

Hochschule Wismar

Fakultät für Ingenieurwissenschaften

Master-Thesis

**Design und Implementierung von Social Bots:
Methoden der Künstlichen Intelligenz zur
Generierung von automatisierten Nachrichten**

Abschlussarbeit zur Erlangung des Grades eines

Master of Engineering

der Hochschule Wismar

Eingereicht am: 30. September 2019

von: Virginia Dei
Studiengang IT-Sicherheit und Forensik

Erstgutachterin: Frau Prof. Dr.-Ing. A. Raab-Düsterhöft

Zweitgutachter: Herr Prof. Dr. Ernst Jonas

Aufgabenstellung

Heutzutage findet die Kommunikation der Menschen häufig über die sozialen Kanäle im Internet statt. Doch nur den wenigsten Nutzern dieser Medien fallen die menschlichen Imitationen auf, welche als scheinbar menschliche Nutzer agieren, die sogenannten Social Bots. Das Ziel dieser Master-Thesis soll sein, einen Social Bot für das soziale Netzwerk Twitter zu entwickeln. Dieser soll mithilfe von Methoden der künstlichen Intelligenz automatisiert Posts und Retweets erstellen und auf Kommentare selbstständig antworten können. Somit kann eine reale Kommunikation zwischen Mensch und Maschine stattfinden. Es soll herausgestellt werden, welche Gefahren und potenziellen Angriffe durch solche Programme möglich sind. Zudem werden der Aufbau und die Implementierung von Social Bots in den sozialen Medien aufgezeigt. Zusätzlich werden einige Erkennungsprogramme auf ihre Funktionalität und Einsatzfähigkeit für den entwickelten Social Bot getestet. Die Thesis soll außerdem herausstellen, welche Möglichkeiten in den Bereichen Social Bots und künstlicher Intelligenz existieren und ob die derzeitigen Erkennungsmethoden ausreichen, um intelligente Social Bots zu erkennen. Das Ergebnis dieser Arbeit soll die Nutzer von sozialen Medien für dieses Thema sensibilisieren und das Gefahrenbewusstsein stärken.

Abstract

Today, people's communication often takes place via Social Media on the Internet. But only very few users of these media notice the human imitations, which act as seemingly human users, the so-called Social Bots. The objective of this master thesis is to develop a Social Bot for the Social Network Twitter. With the help of Artificial Intelligence methods, this bot will automatically create posts, retweets and be able to respond to comments independently. Thus a real communication between man and machine can take place. The dangers and potential attacks that are possible will be highlighted. In addition, the structure and implementation of Social Bots in Social Media will be demonstrated. Besides, some detection programs will be tested for their functionality and usability for the developed Social Bot. The thesis should also highlight which possibilities exist in the areas of Social Bots and Artificial Intelligence and whether the current detection methods are sufficient to detect intelligent social bots. The result of this work is to sensitize the users of Social Media to this topic and to strengthen the awareness of dangers.

Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich während der Anfertigung meiner Master-Thesis motiviert und unterstützt haben.

Zu Beginn möchte ich Frau Prof. Raab-Düsterhöft danken, die meine Arbeit betreut, mich richtungsweisend und mit viel Engagement begleitet hat. Des Weiteren möchte ich mich gerne für die konstruktive Kritik und hilfreichen Anregungen von Herrn Prof. Jonas herzlich bedanken.

Ein herzliches „Dankeschön“ geht ebenfalls an meine Kollegen und Freunde für den emotionalen und starken Rückhalt über die gesamte Dauer meines Studiums und besonders in der Zeit der Erstellung meiner Master-Thesis.

Der größte Dank gilt meinem Mann und meinen Eltern, die stets ein offenes Ohr für mich hatten. Vielen Dank für euren emotionalen und motivierenden Beistand während meines gesamten Studiums!

Inhaltsverzeichnis

1	EINLEITUNG	6
1.1	PROBLEMSTELLUNG	6
1.2	ZIELSETZUNG	7
1.3	STAND DER FORSCHUNG	7
1.4	ABGRENZUNG	9
1.5	VORGEHENSWEISE	9
2	STAND DER WISSENSCHAFT UND TECHNIK.....	11
2.1	SOZIALE MEDIEN	11
2.1.1	<i>Soziale Netzwerkplattformen.....</i>	<i>13</i>
2.1.2	<i>Facebook</i>	<i>13</i>
2.1.3	<i>Instagram.....</i>	<i>14</i>
2.1.4	<i>Twitter.....</i>	<i>16</i>
2.1.5	<i>Bedeutung und Einfluss der sozialen Medien.....</i>	<i>17</i>
2.2	SOCIAL BOTS	23
2.2.1	<i>Definition und Eigenschaften</i>	<i>23</i>
2.2.2	<i>Arten</i>	<i>26</i>
2.2.3	<i>Aufbau und Funktionsweise</i>	<i>28</i>
2.2.4	<i>Erkennung von Social Bots</i>	<i>31</i>
2.2.5	<i>Einsatzbereiche, Chancen und Gefahren.....</i>	<i>35</i>
2.3	KÜNSTLICHE INTELLIGENZ	39
2.3.1	<i>Social Bots und künstliche Intelligenz</i>	<i>40</i>
2.3.2	<i>Machine Learning</i>	<i>41</i>
2.3.3	<i>Natural Language Processing</i>	<i>43</i>
3	ENTWICKLUNG EINES EIGENEN TWITTER-BOTS	45
3.1	FRAMEWORKS	45
3.1.1	<i>Twitter API</i>	<i>47</i>
3.1.2	<i>Python, Node.js und Ruby</i>	<i>49</i>
3.1.3	<i>KI Frameworks.....</i>	<i>54</i>
3.2	KONZEPT - TWITTER-BOT „ANIMAL TWEETER“	57
3.3	IMPLEMENTIERUNG UND TEST - TWITTER-BOT „ANIMAL TWEETER“	59
3.3.1	<i>Erstellung eines Tweets.....</i>	<i>59</i>
3.3.2	<i>Retweeten, Liken und Kommentieren</i>	<i>62</i>
3.3.3	<i>Unterhaltung in den Direct Messages.....</i>	<i>67</i>

3.3.4	Unterhaltung in den Kommentaren	70
3.4	TEST DER ERKENNUNGSSOFTWARE „BOTOMETER“	72
4	AUSBLICK IN DIE ZUKUNFT	75
5	LITERATURVERZEICHNIS	77
6	ABKÜRZUNGSVERZEICHNIS	86
7	BILDVERZEICHNIS	87
8	TABELLENVERZEICHNIS.....	88
9	ANLAGENVERZEICHNIS.....	89
9.1	PYTHON QUELLCODES	89
9.2	JSON GRAMMATIK DATEI.....	96
9.3	DATEN DER VIRTUELLEN MASCHINE FÜR TENSORFLOW	97
9.4	KONZEPTBILD.....	99
9.5	TWEET - TESTSZENARIO 1.....	100
9.6	TWEET - TESTSZENARIO 2.....	101
9.7	RETWEET - TESTSZENARIO 1	102
9.8	RETWEET - TESTSZENARIO 2	107
9.9	UNTERHALTUNG IN KOMMENTAREN - TESTSZENARIO 1.....	112
9.10	UNTERHALTUNG IN KOMMENTAREN - TESTSZENARIO 2	113
9.11	UNTERHALTUNG IN DEN DIRECT MESSAGES - TESTSZENARIO 1	114
9.12	UNTERHALTUNG IN DEN DIRECT MESSAGES – TESTSZENARIO 2	114
10	SELBSTSTÄNDIGKEITSERKLÄRUNG.....	116
11	THESEN.....	117

1 Einleitung

1.1 Problemstellung

Die Kommunikation der Menschen findet fast nur noch über die sozialen Kanäle im Internet statt. In den sozialen Netzwerken, wie Facebook, Twitter oder Instagram, werden Bilder und Videos gepostet, andere Beiträge gelikt oder geteilt und die eigene Meinung wird mithilfe von Kommentaren publik gemacht. Doch den wenigsten Nutzern dieser Medien fallen die menschlichen Imitationen auf, welche als scheinbar menschliche Nutzer agieren. Die sogenannten Social Bots sind von Menschen entwickelte Programme, welche selbständig in den sozialen Medien ein menschliches Profil anlegen und mithilfe eines Algorithmus automatisch und eigenständig auf bestimmte Eingaben reagieren können [Gent18].

Das Zitat von Alan Turing: „We can only see a short distance ahead, but we can see plenty there that needs to be done [Turi50].“ verdeutlicht sehr gut, dass die Menschheit ständig für die Zukunft gerüstet sein muss. Social Bots werden mit der Zeit immer intelligenter und fortgeschrittener. Bislang können die Social Bots lediglich einen eigenen Post verfassen und auf bestimmte Stichworte einen Kommentar erstellen. Momentan besteht eine reale und dynamische Kommunikation mit Menschen nur vereinzelt und noch nicht sehr umfangreich. Es ist jedoch nur noch eine Frage der Zeit, bis auch dies möglich ist, da die Social Bots kontinuierlich weiterentwickelt werden. Auch die stetige Erforschung und Entwicklung der künstlichen Intelligenz (KI) wird immer schneller. Um menschlicher zu wirken, setzen Social Bots immer mehr menschliche und KI bezogene Methoden ein. Dadurch können sie eigenständige dynamische Diskussionen führen, ohne enttarnt zu werden [Bund18].

All diese Funktionen der Social Bots können zur Manipulation und Täuschung von Personen oder Personengruppen führen, da die Menschen oft nicht wissen, dass sie mit einer Maschine kommunizieren. Die Social Bots werden dementsprechend kaum als Maschinen wahrgenommen. Meinungen und Trends können beeinflusst und Nutzer der sozialen Plattformen ausgespäht und getäuscht werden. Aber auch Straftaten, wie der Betrug oder Identitätsdiebstahl, wären bei dieser zügigen Entwicklung möglich

[Bund18]. Dementsprechend müssen die Erkennungsmethoden neugestaltet und ein Gefahrenbewusstsein in der Kommunikation mit Social Bots geschaffen werden.

Die Thesis wird sich dahingehend mit der Entwicklung, der Erkennung und dem Gefahrenpotenzial von Social Bots im Zusammenhang mit der künstlichen Intelligenz befassen.

1.2 Zielsetzung

Das Ziel dieser Thesis soll sein, einen Social Bot für das soziale Netzwerk Twitter zu programmieren, welcher mithilfe von maschinellen Lernmethoden eigenständig einen Beitrag posten, automatisiert Nachrichten an einen anderen menschlichen Nutzer verschicken und auf Kommentare und Nachrichten selbstständig antworten kann. Somit soll eine reale und echt wirkende Kommunikation zwischen Mensch und Maschine vorgetäuscht werden. Es soll gezeigt werden, welche Gefahren und potenziellen Angriffe durch diese Programme möglich sind. Es sollen einige Erkennungsprogramme vorgestellt und ein System auf seine Funktionalität und Einsatzfähigkeit für den entwickelten Social Bot getestet werden. Mit der Thesis soll herausgestellt werden, was momentan in den Bereichen Social Bots und künstlicher Intelligenz möglich ist und ob die aktuellen Erkennungsmethoden ausreichen, um Social Bots zu erkennen. Das Ergebnis dieser Arbeit soll den Nutzer von sozialen Medien für dieses Thema sensibilisieren und das Gefahrenbewusstsein stärken. Zudem soll ein Impuls für die Weiter- und ggfs. Neuentwicklung von Erkennungssoftware gegeben werden.

1.3 Stand der Forschung

Das Ergebnis der Studie im Jahr 2016 von Incapsula zeigt, dass über die Hälfte des Internetverkehrs auf Bots zurückzuführen ist. Knapp 30% dieser Bots seien sogenannte Bad Bots, welche auf Manipulation, Täuschung und Betrug spezialisiert sind [Zeif17].

Mit der US-Wahlkampagne 2016 und dem US-Wahlkampf zwischen Hillary Clinton und Donald Trump wurde erstmals in der Öffentlichkeit über das Thema der Social Bots kritisch diskutiert. Nach einer Studie der Oxford Universität hatten eine Vielzahl von

Bots in dem sozialen Netzwerk Twitter Nachrichten zur Unterstützung der beiden Kandidaten veröffentlicht. Jeder vierte Tweet für Hillary Clinton und jeder dritte Tweet für Donald Trump wurde von einem Social Bot vorgetäuscht. Zudem waren mehrere Millionen Follower der Kandidaten keine Menschen, sondern Bots, welche Tweets teilten und versuchten für die jeweilige Partei Stimmen zu gewinnen [SoFr16, TeHa18]. Doch nicht nur im US-Wahlkampf wurde die Gefahr von Social Bots deutlich, sondern auch beim Brexit-Votum im Juni 2016. An den diversen Diskussionen in den sozialen Medien nahmen auch Social Bots teil. Laut der Studie von The Independent gab es eine Reihe Social Bots, die für den Austritt waren und diese Meinung in den sozialen Medien öffentlich machten [Sull17].

Diese Studien zeigen die Existenz und das Gefährdungspotenzial dieser Programme, jedoch ist es noch nicht wissenschaftlich belegt, inwiefern Bots ausschlaggebend für die Manipulation der eigenen Meinung oder bei politischen Entscheidungen sind. Zusätzlich gibt es bisher nur einige wenige Programme, welche sich auf das Themengebiet der Erkennung von Social Bots spezialisiert haben. Zu diesen zählen u.a. die Software „Bot or Not“ oder auch die Webseite „Botwatch.de“, welche ein besonderes Augenmerk auf Tweets zu Polit-Talkshows legen. Da jedoch die Entwicklung von Social Bots immer schneller und effektiver wird, müssen diese Erkennungsprogramme stetig optimiert und erweitert werden.

Bislang haben die Social Bots die Macht andere Nutzer im Internet zu beeinflussen und Fake News zu veröffentlichen. Es ist momentan schwierig abzuschätzen und noch nicht ausreichend erforscht, für welche schadhaften Zwecke intelligente Bots eingesetzt werden können. Sie können gerade dann eine Gefahr sein, wenn sie mithilfe einer täuschend echt wirkenden Kommunikation den Menschen als potenzielle Schwachstelle erkennen und ihn für mögliche Angriffe instrumentalisieren, wie bspw. zum Installieren einer Schadsoftware oder dem Versenden von manipulierten Links [Ki17].

Durch die aktuellen Forschungsergebnisse wird deutlich, dass besonders die Gefahren von Social Bots, welche mit Methoden der künstlichen Intelligenz auftreten, noch sehr wenig untersucht sind.

1.4 Abgrenzung

In dieser Thesis wird weder eine eigene Erkennungssoftware für Social Bots entwickelt noch eine Empfehlung für eine Software ausgesprochen. Zudem werden nicht alle Erkennungsprogramme aufgezeigt, da der Schwerpunkt der Arbeit auf der Entwicklung eines Twitter-Bots liegt. Des Weiteren wird nicht im Detail auf alle Methoden der künstlichen Intelligenz eingegangen. Es wird lediglich ein Überblick über mögliche Methoden und Algorithmen dargelegt. Der entwickelte Social Bot wird ausschließlich für die soziale Plattform Twitter programmiert. Der entwickelte Bot kann selbstständig eigene Tweets verfassen, Beiträge anderer Nutzer veröffentlichen, diese kommentieren und liken. Weiterhin kann er auf Kommentare und direkte Nachrichten anderer Nutzer reagieren und eine Unterhaltung führen.

1.5 Vorgehensweise

Die Thesis gliedert sich in drei Hauptabschnitte.

Der erste Abschnitt beschäftigt sich mit den theoretischen Hintergründen und den Fachbegriffen in der Wissenschaft. Er umfasst die Themen soziale Medien, Social Bots und die künstliche Intelligenz. Es werden die sozialen Medien mit den verschiedenen sozialen Netzwerken und der mögliche Einfluss derer auf die Meinung der Nutzer beschrieben. Das Kapitel Social Bots legt die unterschiedlichen Arten sowie den Aufbau und die Funktionsweise, die Erkennung, die Gefahren, aber auch die Chancen dar. Hierbei wird ein Augenmerk auf die potenziellen Angriffe durch Social Bots gelegt. Schlussendlich wird im ersten Abschnitt das Phänomen künstliche Intelligenz vorgestellt. Es wird ein Überblick über das maschinelle Lernen, die Verarbeitung der natürlichen Sprache und die verschiedenen Einsatzbereiche gegeben. Zusätzlich wird in diesem Abschnitt der Zusammenhang zwischen künstlicher Intelligenz und Social Bots untersucht.

Der zweite Abschnitt befasst sich mit der Entwicklung eines Social Bots für den Kurznachrichtendienst Twitter, welcher automatisiert Nachrichten bzw. Kommentare erstellen, auf Kommentare und Nachrichten reagieren und somit mit anderen Nutzern kommunizieren kann. Hierbei stehen die verschiedenen Frameworks, die API von

Twitter, die Entwicklungsumgebung und das Design und die Implementierung des Bots mit verschiedenen Tools des maschinellen Lernens im Vordergrund.

Im letzten Abschnitt wird mithilfe des erstellten Bots ein zuvor dargestelltes Erkennungsprogramm auf ihre Funktionalität analysiert.

Abschließend folgen eine Zusammenfassung und ein Ausblick in die Zukunft mit weiteren Potenzialen, Einsatzgebieten und Chancen von Social Bots.

2 Stand der Wissenschaft und Technik

Im folgenden Abschnitt der Arbeit werden vorhandene Theorien und Methodiken der Wissenschaft und Technik im Zusammenhang mit sozialen Medien, Social Bots und künstlicher Intelligenz vorgestellt. Somit werden die Grundlagen für die Implementierung eines Twitter-Bots geschaffen.

2.1 Soziale Medien

Wir vernetzen uns mit anderen, posten und teilen Ereignisse aus unserem Leben, geben unsere Meinung öffentlich ab und verabreden uns über digitale Nachrichten.

Die Geschichte der Sozialen Medien liegt nicht nur ein paar Jahre, sondern mehrere Jahrzehnte zurück. Denn in den 1990er Jahren waren bereits Anfänge des Internets zu erkennen. Das Internet entwickelte sich zu einem Gegenspieler der Massenmedien wie dem Fernsehen, dem Radio oder der Zeitung. Mit seinen universell einsetzbaren Kanälen ermöglicht das Internet, eine zeitversetzte, aber auch zeitgleiche interpersonale Kommunikation. Zudem kann eine Interaktion zwischen Mensch und Maschine stattfinden. In den 2000er Jahren wurde das multimediale Internet geschaffen, welches uns die Möglichkeit bietet, eine außerordentliche Vielfalt verschiedener Medienangebote nutzen zu können. Seit Anfang der 2010er Jahre nahm die Verbreitung von Tablets und Smartphones stetig zu. Günstige Flatrate-Tarife und Breitbandverbindungen unterstützen diesen Trend. Somit können die multimedialen Dienste von jedem Ort aus und zu jeder Uhrzeit genutzt werden. Diese Infrastruktur ermöglichte die Erschaffung der Sozialen Medien [Schm18].

Der Begriff der sozialen Medien wird nach Prof Dr. Oliver Bendel wie folgt definiert [Bend18]:

„Soziale Medien (Social Media) dienen der – häufig profilbasierten – Vernetzung von Benutzern und deren Kommunikation und Kooperation über das Internet. Das Web 2.0, das Mitmachweb, ist wesentlich durch sie bestimmt.“

Dieser Terminus steht grundsätzlich für eine Vielzahl an digitalen Anwendungen und Medien aus dem Web 2.0. Die Nutzer dieser Technologien interagieren mittels Nachrichten und Kommentaren miteinander. Zudem gestalten sie Inhalte dieser Plattformen aktiv mit und verbreiten Nachrichten und Meinungen. Dies wird oft als User Generated Content beschrieben. Der Nutzer wird hierbei selbst zum Designer und Entwickler [Evs1].

Das Ziel sozialer Medien ist es, sich mit anderen Personen, wie Freunden und Bekannten, aber meist auch Fremden auszutauschen, zu kommunizieren oder gemeinsam an bspw. einem persönlichen oder beruflichen Projekt zu arbeiten. Hierbei kann sowohl Text als auch Sprache und Bild verwendet werden [Bend18].

Somit kann festgehalten werden, dass soziale Medien durch zwei wesentliche Merkmale charakterisiert sind [Schm18]:

Digitale Veröffentlichung und Bearbeitung von Inhalten

Soziale Medien nutzen nicht nur vorhandene Inhalte und Informationen, sondern können diese mithilfe von ihren Nutzern selbst generieren. Sie können folglich Inhalte im Internet zur Verfügung stellen und diese gleichzeitig bearbeiten bzw. neu erschaffen [Schm18].

Austausch mit anderen

Soziale Medien geben zudem die Möglichkeit, sich mit anderen auszutauschen, soziale Beziehungen zu pflegen oder neue Bekanntschaften zu knüpfen [Schm18].

Es gibt eine Reihe verschiedener Arten von Sozialen Medien, unter anderem Soziale Netzwerkplattformen, Multimediaplattformen, Blogs oder auch Wikis. Diese können jedoch nicht immer exakt voneinander abgegrenzt werden. Die Thesis bedient sich dem allgemeinen Begriff der Sozialen Netzwerke, da diese die Grundlage für Multimediaplattformen und Blogs bilden. Daher wird der Begriff „Soziale Netzwerke“ als Oberkategorie verwendet.

2.1.1 Soziale Netzwerkplattformen

Die sogenannten „Social Networks“ sind Webseiten im Internet, mit welchen eine digitale Vernetzung und ein Treffen in der virtuellen Welt stattfinden kann. Dies erfolgt mit Freunden oder auch Fremden, die die gleichen Interessengebiete haben. Hierbei können unter anderem persönliche Informationen ausgetauscht, miteinander geteilt und diskutiert werden. Auf sozialen Plattformen können sich Nutzer anmelden und ein eigenes individuelles Profil mit bestimmten persönlichen Inhalten, wie Interessen oder fachlichen Kompetenzen und einem Profildfoto erstellen [Vert19b]. Mithilfe von themenbezogenen Gruppen oder Direktnachrichten können Unterhaltungen mit den eigenen Kontakten oder noch unbekannten Personen stattfinden [Schm18].

Somit wird deutlich, dass „Social Networks [...] Gemeinschaften innerhalb des Internets [sind]“ [Vert19b].

Ob Katzen- oder Hundeliebhaber, Student, Schüler, Onlinespieler oder Handwerker, für jedes Interessensgebiet gibt es das passende soziale Netzwerk, um sich mit Gleichgesinnten auszutauschen oder selber neue Inhalte und Informationen zu erstellen. Somit gibt es interessenbezogene Plattformen, aber auch nationale sowie internationale Netzwerke [Vert19b].

Neben LinkedIn oder Xing, welche sich auf die beruflichen Interessen spezialisiert haben, ist die wohl bekannteste und meist genutzte Plattform Facebook.

2.1.2 Facebook

Facebook wurde 2004 von dem Harvard Studenten Mark Zuckerberg gegründet, welcher mittlerweile zu den reichsten IT-Unternehmern in den USA zählt. Anfangs diente Facebook lediglich zur Kommunikation der Studenten des Campus untereinander. Im September 2006 wurde die Plattform dann für die Öffentlichkeit zugänglich und im Jahr 2008 ging die deutsche Version online [KnRi10].

Der Börsenwert des Internetunternehmens liegt (Stand Mai 2018) bei 538 Milliarden US-Dollar und befindet sich weltweit unter den 10 wertvollsten Internetunternehmen [Klei18]. Zudem verzeichnet Facebook im 1. Quartal 2019 weltweit rund 2375 Millionen

aktive Nutzer [Face19]. Auf die am Anfang fokussierte Zielgruppe der eher jüngeren Nutzern, meist Studenten, lässt sich mittlerweile abzeichnen, dass die Gruppe der 25-34 Jährigen Facebook am Meisten nutzen [WHD19b]. Somit gewinnt Facebook in unterschiedlichen Bevölkerungskreisen immer mehr an Bedeutung.

Wie bei den meisten Social Networks kann ein Nutzer auch auf Facebook ein eigenes Profil mit Informationen über sich selbst in Form von Text, Bildern oder Videos erstellen, die dann auf der sogenannten Chronik erscheinen. Zusätzlich gibt es die Möglichkeit, eine Statusmeldung zu generieren, welche Informationen über derzeitige Aktivitäten oder Stimmungen enthält. Eigene Veröffentlichungen können von anderen Nutzern kommentiert oder mithilfe des „Gefällt mir“-Buttons als positiv oder des „Gefällt mir nicht“-Buttons als negativ bewertet werden. Aktuell ist es so, dass statt dem „Gefällt mir“- Button die Möglichkeit besteht, verschiedene Smileys auszuwählen, die eine Stimmung kennzeichnen wie z.B. herzlich, lustig, traurig oder wütend. Ein interner, nicht öffentlicher Chat und ein Postfach mit direkten Nachrichten an Nutzern geben die Möglichkeit, mit anderen privat zu kommunizieren. Wenn sich die Nutzer über bestimmte Themen, Anlässe oder Interessen mit einer Vielzahl von anderen Menschen unterhalten wollen, gibt es die Facebook-Gruppen, denen beigetreten werden kann [KnRi10].

2.1.3 Instagram

Instagram kommt aus dem Englischen und bildet sich aus zwei Hauptwörtern, **instant** camera = Sofortbildkamera und **telegram**= Telegram [Dude19]. Gegründet wurde Instagram 2010 und gehört seit August 2012 zu der Facebook Incorporated [Stat19c]. Instagram gehört zu der Untergruppe der Multimediaplattformen, welche einen Fokus auf spezialisierte Inhalte legen [Schm18].

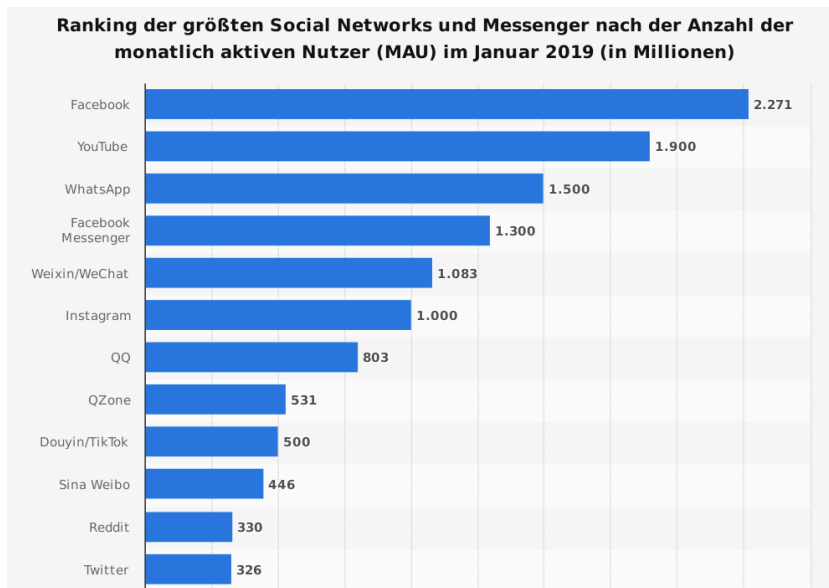


Bild 1: Ranking der größten Social Networks und Messenger [WHD19a]

Zudem gehört Instagram mit seinen 1000 Millionen monatlich aktiven Nutzern (siehe Bild 1) zu den 10 größten sozialen Netzwerken und Messengern [WHD19a].

Die kostenfreie App ermöglicht es, dem Nutzer ausschließlich Fotos und Videos zu erstellen, mit einer Vielzahl von Filtern zu versehen und es auf dem eigenen Profil für die breite Öffentlichkeit oder nur für enge Freunde hochzuladen. Zudem können diese erstellten Beiträge von anderen Nutzern gelikt bzw. geherzt und kommentiert werden [Stat19c]. Charakteristisch für dieses soziale Netzwerk ist die Form der Beiträge. Fotos und Videos können lediglich in einem quadratischen Format aufgenommen und hochgeladen werden.

2016 fügte Instagram eine weitere Funktion hinzu, das Erstellen und Hochladen einer Instagram-Story. Diese sogenannte Story kann mehrere Fotos oder Videos, welche einzeln maximal 15 Sekunden lang sein dürfen, enthalten und verschwindet dann nach 24 Stunden. Die Fotos und Videos kann der Nutzer mit verschiedensten Filtern, Emojis oder Texten bearbeiten [Stat19c].

Wie Facebook hat auch Instagram eine Startseite, in welcher sich der Nutzer Beiträge von seinen abonnierten Instagram-Seiten ansehen kann. Dies geschieht jedoch nicht nur nach Aktualität der Beiträge so wie es bei Facebook üblich ist. Der Algorithmus von Instagram orientiert sich an drei Faktoren. Zum einen analysiert Instagram das Verhalten

seiner Nutzer und stellt heraus, welche bestimmten Themen das Interesse wecken. Zusätzlich kommt die Aktualität der Beiträge hinzu. Je aktueller der Post ist, desto wahrscheinlicher wird er als Erstes auf der Startseite angezeigt. Abschließend ist es wichtig, mit welchen anderen Accounts der Nutzer kommuniziert und interagiert, wie oft er bspw. einen Account öffnet und Bilder likt. Auch die Anzahl der abonnierten Accounts, die Häufigkeit, wie oft der Nutzer die App öffnet und wie lange er Instagram nutzt, spielen eine entscheidende Rolle [Bett19].

