (2015). *Twitter trend visualisations - Trendsmap solutions*. URL: http://solutions.trendsmap.com/visualisations/ (besucht am 10. 09. 2015).

Takhteyev, Yuri, Anatoliy Gruzd und Barry Wellman (2012). „Geography of Twitter networks". In: *Social Networks* 34.1. Capturing Context: Integrating Spatial and Social Network Analyses, S. 73–81. DOI: http://dx.doi.org/10.1016/j.socnet.2011.05.006. URL: http://www.sciencedirect.com/science/article/pii/S0378873311000359.

Tobler, Waldo (1970). „A computer movie simulating urban growth in the Detroit region". In: *Economic Geography*.

Tweetping (2015). *Tweetping*. URL: http://tweetping.net/ (besucht am 09. 09. 2015).

Twitter (2012). *Global town square. The global town square is where the biggest conversations of the year took place: events that generated a large number of Tweets and Retweets. Here are the major occurrences in 2012 for which the world came together to witness and discuss.* Englisch. URL: https://2012.twitter.com/en/global-town-square.html (besucht am 23. 05. 2015).

– (2014). *FAQs zum Hinzufügen Deines Standorts zu Deinen Tweets*. URL: https://support.twitter.com/articles/484789-faqs-zum-hinzufugen-deines-standorts-zu-deinen-tweets (besucht am 23. 05. 2015).
– (2015a). *About Twitter*. Englisch. URL: https://about.twitter.com/company (besucht am 20. 05. 2015).
– (2015b). *API Rate Limits | Twitter Developers*. URL: https://dev.twitter.com/rest/public/rate-limiting (besucht am 03. 09. 2015).
– (2015c). *Das Twitter Glossar*. URL: https://support.twitter.com/articles/473379-das-twitter-glossar (besucht am 20. 05. 2015).
– (2015d). *Erste Schritte auf Twitter*. URL: https://support.twitter.com/articles/324311-erste-schritte-auf-twitter (besucht am 20. 05. 2015).
– (2015e). *The Search API | Twitter Developers*. URL: https://dev.twitter.com/rest/public/search (besucht am 09. 09. 2015).

Zhao, Yanchang (2013). *R and Data Mining*. Hrsg. von Yangchang Zhao. Academic Press, S. 105–122. DOI: http://dx.doi.org/10.1016/B978-0-12-396963-7.00010-6. URL: http://www.sciencedirect.com/science/article/pii/B9780123969637000106.

– (2015). *Text Mining with R: Twitter Data Analysis*. URL: http://www.rdatamining.com/docs/text-mining-with-r-of-twitter-data-analysis (besucht am 03. 09. 2015).

A. Hinweise zur Nutzung des Prototypen

Im folgenden Abschnitt sind die einzelnen Arbeitsschritte zur Aufbereitung von Daten mit Hilfe des entwickelten Prototypen beschrieben. Dabei wird auf einen Beispieldatensatz zurückgegriffen, der alle georeferenzierten Tweets aus dem April 2015 um Rom in Italien enthält. Dieser wird standardmäßig mit passenden Limits beim Start der Anwendung geladen.

A.1. Auswahl

Die Maske für die Auswahl bleibt die gleiche, unabhängig davon, welche Datenquelle ausgewählt wird. Da das Verhalten leicht variieren kann, ist in den folgenden Abschnitten erklärt, was jeweils zu beachten ist. In jedem Fall muss jedoch nach dem Ändern von Kriterien oder der Datenquelle immer neu gesucht werden.

Das Suchergebnis wird stets in der `Tabelle` ausgegeben und kann durchgesehen werden. Im Tabellenkopf ist es möglich, Sortierungen vorzunehmen. Auf der `Karte` werden bis zur eingestellten Limit die Texte mit Markern verortet und können durch Klicken aufgerufen werden. Der `Zusammenfassung` sind die Minimal- und Maximalwerte für die einzelnen Felder zu entnehmen. Bei den Sprachen wird aufgelistet, welche am häufigsten vorkommen.

A.1.1. Beispieldatenbank

Der Zeitraum für die Beispieldatenbank reicht vom 1. April bis zum 30. April 2015. Der Breiten- und Längengrad ist bereits passend für Rom eingestellt, so dass sofort mit der Suche begonnen werden kann. Steht im Feld für den Suchbegriff oder den Nutzer das Sternchen `*` oder ist es leer, so werden alle Ergebnisse ausgegeben. Trägt man in das Feld für Nutzer beispielsweise *trendinaliaIT* ein, so erhält man alle Ergebnisse von diesem Nutzer für den angegebenen Zeitraum. Dazu kann weiterhin *roma* als Suchbegriff eingegeben werden. Dann werden ausschließlich Ergebnisse ausgeliefert, die beide Kriterien erfüllen.

