Die Sentiment Analyse: von der Linguistik bis zum alltäglichen Gebrauch

Posavac, Ines

Undergraduate thesis / Završni rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Humanities and Social Sciences / Sveučilište u Rijeci, Filozofski fakultet u Rijeci**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:186:420201

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-13**

Repository / Repozitorij:

Repository of the University of Rijeka, Faculty of Humanities and Social Sciences - FHSSRI Repository

zir.nsk.hr

UNIRI DIGITALNA KNJIŽNICA

dabar
DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

UNIVERSITÄT IN RIJEKA
PHILOSOPHISCHE FAKULTÄT IN RIJEKA
ABTEILUNG FÜR GERMANISTIK

Die Sentiment Analyse:
Von der Linguistik bis zum alltäglichen Gebrauch

Abschlussarbeit

Verfasst von: Ines Posavac

Geleitet von:doc. dr. sc. Suzana Jurin

In Rijeka, September 2018

Inhalt

Inhalt 2
1 Einführung 3
2 Die linguistische Vorbereitung für die Sentiment- Analyse 4
2.1 Die Gestaltung des Sentiment Lexikons 5
2.1.1 Die „German Polarity Clues Datenbank“ 6
2.2 Die Syntax- Analyse für die Sentiment- Analyse 8
2.2.1 Die generative Transformationgrammatik 8
3 Die Kommunikation verbindet die Menschen mit dem Computer 9
3.1 Die digitale und analoge Kommunikation 9
3.1.1 Digital und analog bei Computern 10
4 Die Erstellung des Algorithmus für die Sentiment- Analyse 11
4.1 Von der Sprache zum Automat 12
4.1.1 Die Grammatik für den *Generator* 13
4.1.2 Der Ableitungsbaum für die natürliche Spache 13
4.2 Der Parser für die Sytaxanalyse 14
4.3 Deep Learning und Data Mining 16
5 Die Beispiele der Algorithmen für die Sentiment- Analyse 17
5.1.1 Die „Sentiment Analysis with Python NLTK Text Classification“ 17
5.1.2 Die „Sentiment Text Analysis Software“ 18
5.2 Die Bereiche wo man die Sentiment- Analyse Algorithmen benutzt 19
6 Schlussfolgerung 20
7.1 Quellenverzeichnis 21
7.2 Literaturverzeichnis 22

1 Einführung

Die Sentiment- Analyse[1] ist heutzutage ein ganz beliebtes und aktuelles Thema. In dieser Zeit, wenn die Computer immer mehr die Arbeit der Menschen einnehmen, können Maschinen auch das Sentiment erkennen. Doch was ist eigentlich das Sentiment? Nach Duden bedeutet der Begriff Empfindung oder Gefühl und kommt von lateinischen Ausdruck *sentire*, zu der Form *Sentenz* (Meinung, Urteil, Gedanke) zum *Sentiment* (vgl. Dudenverlag 2007: 1533). Bei den Computern geht die Erkennung von Gefühlen nicht instinktiv wie bei den Menschen, deshalb hat die Fachwelt einen Weg gefunden, wie man es mathematisch ausrechnen kann und folgend in ein Programm implementieren. Einerseits stellt sich auch die Frage, wie korrekt kann eingentlich eine erwachsene Person die Empfindung und das Gefühl der anderer Person einschätzen.

Dabei muss man in Acht nehmen, dass es sich hier auch um mehrere Disziplinen der Linguistik handelt. Es geht nämlich um die natürliche Sprache die man bearbeitet. Hier ist dieses Kommunikationsmittel im Mittelpunkt, weil das System durch die digitale Mitteilung[2] von den eingegebenen Texten per Computer, formal nur die Fakten objektiv wahrnehmen kann.

Um die Sentiment- Analyse zu verstehen, ist sowohl die Konstruktion des Algorithmus[3] für die Sentiment- Analyse, als auch der linguistiche Teil wichtig. Deswegen hat die Arbeit zum Ziel, die zwei Bereiche zu verbinden und die ganze Struktur der Sentiment- Analyse darzustellen, um zu zeigen, wie die verschiedene Sachbereiche gut im Zusammenhang funktionieren.

Also wird in dieser Arbeit zuerst der Hintergrund der Sentiment- Analyse beschrieben, der mit der Linguistik anfängt. Die Textanalyse geht, wie auch jede Analyse, von einem größen Umfang bis zu den kleinen Elementen. Wichtig ist es auch die ganz kleinen Eigenschaften für die Bearbeitung vorzubereiten, damit wird nämlich der Algorithmus effektiver. Deshalb beschreibt diese Arbeit, wie das Lexikon aufgebaut wird und wie es als Datenbank strukturiert und gespeichert aussieht. Danach wird in dieser Arbeit die syntaktische Strukturierung vorgestellt, also wie es von diesem Lexikon über die Sätze zum Text kommt. Dabei beschäftigt sich diese Arbeit mit der Methode des Linguisten Noam Chomsky. Im letzten Teil der linguistischen Vorbereitung, wird die Funktion der Semantik erklärt, da die Disziplin die den Inhalt erkundet, eng mit dem Sentiment verbunden ist.

[1] Die Sentiment- Analyse kommt von dem englischen Ausdruck „Sentiment Analysis“ und wird durch die ganze Arbeit als solcher benutzt.
[2] Sehe Kapitel 4.

[3] Verfahren zur schrittweisen Umformung von Zeichenreihen; Rechenvorgang nach einem bestimmten [sich wiederholenden] Schema. (DUDEN 2007: 121)

Folgend wird im Kapitel 4 die Computerlinguistische Perspektive thematisiert. Was sie bedeutet, wie kommunizieren wir mit dem Computer und wie sich nicht nur die Technologie mit der analogen und digitelen Modalitäten der Kommunikation bedient, sondern auch die natürliche Sprache.

Die Programmrealisierung des Algorithmus für die Sentiment- Analyse kommt nach den vorbereiteten linguistischen Daten. In dem Kapitel 4, der sich mehr auf die Informatik basiert und wie es von den vorbereiteten Daten zu dem Algorithmus für die Sentiment- Analyse kommt, beschreibt die Arbeit den Aufbau des Programms, das speziell für die Sentiment- Analyse aufgebaut ist. Zudem hat diese Arbeit zum Ziel die Begriffe Deep Learning und Data Mining zu beschreiben und zu erklären, wie diese Begriffe mit dem primären Thema verbunden sind.

Zu letzt sollen in dieser Arbeit bestimmte Beispiele von den Algorithmen für die Sentiment- Analyse gezeigt werden um die beschriebenen Fakten darzustellen, aber auch wie man die Methode der Sentiment- Analyse nutzen kann und in welchen Bereichen man solche Programme gebraucht.

