INSTITUT FÜR INFORMATIK
Datenbanken und Informationssysteme

Universitätsstr. 1 D–40225 Düsseldorf

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

Themenerkennung in Twitter

Joseph Cornelius

Bachelorarbeit

Beginn der Arbeit:	28. Mai 2015
Abgabe der Arbeit:	5. August 2015
Gutachter:	Prof. Dr. Stefan Conrad
	Jun.-Prof. Dr. Dorothea Baumeister

Erklärung

Hiermit versichere ich, dass ich diese Bachelorarbeit selbstständig verfasst habe. Ich habe dazu keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Düsseldorf, den 5. August 2015

_____

Joseph Cornelius

Zusammenfassung

Bei der Verbreitung von Nachrichten gewinnt die Mikroblogging-Plattform Twitter immer mehr an Bedeutung. Dabei werden Hashtags, eine spezielle Form von Schlagwörtern, benutzt, um besonders populäre Themen zu verfolgen. Die von den Benutzern geposteten Kurznachrichten, sogenannte Tweets, hindern etablierte Themenerkennungsverfahren daran ihre volle Wirkung zu entfalten. Dies ist auf ihre Kürze und den oftmals unkonventionellen Sprachgebrauch zurückzuführen.

Im Rahmen dieser Bachelorarbeit wurden mehrere Twitter spezifische Themenerkennungsverfahren in Java implementiert. Hierbei wurden entgegen klassischer Verfahren nicht nur Substantive als Themen bildende Wörter verwendet, sondern verschiedene Wortarten zum Vergleich benutzt. Anschließend wurden die erstellten Verfahren mit zwei beliebten Themenerkennungsverfahren verglichen, der Latent Dirichlet Allocation (**LDA**) und der nicht-negativen Matrix-Faktorisierung (**NMF**). Der Vergleich fand anhand eines selbst erstellten Datensatzes zu 15 trendigen Hashtags statt.

Bei der Evaluation hat sich gezeigt, dass NMF nach intensiver Selektierung der Wörter der Tweets mit Part-of-Speech (**POS**) Tags, Dependenzen und Stoppwort-Listen gute Ergebnisse liefert. Allerdings wurden die Ergebnisse der LDA-Methode von einer Kombination aus LDA mit N-Grammen, das sind Textzerlegungen, die die Textstruktur berücksichtigen, übertroffen. Außerdem konnte festgestellt werden, dass die verschiedenen Datensätze einen Einfluss auf die Ergebnisse einzelner Verfahren haben.

Einleitend wird die Motivation und die Zielsetzung für die Arbeit erläutert. Im zweiten Kapitel werden die Grundlagen für die Themenerkennung erklärt. Als nächstes werden im dritten Kapitel externe Programme und die Helper-Methoden vorgestellt. Darauf aufbauend werden im vierten Kapitel die einzelnen Methoden zur Erkennung von Themen dargestellt. Anschließend wird die Evaluation der Methoden vollzogen und der dazu verwendete Datensatz beschrieben. Im sechsten Kapitel wird der aktuelle Stand der Forschung vorgestellt und Bezüge zu dieser Arbeit werden hergestellt. Abschließend wird das Fazit der Bachelorarbeit gezogen und ein Ausblick auf mögliche zukünftige Arbeiten gegeben.

Inhaltsverzeichnis

1 Einleitung

In diesem Kapitel wird die gegenwärtige Relevanz des Themas verdeutlicht. Dazu wird deren Motivation und die Zielsetzung vorgestellt.

1.1 Motivation

Über das heutige Zeitgeschehen wird immer häufiger mit Hilfe von sozialen Medien berichtet. Diese ermöglichen es, aktuellen Themen fast ohne Zeitverzögerung zu folgen. Deshalb wird der Auswertung von Texten auf den sozialen Plattformen eine stetig wachsende Bedeutung zugeschrieben. Twitter[1], eine der größten Mikroblogging-Plattformen mit durchschnittlich 302 Millionen aktiven Nutzern im Monat, ist dabei aufgrund seiner extremen Zeitnähe zum realen Geschehen besonders relevant.[2]
Benutzer von Twitter können sogenannte Tweets posten. Um allgemeine Themen zu klassifizieren werden Schlagwörter zur Vereinfachung der Suche benutzt, sogenannte *Hashtags* [Kau10]. Hierzu fügen die Benutzer Terme mit einem vorangestellten Rautezeichen (#) in ihre Posts ein [RDL10].
Bei außergewöhnlichen Ereignissen werden schnell mehrere Tausend Tweets zu den Geschehnissen gepostet. Diese Trends auf Twitter werden durch häufig auftretende Hashtags sichtbar. Jedoch wird bei Hashtags, wie beispielsweise *#wwdc15* oder *#got*, oft nicht ersichtlich, was das eigentliche Thema des Ereignisses ist.
Eine weitere Komplikation in der Auswertung von Tweets entsteht durch deren besondere Form, wie in Abbildung 1 ersichtlich wird.

OMG! #got 💘💥 Best #GameofThrones Ep ever!!! ⛄❄🐴 http://t.co/WzCd6Rp

Abbildung 1: Beispiel Tweet

Tweets können nicht mehr als 140 Zeichen enthalten. Wegen dieser Verknappung der Nachrichtenlänge haben die Nutzer unkonventionelle Methoden erfunden, um die Nachricht zu komprimieren und somit mehr Inhalt in einen Post zu bekommen. Beispielsweise werden Wörter durch Abkürzungen oder sogar Emoticons ersetzt.
Dies hat jedoch zur Folge, dass es sich bei Tweets nicht mehr um übliche, grammatikalisch und sprachlich korrekte Texte handelt. Gängige Themenerkennungsverfahren gehen aber gerade von sprachlich korrekten Texten aus.

1.2 Zielsetzung

Ziel dieser Arbeit ist es, ein Verfahren zu entwickeln, das es ermöglicht, das Thema eines und das Gesamtthema aller Tweets trotz derer Besonderheiten zu ermitteln. Dazu wurden ausschließlich Tweets aus dem englischsprachigen Raum analysiert. Zur Findung des besten Verfahrens wird ein Vergleich verschiedener Methoden herangezogen.

[1] www.twitter.com
[2] https://investor.twitterinc.com/releasedetail.cfm?ReleaseID=909177

2 Grundlagen

In diesem Kapitel wird der Begriff der Themenerkennung, das Verfahren des POS Taggings, das WordNet und die Lemmatisierung von Wörtern erläutert. Außerdem wird auf die Problematik des Begriffes „Thema“ eingegangen.

2.1 Themenerkennung

Themenerkennung (engl. *topic detection*) ist ein Gebiet aus dem Information Retrieval, der Suche nach relevanten Inhalten [Car01]. In der klassischen Themenerkennung werden aus Dokumenten mit inhaltlich verschiedenen Themen die einzelnen Themen herausgearbeitet [Sto07].
Durch den Hashtag oder das Suchwort, die als inhaltlicher Selektor der Tweets fungieren, findet meist ein Clustering der Tweets bezüglich eines allgemeinen Themas statt. Im Gegensatz zur klassischen Themenerkennung ist die Aufgabe der Themenerkennung in dieser Arbeit, möglichst treffend das Thema eines Tweets und das Gesamtthema aller Tweets zu bestimmen.

2.2 Das Problem des Themas

Die fast trivial wirkende Frage nach dem **Thema** stellt sich bei genauer Betrachtung als nur schwer beantwortbar heraus. Wayne Xin Zhao definiert das Thema als „Subjekt das in einem oder mehreren Dokumenten diskutiert wird“ [ZJW+11].
Im Gegensatz dazu macht Teun van Dijk deutlich, dass das Thema keineswegs so einfach bestimmt werden kann. Er unterscheidet hierbei das „Satz Thema“ vom „Diskurs Thema“.
Das Thema ist bei beiden die Antwort auf die Frage „Um was geht es?“. Ein Satz Thema besteht nur aus Substantiven. Jedoch ist beim Satz *„Peter fährt zur Schule“* beispielsweise noch nicht geklärt, ob Peter, Schule oder Peter und Schule das Thema sind. Hierzu müsste der Kontext herangezogen werden.
Bei Diskursen besteht das Thema nicht mehr nur aus einem Substantiv, sondern aus einer „Makro-Struktur“ [Dij77]. Diese ist eher ein zusammenfassender Satz und ermöglicht dadurch ein informativeres Gesamtbild [Dij77].
Diese unterschiedlichen Erklärungen von *„Thema“* verdeutlichen, dass es keine einheitliche Definition von *„Thema“* gibt. In dieser Arbeit soll durch die Bestimmung des Themas - vom Inhalt der Tweets - ein Verständnis oftmals kryptischer Hashtags erreicht werden. Deshalb wird weniger darauf Wert gelegt, ob es sich um Substantive handelt, sondern ob der Term oder die Wortart Informationen liefern.

2.3 Part-of-Speech Tagging

Die Grundeinheit eines Textkorpus bilden Zeichenketten, sogenannte *Token*. Part-of-Speech (**POS**) Tagging ist ein Verfahren, um die Wortart eines Tokens zu bestimmen. Zu jedem Token wird ein Tag aus einem vorher definierten Tagset bestimmt. Hierbei werden

den unterschiedlichen Tags Wortarten, wie Adjektiv und Substantiv, aber auch Nicht-Wortformen, wie Zahlen oder Interpunktion, zugeordnet [Car01].
Da Wörter existieren, die zu mehreren Wortarten gehören, wie zum Beispiel das Wort *„licht“*, das sowohl Substantiv, als auch Adjektiv sein kann, gilt es derartige Ambiguitäten zu vermeiden.
Gute *Tagger*, die Werkzeuge des POS Tagging, erreichen eine Treffsicherheit von 97%, wobei die Obergrenze der menschlichen Treffsicherheit bei 98% liegt.[3]

2.4 WordNet

WordNet [Mil95] ist eine lexikalische Datenbank für englische Wörter. Diese beinhaltet in Verben, Substantive, Adjektive und Adverbien kategorisierte *Synsets*, Mengen von synonymen Wörtern. So ein Synset lautet zum Beispiel {*auto, car, automobile*}. *WordNet* enthält somit nicht nur lexikalische, sondern auch semantische Beziehung von Wörtern [Mil95].

2.5 Lemmatisierung

Ein Problem beim Arbeiten mit Sprache ist, dass Wörter mit gleicher Semantik in unterschiedlichen Wortformen (z.B. *Versuche, Versuch, Versuches*) auftreten. Um Wörter mit gleicher Bedeutung, aber unterschiedlicher Wortform, zu finden, führt man diese auf ihren Wortstamm oder ihr *Lexem*, die Grundform, zurück. Im Gegensatz zum Wortstamm muss das Lexem eine linguistisch gültige Form sein. Beispielsweise wäre für die Wörter *„bittet“* und *„bittest“* das Lexem *„bitten“*, wohingegen der Wortstamm *bitt* ist. Um nur linguistisch gültige Wörter zu erhalten, wird im Rahmen dieser Arbeit eine **Lemmatisierung**, sprich die Rückführung des Wortes auf dessen Lexem, verwendet [Sto07].
Die *Flexion*, wie beispielsweise *versuch-en, versuch-st, versuch-te*, ist eine Art der Wortformbildung [Car01].
Bei einer wörterbuchbasierten Lemmatisierung wird eine Überschmelzung vermieden, d.h. das Wortformen wie *„versuche“* nicht sowohl auf *„Versuch“*, als auch auf *„versuchen“* zurückgeführt werden. Dies geschieht, indem das Wörterbuch alle Lexeme einer Sprache und deren Flexionsform in Substantive, Verben, Adjektive und Adverbien kategorisiert. *WordNet* ist ein solches Wortverzeichnis [Sto07].

[3] https://class.coursera.org/nlp/lecture/150

3 Helper-Methoden und Programmierschnittstellen

Im Rahmen dieser Bachelorarbeit wurden zwei Programme implementiert. Ein Wrapper für ein Twitter Application Program Interface (**API**) und eines, das mit unterschiedlichen Verfahren das Thema des jeweiligen Datensatzes bestimmt.
Zur besseren Verarbeitung der Tweets wurden externe Programme verwendet, die im Folgenden erläutert werden. Daran anschließend wird die Implementierung erklärt.