A.1.2. MySQL

Momentan ist als Beispiel über MySQL der Flickr-Datensatz angebunden. Alternativ kann selbstverständlich an dieser Stelle jede andere MySQL-Datenbank mit den passenden Feldnamen eingebunden werden. Hinweise dazu sind auf der Seite `Einstellungen` vermerkt. Greift man auf die Flickr-Daten zu, so müssen die Koordinaten auf Dresden angepasst werden. Der Zeitraum vom 5. Juli 2000 bis zum 4. Juli 2013 ist durch die Datenbank abgedeckt. Die Suche nach Begriffen oder Nutzern verhält sich wie bei der Beispieldatenbank.

A.1.3. Twitter

Beim direkten Zugriff auf Twitter wird die REST-API genutzt. Dies bringt einige Limitierungen seitens Twitter mit sich. Beispielsweise kann nicht vollständig auf alle Daten zugegriffen werden, sowie ist der Zeitraum begrenzt. Daher wird bei der Auswahl der Datenquelle automatisch das Suchdatum auf die letzten zwei Tage gesetzt. Dies kann selbstverständlich zu einem späteren Zeitpunkt wieder angepasst werden.

Der Abruf der Daten über die API dauert, abhängig vom Datenvolumen, einen Moment. Im Anschluss daran wird eine Sprachdetektierung durchgeführt und nur die Ergebnisse ausgegeben, die auch wirklich eine Koordinate besitzen. Gegebenenfalls sollte die Menge der Suchergebnisse zunächst einmal verringert werden.

Die Verwendung von Suchbegriffen ist auch hier möglich. Wird allerdings nach einem bestimmten Nutzer auf Twitter gesucht, so wird auf eine weitere API-Funktion zurückgegriffen. Daher ist in diesem Fall die Kombination mit einem Suchbegriff und einer räumlichen Einschränkung vorerst nicht möglich. Falls hier keine Ergebnisse zu einem bestimmten Nutzer ausgegeben werden, kann es sein, dass dieser keine Tweets mit Koordinaten veröffentlicht hat.

A.2. Filtern

Im Abschnitt `Filtern` können Duplikate, wie beispielsweise Retweets, Nutzernamen oder Hashtags aus der Auswahl entfernt werden. Ebenfalls können einzelne Nutzer herausgefiltert werden. Dazu muss zunächst die zugehörige Checkbox mit einem Häkchen versehen werden, um die Option zu aktivieren. Das Format für die Eingabe erklärt sich durch die R-typische Struktur für Mengen. Ein Beispiel hierfür wäre: `c('trendinaliaIT','ziaTata72')`. Des Weiteren ist die Parametrisierung für die `Zeitliche Verteilung` zu beachten. Abhängig vom Zeitraum kann es vorkommen, dass 60 Minuten zum Zusammenfassen nicht genügen. Nachdem der Wert geändert wurde, wird nach einem Klick auf `Filtern` die Grafik neu erstellt.

A.3. Zeichen filtern

Als erster Arbeitsschritt unter `Zeichen filtern` muss eine Sprache aus der Auswahlliste gewählt werden. Nach dem Klicken auf `Anwenden` wird die Auswahl auf der Karte dargestellt. Alle weiteren Operationen gelten ab jetzt nur für Texte mit dieser Sprache. Die folgenden Optionen ermöglichen das Entfernen von bestimmten Zeichen aus den Texten. Das Ergebnis kann in der `Wortwolke`, dem `Histogramm` oder der `Tabelle` betrachtet werden. Die Schwellwerte lassen sich für die Grafiken ändern und erlauben eine Anpassung der Visualisierung. Wenn *English* als Sprache ausgewählt wurde, sollten sie bereits passen. Falls nur bestimmte Zeichen entfernt werden sollen, so kann dies nach der Aktivierung der Funktion `Zeichen entfernen` erfolgen.