2 Die linguistische Vorbereitung für die Sentiment- Analyse

Die Sentiment- Analyse ist ein Zugang der Neuzeit. Doch schon jetzt ist dieser Bereich sehr attraktiv, sodass man die Algorithmen für die Sentiment- Analyse für viele Sprachen entwickelt hat. Deshalb erstellt man Datenbanken, die den Vokabular der Sprache enthalten, wobei jedem Eintrag bestimmte Charakteristiken zugeordnet sind. Diese Bearbeitung ist ein komplexer Prozess, wenn man in Acht nimmt, wie viele Einträge es in diesem Dokument geben wird. Deswegen ist die Analyse des Lexikons eine Arbeit für sich selbst. Doch um das Sentiment einer Aussage zu ermitteln, muss man den Inhalt des ganzen Satzes bzw. Textes bearbeiten. Diese Funktion übernimmt die Syntax, die durch bestimmte Regeln die Konstruktion diktiert. Die Gestaltung von Sätzen kann man durch mehrere Methoden durchführen, dabei kann man sie mit Hilfe von kleinen Einheiten generieren. Hier sind nicht nur die Struktur und die Funktion wichtig, sondern auch die semantische Bedeutung der Wörter, Sätze und Texte. Erst dann, wenn man diese Bereiche ermittelt hat, kann man leicht den Sentiment erkennen. (vgl. Waltinger : 21.8.2018)

2.1 Die Gestaltung des Sentiment Lexikons

Je mehr Informationen der Eintrag hat, desto schneller ist die Suche bei der Durchführung des Programms. Im Text wird nicht immer das Lemma[4] benutz. Das Wort kann durch Komparation oder Deklination geprägt werden. Wenn es sich um das Nomen, das Pronomen oder den Artikel handelt, werden diese Wortarten nach dem Gebrauch dekliniert. Deshalb werden alle Formen des Wortes eingetragen, wobei man die Grundform, also das Lemma, auch anführt. So können die Fehler vermieden werden, wenn sich zum Beispiel zwei oder mehrere Wörter in bestimmten Fällen übereinstimmen. Erst dann wird ein Wort zu einem Eintrag in dem Lexikon.

Nachdem folgt die Wortart. In der deutschen Sprache unterscheidet man das Verb, das Substantiv, das Adjektiv, das Adverb, die Präpositionen, die Konjunktionen und die Interjektionen. In englischer Sprache nennt man sie Part-of-Speech. Sie dienen zuerst um bessere Unterscheidung zwischen gesuchten Einträgen, aber später auch, wenn ein Satz analysiert wird. Die Sentiment- Analyse begrenzt sich von weitem nicht nur auf die Wort-für-Wort Erkennung.

Neben allen anderen Angaben des Eintrags, ist natürlich die Wortart die wichtigste Angabe, nach der sich auch die anderen zuweisen. Außerdem ist der nächste Schritt die Semantik. „Die Syntax einer Sprache legt fest, welche Ausdrücke zur Sprache gehören. Die Semantik legt fest, was die Ausdrücke bedeuten." (Digitale Informationsverarbeitung: 20.12.2001: 1) Da die Sentiment- Analyse die Funktion hat, die Gefühle und Meinungen mit Hilfe eines Algorithmus zu erkennen, muss man erst den Inhalt verstehen.

Es ist in der menschlichen Natur, mit Jahren von Erfahrung, das Gefühl in der Aussage der sprechender Person zu erleutern. Aber wenn es sich um ein Programm handelt, das zum solchen Zweck dient, bedürft es viele Informationen. Die Semantik hilft dabei, die Bedeutung des Wortes, Phrase, Satzes oder Textes festzustellen. Nach der Bedeutung erkennt man dann die Polarität weiter. Wenn es sich um ein Wort handelt, stellt es am Anfang noch ein einfaches Prozess dar. Zum Beispiel „gelb" ist ohne Kontext neutal, „glücklich" wäre als positiv gekennzeichnet und „traurig" als negativ. So bekommen Einträge eine binäre Wertung, was bedeutet, dass von den drei möglichen Empfindungen (negativ, positiv und neutral), eine von ihnen die Wertung 1 bekommt, wobei die anderen zwei Empfindungen den Wert 0 haben.

Außer der allgemeinen Polarität, kann die Polarität auch von dem Text abhängen. Es gibt auch drei Möglichkeiten (negativ, positiv und neutral), aber es gebraucht einen anderen Zugang, als bei dem

[4] *„Alles, was man nimmt, (Fachspr. Stichwort in einem Nachslagewerk- Wörterbuch oder Lexikonon)."* (Dudenverlag 2007: 1071)

Wert, der im letzten Abschnitt beschrieben wurde. Es handelt sich nämlich um die Wahrscheinlichkeit. Die Wahrscheinlichkeit eines Vorkommens liegt zwischen 0 und 100 Prozent. Gleich ist es in diesem Fall. Wenn man den Eintrag im Lexikon für die Sentiment- Analyse als kontextabhängig anschaut, bekommen die Polaritäten nach der Formel der Markow Kette[5] einen Wahrscheinlichkeitswert.

(vgl. Waltinger: 21.8.2018)

Nach der Formel bezeichnet man die Wahrscheinlichkeit mit p und den Text mit w. So ist nach der Formel $p(w) = p(w_1,w_2,w_3,....w_m)$ die Wahrscheinlichkeit, dass alle Wörter, die in diesem Text beinhaltet sind, vorkommen, bei 100 Prozent. Nach Markow gilt dann die nächste Gliederung: $p(w)= p(w_1)p(w_2|w_1)p(w_3|w_1,w_2)\ldots p(w_m|w_1,w_2\ldots w_{m-1})$. Bei dem konkteren Fall der Polaritätwahrscheinlichkeit bedeutet das, dass die Wahrscheinlichkeit nur von dem fixen Wert vorkommt und dass die vorigen Werte zeitlich keinen Bezug auf die zükunftigen haben. Also ist die Wahrscheinlichkeit für die Polarität eines Wortes in deriger Zeit augerechnet, unabhängig davon, ob das Wort schon detektiert würde und mit welchen Auskommen. (IJCSI International Journal of Computer Science Issues: 21.8.2018)

2.1.1 Die „German Polarity Clues Datenbank“

Die Bereitung des Lexikons ist eine umfangliche Arbeit für sich und es gibt immer mehr Datenbanken die aus verschiedenen Texten, aus unterschiedlichen Bereichen die Liste der Wörter ausgezogen haben. Die Wörter werden dann mit Vorbereitung, die im vorigen Kapitel beschrieben sind, zu Einträgen im Lexikon. Solche umfangliche Listen gelten als Datenbanken für die nachträgliche Benutzung. Ein Beispiel von so einer Datenbank für die Sentiment- Analyse in der deutschen Sprache ist die „German Polarity Clues Datenbank“. Der Autor ist dr. Ulli Waltinger, wessen Arbeit im folgenden Kapitel erläutert wird. Er verbindet dieses Sachbereich mit der Polarität-Klassifikation, der Web- Intelligenz und dem Maschinellem Lernen (Machine Learning). Im Jahr 2011 war die erste Version des Lexikons veroffentlicht. Den Inhalt seiner Arbeit kann man frei und kostenlos herunterladen und auch benutzen. Das Lexikon verfügt über 10 141 Einträge mit den Merkmalen, die in dem Kapitel 3.1 beschrieben sind. Jede Zeile in der Datai ist ein Eintrag, wobei die Einträge voneinander mit einem Tabulator abgegliedert sind. (vgl. Waltinger : 21.8.2018)

[5] Die Markow Kette bezieht sich auf ein Prozess, wobei die Wahrscheinlichkeit des nächsten Vorkommens hängt nur von dem derzeitigen Status, aber hat kein Bezug auf die vorigen Vorkommen. (IJCSI International Journal of Computer Science Issues: 21.8.2018)

Die erste Version aus dem Jahr 2011 war mit SubjectivityClues (Wiebe et al. 2005) und SentiSpin (Takamura et al., 2005) synchronisiert.