3.1 Twitter-Crawler

Der **Twitter-Crawler** erstellt mit Hilfe der trendigen Hashtags Dokumente, die aus den dazu gesammelten Tweets bestehen.
Als Programmierschnittstelle zu Twitter wird *Twitter4J*[4] verwendet, das eine Abfrage der letzten Tweets zu einem bestimmten Term ermöglicht. Bei der Abfrage werden nur die von Twitter als Englisch markierten Tweets verarbeitet. Anschließend werden alle Retweets, unverändert übernommene Tweets anderer Benutzer, und doppelte Tweets des gleichen Benutzers herausgefiltert.
Die Twitter API erlaubt nur maximal 100 Tweets pro Suchanfrage und maximal 180 Suchanfragen innerhalb von 15 Minuten. Da bei den ersten Versuchen allerdings vor jeder Filterung bis zu $79{,}52\%$ Retweets enthalten waren und die erwünschte Kollektionsgröße 4000 Tweets beträgt, läuft der Crawlvorgang mehrmals hintereinander mit Pausen von 15 Minuten ab. Anschließend werden die Tweets in einer Textdatei gespeichert.

3.2 Tweebo-Parser

Ein *dependency parser* erkennt die Dependenzen, das sind die Abhängigkeiten von Wörtern zueinander. Der **Tweebo-Parser** [KSS+14] von Noahs Ark bietet einen dependency parser für Tweets, der gleichzeitig einen *POS Tagger*[GSO+11] beinhaltet.
Der auf Twitter spezialisierte POS Tagger erkennt neben Wortarten und Interpunktion auch Hashtags (Tag #), vorgestellte @ (Tag @), URLs (Tag U), Emoticons (Tag E) und fremdsprachige oder keinen Sinn ergebende Wörter (Tag G) wie zum Beispiel *„gggrrhhh“* [OOD+13]. Die Tabelle 2 im Anhang bietet eine Übersicht aller Tags. Mit Hilfe von *Tweebo-Parser* werden die Tweets auf folgende Weise getaggt:

Tag :	A	N	N	N	,	V	O	,	#
Token :	Happy	fathers	day	dad	,	love	you	!	#love

In Abbildung 2 werden die Dependenzen von Wörtern veranschaulicht. Diese zeigt, dass ausgehend von den Roots, die einzelne Satzsegmente markieren, sich aufeinander beziehende Wörter gefunden werden. Zum Beispiel wird im Satzteil „want to have his babies“ die Abhängigkeit vom Verb aus wie folgt aufgezeigt: *want -> to -> have -> babies -> his*.

Auch werden multi word expressions (**MWEs**), wie „LA Times“, als Beziehung von *Times* zu *LA* erkannt.

[4] http://twitter4j.org/en/index.html

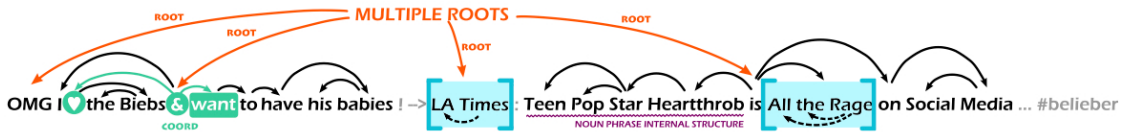

Abbildung 2: Dependenzen [KSS+14]

Der *HEAD* bezeichnet die Abhängigkeit eines Token zu einem anderen Token. Der Prozentsatz der Tokens mit einem korrekten *HEAD* ist der *unlabeled attachment score* [NS04]. *Tweebo-Parser* arbeitet mit einem *unlabeled attachment score* von bis zu 80,1% [KSS+14].

3.3 Java WordNet Interface

Nachdem die Abhängigkeiten und die Art der Wörter bestimmt sind, gilt es deren Wortform zu vereinfachen. Das *Java WordNet Interface* (**JWI**) [Fin14] ist eine Java API für *WordNet* und bietet eine Lemmatisierung mit Hilfe von *WordNet* an.
Im Rahmen dieser Bachelorarbeit wird die von *Tweebo-Parser* erstellte *CoNLL-Datei* tokenweise ausgelesen und mit Hilfe der durch die POS Tags bestimmten Wortart und dem *WordNetStemmer* des *JWI* lemmatisiert, sofern es dafür einen Eintrag in *WordNet* gibt. Gibt es diesen nicht, wird das Ursprungswort verwendet. So wird beispielsweise *„resigned“* zu *„resign“* und *„champions“* zu *„champion“*.

3.4 Implementierung

Sowohl der *Twitter-Crawler* als auch das Hauptprogramm **Topic-Recognition** wurden in der Programmiersprache *Java* geschrieben. Beide sind reine Kommandozeilen-Programme. Der *Twitter-Crawler* erstellt einen Datensatz aus Tweets, der anschließend vom *Topic-Recognition* Programm auf das Thema der Tweets analysiert wird.
In Abbildung 3 wird der Ablauf der beiden Programme veranschaulicht.

Zuerst startet das Programm Topic-Recognition über die Eingabe von:

```
--run --tag BeispielHashtag --method BeispielMethode
```

Der als Kommandozeilenargument übergebene Hashtag wird an den Twitter-Crawler übergeben, wie in Abbildung 3 im ersten Schritt ersichtlich wird.

Die vom Twitter-Crawler erstellte Textdatei wird, wie nach Schritt 2 gezeigt, vom *Tweebo-Parser* geparst. Dabei wird für jedes Token in jedem Tweet das POS Tag und die Dependenzen bestimmt. Diese für alle Tweets gesammelten Informationen werden vom *Tweebo-Parser* in einer *CoNLL-Datei* [BM06] gespeichert.

Nach dem dritten Schritt ist erkennbar, dass nach der Benutzung von *JWI* die Dependenzen, POS Tags und lemmatisierten Wörter gespeichert werden. Im Topic-Recognition Programm wird eine Liste aus allen Tweets, zu jedem Tweet eine Liste aus allen Sätzen und zu jedem Satz eine Liste aus allen Tokens erstellt. Hierbei wird neben den POS Tags, Dependenzen und der lemmatisierten Form jedes Wortes auch noch die Position des Tokens im Satz und die Satznummer gespeichert.

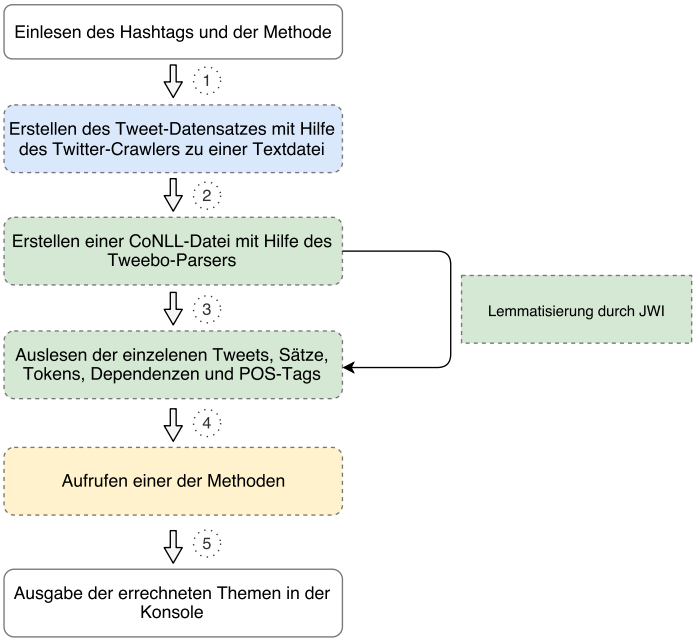

Abbildung 3: Ablauf vom Topic-Recognition Programm mit Twitter-Crawler

Die Listen mit den gesammelten Informationen über den kompletten Datensatz zu einem Hashtag werden an die unter `-- method BeispielMethode` bestimmte Methode weitergegeben, wie im vorletzten Schritt zusehen ist.

4 Methoden zur Themenerkennung

Mit Hilfe der Listen wird von den verschiedenen Methoden eine Themenerkennung vorgenommen. Im folgenden Kapitel werden die unterschiedlichen Methoden zur Gewinnung des Themas vorgestellt.

4.1 Count Methoden

Eine **Bag-of-Words** ist eine vereinfachte Darstellung des Textes, eine ungeordnete Sammlung der im Text enthaltenen Tokens. Hierbei wird die Häufigkeit der einzelnen Tokens mit Hilfe eines Vektors bestimmt, wobei der Vektor nicht alle im Text auftretenden Token enthalten muss [JM08]. Da alle Tweets in einer einzigen Datei gesammelt sind, wird hier nur ein Textdokument betrachtet. Eine *Bag-of-Words* mit dem Vektor

$$[\textit{Alter, Haus, Liebe, vor, aber}]$$

über den Satz *„Alter schützt vor der Liebe nicht, aber Liebe vor dem Altern“*, wäre

$$[1, 0, 1, 2, 1].$$

Hierbei steht die erste 1 für Häufigkeit des Wortes Alter, die nachfolgende 0 steht für die Häufigkeit des Wortes Haus und für die restlichen Wörter funktioniert es analog.

4.1.1 Counter-Methode

Die zentrale Methode ist die **Counter-(C-)Methode**. Die C-Methode sucht nach am häufigsten auftretenden Token im Dokument und verfolgt dabei einen *Bag-of-Words* Ansatz. Für jeden Tweet werden alle in den Listen gespeicherten Token durchlaufen. Sie werden alle (bis auf die Interpunktion) in einer Hashtabelle abgelegt, sodass zu jedem Token die entsprechende Anzahl erkenntlich wird. Von dieser *Bag-of-Words* werden die 20 häufigsten Token ausgegeben.

4.1.2 StopWordCount-Methode

Das Problem von häufigen Wörtern wie *„the“*, *„to“* oder *„if“* ist, dass sie keinen großen Informationsinhalt liefern. Diese Wörter nennt man Stoppworte [MS99].
Der **StopWordCount-(SWC-)Methode** liegt die C-Methode zugrunde. Anders als die C-Methode verwendet sie eine englische Stoppwörter-Liste[5]. Bei der SWC-Methode wird jedes Token, das ein Stoppwort ist, herausgefiltert bevor es an die Hashtabelle gegeben wird.

[5] http://xpo6.com/list-of-english-stop-words/

4.1.3 LemCount-Methode

Die **LemCount-(LC-)Methode** basiert auf der SWC-Methode. Da sich, wie in Abschnitt 2.5 beschrieben, Wörter mit gleicher Semantik oft in der Form unterscheiden, verwendet die LC-Methode die lemmatisierte Form der Token. In der Liste der Token und Zusatzinformationen befindet sich die lemmatisierte Form zu jedem Wort. Sofern sie nicht existiert, werden die ursprünglichen Token verwendet. So wird beispielsweise statt *„felt“* das Lemma *„feel“* an die Hashtabelle übergeben.

4.2 Verwendung von Part-of-Speech Tags und Dependenzen

In den folgenden Methoden werden zusätzlich zu den Verfahren von der LC-Methode die POS Tags und Dependenzen in Betracht gezogen.
Um Eigennamen wie *„Sepp Blatter“* nicht als getrennte Token, sondern als einen Token aufzufassen, verwenden die folgenden Methoden die extrahierten Dependenzen.

4.2.1 NounPOSCount-Methode

Eine häufige Annahme ist, wie in 2.2 erwähnt, dass die Substantive ausschlaggebend für das Thema des Satzes sind. Die **NounPOSCount-(NC-)Methode** filtert aus den Tweets die Substantive, das heißt die Gattungsnamen (Tag N), Eigennamen (Tag ^), besitzanzeigenden Gattungsnamen (Tag S) und besitzanzeigenden Eigennamen (Tag Z), sowie die Zahlen (Tag $) [OOD$^+$13]. Die mit Hilfe von Tweebo-Parser gefundenen Tags zu jedem Token werden ebenfalls aus der List der Token abgerufen. Dadurch gelangen nur als Substantive erkannte Wörter in die *Bag-of-Words*.

4.2.2 VerbPOSCount-Methode

Obwohl das Prädikat generell als der Bestandteil des Satzes gesehen wird, der einen Kommentar gibt, sprich zum Thema den Kontext beschreibt, wird mit der **VerbPOSCount-(VC-)Methode** der Versuch unternommen, allein durch das Aufzeigen von häufigen Handlungen, den Inhalt zu bestimmen [Dij77]. In dieser Methode werden nur Verben benutzt. Diese werden über den POS Tag *V* gefiltert [OOD$^+$13]. Hinzu kommt die Benutzung einer besonderen Stoppwörter-Liste, die mehr allgemeine Verben beinhaltet als die bisher genutzte Stoppwörter-Liste.

4.2.3 NounVerbPOSCount-Methode

Die **NounVerbPOSCount-(NVC-)Methode** verbindet die NC-Methode und die VC-Methode. Beispielsweise würde der Tweet *„#AgeOfUltron is another masterpiece!!!“* mit der NVC-Methode auf {*#AgeOfUltron, is , masterpiece*} reduziert werden.