Neben den Instagram-Stories und dem individuellen Algorithmus definiert sich Instagram ebenso durch sogenannte Hashtags. Mit dem Setzen einer Raute vor einem Wort existiert eine Verlinkung zu ähnlichen Inhalten mit demselben Wort. Somit können die Bilder oder Videos durch das Suchen von bestimmten Schlagwörtern schneller von anderen gefunden werden. Der Vorreiter dieser Hashtags war jedoch im Jahr 2008 der Kurznachrichtendienst Twitter [Vett15].

2.1.4 Twitter

Der Microblog Twitter ist ein soziales Netzwerk, welches 2006 in den USA gegründet wurde. Twitter ermöglicht es dem Nutzer eine Statusmeldung, einen sogenannten Tweet (deutsch: Gezwitscher), welcher auf 140 Zeichen begrenzt ist, zu verfassen. Seit 2017 ist es möglich, Tweets in einer Länge von maximal 280 Zeichen, ausgeschlossen davon sind angehängte Fotos oder Videos, hochzuladen. Der Microblogging-Dienst verzeichnet im 3. Quartal 2018 326 Millionen monatlich aktive Nutzer. Im E-Commerce Bereich steigt die Nutzung von Twitter [Stat19a]. „So hatten laut einer Erhebung des EHI Retail Institutes und Statista im Jahr 2018 beispielsweise rund 74,4 Prozent aller Online-Shops einen Twitter-Account. [Stat19a]“

Auch Twitter besteht aus einer Startseite, der sogenannten »*timeline*«. Sie informiert in Echtzeit über Tweets und Updates von allen gefolgt Personen. Dabei kann jedoch entschieden werden, ob die eigenen Beiträge allen Nutzern oder nur einer bestimmten Gruppe von Personen verfügbar gemacht werden sollen. Tweets von anderen Nutzern können abonniert und somit auch dem Kanal gefolgt werden. Mithilfe eines »@«-Zeichens vor einem beliebigen Nutzernamen, kann explizit an diesen Nutzer ein Tweet verfasst werden, welcher dann aber auch für alle anderen Nutzer sichtbar ist. Auch

Hashtags können als Verlinkung zu anderen dazu passenden Tweets verstanden werden. Dies ermöglicht das Auffinden spezieller Themen und Gruppen [Schm18].

Der Kernpunkt von Twitter besteht in dem Retweeten von Beiträgen. Hierbei werden Accounts zitiert oder Tweets von anderen Nutzern auf dem eigenen Account wiederholt. Dies unterstützt die virale Verbreitung von Inhalten, indem andere Nutzer auf den eigenen Tweet mit Vermerk auf den Urheber reagieren und somit den Inhalt wiedergeben [Vert19a].

2.1.5 Bedeutung und Einfluss der sozialen Medien

Die Mehrheit – danach orientieren sich die meisten Menschen. Jedoch kann diese wahrgenommene Mehrheit auch verzerrt sein. Soziale Netzwerke begünstigen dieses Phänomen. Sie beeinflussen teilweise unsere Meinung und unsere Wahrnehmung [LYW16].

Social Media Plattformen generieren selbst keine eigenen Inhalte. Sie stellen für uns ausschließlich Inhalte und Informationen von anderen Nutzern oder Nutzergruppen bereit. Somit haben sie einen weitreichenden Einfluss darauf wie uns bestimmtes Wissen erreicht. Sie vermitteln uns also Informationen zu bestimmten Sachverhalten entweder oberflächlich oder detailliert. Diese können beiläufig erfahren oder durch gezieltes Suchen und Recherchieren gefunden werden. Aktuelle Nachrichten aus der ganzen Welt können durch Social Media schneller an uns herangetragen werden. Neben dem Wissen, welches Soziale Netzwerke offenlegen, vermitteln sie Bewertungen zu bestimmten Themen. Zusammenfassend kann gesagt werden, dass soziale Medien die gesellschaftliche Relevanz eines Ereignisses herausstellen, dieses in einen Zusammenhang mit übergeordneten Themen bringen, schließlich unterschiedliche Haltungen aufzeigen und aufklären, inwiefern die eigene Meinung in die Minderheit oder Mehrheit passt. Hierbei werden bereits die ersten Impulse zur Bildung der eigenen Meinung zu einem bestimmten Thema gesetzt [Schm18].

„In vielen Fällen gilt aber, dass die eigentliche Meinungsbildung, also die Formierung, Stabilisierung oder Veränderung meiner eigenen Haltung zu einer Streitfrage oder einem Thema, sich erst im Austausch mit anderen Menschen abspielt.“ [Schm18]

Die eigene Meinung mit der anderer abzugleichen, ist ein innerlich festsitzender und menschlicher Impuls und ein großes Bedürfnis. Menschen müssen sich ein Bild der leitenden Meinung ihres Umfeldes machen.

Menschen vertrauen besonders den Personen in ihrem sozialen Umfeld. Wie im realen Leben hängt das eigene Verhalten oft von den Handlungen der engsten Vertrauten ab. Wir beobachten Charaktereigenschaften oder Haltungen unsere Familien und Freunde und gehen automatisch davon aus, dass es sich hierbei um eine überlegende Meinung bzw. Interesse handelt. Obwohl der Rest der Welt hierzu ganz anders steht [Münc15]. Dieses Phänomen wird als Mehrheitsillusion betitelt [LYW16].

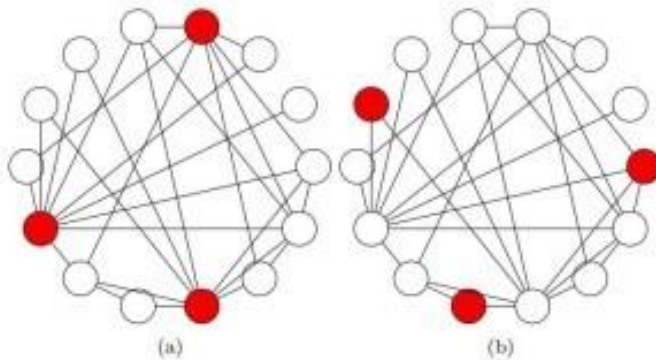


Bild 2: Die Mehrheitsillusion [LYW16]

Die drei roten Punkte (siehe Bild 2) stellen die Minderheitsmeinung da, also die Meinung, die nicht sehr oft in der gesamten Gesellschaft vertreten ist. Im Schaubild A haben die drei (roten) Menschen sehr viele Kontakte und würden hiermit auch mehr Menschen mit ihren Ansichten erreichen. Die drei (roten) Menschen im Schaubild B haben dagegen nur sehr wenige Kontakte [Bähr15]. Nehmen wir das Beispiel, dass die Mehrheit der Menschen denkt, dass Computerspiele legitim sind und die Minderheit aber sagt, dass diese abgeschafft werden sollten. Wenn nun diese beiden Schaubilder betrachtet werden, wächst schnell die Erkenntnis, dass die Kontakte im Schaubild A den Wunsch nach Abschaffung von Computerspielen als vorherrschende Meinung halten. Da hier die drei roten Punkte eine viel höhere Netzwerkverknüpfung haben, sehen ihre Kontakte diese Meinung viel öfter als die tatsächlich vorherrschende Meinung. Auf dem Schaubild B ist dies umgekehrt. Hier erhält keiner der Kontakte den Eindruck, dass die meisten Menschen für ein Verbot von Computerspielen sind.

Nutzer, die viele Follower haben, sind meistens sehr aktiv und posten regelmäßig. Sie haben eine viel größere Reichweite als andere Nutzer mit weniger Kontakten. Inhalte dieser Nutzer werden demnach viel stärker verbreitet oder auch zitiert als die Eigenen. Umso mehr Kontakte bzw. Follower ein Account hat, umso größer ist die Informationsmacht. Eine Mindermeinung kann also mithilfe von sogenannten Meinungsführern mit großen Netzwerken den Eindruck machen als sei sie die Massenmeinung [Münc15]. Die sozialen Netzwerke haben ihre eigenen Meinungsführer entwickelt, die Influencer (deutsch: Beeinflusser). Diese stellen eine wichtige Wissensquelle für Nutzer dar. Sie sind überwiegend auf Lifestyle Themen spezialisiert, haben meist mehrere Millionen Follower und sind für junge Menschen Vorbilder, die bestimmte Leitwerte vermitteln [Schm18]. Influencer haben einen bestimmten Einfluss auf ihre Community. Dies zeigt eine Umfrage vom Jahr 2019. 220 von 582 Befragten gaben an, bereits ein Produkt gekauft zu haben, weil ein bestimmter Influencer dieses Produkt vorgestellt hat. Hervorgehoben werden muss hierbei, dass insbesondere die jüngeren Nutzer (18-29 Jahre) zu 49% schon einmal ein Produkt aufgrund der Werbung eines Influencers gekauft haben [Stat19b]. Studien zeigen, dass besonders die jüngeren Menschen Soziale Netzwerke nutzen. 63% der Facebook Nutzer und 50% der Instagram Nutzer sind 14-29-Jährige [ARZD19].

Doch nicht nur die Meinungsführer selbst, greifen in den Prozess der eigenen Meinungsbildung ein. Auch die sozialen Plattformen an sich sind sehr lenkend. Im Folgenden werden einige potenzielle Einflussfaktoren genannt, die zu einer Manipulation der eigenen Meinung oder sogar des eigenen Verhaltens eingesetzt werden können.

Algorithmische Selektion

„Wir wissen besser als Du, was gut für dich ist [Jane17].“ Diese Haltung zeigt deutlich, wie uns soziale Netzwerke in bestimmte Richtungen leiten wollen. Wie bereits beschrieben, werden mithilfe bestimmter Algorithmen Inhalte für den Nutzer von Social Media individuell selektiert, welche dann auf der jeweiligen Startseite oder dem jeweiligen Feed angezeigt werden. Diese Personalisierung von Inhalten hat zumeist das Ziel, die Relevanz bestimmter Themen zu steigern. Auch eine wachsende Informationsflut kann mithilfe dieser algorithmischen Filterung bezwungen werden.

Jedoch haben die Verfahren algorithmischer Selektion oft den Nachteil der Intransparenz. Die Intermediäre, wie Facebook oder Google, sind weitgehend unübersichtliche und segregierte virtuelle Systeme, welche ständig erweitert werden. Die Abgrenzung, wann die Selektion beginnt, ist selbst für Fachleute von außen nicht klar erkennbar. Daher ist es nicht auszuschließen, dass die Algorithmen veraltet, falsch oder gehackt sind [Jane17].

Astroturfing

Astroturfing bezeichnet Täuschungsstrategien, mit denen Großkonzerne oder Politiker versuchen ihre Äußerungen wie die eines „einfachen“ Bürgers erscheinen zu lassen. Astroturfing wurde schon in den Medien vor dem Internet wie Fernsehen oder Radio angewandt. Heutzutage erfolgt dies jedoch auch in den sozialen Netzwerken z.B. via *Social Bots* [Jane17].

Disintermediation

Disintermediation bedeutet, dass besonders bei politischen Themen die klassischen Intermediären wie Fernsehen oder Radio in der Kommunikation mit anderen Menschen umgangen werden. Hierbei sollen die klassischen Medien falsche Nachrichten verbreiten und nur einseitig berichten. Das Ziel der Disintermediation ist, das Vertrauen und die Glaubwürdigkeit in die klassischen Medien zu senken und die größte Zahl der Menschen dazu zu bringen ihre Informationen aus dem Internet (insb. Soziale Netzwerke) zu beziehen [Jane17].

Echoklammern

Die Echoklammern fördern stark die Gruppen- und Meinungsbildung von Menschen insbesondere in den sozialen Medien. Gleichgesinnte finden sich zusammen und bilden Gruppen. Die eigene Meinung kann dadurch gefördert werden. Mit Gleichgesinnten in einer Gruppe kann die vertretende Meinung untereinander bestätigt werden, ohne dabei abweichende Meinungen mit einzubeziehen. Jedoch ist diese sogenannte Clusterbildung in den sozialen Netzwerken bislang noch sehr umstritten. Einige Wissenschaftler sind der Ansicht, dass die sozialen Medien und die einhergehende Digitalisierung das

Phänomen der Echoklammern begünstigen, andere sehen dies jedoch sehr kritisch [Jane17].

Filterblase

Die Startseite der meisten sozialen Netzwerke umfasst immer wiederkehrende Inhalte, die sich auf die vergangenen Benutzerpräferenzen beziehen. Das bedeutet, dass dem Nutzer meist nur eine bestimmte Ansicht vorgeschlagen wird. Das Modell der Filterblase setzt hier ein und beschreibt, dass mithilfe dieser personalisierten Feeds, Aussagen und Auffassungen verfestigt werden, ohne dabei andere abweichende Meinungen wahrzunehmen. Dies könnte den Aufbau einer Radikalisierung fördern. Auch dieser Einflussfaktor ist noch nicht vollständig wissenschaftlich bestätigt worden [Jane17].

Fake Follower

Die Anonymität im Internet hat sowohl Vorteile als auch Nachteile. Der Zeichner Peter Steiner hat in der New Yorker Zeitschrift einen Cartoon (Siehe Bild 3) veröffentlicht, welcher auf vereinfachte Weise die Anonymität im Internet ausdrückt. Ohne diese Möglichkeit bliebe wahrscheinlich die Meinungsfreiheit auf der Strecke. Jedoch kann diese Anonymität ebenso für Betrug oder Täuschung missbraucht werden. Keinem Nutzer von sozialen Medien fällt sofort auf, dass Bewertungen von Agenturen geschrieben, Follower, Likes oder gar ganze Twitter-Konten verkauft und gekauft wurden [Jane17].



Bild 3: Cartoon von Peter Steiner aus dem Jahre 1993 [Jane17]

Fake News

Falschmeldungen gibt es schon eine sehr lange Zeit, auch bevor es das Internet gab. Jedoch wurde mit der wachsenden Digitalisierung, die schnelle und einfache Ausbreitung durch soziale Netzwerke und die dazugehörige Reichweite, die Möglichkeit der Täuschung, Lügen und Propaganda immer leichter. Fake News stellen die Glaubwürdigkeit von Medien in Frage und können dank des Vertrauens in soziale Netzwerke an mehr Einfluss gewinnen. Fake News und damit einhergehende Manipulationen und Täuschungen können durch Social Bots verbreitet werden [Jane17].

Im nachfolgenden Kapitel wird detailliert und umfassend auf das Thema der Social Bots eingegangen.

2.2 Social Bots

In diesem Kapitel wird die mit Social Media kommende Technologie der Social Bots näher erläutert. Dieser Abschnitt beschreibt die verschiedenen Arten von Social Bots sowie die Funktionsweise und die technische Voraussetzung im Detail. Es werden die Fragen, inwiefern Social Bots Einfluss auf die Nutzer der sozialen Medien nehmen, neuartige Gefahren aufweisen aber auch wie nützlich sie sein können, geklärt. Auch die Erkennung solcher Programme und ihre Schwellwerte werden dargestellt. Dieses Kapitel wird als Vorlage für die Entwicklung und Implementierung des Social Bots für Twitter dienen.

2.2.1 Definition und Eigenschaften

Der Begriff „Bot“ ist die englische Kurzform von „Robot“ (deutsch: Roboter) und bezeichnet autonom handelnde Programme, welche Aufgaben automatisiert durchführen [Reut19]. Dabei muss aber beachtet werden, dass bspw. Programme oder Skripte wie Suchmaschinen, die das Internet durchsuchen oder Schadsoftware, die Computer beschädigen und sich verselbstständigen, ebenso Bots genannt werden. Jedoch wird der Begriff aktuell in Bezug auf soziale Medien für automatisierte Profile genutzt. Hierbei können sie sehr praktisch sein und ein berechtigtes Dasein haben wie zum Beispiel Chat- oder Assistenz-Bots [Hege16]. Sie können aber auch für das Manipulieren und das Beeinflussen von Meinungen und Handlungen in sozialen Netzwerken verantwortlich sein.

Die sogenannten Social Bots verschleiern ihre wahre Identität, indem sie reale Menschen in sozialen Netzwerken, wie dem Kurznachrichtendienst Twitter, imitieren und den Nutzern vorgeben, sie wären echte Menschen. Mithilfe dessen können sie automatisiert Inhalte verbreiten, Trends setzen und das Kommunikationsverhalten in einem gewissen Maß steuern ohne Kenntnis der anderen Nutzer [Inst18].

Social Bots werden immer häufiger in der Wirtschaft und Politik eingesetzt. Mithilfe des Microtargeting werden gezielt bestimmte einzelne Personen oder kleine Gruppen angesprochen. Dadurch wird versucht, das Konsum- bzw. Wahlverhalten zu beeinflussen. Hierbei werden bspw. individuelle Profile aus großen Datensätzen (Big

Data) erstellt, damit Unternehmen herausfinden können, wie sie ihre Zielgruppe am besten erreichen und wie sich deren Vorlieben verändern. Im politischen Umfeld können Parteien mithilfe von Bewegungsprofilen und Adressen abschätzen, wo sich ihre Zielwähler am häufigsten befinden. Auch persönliche Informationen, wie Lieblingessen, Kaufverhalten oder Handytarife ermöglichen es, gezielt persönliche Werbung zu schalten [Inst18].

Social Bots bedienen sich des Prinzips der sozialen Bewährtheit. Dieses Prinzip wird nach dem Psychologen Robert Cialdini auch Social Proof genannt. Dabei geht es darum, dass Menschen sich vom Grund auf bei der Bewertung von Informationen und Wissen an anderen Menschen, meist aus dem eigenen sozialen Umfeld, orientieren. Als Beispiel wäre anzuführen, dass bei Comedyserien meist „inszenierte“ Lacher reingeschnitten werden. Der Zuschauer bekommt das Gefühl, dass die Serie witzig ist, obwohl dies ohne den Fake-Lachern nicht zutreffen würde. Menschen neigen somit meist dazu sich der Mehrheit anzuschließen. Menschen, die uns ähnlich sind, im besonderen Maße unsere Freunde oder Familien, ahmen wir eher nach. Dies geschieht auch im Internet. Hierbei setzen Social Bots ein und manipulieren die Mehrheitsmeinungen verdeckt. Sie werden dazu genutzt, Emotionen und Ansichten eine große Bedeutung beizumessen. Auch wenn diese Beeinflussung sehr große Gefahren in sich birgt, sind Social Bots noch nicht gesetzlich reguliert [SaZy18].

Es existieren derzeit zwei verschiedene Ausprägungen von Social Bots. Die einfachen Social Bots erfassen Stichworte und Schlüsselbegriffe, wie z.B. Hashtags, und reagieren darauf mittels Kommentare, Likes oder Retweets. Einige von ihnen imitieren das Verhalten realer Nutzer, indem sie Aktivitäten zu unterschiedlichen Zeiten und in unterschiedlichen Arten vortäuschen. Komplexe Social Bots analysieren das Kommunikationsverhalten und können somit komplexe Dialoge führen. Momentan überwiegen die einfachen Social Bots [Ki17].

Laut der PWC Studie 2018 informiert sich jeder Vierte über Social Media zu aktuellen Ereignissen. [Pric18a] Die dafür meist genutzten Sozialen Netzwerke sind Facebook mit 80%, Youtube mit 46%, WhatsApp mit 42% und Twitter mit 32%. Dabei werden hauptsächlich zu 54% die Kanäle und Profile von Printmedien, zu 50% den öffentlich-

rechtlichen Sendern und zu 47% die von Meinungsbildnern bevorzugt (Siehe Bild 4) [Pric18a].

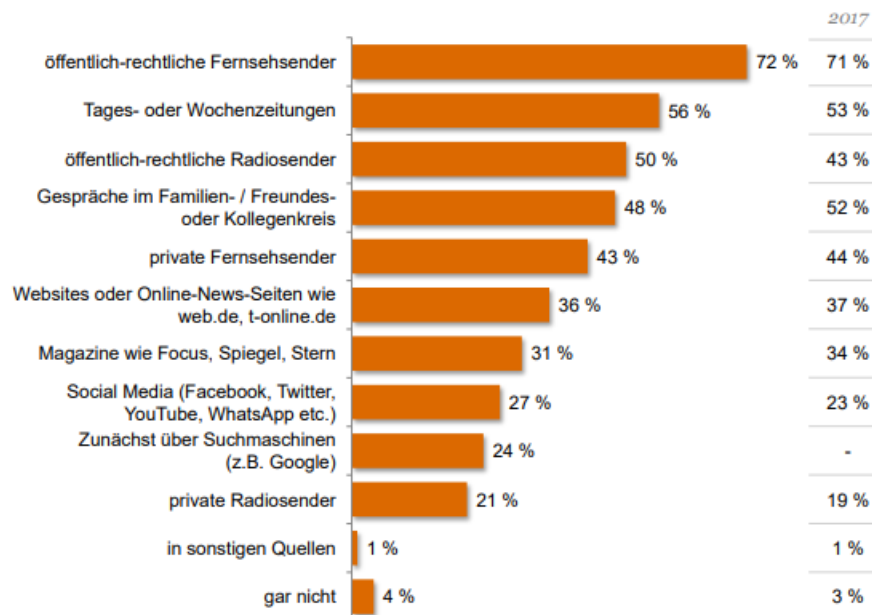


Bild 4: Umfrage 2018 zu „Welche Informationskanäle nutzen Sie?“ [Pric18a]

Besonders die jüngeren Menschen (18-29 Jahre) informieren sich zu 45% in den Sozialen Medien oder durch Gespräche im Familien-/Freundeskreis [Pric18a].

Dadurch wird es für Social Bots einfacher, eine große Zahl an Nutzer zu erreichen, und wahrscheinlicher, diese zu beeinflussen.

Social Bots statten sich mit einem eigenen Profil und Profileigenschaften aus, wie z.B. Alter, Wohnort und Interessen. Ihr Ziel ist es, einen virtuellen, nach außen hin echten, Freundeskreis aufzubauen und sich mit anderen realen Nutzern zu vernetzen. Wenn die Imitation dann auf andere Inhalte mit Kommentaren, Retweets oder Likes antwortet, wird dies den echten Nutzern aufgezeigt. Wenn dieser wiederum ebenfalls auf den Inhalt reagiert, wird auch sein gesamtes Netzwerk davon in Kenntnis gesetzt. Somit bildet sich ein Schneeballprinzip. Die Reichweite dieser Inhalte und Beiträge wird somit immer größer und größer. Beiträge und Tweets mit vielen Kommentaren und Likes können den Anschein geben, dass viele Nutzer diesen Post zustimmen und der gleichen Meinung sind. Wenn aber unter diesen Likes und Kommentaren die Hälfte Social Bots sind, verändern diese das allgemeine Meinungsbild [SaZy18].

2.2.2 Arten

Wie bereits im vorangegangenen Abschnitt beschrieben, gibt es verschiedene Arten von Bots. Generell wird in der Forschung zwischen „Good Bots“ (deutsch: Gute Roboter), die für den Menschen eher eine Hilfe und Unterstützung darbieten, und „Bad Bots“ (deutsch: Schlechte Roboter), welche dem Menschen schaden wollen, unterschieden.

Good Bots

Die *Chatbots* und *digitale Assistenten* dienen dazu mit dem Menschen zu kommunizieren. Sie können Fragen zu bestimmten Themen beantworten oder durch eine Webseite führen. [GrK118] Auch die Sprachassistenten Siri, Alexa oder OK Google gehören zu dieser Technologie. Sie unterstützen den Menschen im Alltag und machen diesen damit leichter [IONO18].

Crawlers agieren eher im Hintergrund und sammeln Daten von Webseiten oder APIs (Application Programming Interface = Anwendungsprogrammierschnittstelle). Sie werden meist von Suchmaschinen wie Google genutzt, um automatisiert und selbstständig das Internet zu durchsuchen und Webseiten zu analysieren [IONO18].

Die *Transactional Bots* arbeiten im Auftrag von Menschen als Agenten. Sie interagieren mit anderen Systemen, führen Transaktionen durch und übertragen Daten von einer auf die andere Plattform [GrK118].

Informational Bots leiten hilfreiche Informationen mittels Push-Benachrichtigungen weiter [GrK118]. Dadurch lässt sich zum Beispiel immer das aktuelle Wetter um 8:00 Uhr morgens auf dem Smartphone anzeigen.

Entertainment Bots sind für die Unterhaltung der Menschen gedacht. Häufig kommt diese Form als virtueller Computerspieler zur Geltung. In vielen Spielen werden virtuelle Akteure benötigt, die mit einem zusammenspielen oder gegen die selbst gespielt werden muss. Diese reagieren variabel auf den jeweiligen Menschen, der dieses Spiel zurzeit spielt. Sie sind auch unter den Begriffen Aim-Bots, KI Spieler oder Poker-Bots bekannt. Mithilfe von Techniken der künstlichen Intelligenz reagieren sie aktiv auf die Handlungen des menschlichen Spielers [IONO18].

Trader Bots durchsuchen Webseiten, wie Amazon oder Ebay, um den niedrigsten Preis für ein Produkt zu finden. Einsatz finden diese meistens bei Online Händlern, um einen Überblick über die Mitbewerber zu erhalten und den besten Preis von allen zu setzen [Koro17].

Bad Bots

Hacker-Bots dienen zur Verbreitung von Malware, zur Täuschung von Einzelpersonen oder zum Angriff auf Webseiten oder ganzen Netzwerken. Sie sind oft verantwortlich für DDoS Angriffe. Dieser Angriff beinhaltet das Verschicken von tausenden Anfragen an ein Zielsystem. Dieses kann dann nicht alle Anfragen schnell genug abarbeiten und schalten sich ab [Koro17].

Scrappers stehlen Inhalte wie Mailadressen, Bilder oder Texte von anderen Webseiten [GrK118]. Der gestohlene Inhalt wird oft ein wenig verändert und dann wieder veröffentlicht.

Spammers verbreiten Spam im gesamten Web, meistens in E-Mails, Blog-Posts oder Kommentaren. Meistens beinhalten diese E-Mails oder Kommentare bestimmte Links, die zu der Webseite des Spammers führen, damit dort u.a. Schadsoftware heruntergeladen und auf dem jeweiligen Zielgerät installiert werden kann [Koro17].

Social Bots lassen sich nicht konkret in eine der beiden Kategorien einordnen. Hierbei handelt es sich um eine Grauzone. Es ist noch nicht genau bewiesen, ob Social Bots nur negative Konsequenzen mit sich ziehen. Denn sie können auch für gute Dinge genutzt werden. Einige von ihnen werden bewusst eingesetzt, um z.B. Mobbing oder Rassismus in Sozialen Medien zu steuern und auf bestimmte Hashtags mit passenden nicht manipulierten Links zu reagieren und spezielle Themen abzuschwächen.

Jedoch werden Social Bots immer mehr für manipulative und schädliche Handlungen eingesetzt. Generell wird innerhalb der Social Bots in Überlaster, Trendsetter und Auto-Trolle differenziert. Diese drei Formen der Social Bots fließen ineinander und können auch in ihren Aufgaben überlappen [Schö17].

Der **Überlaster** sendet gezielt Nachrichten bzw. Kommentare auf bestimmte Posts und Beiträge einer Seite mit einer expliziten Aussage. Er überflutet damit den Feed der jeweiligen Seiten. Äußert sich ein Kanal zu einem bestimmten Thema, meistens sind dies Nachrichtenkanäle, postet der Überlaster immer wieder in Form von Kommentaren eine Gegenaussage. Gerade dann, wenn andere Bots diese Kommentare nochmals kommentieren oder liken, überwiegen sie in ihrer Anzahl gegen die der „echten“ Nutzer. Er überlastet damit einen Post und somit können z.B. Social Media Beauftragte diese Kommentare nicht schnell genug löschen [Schö17].

Der **Trendsetter** bedient sich dem Phänomen der Mehrheitsillusion. Umso mehr Leute sich über ein bestimmtes Thema unterhalten, umso wahrscheinlicher ist es, dass es sich zu einem der Trendthemen auf Twitter entwickelt. Hierfür braucht es lediglich eine große Bot-Armee, die kontinuierlich ein und denselben Hashtag verbreitet. Posten dann tausende von Bots diesen Hashtag und liken Beiträge mit diesen, kann das Thema schnell an Bedeutung gewinnen, obwohl es in Wirklichkeit gar keine Aufmerksamkeit hat. Dieser Fake-Bewegung schließen sich dann echte Nutzer an, weil sie denken, dass die Mehrheit dahintersteht [Schö17].