March, 2011 Format

```
Feature    (\t)     Lemma     (\t)  Part-of-Speech (\t)  PositiveRating (\t)  NegativeRating (\t)  NeutralRating (\t)
überzahlte          überzahlen          V                  0                      1                    0

PositiveCorpusProbability (\t)  NegativeCorpusProbability (\t)  NeutralCorpusProbability
0                               0.6                              0.4
```

Bild 1 Die German Polarity Clues Datenbank, 2011 Format (http://www.ulliwaltinger.de/sentiment/)

Die Einträge sind in der ersten Version mit mehr Details dargestellt, wobei man den Wert der positiven, negativen und neutralen Polarität sehen kann. Dabei kann man, allgemein gesehen, einem Eintrag nur eine Polarität zuordnen und die andere Polaritäten bekommen einen 0- Wert. Bei den Wahrscheinlichkeiten ist auch der Wert einzeln gezeigt. Das Beispiel auf dem *Bild1* hat im Korpus einen Wert von 60 Prozent der negativen Polarität- Wahrscheinlichkeit und 40 Prozent der neutralen Polarität- Wahrscheinlichkeit ausgegeben. Wobei der Eintrag als Wort im Text in keinem Fall als positiv benutzt war. Da dieses Format zu viele Details enthielt, veroffentlichte dr. Ulli Waltinger ein Jahr später eine neue Version der „German Polarity Clues Datenbank“.

Im Jahr 2012 erschien dann das verkürzte Format, das auf dem *Bild 2* unten gezeigt ist. Es verfügt über die gleiche Merkmale wie der Vorige und als so zusammengefasst, ist es leichter im Programm zu benutzen. Der Algorithmus für die Sentiment- Analyse kann leichter und schneller mit den Daten arbeiten und auch für den Programmierer ist es leichter einen Code für solchen Algorithmus für die Sentiment- Analyse zu schreiben. Im Beispiel auf dem *Bild 2* seht man, dass das Adjektiv als negativ bezeichnet ist und nach der Polarität- Wahrscheinlichkeit im Text mit einem Wert von 0.3412 ausgekommen ist.

April, 2012 Format

```
Feature    (\t)     Lemma     (\t)  Part-of-Speech (\t)  Polarity (\t)  Probability (\t)
hoffnungsloser          hoffnungslos        AD                          negative          -/-0.3412/
```

Bild 2 Die German Polarity Clues Datenbank, 2012 Format (http://www.ulliwaltinger.de/sentiment/)

(vgl. Waltinger : 21.8.2018)

2.2 Die Syntax- Analyse für die Sentiment- Analyse

Der Name Chomsky wird oft in der Linugistik, als auch in der Informatik erwähnt. Demzufolge können wir diese Bereiche mit Hilfe seiner Arbeiten verbinden. Die Grammatik ist ein geregeltes System, dass uns diktiert wie man eine Sprache richtig benutzt. Da der Begriff Sprache sich nicht nur auf die natürliche Sprache begrenzt, sondarn auch auf die formale Sprache zum Beispiel, kann man die Begriffe Sprache und Grammatik durch die Linguistik, aber auch die Informatik beschreiben. Chomskys Wirkung auf die formale Sprachen und wie es im Zusammenhang mit der Sentiment- Analyse ist, wird im Kapitel 5 beschrieben. Im Fokus dieses Kapitels ist aber seine Grammatik und sein Zugang zu der natürlichen Sprache.

2.2.1 Die generative Transformationgrammatik

„Diese Grammatik hat sich zum Ziel gesetzt, diejenigen Regeln zu beschreiben (oder aufzustellen), nach denen Sätze in einer gegebenen Sprache generiert (= erzeugt) oder interpretiert (= verstanden) werden.“ (Petrović 2001: 89) Wie der Name auch schon ansagt, generiert Chomskys Methode Sätze mit Hilfe von Phrasen, die bestimmte Merkmale tragen. So sollen nach Chomsky, alle Menschen, allein durch diese Merkmale den Text verstehen, auch wenn sie diese Sprache nicht sprechen.

Die Generierung funktioniert allgemien so, dass man Phrasen hat, mit denen man arbeitet und die man mit bestimmten Regeln kombinieren kann. Die Funktion des Wortes im Satz diktiert Regeln zu seinem Gebrauch. Nach Chomsky teilt sich jeder Satz in der englichen Sprache binär- auf die Nominal- und Verbalphrase, doch in der deutschen Sprache kann die zweite Stufe neben dem Subjekt und Prädikat, auch ein Objekt beinhalten. Nachdem gliedern sich dann diese Phrasen in kleinere Einheiten, bis alle Wörter gruppiert sind und nach der Funktion gekennzeichnet. Das Wort wird dann mit dem Eintrag im Lexikon vergleicht und so wird die Polarität hinzugefügt. Die Merkmale kann man in ein Programm schreiben und so ein Automat als Ziel bekommen, der die Generierung selbst bearbeitet. Mehr davon wird im Kapitel 5 erwähnt.

Doch semantiche Merkmale haben die Phrasen auch, oder Symbole nach der Chomskys Benennung. Mit Bedacht auf die Merkmale, muss man sie aber gar nicht verstehen. Zum Beispiel können wir Phrasen A-> die Katze, B-> die Maus , C-> sehen haben und mit denen Sätze bilden. Aus denen können wir zwei Sätze generieren. Der erste wäre dann *Die Katze sieht die Maus.* und der zweite *Die Maus sieht die Katze*.

Um die Phrasen zu generieren, stellt Chomsky vier Merkmal- Regeln. Sie charakteriseren die Semantik der Phrase bzw. des Symbols. Das Symbol kann human, belebt, zählbar oder abstrakt sein. Im oberen Beispiel *Die Katze sieht die Maus.* ist das Verb *sehen,* dass ein Symbol mit dem Merkmal *belebt* gebraucht. Nur die Lebewesen können nämlich sehen. Diese Regeln sind auch für die Sentiment- Analyse wichtig, um die Generierung von falschen Sätzen zu vermeiden.

(vgl. Petrović 2001: 89-94)

3 Die Kommunikation verbindet die Menschen mit dem Computer

"Man kann nicht nicht kommunizieren." (Walzlawick 1996: 53) Im Mittelpunkt dieses Kapitels steht die Frage, wie man die Kommunikation der Menschen mit der Kommunikation mit dem Computer verbinden kann. Im Ziel das zu beschreiben, wird Paul Walzlawick erwähnt. Da sich diese Arbeit mit der Sentiment- Analyse beschäftigt, die versucht den Computer die Erkennung von Gefühlen „beizubringen“, wird im folgenden Teil der Arbeit erklärt, wie die Idee der Gefühlerkennung eigentlich aussieht; bei Menschen miteinander und bei den Menschen zu dem Computer.