4.2.4 FirstSentenceCount-Methode

Die Absatzstruktur ist ein Merkmal von inhaltlich relevanten Sätzen, die sich seit Edmunson [Edm69] bewährt hat. Simplifiziert dargestellt ist sie die Erkenntnis, dass der Informationsgehalt eines Satzes von der Lage innerhalb des Textes abhängt [Car01].
In Anlehnung an die Absatzstruktur wird mit der **FirstSentenceCount-(FSC-)Methode** von der These ausgegangen, dass der erste Satz eines Tweets im Mittel den größten Informationsgehalt besitzt. Daher wendet die FSC-Methode die Verfahren der NVC-Methode auf den jeweils ersten Satz eines Tweets an.
Beim Auslesen der Tweets aus der CoNLL-Datei wird jeder Tweet in seine Sätze aufgeteilt. Dies geschieht anhand der Interpunktion und der Tags der einzelnen Token. Zu jedem Satz wird die Satznummer, bezogen auf den Tweet, in den Zusatzinformationen gespeichert. Diese Satznummer wird hier zum Filtern der Sätze benutzt.
So wird zum Beispiel der Teiltweet *„#Marvel The Avengers are back!“* anstatt *„#Marvel The Avengers are back! Gear up with 20% off all #Avengers #AgeOfUltron merch from @cafepressinc (CafePress... http://t.co/doNI8AoXSP)“* zur Auswertung verwendet.

4.3 N-Gramme und NGramCount-Methode

Um der Struktur von Texten oder Wörtern eine höhere Bedeutung beizumessen, werden *N-Gramme* verwendet. Diese zeichnen sich durch die Betrachtung n aufeinanderfolgender Token aus, anstelle eines einzelnen Tokens. Dies ermöglicht die Abhängigkeiten innerhalb eines Textes besser darzustellen [Sto07].
In der folgenden Auflistung ist der Beispielsatz *„great to see #fifa unholy empire begin to crumble“* mit den Werten $n \in \{1, \ldots, 6\}$ dargestellt.

Name	**N**	**Beispiel**
Monogramm	1	great, to, see, #fifa, unholy, empire, …
Bigramm	2	great to, to see, see #fifa, #fifa unholy, …
Trigramm	3	great to see, to see #fifa, see #fifa unholy, …
Tetragramm	4	great to see #fifa, to see #fifa unholy, …
Pentagramm	5	great to see #fifa unholy, to see #fifa unholy empire, …
Hexagramm	6	great to see #fifa unholy empire, …

Die **NGramCount-(NGC-)Methode** sammelt *N-Gramme* in einer *Bag-of-Words*. Die NGC-Methode verwendet als Defaultwert $n = 3$.
In N-Grammen können keine Abhängigkeiten von Wörtern die mehr als $n - 2$ Wörter auseinander liegen dargestellt werden. Es gilt also die Distanz von informativeren Token zu verringern. Deshalb werden mit Hilfe der POS Tags alle Wortarten bis auf Adjektive, Zahlen, Verben, Eigennamen und Gattungsbegriffe, sowie deren possessive Formen herausgefiltert. Allerdings werden keine Dependenzen zum Clustering der Eigennamen benutzt.

4.4 Latent Dirichlet Allocation

Eines der populärsten unüberwachten maschinellen Lernverfahren zur Themenerkennung ist die *Latent Dirichlet Allocation* (**LDA**) [BNJ03] von Blei et al. [RDL10].
LDA ist darauf spezialisiert, verborgene Themen in einer großen Sammlung von Dokumenten zu finden. Hierbei wird jedes Dokument als eine *Bag-of-Words* angesehen. Ein Thema ist eine Mischung aus Wörtern. Jedem Wort wird dabei eine Wahrscheinlichkeit zugeordnet, mit der es zum Thema gehört. *LDA* beruht auf der Annahme, dass ein Dokument eine Mischung aus Themen ist.
Gegeben einer bestimmten Anzahl von Themen und der Wahrscheinlichkeiten der in ihnen enthaltenen Wörter können mit dem folgenden Prozess für jedes Dokument in der Sammlung die Themenzuordnungen der Wörter generiert werden [Ble12]:

1. Wähle zufällig eine Themenverteilung
2. Für jedes Wort im Dokument
   a) Wähle zufällig ein Thema von der Themenverteilung in Schritt 1.
   b) Wähle zufällig ein Wort von der zum Thema in Punkt a) korrespondierenden Wortverteilung

Aus der Themenzuordnung der Wörter folgen die im Dokument enthaltenen Themen. Dieser Prozess setzt aber, wie erwähnt, gegebene Themen und deren zugehörige Wörter voraus. Da die Themenzuordnung aller Wörter aber nicht gegeben ist, gilt es die A-posteriori-Verteilung der Themenordnung zu bestimmen. Generell wird nur die approximative A-posteriori-Verteilung benutzt [BL09].

Gibbs Sampling ist ein Verfahren zur Berechnung der approximativen A-posteriori-Verteilung. Hierbei wird eine Markov Kette konstruiert. Das ist eine Sequenz von Zufallsvariablen, die vom Vorgänger abhängig sind. *Gibbs Sampling* ist ein iterativer Prozess. Bei jeder Iteration werden folgende Stufen durchlaufen [Ble12, Gri04]:

Beim ersten Durchlauf wird zufällig für jedes Wort eine Themenzuordnung bestimmt.

- Für jedes Dokument
  - Für jedes Wort im Dokument
    - Bestimme die Themenzuordnung des Wortes neu anhand der aktuellen Wahrscheinlichkeit des Wortes in allen Themen und anhand der aktuellen Verteilung der Themen in dem Dokument

Hierbei wird in Abhängigkeit von allen anderen Zuordnungen für jedes Wort (sample) die Themenzugehörigkeit berechnet. Dies geschieht solange, bis die aus den samples errechnete A-posteriori-Verteilung konvergiert [Ble12].

Durch die zufällige Themenverteilung ist LDA mit Gibbs Sampling keine deterministische Methode. Außerdem hängt das Ergebnisse stark von der Anzahl der Iterationen und der gewählten Themenanzahl ab.

Nachfolgendes Beispiel dient der Veranschaulichung des LDA Verfahren unter Verwendung von Gibbs Sampling. Die linke Tabelle zeigt ein beliebiges Dokument i. Dabei ist jedem Wort des Dokuments ein Thema zugeordnet. Die rechte Tabelle zeigt alle Worte aus allen Dokumenten und deren Zuordnung zu den jeweiligen Themen. Das Wort Auto beispielsweise kommt in allen Dokumenten zusammen 13 mal vor und wird in Schritt 1 12 mal dem Thema 1 und einmal dem Thema 2 zugeordnet.

Schritt 1**:**
Zunächst wird die Themenzuordnung zufällig gewählt. In diesem Beispiel mit $k = 3$ Themen.

Thema:	1	1	3	2
Dokument i:	Auto	Rennbahn	Monitor	Reifen

	Thema 1	*Thema 2*	*Thema 3*
Auto	12	1	0
Monitor	2	0	40
Reifen	15	3	0
Rennbahn	3	0	2

Schritt 2:
Nun wird für Dokument i in der ersten Iteration das Thema des Wortes „Reifen“ neu zugeordnet. Im Dokument i ist das Thema 1 zweimal vertreten. Außerdem ist das Wort „Reifen“ dem Thema 1 insgesamt 15 mal zugeordnet. Deshalb ist die Wahrscheinlichkeit, dass „Reifen“ zu Thema 1 gehört, am höchsten.

Thema:	1	1	3	?
Dokument i:	Auto	Rennbahn	Monitor	Reifen

	Thema 1	*Thema 2*	*Thema 3*
Auto	12	1	0
Monitor	2	0	40
Reifen	15	2	0
Rennbahn	3	0	2

Schritt 3:
Nach der Zuordnung zu Thema 1 steigt die Gesamtzuordnung von “Reifen“ zum Thema 1 auf 16

Thema:	1	1	3	1
Dokument i:	Auto	Rennbahn	Monitor	Reifen

	Thema 1	*Thema 2*	*Thema 3*
Auto	12	1	0
Monitor	2	0	40
Reifen	16	2	0
Rennbahn	3	0	2

Bei einer Auswertung der Themen zu diesem Zeitpunkt könnte beispielsweise Thema 1 Motorsport, Thema 2 Fahrzeuge und Thema 3 Technik sein

Für die **LDA-Methode** wird im Rahmen dieser Arbeit die JGibbLDA[6] Implementierung verwendet, diese benutzt für die approximative A-posteriori-Verteilung *Gibbs Sampling*. Im Rahmen dieser Arbeit wird die Anzahl der zu findenden Themen auf 2 festgelegt. Wie in Abschnitt 2.1 erläutert findet durch den Such-Hashtag eine Bündelung der Tweets zu einem allgemeineren Thema statt. Die Anzahl der zu durchlaufenden Iterationen beträgt 1000. Jeder Tweet wird als ein Dokument angesehen. Um Wörter mit wenig Informationsinhalt zu filtern, wird jeder Tweet mit Hilfe der POS Tags und Stoppwort-Liste auf Verben, Substantive - (possesive) Eigennamen und Gattungsnamen - , Adjektive und Zahlen reduziert. Zudem werden alle Wörter in der lemmatisierten Version verwendet und Eigennamen mit Dependenzen als ein Token aufgefasst.

[6] http://jgibblda.sourceforge.net

4.5 Nicht-negative Matrix-Faktorisierung mit TF-IDF

Die nicht-negative Matrix-Faktorisierung (**NMF**) [LS99] ist eine Zerlegung einer nicht-negativen Matrix in ein approximatives Produkt zweier nicht-negativer Matrizen, mit Hilfe derer man eine Themenerkennung in Dokumenten vollziehen kann [XLG03].

Eine bewährte Termgewichtung ist die Kombination der dokumentspezifischen Termhäufigkeit **(TF)** mit der inversen Dokumentenhäufigkeit **(IDF)**, kurz **TF-IDF**. Bei der *TF-IDF* geht man davon aus, dass Terme einerseits an Bedeutung gewinnen, wenn sie innerhalb eines Dokumentes häufig vorkommen, aber andererseits an Bedeutung verlieren, wenn sie in vielen Dokumenten enthalten sind. [Sto07, JM08] Das TF-Gewicht lässt sich formal wie folgt beschreiben.

$$g_{t,d} = \begin{cases} 1 + log(tf_{t,d}) & \text{if } tf_{t,d} \geq 1 \\ 0 & \text{if } tf_{t,d} = 0 \end{cases} \tag{1}$$

$g_{t,d}$ ist das Termgewicht für den t-ten Term im d-ten Dokument. Die dokumentspezifische Termhäufigkeit $tf_{t,d}$ ist die Häufigkeit des t-ten Terms im d-ten Dokument. IDF wird wie folgt berechnet,

$$idf_t = log\left(\frac{N}{df_t}\right) \tag{2}$$

mit df_t der Anzahl der Dokumente, die den Term t enthalten und N der Anzahl aller Dokumente [MS99]. Mit (3) und (4) lässt dich das *TF-IDF*-Gewicht folgendermaßen beschreiben:

$$g_{t,d} = (1 + log(tf_{t,d})) * \left(log\left(\frac{N}{df_t}\right)\right) \tag{3}$$

Im Rahmen dieser Arbeit beinhaltet eine Matrix über die Dokumente D in jeder Spalte ein Dokument d_i und in jeder Zeile einen eindeutigen Term t_i aus der Sammlung aller in D vorkommenden Terme. Diese Matrix benutzt die Termgewichtung *TF-IDF* und ist somit eine nicht-negative Matrix. Sie kann nun mit *NMF* zerlegt werden.
Sei V eine $m \times n$ Matrix mit m der Anzahl der Zeilen und somit aller Wörter und n der Anzahl der Spalten und somit der aller Dokumente, dann gilt für die approximative Faktorisierung

$$V \approx WH \tag{4}$$

mit W einer $m \times r$ und H einer $r \times n$ Matrix. Hierbei kann r deutlich kleiner als m oder n gewählt werden. Dies dient der Datenreduzierung.
Die Spalten r von W werden Basisvektoren genannt. Sie enthalten pro Vektor die zu einem Thema geordneten Wörter [LS99]. Also ist r gleich der Anzahl der unterschiedlichen Themen.

In dieser Arbeit verwendet die **NMF-Methode** eine NMF Implementierung aus dem LAMF[7]-Package. Dabei ist $r = 2$. Wie in der LDA-Methode werden alle Wörter außer (possessive) Substantive, Verben, Adjektive und Zahlen aussortiert. Die selektierten Wörter werden an die TF-IDF-Methode übergeben, die wie beschrieben für jedes Wort in jedem Dokument, hier Tweet, die Termgewichtung bestimmt. Die dadurch entstandene Matrix wird an die NMF Implementierung aus LAMF weitergegeben. Aus resultierenden Basisvektoren werden jeweils die besten zwanzig Wörter ausgegeben.