Auto-Trolle lenken gezielt einzelne Personen ab, damit diese sich mit sinnlosen Diskussionen beschäftigen. Unterhalten sich bspw. zwei Nutzer über ein Thema steigt der Auto-Troll mit ein und versucht mit Beleidigungen oder unpassenden Argumenten die eigentliche Diskussion in den Hintergrund zu rücken. Die meisten Nutzer merken an dieser Stelle nicht, dass sie mit einem Programm schreiben. Somit können sie mehrere Stunden mit diesem Bot beschäftigt sein [Schö17].

2.2.3 Aufbau und Funktionsweise

Ein einfacher und nicht komplexer Social Bot, der auf bestimmte Schlüsselwörter immer mit dem gleichen Text reagiert, ist leicht zu programmieren. Es gibt im Internet viele verschiedene frei verfügbare Frameworks und Baukästen, die bereits vorgefertigte Strukturen und Funktionen beinhalten. Wenn es sich jedoch um einen komplexeren und intelligenteren Social Bot handeln soll, sind meist weiterführende Programmierkenntnisse notwendig, insbesondere im Machine Learning Bereich. Zusätzlich kann auch auf den Aufwand des eigenständigen Programmierens verzichtet

werden und eine Software oder sogar fertige Bots gekauft werden. Beispielsweise können 10.000 Twitter-Accounts, welche automatisch erstellt oder gehackt worden sind, für 100 Dollar online erworben werden. Für weitere 500 Dollar kann auch das Programm, welches die Accounts bzw. Bots steuert, gekauft werden. Mit einer solchen Bot-Armee können in kürzester Zeit die Trends auf Twitter und somit die öffentliche Meinung manipuliert werden [boxc17].

Für die Erstellung eines eigenen Social Bots sind grundsätzlich folgende Schritte zu beachten:

- 1) Konzeptioneller Rahmen
- 2) Aufbau eines Fake Accounts
 - a. Datensammlung (Profilbild, Eigenschaften, aussagekräftige Posts...)
 - b. Genügend Follower und Likes
- 3) Zugriff auf die Programmierschnittstelle (API)
- 4) Bot Programm (Frameworks, Programmiersprachen und Methoden der künstlichen Intelligenz, um regelmäßige Posts zu erstellen)

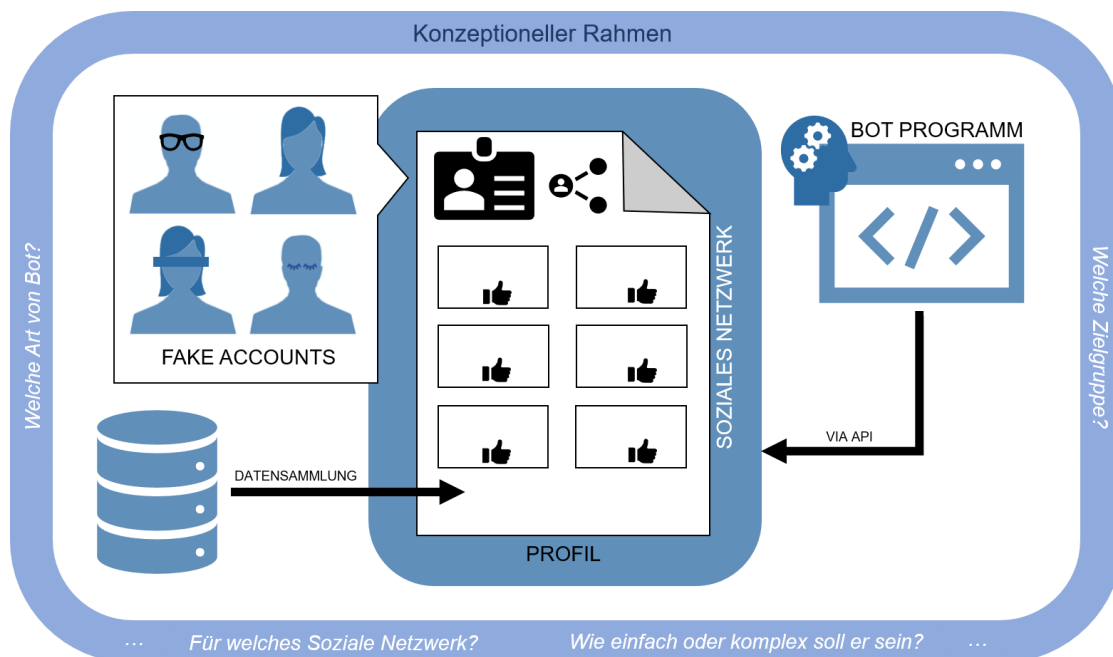


Bild 5: Aufbau eines Social Bots, eigene Darstellung

Bevor sich dafür entschieden wird, einen Social Bot für ein bestimmtes Netzwerk zu entwickeln, wird ein konzeptioneller Rahmen erstellt. Dabei spielen folgende Fragen eine Rolle: Welche Art von Bot wird benötigt? Für wen soll dieser Bot erstellt werden? Für welches Soziale Netzwerk wird er gebaut? Soll ein einfacher oder komplexer Bot erstellt werden?[Sieb19]

Für den reinen Aufbau eines Social Bots müssen registrierte Fake-Accounts in dem jeweiligen Sozialen Netzwerk vorhanden sein [Hege16]. Für die Erstellung der Fake-Accounts sollten vorab für das Profilbild, die Profileigenschaften und zukünftige Posts Daten gesammelt werden, die zusammen ein schlüssiges Bild ergeben.

Es muss auf die Programmierschnittstelle (API) des Sozialen Netzwerkes zugegriffen werden können. Die API stellen die meisten Sozialen Netzwerke kostenlos zur Verfügung, um Entwicklern die Chance zu geben, das Netzwerk mit Funktionalitäten zu erweitern. Je nach Netzwerk kann jedoch der Registrierungsprozess und die Benutzerfreundlichkeit unterschiedlich und komplizierter sein. Beispielsweise gibt es auf Twitter und Instagram mehr Bots als bei Facebook, da deren Zugriff auf die API leichter ist [Hege16].

Weiterhin wird ein Programm benötigt, durch welches die Accounts bzw. Social Bots automatisch gesteuert werden [Hege16]. Für die Entwicklung eines eigenen Programms zur Steuerung von einem Bot kommen Frameworks zum Einsatz, die bereits vorgefertigte Strukturen aufweisen. Frameworks sind sogenannte Programmiergerüste, welche meistens im Zusammenhang mit objektorientierter Entwicklung genutzt werden. „Frameworks bieten, wie es der Name bereits sagt, einen Rahmen, der eine professionelle Quellcodeverwaltung ermöglicht und somit die Plattform für eine Eigenentwicklung liefert. [Sieb19]“ Beispielsweise können auf der Webseite Botwiki.org eine Vielzahl an Frameworks und Tutorials gefunden werden, die einem die Erstellung eines Twitter-Bots vereinfachen [Botw19a]. Zusätzlich sollte sich darüber Gedanken gemacht werden, wo die eigenen Bots gehostet werden sollen. Auch hier gibt es verschiedenste Anbieter, die dies auch kostenlos ermöglichen, wie z.B. Glitch [Botw19b].

Zum Schluss sollte darauf geachtet werden, dass der Social Bot auch genügend Follower, eigene Beiträge und Likes hat, da insb. Accounts mit vielen Abonnenten und Likes für

echt und menschlich wahrgenommen werden. Daneben ist es wichtig zu bedenken, dass Social Bots beliebig skalierbar sind. Nachdem ein Programm für ein Bot entwickelt worden ist, kann damit auch eine große Bot-Armee gesteuert werden.

2.2.4 Erkennung von Social Bots

Die Erkennung von Social Bots ist eine dauerhafte Herausforderung. Umso ausgereifter die Erkennungsmethoden werden, umso schneller und funktioneller entwickeln sich die Bots. Sie werden von Zeit zu Zeit intelligenter und wissen sich immer besser zu tarnen. Durch diesen rasanten Entwicklungsfortschritt müssen sich im gleichen Maße die Erkennungsmethoden weiterentwickeln. Denn eine Erkennungssoftware von vor zwei Jahren wird einen heute entwickelten Social Bot mit großer Wahrscheinlichkeit nicht mehr mit dem gleichen Grad an Genauigkeit identifizieren können. Die Erkennungsmethoden sind dadurch weniger effektiv. Daher ist es wichtig, ständig neue Methoden zur Erkennung zu entwickeln. Die Schwierigkeit hierbei liegt in der Tatsache, dass Social Bots so gestaltet sind, dass sie wie Menschen wirken und sich in die Menge einfügen. Besonders schwierig wird es dann, wenn Methoden der künstlichen Intelligenz verwendet werden, um ein perfektes menschliches Imitat zu erstellen. Daher ist es essenziell, die Merkmale von Social Bots zu erkunden und diese immer wieder zu erweitern bzw. zu optimieren

Die Kommunikation in den sozialen Netzwerken findet meistens sehr unpersönlich und zeitlich versetzt statt. Daher ist es schwierig, für den Nutzer zu erkennen, ob das Gegenüber im Chat oder in den Kommentaren ein realer Mensch ist. Es gibt hierbei noch keinen Algorithmus, der eine 100%ige Zusage machen kann, ob ein Account menschlich oder bot-gesteuert arbeitet. Ein einzelner Faktor reicht nicht aus, um einen Bot zu erkennen. Entscheidend ist das Zusammenspiel mehrerer Faktoren [Nimm17]. Es existiert daher nicht die eine Lösung für die Erkennung von Social Bots. Es existieren aber verschiedene Merkmale und Algorithmen, die das Erkennen leichter machen können.

2017 wurde von Forschern der Indiana University und University of Southern California ein solcher Algorithmus entwickelt, welcher auf Grundlage von sechs Kriterien beruht, Network, User, Friends, Temporal, Content und Sentiment Features [Va17]. Dieses

freizugängliche Bot Detection Framework nennt sich „BotOrNot“. Die sechs Kriterien wurden mit einem Framework zuvor aus einer großen Sammlung an Metadaten von Social Media Nutzern extrahiert. Mit diesen Merkmalen trainierten sie dann Modelle, welche Bots in dem Sozialen Netzwerk Twitter erkennen können. 9 bis 15 % aller Profile auf Twitter sollen der Untersuchung nach als Social Bots identifiziert worden sein.

Im Folgenden werden zu Beginn einige Merkmale dieser Untersuchung, welche auf einen Social Bot hinweisen, mit angeführt. Dies soll aufzeigen, wie schwierig es bis heute noch ist, einen Social Bot gänzlich zu erkennen und einen perfekten Algorithmus zu erstellen. Abschließend werden das Erkennungssystem *BotOrNot* und ein weiteres Framework zur Erkennung von Social Bots, *Botswatch*, kurz vorgestellt.

Merkmal: Aktivität [Nimm17] bzw. Timing [Va17]

Einer der offensichtlichsten Merkmale ist die Aktivität eines Nutzerkontos. Hier wird untersucht, wie oft ein Nutzer Beiträge postet oder Beiträge anderer Nutzer teilt seitdem das Profil eröffnet wurde. Social Bots können wesentlich schneller, innerhalb von Sekunden, auf Informationen oder Themen reagieren [SaZy18]. Der Richtwert, wie viele Veröffentlichungen am Tag darauf hinweisen, ob das Konto menschlich oder bot-gesteuert ist, schwankt immer wieder. Das Oxford Internet Institute oder auch die zuvor genannte Studie gibt hier einen Durchschnittswert von 50 Beiträgen und Likes pro Tag an. Andere wiederum sehen einen Account als verdächtig an, wenn dieser pro Tag 72 Posts veröffentlicht. Dennoch müssen diese Kennzahlen kritisch betrachtet werden. Es kann sich bspw. auch bei 50 abgesetzten Tweets an einem Tag, um einen sehr aktiven Nutzer handeln [Nimm17].

Auch die These, dass ein Bot immer zur gleichen Uhrzeit seine Beiträge postet, kann mittlerweile entkräftet werden. Mithilfe künstlicher Intelligenz täuschen sie nun auch Schlaf vor und haben einen bestimmten Rhythmus in ihrem Verhalten. Sie antworten bspw. teilweise wie ein menschlicher Nutzer verzögert, damit sie den Anschein erwecken, dass sie tippen oder gerade etwas anderes zu tun haben [Schö17].

Dennoch kann festgehalten werden, dass, wenn ein Nutzer über 50 Beiträge gleichzeitig teilt und postet, in Sekundenschnelle auf Themen reagiert, sich nur einem einzigen

Thema widmet und immer wieder im Abstand von wenigen Sekunden antwortet, sich hinter diesem Account mit einer sehr hohen Wahrscheinlichkeit kein Mensch verbirgt.

Merkmal: Profileigenschaften

Profilinformationen geben Aufschluss darüber, ob der Nutzer echt ist. Ist das Profil weitestgehend anonym gehalten und gibt es weder Profilfoto noch Profileigenschaften, sollte davon ausgegangen werden, dass hinter diesem Account ein Bot steckt [Nimm17].

Passen das Profilbild und die Profileigenschaften nicht zusammen, kann auch hier ein Nutzer skeptisch werden. Meist sind bei Bots die Profilinformationen generisch erstellt und weisen keinerlei Zusammenhang auf. Bspw. wird ein Foto eines Mannes verwendet und der Nutzernamen ist der einer Frau. Es werden zufällige Fotos aus dem Internet genommen und mit automatisch generiertem Alter und Wohnort als Profil verwendet. Dennoch kann schnell mit der Reverse Image Search (rückwärts gerichtete Bildersuche) ein Profilbild auf seine Echtheit untersucht werden. Der Nutzer kann mithilfe eines Rechtsklicks auf das Profilbild „Search Google for image“ mit der Suchmaschine Google nach passenden Bildern suchen. Dabei kann oft festgestellt werden, dass das jeweilige Bild für mehrere andere Profile genutzt wird [Nimm17].

Ebenso sollte davon ausgegangen werden, dass ein Profil nicht echt ist, wenn im Namen alphanumerische Chiffren verwendet werden. Diese werden oft von Algorithmen automatisch erstellt. Aber auch wenn der Nutzernamen anders betitelt wird als der Name im Profil, sollte dieses Profil näher betrachtet werden [Nimm17].

Merkmal: Zwischenmenschliche Kommunikation

Einfache Bots können auf komplexere Fragestellungen nicht reagieren. Auch Kontextwissen können die Bots nicht verarbeiten. Mit Fragestellungen wie „Was ist über dir oder unter dir?“ wird ein Bot überfordert sein und merkwürdige Antworten geben [Sick17]. Aber auch aus dem Sprachstil lassen sich einige Schlüsse ziehen. In der Oxford Studie wird dies als „Entropie der Sprache“ bezeichnet. Die Länge eines Postings oder auch der Satzbau kann Aufschluss darüber geben, ob dies ein Mensch verfasst hat. Verwendet der Bot zum Beispiel immer wieder dieselben Wörter oder die gleiche Grammatik, liegt es nahe, dass es sich um einen Social Bot handelt [Va17]. Ebenso ist

das Verhältnis aus Abonnenten und Abonnierten ein wichtiger Anhaltspunkt. Oft folgen Bots vielen anderen Accounts, jedoch haben sie selbst meist nicht sehr viele Follower [IONO18].

BotOrNot (Botometer)

Die kostenlose und öffentlich zugängliche Web-API für Twitter *BotOrNot*, inzwischen geändert in *Botometer*, vergleicht einen eingegebenen Twitter-Account mit bekannten Eigenschaften von denen eines Social Bots. Hierbei werden verschiedene Kennzahlen bzw. Features extrahiert (s.o.), welche Profilinformationen umfassen. Diese werden als Grundlage für ein Lernverfahren genutzt, welches für jeden Account eine Bewertung erstellt. Mithilfe dieser Bewertung kann abgeleitet werden, inwiefern sich hinter diesem Nutzer ein Mensch oder ein Roboter verbirgt. Jedoch wurden nur die Bots, welche auf Honeypots reagiert haben, evaluiert. Honeypots sind getarnte Accounts, die als Köder genutzt werden, um Social Bots anzulocken, damit sie sich mit ihnen vernetzen. Bei diesem Modell werden jedoch nur Bots erkannt, die kongruierende Eigenschaften wie die Social Bots haben, welche auf den Honeypot reingefallen sind. Demnach wird klar, dass dieses Modell nur eine Ergänzung zur eigenen Einschätzung ist und nicht als 100%ige Erkennung von Social Bots genutzt werden kann [Pric17a].

Botswatch

Die Firma Botswatch entwickelte ein Tool, welches sich auf das Scannen von gesamten Inhalten sozialer Netzwerke - im besonderen Maße Twitter – spezialisiert hat. Das Framework arbeitet mit einer Vielzahl von Kriterien, um Social Bots frühzeitig zu erkennen [bots19]. So können Nachrichtenagenturen und auch Firmen von dieser frühzeitigen Erkennung profitieren, Reputationsschäden vermeiden und potenzielle Kommunikationsstrategien erstellen. Die Funktionsweise des Tools wird von den Entwicklern geheim gehalten, daher kann keine Aussage über die Genauigkeit des Tools gemacht werden [Plic18b].

Die Herausforderung der nächsten Jahre ist es, Entwicklern von böartigen Social Bots immer einen Schritt voraus zu sein. Social Bots werden wie alle anderen Technologien immer weiterentwickelt und mit immer neuartigen Methoden optimiert.

Gleichermaßen müssen hier die Entwickler von Erkennungsprogrammen mitziehen und ihre Algorithmen dem Technologiewachstum anpassen.

2.2.5 Einsatzbereiche, Chancen und Gefahren

Social Bots können auf verschiedene Arten in den sozialen Medien eingesetzt werden. Je nach Einsatz stellen sie neue Gefahren dar oder unterstützen Firmen im Marketing und uns Menschen im Alltag. Der aktuelle primäre Wirkungsraum von Social Bots, welcher sich aus Forschungsarbeiten und Studien ableitet, ist der Kurznachrichtendienst Twitter. Dies kommt vermutlich dadurch zustande, dass Twitter sehr geringe technische Hürden aufweist und die Daten leicht zugänglich und extrahierbar sind. Jedoch können auf allen sozialen Netzwerken Bot-Aktivitäten stattfinden, solange diese eine frei zugängliche API besitzen [Ki17].

Politik

Social Bots können als digitale Wahlhelfer eingesetzt werden. In den US-Präsidentschaftswahlen 2016 wurde erstmals öffentlich über das Thema der Social Bots diskutiert. Social Bots wurden hierbei eingesetzt und dienten als Unterstützung für den jeweiligen Wahlkampf. Laut der Oxford University Studie ist jeder dritte Follower kein Mensch sondern ein Bot gewesen [Hein17]. In politischen Wahlkämpfen ist es das Ziel, gegnerische Aussagen in den sozialen Medien mithilfe von Hashtags zu unterdrücken. Social Bots können so bestimmte Wahlkampfaussagen automatisiert innerhalb von wenigen Sekunden in die Trends befördern. Auch während der Ukrainekrise 2014 konnten Bot-Aktivitäten verzeichnet werden. Diese förderten das Gerücht, dass die USA ein Flugzeug mit hunderten von Passagieren abgeschossen hätte. 45 Prozent der 1,3 Millionen Twitter-Profilen in Russland waren laut einer Studie Bots. Ebenso waren in der Brexit Debatte zwei der aktivsten Twitter-Nutzer Social Bots [Inst18]. All diese Social Bot Einsätze wurden auf dem Kurznachrichtendienst Twitter nachgewiesen.

Demnach können Social Bots dazu genutzt werden, mithilfe von unzähligen Hashtags und polarisierenden oder auch einfachen Inhalten, gegnerische Meinungen zu überfluten, Propaganda zu verbreiten oder Followerzahlen künstlich nach oben zu treiben, um die eigene Position zu verdeutlichen. Dies geschieht auch bei einigen deutschen Politikern. Beispielsweise beim Twitteraccount von der Partei Die Grünen. 2015 wurde eine

schlagartige Veränderung der Followerzahlen auf dem Twitter-Account vermerkt. Die Vermutung lag nah, dass es sich hierbei, um Fake-Follower handelte [Ki17].

Aber auch die gezielte Meinungsmache in besonderen politischen Bewegungen kann Gefahren mit sich bringen. Es können bestimmte Inhalte, wie Bilder, Videos aber auch Nachrichten, veröffentlicht werden, welche z.B. die Bildung von extremen Gruppen fördert und Propaganda, beispielsweise im rechtsextremen Raum, zulässt. Studien widerlegen aber, dass eine Nachricht in den sozialen Medien die gesamte politische Meinung eines Nutzers ändert. Es ist aber umso wahrscheinlicher, dass massenhaft veröffentlichte und extreme Inhalte in einem bestimmten Kontext (Gruppe oder Hashtag) an mehr Bedeutung und Zurufe gewinnen und somit Diskussionen aufgeheizt werden [Hege16].

Wirtschaft

Das Influencer-Marketing gehört ebenso zu einem Einsatzbereich der Social Bots. Hierbei werden Nutzer mit sehr hohen Followerzahlen und somit großen Einfluss von Firmen genutzt, um deren Produkte oder Marke möglichst vielen Personen näherzubringen und diese zu einer positiven Kaufentscheidung zu beeinflussen. Momentan ist der Einsatz von Bots in diesem Bereich noch unbekannt, jedoch liegt aufgrund der Stärke des Umsatzes ein großes Potenzial vor. Social Bots können zu Werbezwecken verwendet werden, um neue Produkte verdeckt auf verschiedenen Kanälen zu bewerben. Zusätzlich können börsennotierte Finanzprodukte manipuliert werden. Die Finanzmärkte werden gestört und somit auch das Vertrauen in deren Zuverlässigkeit. Das Bundeskriminalamt (BKA) identifizierte drei potenzielle Angriffspunkte [Ki17].

1. Der Aktienwert eines Unternehmens kann durch präzise Falschmeldungen von Social Bots nach oben aber auch nach unten getrieben werden. Als Beispiel wäre hier das amerikanische Technologieunternehmen Cynk zu nennen. Dieses hat lediglich einen Mitarbeiter und erwirtschaftete keinen Umsatz. 2014 stieg jedoch die Aktie innerhalb weniger Wochen und verzeichnete kurzzeitig einen Börsenwert von ca. sechs Milliarden Dollar. Damit einhergehend konnten zahlreiche positive Twitter-Kommentare registriert werden. Aufgrund dessen

könnten Handelsroboter die Aktie weiterhin gekauft haben, auch wenn menschliche Marktteilnehmer diese Manipulation erkannten [Pric17b].

2. Social Bots können dazu genutzt werden, künstliche, nichtreale Märkte zu erschaffen. Mittels falscher Meldungen über mögliche Geschäftsoptionen und lukrative Angebote kann ein Anleger dazu verleitet werden, in ein nicht existierendes Finanzprodukt zu investieren [Ki17].
3. Traditionelle Beratungs- und Vertriebsmodelle werden von Social Bots angegriffen, um mithilfe von verbreiteten Informationen den Marktpreis eines Finanzinstrumentes zu manipulieren [Ki17].

IT-Sicherheit

Laut dem Stand der aktuellen Forschung greifen Social Bots weder Soft- noch Hardware von IT-Systemen direkt an. Dadurch ist eine DDoS-Attacke sehr unwahrscheinlich. Diese werden häufig jedoch von einem Bot-Netz durchgeführt, welches eher an die Hard- und Software adressiert ist und nicht wie die Social Bots den Menschen als Schwachstelle sehen. Durch die schnelle Entwicklung intelligenter Technik und den wachsenden Fähigkeiten von Bots könnten in Zukunft die verschiedensten Gefahren, wie das Eindringen in Geräte, mit kriminellern Hintergrund entstehen.

Doch bereits jetzt können Social Bots so eingesetzt werden, dass sie großen Schaden anrichten. Mithilfe von Automatic Spear Phishing besteht die Möglichkeit, IT-Infrastrukturen und -Systeme anzugreifen. Phishing bedeutet, dass vorgetäuschte Nachrichten an potenzielle Opfer versendet werden, die Vertrauen erwecken und die Interessen desjenigen ansprechen sollen. Diese dienen dafür, dass die Opfer dazu bewegt werden, auf einen Link in der Nachricht z.B. auf eine vermeintliche Webseite, auf welche sie ihre Passwörter eingeben sollen, zu klicken. Hierbei könnte sich hinter den Links Schadsoftware verstecken, welche dann auf dem Gerät des Opfers installiert wird. Die Viren werden verteilt und die installierte Software kann unbemerkt im Hintergrund Schaden anrichten [Ki17]. Im März 2017 gab es im großen Stil solch einen Angriff auf das US-Verteidigungsministerium. Über Twitter erhielten über 10.000 Ministeriumsmitarbeiter täuschend echt wirkende Mitteilungen, welche die Interessen des Empfängers ansprachen. Dank der Social Media Accounts können Angreifer auf die

Vorlieben und Interessen des potenziellen Opfers zugreifen und somit zugeschnittene Nachrichten verteilen. Ein Tier-Fan erhält beispielsweise einen Link zu einer Tierschutzorganisation. Jedoch verbergen sich hinter diesen Links Schadprogramme [Pric17b]. Diese Angriffe sind momentan mit Social Bots gerade wegen ihrer Skalierbarkeit möglich. Gegenüber 2% bei normalen Phishing-Nachrichten werden beim Spear Phishing ungefähr 50% Klickraten von den Opfern erzielt und somit ist die Art des Angriffes sehr effizient.

Auch das massive Täuschen und Manipulieren von Nutzern, um Geld zu erhalten, ist mittlerweile eine Angriffsmöglichkeit geworden. So verbargen sich auf der US-Datingplattform Ashley Madison hinter 70.000 Frauen-Profilen Bots, welche Männer in kostenpflichtige Chats verführten [Pric17b].

Positive Einsatzmöglichkeiten

Social Bots manipulieren die Trends in den sozialen Medien und beeinflussen somit die politischen und wirtschaftlichen Entscheidungsprozesse. Durch Social Bots können nicht nur Hass und Manipulation verbreitet werden, sondern auch positive Konsequenzen erfolgen wie Toleranz.

Es können Social Bots im Zusammenhang mit Falschinformationen eingesetzt werden, um diese aufzuspüren und ihnen entgegenzuwirken oder sie werden als Köder verwendet, um böartige Bots zu identifizieren. Zudem können auch verletzend und beleidigende Tweets mithilfe von gutartigen Social Bots eingedämmt werden [Ki17].

Ein Beispiel für einen harmlosen Bot ist der Twitteraccount @PfannKPolizei, Pfannkuchenpolizei, welcher sich darauf spezialisiert hat, Tweets mit dem Wort „Berliner“ mit dem Satz „dit Pfannkuchen heißt!“ zu kommentieren [Ki17].

2.3 Künstliche Intelligenz

Mithilfe von sogenannter künstlicher Intelligenz können Social Bots wie Menschen wirken. Automatisierte intelligente Bots haben die Möglichkeit skaliert auf tausende Konten Manipulation und Täuschung in den sozialen Netzwerken auszuführen ohne jemals als Roboter erkannt zu werden.

Es gibt viele unterschiedliche Definitionen für künstliche Intelligenz. In der Forschung konnte sich bislang auf keine einheitliche Begriffserklärung geeinigt werden, da es schwierig ist diesen Begriff präzise und einfach zu definieren. Einerseits aufgrund der Breite des Gebietes und andererseits, da selbst für den Begriff Intelligenz keine eindeutige Definition herrscht. Gemeinsam haben alle, dass es sich hierbei um ein Teilgebiet der Informatik handelt, da sich die KI mit der Entwicklung von „intelligenten Agenten“ beschäftigt [BuSc19].

Nach Carsten Felden soll die künstliche Intelligenz „... Maschinen in die Lage versetzen, menschliche Tätigkeiten zu übernehmen. [...] Die KI versucht, das menschliche Gedächtnis, sein Lernverhalten und seine Entwicklung nachzubilden.“ [Feld16]

Die Unterscheidung zwischen starker und schwacher künstlicher Intelligenz wird in vielen Literaturen bezeichnet. Die starke künstliche Intelligenz, auch Strong AI genannt, beschäftigt sich mit der Abbildung von menschlichen Emotionen, Empathie und Vorgängen im Gehirn. Jedoch ist diese Art der künstlichen Intelligenz bislang nicht machbar und noch nicht weit genug erforscht. Die schwache künstliche Intelligenz, auch Weak AI genannt, ist bereits machbar und wird in vielen Gebieten eingesetzt. Hierbei wird nicht mehr auf die Imitation von menschlichen Denkprozessen und Kreativität abgezielt, sondern der Fokus liegt auf der Entwicklung von Algorithmen zur Bewältigung konkreter Problemstellungen durch Techniken der Mathematik und Informatik [BuSc19].

Die künstliche Intelligenz umfasst im Allgemeinen das Nachbilden von Fähigkeiten des Menschen wie dem Lernen aus Beispielen oder dem Problemlösen mithilfe von mathematischen und informationstechnischen Methoden. Damit können Computer ihre

eigene Leistung verbessern und den Menschen beim Bewältigen von täglichen Aufgaben unterstützen.