A.4. Wörter filtern

Die Seite `Wörter filtern` ermöglicht es, bestimmte Zeichenketten durch andere zu ersetzen. Beispielsweise kann hier aus *rome* ein *roma* werden. Mit beiden Wörtern ist ja die gleiche Stadt gemeint. Des Weiteren können die passenden Stoppwörter für die ausgewählte Sprache entfernt und das *Stemming* durchgeführt werden. Falls noch weitere Wörter mit Hilfe der Visualisierungen als Stoppwörter auffallen, so können diese unter `weitere Stoppwörter` eingetragen werden. Hier könnten unter anderem `c('follow','and','red','want')` stehen. Zur weiteren Verfeinerung der Aufbereitung können die Visualisierungen genutzt werden. Unter `Stoppwörter` wird die interne Auswahl dieser ausgegeben.

A.5. Analyse

Unter dem Punkte `Analyse` kann auf Basis der Korrelation und der Häufigkeit des Auftretens von Wörtern ausgetestet werden, wie die Wörter miteinander verknüpft sind. Bei `Begriff` kann exemplarisch *roma* eingetragen werden. Nach dem Klick auf `Suchen` erfolgt die Ausgabe der Wörter mit dem Korrelationswert. Setzt man zu dem die `Anzahl` (wie oft ein Wort mindestens vorkommt) auf *20*, dann entsteht ein ansehnlicher `Graph`, der die Zusammenhänge zwischen den Wörtern darstellt. Unter `Vorkommen` erscheint eine Liste der verwendeten Begriffe. Der Schwellwert für die Korrelation kann ebenfalls angepasst werden.

A.6. Export

Auf der nächsten Seite `Export` kann ausgewählt werden, ob die Auswahl, der Corpus (die aufbereiteten Daten) oder die *Document-Term-Matrix* als CSV-Datei exportiert werden soll. Klickt man auf `Übernehmen` so wird eine Vorschau auf die Daten angezeigt.

A.7. Einstellungen

In den `Einstellungen` kann die Datenquelle ausgewählt und konfiguriert werden. Hier sind weiterhin explizite Hinweise zu Beachtenswertem gegeben. Unter `Ort für Suche` kann *Dresden* oder *Rom* ausgewählt werden. Dann werden jeweils passende Koordinaten unter dem Punkt `Auswahl` eingesetzt. In Dresden wurde die Frauenkirche und in Rom das Kolosseum als Basis für den exakten Standort ausgewählt. Die Limits für die Anzahl der Marker auf der Karte und die Anzahl der Suchergebnisse können beliebig angepasst werden. Sie sind jeweils von der Leistungsfähigkeit der verwendeten Hardware abhängig.