[7] https://sites.google.com/site/qianmingjie/home/toolkits/laml

4.6 Kombinierte Methoden

Im folgenden Abschnitt werden zwei Methoden erläutert, die aus Kombinationen der bereits vorgestellten Methoden bestehen.

4.6.1 VerbPhraseCount-Methoden

Jeder Satz beinhaltet als Kern ein Substantiv und ein sich darauf beziehendes Prädikat. Die Idee der **VerbPhraseCount-(VPC-)Methode** ist, dass die häufigsten Konstruktionen von *Substantiv <- Prädikat* das Thema bilden. Da ein Prädikat auch eine Verbalphrase sein kann, wird in dieser Arbeit nur das darin enthaltene Verb betrachtet.
Beispielsweise wäre bei dem Satz *„Peter und Hans fahren nach Hause“* das Thema { *Peter, Hans, fahren* } .
Unter Verwendung der POS Tags und der Dependenzen werden die als Verben und gleichzeitig als Root (HEAD 0) gekennzeichneten Prädikate (-bestandteile) in einer weiteren *Bag-of-Words* gesammelt. Anschließend wird zu jedem der 20 häufigsten Verben, die in direkter Abhängigkeit stehenden Substantive, das heißt (possessive) Gattungsnamen und Eigennamen, in einer *Bag-of-Words* gesammelt. Um die häufigsten *Substantiv <- Prädikat* Konstruktionen zu finden, wird zu jedem Verb das häufigste Substantiv zugeordnet.
In der VPC-Methode werden jeweils die lemmatisierte Form der Wörter und die Eigennamen-Phrasen verwendet. Außerdem werden alle auf der Stoppwort-Liste stehenden Wörter aussortiert. Hierbei wird die für Verben spezialisierte Stoppwort-Liste verwendet.

4.6.2 N-Gramm-LDA-Methoden

Wie in Abschnitt 4.4 erwähnt ist *LDA* eines der weitverbreitetsten Verfahren zur Themenerkennung. Jedoch ist aus den Ergebnissen die Struktur einzelner Sätze nicht erkennbar. Die Idee der **N-Gramm-LDA-(NLDA-)Methode** ist es einerseits, die guten Resultate der LDA-Methode bei der Erkennung von Themen zu nutzen, und andererseits, der Abhängigkeiten von Wörtern zueinander größere Geltung zukommen zu lassen, analog der NGram-Methode.

Die NLDA-Methode kombiniert die LDA-Methode mit der NGram-Methode mit $n = 2$. Zunächst wird jeder Tweet mit den POS Tags und der Stoppwort-Liste auf die lemmatisierte From von Verben, Substantiven, Adjektiven und Zahlen reduziert. Hierbei werden keine Dependenzen zur Zusammenfügung von Eigennamen verwendet. Danach werden die Wörter in jedem Tweet durch die zu ihnen korrespondierenden Bigramme ersetzt. Anschließend werden die Tweets an die LDA-Methode übergeben, sodass jeder Tweet einem Dokument entspricht. Die LDA-Methode berechnet nun die Themen anhand der häufigsten Bigramme.

5 Evaluation

Dieses Kapitel beginnt mit einem Überklick über den verwendeten Datensatz, anschließend erläutert es die Bewertung der Vorbereitung und Helper, um mit der Auswertung der Themen einzelner Tweets und der Tweet-Datensätze zu schließen.

5.1 Datensatz

Der verwendete Datensatz wurde mit dem *Twitter-Crawler* Programm erstellt, das mit einer Java API für Twitter einzelne Tweets zu einem Hashtag zusammenträgt.
Der erstellte Datensatz umfasst 15 Textdateien und 15 CoNLL-Dateien mit Tweets zu jeweils einem Hashtag. Alle Retweets und die meisten doppelt geposteten Tweets wurden direkt beim Herunterladen herausgefiltert. Werbetweets enthalten oft URLs, die sich trotz inhaltlich gleichen Tweets verändern. Deshalb wurden zusätzlich zur automatischen Filterung gleicher Tweets auch manuell größere Ansammlungen derer beseitigt. Der Datensatz beinhaltet nur Tweets, die von Twitter als englischsprachig markiert wurden.
Die 15 verschiedenen themenbezogenen Sammlungen lassen sich den Kategorien Unterhaltung, Kultur, Politik, Sport, Technologie und Gesellschaft zuordnen, wie in Tabelle 1 ersichtlich wird. Der Datensatz wurde zu besonders stark diskutierten Suchbegriffen im Zeitraum von Mai 2015 bis Juli 2015 angefertigt.
Insgesamt enthält der Datensatz 56200 Tweets, die im Durchschnitt $12{,}12$ Wörter beinhalten. Je nach Hashtag umfassen die Sammlungen zwischen $1042 - 4972$ Tweets. Von den insgesamt 681343 Wörtern im Datensatz sind $17{,}47\%$ Hashtags.

Hashtag	Zusammenfassung	Kategorie	# Tweets
#bbking	BBKing ist gestorben	Unterhaltung, Kultur	1974
#charlestonShooting	Amoklauf in Charleston, USA	Gesellschaft, Politik	2049
#endAusterityNow	Proteste gegen die Sparpolitik in UK	Gesellschaft, Politik	4909
#fifa	Korruptionsskandal bei der FIFA	Sport, Politik	4851
#GameofThrones	GoT Staffel 5 Episode 8 Release	Unterhaltung	4595
#love	Unbestimmbar, da allzeit trendiger #-Tag	Diverse	4192
#nbafinals	NBA-Finals: Warriors siegen gegen die Cavaliers	Sport	4627
#ohNoHarry	Harry Styles fällt von der Bühne	Unterhaltung	1264
#PSYAngBatasNgApi	1 Millionen Tweets zum #PSYAngBatasNgApi	Unterhaltung	4902
#seppblatter	Sepp Blatter tritt zurück als FIFA Präsident	Gesellschaft, Politik, Sport	4972
#uswnt	Frauen-Fussballteam der USA gewinnen WM	Sport	4025
#windows10	Bekanntgabe des Windows 10 Release-Datums	Technologie	1042
#wwdc15	Apples Developer Konferenz: Apple Music	Technologie	4680
#WWEChamber	World Wrestling Entertainments: Owens vs Cane	Sport, Unterhaltung	3967
Suchbegriff: broner	Boxen: Adrian Broner verliert gegen Shawn Porter	Sport	4151
Gesamtanzahl an Wörtern:	681343	**Gesamtanzahl an Tweets:**	56200

Tabelle 1: Datensatz

5.2 Auswertung der Vorbereitung

Beim Parsen der Wörter mit dem **Tweebo-Parser** ist es schwierig, bei den in Abschnitt 3.2 erwähnten *MWEs* die Häufigkeit zu bestimmen. Die Schwierigkeit besteht darin, zur Häufigkeit der *MWEs* mit ihnen bedeutungsgleiche Wörter hinzuzurechnen. Zum Beispiel sind *„apple watch“*, *„apple music“* feste Terme, *„sepp blatter“* kann im Gegensatz dazu als *„sepp“* oder als *„blatter“* vorkommen. Um die Häufigkeit richtig darzustellen, müsste *„sepp blatter“* mit *„sepp“* und *„blatter“* verrechnet werden, wohingegen *„apple“* nicht mit *„apple watch“* verrechnet werden kann. Weiter ist nicht geklärt, ob jede Erwähnung von *„sepp“* auch *„sepp blatter“* meint. Diese Fehleranfälligkeit könnte durch eine bessere Kontextanalyse verringert werden. Im Rahmen dieser Bachelorarbeit war dies nicht möglich. Obwohl alle Emoticons über den Tag *E* herausgefiltert werden, tauchen diese in den Ergebnissen der Methoden auf. Dies liegt nicht nur an einer fehlerhaften Zuordnung, sondern auch daran, dass der *Tweebo-Parser* Emoticons teilweise als Substantiv oder Verb betrachtet, wie in Abbildung 2 veranschaulicht ist.

Beim **Twitter-Crawler** hat sich eine extreme Differenz gezeigt zwischen der Menge der Tweets, die von Twitter zum jeweiligen Hashtag gelistet wurden, und der Menge der Tweets mit dem gleichen Hashtag, die nach dem Aussortieren der doppelten Tweets eines Benutzers und der Retweets übrig blieben. Beispielsweise wurden von Twitter für *#ohNoHarry* zum Zeitpunkt des Crawlvorgangs 418000 Tweets mit diesem Hashtag gezählt, aber es konnten nur 1042 gesammelt werden. Beim Hashtag *#charlestonShooting* wurden 332000 vorhandene Tweets von Twitter angezeigt, gesammelt werden konnten nur 2049.

Es wurde festgestellt, dass in **WordNet** nach dem Filtern aller Wörter mit den *Tags* # , @,∼ , *U, E, $ und &* (für die Bedeutung siehe Tabelle 2 im Anhang) im Mittel $92{,}96\%$ der Wörter enthalten waren. Mit $95{,}67\%$ war bei *#bbking* der höchste Anteil und mit $87{,}11\%$ bei #love der niedrigste Anteil enthalten.

5.3 Auswertung des Themas einzelner Tweets

Alle im Rahmen dieser Arbeit vorgestellten Methoden zur Themenerkennung gehen davon aus, dass bestimmte Wörter mehrfach vorkommen. Bei einzelnen Tweets jedoch besteht das Problem, dass meistens alle Wörter nur einmal vorkommen. Wörter, die häufiger vorkommen, sind fast ausschließlich solche wie *„the“*, *„if“* und *„is“*. Diese beinhalten kaum Informationen zum Thema.
Aufgrund ihrer Kürze sind Tweets vergleichbar mit einzelnen Sätzen. Wie in Abschnitt 2.2 erläutert, kann bei einzelnen Sätzen die Annahme getroffen werden, dass das Thema aus den in ihnen enthaltenen Substantiven gebildet werden kann. Deshalb wird im Rahmen dieser Arbeit die NC-Methode als am geeignetsten angesehen. Allerdings kann anhand der Position im Satz ohne den Kontext nicht bestimmt werden, welche Substantive thementragend sind. Dies bedürfte einer Kontextanalyse, die in einzelnen Tweets nur schwer möglich ist.

#seppblatter

C	VC	NGC (n=2)	FSC	VPC	NMF
#seppblatter = 4822	resign = 812	sepp, blatter = 306	#seppblatter = 1364	resign (blatter) = 333 ; 49	*Thema 1:*
the = 2352	say = 241	#seppblatter, resign = 221	blatter sepp = 788	wonder (president) = 45 ; 1	
#fifa = 1975	step = 187	fifa, president = 149	resign = 652	quit (blatter) = 40 ; 8	blatter sepp = 1.52
to = 1672	think = 182	world, cup = 99	fifa president = 419	step (#seppblatter) = 35 ; 35	fifa prez = 1.15
a = 1167	know = 133	blatter, resign = 88	fifa prez = 368	leave (fifa) = 30 ; 3	fifa president = 1.13
of = 1084	come = 132	#fifa, president = 83	#fifa = 360	break (news) = 30 ; 13	america corruption = 0.52
is = 1036	quit = 125	resign, fifa = 73	resignation = 276	feel (president) = 27; 1	scandal = 0.40
for = 756	it' = 120	resign, #fifa = 64	cup qatar world = 253	hear (name) = 26; 3	abc news = 0.28
in = 725	make = 117	good, riddance = 64	step = 221	announce (#seppblatter, day) = 24 ; 4	investigation = 0.23
and = 665	need = 95	#seppblatter, resignation = 63	go = 180	watch (news) = 24 ; 2	time = 0.19
					year = 0.19
					fifa' = 0.15

SWC	NC	NGC (n=3)	LDA	NLDA	
#seppblatter = 4822	#seppblatter = 1623	sepp, blatter, resign = 74	*Thema 1:*	*Thema 1:*	*Thema 2:*
#fifa = 1975	blatter sepp = 929	resign, fifa, president = 46			
i = 656	fifa president = 542	blatter, resign, fifa = 26	#seppblatter = 0.114	#seppblatter, resign = 0.016	cup qatar world = 66.81
president = 541	fifa prez = 483	president, sepp, blatter = 24	#fifa = 0.040	world, cup = 0.007	cup final = 40.86
fifa = 480	#fifa = 456	resign, #fifa, president = 22	i = 0.031	resign, #fifa = 0.005	resignation = 33.97
blatter = 473	resignation = 363	fifa, president, sepp = 22	resignation = 0.017	resign, president = 0.005	football world = 27.13
sepp = 457	cup qatar world = 353	sepp, blatter, step = 20	cup qatar world = 0.017	good, riddance = 0.004	#seppblatter' = 17.28
resignation = 363	football world = 251	#seppblatter, resign, #fifa = 19	say = 0.011	#seppblatter', resignation = 0.004	city soccer = 12.48
world = 297	america corruption = 247	resign, allegation, corruption = 18	football world = 0.011	#seppblatter, step = 0.004	day = 11.78
resigns = 266	time = 217	#sepp, #blatter, meeting = 17	like = 0.011	#seppblatter, #fifa = 0.003	africa south = 10.00
			just = 0.011	think, #seppblatter = 0.003	time = 9.44
			think = 0.009	replace_#seppblatter = 0.002	country = 8.74