Neben dem Teilgebiet des maschinellen Lernens wurde in den letzten Jahren das tiefe Lernen, Deep Learning, stark erforscht. Das Deep Learning basiert auf künstliche neuronale Netze (KNN), welche die Strukturen des menschlichen Gehirns mit den Nervenzellen nachbilden [Witt19]. Das Ziel ist es, das menschliche Gehirn zu imitieren. Abgegrenzt vom maschinellen Lernen können neuronale Netze durch ihre mehrschichtigen Netzwerke Zusammenhänge erlernen und das mithilfe einer viel größeren Anzahl an Trainingsdaten. Ein KNN Modell besteht aus Kanten, den Synapsen und Knoten, den Neuronen [BuSc19].

In den nächsten Kapiteln wird der Zusammenhang zwischen künstlicher Intelligenz und Social Bots näher erläutert. Es sollen kurz die Gebiete des maschinellen Lernens und die Verarbeitung der natürlichen Sprache beschrieben werden. Diese stellen Teilgebiete der künstlichen Intelligenz dar.

2.3.1 Social Bots und künstliche Intelligenz

Als ersten Chatbot wurde ELIZA im Jahr 1966 vom deutsch-amerikanischen Informatiker Joseph Weizenbaum entwickelt. ELIZA konnte Gespräche in natürlicher Sprache nachahmen und über Skripte unterschiedliche Gesprächspartner simulieren. Sie sprach mit den Menschen wie Psychotherapeuten und diese gaben ihre intimsten Geheimnisse preis. Dies geschah, obwohl bekannt war, dass ELIZA kein Mensch sondern ein Bot war.

Die Menschen bemerken irgendwann nicht mehr, wer oder was ihnen Antworten bei Suchanfragen oder Nachrichten in sozialen Medien liefert. Grund dafür ist die Möglichkeit, eine Vielzahl an Daten sammeln und diese nach Muster automatisch auswerten zu können (Big Data). Diese Auswertungen können dann genutzt werden, um Nutzer und Menschen in explizite Richtungen zu lenken [Rese17].

Dagegen arbeiten Social Bots anders. Sie verbreiten fremde Inhalte und Informationen nach bestimmten Regeln und Mustern. Sie wiederholen immer gleiche Aufgaben, wie das Beantworten von leichten und nicht zu komplexen Fragen. Meist reagieren sie auf

bestimmte Stichworte oder Hashtags und retweeten diese. Weniger vertreten sind diejenigen Social Bots, die eine Analysemöglichkeit bieten. Diese können Inhalte in die dafür vorgesehenen Zielgruppen verteilen [Hert17].

Ein natürlich wirkendes Gespräch von Social Bots wurde bisher noch nicht im vollen Maße belegt. Der intelligente weibliche Twitter-Bot „Tay“ von Microsoft lernte durch die Interaktion mit anderen Nutzern. Je intensiver der Bot mit anderen Nutzern chattete und interagierte, desto schlauer wurde er. Tays Zielgruppe waren die 18-24-jährigen Amerikaner. Jedoch wurde der eigentlich freundlich entwickelte Social Bot innerhalb weniger Stunden zu einem Hass-Bot, welcher rassistische und beleidigende Tweets absetzte. Tay nahm alle Informationen und Inhalte ungefiltert auf und lernte nicht nur von freundlichen, sondern auch von hasserfüllten Nutzern. Nach diesem Ereignis nahm Microsoft den Bot sofort von Twitter und löschte alle Hass-Tweets. Daraufhin meldeten sich einige Twitter-User zu Wort und sind der Meinung, dass dies falsch wäre, da damit Microsoft vertuschen würde, welche Gefahren von KI ausgehen würde [Sick16].

Dieses Beispiel zeigt, dass die momentane Situation noch keine allzu großen Gefahren bergen. Jedoch zeigt es auch, wie schnell die Entwicklung der Technologien voranschreitet und welche Möglichkeiten in Zukunft bestehen. Die Mehrheit der Social Bots kann aktuell lediglich auf Inhalte mit bestimmten Stichworten reagieren. Die Antworten werden meist von den Entwicklern vorgegeben und beziehen sich häufig auf ein kleines Themengebiet. Dennoch befinden sich im Internet zahlreiche Informationen und Frameworks zum Thema Social Bots und künstliche Intelligenz. Für eine genauere Betrachtung und um festzustellen, was momentan möglich ist und welche intelligenten Funktionen von Social Bots verwendet werden können, wird im zweiten Teil der Thesis ein eigener Twitter-Bot mit einigen intelligenten Funktionen erstellt.

2.3.2 Machine Learning

In den 1980er Jahren wurde der Grundstein für das maschinelle Lernen gelegt. Maschinelles Lernen soll einen Computer, der eine bestimmte Aufgabe hat, dazu bringen, aus Erfahrungen zu lernen, um die Aufgabe damit zukünftig besser erledigen zu können [Witt19]. Als Beispiel sollen Katzen- und Hundefotos dienen. Damit das System in den Bildern zwischen einem Hund und einer Katze unterscheiden kann, wird nicht

mehr explizit formuliert, dass ein Hund zwei Augen, vier Beine und Fell hat. Das System wird vielmehr mit vielen unterschiedlichen Fotos trainiert. Jedes Foto wird dann einer Kategorie bspw. „Hundefoto“ oder „Katzenfoto“ zugeordnet. Damit kann der Algorithmus selbstständig lernen, welche Merkmale jedes Tier auf den Fotos hat und wie sie sich von anderen Tierfotos unterscheiden [BuSc19].

Grundsätzlich existieren drei verschiedene Lernverfahren, das überwachte Lernen, das unüberwachte Lernen und das verstärkte Lernen.

Das überwachte Lernverfahren, auch Supervised Machine Learning genannt, gibt einem Computerprogramm ihm bekannte Eingabedaten. Dieser wird dann mit den Daten auf eine vorliegende Interpretation und anschließende Ausgabe trainiert. Aus den verbundenen Eingaben mit deren Ausgaben, welche bereits vordefiniert sind, werden generelle Regeln definiert, um neue Eingabedaten richtig zuordnen zu können und neue Ausgaben zu interpretieren. Es lassen sich also mithilfe der zuvor gelernten Prognosen zukünftige unbekannte Eingabe- und Ausgabedaten aufstellen. Dieses Verfahren lernt somit von Beispielen. Ein Verfahren von diesem überwachten Lernen ist das Modell der Regression. Zum Beispiel gibt es einen linearen Zusammenhang zwischen der Körper- und Schuhgröße. Je größer der Körper, umso größer auch der Schuh. Mit der Eingangsvariable Körpergröße und dem abhängigen Ausgangswert Schuhgröße lässt sich eine lineare Funktion zeichnen. Mit diesem Verfahren der Regression und den ermittelten Parametern wird ein mathematisches Modell erstellt, welches für zukünftige Körpergrößen die passende Schuhgröße ermitteln kann. Ein zweites wichtiges Modell des überwachten Lernens wäre die Klassifikation. Das Beispiel mit den Tierfotos, welches bereits im oberen Abschnitt vorgestellt wurde, zählt zum Verfahren der Klassifikation. Innerhalb des Lernens werden mehrere Werte in Klassen unterteilt. Bei späteren Vorhersagen wird dann der Wert einer konkreten Klasse zugeordnet [Witt19].

Das unüberwachte Lernen, auch Unsupervised Machine Learning genannt, lernt nicht mit vordefinierten Eingabedaten und Zuordnungen. Hier wird versucht zwischen den Eingabedaten Ähnlichkeiten zu erkennen. Wenn Ähnlichkeiten gefunden worden sind, werden diese Daten zu einer Kategorie zusammengefasst. Das beschreibt das Verfahren des Clusterings, welches zum unüberwachten Lernen zählt [Witt19].

Das verstärkte Lernen, auch Reinforcement Machine Learning genannt, funktioniert mit Erfahrungen. Das Computerprogramm interagiert mit seiner Umgebung und erhält für sein richtiges Verhalten Belohnungen. Hierbei soll das Programm merken, wenn es eine falsche Antwort gibt oder Handlung ausführt, dass dies Konsequenzen als Folge haben wird. Es versucht daher immer, die richtigen Ergebnisse zu erzielen, damit es die maximale Belohnung erhält. Die Belohnung ist dabei die Regelgröße, welche immer wieder angepasst wird. Ein bekanntes Beispiel wurde von der Firma Fanuc entwickelt. Roboterarme erlernten mittels des verstärkten Lernens innerhalb weniger Stunden unbekannte Objekte zu greifen und zu bewegen [Witt19].

2.3.3 Natural Language Processing

Natural Language Processing (NLP) verarbeitet natürliche Sprache maschinell. NLP beschreibt die Verfahren der Sprachverarbeitung und Spracherkennung. Ziel ist es hierbei eine natürliche Kommunikation zwischen Maschine und Mensch zu ermöglichen. Die Herausforderung dabei besteht darin, dass die natürliche Sprache unstrukturiert und unbestimmt ist. NLP verwendet daher verschiedene Techniken aus der Informatik und Linguistik. Mit der Etablierung des maschinellen Lernens in den 1980er Jahren wurde auch das NLP erforscht. NLP ist somit eine Unterkategorie des Machine Learnings [Vogt17].

Die Übersetzung in eine andere Sprache, die Erkennung gesprochener Sprache oder das automatische Beantworten von Fragen stellt immer wieder eine Herausforderung für Computer da, besonders wenn es um doppeldeutige Aussagen, Ironie oder komplexe Sätze geht. Maschinen müssen lernen, nicht nur jedes Wort einzeln zu betrachten, sondern das Wort im Zusammenhang mit dem Satz und den anderen Wörtern zu bringen. Verschiedene Kenntnisse aus den Sprachwissenschaften helfen bei dem Erlernen und Verstehen der natürlichen Sprache [IONO19].

Die Zusammensetzung von Wörtern und ihre Beziehungen untereinander stellt einen großen Bereich der **Morphologie** dar. Sie analysiert den Satzbau und kann somit Aufschluss über die Stellung im Satz, wie Subjekt oder Prädikat, und die Beziehung zu allen Wörtern geben. Die Morphologie bietet verschiedene Tagging Methoden. Die bekannteste Methode ist der Brill-Tagger. Dieser Algorithmus erfasst zunächst die am

häufigsten verwendeten Wortfunktionen im Satz und leitet dann mithilfe von Regeln die anderen Wortfunktionen ab. Eine Regel wäre beispielsweise, dass das erste Wort eines Satzes immer ein Eigenname ist und das zweite dann mit hoher Wahrscheinlichkeit ein Verb. In dem Satz „Heike hat die fehlenden Teile gefunden“ kann somit das Wort „hat“ als Verb klassifiziert werden [IONO19].

Wie Wörter zu Sätzen zusammengestellt werden, beschreibt die **Syntax**. Sie analysiert die Struktur von Sätzen. Mithilfe von Baumdiagrammen wird der Satz in Phrasen aufgeteilt. Jede Sprache hat ihre eigene Grammatik und somit ihre eigene Zusammenstellung und Hierarchie der Phrasen. Diese Grammatik kann dann von einem Computer erlernt werden [IONO19].

Die **Semantik** beschäftigt sich mit der Bedeutung von Wörtern und Wortgruppen. Denn ein Wort kann für uns Menschen verschiedene Bedeutungen haben. Beispielsweise folgende Sätze:

„Im Topf wachsen schöne Blumen“ und „Der Topf ist voller Kartoffeln“ [IONO19]

Ein Mensch, der die deutsche Sprache beherrscht, weiß sofort, dass es sich im ersten Satz um einen Blumentopf und in dem nächsten Satz um einen Kochtopf handelt. Für einen Computer jedoch ist dies einer der größten Herausforderungen. Sie versuchen daher oft, die Bedeutung des Wortes aus den Wörtern, die vor oder hinter dem Satz stehen, zu bestimmen. Hinter dem Topf im zweiten Satz steht Kartoffeln. Also kann der Computer hier damit rechnen, dass es sich um einen Kochtopf handelt. Er lernt von Textkorpora oder Wörterbüchern, die für jedes Wort die korrekte Bedeutung wiedergeben können und gleicht diese mit den Wörtern im Text ab.

Der Kontext von sprachlichen Änderungen wird in der **Pragmatik** näher betrachtet. Schließlich wird dann in der **Phonologie** der akustische Aufbau von gesprochener Sprache analysiert [IONO19].

3 Entwicklung eines eigenen Twitter-Bots

Für die Entwicklung eines Twitter-Bots sind drei wichtige Schritte notwendig [Ramí18].

1.) Erstellung einer Twitter App

Zu Beginn der Entwicklung muss ein Twitter-Account eingerichtet werden. Hierbei wird zur Verifikation eine Telefonnummer genutzt. Bei diesem Schritt sollte gerade für Betrüger das erste Hindernis liegen, da durch eine Telefonnummer auch der Besitzer dieser herausgefunden werden kann. Jedoch kann dies mithilfe von bestimmten Webseiten umgangen werden. Webseiten, wie receive-smss.com, bieten Telefonnummern an, bei denen direkt auf der Webseite die letzten 50 SMS angezeigt werden. Somit kann eine dieser Nummern verwendet werden, um den Twitter-Code für das Verifizieren zu erhalten. Nach der Erstellung des eigenen Accounts wird eine eigene App im Developer Zentrum von Twitter erstellt. Dadurch bekommt der Entwickler Zugang zu den Access Tokens und Keys, die später für die Nutzung der App benötigt werden.

2.) Extrahierung von Tweets

Für jegliche Entwicklungsplattformen, ob Python oder Node.js, werden die Tokens und Keys benötigt, um über die Twitter API die Verbindung zur App aufzubauen. Nachdem der Zugang erstellt worden ist, können nun Tweets, welche ein Nutzer erstellt hat oder die einen bestimmten Hashtag vorweisen, extrahiert werden.

3.) Veröffentlichung neuer Tweets

Die Tokens und Keys werden benötigt, um einen bestimmten String in einem neuen Tweet auf dem eigenen Twitter-Account zu veröffentlichen.

3.1 Frameworks

Die Erstellung eines Twitter-Bots wird durch eine Vielzahl von offenen und frei verfügbaren Frameworks unterstützt. Es gibt einige Fertigbausteine, bei denen ein Entwickler weder viel Zeit investieren noch Programmierkenntnisse haben muss. Die zwei weit verbreitetsten Tools sollen in diesem Abschnitt kurz vorgestellt werden.

Das bekannteste Tool, um einen Twitter-Bot innerhalb weniger Minuten und ohne Programmierkenntnisse zu entwickeln, nennt sich **Twitter Bots V2**. Hierbei müssen nur die API Tokens des eigenen Twitter-Accounts übergeben werden. Anschließend kann der Nutzer aus verschiedenen Optionen wählen. Beispielweise generiert die Aktion „DM Followers“ für neue Abonnenten eine vorher definierte Direct Message auf Grundlage einer Twitter Search, welche aus Hashtags, Nutzernamen und einer definierten Zeitspanne bestehen kann. Ein weiteres Beispiel wäre die Aktion „Quote Tweet“, welche Tweets mit einem vorgegebenen Text kommentiert [Digi19].

Bild 6: Twitter Bots V2 - Example [Digi19]

Bei diesem Framework sind keinerlei Programmierkenntnisse notwendig und ein Bot ist, wenn bereits vorher auf der Plattform Twitter eine App konfiguriert wurde, innerhalb weniger Minuten erstellt. Das Tool achtet darauf, dass die erstellten Bots nicht auf alle Tweets reagieren. Die Twitter API markiert Tweets mit einem booleschen Flag „possibly_sensitive“, um sensible Inhalte wie Gewalt oder Pornografie zu kennzeichnen. Diese Tweets werden von dem Framework ignoriert. Zusätzlich wird nicht jeder einzelne Tweet vom Bot verarbeitet. Hier wird versucht alle 15 Minuten auf 5-10 Tweets zu reagieren. Dadurch werden die Automatisierungsregeln von Twitter umgangen [Digi17].

Ein weiteres Framework, welches sich auf die schnelle und einfache Entwicklung von Twitter-Bots spezialisiert hat, ist „**Cheap Bots, Done Quick!**“. Diese JavaScript-

Bibliothek ermöglicht dem Ersteller solcher Twitter-Bots mithilfe von Tracery (einem Tool zur Generierung von Texten), einen eigenen Text automatisch zu erweitern und daraus einen Tweet zu verfassen. Der Text beginnt mit einem „Ursprungs“-Knoten (origin), das Tool erweitert diesen dann. Dieser Tweet wird dann innerhalb einer festgelegten Zeit getwittert.



Bild 7: Tracery Grammar [Comp19]

Neben den Frameworks gibt es zusätzlich viele fertige Codes auf GitHub für die unterschiedlichsten Anwendungen. Hierbei müssen nur die Twitter API Tokens der eigenen App bekannt sein.

3.1.1 Twitter API

Für den entwickelten Bot wurde das soziale Netzwerk Twitter ausgewählt. Die Twitter API ist derzeit frei zugänglich und ohne große Einschränkungen verfügbar. Es wurde sich gegen Facebook und Instagram entschieden, da auf Twitter die meisten Social Bots anzutreffen sind und die Erkennungsprogramme als Ziel Twitter verwenden.

Mithilfe der Twitter API kann der Entwickler auf die wichtigsten Funktionen von Twitter zugreifen und gibt diesem somit das Recht, Informationen zu lesen und zu schreiben. Viele HTTP-Endpunkte werden von der API verfügbar gemacht, mit der auf folgende Punkte zugegriffen werden kann: Tweets, Retweets, Likes, Direct Messages, Favoriten und Trends [Garc19].

Der erste Schritt, um einen Twitter-Bot zu erstellen und die dafür notwendigen API Schlüssel zu erhalten, besteht darin, sich einen Twitter-Account anzulegen. Twitter hat

hier bereits einige Sicherheitsvorkehrungen getroffen, um Nutzer mit krimineller Absicht auszuschließen. Das erste Hindernis stellt die E-Mail-Adresse dar, die zusammen mit einem Nutzernamen und Passwort angegeben werden muss. Nutzer, welche Social Bots für Betrugszwecke einsetzen wollen und zudem mehrere Accounts damit steuern möchten, bedienen sich dem einfachen Trick einen beliebigen String an ihrer E-Mail-Adresse zu hängen. Die Bestätigungsmail kommt dennoch auf der eigentlichen Mail an. Somit ist es möglich, mehrere Twitter-Accounts mit derselben E-Mail-Adresse zu verknüpfen. Zum Beispiel heißt die normale Gmail-Adresse „hans.haus@gmail.com“ und die weiteren „hans.haus+fakebot@gmail.com“ oder „hans.haus+fakebot2@gmail.com“.

Nach erfolgreichem Erstellen eines Twitter-Accounts muss nun noch ein Developer Account beantragt werden, damit danach eine eigene App für den Twitter-Account entwickelt und auf die Twitter API zugegriffen werden kann. Für die Beantragung eines solchen Developer Accounts müssen verschiedene Fragen beantwortet werden und danach entscheidet Twitter, ob diese ausreichen, um einen eigenen Account zu erhalten. Dabei ist zu beachten, dass es keinen erneuten Zugang zur Developer Oberfläche gibt, falls die Beantragung beim ersten Mal nicht genehmigt wurde. Dieses stellt das zweite Hindernis dar. Denn nicht alle Developer Accounts werden freigegeben. Die RealSocialSEO veröffentlichte 2018 solche Antworten, mit denen angeblich 90% der Accounts, die diese Antworten benutzten, freigegeben worden sind [Real18]. In diesem Abschnitt wurde darauf verzichtet alle Frage aufzuführen.

Ein drittes Hindernis innerhalb des Twitter-Account umfasst die Verifizierung mit einer Handynummer. Ein Code wird dann per SMS an die Nummer verschickt und der Nutzer muss diese in seinem Twitter-Account angeben, um auszuschließen, dass es sich hierbei nicht um eine Maschine handelt. Dennoch gibt es bereits viele Möglichkeiten diesen Weg zu umgehen, bspw. mithilfe von Handynummern auf der Webseite „receive-smss.com“, die sich frei zugänglich im Internet befinden und deren letzten 50 SMS eingesehen werden können.

Nach dem Erstellen eines Twitter- und dem dazugehörigen Developer-Accounts kann nun die Entwicklung einer ersten App beginnen. Alle Schreib- und Leserechte müssen vergeben werden, damit der Bot auch alle Funktionen von Twitter nutzen kann.

Abschließend kann der Entwickler in der App-Umgebung die Twitter API Consumer und Access Tokens finden, die er später für die Verbindung seines Programmiercodes mit seinem Twitter-Account benötigt.

3.1.2 Python, Node.js und Ruby

Damit der eigene Social Bot selbstständig twittern kann, ist ein Programmiercode notwendig. Damit einher geht die Frage der passenden Entwicklungsumgebung. Es gibt viele verschiedene Programmiersprachen und somit Umgebungen, mit denen es möglich ist, einen komplexen Bot zu erstellen. In dieser Arbeit soll ein Teil dieser Tools vorgestellt und miteinander verglichen werden [Botw19a].

Folgende Anforderungen werden an eine geeignete Programmiersprache bzw. Programmierumgebung gestellt:

Tabelle 1: Anforderungen an eine Programmiersprache bzw. -umgebung

Einfachheit:	Die Sprache soll für jeden Entwickler verständlich sein. Zudem soll sie einfach in ihrer Codierung sein ohne, dass besonders auf die Syntax geachtet werden muss.
Erweiterbarkeit:	Der Programmiercode soll schnell und unkompliziert erweitert werden können. Zusätzlich sollen viele Bibliotheken oder Frameworks existieren, um jegliche Anwendungsfälle umsetzen zu können.
Einsatz KI:	Der Einsatz von intelligenten Methoden muss gegeben sein. Da der Social Bot automatisch und selbstlernend agieren soll, sollte die Sprache Anbindungen zu intelligenten Technologien besitzen.
Community:	Für Fehlerbehebungen oder Probleme wird eine große und aktive Community benötigt, um sich während der Entwicklung mit anderen austauschen zu können.

Fehlerbehebung:	Mithilfe von einfachem und schnellem Debugging sollen Fehler zügig behoben werden. Somit wird gewährleistet, dass die Konzentration auf den Funktionen der Software liegt.
Twitter Integration:	Da der Social Bot in Twitter interagiert, muss eine geeignete und effektive Twitter Integration für die Sprache existieren.

Python

Python ist eine objektorientierte Programmiersprache, entstanden durch Guido Van Rossum 1991, welche besonders oft für Skripte, aber auch für schnelle Anwendungsentwicklung genutzt wird. Python besteht aus einer klaren Syntax und einer einfachen Struktur. Eine große Standardbibliothek und der Python-Interpreter sind für alle Plattformen frei verfügbar. Zudem sind viele freie Module, Werkzeuge und Distributionen (auch von Drittanbietern), welche als Grundlage für eigene Programme verwendet werden können, vorhanden. Python hat zusätzlich eine Vielzahl an abstrakten Datentypen, mit denen komplexe Operationen in nur einer Anweisung deklariert werden können. Anweisungen werden nicht mit einer offenen und einer schließenden Klammer definiert, sondern lediglich durch Einrückungen. Zusätzlich sind Deklarationen von Variablen und Argumenten nicht nötig [Mich17].

Tweepy ist ein Open Source Python Package, welches den Zugriff auf die Twitter API über Python erlaubt. Das Package enthält viele Methoden und Klassen, die die API Endpunkte darstellen. Mithilfe von Tweepy können sich Entwickler auf die Funktionen konzentrieren, die sie auch wirklich benötigen. Die Mehrzahl der Twitter API Funktionen können über Tweepy genutzt werden. Ab Version 3.7.0 bestehen jedoch einige Einschränkungen in den Direct Messages, da vorgenommene Änderungen in der API nicht ordnungsgemäß funktionieren [Garc19].

KI-Projekte erfordern eine Menge unterschiedlicher Fähigkeiten und benötigen eine intensive Forschung. Python eignet sich mit seiner Stabilität, Flexibilität und der Vielzahl an Tools hervorragend für die Umsetzung von KI-Projekten. Mithilfe von Python können

Entwickler von der Erstellung über die Bereitstellung bis hin zur Wartung ihrer Software produktiv und zuverlässig arbeiten. Pythons Einfachheit, Konsistenz, der Zugang zu unterschiedlichen Bibliotheken und Frameworks für maschinelles Lernen, die Plattformunabhängigkeit, eine große Community und Flexibilität tragen dazu bei, dass Python einer der beliebtesten Sprachen in Bezug auf künstliche Intelligenz ist [Bek119].

Node.js

Node.js wurde 2009 von Ryan Dahl gegründet [Inte17]. Es basiert auf JavaScript und erlaubt das serverseitige Scripting. Node.js ist eine plattformunabhängige Open Source JavaScript Umgebung, welche meist für die Webserver Programmierung und die Ausführung von datenintensiven Echtzeitanwendungen verwendet wird [EDUC18a]. Es wird ein einzelnes Thread Modell genutzt, welches Ereignisschleifen verwendet. Dadurch reagiert der Server blockierend. Node.js basiert auf einer sehr leistungsstarken Engine Chrome V8 und ist somit optimal für beispielsweise Anwendungen mit Chat-Funktion [DA-117]. Mit dem Erscheinen von neueren Versionen kann es passieren, dass die gesamte Code-Basis neu erstellt werden muss. Dies geschieht bei allen JavaScript Technologien und zeigt die noch nicht ganz ausgereifte Entwicklung. Auch bei JavaScript muss keine große Zeit damit verbracht werden, neue maschinelle Lernverfahren zu programmieren. Für Node.js gibt es ebenfalls einige Bibliotheken, mit denen der Einstieg in das maschinelle Lernen vereinfacht wird [Mitt18].

Ruby

Ruby wurde 1995 das erste Mal veröffentlicht und von Yukihiro Matsumoto entwickelt. Ruby ist eine objektorientierte Open-Source-Programmiersprache, welche durch ihre Einfachheit, Portabilität und Erweiterbarkeit heraussticht. Anfangs wurde es unter Linux entwickelt, arbeitet jedoch heute bereits auf verschiedenen Plattformen wie Windows, Mac Os oder UNIX. Eine definierte Ordner- und Dateistruktur und strenge Regeln für Dateinamen sind essenziell für die Entwicklung eines Ruby Projektes. Es unterstützt Multi-Threading und eignet sich daher besser für CPU-intensive Anwendungen. Ruby funktioniert nicht asynchron, sondern arbeitet alle Anweisungen sequenziell ab. Es gibt daher keine Rückrufe. Zudem ist Ruby eine sehr beständige Programmiersprache, die

wenig verändert wird. Zusätzlich verfügt sie über wenig Skalierbarkeit, jedoch über einer großen Erweiterbarkeit und Flexibilität [EDUC18b].

Es existiert eine große Standardbibliothek für Ruby und sie beinhaltet eine Schnittstelle für UNIX. Eine Mehrfachvererbung, die ermöglicht, dass zuvor geschriebener Code wiederwendet werden kann, existiert in Ruby nicht. Dafür gibt es das „Mixins“ Konzept, welches Codeklassen beinhaltet. Diese können mit dem Programmiercode verbunden werden. Alle Methoden können in Ruby ausgetauscht werden, da alle Klassen offen sind. Mithilfe des Frameworks Ruby on Rails wird sie als Scriptsprache auf Webservern verwendet [10ho19].

Eine Vielzahl von Bibliotheken für maschinelles Lernen und KI gibt es leider nicht. Daher wird im Zusammenhang mit KI fast nie die Sprache Ruby genannt und es wird dort oft auf Python zurückgegriffen.

Tabelle 2: Vergleich von Python, Node.js und Ruby

	Python	Node JS	Ruby
Sprache	High level, plattform-unabhängige und objektorientierte Sprache	JavaScript-Umgebung für serverseitigen Code	Open Source und plattform-unabhängige Sprache
Entwicklung	1991	2009	1995
Coding	Wenige Zeilen Code, einfache Sprache	JavaScript, Grundlagen einfach zu lernen	Einfache Sprache, viele Lerntutorials
Performance	Nicht empfehlenswert für speicherintensive Aktivitäten	Dank V8 Engine leistungsstark	Langsam in der Leistung

Einsatzbereiche	Webanwendungen, Integration mit Back-End Anwendungen, maschinelles Lernen, KI, Netzwerkprogrammierung	Webanwendungen und Entwicklung von Webseiten	Häufig in Big Data Projekten, CPU-intensive Anwendungen
Projektgröße	Ideal für große Projekte	Ideal für kleinere Projekte	Ideal für große Projekte
Fehlerbehebung	Einfaches Debugging und schnelle Fehlerbehandlung möglich	Im Gegensatz zu Python schlechter	Debuggen von Anwendungen aufgrund von Abstraktheit zeitaufwändig
Community	Große Community	Große Community	Aktivste Entwickler-Community
AI Integration	Viele Pakete	Einige Bibliotheken	Wenig Bibliotheken

Alle drei Technologien eignen sich besonders gut für serverseitige Entwicklungen. Sie beschränken sich auf minimale Codezeilen, um Aufgaben schnell zu erledigen. Zudem setzen alle auf eine schnelle Entwicklung. Ruby, Python und Node.js zeichnen sich durch ihre objektorientierte Logik und das schnelle Einbinden von Apps oder Funktionen aus. Entwickler müssen bei keiner der Technologien von Grund auf neu programmieren, sondern können sich an vielen Bibliotheken und Frameworks bedienen.