3.1 Die digitale und analoge Kommunikation

" Menschliche Kommunikation bedient sich digitaler (verbaler) und analoger (non-verbaler, nicht-sprachlicher) Modalitäten (Ausdrucksmittel). Digitale Kommunikationen haben eine komplexe und vielseitige logische Syntax aber eine auf dem Gebiet der Beziehungen unzulängliche Semantik (Bedeutungslehre). Analoge Kommunikationen hingegen besitzen dieses semantische Potential, ermangeln aber die für eindeutige Kommunikation erforderliche logische Syntax". (Walzlawick 1996: 68) Auf die Frage was digital und analog bedeutet, beantwortet das 4. Axiom von Paul Walzlawick. Die digitale Mitteilung beschreibt Walzlawick als die Verbindung von Objekt und Begriff, oder die Inhaltsebene, deshalb kann man diese Mitteilung mit der verbalen Kommunikation verbinden. Ob in der schriftlichen oder gesprochenen Kommunikation, übermittelt die digitale Mitteilung nur den konkreten Inhalt der Aussage. Natürlich kann man so eine Art von Kommunikation nur theoretisch besprechen, wenn man die Kommunikation zwischen zwei sprechenden Personen erklärt. Die Gestik und Mimik gehören zu der analogen Kommunikation, wobei es nicht möglich wäre, dass zwei Personen miteinander sprechen, ohne mit dem Blickkontakt oder der Körpersprache zu kommunizieren. Bei der Sprache in der Schrift ist es ein bisschen anders. Wenn man ein Satz wie „Das ist das schönste Kleid das ich je gesehen habe.“ in Acht nimmt, kann es entweder eine sehr

positive Reaktion bedeuten, oder durch Sarkasmus eine sehr schlechte Reaktion auswirken. Um die non- verbale oder analoge Mitteilung auch in schriftlicher Kommunikation zu erreichen, gebraucht man Emoticons in Chats auf verschiedenen Plattformen. Die Emoticons spiegeln immerhin den Gesichtsausdruck wieder. Wichtig ist es auch zu erwähnen, dass die analoge Mitteilung nicht immer die Garantie für das Verstehen des Kontexts ist, weil man auch dann die Mimik, Gestik oder das Emoticon falsch verstehen kann. (vgl. Bauer. DIGITALE & ANALOGE KOMMUNIKATION – BEISPIEL AUS DEM ALLTAG: 23.8.2018)

Doch die analoge Mitteilung reflektiert bei den Menschen Gefühle, Emotionen und Reaktionen, also das Sentiment. In unserer Natur ist es zu erkennen, dass, wenn die Person lacht, sie glücklich ist. Bei dem Beispiel „Das ist das schönste Kleid das ich je gesehen habe." kann man wegen der Übertreibung, leicht ein Grinsen erwarten, wobei es sich dann um eine sarkastische Aussage handelt.

3.1.1 Digital und analog bei Computern

Die Begriffe analog und digital sind viel mehr in der Technik verbreitet und in dem Bereich auch bekannter. Doch nach der Definierung in der Linguistik, kann man diese auch mit der Definierung in der Technik vergleichen. Robert Schanze erklärt es mit Hilfe der digitalen und analogen Uhr. Die analoge Uhr kann den Zeiger zwischen zwei Einheiten haben, wobei es sich dann um eine Zwischenstufe handelt und man nicht genau weiß auf was der Zeiger anzeigt. Bei der digitalen Uhr ist es auf der anderen Seite immer klar welchen Wert sie anzeigt, aber den Zwischenwert kann man nicht sehen, weil sie ihn immer für genau eine ganze Stufe erhöcht. Die Begriffe kommen in der Technik auch bei Fotos, Audio und Video, bei Fernsehern usw. vor. (vgl. Schanze. Unterschied zwischen analog und digital? – Einfach erklärt: 23.8.2018)

Mit der Ausarbeitung zeigt sich der Zusammenhang zwischen der menschlichen Kommunikation und der Technik. Wenn ein Programmierer einen Code schreibt, kommuniziert er mit dem Computer und gibt ihm bestimmte Instruktionen, wie und was das Programm machen soll. In diesem Fall handelt es sich um ein Beispiel für den Algorithmus für die Sentiment- Analyse. Auf der anderen Seite, kommuniziert auch der Benutzer des Algorithmus für die Sentiment- Analyse mit dem Computer.

Wenn das Programm in dem Prozess des Schreibens ist, kann der Computer unsere Erkennung von Gefühlen nicht einfach erlernen. Der Computer kann auch unsere Intension nicht erkennen, weil das Programm digital nach den Anweisungen funktioniert, die der Code beinhaltet. Deswegen ist es auch schwer einen funktionalen Algorithmus für die Sentiment- Analyse zu programmieren, weil das Programm nur ein Automat ist, das den Gebraucher analog versteht. (vgl. Haugender 1989: 68-70)

Die Kommunikation mit den Computern kann man mit der Kinderkommunikation vergleichen. Die Erkennung von Sentiment ist uns durch Erfahrung zugeteilt. Erst mit der Zeit und der mentalen Entwicklung kann die Person die Fähigkeit der analogen Kommunikation ausüben. Ein dreijähriges Kind wird eine Aussage nicht als ironisch oder sarkastisch erkennen können. Also wenn man einem Kind das oben erwähntes Beispiel „Das ist das schönste Kleid das ich je gesehen habe." sagt, wird es die Aussage nur so begreifen können, wie es digital und rein faktisch bedeutet. Auf diese Art und Weise funktionieren auch Computerprogramme. Das Kind lernt durch die Erfahrung das Sentiment zu erkennen, wenn es sich zum Beispiel um eine sarkastische Aussage handelt. Eine Art von Erfahrung soll auch der Computer erwerben. Deshalb ist die Sentiment- Analyse ein komplexes Projekt, wobei man den Algorithmus für die Sentiment- Analyse Instruktionen gibt, dass es die ausgerechneten Werte des Sentiments im Lexikon- Einträgen zu den Wörtern im Text zuordnet. (vgl. Hutson. How researchers are teaching AI to learn like a child : 12.9.2018)

Die Negation ist übrigens auch ein komplexes Thema. Die Aussage „Ich mag den Film nicht." trägt eine negative Polarität, obwohl das Verb *mögen* für sich positiv wirkt. Auf der anderen Seite ist die Aussage „Man kann den Film nicht nicht mögen." trotz der Negation positiv. Im Beispiel „Der Film ist nicht nur komisch, sondern auch informativ." hat man eine neue Form von der nicht- Negation, wo es komplex wäre dem Programm eine Anweisung zu geben, wobei der Begriff meistens eine positive Polarität besitzt und hier verneint ist. Am Beispielen seht man, wie viele Kombinationen und Ergebnisse der Algorithmus für die Sentiment- Analyse haben kann. (Negation Handling in Sentiment Analysis at Sentence Level: 6.9.2018)