LC	NVC	NGC (n=4)	*Thema 2:*	*Thema 2:*
#seppblatter = 4822	#seppblatter = 1623	sepp, blatter, resign, fifa = 24	#seppblatter = 0.115	sepp, blatter = 0.021
#fifa = 1975	blatter sepp = 929	fifa, president, sepp, blatter = 20	#fifa = 0.054	fifa, president = 0.011
resign = 814	resign = 814	blatter, resign, fifa, president = 20	blatter sepp = 0.044	blatter, resign = 0.006
i = 656	fifa president = 542	#sepp, #blatter, meeting, say = 17	resign = 0.0388	#fifa, president = 0.006
president = 543	fifa prez = 483	#leak, #sepp, #blatter, meeting = 17	fifa president = 0.026	resign, fifa = 0.005
fifa = 480	#fifa = 456	president, resign, allegation, corruption = 16	fifa prez = 0.023	#seppblatter, resignation = 0.005
blatter = 473	resignation = 363	fifa', president, resign, allegation = 15	step = 0.013	president, resign = 0.004
sepp = 459	cup qatar world = 353	successor, elect, change, constitution = 13	america corruption = 0.011	corruption, scandal = 0.003
resignation = 363	step = 273	elect, change, constitution, set = 13	good = 0.009	#fifa, #seppblatter = 0.003
world = 297	football world = 251	nominate, mysterious, newcomer, successor = 13	#seppblatterresigns = 0.009	president, #seppblatter = 0.002

Abbildung 4: Ergebnisse aller Methoden zum Hashtag #seppblatter

5.4 Auswertung der Methoden

Die Evaluation von Verfahren zur Themenerkennung ist keine triviale Aufgabe [CYLG14]. Zunächst werden dabei vom Autor für jede Methode die besonderen Merkmale bezüglich der Themenerkennung dargestellt. Die darauf folgende Bestimmung, ob das Ergebnis einer Methode das Thema der Tweets trifft, ist jedoch subjektiv. Dies liegt einerseits daran, dass Tweets vom Auswertenden selbst zusammengefasst werden müssen, was oftmals zu Verständnis-Komplikationen führt. Andererseits liegt es in der Natur der Sprache, dass Wörter abhängig von der Person als unterschiedlich treffend empfunden werden.

Durch sechs selbst gewählte Kriterien wird in dieser Arbeit versucht, eine in sich vergleichbare Auswertung der Ergebnisse der Methoden zu vollziehen. Außerdem bemerken Chang et al. [CBGW+09] in diesem Zusammenhang, dass Themenerkennungsverfahren daran gemessen werden sollten, inwiefern sie in Realität ihrem Verwendungszweck gerecht werden.
In dieser Arbeit geht es um die Identifikation von Methoden, die das Gesamtthema der Tweets menschenverständlich ermitteln. Dies und der Versuch die Bewertung des Autors ins Verhältnis zu setzten, motivierten zu einer zusätzlichen Auswertung unter Zuhilfenahme eines von 20 Personen beantworteten Fragebogens.

Im Folgenden werden die Auswertung des Autors und die des Fragebogens dargestellt. Daran anschließend werden die Resultate der beiden Auswertungen verglichen.

5.4.1 Auswertung des Autors

Zunächst werden die Merkmale der Ergebnisse der einzelnen Methoden beschrieben. Hierbei wird zur Veranschaulichung der allgemeinen Merkmale stets auf die in Abbildung 4 dargestellten Ergebnisse Bezug genommen. Diese zeigt für jede Methode die besten 10 Ergebnisse.

Die **Counter-Methode** liefert bei allen Hashtags viele Wörter ohne großen Informationsinhalt, wie *„the“*, *„to“*, *„of“* und *„is“*.

Die **StopWordCount-Methode** trägt sehr deutlich zur Verbesserung der Filterung von Wörtern mit wenig Informationsinhalt bei. Allerdings zeigt sich, dass *„resign“* mit 266 Treffern, *„resigns“* mit 241 Treffern und *„resigned“* mit 234 Treffern jeweils die gleiche Information liefern, aber aufgrund der Aufspaltung in verschiedene Wortformen nicht unter den besten 9 Wörtern sind.

Die **LemCount-Methode** erreicht durch die Lemmatisierung eine eindeutige Verbessrung. *„resign“* ist nun mit 814 Treffern das dritthäufigste Wort. Weiter treten neue Wörter unter die besten 20, wie beispielsweise *„step“*.

Die **NounPOSCount-Methode** erzielt mit der Verwendung der Dependenzen für Eigennamen eine Steigerung der Semantik. Aus den Wörtern *„sepp“* und *„blatter“* wird *„sepp blatter“*, auch Eigennamen wie *„fifa president“* und *„qatar world cup“* werden erkannt. In der Abbildung sind die Eigennamen alphabetischer Ordnung dargestellt, um sie einfacher vergleichen zu können.

Bei der **VerbPOSCount-Methode** es ist wichtig, eine für Verben spezifizierte Stoppwörter-Liste zu benutzten. Die Handlung wird nicht ersichtlich, da die handelnden Subjekte fehlen. Im Gegensatz zu der NC-Methode ist kein klarer Zusammenhang der Wörter erkennbar.

Die **NounVerbPOSCount-Methode** lässt im Gegensatz zur NC-Methode und zur VC-Methode deutliche Rückschlüsse auf die Handlung zu. So wird zwar durch *„resignation“* in der NC-Methode auch auf den Rücktritt Sepp Blatters verwiesen, jedoch wird in der NVC-Methode mit *„sepp blatter“* an zweiter Stelle und *„resign“* an dritter Stelle klar die Handlung beschrieben. In Abbildung 5 werden die Ergebnisse der NLDA- und NVC-Methode für unterschiedliche Hashtags gezeigt. Hierbei kann ebenfalls erkannt werden, dass durch die Filterung von Verben und Substantiven bessere Rückschlüsse auf das Geschehen gemacht werden können. Dies wird bei *#fifa* an *„fifa mafia“* und *„arrest“* als auch bei *#GameofThrones* an *„episode“* und *„watch“* ersichtlich.

Die **FirstSentenceCounts-Methode** zeigt in ihren Ergebnissen kaum Unterschiede zur NVC-Methode. In Abbildung 4 wird ersichtlich, dass sich unter den besten 9 nicht einmal eine Veränderung der Termenreihenfolge vollzieht. Dieses ist insofern erstaunlich, da im Mittel $59{,}14\%$ der Tweets aus zwei oder mehr Satzkonstrukten bestehen.

Die Ergebnisse der **VerbPhraseCount-Methode** hängen, wie bei der VC-Methode, sehr stark von der Stoppwörter-Liste ab. Ein großer Nachteil ist, dass zusammengehörige Substantive und Verben gleichzeitig als *Root* vorkommen können und dadurch ihre Dependenz zueinander nicht eindeutig erkannt wird.

Die **NGramCount-Methode** wird im Rahmen dieser Arbeit mit $n \in \{2{,}3{,}4\}$ verwendet, da für $n > 4$ zu wenig vergleichbare N-Gramme vorhanden sind.
Die NGC-Methode unterstreicht die Vermutung, dass *„sepp blatter“* und *„resign“* zusammengehören, da sie zusammen in einem N-Gramm auftreten. Gerade in den ersten drei Trigrammen *(sepp, blatter, resign), (resign, fifa, president)* und *(blatter,resign,fifa)* die zusammen den Satzkonstrukt *„sepp blatter resign fifa president“* bilden wird die Stärke der N-Gramme im Aufzeigen von Textzusammenhang und Kontext sichtbar. Jedoch ist es auffällig, dass die Ergebnisse eine Großzahl an Kombinationen einer bestimmten Wortgruppe haben. Dies zeigt einerseits die für das Thema zentralen Wörter auf, aber anderseits nimmt dies den Platz für andere wichtige Wörter. Im Vergleich von Bigrammen, Trigrammen und Tetragrammen ist festzustellen, dass mit größer werdendem n das Verständnis für den Kontext wächst. Dieses Wachstum gilt aber gerade nicht für den Themenbezug. Tetragramme tendieren dazu, aufgrund ihrer sinkenden Anzahl, mehrfach auftretende Tweets und Werbetweets zu beinhalten, da diese eine gleichbleibende oder sehr ähnliche Satzstruktur haben.

Für die **NMF-**, **LDA-** und **NLDA-Methoden** konnte erkannt werden, dass die Ausgabe eines zweiten Themenvorschlags kein zweites Thema findet. Entweder waren beide Themen inhaltlich gleich, oder aber einer der Themenvorschläge bestand aus nicht im Zusammenhang stehenden Wörtern. Dies lässt vermuten, dass die Annahme richtig ist, der Datensatz zu einem trendigen Hashtag habe nur ein Thema.
Bei allen drei Methoden konnte festgestellt werden, dass sowohl die LDA-Methode als auch die NMF-Methode ohne das vorherige Auswählen der Wörter über POS Tags zu schlechten Ergebnissen kommt. Diese entstehen durch Verbindung der Twitter spezifi-

	#nbafinals	#PSYAngBatasNgApi	#fifa	#GameofThrones
NLDA	game, 1 golden, state 1, #nbafinals lebron, james win, game state, warriors stephen, curry great, game 108-100, ot 44, point	million, tweet 1, million 1.05, m m, tweet say, spammer 1m, say make, history kathniel, fan 1m, na tweet, guy	#fifa, corruption #fifa, scandal #fifa, arrest beautiful, game corruption, #fifa jack, warner fifa, corruption corruption, scandal arrest, #fifa officials, arrest	game, throne best, episode episode, #gameofthrones episode, season holy, shit #gameofthrones, episode watch, #gameofthrones throne, season new, episode late, episode
NVC	cavs game #nbafinals golden state warriors james lebron win curry steph golden state warrior cavs kelp wiggins cavaliers cleveland watch	tweet fandom kn guy make amor maria love kathniels 😍😍💑👨🏻‍🤝‍👨🏻💙💙 iconic scene johndria kathniel manford fan kn	#fifa cup world blatter sepp corruption fifa = fifa mafia arrest football israeli say time soccer u.s.	episode #gameofthrones jon snow watch season game children walker white throne direwolf winter come

Abbildung 5: Vergleich von NLDA und NVC

schen Sprachprobleme, wie in Abschnitt 1.1 erklärt, und der Kürze der Tweets.
Die NMF-Methode lieferte im Gegensatz zur LDA-Methode Ergebnisse mit besonders wenigen Hashtags, was zum Verständnis beiträgt. Die NLDA-Methode hat, im Vergleich zu den beiden anderen Methoden den Textzusammenhang deutlich besser dargestellt. Beispielsweise wird als bester Treffer der NLDA-Methode im ersten Thema *„#seppblatter resign“* geliefert, wohingegen bei LDA und NMF eine Form des Wortes *„resign“* erst an dritter und vierter Stelle auftritt.

Für eine vergleichende Evaluation der Methoden wurden die jeweils ersten 20 Resultate der LC-, NVC-, VPC-, NGC-, LDA-, NLDA- und NMF-Methode herangezogen. Sowohl im Fragebogen als auch in der eigenen Evaluation ist bei der NGC-Methode $n = 3$. Hierbei wurden die C- und SWC-Methode aufgrund der LC-Methode vernachlässigt. Die FSC-, VC- und NC-Methoden wurden durch die ähnlichen und besseren Ergebnisse der NVC-Methode nicht verglichen. Bei der NGC-Methode wurden nur Trigramme betrachtet, da sich diese die Waage zwischen großem Kontextbezug und geringer Anfälligkeit für Werbetweets halten.
Die Methoden wurden mit sechs Kriterien auf einer Punkte Skala von 1 (schlecht) bis 5 (sehr gut) verglichen. Es wurden die gleichen Hashtags zum Vergleich herangezogen, wie im Fragebogen, Abschnitt 5.4.2.