B. Abbildungen

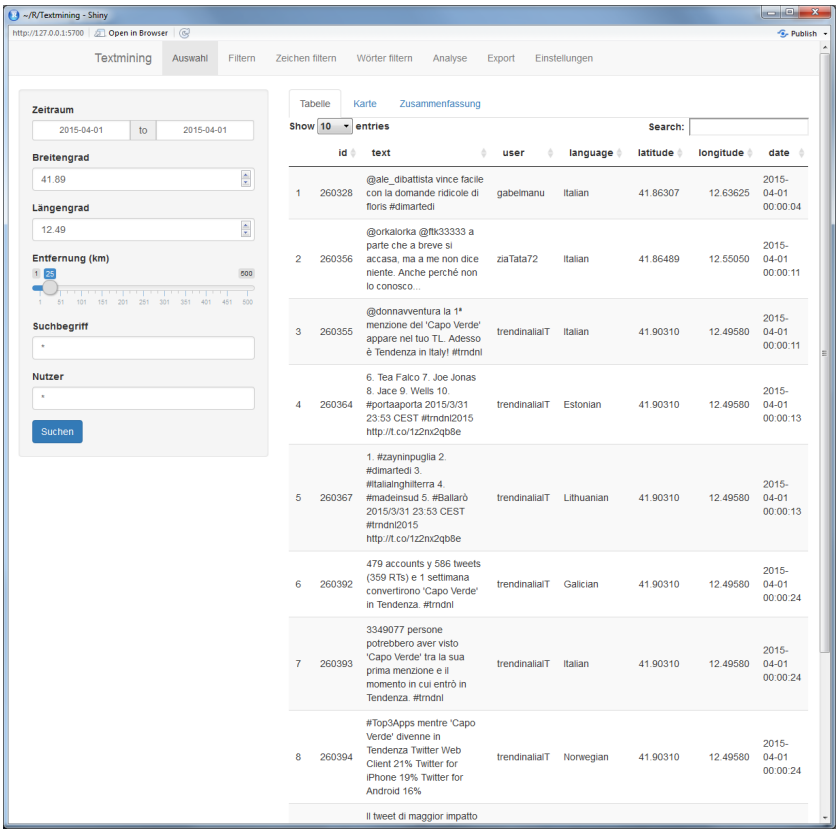

Abbildung B.1.: Auswahl

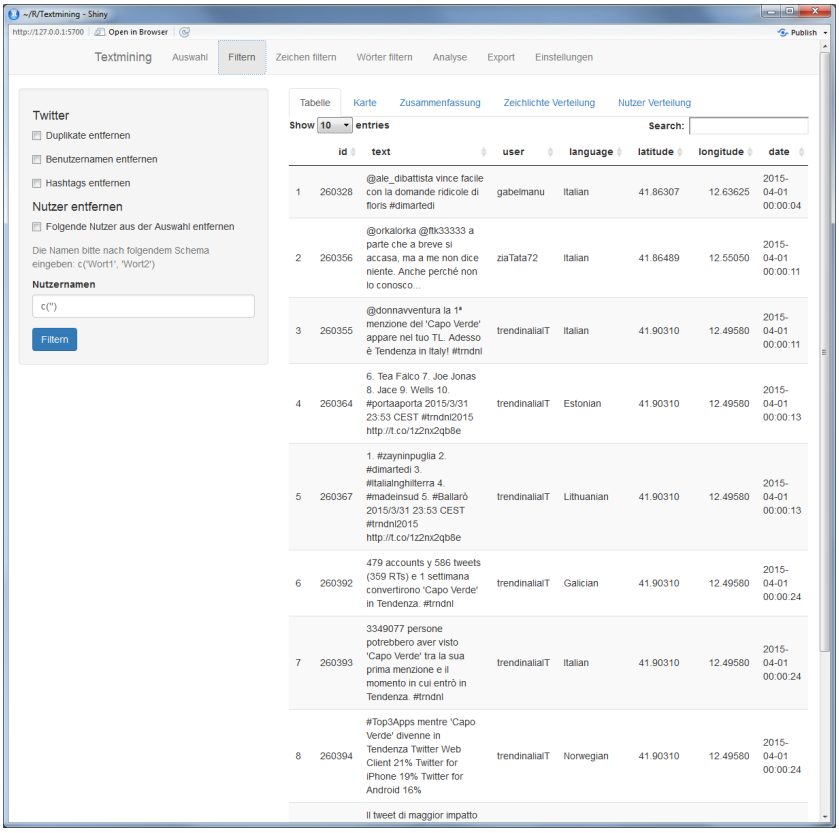

Abbildung B.2.: Filtern

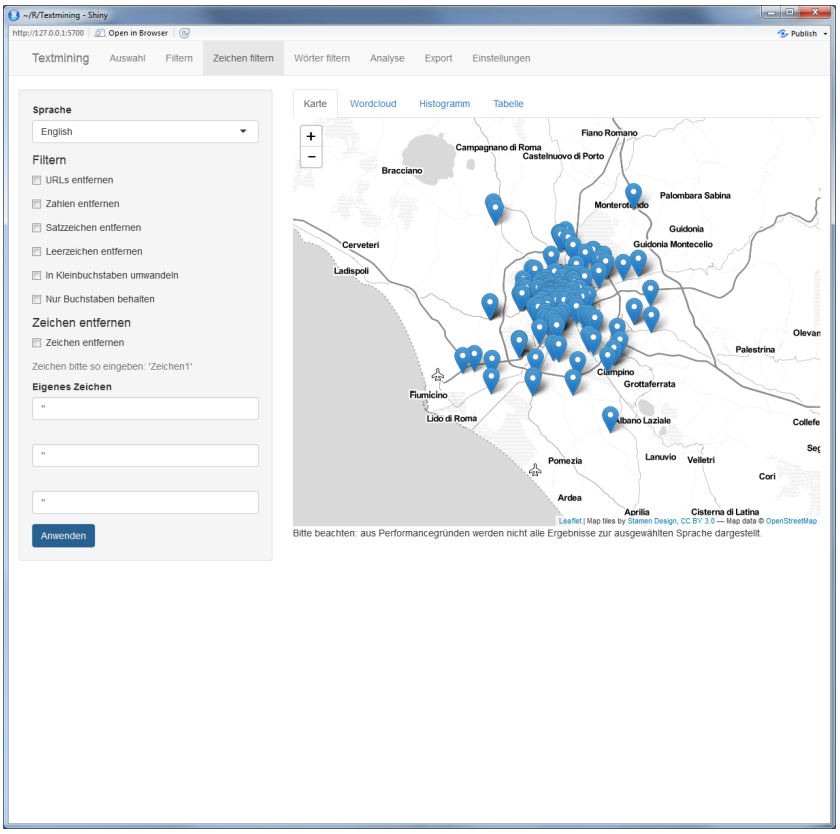

Abbildung B.3.: Zeichen filtern

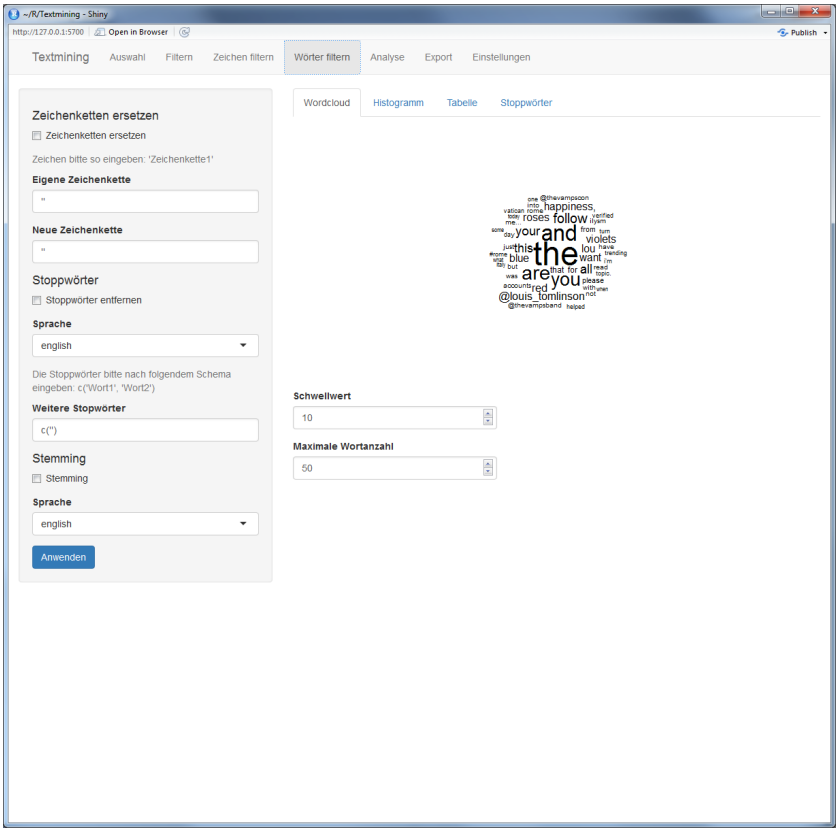

Abbildung B.4.: Wörter Filtern

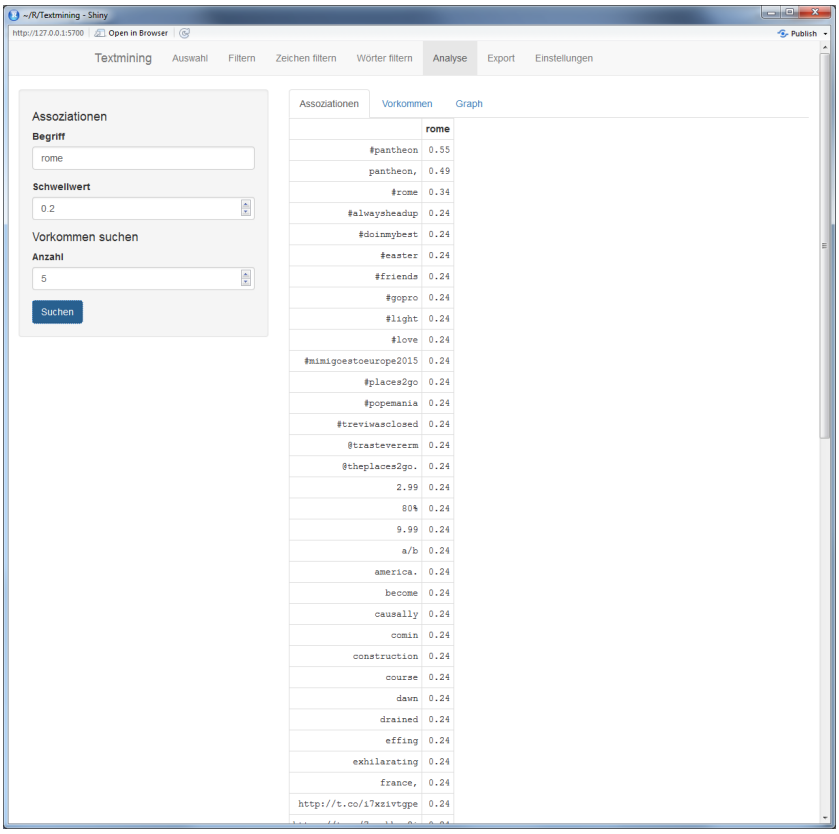

Abbildung B.5.: Analyse

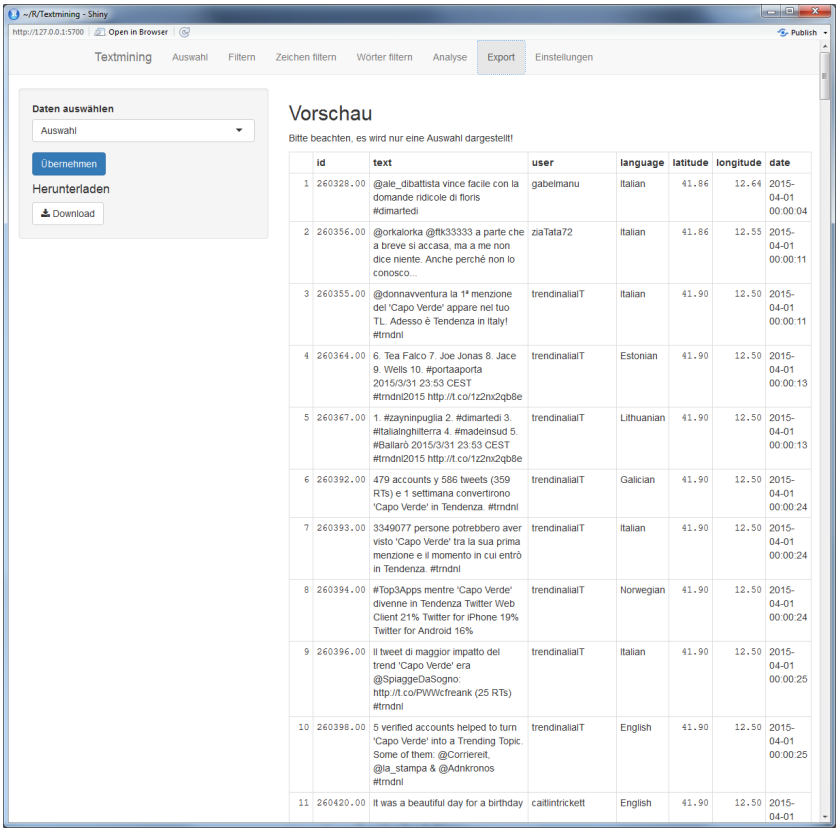

Abbildung B.6.: Export

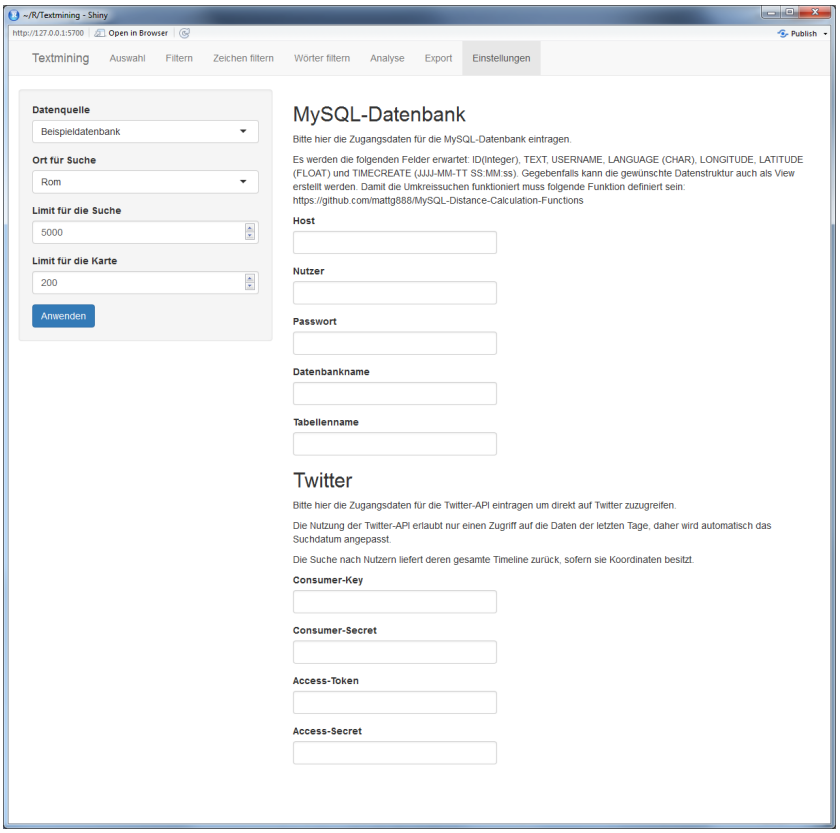

Abbildung B.7.: Einstellungen

C. Tabellen

Tabelle C.1.: Tabelle mit allen in Prototypen verwendeten R-Pakten. Sie können unter `https://cran.r-project.org/` nachgeschlagen oder heruntergeladen werden.