Für den in dieser Arbeit entwickelten Twitter-Bot wurde die Sprache Python mit dem Package Tweepy ausgewählt, da die Vorteile von Python überwogen haben. Besonders die Möglichkeit mehrere Frameworks für das maschinelle Lernen anzubinden und somit das Ziel einen intelligenten Twitter-Bot zu entwickeln, machte die Entscheidung für Python einfach.

3.1.3 KI Frameworks

Viele verschiedene Bibliotheken und Frameworks für das maschinelle Lernen existieren bereits heutzutage. Besonders für Python hat fast jedes KI Tool eine Implementierungsschnittstelle. Einige dieser Frameworks sollen im Folgenden näher betrachtet werden und aufzeigen, wie viele Technologien für KI bereits im Internet zu finden sind.

TensorFlow

TensorFlow wurde 2015 von Google entwickelt und ist ein Open-Source Framework für künstliche Intelligenz und maschinelles Lernen. TensorFlow gibt die Möglichkeit lernende neuronale Netze, Algorithmen für maschinelles Lernen und Deep Learning Modelle zu entwickeln [LuLi18a]. KNN soll die Verknüpfungen in unserem Gehirn und das menschliche Gehirn selbst nachempfinden. Miteinander verbundene Neuronen in einem abstrakten Modell sollen durch ihre Verknüpfung miteinander verschiedene Anwendungsprobleme u.a. in der Statistik oder Wissenschaft lösen. Neuronale Netze gehören zu dem Bereich der KI und müssen, bevor sie ein Problem lösen können, trainiert werden [LuLi18b].

Die Grundlage von TensorFlow besteht aus einem Datenflussgraphen, der aus einem Diagramm besteht, welcher ein mathematisches Problem abstrahiert darstellt. Das Diagramm besteht aus Knoten und Kanten, welche miteinander verbunden sind. Durch die Knoten werden mathematische Operationen und durch die Kanten Daten dargestellt, welche von einem zum anderen Knoten übertragen werden. Somit entsteht dann ein Graph, der ein mathematisches Modell für ein neuronales Netz repräsentiert [LuLi18a].

Aufgrund einer großen, umfangreichen und detaillierten Dokumentation und das regelmäßige Beliefern von Updates steigt die Beliebtheit von TensorFlow bei Entwicklern enorm [Nguy18].

Microsoft Bot Framework – Service „LUIS”

Das Microsoft Bot Framework beinhaltet eine große Sammlung von Tools, Diensten und SDKs (Software Development Kit), welche den Entwicklern die Möglichkeit bieten, Bots an einem Ort zu erstellen, zu testen und zu verwalten [Micr19a].

Als Teil vom Azure Bot Service stellen die Cognitive Services von Microsoft mehrere KI Dienste bzw. APIs zur Verfügung, um den eigenen Bot eine künstliche Intelligenz zu verschaffen. Die Intention des Users kann beispielsweise der cloudbasierte API-Service LUIS erkennen. Der Language Understanding Intelligent Service (LUIS) kann akustische Sprache verstehen und eigene Texte in Sprache umwandeln [WaFl18]. Der Dienst basiert auf maschinellem Lernen und verwendet dafür das verstärkte Lernen, damit die Qualität der erstellten Modelle ständig verbessert werden kann [Micr19b]. Damit LUIS die menschliche Sprache versteht, muss diese vorher in einem neuen eigenen oder in einem vordefinierten Modell angelernt werden.

Dialogflow

Dialogflow ist ein Google-Dienst, welcher auf der Google Cloud Plattform ausgeführt wird und sich mit dem Natural Language Processing und der digitalen Kommunikation beschäftigt [Dial19]. Sie unterstützt Entwickler dabei, eigene Chatbots für verschiedenste Einsatzbereiche zu erstellen und bietet dies in mehr als 20 Sprachen. Die Plattform stellt eine komplette Entwicklungsumgebung mit eigenem Editor, einer Library und verschiedenen Tools bereit. Dialogflow verwendet die Cloud Services von Google in Verbindung mit maschinellem Lernen und KI. Eine Vielzahl an Anbindungen mit unterschiedlichen Programmiersprachen und Schnittstellen zu diversen Anwendungen wie Facebook, Slack oder Twitter sind möglich.

Ein sogenannter Wissensconnector arbeitet mit Google-Technologien und bietet die Möglichkeit Unternehmensdaten zu nutzen. Auch die Services *Google Cloud Speech-to-Text* und *Text-to-Speech* finden hier für die Spracherkennung ihre Anwendung [LuLi19].

Watson

Im Jahr 2011 gewann der Supercomputer Watson von der amerikanischen Firma IBM in der US Show „Jeopardy“ gegen die besten menschlichen Spieler. Das Ziel von „Jeopardy“ ist es, Fragen aus verschiedenen Wissensbereichen so schnell wie möglich zu beantworten. Watson schaffte dies ohne jeglichen Internetzugriff. Er benötigte nur eine lokale Datenbank. Nach diesem großen Erfolg wurde viel in neue erweiterte Technologien und in den Aufbau des neuen Geschäftsbereiches investiert. Heutzutage

wird Watson nicht mehr als ein System verstanden. Die in der IBM Cloudplattform Bluemix verfügbaren Watson Services bzw. APIs stellen heute das Kernkonzept von Watson dar. Die Services bestehen aus den kognitiven Funktionen und können dank API für externe Anwendungen verfügbar gemacht werden [Beck19].

Die Nutzer der Watson Services erhalten Zugang zu umfangreichen kognitiven Technologien, welche eine sichere und schnelle Entwicklung einer intelligenten Anwendung ermöglichen. Es können Bilder oder Videos analysiert oder Textfragmente, Schlüsselwörter und Stimmungen interpretiert werden. Der Vorteil dieser Services liegt bei der stetigen Weiterentwicklung und Verbesserung. Jede Optimierung erfolgt umgehend in den gesamten eigenen Anwendungen, sodass eine kontinuierliche Verbesserung ohne Änderung in den Anwendungen erfolgen kann. Zusätzlich können die bestehenden Services individuell erweitert und mit eigenen Daten trainiert werden.

Drei der Watson Services in Bluemix sollen kurz vorgestellt werden. Der **Visual Recognition** Service beschäftigt sich mit der Bilderkennung. Mit dieser API lassen sich mithilfe von Objekterkennung, Gesichtserkennung und Content-Tagging verschiedenste Bilder analysieren. Das vordefinierte Basismodell lässt sich individuell erweitern und auf die eigenen Einsatzbereiche anwenden.

Anhand von Social Media Quellen lassen sich mit dem **Watson Personality** Service Persönlichkeitsmerkmale oder auch Bedürfnisse eines Nutzers ableiten. Somit können beispielsweise für die Personalabteilung, die Bewerber mit den besten Werten und Merkmalen herausgesucht werden.

Der **Watson Tone Analyzer** analysiert in einem Text die verwendete Tonalität und Emotionen. Diese Daten können dabei helfen, dass bei der Kommunikation zwischen einem Bot und einem Menschen die Unterhaltung nicht in eine falsche Richtung läuft. Der Service gibt für jede Emotion eine Wahrscheinlichkeit aus, mit der sie im jeweiligen Text vorkommt [Beck19].

3.2 Konzept - Twitter-Bot „Animal Tweeter“

Im Folgenden wird kurz das Konzept des Twitter-Bots „Animal Tweeter (@tweeter_animal)“ vorgestellt, welcher innerhalb dieser Master-Thesis entwickelt wurde. Der Account beschäftigt sich mit den verschiedensten Themen des Tierschutzes und der Tiere allgemein auf unserem Planeten. Weiterhin wird der Account auf Englisch entwickelt, da Englisch die meist verwendete Sprache ist und besonders die maschinellen Lernverfahren am effektivsten in dieser Sprache arbeiten. Der Hauptfokus dieses Bots liegt auf dem automatisierten Tweeten und Retweeten und der selbstständigen Kommunikation mit anderen Nutzern per Direct Message oder auch in den Kommentaren. Dafür wurden verschiedene KI-Technologien verwendet, die bereits im vorherigen Kapitel kurz dargestellt wurden.

Der „Animal Tweeter“ Account wurde dank der Antworten von RealSocialSEO als Developer Account freigegeben. Als Entwicklungsumgebung und Sprache wurde Python verwendet, da alle genutzten Technologien für maschinelles Lernen einfach über eine API und ein Package erreichbar sind. Alternativ dazu können auch die Sprachen Ruby und JavaScript mit der Umgebung Node.js verwendet werden. Zusätzlich wurde das Package Tweepy genutzt, um auf die Twitter API mit all ihren Funktionen zugreifen zu können.

Der Bot umfasst mehrere Python-Dateien (siehe Bild 8). Zum einen gibt es die unterstützenden Dateien, wozu die credentials.py Datei und die watson_toneanalyzer.py Datei zählt. Die „Credentials“-Datei enthält die Twitter API Tokens, welche es erlauben, mit dem Twitter-Account zu kommunizieren. Die „Watson Tone Analyzer“ Datei kommuniziert mit dem KI Framework Watson und gibt diesen einen Tweet mit, damit das Tool dann die Stimmung des Tweets erkennt und der Datei zurückgibt. Als Alternative dazu kann auch der Microsoft Dienst LUIS genutzt werden, da dieser ebenso die Fähigkeit besitzt, die Intention eines Nutzers und dessen Stimmungslage zu erkennen.

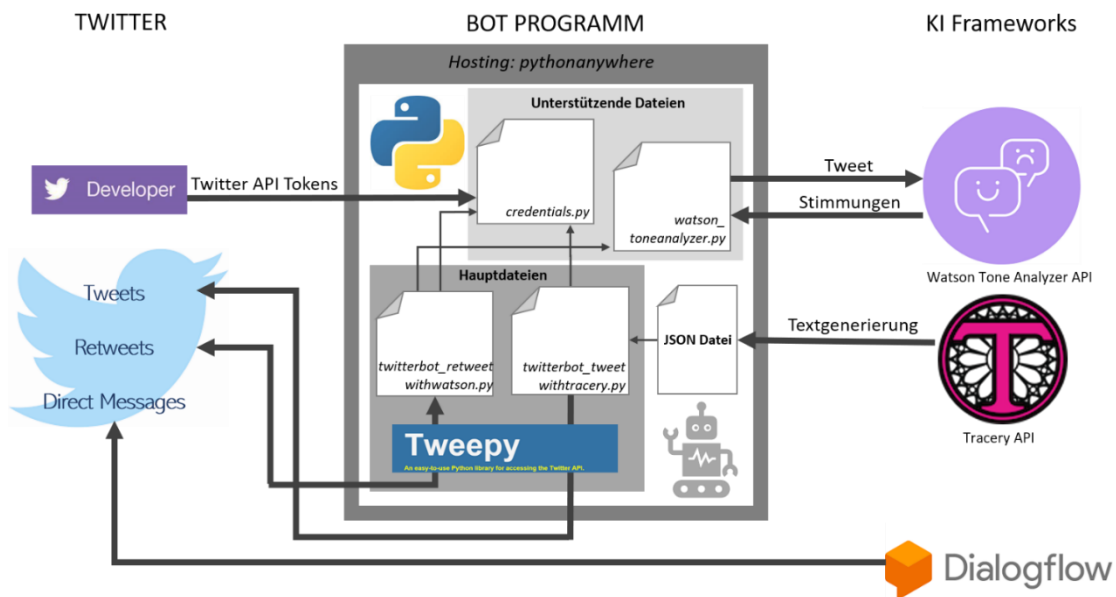


Bild 8: Konzept des Bots „Animal Tweeter“, eigene Darstellung (Größere Grafik s. Anhang)

Zum anderen existieren zwei Hauptdateien zum Tweeten und Retweeten. Die „Retweet“-Datei soll einen Tweet, welcher über bestimmte Hashtags gesucht wird, retweeten. Die Datei nutzt die „Watson“-Datei, um die Stimmung eines Tweets zu erhalten. Anhand der Stimmung kann dann entschieden werden, ob der Tweet überhaupt retweetet wird. Die „Tweet“-Datei soll einen automatisch generierten Text in der eigenen Twitter Timeline posten. Dafür wird das Tool Tracery genutzt. Mithilfe einer individuell erstellten JSON-Datei, die verschiedenste Worte und Parameter beinhaltet, kann Tracery diese zu einem sinnvollen Text zusammenfügen. Zu Beginn der Entwicklung wurde jedoch das Tool Tensorflow genutzt. In den nachfolgenden Kapiteln werden die Gründe für die Nichtnutzung dieses Frameworks genannt. Da Tensorflow nicht den Anforderungen entsprach, wurde es auch nicht in das Konzeptbild mit aufgenommen.

Die zuvor beschriebenen Python-Dateien werden auf der Hosting Plattform „pythonanywhere“ jeden Tag zu einer bestimmten Uhrzeit ausgeführt. Alternativ dazu kann bspw. auch ein eigener Amazon Web Services (AWS) Server mit einem beliebigen Betriebssystem genutzt werden. Da dies jedoch nach ein paar Wochen kostenpflichtig wird, wurde die freie Webseite „pythonanywhere.com“ verwendet, bei der die Python Dateien direkt gehostet werden.

Um nun auch mit anderen Nutzern kommunizieren zu können, wurde die KI Technologie Dialogflow genutzt. Eine andere NLP Technologie wären die Microsoft Cognitive Services, mit denen ebenfalls ein Chatbot konfiguriert werden kann. Dialogflow wurde ausgewählt, da diese Technologie sehr benutzerfreundlich ist und vordefinierte Agenten besitzt. Der selbst angelegte Dialogflow Agent arbeitet unabhängig von den Python-Dateien und wurde auf der Webseite direkt konfiguriert. Diese hat eine Twitter Integration und kann somit mithilfe der API Tokens direkt auf Twitter zugreifen und auf Nachrichten reagieren.

3.3 Implementierung und Test - Twitter-Bot „Animal Tweeter“

3.3.1 Erstellung eines Tweets

Für die selbständige und automatische Erstellung eines Tweets wurde zunächst die Technologie TensorFlow genutzt. Diese ermöglicht es, ein Modell auf der Grundlage eines oder mehrerer Texte bzw. Bücher anzulernen. Damit können automatisch Texte mit einer beliebigen Anzahl von Satzzeichen generiert werden und somit auch Tweets.

Als Grundlage dieser Deep Learning Funktion wurde der Python-Code von Max Deutsch, einem Entwickler im Deep Learning Bereich, herangezogen. Der Programmiercode nutzt einen beliebigen Input-Text, wie einen Artikel oder ein Buch, und die TensorFlow API, um eigene Grammatikregeln zu erlernen und zu trainieren. Die Nutzer dieses Codes müssen lediglich die KI Technologie TensorFlow installieren und einen Text ihrer Wahl in den Code einfügen. Danach wird das Modell über mehrere Stunden mit diesem Text trainiert. Da dieses Lernen sehr zeit- und rechenintensiv ist, wurde eine virtuelle Linux Maschine verwendet. Im Anhang dieser Arbeit befinden sich die Daten und Informationen zu diesem System.

Der Bot sollte mithilfe dieses Frameworks eigenständig und regelmäßig eigene Tweets verfassen, welche dem Thema Tierschutz, Tiere und Umwelt zuzuschreiben sind. Dafür wurden verschiedene Bücher angelernt.

Als erstes Testszenario wurden mehrere Bücher aus dem Bereich Animal Welfare und Animal Encyclopedia genutzt. Nachdem das System viele Stunden trainiert wurde,

erfolgte der Test auf die Nutzbarkeit dieses Modells. Die Zeichenanzahl wurde auf 25 Zeichen beschränkt. Im nächsten Abschnitt folgen einige selbst vom System generierte Texte.

```
Zoos identifying theorizations scientific than the actions of justice, of anima
ls[00]floors service Part and occur between dogs and law argue that nonhuman anim
als are part
```

```
derive scientific than attempting to collaborate with animals. The sake of zoo
mice whose government use related to animals, both some more ethical agencies w
ill study
```

```
advanced: ecological se However, animal protection are derived on the ability o
f stress and, even in this demonstrated as that people begin to our pain anatom
ical,
```

Keiner dieser Texte kommt in den Beispieltextrn, welche antrainiert wurden, vor. Es fällt sofort auf, dass zwar Texte generiert wurden und in dieser Form in den antrainierten Büchern vorliegen, jedoch die Grammatik noch nicht einwandfrei funktioniert. Bleibt das Modell auf diesem Stand, müsste der Entwickler manuell die Texte formatieren, damit sie die richtige Satzbaustellung und Klarheit bekommen.

Auch das zweite Trainingsbeispiel zeigt, dass das Modell in den hier genannten zwei Versuchen nicht effektiv ist. Dazu wurde das Buch „Animal Farm“ von George Orwell zum Trainieren verwendet. Folgende Texte wurden vom System erstellt:

```
installed, banner own march which and them had two times [00]round further rumour
s for not able of black been tell such had thought, paused and
```

```
did, market dangerous reverent dispersed. recovery. Boxer[00]begun cropped Re-edu
cation memories of fired thanks even the more only beasts of moment, a directio
n It on a
```

```
name herself today. false step, Snowball could so, out could So if seemed they
the Here down But down they and few that absented was cannot
```

Es wird davon ausgegangen, dass mehr Trainingsdaten benötigt werden, um eine qualitative und hochwertige Textgenerierung zu ermöglichen. Weiterhin muss bei solchen Deep Learning Systemen oder auch beim maschinellen Lernen generell darauf geachtet werden, dass die Trainingsdaten auf ihre Qualität hin kontrolliert werden. Somit können Fehler in der Generierung vermieden werden. Codierungsfehler oder auch eine

falsche Grammatik oder Rechtschreibung führen automatisch zu einer verkehrten Erstellung eines Textes. Max Deutsch zeigt jedoch in unterschiedlichen Szenarien, wie die Erstellung von Szenen für Star Wars oder die Generierung von Harry Potter Texten, dass dieses Modell in einem gewissen Grad funktioniert.

Jedoch musste in dieser Arbeit festgestellt werden, dass dieses Modell in dieser Form nicht auf jeden Anwendungsbereich erfolgreich anpassbar ist. Da diese Technologie für den hier genannten Twitter-Bot „Animal Tweeter“ nicht funktionierte und nicht den Anforderungen eines automatisch generierten Tweets entsprach, wurde der Fokus auf das Retweeten und das Kommunizieren mit anderen Twitter-Accounts gesetzt.

Dennoch wurde mithilfe der Plattform Tracery ein Modell zum automatischen Erstellen eines Tweets genutzt. Mithilfe von Tracery können verschiedene Regeln erstellt werden, die dem System aufzeigen, wie der Text Stück für Stück zusammengesetzt wird. Eine Tracery Grammatik sieht immer wie folgt aus [Parr16] :

```
{  
  "origin": "Hallo #Begrüßung#!"  
  "Begrüßung": "Welt"  
}
```

Somit entsteht der Satz „Hallo Welt“. Jedoch können auch Alternativen hinzugefügt werden, damit jedes Mal ein anderer Satz herauskommt. Beispielsweise:

```
{  
  "origin": "Hallo #Begrüßung#!"  
  "Begrüßung": ["Welt", „Zusammen“, „ihr da“]  
}
```

Damit können bereits die drei Sätze „Hallo Welt!“, „Hallo, ihr da!“ und „Hallo Zusammen!“ generiert werden. Demnach können komplexe Modelle gebaut und immer wieder neue Texte erstellt werden.

Weiterhin konnte der Animal Tweeter von diesem Tool profitieren und es wurde ein komplexes Modell bzw. eine JSON Datei und eine ausführbare Python Datei *twitterbot_tweetwithtracery.py* erstellt (siehe Anhang), mit welchen maximal dreimal pro Tag ein Tweet verfasst wird. Diese Tweets befassen sich ebenso mit Tieren, kurzen Smalltalks und dem Tierschutz. Dieses Modell kann mit weitaus mehr Regeln erweitert

werden. Jedoch reicht dies aus, um das einfache automatisierte Erstellen von Tweets darzustellen.



Bild 9: Automatisch verfasste Tweets

Schließlich kann der Animal Tweeter nun eigene Tweets verfassen und in seiner Timeline veröffentlichen (weitere Testszenarien im Anhang). In dem nachfolgenden Kapitel wird gezeigt, wie auf die API zugegriffen wird und die Funktionen genutzt werden können. Das Package Tweepy, welches auch für die Erstellung eines eigenen Tweets in Python verwendet wurde, wird im folgenden Abschnitt ebenso erläutert.

3.3.2 Retweeten, Liken und Kommentieren

Wie bereits in den vorherigen Kapiteln angesprochen, wird das Python Framework Tweepy genutzt, mit dessen Hilfe mit der Twitter API agiert werden kann. Dies lässt sich sehr einfach mit dem Standard Package pip, welches Python Module managt, installieren. Zu Beginn wird eine Datei *credentials.py* erstellt, die die eigenen Twitter Access und Consumer Tokens beinhaltet. Damit kann dann eine weitere Datei angelegt werden, die den eigentlichen Quellcode umfasst.

Retweetet werden Posts von anderen Nutzern, die seit dem 01. August 2019 auf Twitter erstellt und mit verschiedenen Hashtags zu den Themen Tierschutz versehen wurden. Ziel ist es, die Tweets herauszusuchen, die ein positives Gefühl bei den Nutzern hinterlassen. Der Bot soll keine Hass-Tweets oder ähnliches unterstützen und steht lediglich für den guten Zweck.

Mithilfe einer Schleife werden Tweets mit den bereits vorgegebenen Merkmalen gesucht und dann gepostet. Hierbei wurde darauf geachtet, dass menschliche Schlafenszeiten eingehalten werden. Der Bot startet mithilfe eines Cronjobs auf der Hosting Seite `pythonanywhere.com` um 8:00 Uhr morgens und endet abends um 22:00 Uhr. Dafür wurde eine Datumsvariable erstellt, welche jedes Mal, bevor ein Tweet gepostet wird, abgefragt wird. Findet die Abfrage nach der festgesetzten Endzeit von 22:00 Uhr statt, wird die Schleife sofort abgebrochen und der Bot agiert nicht mehr. Somit weckt es den Anschein, dass sich hinter diesem Profil keine Maschine, sondern ein Mensch verbirgt, der tatsächlich schläft und abwesend ist.

Um nun die Stimmung eines Tweets herauszufinden und mögliche Reaktionen daraus ableiten zu können, muss diese in irgendeiner Form erfasst werden. Da die Entwicklung von NLP sehr weit fortgeschritten ist, gibt es bereits in der Wissenschaft mehrere Tools, um die Emotionen eines Textes erkennen zu können. Für den vorliegenden Account wurde der Watson Service „Watson Tone Analyzer“ eingesetzt. Dieser verfügt über eine Python Schnittstelle und kann in jedes Projekt mithilfe einer API leicht integriert werden. Für dieses Tool wurde eine neue Datei `watson_toneanalyzer.py` erstellt. Neben der API Integration findet sich in dieser Datei eine Funktion zum Ableiten der Stimmung in einem Tweet. Der Service bietet die Emotionen „Joy“, „Analytical“, „Confident“, „Sadness“, „Fear“, „Tentative“ und „Anger“. Für jede Emotion, die in dem Tweet vorkommt, wird ein Score erstellt. In der erstellten Funktion wird nun die Stimmung zurückgegeben, die den größten Score hat und somit die Emotion des Tweets am besten repräsentiert.

Folgende Beispielsätze sollten vom Service analysiert werden:

```

text1 = 'I did not like you'
text2 = 'Oh I Love animals!!'
text3 = 'I wanna cry. The poor animals fight for their lives in the fishing
nets. '
text4 = 'I can not do anything with this stuff! I do not need that and I do not
like animals!!'
text5 = 'I support animal welfare and try to do everything to make their life
easier!'

```

Bild 10: Beispielsätze für das Testen des Watson Tone Analyzer Services.

```

Text1: Tentative
Text2: Joy
Text3: Sadness
Text4: Tentative
Text5: Joy

```

Bild 11: Ergebnisse der Beispielsätze

Das Ergebnis zeigt, dass der Watson Tone Analyzer die Stimmung in den meisten Fällen sehr gut zuordnet. Daher wurden folgende Bedingungen in der Schleife vom Hauptquellcode gewählt.

Hat ein Tweet die Stimmung „Joy“, „Analytical“ oder „Confident“ handelt es sich bei diesem Tweet höchstwahrscheinlich um eine positive Stimmung. D.h. Animal Tweeter wird diesen Post retweeten und liken. Anhand von Zufallswerten wird dann dem Nutzer, der den Tweet ursprünglich verfasst hat, gefolgt und ein Kommentar wird hinterlassen. Der Zufallswert MaybeComment gibt eine eins oder null an. Bei einer eins wird ein Tweet zusätzlich zum Retweeten auch mit einem Text kommentiert. Dafür wurden zwei Listen angelegt, eine für positive Stimmungen (*happy_list*) und eine für traurige (*sadness_list*). Mithilfe des Zufalls werden demnach Worte oder Wortgruppen aus den Listen in einen Kommentar implementiert. Somit wird sichergestellt, dass der Bot nicht immer nach denselben Mustern arbeitet und einen echten Menschen imitiert. Dadurch wird bei den positiven Stimmungen ein Kommentar mit einem String aus der Happy-Liste verfasst.

Wenn der Tweet jedoch traurig ist und mit der Emotion „Sadness“ ausgegeben wird, wird der Tweet gepostet, jedoch nicht favorisiert. Zudem wird dem Ersteller des Posts nicht gefolgt, auch nicht zufällig. Dennoch wird, wenn die Zufallszahl MaybeComment auf eins steht, ein trauriger Kommentar mit einem String aus der Sadness-Liste hinterlassen. Befindet sich der Tweet in keiner der zuvor genannten Stimmungen, wird

der Tweet nicht beachtet und somit nicht getweetet. Hierbei wird es sich mit einer hohen Wahrscheinlichkeit, um einen negativen Post handeln.

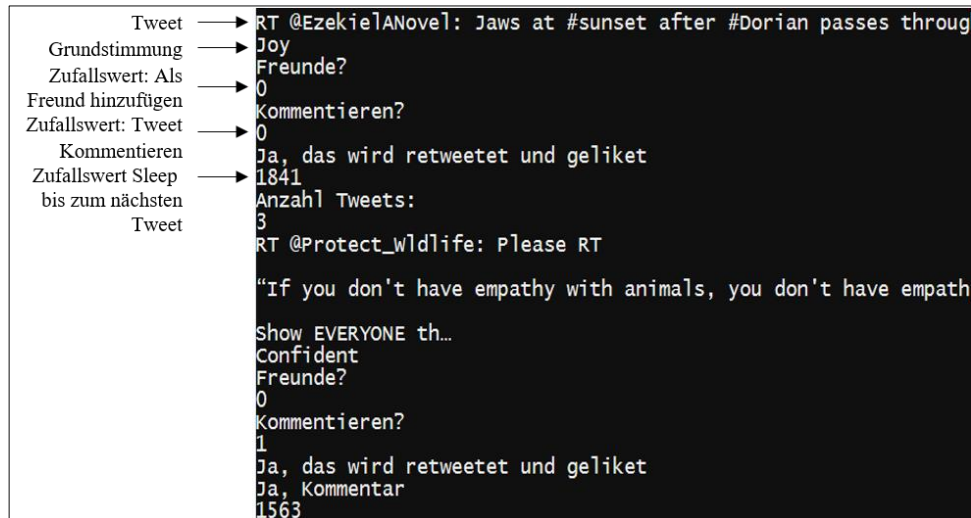


Bild 12: Retweet Beispiel

Mit der Sleep Funktion wird sichergestellt, dass die Retweets nicht nach bestimmten Regeln bspw. alle 20 Minuten, wie dies oft der Fall bei den Standard-Bots ist, erfolgen. Animal Tweeter retweetet zwischen 1500 und 5400 Sekunden. Die Schleife wird dann verlassen bzw. abgebrochen, wenn die Endzeit erreicht wurde oder die maximale Anzahl an Retweets erfolgte. Die Maximale Anzahl an Retweets pro Tag wurde auf 15 gesetzt, da somit gewährleistet wird, dass der Bot mindestens einmal pro Stunde retweetet.

Mit diesen Funktionen lassen sich nun Tweets mit bestimmten Hashtags über einen festgelegten Zeitraum suchen und dann retweeten. Darüber hinaus kann der Bot dem Ersteller des Tweets folgen, diesen Tweet liken und ein passendes Kommentar hinterlassen. Weiterhin kann der Bot Tweets, welche sich auf seinen Account beziehen, bspw. sogenannte Replies (Antworten auf verfasste Tweets oder Kommentare), favorisieren. Der folgende Tweet wurde als Antwort zu einem Kommentar vom Animal Tweeter verfasst (siehe Bild 13). Als der Bot das nächste Mal wieder aktiv wurde, wurde auch dieser Tweet von ihm gelikt.