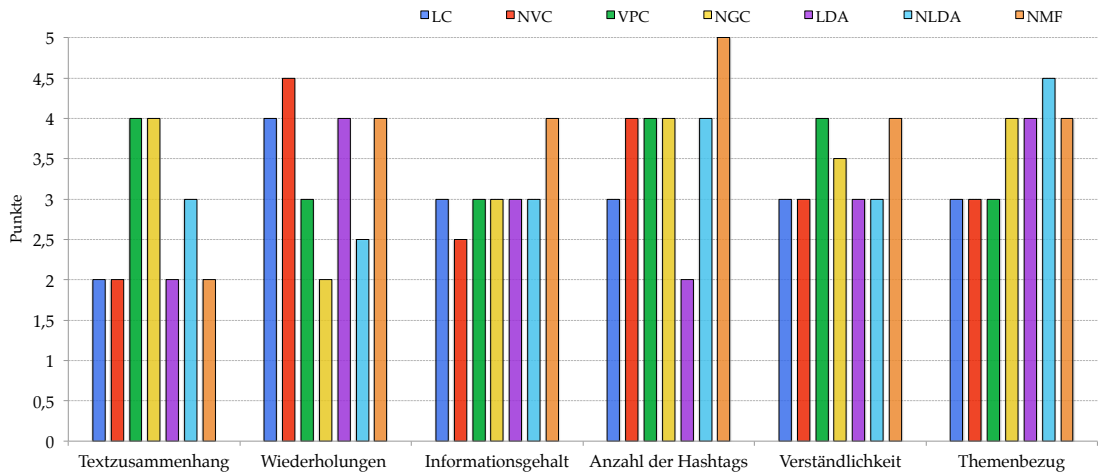

Abbildung 6: Vergleich der Mediane der Kriterien des Autors

- Der *Textzusammenhang*: Lassen sich aus der Anordnung oder Reihenfolge der Wörter Satzteile bilden? Wird der Kontext ersichtlich?
- Die Häufigkeit der *Wiederholungen* einzelner Wörter. Bis zu 5% Wiederholungen ergibt 5 Punkte, bis 20% 4 Punkte, 30% 3 Punkte, bis 40% 2 Punkte und danach 1 Punkt.
- Der *Informationsgehalt* der Wörter: z.B. hätte *„if“* einen niedrigen und *„corruption“* einen hohen subjektiven Informationsgehalt.
- Die Häufigkeit der *Hashtags*: Viele Hashtags tragen zur Unverständlichkeit bei und werden daher negativ bewertet. Bis zu 5% Hashtags 5 Punkte, bis 10% 4 Punkte, bis 15% 3 Punkte, bis 25% 4 Punkte und danach 1 Punkt.
- Die *Verständlichkeit* der Wörter
- Der *Themenbezug* der Wörter

Abbildung 6 zeigt, dass die VPC- und NGC-Methoden beim Erstellen eines **Textzusammenhangs** mit einem Median von 4 am besten abschneiden. Die NLDA-Methode ist mit einem Median von 3 auf dem dritten Platz. Dies kann durch die Bildung der N-Gramme einerseits und durch die VPC-Methode andererseits erklärt werden, die gerade die Abhängigkeit von Verben und Substantiven aufzeigt.
Wiederholungen mit durchschnittlich über $35{,}01\%$ finden bei der NGC-Methode am häufigsten statt, mit nur $2{,}30$ Wiederholungen im Mittel hat die NVC-Methode am wenigsten.
Beim **Informationsgehalt** der einzelnen Ergebnisse gab es keinen großen Unterschied. Nur bei der NMF-Methode, die mit einem Median von 4 am besten bewertet ist, konnten vereinzelt Wörter festgestellt werden, die seltener aber präziser waren.
Die LDA-Methode hat im Mittel mit $3{,}90$ **Hashtags** pro Ergebnis am meisten Hashtags

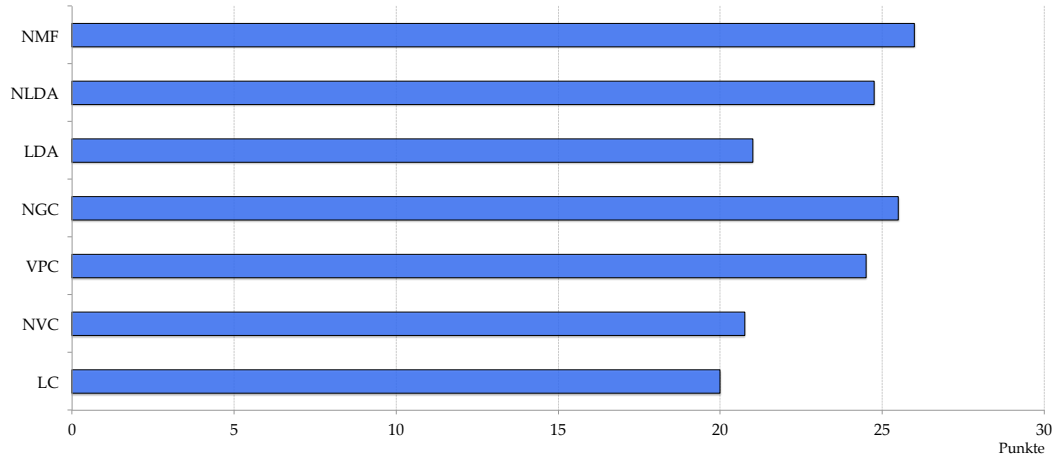

Abbildung 7: Gesamtbewertung des Autors

beinhaltet. Wohingegen die NMF-Methode mit im Mittel nur $0{,}30$ am wenigsten Hashtags aufweist. Damit hat sie im Mittel $0{,}50$ weniger als die VPC-Methode auf Platz zwei am wenigsten Hashtags aufweist.
Bei der **Verständlichkeit** der Ergebnisse der Methoden gab es keinen großen Unterschied.
Beim **Themenbezug**, wie sich in Abbildung 6 zeigt, ist die NLDA-Methode am besten bewertet. Die LC-, NVC- und VPC- Methoden sind mit einem Median von 3 am geringsten bewertet.

Die sechs selbstgewählten Kriterien haben aus Sicht des Autors verschieden starken Einfluss auf die Bewertung der Methoden bezüglich ihrer Eignung zur Themenerkennung. Dies spiegelt sich in der Gewichtung der einzelnen Kriterien wider.
Die kleinste Gewichtung hat das Kriterium „Wiederholungen“ erhalten. Dies ist damit zu begründen, dass Wiederholungen auch den Fokus des Themas verdeutlichen können. Die Verständlichkeit, die Anzahl der Hashtags und der Informationsgehalt wurden mit 1 bewertet. Die Faktoren werden als gleichermaßen wichtig empfunden und dienen zur Normierung der anderen Gewichtungen.
Da das Kontextverständnis als sehr wichtig erachtet wird, um die Handlung zu begreifen, wurde der Textzusammenhang mit $1{,}5$ gewichtet.
Zu guter Letzt erhielt der Themenbezug eine Gewichtung von $2{,}0$, nachdem Themenerkennung im Zentrum dieser Arbeit steht.
Die **Gesamtbewertung** wird in Abbildung 7 dargestellt. Die NMF-Methode und die NGC-Methode belegen mit marginalem Unterschied die ersten beiden Plätze. Auf Platz drei und vier befinden sich die NLDA-Methode und die VPC-Methode. Auch diese beiden Methoden trennt nur ein marginaler Unterschied. Die LDA-Methode landet auf Platz fünf. Sie ist jedoch nur wenig besser als die NVC-Methode auf Platz sechs. Die LC-Methode schneidet am schlechtesten von allen Methoden ab und findet sich somit auf Platz sieben wieder. Der größte Sprung findet zwischen der VPC-Methode und der LDA-Methode statt. Dementsprechend zeichnet sich ein Bild mit vier relativ gut geeigneten und drei vergleichsweise schlecht geeigneten Methoden ab.

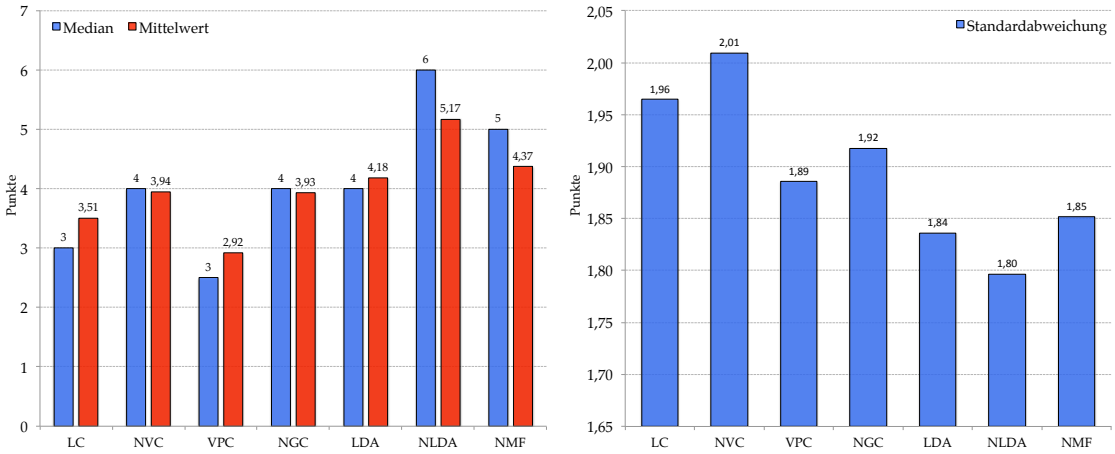

Abbildung 8: Auswertung des Fragebogens

5.4.2 Auswertung des Fragebogens

Die Aufgabenstellung im Fragebogen lautete, zu einem bestimmten Hashtag aus dem Datensatz, 20 zufällig ausgewählte Tweets zu lesen. Mit dem gewonnenen Verständnis wurden die Ergebnisse der Methoden in eine Reihenfolge von 1 bis 7 geordnet. Hierbei bedeutete 1, dass das Ergebnis das Thema der Tweets am schlechtesten beschreibt. 7 bedeutete, dass das Ergebnis das Thema der Tweets am besten beschreibt. Diese Aufgabenstellung galt es für 10 Hashtags zu beantworten. Insgesamt wurden 20 Personen befragt.

Abbildung 8 zeigt den Median und den Mittelwert der Bewertung der jeweiligen Methoden, bezogen auf alle Hashtags. Der Median hat die Eigenschaft, die Ausreißer in der Bewertung zu vernachlässigen, wohingegen auf dem Mittelwert die Ausreißer einen größeren Einfluss haben. Durch den direkten Vergleich von Mittelwert und Median wird die Tendenz der Bewertung besser ersichtlich.
Die VPC-Methode, siehe Abbildung 8, hat mit nur 2,92 im Mittel die schlechteste Bewertung. Sie wurde mit 67 von 200 am häufigsten mit 1 Punkt bewertet. Möglicherweise liegt diese Bewertung daran, dass die Richtung der Abhängigkeit von Verb und Substantiv nicht immer ersichtlich wird.
An vorletzter Stelle liegt die LC-Methode, mit dem gleichen Median wie die VPC-Methode. Sie ist allerdings im Mittel um fast 0,60 Punkte besser ist.
Die NVC-, LDA und NGC-Methoden liegen jeweils mit einem Median von 4 gleich auf. Jedoch hat die NVC-Methode die höchste Standardabweichung von 2,01 Punkten.
Am zweitbesten wurde die NMF-Methode mit einem Median von 5 und einem Mittelwert von 4,37 bewertet.
Die Abbildung lässt erkennen, dass die NLDA-Methode mit einem Median von 6 am besten bewertet wurde. Auch im Mittel hat sie mit 5,17 Punkten noch die höchste Bewertung. Hinzu kommt, dass die Standardabweichung für die NLDA-Methode am geringsten war, folglich hier die größte Übereinstimmung in der Bewertung geherrscht hat. Außerdem wurde sie 59 mal mit 7 Punkten bewertet und damit über 20 mal mehr

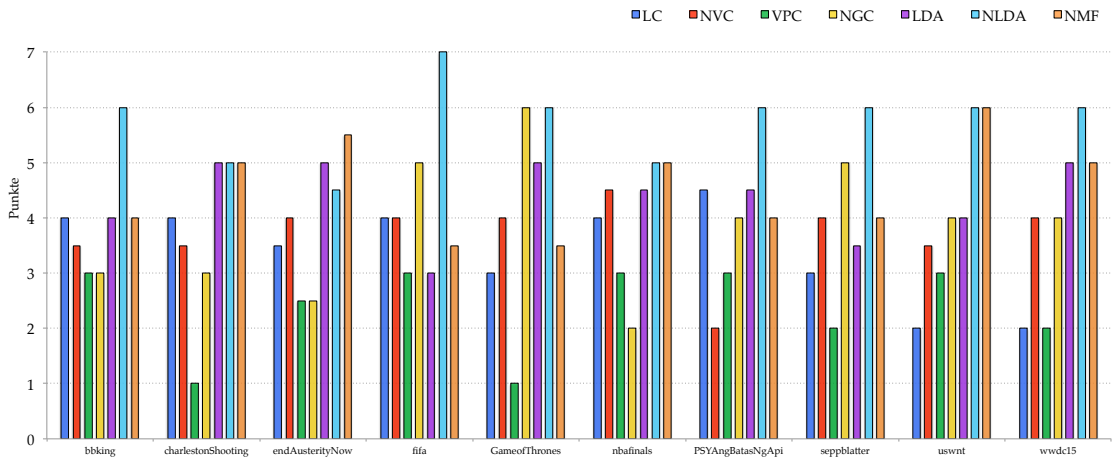

Abbildung 9: Vergleich der Mediane der Methoden bezüglich der Hashtags

als die Methode mit den zweitmeisten 7 Punkte Bewertungen (NVC). Insgesamt wurde somit die NLDA-Methode deutlich am besten bewertet.