Name	Version	direkte Nutzung	Verwendung
assertthat	0.1		
base64enc	0.1		
BH	1.58.0		
BiocGenerics	0.15.6		
bit	1.1		
bit64	0.9		
bitops	1.0		
caTools	1.17.1		
colorspace	1.2		
curl	0.9.1		
DBI	0.3.1		
dichromat	2.0		
digest	0.6.8		
dplyr	0.4.2	✓	Datenfilterung
DT	0.1	✓	Tabellen
ggplot2	1.0.1	✓	Diagramme
graph	1.47.2		
gridBase	0.4		
gtable	0.1.2		
htmltools	0.2.6		
htmlwidgets	0.5		
httpuv	1.3.2		
httr	1.0.0		
jsonlite	0.9.16		

labeling	0.3		
lattice	0.20		
lazyeval	0.1.10		
leaflet	1.0.0	✓	Karten
magrittr	1.5		
markdown	0.7.7		
mime	0.3		
munsell	0.4.2		
NLP	0.1		
packrat	0.4.4		
plyr	1.8.3		
png	0.1		
proto	0.3		
R6	2.1.0		
raster	2.4		
RColorBrewer	1.1		
Rcpp	0.12		
reshape2	1.4.1		
Rgraphviz	2.13.0	✓	
rjson	0.2.15		
RJSONIO	1.3		
RMySQL	0.10.3	✓	Datenbankanbindung
RSQLite	1.0.0		
scales	0.2.5		
shiny	0.12.1	✓	GUI
slam	0.1		
SnowballC	0.5.1	✓	Stemming
sp	1.1	✓	Geofunktionen
stringi	0.5		
stringr	1.0.0		
tau	0.0		
textcat	1.0	✓	Spracherkennung
tm	0.6	✓	Text Mining
twitteR	1.1.8	✓	Twitter API
wordcloud	2.5	✓	Wort Wolken
xtable	1.7		
yaml	2.1.13		

Tabelle C.2.: Anforderungen an die zu entwickelnde Software und ihre Umsetzung.

Anforderung	Priorität	Umsetzung
Auswahl		
Zeit	1	✓
Ort	1	✓
Thema	1	✓
Nutzer	1	✓
Aufbereitung		
Erkennung der Sprache	1	✓
URLs entfernen	1	✓
Umwandlung in Kleinbuchstabe	1	✓
Zahlen entfernen	1	✓
Sonderzeichen entfernen	1	✓
Satzzeichen entfernen	1	✓
Auswahl einer Sprache	1	✓
Stoppwörter entfernen	1	✓
Stemming	2	✓
Nutzer filtern	2	✓
Visualisierung		
Tabelle	1	✓
Kartendarstellung	1	✓
Histogramm	1	✓
Wortwolke	1	✓
weitere Anforderungen		
graphische Benutzeroberfläche	1	✓
Export der Daten	1	✓
Interaktie Bearbeitung	2	✓
austauschbare Datenquelle	2	✓
Zugriff auf aktuelle Daten	2	✓
einfache Installation	3	*