4 Die Erstellung des Algorithmus für die Sentiment- Analyse

Nachdem linguistich die Grammatik vorbereitet ist, bestimmte Regeln eingeordnet wurden, sowie syntaktisch, als auch semantisch, beginnt man mit der Erstellung des Algorithmus für die Sentiment- Analyse. Erst einmal bearbeitet dieses Kapitel die Sprache allgemein und wie man sie von verschiedenen Aspekten verstehen kann. Im dritten Kapitel war die natürliche Sprache beschrieben, wenn man aber über die Programmerstellung spricht, ist im Mittelpunkt die formale Sprache. Hier ist dieses Kommunikationsmittel die geregelte Art, wie man das Computerprogramm leitet. Aber zur Bearbeitung der gleichen, gebrauchen wir ein Programm das speziell für diese Anwendung programmiert ist. Die Sentiment- Analyse vergleicht den eingegebenen Text mit dem Daten aus der Datenbank. Deshalb muss man vorher die Daten vorbereiten, die Wörter in Wortarten einordnen und ihnen bestimmte lexikalische, syntaktische und semantische Charakteristiken einordnen. Solche

Automate generieren Sätze in Phrasen, durchsuchen einzeln, aber auch als Einheit. Zu letzt bekommen die Wörter auch eine Sentimentmarkierung. Die lingusitische Vorbereitung ist im dritten Kapitel bearbeitet. Die ganze Struktur kann man graphisch als ein sogennantes *Treebank* (der Ableitungsbaum) darstellen. Wie es von der Grammatik zum *Treebank* kommt und wie diese Analyse ein Computerprogramm machen kann, beschreiben die nächste Abschnitte. Um die Erstellung des Algorithmus für die Sentiment- Analyse ganz zu verstehen, werden in diesem Kapitel auch die Begriffe Deep Learning und Data Mining erklärt, weil sie zu Prinzipien für die Sentiment- Analyse gehören.

4.1 Von der Sprache zum Automat

Zum Verstehen des ganzes Systems der Sentiment- Analyse muss man erst die Sprache als Sachbegriff in dieses Sachbereich einordnen. Die Sprache, mit ihrem entsprechenden Vokabular, enthält nämlich die primäre Komponenten für den Aufbau der Struktur, die später ermittelt werden und zu Resultaten führen kann. Hier spricht man natürlich über die natürliche Sprache, immerhin sind die Eingabedaten Wörter, Phrasen, Sätze oder Texte, die man mit dem Korpus verbindet. Aber wenn man über die Sentiment- Analyse spricht, werden, wie schon erwähnt, die Linguistik und die Informatik verbunden. So bekommt der Begriff „Sprache“ eine neue Dimension. Also die Eingabedaten als natürliche Sprache, werden mit Hilfe eines, durch eine Programmiersprache implementierten, Algorithmus verarbeitet und analysiert, wobei man die Elemente mit den Einträgen aus der Datenbank vergleicht und so zu Resultaten kommt.

Nach Dovedan, „ist die Sprache als ein Unterraum der Menge von Zeichenfolgen aus dem Alphabet definiert.“[6] (Dovedan 2003: 19) Diese Menge ist natürlich sehr groß und als unendlich bezeichnet. Man kann sich gar nicht vorstellen wie viele Kombinationen von Zeichenfolgen da vorkommen können. Diese sind aber die Elemente die den Unterraum, durch das geregelte System der Grammatik, bilden. Deshalb benutzt man oft nicht die konkreten Elemente zur Beschreibung des Gleichen, sondern andere Methoden, die einfacher zu ermitteln sind, aber auch einfacher zu verstehen. Dovedan führt sich auch nach diesem Zugang und stellt drei Methoden vor. Die erste definiert die Sprache durch die regulären Ausdrücke, die zweite bezieht sich auf die Grammatik selbst, die als Hilfe um Konstruktionen zu bilden, dient. Diese Konstruktionen, die in den Sprachwissenschaften Sätze genannt werden, gelten hier als Produktionen. Die dritte Methode generiert durch mehrere Abläufe, die zum schon früher ausgerechneten Zuständen führen, vollständige Sätze. Das machen Automaten die zu diesem Zweck programmiert sind. Zu solchen

[6] „Jezik je definiran kao podskup skupa svih nizova znakova nad alfabetom.“

generierenden deterministischen endlichen Automaten, *Generatoren* genannt, also mit garantierten Ausgang, gehören auch Programme, die zur Sentiment- Analyse dienen. (vgl. Dovedan 2003: 19)

4.1.1 Die Grammatik für den *Generator*

Für den *Generator* brauchen wir, wie schon erwähnt eine Grammatik. Sie dient uns als Struktur für den Aufbau des Automaten. Nach der dritten Methode des Spezifikationssystems, kommt man mit Generierung von Wörtern und Phrasen zum endlichen Satz, aber so kann man auch von dem Satz anfangen und ihn in Satzglieder aufteilen. Die Phrasen, die als spezifisch in dieser Methode vorkommen, kommen auch bei der Bildung des *Generators* selbst vor, deshalb nennt man die Grammatik, die man dafür benutzt, die Grammatik der Phrasenstuktur. Ein anderer Name für diese Grammatik kommt von ihrem Autor und wird in der Literatur als Chomsky Grammatik gefunden.

Solche Grammatik benutzt nach Dovedan, zwei disjunktive Zeichensätze. Den nichtterminalen Zeichensatz, mit N markiert, benutzt man zur Bildung der Regeln für die Grammatik. Er gehört nicht zum Alphabet, sondern hilft nur zur Generierung von Sätzen in der Sprache. Auf der anderen Seite ist die terminale Zeichensatz T, aus dem Alphabet gebildet und steht als Basis für Generierung. Zur Bildung von Produktionen kommt man mit der Formel (N U T)* N (N U T)* x (N U T)*, wobei eine solche Produktion eine fertige Zeichenkette ist. (vgl. Dovedan 2003: 25)

Ein Programm der zur Analyse des Sentiments dient, vergleicht also Wort für Wort, oder mehrere zusammen, mit den Daten bzw. Einträgen, die vorher vorbereitet wurden. Zur Vorbereitung einer solchen Datenbank gebraucht man eine Analyse der Syntax. Neben allen anderen Angaben des Wortes, ist natürlich die Wortart die wichtigste, nach der sich auch die anderen zuweisen. Außerdem ist der nächste Schritt die Semantik: „Die Syntax einer Sprache legt fest, welche Ausdrücke zur Sprache gehören. Die Semantik legt fest, was die Ausdrücke bedeuten.“ (Digitale Informationsverarbeitung: 20.12.2001: 1)

4.1.2 Der Ableitungsbaum für die natürliche Spache

Am leichsten ist es die syntaktische Gliederung graphisch darzustellen. Das macht man in der Informatik, aber auch in der Linguistik mit einem Ableitungsbaum (eng. Treebank). Im dritten Kapitel war die Syntax- Analyse mit Hilfe der generativen Transformationsgrammatik bearbeitet. Die graphische Darstellung geht empirisch von dem Satz, teilt ihn ein und kommt bis zu den kleinsten Einheiten, wessen Funktion sich mit dem Part-of-Speech Tags bezeichnet.