Der Tweet wurde geliket:1173311647609511936

Bild 13: Favorisieren eines Replies von einem Nutzer

All diese Funktionen kann der Bot voll automatisch und selbständig. Das manuelle Eingeben von Daten und die Kontrolle des Entwicklers entfallen.

Folgend wurden die zuvor genannten Funktionalitäten in einem kleinen Beispiel getestet (im Anhang siehe Retweet Testszenario 2). Der Bot wird um 10:15 Uhr gestartet und soll um 11:15 Uhr wieder enden. Es sei denn, die maximale Tweet Anzahl von 5 wurde erreicht, dann wird der Bot ebenfalls beendet. Die Tweets können im Abstand von 5 bis 25 Minuten erfolgen. Daher wird der Sleep-Wert zwischen 300 und 1500 Sekunden gesetzt.

Der erste und zweite Tweet wurde vom Animal Tweeter erfolgreich retweetet und kommentiert. Zudem wurde den beiden Nutzern gefolgt. Der dritte Tweet konnte nicht analysiert werden, da keine Stimmung gefunden wurde. Daher wurde dieser nicht gepostet. Der letzte und somit vierte Tweet wurde gepostet, aber nicht kommentiert. Da danach die Abfrage eine Uhrzeit von 11:28 Uhr aufwies, wurde die Schleife verlassen und der Bot ist inaktiv. Alle Funktionen wurden erfolgreich ausgeführt inklusive der Sleep-Funktion.

Der Bot Animal Tweeter kann somit erfolgreich automatisch Retweets absetzen und selbstständig Kommentare hinterlassen.

Die zwei Dateien *twitterbot_tweetwithtracery.py* und *twitterbot_rewteetwithanalyzer.py* werden auf der Hosting Plattform pythonanywhere täglich um 8:00 Uhr automatisch gestartet.

3.3.3 Unterhaltung in den Direct Messages

Damit der Bot autonom eine Unterhaltung mit einem anderen Nutzer in den Direct Messages in Twitter führen kann, wurde ein Tool gebraucht, welches mit Sätzen des Small Talks anlernbar ist. Dafür wurde die Technologie von Google - „Dialogflow“ - genutzt. Dieses Framework hat eine einfache und übersichtliche Benutzeroberfläche und konnte durch seine Vielzahl vorhandener Sprachagenten, wie bspw. einen Smalltalk Agenten, herausstechen. Besonders bei solchen Technologien muss darauf geachtet werden, den Bot anhand von korrekten Trainingsdaten anzulernen und somit die Möglichkeit zu bieten, möglichst viele potenzielle Konversationen abzudecken. Für den hier vorliegenden Bot wurden zuvor definierte Small Talks in das Training integriert, aber auch eigene Unterhaltungstexte hinzugefügt und angelernt.

Zu den üblichen Small Talk Sätzen gehören unter anderem die Fragen nach dem Geburtstag, dem Hobby oder ob es sich hierbei um ein Chatbot handle. Zudem gehören unter anderem die Unterhaltung über Langeweile oder eine einfache Begrüßung oder Abschied dazu.

Der Ablauf einer Konversation in Dialogflow umfasst drei wichtige Vorgangsweisen. Zu Beginn gibt der Nutzer einen beliebigen Text ein, diese Eingabe wird dann von dem zuvor erstellten Agenten geparkt und danach erfolgt dann eine Antwort an den Nutzer. Mithilfe von Intents werden Antworten zu bestimmten Nutzereingaben definiert und zugeordnet. DialogFlow beinhaltet eine Liste von Intents, die entweder selbst angelegt oder von bereits existierenden Agenten übernommen werden können. Beispielsweise kann ein Nutzer die Frage „When is your birthday?“ (siehe Bild 14) stellen. Der Agent ordnet dann diese Eingabe einem entsprechenden Intent zu, bspw. „Birthday“ und gibt eine vordefinierte Antwort zurück.

Training phrases ?

Add user expression

when were you born

when do you have birthday

when is your birthday

your birth date

when do you celebrate your birthday

date of your birthday

what's your birthday

Responses ?

DEFAULT +

Text Response

- Wait, are you planning a party for me? It's today! My birthday is today!
- I'm young. I'm not sure of my birth date.
- I think in april.
- Is that so important?
- Enter a text response variant

ADD RESPONSES

☐ Set this intent as end of conversation ?

Bild 14: Einstellungen für ein Intent

Ein Intent besteht somit aus zwei großen Hauptkomponenten, den Trainingssätzen und den Antworten. Die Trainingssätze dienen als Beispiel dafür, was der Nutzer sagen könnte, damit diese dann dem Intent zugeordnet werden können. Dialogflow kann diese Formulierungen automatisch um weitere Äußerungen erweitern. Daraufhin können dann bestimmte Antworten erstellt werden, die dann zusammen mit den Trainingssätzen nach dem Speichern des neuen Intents antrainiert werden. Zusätzlich kann innerhalb des Intents auch eine Entität konfiguriert werden. Mithilfe von Entitäten können bestimmte Informationen und Elemente aus der Eingabe eines Nutzers entnommen werden, wie zum Beispiel Namen, Adressen oder Datumseingaben. Diese können dann für weitere Intents oder auch Antworten genutzt werden. Zudem können auch sogenannte Kontexte erstellt werden. Diese helfen dabei, von einem Intent zum anderen Intent zu springen und in einer Unterhaltung beim Kontext zu bleiben. Beispielsweise fragt der Agent einen Nutzer „Magst du Tiere?“, der Nutzer antwortet mit „Ja“, dann wird zu einem anderen Intent gesprungen und der Agent fragt „Hund oder lieber Katze?“. Der Nutzer antwortet hierbei „Katze“ und der Agent springt somit zum nächsten Intent und schickt ein Bild von einer Katze. Bei diesem Unterhaltungsstrang wird der Kontext nicht verlassen. Somit ist eine umfangreiche Konversation möglich.

Bei dem in dieser Arbeit beschriebenen Agenten wurde auf Entitäten und Kontext verzichtet, da als höheres Ziel die einfache Unterhaltung mit dem Nutzer und das automatische Versenden von Nachrichten im Vordergrund steht. Jedoch kann dies stetig erweitert werden und der Bot kann am Ende noch viel authentischer wirken.

Ein Intent kann der Entwickler auch als das Ende einer Konversation klassifizieren. Dies wurde bei dem Bot Animal Tweeter bspw. bei dem Satz „Thank you“, „Thanks“ oder „Thank you very much“ verwendet. Der Bot antwortet auf diesen Satz dann nicht mehr. Mithilfe von Fallback Intents können alle Nutzereingabe erfasst werden, die nicht als regulärer Intent in das Training aufgenommen wurden.

Als eine weitere Funktion, die sich jedoch laut Dialogflow noch in der Beta-Version befindet, ist die Erstellung einer Knowledge Base. Hier können FAQs, sogenannte Fragen-Antworten, aus Webseiten, Artikel oder Dateien verwendet und antrainiert werden. In dem beschriebenen Agenten wurden neben den vielen Small-Talk Intents auch einige Tierschutz FAQs aus dem Internet antrainiert. Somit können auch einige Fragen zu diesem Thema vom Bot beantwortet werden.

Der erstellte Agent „SocialBotTwitter“ wurde nicht über eine API angebunden, sondern über eine Integrationsschnittstelle von DialogFlow selbst. Hierbei können sehr einfach und übersichtlich die Twitter App Daten eingegeben werden und der Agent ist auf dem jeweiligen Twitter-Account aktiv.

Anhand einer Beispiel-Unterhaltung mit einem anderen Twitter-Nutzer (siehe Bild 15) kann erkannt werden, dass das automatische Antworten und Versenden von Nachrichten sehr erfolgreich ist. Wenn nun zusätzlich noch Kontexte und Entitäten mit verarbeitet werden, kann der Bot sich problemlos mit einem anderen Nutzer unterhalten, ohne dass dieser von seiner Existenz als Bot erfährt. Es muss noch dringend daran gearbeitet werden, dass der Bot sich bereits geführte Konversationen merkt und nicht jedes Mal auf dieselbe Frage reagiert. Zudem müssten noch Antwort- und Schlafzeiten integriert werden. Da dieser Agent nicht über die API angesprochen wird, sondern eine direkte Schnittstelle zum Twitter-Account hat, konnte dies nicht umgesetzt werden. Daher sollte in Zukunft die Nutzung der API im Vordergrund stehen. Somit wird gewährleistet, dass der Bot nur zwischen 8:00 und 22:00 Uhr funktioniert und die Antwortzeit vom Bot

zwischen 5 und 30 Minuten betragen kann. Mithilfe dessen kann eine menschliche Imitation mit hoher Wahrscheinlichkeit zum Erfolg führen.

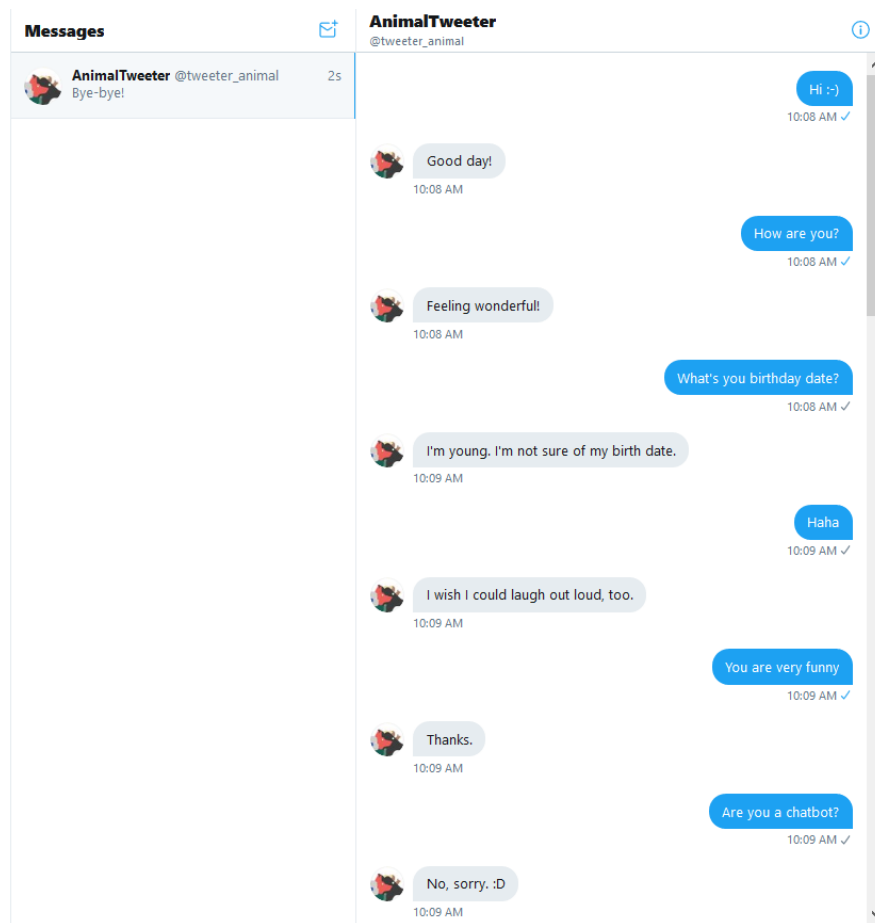


Bild 15: Unterhaltung mit einem anderen Twitter Nutzer

3.3.4 Unterhaltung in den Kommentaren

Der Bot Animal Tweeter kann neben den Direct Messages auch in den Kommentaren Unterhaltungen führen. Hierbei wird auch das Dialogflow Framework angesprochen, sobald der Nutzer Animal Tweeter mit einem „@“ Zeichen in einem Kommentar erwähnt und ein entsprechender Intent angesprochen wird. Zudem kann der Bot automatisch eigene Kommentare verfassen, wenn er Tweets anderer Nutzer in seiner Timeline retweetet. Diese Funktion wurde bereits in den vorherigen Kapiteln genauer beschrieben. Folgendes Beispiel soll aufzeigen, dass der Bot automatisch auf Kommentare reagiert, welche eine bestimmte Absicht haben (siehe Bild 17). Jedoch fällt auf, dass diese Funktion noch nicht einwandfrei funktioniert (siehe Bild 16). Der Bot postet manchmal

zwei Antworten hintereinander und beantwortet somit Fragen meist zweimal. Dieses Problem könnte in Zukunft umgangen werden, wenn mit der DialogFlow API gearbeitet wird und nicht mit der integrierten Twitter Schnittstelle auf der Webseite.



Bild 16: Fehlerhafte Unterhaltung – Zwei Antworten



Bild 17: Fehlerfreie Unterhaltung

3.4 Test der Erkennungssoftware „Botometer“

Für die Analyse des eigenen Twitter-Bots „Animal Tweeter“ wurde das Erkennungssystem „Botometer“ verwendet. Bei dieser für jeden zur Verfügung stehenden Webseite kann ein beliebiger Twitter-Account analysiert werden. Es muss lediglich der zu analysierende Twitter-Name eingegeben werden und ein Score wird ausgegeben. Dieser Score gibt an, ob es sich hierbei um ein Bot oder einen Menschen handelt. Zusätzlich dazu können auch die Follower überprüft werden.

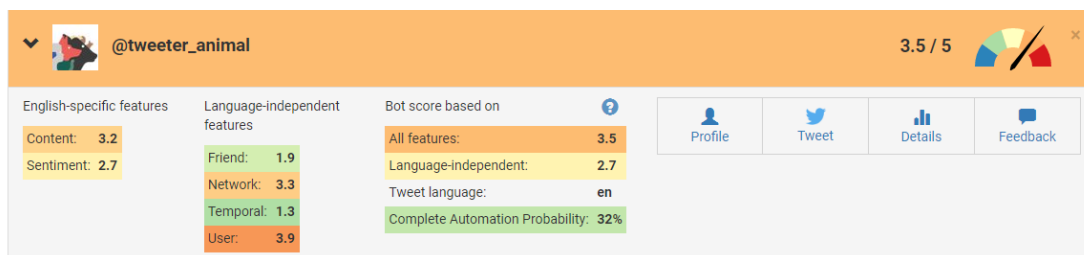


Bild 18: Bot-Score vom Twitter-Account „Animal Tweeter“, [OSoM19]

Der Twitter-Bot wird mit einer hohen Wahrscheinlichkeit von 3,5 Punkten vom Tool „Botometer“ als Social Bot erkannt. Hierbei muss jedoch beachtet werden, dass sich der Bot zu diesem Zeitpunkt noch in der Testphase befand. Es wurden mehrere Tweets pro Tag abgesetzt und an anderen Tagen nicht. Der Bot wurde noch nicht vollständig aktiviert und sollte zunächst nur als Prototyp laufen. Im Bild 19 ist erkennbar, dass unregelmäßig Beiträge veröffentlicht wurden. Besonders am Dienstag wurde eine Vielzahl an Tweets abgesetzt, da dieser bspw. ein Testtag war.

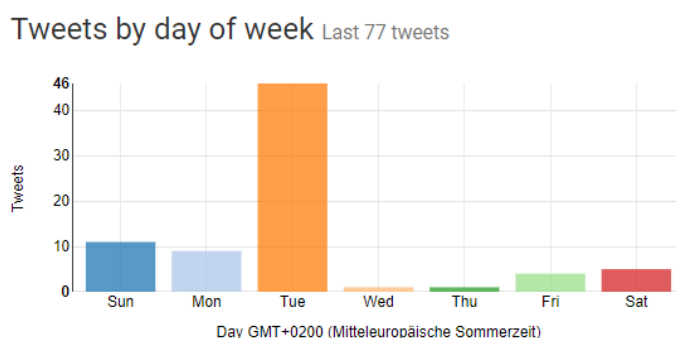


Bild 19: Tweets by day of week, [OSoM19]

Der Botometer schätzt die gesamte Automatisierung des Accounts lediglich auf 32% ein. Als Gegenbeispiel zu dem Twitter-Account „Animal Tweeter“ wird der offensichtliche Bot-Account @fediversebot herangezogen. Dieser steht bei einem Score von 4,9 und bei einer Automatisierung von 94 % (siehe Bild 20).

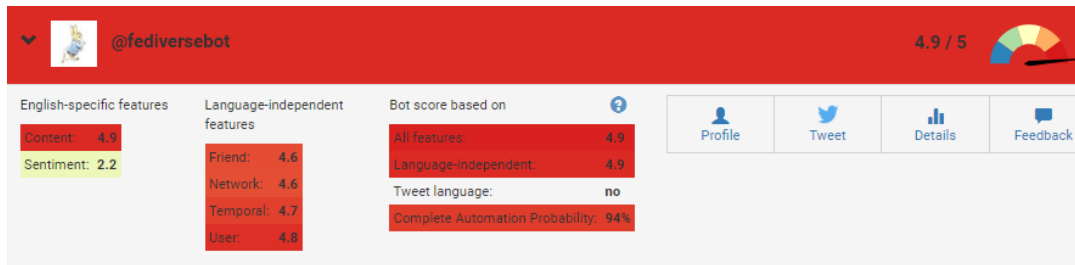


Bild 20: Bot-Score vom Twitter-Account „@fediversebot“, [OSoM19]

Dennoch wurde der Account „Animal Tweeter“ nicht zu 100% als Bot erkannt und es ist ersichtlich, dass erst eine lange Testphase zeigen wird, ob der Bot wirklich über solche Verfahren erkannt werden kann. Schließlich soll noch ein Beispiel zeigen, dass solche Systeme auch falsch klassifizieren können. Beispielsweise wird der Nutzer, welcher selbst von mir in diesem Zusammenhang erstellt und verwaltet wurde, ebenfalls als Bot mit einer hohen Wahrscheinlichkeit erkannt (siehe Bild 21). Dies kann durch unterschiedliche Merkmale, die vom Botometer gesetzt werden, hervorgerufen werden.

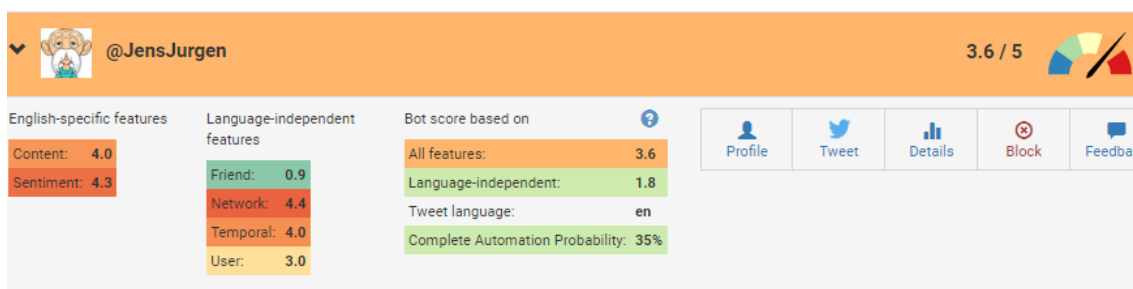


Bild 21: Bot-Score vom Twitter-Account „@JensJurgen“, [OSoM19]

Abschließend lässt sich sagen, dass keine valide Aussage darüber getroffen werden kann, ob das Erkennungssystem Botometer zu 100% Social Bots erkennen kann. Ersichtlich wird aber, dass die Tendenzen stimmen, jedoch auch Falscherkennungen vorliegen

können. Besonders im Zusammenhang mit der künstlichen Intelligenz sollten hier neue Schwellwerte und Merkmale gesetzt werden.

4 Ausblick in die Zukunft

Schlussendlich wurde in dieser Thesis durch den entwickelten Twitter-Bot deutlich, wie einfach und unkompliziert ein Social Bot erstellt werden kann. Es existieren eine Menge Tools und Frameworks für intelligente Funktionen, die über eine API Schnittstelle schnell angebunden werden können. Dennoch muss gesagt werden, dass nicht alle Frameworks einwandfrei funktionieren und nicht auf jeden Einsatzbereich anwendbar sind. In dieser Thesis musste auf einige wenige Technologien verzichtet und dafür andere herangezogen werden. Dadurch ist erkennbar, dass es nicht nur ein Framework oder Tool für eine bestimmte Funktion gibt, sondern mehrere. Weiterhin wurde offensichtlich, dass insbesondere maschinelle und Natural Language Processing Lernsysteme mit einer Menge von fehlerfreien Beispielen antrainiert werden müssen. Beispiele haben gezeigt, wie rasant sich ein selbstlernendes System mit falschen Eingabedaten zu einem Hass-Bot entwickeln kann.

Die selbstständige Kommunikation und automatische Generierung von Nachrichten sind bereits jetzt möglich ohne das menschliche Nutzer hinter diesen Aktionen einen Roboter vermuten. Es existieren Social Bots, welche mit Methoden der künstlichen Intelligenz entwickelt worden sind. Sie dienen in der Mehrheit dem guten Zweck. Bis diese jedoch für kriminelle Taten zweckentfremdet werden, brauch es nicht mehr lange. Der Einsatz von Social Bots stellt die Menschen vor moralische, politische und ethische Herausforderungen. Deutlich wurde, dass Potenzial in der Manipulation der Meinungsbildung besteht und Menschen meist der Mehrheit folgen und sich schnell beeinflussen lassen. Mit der Entwicklung des Social Bots wurde deutlich, wie einfach es ist, sich als Mensch in den sozialen Medien auszugeben und natürliche Unterhaltungen zu führen. Demnach ist es denkbar, dass somit das Vertrauen dieser Nutzer gewonnen werden und eine Täuschung oder Betrug stattfinden kann.

Ferner müssen die Erkennungsmethoden neu evaluiert und weiterentwickelt werden. Die existierenden Systeme konnten positiv getestet werden, dennoch wurde auch ein menschliches Profil als Bot klassifiziert. Die Schwellwerte für solche System müssen überdacht werden und eine zügige Weiterentwicklung dieser muss folgen.

Abschließend ist erkennbar, dass zukünftig mit der rasanten Entwicklung der künstlichen Intelligenz die Social Bots noch intelligenter werden. Des Weiteren ist es nötig, Fragen nach der Kennzeichnung und Regulierung von Bots, den Datenschutzrechten und ethischen Grundregeln zu klären. All diese Themen wurden bisher noch nicht diskutiert und reichlich erforscht. Dennoch müssen wir uns mit diesen Fragen auseinandersetzen, wahrscheinlich in Zukunft sogar gemeinsam mit Bots.

5 Literaturverzeichnis

- [10hoch19] 10hoch10 GmbH: Programmiersprache Ruby - Definition und Einsatzgebiete. <https://www.10hoch10.com/wiki/ruby/>, 2019, 20.09.2019.
- [ARZD19] ARD; ZDF: Anteil der Nutzer von Social-Media-Plattformen nach Altersgruppen in Deutschland im Jahr 2018. <https://de.statista.com/statistik/daten/studie/543605/umfrage/verteilung-der-nutzer-von-social-media-plattformen-nach-altersgruppen-in-deutschland/>, 2019, 14.05.2019.
- [Bähr15] Bähr, J.: Meinungsforschung im Netz: Wer stellt die Mehrheit? <https://www.faz.net/aktuell/feuilleton/medien/meinungsbildung-im-internet-und-in-sozialen-netzwerken-13692445.html>, 2015, 08.05.2019.
- [Beck19] Becker, R.: IBM Watson KI Services Übersicht. <https://jaai.de/ibm-watson/>, 2019, 20.09.2019.
- [Bek119] Beklemysheva, A.: Why Use Python for AI and Machine Learning? <https://steelkiwi.com/blog/python-for-ai-and-machine-learning/>, 2019, 20.09.2019.
- [Bend18] Bendel, P. D. O.: Soziale Medien. Definition. <https://wirtschaftslexikon.gabler.de/definition/soziale-medien-52673/version-275791>, 2018, 10.04.2019.
- [Bett19] Bettendorf, S.: Instagram-Journalismus. Ein Leitfaden für Redaktionen und freie Journalisten. Springer VS, Wiesbaden, Germany, 2019.
- [bots19] botswatch Technologies GmbH: botswatch Technologies – Cyber Intelligence at Rapid Scale. <https://botswatch.de/>, 2019, 13.07.2019.
- [Botw19a] Botwiki: Resources for Twitterbots | Botwiki. <https://botwiki.org/resources/twitterbots/>, 2019, 10.06.2019.

- [Botw19b] Botwiki: Hosting bots on Glitch | Botwiki.
<https://botwiki.org/resource/tutorial/hosting-bots-on-glitch/>, 2019,
10.06.2019.
- [boxc17] boxcryptor: Social Bots: Wie sie helfen oder Ärger machen können.
<https://www.boxcryptor.com/de/blog/post/social-bots-everything-you-need-to-know/>, 2017, 10.06.2019.
- [Bund18] Bundesamt für Informationstechnik: Social Bots und Chat Bots: Kleine Mini-Roboter mit eigener Identität im Netz. https://www.bsi-fuer-buerger.de/BSIFB/DE/DigitaleGesellschaft/SozialeNetze/Bots/bots_node.html, 2018, 20.09.2019.
- [BuSc19] Buxmann, P.; Schmidt, H. Hrsg.: Künstliche Intelligenz. Mit Algorithmen zum wirtschaftlichen Erfolg. Springer Berlin Heidelberg, Berlin, Heidelberg, 2019.
- [Comp19] Compton, K.: Tracery - Grammar. <http://brightspiral.com/tracery/>, 2019, 27.08.2019.
- [DA-117] DA-14 Corp.: Python vs Node.js: Which is Better for Your Project | DA-14. <https://da-14.com/blog/python-vs-nodejs-which-better-your-project/>, 2017, 20.09.2019.
- [Dial19] Dialogflow: Build natural and rich conversational experiences. <https://dialogflow.com/>, 2019, 20.09.2019.
- [Digi17] Digital Inspiration: Twitter Bots Not Working - Troubleshooting Guide. https://digitalinspiration.com/twitter-bots-troubleshoot-5018?_ga=2.155395323.1705152773.1566889339-2114973079.1566889339, 2017, 27.08.2019.
- [Digi19] Digital Inspiration: Twitter Bots v2. <https://script.google.com/macros/s/AKfycbwWSRgdQ7ji6TH1njlimjZiM81M7MdHA34BliELrANouhfIBXz/exec>, 2019, 27.08.2019.

- [Dude19] Dudenredaktion: Duden | Instagram | Rechtschreibung, Bedeutung, Definition, Herkunft. <https://www.duden.de/node/71461/revision/71497>, 2019, 02.05.2019.
- [EDUC18a] EDUCBA: Python vs Node.js - 5 Best Comparisons You Need To Know. <https://www.educba.com/python-vs-node-js/>, 2018, 20.09.2019.
- [EDUC18b] EDUCBA: Ruby vs Node - Know The Top 7 Most Useful Differences. <https://www.educba.com/ruby-vs-node/>, 2018, 20.09.2019.
- [Evs19] Evsan, I.: Social-Media. Was ist Social-Media? <https://www.gruenderszene.de/lexikon/begriffe/social-media?interstitial>, 2019, 10.04.2019.
- [Face19] Facebook: Anzahl der monatlich aktiven Facebook Nutzer weltweit vom 3. Quartal 2008 bis zum 1. Quartal 2019 (in Millionen). <https://de.statista.com/statistik/daten/studie/37545/umfrage/anzahl-der-aktiven-nutzer-von-facebook/>, 2019, 26.04.2019.
- [Feld16] Felden, C.: Künstliche Intelligenz — Enzyklopaedie der Wirtschaftsinformatik. <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/technologien-methoden/KI-und-Softcomputing/Kunstliche-Intelligenz/index.html>, 2016, 21.09.2019.
- [Garc19] Garcia, M.: How to Make a Twitter Bot in Python With Tweepy – Real Python. <https://realpython.com/twitter-bot-python-tweepy/#twitter-api>, 2019, 20.09.2019.
- [Gent18] Gentsch, P.: Künstliche Intelligenz für Sales, Marketing und Service. Mit AI und Bots zu einem Algorithmic Business - Konzepte, Technologien und Best Practices. Springer Gabler, Wiesbaden, 2018.
- [GrK118] Gräbe, P. D. H.-G.; Kleemann, K. P.: Social Media und Social Bots. Präsentation, Leipzig, 2018.