In Abbildung 9 wird anhand des Medians ersichtlich, dass die Bewertung der Methoden vom Hashtag abhängt ist:
Gerade die NGC-Methode variiert stark in ihrer Bewertung von Hashtag zu Hashtag. Bei *#GameofThrones* ist sie zusammen mit der NLDA-Methode am besten bewertet, wohingegen sie bei *#nbafinals* mit einem Median von 2 die deutlich schlechteste Bewertung hat. Sowohl bei der NGC-Methode als auch bei der NVC-Methode zeigt sich die Abhängigkeit der Bewertung vom Hashtag. Es ist somit kein Einzelfall. Diese hat bei *#PSYAngBatasNgApi* die schlechteste Bewertung. Hingegen ist sie bei *#nbafinals*, *#fifa* und *#GameofThrones* jeweils unter den besten vier Methoden. Dies zeigt sich auch anhand der Ergebnisse der NVC-Methode in Abbildung 5, die Resultate unter *#PSYAngBatasNgApi* sind kaum verständlich, wohingegen unter *#fifa* mit *„corruption fifa“*, *„fifa = mafia“* und *„arrest“* wichtige Schlagwörter vorkommen. Diese Differenz in der Bewertung zeigt sich auch in den Ergebnissen. Die NVC-Methode erhielt sowohl am zweit häufigsten 7 Punkte, als auch 2 Punkte.
Allerdings wird anhand der VPC-Methode, die bei keinem Hashtag einen Median von 3 überschreitet und anhand der NLDA-Methode, die bei keinem Hashtag einen Median von 5 unterschreitet erkennbar, dass es durchweg einheitlich bewertete Methoden gibt. In Abbildung 5 wird ersichtlich, dass die NLDA-Methode im Vergleich zur NVC-Methode verständliche Ergebnisse liefert.

Allgemein wurde von den Befragten folgende Kritik angebracht: Die Tweets wurden durchweg als unverständlich kritisiert. Dies zeigt, dass die zuvor in Punkt 1.1 festgestellte Eigenart von Twitter erheblichen Einfluss auf das Textverständnis hat.
Weiter wurde häufig angemerkt, dass kaum ein Unterschied in der Qualität der Ergebnisse der Methoden ersichtlich wurde. Wenn eine Unterscheidung gemacht werden konnte, dann meist nur in zwei Hälften: Die schlechteren Methoden und die besseren Methoden. Diese Kritik zeigt sich allerdings nicht in der Auswertung.

5.4.3 Vergleich von Fragebogen und der Auswertung des Autors

Beim Vergleich der Auswertung des Fragebogens mit der Bewertung des Autors fallen insbesondere zwei Unterschiede auf: Die Bewertung der VPC-Methode und die Bewertung der NGC-Methode. Den drastischsten Bewertungsunterschied weißt die VPC-Methode auf. Sie wurde vom Autor deutlich besser bewertet als von den Fragebogen-Teilnehmern. Zum einen wurde bei der Bewertung des Autors der Textzusammenhang womöglich höher bewertet als dies bei den Fragebogen-Teilnehmern intuitiv der Fall war. Noch ausschlaggebender könnte jedoch sein, dass die Ausgabe der VPC-Methode auf den ersten Blick wenig verständlich erscheint. Ohne Wissen über die Methode sind die Bezüge innerhalb der Aussage nur schwer erkennbar.
Der Unterschied in der Bewertung der NGC-Methode liegt womöglich ebenfalls in der Bewertung des Kriteriums "Textzusammenhang“.
Die NLDA-Methode und die NMF-Methode schneiden sowohl bei der Bewertung des Autors als auch beim Fragebogen gut ab. Zwar ist ihre Platzierung vertauscht, aber in beiden Fällen gehören beide zu den drei am besten bewerteten Methoden. Die LC-Methode schneidet mit einem letzten und einem vorletzten Platz in beiden Fällen schlecht ab.

6 Stand der Forschung

Es gibt diverse Forscher, die sich mit Themenerkennung in Twitter auseinandersetzen. In diesem Kapitel werden ähnliche Fragestellungen und der aktuelle Stand der Forschung zusammengefasst.

6.1 TNMF

Die in dieser Bachelorarbeit vorgestellte NMF-Methode faktorisiert eine Matrix, die Dokumente und Terme in Beziehung setzt, wie die Matrix X in Abbildung 10 (a), (c). Im Gegensatz dazu stellen Yan et al. [YGL+13] eine Variante der NMF vor, die **TNMF**, die neben einer Term-Dokumenten Matrix eine Matrix faktorisiert, die Term Korrelationen bemisst. Eine solche zu faktorisierende Matrix S wird in Abbildung 10 (b) dargestellt. Das heißt, dass jeder Vektor in der Matrix S in Abbildung 10 (b) als Einträge die Wahrscheinlichkeiten des gemeinsamen Auftretens zweier Terme beinhaltet. Aus der Faktorisierung entsteht eine Matrix U, die zu jedem Thema die Terme enthält. Anschließend berechnen sie mit der Dokumenten-Term Matrix X und der erstellten Term-Thema Matrix U eine Thema-Dokumenten Matrix V in Abbildung 10 (c).
Ein als *sparsity* bekanntes Problem bei Kurztexten ist, dass in einer Dokumenten-Term Matrix bei einem großen Datensatz viele Elemente 0 sind, da die Terme nicht im Dokument enthalten sind. Mit diesem Verfahren umgeht [YGL+13] das Problem der *sparsity*.
Ein Twitter Datensatz, Tweets2011[8], wird hier neben anderen zur Evaluation benutzt.
Beim Vergleich zwischen LDA und zwei Varianten ihres Verfahrens, TNMF_E und TNMF_I, die mit verschiedenen Distanzmaßen die Matrix V berechnen, übertreffen sie die Ergebnisse von LDA auf allen getesteten Datensätzen. Im Gegensatz zu LDA, das viele allgemeine Wörter in den Themen hatte, haben die Varianten der TNMF lesbarere und spezifischere Wörter hervorgebracht. Auch im Rahmen dieser Bachelorarbeit hat die NMF-Methode mehr spezifischere Worte als die LDA-Methode geliefert.

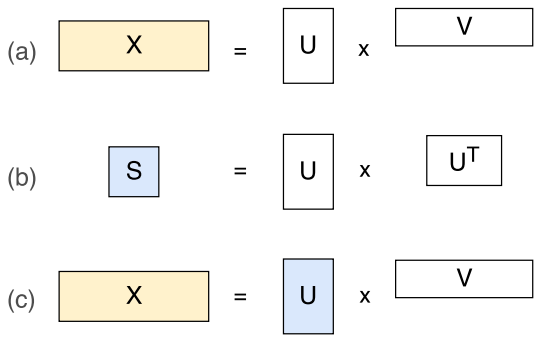

Abbildung 10: Matrix Faktorisierungen bei TNMF [YGL+13]

[8] http://trec.nist.gov/data/tweets/

6.2 Biterm Topic Model

Von Cheng et al. [CYLG14] wird ein LDA entlehnter Ansatz für ein Topic Model vorgestellt. Ihr *Biterm Topic Model* (**BTM**) ist eine Spezialisierung auf Kurztexte. Beim *BTM* werden die einzelnen Dokumente zunächst in ihre Biterme, ungeordnete Paare gemeinsam auftretender Wörter, zerlegt und diese dann zu einer Sammlung aggregiert. Anschließend werden die Biterme mit einer vereinfachten Form des LDA Verfahrens zu Themen geordnet.
[CYLG14] vergleichen *BTM* und LDA unter anderem ebenfalls in Bezug auf den Datensatz Tweets2011. Zum Vergleichen der Methoden benutzen sie den „pointwise mutual information“ Score (*PMI Score*) [NKC09], der durch punktweises Vergleichen gemeinsamer Informationen mit großen Datensätzen die Kohärenz der Themen bemisst. Hierbei wurde festgestellt, dass *BTM* LDA konsistent und signifikant übertrifft. Außerdem hat *BTM*, bei einem händischen Vergleich des Themenbezugs der zu einem Thema zugeordneten Wörter übertroffen.
Neben *BTM* wird in [CYLG14] auch online BTM (**oBTM**) und incremental BTM (**iBTM**) vorgestellt. *oBTM* teilt die Dokumente in Zeitabschnitte ein und *iBTM* updated das Model zum Trainieren der Themen konstant mit Bitermen. Beide Formen dienen der besseren Verarbeitung von großen Textdatenmengen, die sich ständig aktualisieren, wie auf Mikroblogging Plattformen, z.B. Twitter. Bei einem Vergleich zwischen einer online LDA Version (iLDA) mit *oBTM* und *iBTM* war der *PMI-Score* von *iBTM* und *oBTM* sehr ähnlich aber besser als der von *iLDA*.
Das BTM-Verfahren beruht wie auch die NLDA-Methode auf dem Gedanken, dass die Information darüber, dass Wörter gemeinsam auftreten, berücksichtig werden sollte. Denn diese Information gibt ein besseres Verständnis für die Abhänigkeiten der Wörter. Beide Ansätze bauen auf dem LDA Verfahren auf. Ein entscheidender Unterschied zur NLDA-Methode besteht darin, dass das BTM-Verfahren nicht nur die einzelnen Tweets in Biterme auflöst, sondern den ganzen Datensatz.

6.3 Aggregation von Tweets

Das Author-Topic-Model (AT) [RZGSS04] ist eine Erweiterung von LDA, bei dem, zu jedem Dokument zusätzlich zu Themen Autoren zugeordnet werden. Hong und Davison [HD10] haben in ihrer Arbeit die LDA und AT zur Themenerkennung benutzt, um diese speziell auf Mikroblogging-Plattformen wie Twitter zu trainieren. Sie entdeckten, dass die Länge der Dokumente ausschlaggebend für die Ergebnisse der Methoden sind. Hierbei werden durch die Vergrößerung der Dokumente die Ergebnisse verbessert. In Ihrer Arbeit stellen sie fest, dass eine Aggregation der Tweets nach Benutzern und nach Wörtern (Hashtags) die Ergebnisse positiv beeinflusst, wobei das Clustering nach thematisch bestimmten Benutzern die besten Resultate liefert. Nach der Aggregation spielt auch die Wahl der Anzahl der Themen für die Qualität der Ergebnisse eine entscheidende Rolle. Generell verringert die Aggregation die zu wählende Themenanzahl.
Diese Erkenntnis unterstützt die zu Beginn gemachte Annahme, dass das Clustering der Tweets bezüglich eines trendigen Hashtags zu einem allgemeinen Thema statt vieler Themen führt.

7 Fazit

Das Ziel der vorliegenden Arbeit war es Methoden zur Erkennung der Themen in Tweets zu finden. Dabei wurden Datensätze aus Tweets trendiger Hashtags benutzt. Ein besonderes Augenmerk wurde auf die Verwendung eines für Twitter optimierten POS Taggers und dependency parsers, den *Tweebo-Parser*, gelegt. Dieser ermöglichte es, die besondere Ausdrucksweise auf Twitter mit Hilfe von *JWI* auf das Wesentliche zu reduzieren.

Bei der Bestimmung des Themas wurden nicht nur Substantive, sondern auch Wörter anderer Wortarten berücksichtigt, obwohl dies nicht der klassische Weg ist. Hierbei hat sich gezeigt, dass die Kombination von Verben und Substantiven den Textzusammenhang deutlich verständlicher macht. Dies gilt besonders dann, wenn die Dependenzen erkennt werden.

Um zu bestimmen, wie gut die verschiedenen Methoden dazu geeignet sind, Themen in Twitter zu erkennen, wurden zwei Auswertungen durchgeführt. Eine anhand von sechs vom Autor aufgestellten Kriterien und eine anhand eines Fragebogens mit 20 Teilnehmern. Dabei hat sich gezeigt, dass unter den zu vergleichenden Methoden solche waren, die durchwegs besser bewertet wurden (z.B. die NLDA-Methode) und solche, die durchwegs schlechter bewertet wurden (z.B. die LC-Methode). Aber es gab auch Methoden, bei denen nicht eindeutig festgestellt werden konnte, wie sie im Vergleich zu den anderen zu bewerten sind. Die Bewertung der NGC-Methode war stark abhängig vom Hashtag und die VPC-Methode hat in der Bewertung des Autors deutlich besser abgeschnitten als beim Fragebogen.