Die Wurzel des Ableitungsbaums ist der Knoten, der den Satz darstellt, von welchem alle kleinere Zweige ausgehen. Der Satz teilt sich auf das Subjekt, Prädikat und Objekt, wenn es da ist. Diese Knoten zweigen sich dann weiter ab. Jeder Knoten hat immer genau einen Vorgänger, außer der Wurzel des Ableitungsbaums. Den Inhalt des Ableitungsbaums schreibt man und liest von der linken Seite nach rechts.

Am Beispiel eines Satzes wird der Ableitungsbaum bildlich gezeigt:

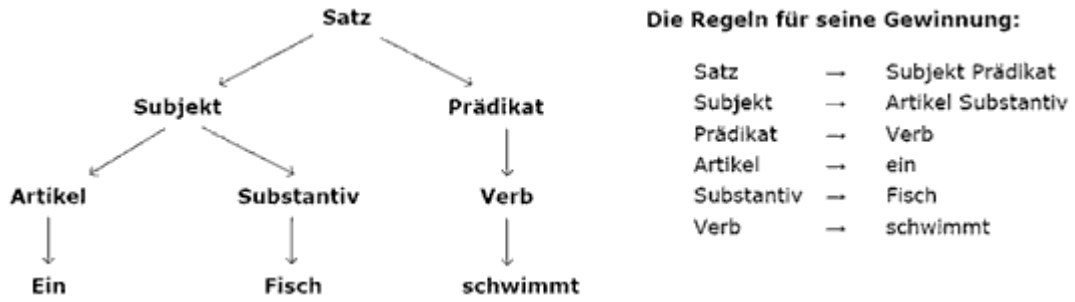

Bild 3 Der Ableitungsbaum (http://hki.uni-koeln.de/archive/hki2016/wisem-2011/basisinformationstechnologie-i/theoretische-informatik-i/praktische-anwendung-von-grammatiken-fuer-compiler.html)

4.2 Der Parser für die Sytaxanalyse

„Der Parser überprüft in der syntaktischen Analyse , ob mit diesen Eingabesymbolen (Tokens) die Programmkonstrukte (Reihungen, Schleifen, Selektionsanweisungen usw.) korrekt gebildet worden sind. Als Ergebnis wird ein so genannter Ableitungsbaum erstellt, aus dem dann nach einer semantischen Analyse schließlich der Zielcode generiert wird.“ (vgl. Historisch-Kulturwissenschaftliche Informationsverarbeitung: 15.8.2018)

Um die Gestalltung des Ableitungsbaums per Hand zu vermeiden, kann man dafür Programme benutzen, die die natürliche Sprache bearbeiten. Solche Programme gehören zu den Automaten für die Prozessierung von Sprachen. Ein Parser ist also ein Programm, der die Syntax- Analyse erledigt, wobei er Regeln benutzt, die man als Grammatik ins Code implementiert. Hier kommt es zu der Verbindung von der Linguistik und Informatik in der Form von der Grammatik. Man benutzt nämlich eine formale Grammatik, die als Regelsystem für das Programm gilt, um die Grammatik der natürlichen Sprache zu verstehen. Was eigentlich bedeutet, dass man die Grammatik für die natürliche Sprache die die Menschen benutzen und verstehen, in eine Grammatik für die Sprache die der Computer versteht, übersetzt. Am Ende wird dann das Programm erstellt und optimisiert. (vgl. Historisch-Kulturwissenschaftliche Informationsverarbeitung: 15.8.2018)

Anhand eines Beispiels wird dieser Prozess dargestellt.

20100041@1@20100041@(1, 0, 1, <0:2>, 1, "In", 0, "null", "IN" 1.0000) (2, 1, 2, <3:7>, 1, "this", 0, "null", "DT" 1.0000) (3, 2, 3, <8:16>, 1, "instance", 0, "null", "NN" 1.0000) (4, 3, 4, <16:17>, 1, ",", 0, "null", "," 1.0000) (5, 4, 5, <18:26>, 1, "industry", 0, "null", "NN" 1.0000) (6, 5, 6, <27:36>, 1, "observers", 0, "null", "NNS" 1.0000) (7, 6, 7, <37:40>, 1, "say", 0, "null", "VBP" 0.9855 "VB" 0.0145) (8, 7, 8, <40:41>, 1, ",", 0, "null", "," 1.0000) (9, 8, 9, <42:44>, 1, "he", 0, "null", "PRP" 1.0000) (10, 9, 10, <45:47>, 1, "is", 0, "null", "VBZ" 1.0000) (11, 10, 11, <48:56>, 1, "entering", 0, "null", "VBG" 1.0000) (12, 11, 12, <57:66>, 1, "uncharted", 0, "null", "JJ" 1.0000) (13, 12, 13, <67:73>, 1, "waters", 0, "null", "NNS" 1.0000) (14, 13, 14, <73:74>, 1, ".", 0, "null", "." 1.0000)@(120, 0, 1, <0:2>, 1, "in", 0, "null") (93, 1, 2, <3:7>, 1, "this", 0, "null") (107, 2, 3, <8:17>, 1, "instance,", 0, "null") (118, 2, 3, <8:17>, 1, "instance,", 0, "null", "NN" 1.0000) (95, 3, 4, <18:26>, 1, "industry", 0, "null") (113, 3, 4, <18:26>, 1, "industry", 0, "null", "NN" 1.0000) (97, 4, 5, <27:36>, 1, "observers", 0, "null") (114, 4, 5, <27:36>, 1, "observers", 0, "null", "NNS" 1.0000) (111, 5, 6, <37:41>, 1, "say,", 0, "null") (112, 5, 6, <37:41>, 1, "say,", 0, "null", "VBP" 0.9855) (99, 6, 7, <42:44>, 1, "he", 0, "null") (101, 7, 8, <45:47>, 1, "is", 0, "null") (115, 7, 8, <45:47>, 1, "is", 0, "null", "VBZ" 1.0000) (103, 8, 9, <48:56>, 1, "entering", 0, "null") (116, 8, 9, <48:56>, 1, "entering", 0, "null", "VBG" 1.0000) (105, 9, 10, <57:66>, 1, "uncharted", 0, "null") (117, 9, 10, <57:66>, 1, "uncharted", 0, "null", "JJ" 1.0000) (109, 10, 11, <67:74>, 1, "waters.", 0, "null") (119, 10, 11, <67:74>, 1, "waters.", 0, "null", "NNS" 1.0000)@80@350@410@380@0@418@119@-1@-1@285432@5472@3197@1908@1419@13@32@7529@16124@-1@-1@87840052@-1@1@1@30-8-2012 (11:15:40)@@(:nmeanings . 0) (:mtcpu . 0) (:clashes . 2276) (:pruned . 6) (:subsumptions . 3006) (:trees . 13) (:frozen . 171) (:equivalence . 85) (:proactive . 64) (:retroactive . 33) (:utcpu . 30) (:upedges . 402) (:failures . 1) (:hypotheses . 507) (:rtrees . 0) (:rreadings . 0)

Bild 4 Ein Beispielsatz aus dem IULA-Penn Treebank

Man erstellt Beispielsätze, die in bestimmter Form aufgebaut sind. Im Bild 4 seht ein Beispielsatz aus dem IULA-Penn Treebank. Die Anreihung von Zeichen und Symbolen stellt eine Grundlage für einen Parser dar. Jeder Satz hat ein Anfangssymbol (z.B 20100041@1@20100041@), der oft nach bestimmten Regeln von einer Gruppe von Zeichen besteht. Danach separiert man den Satz Wort für Wort, wobei alle Informationen über das Wort dann in einer Klammer stehen (z.B (2, 1, 2, <3:7>, 1, "this", 0, "null", "DT" 1.0000)). Dabei sind bei diesem Beispiel die Wortstellung im Satz, die Reproduzierbarkeit und Wiederholung im Text angeführt, die Funktion im Satz bzw. Part-of-Speech Tag und das Wort allein wird in Anführungszeichen angegeben. Am Ende der Anführung von Wörtern in Klammern, kann man noch mehr Informationen über dem Satz finden.