- [Hege16] Hegelich, S.: Invasion der Meinungs-Roboter. Konrad-Adenauer-Stiftung, Sankt Augustin, 2016.
- [Hein17] Heinrich-Böll-Stiftung: Social Bots. <https://www.boell.de/de/2017/02/09/social-bots>, 2017, 15.07.2019.
- [Hert17] Hertl, M.: Social Bots und Künstliche Intelligenz als Herausforderung an die Theologie, 2017.
- [Inst18] Institut für Technikfolgen-Abschätzung (ITA): Manipulation in Sozialen Medien, 2018.
- [Inte17] Intersog: Developing Solutions in NodeJS vs Python: Pros and Cons. <https://medium.com/@Intersog/developing-solutions-in-nodejs-vs-python-pros-and-cons-4ab4cea68ff0>, 2017, 20.09.2019.
- [IONO18] IONOS SE: Social Bots: Was können die Meinungsroboter wirklich? <https://www.ionos.de/digitalguide/online-marketing/social-media/social-bots-was-koennen-die-meinungsroboter-wirklich/>, 2018, 30.05.2019.
- [IONO19] IONOS SE: Wie funktioniert Natural Language Processing? <https://www.ionos.de/digitalguide/online-marketing/verkaufen-im-internet/wie-funktioniert-natural-language-processing/>, 2019, 31.07.2019.
- [Jane17] Janetzko, D.: Die neuen Medien und die politische Meinungsbildung. »Fake News« - ein Produkt der neuen Medien? In: *Deutschland & Europa*, 2017, 34.; S. 28–32.
- [Ki17] Kind, S. et al.: Social Bots. TA-Vorstudie. Büro für Technikfolgen-Abschätzung beim Deutsche Bundestag (TAB), Berlin, 2017.
- [Klei18] Kleiner Perkins Caufield & Byers: Börsenwert der größten Internetunternehmen weltweit im Mai 2018 (in Milliarden US-Dollar). <https://de.statista.com/statistik/daten/studie/217485/umfrage/marktwert-der-groessten-internet-firmen-weltweit/>, 2018, 26.04.2019.

- [KnRi10] Kneidinger, B.; Richter, R.: Facebook und Co. Eine soziologische Analyse von Interaktionsformen in Online Social Networks. Zugl.: Wien, Univ., Diplomarbeit, 2010. VS Verl. für Sozialwiss, Wiesbaden, 2010.
- [Koro17] Korotcenko, K.: Bad Bot, Super Bot. Einblick in die verschiedenen Bot-Arten. <https://traineesinsights.maisberger.de/bad-bot-super-bot-kleiner-einblick-in-die-verschiedenen-bot-arten.html>, 2017, 30.05.2019.
- [LuLi18a] Lubner, S.; Litzel, N.: Was ist TensorFlow? <https://www.bigdata-insider.de/was-ist-tensorflow-a-684272/>, 2018, 20.09.2019.
- [LuLi18b] Lubner, S.; Litzel, N.: Was ist ein Neuronales Netz? <https://www.bigdata-insider.de/was-ist-ein-neuronales-netz-a-686185/>, 2018, 20.09.2019.
- [LuLi19] Lubner, S.; Litzel, N.: Was ist Dialogflow? <https://www.bigdata-insider.de/was-ist-dialogflow-a-788455/>, 2019, 20.09.2019.
- [LYW16] Lerman, K.; Yan, X.; Wu, X.-Z.: The Majority Illusion in Social Networks. *In: PLOS ONE*, 2016, 11; 1-4.
- [Mich17] Michael Markert: Das Python-Tutorial. <https://py-tutorial-de.readthedocs.io/de/python-3.3/>, 2017, 20.09.2019.
- [Micr19a] Microsoft Azure: About Azure Bot Service. <https://docs.microsoft.com/en-us/azure/bot-service/bot-service-overview-introduction?view=azure-bot-service-4.0>, 2019, 20.09.2019.
- [Micr19b] Microsoft Azure: Language Understanding Intelligent Service (LUIS) | Microsoft Azure. <https://azure.microsoft.com/de-de/services/cognitive-services/language-understanding-intelligent-service/>, 2019, 20.09.2019.
- [Mitt18] Mittal, A.: Top NodeJS Libraries and Tools For Machine Learning. <https://medium.com/crowdbotics/top-nodejs-libraries-and-tools-for-machine-learning-ae0c106c9a69>, 2018, 20.09.2019.

- [Münc15] Münch, V.: Studie: Wie soziale Netzwerke unsere Meinung manipulieren. <https://www.wiwo.de/technologie/digitale-welt/studie-wie-soziale-netzwerke-unsere-meinung-manipulieren/12003650.html>, 2015, 08.05.2019.
- [Nguy18] Nguyen, H. T.: Einführung in TensorFlow: Einleitung und Inhalt. <https://data-science-blog.com/blog/2018/09/07/einfuehrung-in-tensorflow/>, 2018, 20.09.2019.
- [Nimm17] Nimmo, B.: Zwölf Arten, Bots zu erkennen. <https://correctiv.org/faktencheck/hintergrund/2017/09/13/zwolf-arten-bots-zu-erkennen>, 2017, 15.06.2019.
- [OSoM19] OSoMe: Botometer by OSoMe. <https://botometer.iuni.iu.edu/#/>, 2019, 23.09.2019.
- [Parr16] Parrish, A.: Tracery tutorial. <http://air.decontextualize.com/tracery/>, 2016, 21.09.2019.
- [Plic18b] Plich, N.: Das Unternehmen Botswatch spürt Fake Accounts im Netz auf. <https://www.business-punk.com/2018/03/botswatch-analyse-fake-news/>, 2018, 13.07.2019.
- [Pric17a] PricewaterhouseCoopers GmbH Wirtschaftsprüfungsgesellschaft (PwC): Social Bots: Gefahr für die Demokratie, 2017.
- [Pric17b] PricewaterhouseCoopers GmbH Wirtschaftsprüfungsgesellschaft (PwC): Social Bots – auch eine Gefahr für Unternehmen? <https://www.pwc.de/de/newsletter/it-security-news/social-bots-auch-eine-gefahr-fuer-unternehmen.html>, 2017, 19.07.2019.
- [Pric18a] PricewaterhouseCoopers GmbH Wirtschaftsprüfungsgesellschaft: Studie zum Vertrauen in Medien, 2018.
- [Ramí18] Ramírez, A. N.: Twitter Bot using Neural Networks, 2018.

- [Real18] RealSocialSEO: Successfully Apply to access Twitter APIs - RealSocialSEO. <https://realsocalseo.com/successfully-apply-to-access-twitter-apis/>, 2018, 27.08.2019.
- [Rese17] Research Lab 2017: Social Bots & AI – Research 2017. Social Bots – automatisierte Meinungsmache des 21. Jahrhunderts. <https://research.radical-openness.org/2017/category/social-bots-ai/>, 2017, 02.08.2019.
- [Reut19] Reuter, M.: Social Bots: Was nicht erkannt werden kann, sollte nicht reguliert werden. <https://netzpolitik.org/2019/social-bots-was-nicht-erkannt-werden-kann-sollte-nicht-reguliert-werden/>, 2019, 30.05.2019.
- [SaZy18] Sachs-Hombach, K.; Zywietz, B. Hrsg.: Fake News, Hashtags & Social Bots. Neue Methoden populistischer Propaganda. Springer VS, Wiesbaden, 2018.
- [Schm18] Schmidt, J.-H.: Social Media. Springer VS, Wiesbaden, 2018.
- [Schö17] Schönleben, D.: Welche Social Bots gibt es und wie funktionieren sie? <https://www.gq-magazin.de/auto-technik/article/welche-social-bots-gibt-es-und-wie-funktionieren-sie>, 2017, 02.06.2019.
- [Sick16] Sickert, T.: Künstliche Intelligenz: Vom Hipster-Mädchen zum Hitler-Bot. <https://www.spiegel.de/netzwelt/web/microsoft-twitter-bot-tay-vom-hipstermaedchen-zum-hitlerbot-a-1084038.html>, 2016, 02.08.2019.
- [Sick17] Sickert, T.: Bot oder kein Bot: So erkennen Sie Meinungsroboter. <https://www.spiegel.de/netzwelt/web/social-bots-entlarven-so-erkennen-sie-meinungsroboter-a-1129539.html>, 2017, 09.07.2019.
- [Sieb19] Sieber, A.: Dialogroboter. Wie Bots und künstliche Intelligenz Medien und Massenkommunikation verändern. Springer Fachmedien Wiesbaden, Wiesbaden, 2019.
- [SoFr16] Solfrank, P.; Freisberg, L.: Manipulation durch Social Bots: Wie Meinungsmache im Internet funktioniert.

- <https://www.br.de/nachricht/social-bot-erklaerstueck-100.html>, 2016, 20.09.2019.
- [Stat19a] Statista GmbH: Themenseite: Twitter. <https://de.statista.com/themen/99/twitter/>, 2019, 02.05.2019.
- [Stat19b] Statista-Umfrage: Social-Media-Werbung 2019 | Statista. <https://de.statista.com/statistik/studie/id/59678/dokument/social-media-werbung/>, 2019, 14.05.2019.
- [Stat19c] Statista GmbH: Themenseite: Instagram. Statistiken zu Instagram. <https://de.statista.com/themen/2506/instagram/>, 2019, 02.05.2019.
- [Sull17] Sulleyman, A.: Brexit: Pro-Leave Twitter bots played ‘strategic role’ in EU referendum result, says Oxford University study. undefined, 2017, 20.09.2019.
- [TeHa18] Ternès, A.; Hagemes, H.-P.: Die Digitalisierung Frisst Ihre User. Der Digitale Wahnsinn und Wie Sie Ihn Beherrschen. Springer, Wiesbaden, 2018.
- [Turi50] Turing, A. M.: Computing Machinery and Intelligence. In: *Mind – A Quarterly Review of Psychology and Philosophy*, 1950; S. 460.
- [Va17] Varol, O. et al.: Online Human-Bot Interactions: Detection, Estimation, and Characterization, 2017.
- [Vert19a] Vertical Media GmbH: Twitter Definition. <https://www.gruenderszene.de/lexikon/begriffe/twitter?interstitial>, 2019, 02.05.2019.
- [Vert19b] Vertical Media GmbH: Social Network. Was ist ein Social Network? <https://www.gruenderszene.de/lexikon/begriffe/social-network?interstitial>, 2019, 26.04.2019.

- [Vett15] Vettters, A.: Hashtag - Definition. <http://www.digitalwiki.de/hashtag/>, 2015, 02.05.2019.
- [Vogt17] Vogt, M.: Natural Language Processing: Scheitert KI an natürlicher Sprache? <https://www.ccw.eu/blog/natural-language-processing/>, 2017, 31.07.2019.
- [WaFl18] Wahl, J.; Fleck, H.: Was das Microsoft Bot Framework heute bereits kann. <https://www.computerweekly.com/de/meinung/Was-das-Microsoft-Bot-Framework-heute-bereits-kann>, 2018, 20.09.2019.
- [WHD19a] We Are Social; Hootsuite; DataReportal: Ranking der größten sozialen Netzwerke und Messenger nach der Anzahl der monatlich aktiven Nutzer (MAU) im Januar 2019 (in Millionen). <https://de.statista.com/statistik/daten/studie/181086/umfrage/die-weltweit-groessten-social-networks-nach-anzahl-der-user/>, 2019, 02.05.2019.
- [WHD19b] We Are Social; Hootsuite; DataReportal: Anzahl der Facebook-Nutzer nach Altersgruppen und Geschlecht weltweit im Januar 2019 (in Millionen). <https://de.statista.com/statistik/daten/studie/665793/umfrage/facebook-nutzer-nach-alter-und-geschlecht-weltweit/>, 2019, 26.04.2019.
- [Witt19] Wittpahl, V. Hrsg.: Künstliche Intelligenz. Technologie. Springer Berlin Heidelberg, Berlin, Heidelberg, 2019.
- [Zeif17] Zeifman, I.: Bot Traffic Report 2016. <https://www.imperva.com/blog/bot-traffic-report-2016/>, 2017, 20.09.2019.

6 Abkürzungsverzeichnis

API	Application Programming Interface
BKA	Bundeskriminalamt
IBM	International Business Machine
KI	künstliche Intelligenz
NLP	Natural Language Processing
SDK	Software Development Kit

7 Bildverzeichnis

Bild 1: Ranking der größten Social Networks und Messenger [WHD19a]	15
Bild 2: Die Mehrheitsillusion [LYW16]	18
Bild 3: Cartoon von Peter Steiner aus dem Jahre 1993 [Jane17]	22
Bild 4: Umfrage 2018 zu „Welche Informationskanäle nutzen Sie?“ [Pric18a] ..	25
Bild 5: Aufbau eines Social Bots, eigene Darstellung.....	29
Bild 6: Twitter Bots V2 - Example [Digi19]	46
Bild 7: Tracery Grammar [Comp19]	47
Bild 8: Konzept des Bots „Animal Tweeter“, eigene Darstellung (Größere Grafik s. Anhang).....	58
Bild 9: Automatisch verfasste Tweets	62
Bild 10: Beispielsätze für das Testen des Watson Tone Analyzer Services.	64
Bild 11: Ergebnisse der Beispielsätze	64
Bild 12: Retweet Beispiel	65
Bild 13: Favorisieren eines Replies von einem Nutzer	66
Bild 14: Einstellungen für ein Intent	68
Bild 15: Unterhaltung mit einem anderen Twitter Nutzer.....	70
Bild 16: Fehlerhafte Unterhaltung – Zwei Antworten.....	71
Bild 17: Fehlerfreie Unterhaltung.....	71
Bild 18: Bot-Score vom Twitter-Account „Animal Tweeter“, [OSoM19]	72
Bild 19: Tweets by day of week, [OSoM19]	72
Bild 20: Bot-Score vom Twitter-Account „@fediversebot“, [OSoM19]	73
Bild 21: Bot-Score vom Twitter-Account „@JensJurgen“, [OSoM19]	73

8 Tabellenverzeichnis

Tabelle 1: Anforderungen an eine Programmiersprache bzw. -umgebung	49
Tabelle 2: Vergleich von Python, Node.js und Ruby	52

9 Anlagenverzeichnis

9.1 Python Quellcodes

credentials.py

```
1 #####
2 ## This project is a smart Twitter bot, which can independently retweet,
3 ## comment and like. In addition to this Python code, Dialogflow also exists
4 ## for communicating with other users and Tracery for the automated posting
5 ## of tweets.
6 #####
7 ## Author: Virginia Dei
8 ## Copyright: Copyright 2019, Twitterbot with artificial intelligent methods
9 ## Python Version: 2.7
10 ## Version: 2.0
11 ## Email: gini.elgner@googlemail.com
12 ## Date created: 08/01/2019
13 ## Date last modified: 09/21/2019
14 #####
15
16
17 consumer_key = 'upKY8kVI3jzyi6L3FD00Wq0Md'
18 consumer_secret = 'bWmdsCR105XV7SAtMZDSPnFEh1jz5r1ZgpuMKv2VzwXI6NvoBM'
19 access_token = '1157925366511624194-5bMw0J5ENiKmiKe0i15kx1XaiBLfuv'
20 access_token_secret = 'xyLaosV1aDSxRvCB1qpGMoyAkFKMLvFLsgCQaK3djgChF'
```

Watson_toneanalyzer.py

```

1 import json
2 from ibm_watson import ToneAnalyzerV3
3
4 #####
5 ## This project is a smart Twitter bot, which can independently retweet,
6 ## comment and like. In addition to this Python code, Dialogflow also exists
7 ## for communicating with other users and Tracery for the automated posting
8 ## of tweets.
9 #####
10 ## Author: Virginia Dei
11 ## Copyright: Copyright 2019, Twitterbot with artificial intelligent methods
12 ## Python Version: 2.7
13 ## Version: 2.0
14 ## Email: gini.elgner@googlemail.com
15 ## Date created: 08/01/2019
16 ## Date last modified: 09/21/2019
17 #####
18
19 tone_analyzer = ToneAnalyzerV3(
20     version='2017-09-21',
21     iam_apikey='8GmHS_GD25p9d0nd8zi33a8RtYG2hht_T8ZmXhbY7CNf',
22     url='https://gateway-fra.watsonplatform.net/tone-analyzer/api'
23 )
24
25 text = 'Yes, you are my friend. And I love animals!'
26
27
28 def analyzetone(text):
29     tone_analysis = tone_analyzer.tone(
30         {'text':text},
31         content_type='application/json'
32     ).get_result()
33     jsonText = json.dumps(tone_analysis, indent=2)
34     jsonParse = json.loads(jsonText)
35     score = 0
36     tonename = ''
37     for w in jsonParse['document_tone']['tones']:
38         if w['score'] > score:
39             score = w['score']
40             tonename = w['tone_name']
41     return tonename
42
43 #print jsonParse
44 print(analyzetone(text))

```

Twitterbot_tweetwithtracery.py

```

1  #!/usr/bin/env python -t
2  # -*- coding: utf-8 -*-
3
4  #####
5  ## This project is a smart Twitter bot, which can independently retweet,
6  ## comment and like. In addition to this Python code, Dialogflow also exists
7  ## for communicating with other users and Tracery for the automated posting
8  ## of tweets.
9  #####
10 ## Author: Virginia Dei
11 ## Copyright: Copyright 2019, Twitterbot with artificial intelligent methods
12 ## Python Version: 2.7
13 ## Version: 2.0
14 ## Email: gini.elgner@googlemail.com
15 ## Date created: 08/01/2019
16 ## Date last modified: 09/21/2019
17 #####
18
19 import tweepy
20 import time
21 from datetime import datetime
22 from credentials import *
23 import tracery
24 from tracery.modifiers import base_english
25 import json
26 from random import randint
27 import random
28
29 # Twitter Access and Consumer Tokens from the file credentials.py
30 auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
31 auth.set_access_token(access_token, access_token_secret)
32 api = tweepy.API(auth)
33
34 # current date format
35 dateSTR = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
36
37 # End Time Bot, he works between 8 am and 10 pm (german time)
38 endtime = datetime.now().strftime("%Y-%m-%d" + " 20:00:00")
39
40 # i, variable to count the retweets to stop at a certain number so that the bot
41 # itself is not detected by the number of tweets
42 i = 0
43
44 # use json.loads() and open() to read in a JSON file as a Python data structure
45 rules = json.loads(open("tweet-grammar.json").read())
46

```

```
47 # Composition of the strings from the JSON file, load the tracery grammar
48 grammar = tracery.Grammar(rules)
49 grammar.add_modifiers(base_english)
50
51 # Random Number Sleep (Between 25 and 90 minutes), then only then will the first
52 # tweet be published
53 zufallSleep = randint(1500,5400)
54 print zufallSleep
55 print ("a tweet is created")
56 time.sleep(zufallSleep)
57
58 # Main loop for the bot, he is in the loop from 8am till 10pm and posts
59 # irregular a maximum of 3 tweets a day
60 while i < 3:
61     # grammar is created with the beginning of origin in the JSON File
62     tweettext = grammar.flatten("#origin#")
63
64     print tweettext
65
66     # tweet is created and published
67     api.update_status(tweettext)
68
69     # Variable i for counting tweets
70     i = i + 1
71
72     # Random Number Sleep (Between 4 and 10 hours), how long the break is until
73     # the loop begins again
74     zufallSleep = randint(14400,36000)
75     print zufallSleep
76     print ("a tweet is created")
77     time.sleep(zufallSleep)
78
79     # Current time is queried
80     dateSTR = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
81     print dateSTR
82
83     # If the current time matches the end time, the loop is aborted and the
84     # social bot goes to sleep
85     if (dateSTR == endtime or dateSTR > endtime):
86         print ("The End. The Time!")
87         break
```

Twitterbot_retweetwithanalyzer.py

```

1  #!/usr/bin/env python -t
2  # -*- coding: utf-8 -*-
3
4  #####
5  ## This project is a smart Twitter bot, which can independently retweet,
6  ## comment and like. In addition to this Python code, Dialogflow also exists
7  ## for communicating with other users and Tracery for the automated posting
8  ## of tweets.
9  #####
10 ## Author: Virginia Dei
11 ## Copyright: Copyright 2019, Twitterbot with artificial intelligent methods
12 ## Python Version: 2.7
13 ## Version: 2.0
14 ## Email: gini.elgner@googlemail.com
15 ## Date created: 08/01/2019
16 ## Date last modified: 09/21/2019
17 #####
18
19
20 import tweepy
21 import time
22 from credentials import *
23 from watson_toneanalyzer import *
24 from datetime import datetime
25 from random import randint
26 import random
27
28 # Twitter Access and Consumer Tokens from the file Credentials.py
29 auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
30 auth.set_access_token(access_token, access_token_secret)
31 api = tweepy.API(auth)
32
33 # Differnet variables
34 # current date format
35 dateSTR = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
36
37 # End Time Bot, he works between 8 am and 10 pm (german time)
38 endtime = datetime.now().strftime("%Y-%m-%d" + " 20:00:00")
39
40 # i, variable to count the retweets to stop at a certain number so that the bot
41 # itself is not detected by the number of tweets
42 i = 0
43
44 # Which tweets should be searched for and then retweet them itsself
45 search_terms = ('#animals OR #dogs OR #cats OR #animalwelfare OR #saveanimals')
46

```

```

47 # List for possible comments on happy and friendly tweets
48 happy_list = ['Great Post', ' I like :-)', 'Yay' , 'Nice', 'Great Post' ,
49 'Yees 🤗', '🤗', 'Good Post', 'I like your Post', 'I like it!']
50
51 # List for possible comments on sad tweets
52 sadness_list = ['Oh :-( ', ' :-( ', 'Its very sad..' , 'Sad..']
53
54 # own username
55 user_name = "@tweeter_animal"
56
57 # Function for liking replies to the own account of the last 7 days
58 def like_tweet():
59     for tweet in tweepy.Cursor(api.search, q='to:{} filter:replies'.format(user_name),
60 tweet_mode='extended').items():
61         if not tweet.favorited:
62             # Mark it as Liked, since we have not done it yet
63             try:
64                 tweet.favorite()
65             except Exception as e:
66                 print ("No, it has already a like!")
67
68 # Main loop for the bot, he is in the loop from 8am till 10pm and posts
69 # irregular tweets from other users
70 for tweet in tweepy.Cursor(api.search, q=search_terms, since='2019-08-01',
71 result_type='recent', lang="en").items():
72     try:
73         # Replies are marked with a like
74         like_tweet()
75
76         # Variable i for counting retweets
77         i = i + 1
78
79         # two random parameter (yes = 1, no = 0) and follow the user (yes, no)
80         MaybeFriend = randint(0,1)
81         MaybeComment = randint(0,1)
82
83         # The tone of the tweet is analyzed
84         tone = analyzetone(tweet.text)
85
86         print ("Count of Tweets")
87         print i
88         print tweet.text
89         print tone
90         print ("Friends?")
91         print MaybeFriend
92         print ("Comments?")
93         print MaybeComment
94

```

```

95         # TweetID of the tweet to comment on the tweet afterwards
96         tweetID = tweet.id_str
97
98         # Query whether it is a positive tweet, then retweeted
99         if (tone == 'Joy' or tone == 'Analytical' or tone == 'Confident'):
100             print ("Yes, Retweet and Like")
101             tweet.retweet()
102             tweet.favorite()
103             # If the random number is yes to follow, then the user is
104             # followed, if this has not already happened
105             if MaybeFriend == 1:
106                 if not tweet.user.following:
107                     tweet.user.follow()
108                     print("Followed the user")
109             # If the random number indicates yes to comment, then the tweet
110             # is commented with a string from the Happy list
111             if MaybeComment == 1:
112                 print ("Yes, Comment")
113                 happystring = random.choice(happy_list)
114                 api.update_status(status = happystring,
115                                 in_reply_to_status_id = tweetID, auto_populate_reply_metadata=True)
116
117         # Query if it is a sad tweet, then retweeted but not like
118         if tone == 'Sadness':
119             print ("It is sad, retweet and comment")
120             tweet.retweet()
121             # If the random number is yes to comment, then the tweet will be
122             # commented with a string from the sadness list
123             if MaybeComment == 1:
124                 print ("Yes, Comment")
125                 sadstring = random.choice(sadness_list)
126                 api.update_status(status = sadstring,
127                                 in_reply_to_status_id = tweetID, auto_populate_reply_metadata=True)
128
129         # Query if it's a negative tweet, then nothing is done
130         if (tone == 'Anger' or tone == 'Fear' or tone == 'Tentative'):
131             print ("No not a Retweet")
132         if (tone == ''):
133             print ("No tone, no retweet")
134
135         # Random Number Sleep (Between 25 and 90 minutes), how long the
136         # break is until the loop begins again
137         zufallSleep = randint (1500,5400)
138         print zufallSleep
139         time.sleep(zufallSleep)
140
141         # Current time is queried
142         dateSTR = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
143         print dateSTR
144
145         # If the current time matches the end time, the loop is aborted
146         # and the social bot goes to sleep
147         if (dateSTR == endtime or dateSTR > endtime):
148             print ("The End. The Time!")
149             break
150
151         # When the maximum number of 15 tweets is reached, the loop is left
152         # and the bot goes to sleep.
153         if i == 5:
154             print ("The End. The Count!")
155             break
156     except tweepy.TweepError as e:
157         print(e.reason)
158     except StopIteration:
159         break

```

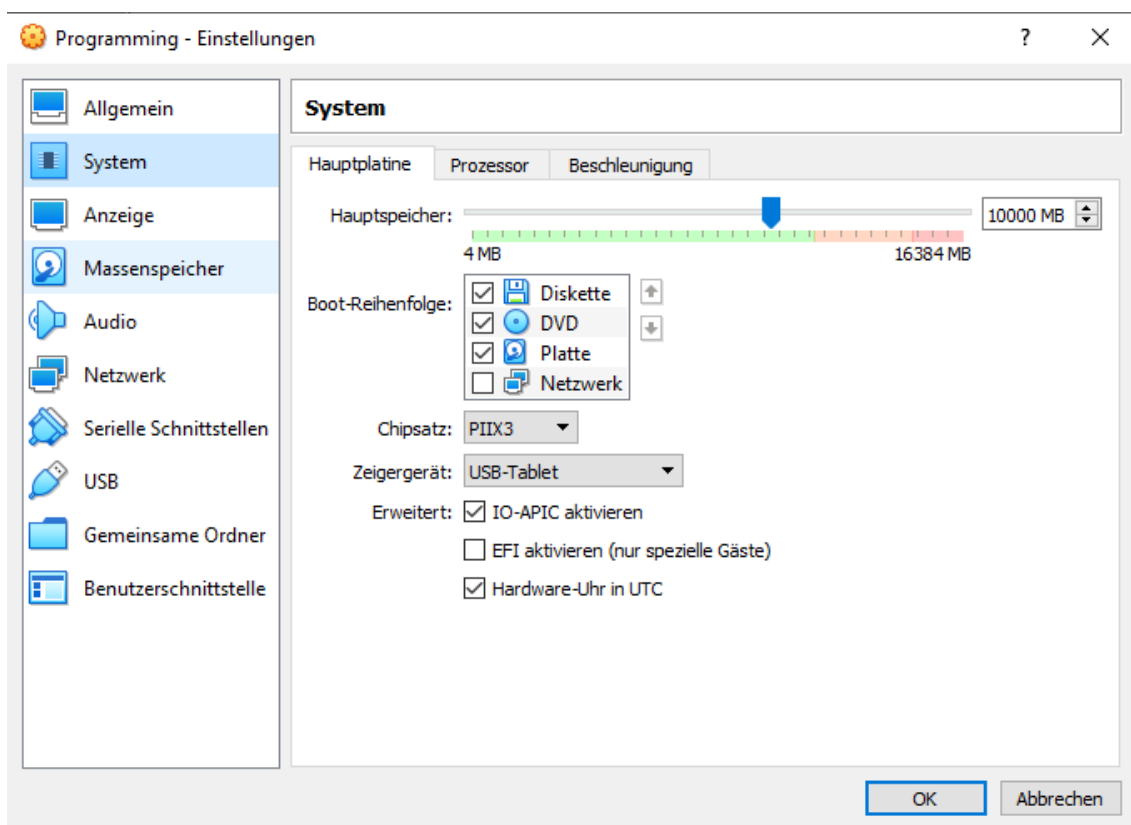
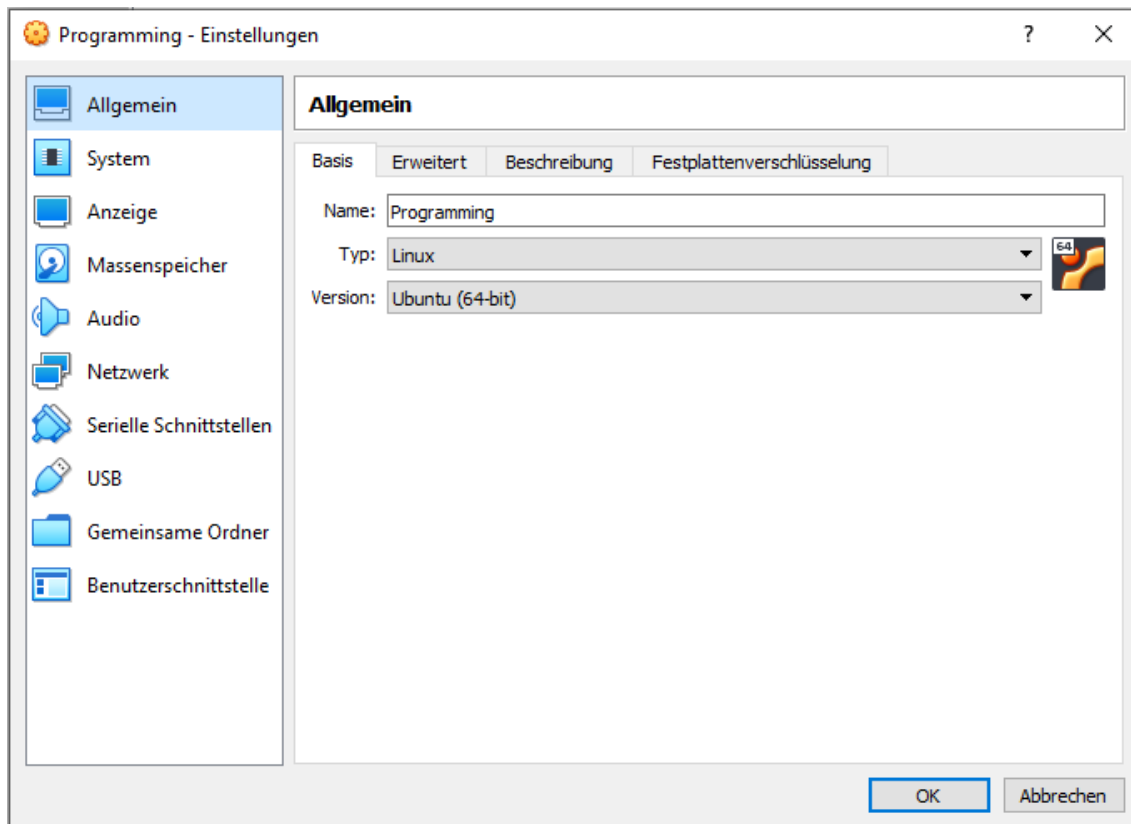
9.2 JSON Grammatik Datei