Bezüglich der hier vorgestellten populären Verfahren konnte festgestellt werden, dass die NMF-Methode auch in dieser Arbeit besser funktionierte als die auf reinen Bag-of-Words beruhenden Methoden. Bemerkenswert ist, dass die NMF-Methode es geschafft hat, kaum Hashtags in ihren Ergebnissen aufzulisten. Außerdem hat die NLDA-Methode, die auf der LDA-Methode aufbaut, durch Verwendung der Bigramme das Ergebnis der LDA-Methode verbessert.

Um die Ergebnisse dieser Arbeit zu bekräftigen, sollten Forschungsarbeiten mit größerer statistischer Aussagekraft angefertigt werden. Im Rahmen weiterer Arbeiten sollten außerdem mehr Informationen mit in die Themenerkennung einbezogen werden, wie etwa das Datum des Tweets, die Häufigkeit des Retweets und die Identität des Benutzers. Lohnenswert wäre auch eine Überprüfung, inwieweit das Clustering von mehreren Tweets zu einem Dokument die Ergebnisse verbessert. Zuletzt ist es ebenfalls erstrebenswert, den Einfluss verschiedener Wortarten auf die Themenerkennung zu analysieren.

A Anhang

Tabelle 2: POS Tagset [OOD+13]

Tag	Bedeutung
N	common noun
O	pronoun (personal/WH; not possessive)
^	proper noun
S	nominal + possessive
Z	proper noun + possessive
V	verb including copula, auxiliaries
L	nominal + verb, verbal + nominal
M	proper noun + verbal
A	adjective
R	adverb
!	interjection
D	determiner
P	pre- or postposition, or subordinating conjunction
&	coordinating conjunction
T	verb particle
X	existential there, predeterminers
Y	X + verbal
#	hashtag
@	at-mention, indicates topics/category for tweet
~	discourse marker, indications of continuation across multiple tweets
U	URL or email address
E	emoticon
$	numeral
,	punctuation
G	other abbreviations, foreign words, possessive endings, symbols, garbage

#endAusterityNow

Tweets:

- Tory Councillor Tells #EndAusterityNow Marchers To Leave Government 'To The Grownups'
- UK anti-austerity demonstrations #EndAusterityNow
- Now marching down Whitehall. #EndAusterityNow
- #EndAusterityNow Protect the vulnerable, save the future #ImThereToo
- @twcuddleston #EndAusterityNow by going shopping - but only buying nice things #madeinengland
- #EndAusterityNow if you voted @David_Cameron into @Number10gov then GO HOME!!! You had your say and your say was he must be prime minister.
- Gathering in Parliament Square #endausteritynow
- Doesn't matter if you couldn't go today, your heart is there, that matters #EndAusterityNow
- #EndAusterityNow fuck the fucking fuckers!
- Rising up against unfounded,unfair and polluting austerity policies #EndAusterityNow
- We're at the destination! #EndAusterityNow Parliament Sq!
- #EndAusterityNow. Put those 1 million people BACK on to the dole.
- Conservatives not so much a political party, more a social disease #EndAusterityNow
- #EndAusterityNow #ImThereToo My family counts too. Don't take away our lives. We are disabled, autistic and ill but we are still human!
- #bbcnews We don't want to end up like Greece say the UKGovernment - but austerity is exacerbating Greece's problems #EndAusterityNow
- Here come the plebs #EndAusterityNow
- Smoke bomb. Photographs swarm, hoping for riot. Everyone else just manoeuvres around it. #EndAusterityNow
- being watched #EndAusterityNow
- We are on the move!! What a turnout!!! #EndAusterityNow ✊🏽
- @sussexlrc all those that want to #EndAusterityNow need to sign up as Labour supporters and vote @Corbyn4Leader

#endAusterityNow

☐ ☐ ☐ ☐ ☐ ☐ ☐

LDA

Thema 1
#endausteritynow
anti austerity march
demo
live
#imtheretoo
make
cut
#junedemo
can't
want

Thema 2
#endausteritynow
london march
i
people
today
protest
march
just
say
good

NGC3

join, demonstrate, social
demonstrate, social, medium
can't, make, demo
make, demo, join
demo, join, demonstrate
labour, supporter, vote
need, sign, labour
sign, labour, supporter
#endausteritynow, need, sign
want, #endausteritynow, need

NVC

#endausteritynow
london march
people
anti austerity march
demo
today
protest
make
cut
march

NMF

Thema 1
london march
today
protest
thousand
solidarity
rally
anti-austerity
march
glasgow 👊🏽👊🏽
bbc news

Thema 2
people
anti austerity
march
demo
labour mp
cut
medium
tory westminster
child credit tax
government tory

LC

#endausteritynow
march
i
people
austerity
london
demo
today
live
#imtheretoo

VPC

march [today]
join [demo]
wish [solidarity]
live [austerit]
watch [video]
stop [cut]
support [miner]
tell [marchers, councillor]
turn [student]
cut [war, welfare]

NLDA

Thema 1
social, medium
can't, make
join, demonstrate
demonstrate, social
make, demo
demo, join
want, #endausteritynow
sign, labour
#endausteritynow, need
need, sign

Thema 2
#endausteritynow, march
#endausteritynow, demo
march, london
#endausteritynow, protest
good, luck
end, austerity
parliament, square
#endausteritynow, rally
bank, england
anti-austerity, rally

Abbildung 11: Fragebogen Beispielseite für #endAusterityNow

B Abkürzungsverzeichnis

API	Application Program Interface
BTM	Biterm Topic Model
C	Counter
FSC	FirstSentenceCount
iBTM	incremental BTM
IDF	inverse Dokumentenhäufigkeit
JWI	Java WordNet Interface
LC	LemCount
LDA	Latent Dirichlet Allocation
MWE	Multi Word Expression
NC	NounPOSCount
NGC	NGramCount
NLDA	N-Gramm-LDA
NMF	nicht-negative Matrix-Faktorisierung
NVC	NounVerbPOSCount
oBTM	online BTM
PMI	Pointwise Mutual Information
POS	Part-of-Speech
SWC	StopWordCount
TF	dokumentspezifische Termhäufigkeit
VC	VerbPOSCount
VPC	VerbPhraseCount

Literatur

[BL09] BLEI, David M. ; LAFFERTY, John D.: Topic Models. In: *Text Mining: Classification, Clustering, and Applications* 10 (2009), S. 71–89

[Ble12] BLEI, David M.: Probabilistic Topic Models. In: *Commun. ACM* 55 (2012), April, Nr. 4, S. 77–84

[BM06] BUCHHOLZ, Sabine ; MARSI, Erwin: CoNLL-X Shared Task on Multilingual Dependency Parsing. In: *Proceedings of the Tenth Conference on Computational Natural Language Learning*, Association for Computational Linguistics, 2006 (CoNLL-X '06), S. 149–164

[BNJ03] BLEI, David M. ; NG, Andrew Y. ; JORDAN, Michael I.: Latent Dirichlet Allocation. In: *J. Mach. Learn. Res.* 3 (2003), März, S. 993–1022

[Car01] CARSTENSEN KAI-UWE, EBERT CHRISTIAN, ENDRISS CORNELIA, JEKAT SUSANNE, KLABUNDE RALF, Langer H.: *Computerlinguistik und Sprache: Eine Einführung*. Berlin : Spektrum, 2001

[CBGW+09] CHANG, Jonathan ; BOYD-GRABER, Jordan ; WANG, Chong ; GERRISH, Sean ; BLEI, David M.: Reading Tea Leaves: How Humans Interpret Topic Models. In: *Neural Information Processing Systems*, 2009

[CYLG14] CHENG, X ; YAN, X ; LAN, Y ; GUO, J: BTM: Topic Modeling over Short Texts. In: *Knowledge and Data Engineering, IEEE Transactions on* PP (2014), Nr. 99, S. 1

[Dij77] DIJK, Teun A.: Sentence topic and discource topics. In: *Papers in Slavic Philology* (1977), S. 49–62

[Edm69] EDMUNDSON, H. P.: New Methods in Automatic Extracting. In: *J. ACM* 16 (1969), Nr. 2, S. 264–285

[Fin14] FINLAYSON, Mark A.: Java libraries for accessing the Princeton WordNet: Comparison and evaluation. In: *Proceedings of the 7th Global Wordnet Conference, Tartu, Estonia*, 2014

[Gri04] Finding scientific topics. In: *Proceedings of the National Academy of Sciences of the United States of America* 101 Suppl (2004), S. 5228–5235

[GSO+11] GIMPEL, Kevin ; SCHNEIDER, Nathan ; O'CONNOR, Brendan ; DAS, Dipanjan ; MILLS, Daniel ; EISENSTEIN, Jacob ; HEILMAN, Michael ; YOGATAMA, Dani ; FLANIGAN, Jeffrey ; SMITH, Noah A.: Part-of-speech Tagging for Twitter: Annotation, Features, and Experiments. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, Association for Computational Linguistics, 2011, S. 42–47

[HD10] HONG, Liangjie ; DAVISON, Brian D.: Empirical Study of Topic Modeling in Twitter. In: *Proceedings of the First Workshop on Social Media Analytics*, ACM, 2010 (SOMA '10), S. 80–88

[JM08] JURAFSKY, Daniel ; MARTIN, James H.: *Speech and Language Processing (2nd Edition) (Prentice Hall Series in Artificial Intelligence)*. 2. Prentice Hall, 2008

[Kau10] KAUFMANN, Max: Syntactic Normalization of Twitter Messages. In: *Studies* 2 (2010)

[KSS+14] KONG, Lingpeng ; SCHNEIDER, Nathan ; SWAYAMDIPTA, Swabha ; BHATIA, Archna ; DYER, Chris ; SMITH, Noah a.: A Dependency Parser for Tweets. In: *Proceedings of EMNLP 2014* (2014)

[LS99] LEE, Daniel D. ; SEUNG, H. S.: Learning the parts of objects by nonnegative matrix factorization. In: *Nature* 401 (1999), S. 788–791

[Mil95] MILLER, George A.: WordNet: A Lexical Database for English. In: *Commun. ACM* 38 (1995), November, Nr. 11, S. 39–41

[MS99] MANNING, Christopher D. ; SCHÜTZE, Hinrich: *Foundations of Statistical Natural Language Processing*. Cambridge, Massachusetts : The MIT Press, 1999

[NKC09] NEWMAN, David ; KARIMI, Sarvnaz ; CAVEDON, Lawrence: External Evaluation of Topic Models. In: *in Australasian Doc. Comp. Symp., 2009* Citeseer, 2009

[NS04] NIVRE, Joakim ; SCHOLZ, Mario: Deterministic Dependency Parsing of English Text. In: *Proceedings of the 20th International Conference on Computational Linguistics*, Association for Computational Linguistics, 2004 (COLING '04)

[OOD+13] OWOPUTI, Olutobi ; O'CONNOR, Brendan ; DYER, Chris ; GIMPEL, Kevin ; SCHNEIDER, Nathan ; SMITH, Noah a.: Improved Part-of-Speech Tagging for Online Conversational Text with Word Clusters. In: *Proceedings of NAACL-HLT 2013* (2013), Nr. June, S. 380–390

[RDL10] RAMAGE, Daniel ; DUMAIS, Susan T. ; LIEBLING, Daniel J.: Characterizing Microblogs with Topic Models. In: *ICWSM*, The AAAI Press, 2010

[RZGSS04] ROSEN-ZVI, Michal ; GRIFFITHS, Thomas ; STEYVERS, Mark ; SMYTH, Padhraic: The Author-topic Model for Authors and Documents. In: *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, AUAI Press, 2004 (UAI '04), S. 487–494

[Sto07] STOCK, Wolfgang G.: *Information retrieval : Informationen suchen und finden*. München : R. Oldenbourg, 2007

[XLG03] XU, Wei ; LIU, Xin ; GONG, Yihong: Document Clustering Based on Non-negative Matrix Factorization. In: *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, ACM, 2003 (SIGIR '03), S. 267–273

[YGL+13] YAN, Xiaohui ; GUO, Jiafeng ; LIU, Shenghua ; CHENG, Xueqi ; WANG, Yanfeng: Learning Topics in Short Texts by Non-negative Matrix Factorization

on Term Correlation Matrix. In: *Proceedings of the SIAM International Conference on Data Mining* (2013), S. 1–9

[ZJW+11] ZHAO, Wayne X. ; JIANG, Jing ; WENG, Jianshu ; HE, Jing ; LIM, Ee-Peng ; YAN, Hongfei ; LI, Xiaoming: Comparing twitter and traditional media using topic models. In: *33rd European Conference on IR Research, ECIR 2011* (2011), S. 338–349

Abbildungsverzeichnis

Tabellenverzeichnis