Den Inhalt eines Treebanks kann der Computer noch immer nicht verstehen. Dabei sind für den Programmierer diese Daten verständlich, deshalb kann man aus denen eine formale Grammatik erstellen. Sie beschreibt inhaltlich was der Treebank zeigt und ist ein Schritt näher und ähnlicher zu dem finalen Code. Man stellt zum Beispiel fest, dass das Alphabet als Buchstaben -> 'A' | 'a' | 'B' | 'b' | ... | 'Z' | 'z' gekennzeichnet ist und später zur Wortbildung dienen wird. So werden dann alle Teile detailliert gezeigt. Das wäre die strukturierte Vorbereitung für den Parser. Nachdem folgt das Schreiben des Codes. zu diesem Beispiel werden zugehörige Kommandoparameter angeführt, die in der Programmierungssprache Python geschrieben ist. Diese Programmierungssprache ist wegen

ihren Vorteilen, oft für die Sprachbearbeitung benutz. Für der Parsierung [7] gebaucht man im Python eine bestimmte Bibliothek- pyparsing. Mit einem Befehl *Buchstaben = pp.Word(pp.alphanums+":,.-").setName("Wort")* stellt man fest, dass eine Reihenfolge von Buchstaben ein Wort bildet. Dannach werden die Anführungszeichen definiert: *Anführungszeichen = pp.Literal('"')* und das wird in Anführungszeichen angegeben: *Wort = pp.Group(Anführungszeichen + Buchstaben + Anführungszeichen)*, weil es so in dem Treebank eingegeben ist. Die Spezifikation geht dann weiter bis die ganze Konstruktion des Satzes bearbeitet ist und die formale Grammatik in die Programmierungssprache übersetzt ist. Das Resultat des Programms wird dann mit allen Eigenschaften die Grundlage für die weitere Analyse bzw. die Sentiment- Analyse. Sie bedient sich mit dem Daten die sie durch die Parsierung bekommen hat und vergleicht sie mit den Eingaben im vorbereiteten Lexikon, wo man den Sentiment- Wert zu dem Wort finden kann.

4.3 Deep Learning und Data Mining

Deep Learning oder in die deutsche Sprache übersetzt „das tiefe Lernen“, ist eine Technik dem Computer etwas beizubringen, dass eigentlich die Menschen von Natur aus haben. Das bezieht sich auf die menschlichen Merkmale, die die Computer als Geräte nicht haben. Nach MathWorks ist Genauigkeit das wichtigste Merkmal, dass der Computer mit Hilfe von Deep Learning erreicht. Der Computer kann ganz neue Dimensionen von Arbeiten erledigen, wobei die Ergebnisse immer erstaunlicher sind. Der Computer ist jetzt fähig bestimmte Klassifikationen aus allen Medien- Bild, Video, Audio und Text durchzuführen. Ein menschliches Merkmal, dass die Computer von selbst nicht haben, ist das Erkennen des Sentiments. Wie schon erwähnt, ist es auch für die Menschen schwer die Gefühle der anderen Person zu erläutern. Der Computer, der selbst ein technisches Gerät und kein Lebewesen ist, hat keine Gefühle. Zum Ziel steht es der Sentiment- Analyse durch die vorbereiteten Werte die das Sentiment aufweisen, ein Programm erstellen, der die Aussage des Benutzers versteht, nicht nur inhaltlich, sondern auch der Meinung betreffend. Der Computer kann natürlich nicht „lernen“ etwas instinktiv zu erkennen, aber durch angemessene Informationen und Werte, kann man Programme mit richtigen Resultaten gestalten. (vgl. MathWorks: 1.9.2018)

Auf der anderen Seite, benutzt sich die Sentiment- Analyse mit dem Data Mining Prinzip. Die Aufgabe des Data Minings ist ein bestimmtes Muster zu erschaffen, das gewisse Fortschritte verspricht. Die Daten die durch das Date Mining erworben sind, kommen aus einer komplexer Datenmenge und benutzen sich für die zukünftige Arbeit. Wichtig ist dabei einen Unterschied in Acht zu haben, zwischen der Filtrierung von wichtigen Data und dem Verstehen von Data. Ein Data Mining

[7] Die Parsierung ist die Übersetzung von dem englischen Begriff Parsing und bezeichnet den Prozess der Überprüfung von Konstruktionen der Sätze nach der Sytax.

Programm kann immer noch nur mit Informationen und Werten umgehen, aber nicht alleine die Wichtigkeit der Daten erkennen. Im Zusammenhang mit Deep Learning, kann so ein System ein sehr weiterentwickeltes Programm sein. Am Beispiel der Sentiment- Analyse kann man sehen wie solche Prinzipien zu einem systematischen Vorteil führen. Durch das Deep Learning „lernt“ das Programm das Sentiment zu erkennen und so kann es mit Hilfe von Data Mining, die Daten die dem Unternehmen für die Förderung brauchen, erweitert werden. (vgl. Datenbanken verstehen für Anfänger und Profies: 1.9.2018)

5 Die Beispiele der Algorithmen für die Sentiment- Analyse

Am besten ist jede Theorie und Disziplin anhand eines Beispiels zu erklären. Die Algorithmen für die deutsche Sprache sind immer noch nicht so weit verbreitet, dass sie für den persönlichen Gebrauch frei und kostenlos sind, deshalb werden im diesem Kapitel zwei Algorithmen für die englische Sprache testiert. Die Sentiment- Analyse Algorithmen die bearbeitet werden sind die „Sentiment Analysis with Python NLTK Text Classification“ und die „Sentiment Text Analysis Software“ von Intentex. Da die Sentiment- Analyse Algorithmen am empfindlichsten auf Sarkasmus reagieren, wird die Effektivität von denen durch drei sarkastische Sätze getestet:

Satz1: „Sometimes I need what only you can provide: your absence.”

Satz2: “Mirrors can’t talk, lucky for you they can’t laugh either.”

Satz3: “You look good when your eyes are closed, but you look the best when my eyes closed.”

(Quote Ambition: 3.9.2018)

5.1.1 Die „Sentiment Analysis with Python NLTK Text Classification“

Dieser Algorithmus für die Sentiment- Analyse ist eine Demonstration der Sentiment-Analyse, die den Textklassifikationprozess NLTK 2.0.4 benutzt. Es klassifiziert den Text hierarchisch, zuerst kommt die Entscheidung, ob der Text neutral ist, danach wenn der Text nicht neutral ist, kommt die Klassifikation, ob er zu der positiven oder negativen Polarität gehört.