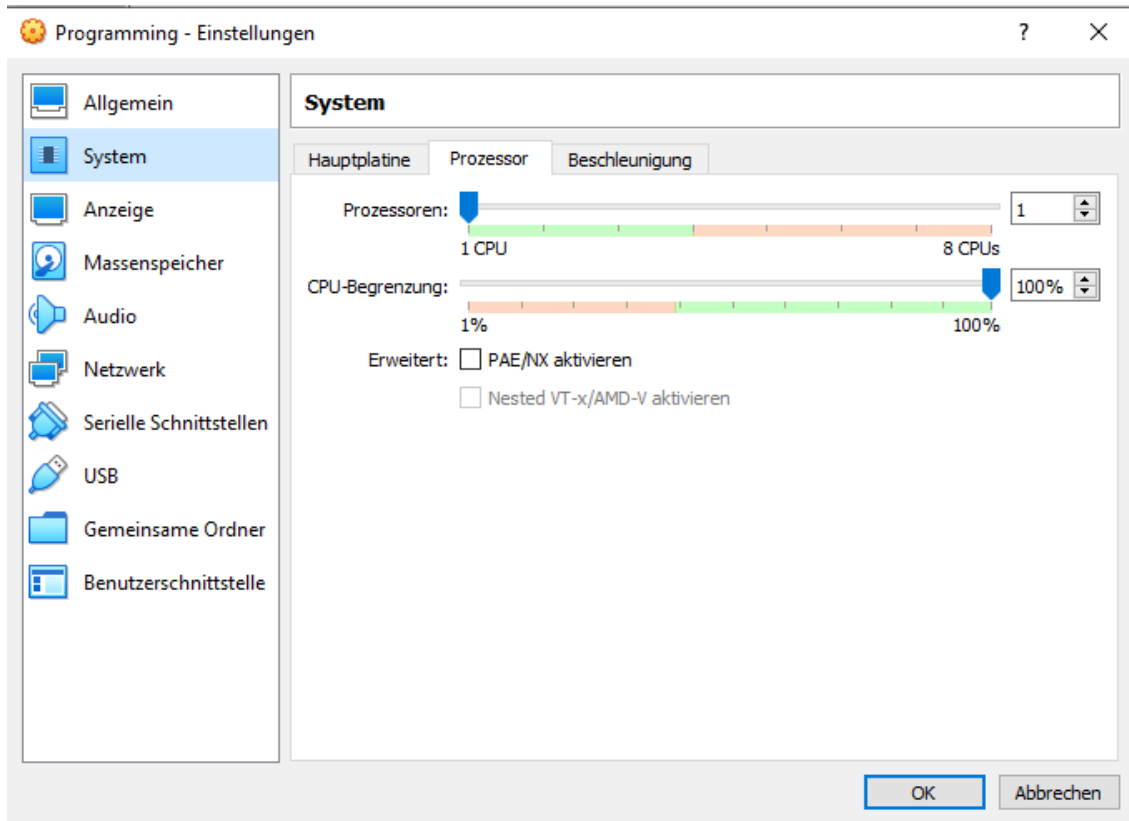
```

1 {
2   "greeting": [
3     "Hi",
4     "Hello",
5     "Whoo",
6     "Hey",
7     "Welcome"
8   ],
9   "origin": [
10    "#[Animal1:#Animal#][Animal2:#Animal2#]story#"
11  ],
12  "story": [
13    "#greeting# #greetingwords#, today is the #Animal1# Day! We #text#! \\##Animal1.s# #hashtags#",
14    "#greeting# #greetingwords#, #how#",
15    "#greeting#, do you have a #nice# #alternative# #hashtags#",
16    "Today is #good.a# day. What do you do?",
17    " I #text#!! \\ #hashtags# #hashtags#"
18  ],
19  "greetingwords": [
20    "Everyone",
21    "Guys",
22    "World",
23    "all",
24    "everyone"
25  ],
26  "alternative": [
27    "#Animal1# photo or video? \\##Animal1.s#",
28    "photo of your pet? \\#petphoto"
29  ],
30  "Animal": [
31    "Cat",
32    "Dog",
33    "Bird",
34    "Animal",
35    "Whale",
36    "Dolphin",
37    "Rabbit",
38    "Wolf",
39    "Elephant",
40    "Lion",
41    "Fish",
42    "Snake",
43    "Shark",
44    "Chicken",
45    "Deer",
46    "Fox",
47    "Bear",
48    "Kangaroo",
49    "Turtle"
50  ],
51  "text": [
52    "hope that all #Animal1.s# are doing well",
53    "hope you are fine",
54    "think of all #Animal1.s#",
55    "love you #Animal1.s#"
56  ],
57  "how": [
58    "how are you?",
59    "What's Up?",
60    "today is a #good day..",
61    "we have to support Animal Welfare \\#animalwelfare \\##Animal.s#"
62  ],
63  "hashtags": [
64    "\\#animalwelfare",
65    "\\#animals",
66    "\\#worldofanimal"
67  ],
68  "nice": [
69    "nice",
70    "beautiful",
71    "kindly",
72    "cute",
73    "sweet",
74    "crazy"
75  ],
76  "good": [
77    "good",
78    "beautiful",
79    "bad",
80    "boring",
81    "crazy"
82  ]
83 }

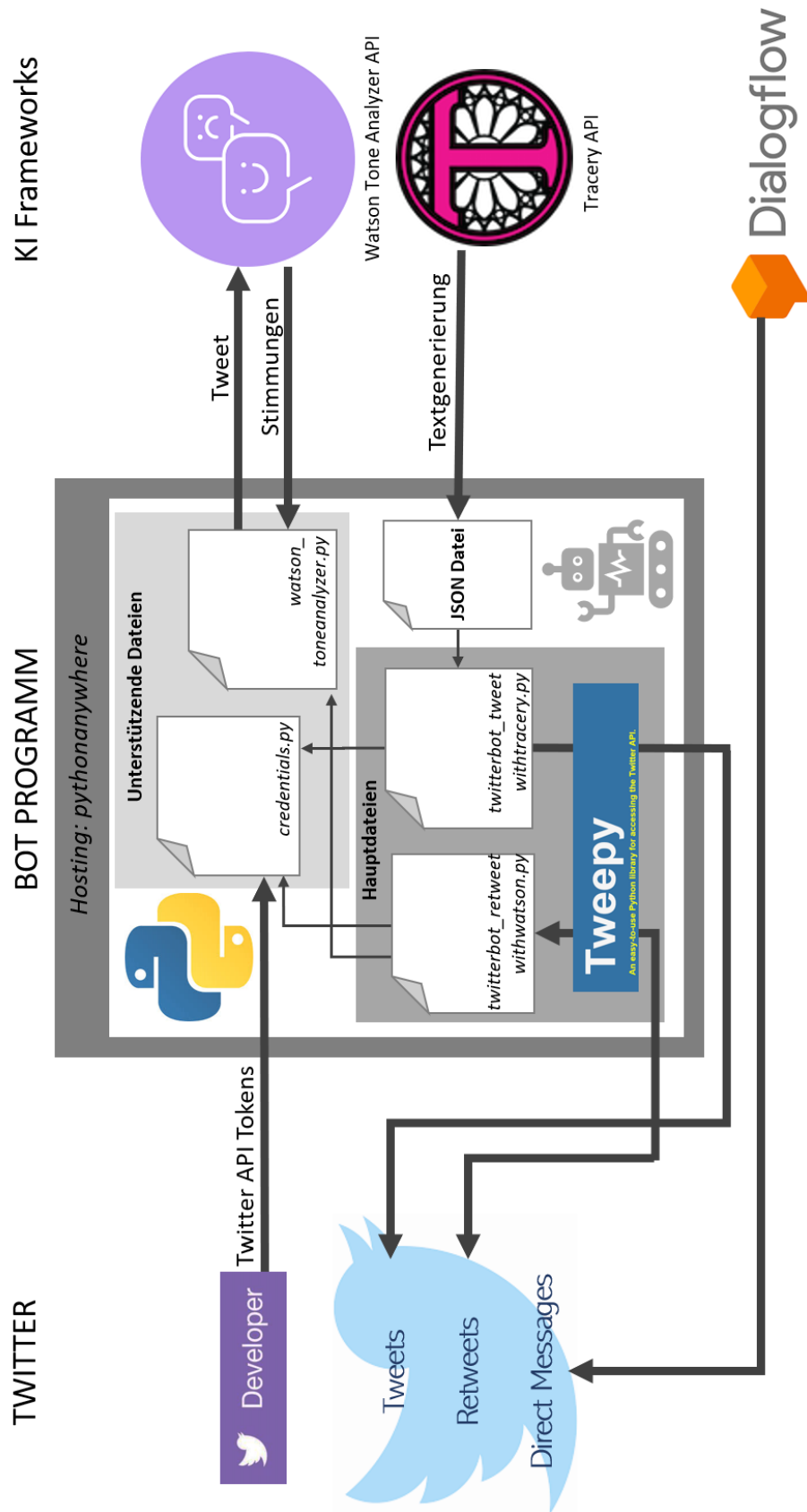
```


9.3 Daten der virtuellen Maschine für TensorFlow





9.4 Konzeptbild

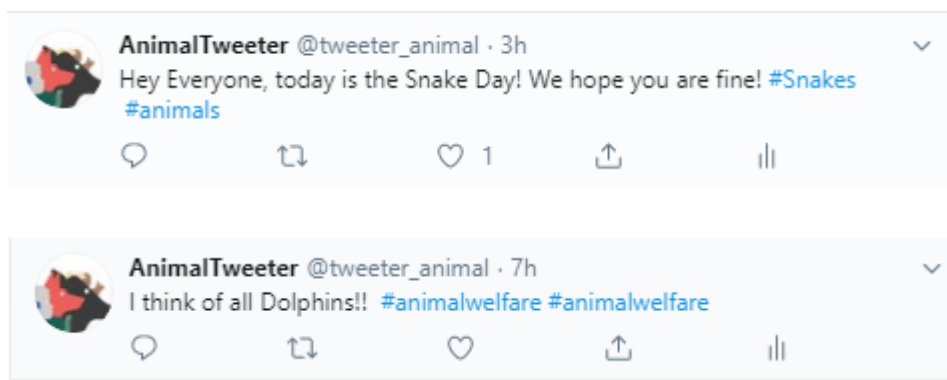


9.5 Testszenarien

9.5.1 Tweet - Testszenario 1

Die Datei wurde um ungefähr 09:00 Uhr gestartet. Um 09:51 Uhr kam dann der erste Tweet, 19:29 Uhr dann der zweite.

Sleep Erster Tweet	1	3081
	2	a tweet is created
	3	I think of all Dolphins!! #animalwelfare
Sleep auf knapp 9 Stunden	4	#animalwelfare
	5	34817
	6	a tweet is created
Zweiter Tweet	7	2019-09-26 19:29:51
	8	Hey Everyone, today is the Snake Day! We hope
Sleep nochmal auf 9 Stunden	9	you are fine! #Snakes #animals
	10	34775
	11	a tweet is created
Endzeit um 05:10 Uhr am nächsten Tag erreicht und dann beendet	12	2019-09-27 05:10:03
	13	The End. The Time!
	14	>>>



9.5.2 Tweet - Testszenario 2

9.5.3 Retweet - Testszenario 1

Die Datei *Twitterbot_retweetwithanalyzer.py* wurde um 16:20 Uhr gestartet. Die Zeit des Servers auf der Hosting Plattform befindet sich immer zwei Stunden vor unserer deutschen Zeit. Daher befand sich zu diesem Punkt der Server bei 14:20 Uhr. Die Datei wurde ausgeführt und sollte sich entweder um 22 Uhr oder bei einer Retweet Anzahl von 5 automatisch schließen. Für die Sleep Funktion wurden zwischen 1500 und 5400 Sekunden gewählt. In den nachfolgenden Screenshots der Konsole und des Twitter-Accounts kann erkannt werden, dass die Funktionen einwandfrei funktionieren. Aufgrund der maximalen Anzahl von Tweets wurde die Datei nach 5 Tweets wieder beendet.

Anzahl der Tweets wurde auf 1 gesetzt	1 2	Count of Tweets 1
Tweet	3 4 5 6	RT @jennireRescue: Good news ♥ 🐾 Today we #rescue this little #puppy, he was eating garbage and drinking dirty water, 🤢🚫 is anyone encourag...
Positiver Tweet, also wird dieser retweetet und anhand der Zufallsvariablen auch kommentiert	7 8 9 10 11 12 13	Joy Friends? 0 Comments? 1 Yes, Reweet and Like Yes, Comment
Sleep-Wert auf knapp 40 Minuten	14 15	2438 2019-09-26 15:08:31

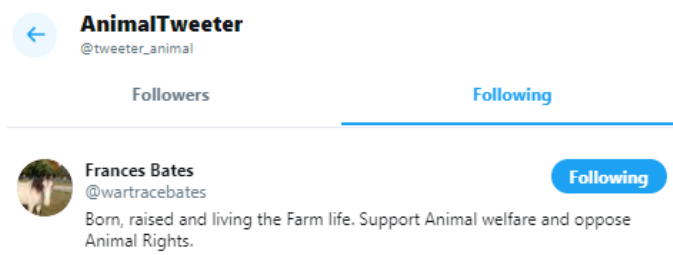
Bild a: Erster Tweet



Anzahl der Tweets wurde auf 2 gesetzt	16 17	Count of Tweets 2
Tweet	18 19 20 21	RT @ProtectHarvest: Animal Welfare includes all #animals, whether used for food, companionship, or sport. Based on a principle of ownership...
Positiver Tweet, also wird dieser retweetet und anhand der Zufallsvariablen auch kommentiert und dem Nutzer wird gefolgt	22 23 24 25 26 27 28 29	Analytical Friends? 1 Comments? 1 Yes, Reweet and Like Followed the user Yes, Comment
Sleep-Wert auf knapp 60 Minuten	30 31	3679 2019-09-26 16:09:54

Bild b: Zweiter Tweet





Dem letzten Retweeter wird dann gefolgt.

Anzahl der Tweets wurde auf 3 gesetzt	32 33	Count of Tweets 3
Tweet	34 35 36 37	3 more days till my kickstarter ends: https://t.co/7adsG4MD53 #animals #pin #pins #raccoon #crow #cat #enamelpins... https://t.co/WTcXKl0vHp
Keine Stimmung, also kein Retweet	38 39 40 41 42 43	Friends? 1 Comments? 0 No tone, no retweet
Sleep-Wert auf knapp 81 Minuten	44 45	4902 2019-09-26 17:31:39

Bild c: Dritter Tweet

Anzahl der Tweets wurde auf 4 gesetzt	46 47	Count of Tweets 4
Tweet	48 49 50 51 52 53	Uh-oh. Now the cat's out of the ba... oh never mind. - - #cat #catsofinstagram #cats #bag #funnycats #catstagram... https://t.co/QSgNCstfUy
Positiver Tweet, also wird dieser retweetet und anhand der Zufallsvariablen auch kommentiert	54 55 56 57 58 59 60	Confident Friends? 0 Comments? 1 Yes, Reweet and Like Yes, Comment
Sleep-Wert auf knapp 62 Minuten	61 62	3776 2019-09-26 18:34:38

Bild d: Vierter Tweet



Anzahl der Tweets wurde auf 5 gesetzt	63 64	Count of Tweets 5
Tweet	65 66 67 68	RT @bjkingape: "There's enough similarity between how #dogs form strong emotional bonds and how people form strong emotional bonds that it'...
Positiver Tweet, also wird dieser retweetet und anhand der Zufallsvariablen wurde dem Nutzer gefolgt	69 70 71 72 73 74 75	Joy Friends? 1 Comments? 0 Yes, Reweet and Like Followed the user
Sleep-Wert auf knapp 80 Minuten Bot wurde aufgrund der Anzahl der Tweets beendet	76 77 78 79	4734 2019-09-26 19:52:59 The End. The Count! >>>

Bild e: Fünfter und letzter Tweet

You Retweeted

 **Barbara J King**
@bjkingape

"There's enough similarity between how [#dogs](#) form strong emotional bonds and how people form strong emotional bonds that it's fair enough to use the love word." Absolutely! [@karinbrulliard](#) quoting [@caninecognition](#) [#animals](#)



What makes dogs so special and successful? Love. After searching for dogs' intelligence, a top canine scientist concluded that their true gift is affection.
[washingtonpost.com](#)

4:17 PM · Sep 26, 2019 · [Twitter Web App](#)

9 Retweets 36 Likes

 **AnimalTweeter**
@tweeter_animal

Followers Following

 **Agustin Fuentes**
@Anthrofuentes
Professor of Anthropology [Following](#)

9.5.4 Retweet - Testszenario 2

Die Datei *Twitterbot_retweetwithanaylzer.py* wurde um 10:15 Uhr gestartet. Die Datei sollte sich entweder um 11:15 Uhr oder bei einer Retweet Anzahl von 5 automatisch schließen. Für die Sleep Funktion wurden zwischen 300 und 1500 Sekunden gewählt. In den nachfolgenden Screenshots der Konsole und des Twitter-Accounts kann erkannt werden, dass die Funktionen einwandfrei funktionieren. Aufgrund der Zeit wurde die Datei beendet.

Anzahl der Tweets wurde auf 1 gesetzt	1 2	Count of Tweets 1
Tweet	3 4 5	I 100% believe in this, my dogs keep me sane and I am ridiculously in love with them 🐶🐶🐶 #animals #truelove https://t.co/rcVLMGjBtZ
Positiver Tweet, also wird dieser retweetet und anhand der Zufallsvariablen auch kommentiert und dem Nutzer wird gefolgt	6 7 8 9 10 11 12 13	Joy Friends? 1 Comments? 1 Yes, Reweet and Like Followed the user Yes, Comment
Sleep-Wert auf knapp 13 Minuten	14 15	803 2019-09-26 10:38:35

Bild A: Erster Tweet

 You Retweeted

 **Tracey Ward**
@traceywLS

I 100% believe in this, my dogs keep me sane and I am ridiculously in love with them 🥰🐶🐾
#animals #truelove

 **Pets As Therapy** @PetsAsTherapyUK · 18h
Research shows that spending time with #animals is an effective way of managing #stress and #anxiety petsastherapy.org/research/how-t...


12:24 PM · Sep 26, 2019 · [Twitter Web App](#)

1 Retweet 1 Like

 **AnimalTweeter** @tweeter_animal · 2h
Replying to @traceywLS
I like :-)

 **AnimalTweeter**
@tweeter_animal

Followers **Following**

 **Tracey Ward**
@traceywLS **Following**

Hospital social worker (neuro rehab) with a special interest in raising awareness around financial abuse through scams/fraud (oh and dogs and camping) Own views

Anzahl der Tweets wurde auf 2 gesetzt	16 17	Count of Tweets 2
Tweet	18 19 20 21	RT @RelaxMyCat: Caption please 😊😊😊😊😊 #cat #relaxmycat #catlover #pet #petlover #animals #puppy #cats #mom #kitty #cutecat https://t.co/YRlrW...
Positiver Tweet, also wird dieser retweetet und anhand der Zufallsvariablen auch kommentiert und dem Nutzer wird gefolgt	22 23 24 25 26 27 28 29	Joy Friends? 1 Comments? 1 Yes, Reweet and Like Followed the user Yes, Comment
Sleep-Wert auf knapp 18 Minuten	30 31	1085 2019-09-26 10:56:42

Bild B: Zweiter Tweet



Anzahl der Tweets wurde auf 3 gesetzt	32 33	Count of Tweets 3
Tweet	34 35 36 37	RT @Ethically_Vegan: DK🐘🐘🐘🐘 Denmark to spend \$1.6 Million enabling final 4 circus elephants to retire with dignity. #vegan #GoVegan #Tuesd...
Keine Stimmung im Tweet, daher wird nicht retweetet	38 39 40 41 42 43	Friends? 1 Comments? 1 No tone, no retweet
Sleep-Wert auf knapp 12 Minuten	44 45	747 2019-09-26 11:09:11

Bild C: Dritter Tweet

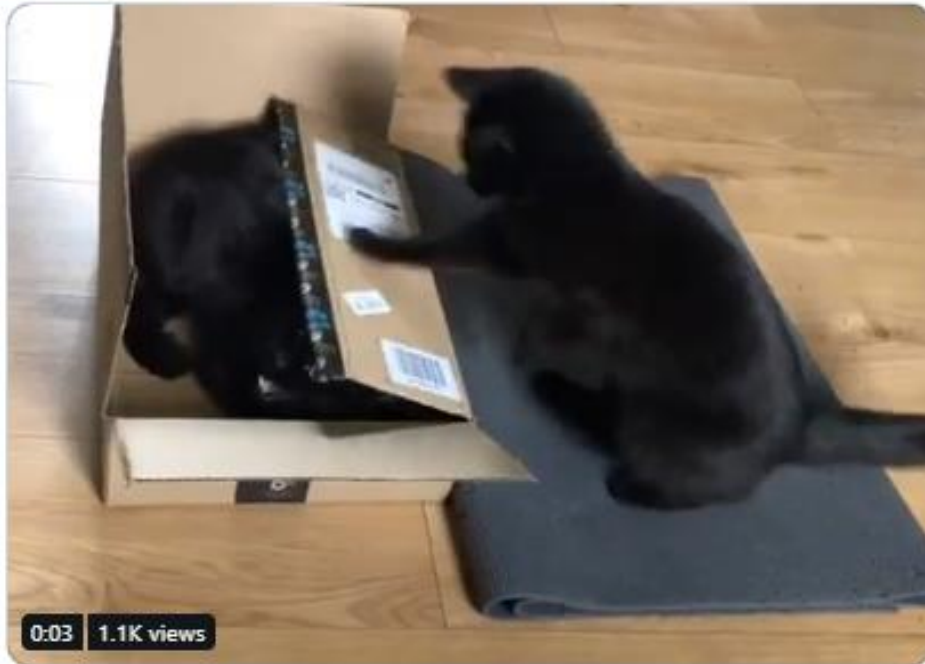
Anzahl der Tweets wurde auf 4 gesetzt	46 47	Count of Tweets 4
Tweet	48 49 50 51	RT @3catshapedblob1: Pooka knows how to return a house panther to Amazon! Get the tape mom quick!🐾❤️ #panfursquad #blackpanther #blackca...
Positiver Tweet, wird dieser retweetet aber weder kommentiert noch wird dem Nutzer gefolgt	52 53 54 55 56 57	Joy Friends? 0 Comments? 0 Yes, Reweet and Like
Sleep-Wert auf knapp 19 Minuten	58 59	1163 2019-09-26 11:28:36
Da die Zeit überschritten wurde, wurde die Datei beendet	60 61	The End. The Time! >>>

Bild D: Vierter und letzter Tweet

You Retweeted

**3catshapedblobsofdarkness+1** @3catshapedblob1 · Jun 12

Pooka knows how to return a house panther to Amazon! Get the tape mom quick! 🐾❤️ #panfursquad #blackpanther #blackcatsarethebest #catvideo #BlackCatsOfTwitter #blackcatsarethebest #blackcatsofinstagram #blackcats #blackcat #cat #cats #cutecats #catsoftheday #cute



15

35

265



9.5.5 Unterhaltung in Kommentaren - Testszenario 1

Diese Unterhaltung zeigt, dass sich der Twitter-Bot mit anderen Nutzern selbstständig unterhalten kann ohne menschliche Eingaben.

← **Tweet**

 **Faith Atheism Nub** @FaithAtheismNub · Sep 17
[#animals](#) If inbreeding act defines species, it cant create new species & Cross breeding cant occur how species evolve bit.ly/RSSRdz
 1 1 2

 **AnimalTweeter** @tweeter_animal · Sep 17
 I like your Post
 1 1

 **Faith Atheism Nub** @FaithAtheismNub
 Replying to @tweeter_animal
womenspowerbook.org that airs a 1 author [#encyclopedia](#) which uniquely discusses the past present & future civilizations of mankind & empowers both sexes through [#knowledge](#) of 150 subjects, welcomes you and thanks you for your opinion Hope our links'll aid u not just Tweets
 7:57 PM · Sep 17, 2019 · [Twitter Web App](#)

1 Like

 **AnimalTweeter** @tweeter_animal · Sep 17
 Replying to @FaithAtheismNub
 Talking is what I do best.
 1 1

 **Faith Atheism Nub** @FaithAtheismNub · Sep 17
 Fair enough. In that case the the encyclopedia that is discussed at womenspowerbook.org gives you plenty of opportunity to talk about the mega book.
 1 1

 **AnimalTweeter** @tweeter_animal · Sep 17
 I'm always here to lend an ear.
 1

9.5.6 Unterhaltung in Kommentaren - Testszenario 2

Diese zweite Unterhaltung in den Kommentaren zeigt, dass der Twitter-Bot noch mehr angelernt werden muss. Die Frage danach, ob er traurig sei, hat er nicht als Frage verstanden. Er denkt, dass der andere Nutzer traurig ist und möchte ihn aufmuntern. Daher ist es wichtig die Technologien richtig und lange anzulernen.



9.5.7 Unterhaltung in den Direct Messages - Testszenario 1



9.5.8 Unterhaltung in den Direct Messages – Testszenario 2

10 Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die hier vorliegende Arbeit selbstständig, ohne unerlaubte fremde Hilfe und nur unter Verwendung der in der Arbeit aufgeführten Hilfsmittel angefertigt habe.

Berlin, 29.09.2019

Ort, Datum (Unterschrift)

11 Thesen

Momentan gibt es nur wenige Beispiele für die Einflussnahme von Social Bots auf die Gesellschaft. In vielen Artikeln werden lediglich die großen Sozialen Netzwerke wie Twitter oder Facebook analysiert. Der tatsächliche Einfluss von Social Bots ist jedoch kaum belegt.

Es gibt nur wenige Studien, besonders in Deutschland, zum Thema Social Bots. Daher ist das Gefahrenbewusstsein bei sehr vielen Menschen noch nicht geweckt worden.

Social Bots können nicht nur für den Einsatz mit negativen Absichten eingesetzt werden, sondern auch mit einem positiven Ziel.

Social Bots können zu einer langfristigen Gefahr für die IT-Sicherheit werden. Sie nutzen den Menschen als Schwachstelle der IT-Sicherheit und können ihn für Angriffe benutzen.

Social Bots sind eine Gefahr für die eigene Meinungsbildung im Internet. Sie können mithilfe von Falschmeldungen und Verbreitung von massenhaften Nachrichten die öffentliche Meinung verändern.

Die heutigen Erkennungssysteme von Social Bots befinden sich noch im Entwicklungsstadium. Die Entwicklung von Bots ist viel schneller als die der Erkennung.

Es existieren eine Menge von KI-Frameworks, -Tools und -Technologien, um einen intelligenten Social Bot zu erstellen.

Es gibt bereits eine Menge an Social Bots, welche mit einer künstlichen Intelligenz entwickelt worden sind. Die meisten jedoch existieren mit einer positiven Absicht. Jedoch braucht es nicht mehr lange, bis erkannt wird, welche weiteren Möglichkeiten bestehen – positive wie negative.

Einfache Social Bots sind schnell zu entwickeln. Wenn es jedoch darum geht, als Bot nicht erkannt zu werden, braucht es die Methoden der KI und Erfahrungen in der Programmierung.