Bei Satz1 ist das Resultat positiv. Zuerst erkennt das System die Aussage als polar, also nicht neutral und dann kommt es zum weiteren Schritt, wo die Polarität gleich verteilt ist, also 50% positiv und 50% negativ. Der Algorithmus klassifiziert die Aussage jedoch als positiv. Wenn man aber den Inhalt des Satzes selbst analysiert, stellt man fest, dass die Aussage nicht positiv ist. Die sprechende Person will sagen, dass es besser ist, wenn die ansprechende Person nicht da ist, also kann dieser Satz nicht positiv auf die andere Person auswirken.

Der Satz2 kommt mit einem höheren Neutralwert aus, doch trotzdem ist er polar. Die Polarität zeigt ihn als 60% negativ und 40% positiv an. Eine Person könnte sagen, dass der Satz eindeutig negativ ist, doch es stellt sich hier die Frage ob der Algorithmus wegen der Negation automatisch die Aussage als negativ bezeichnet.

Der sarkastische Satz3 wird als polar und mit 70% positiv gekennzeichnet. Nach dem persönlichen Urteil wäre der Satz als negativ gesehen und wahrscheinlich nicht als physisches Kompliment angenommen.

Dieser Algorithmus ergibt nur primäre Resultate und obwohl die Datenbank aus Twitter stammen, erkennt es kein Sarkasmus. Wenn man die menschliche Schätzung zu den Resultaten stellen wurde, würden sich die Ergebnisse nicht überstimmen. (vgl. Sentiment Analysis with Python NLTK Text Classification: 3.9.2018)

5.1.2 Die „Sentiment Text Analysis Software“[8]

Anders als der vorige, ist der Algorithmus für die Sentiment- Analyse von Intentex etwas mehr ausarbeitet. Es analysiert mehr Bereiche und so bekommt man einen Polaritätsbericht mit mehr Details. Die persönliche Stellung, die Emotionen, die Ehrlichkeit, der Kommunikationsstil, die Zeit, die Motivation und der Ort bzw. das Wort im Satz, welches die Betonung trägt.

Im Satz1 ist die persönliche Stellung als negativ und schwach. Die Wörter Abwesenkeit und brauchen wirken dabei am meisten darauf. Die Worter im Satz die die Betonung tragen sind die Negationswörter. Dieser Algorithmus, im Unterschied zu dem Vorigen, sucht nach Schlusselwörtern die den Satz am meisten beschreiben. Das sind wie schon angegeben, die Negationswörter die alles was gesagt ist negieren, also in diesem Fall, die positive Aussage im ersten Teil des Satzes. Da der Algorithmus die Wörter gefunden hat, die die Negativität beschreiben, kommt es zu einem richtigen Resultat.

Der Satz2 bearbeitete das Programm nicht korrekt. Die meisten Merkamale blieben hier neutral, wobei die positive und negative Polaritäten den Wert von 67 Prozent bekammen. Die Wörter *sprechen* und *lachen* sortierten den Satz in dem audialen Kommunikationsstil und die Wörter lachen und glücklich deuten Glück an. Das Programm könnte das Übertreiben nicht erkennen und mit dem Sarkasmus verbinden.

Den Satz3 beschreibt das Programm als passiv, weil sich die Wörter *siehst* und *zu* wiederholen. Außerdem verbindet es ihn mit Glück und stellt das Wort du als Betonungswort. Bei dem

[8] deutsch „Die Software für die Sentiment- Analyse des Textes“

Betonungswort ist es ist schon die angesprochene Person betont, doch mehr betont ist eigentlich die Beziehung dein- mein.

Nach der Erkennung von Merkamlen, ist dieser Algorithmus für wissenschaftliche Arbeiten geschaffen. Es fordert den Nutzer mehr neutral und objektiv zu bleiben, deswegen ist es auch schwach bei der Detektierung von Sarkasmus. Der vorige Beispiel des Algorithmus ist auf der anderen Seite auf Grund Twitter Kommentare aufgebaut, doch obwohl die Beispielsätze auf dem sozialen Netzwerken vorkommen, erkannte der Algorithmus die wahre Bedeutung nicht. (vgl. Intentex: 3.9.2018)

5.2 Die Bereiche wo man die Sentiment- Analyse Algorithmen benutzt

Heutzutage ist das Internet das Hauptmedia. Im Internet kann man kaufen, verkaufen, Hotels reservieren, Restaurants nachschlagen usw. Da der Großteil der Population soziale Media wie Twitter, Facebook und Instagram benutzt, kann man sich mit seinem e-Identität auf viele Seiten anmelden und eine Rezension lassen. Durch diese erstehen dann große Mengen von Daten, wobei manche von denen einen großen Wert für ein Unternehmen haben. Die Kunden nehmen auch eine Rezension in Acht, bevor sie eine Dienstleistung benutzen, deshalb hat die Kundenkritik noch einen goßeren Wert für das Unternehmen. Ob der Kunde zufrieden ist oder nicht kann man aus den Komentaren feststellen, aber was wenn es scih um ein großes Unernehmen handelt, wofür dann tausende von Rezesionen stehen? Dafür gebraucht man die Sentiment- Analyse Algorithmen.

Um die Unternehmensleistung zu steigern verbindet man die Wirtschaftsinformatik mit der Computerlinguistik. Sie bedienen sich mit schon dargelegten Methoden der Sentiment- Analyse, also nach der Datensammlung, zergliedert das Programm die Sätze linguistisch, bis er zu kleinsten Einheiten kommt, die er weiter mit den Einträgen in der Datenbank vergleichen kann. Auf diese Weise verbindet er den Satz mit bestimmten Werten, die die Polarität des Satzes erläutern.

Diese Methode benutzt man in der Politik. Wenn man ein Einblick die Wahlprognose haben will, muss man durch das ganze soziale Netzwerk suchen, weil die Politik ein sehr aktuelles Thema ist, vor allem in der Zeit. Durch Facebook oder Twitter kann jeder seine Meinung sagen, aber auch die Meinung der anderen lesen. Auf diese Weise werden dann Politiker bewusst, worüber sie in der Öffentlichkeit sprechen mussen, um auf die Zielgruppe beeinzuflussen und die Stimmen zu bekommen. (vgl. Bernetbog: 4.9.2018)

Auf der anderen Seite ist der Verkauf. Auf Seiten wie zum Beispiel Amazon, hat die Sentiment-Analyse eine große Wirkung. Von den Dashboards der sozialen Netzwerke kommen die großen Datenmengen, die man analysieren muss um das Sentiment der Aussage einschätzen. Diese größte

Onlineversandhändler weltweit bietet Millionen und Millionen Produkte. Das Hauptmedia die sie für die Rezension benutzen ist Twitter. Auf diesem sozialen Netzwerk hat Amazon ein Konto, wo seine Angestellten in jeder Zeit auf Fragen beantworten. Daraus kann man viele nützliche Informationen rausfiltrieren. Die Sentiment- Analyse welche Produkte bei den Kunden gut ankommen und welche nicht, aber auch was sind die neuen Trends, um die zukünftige Produktion zu steigern. (vgl. Haep 2017: 75-77)

6 Schlussfolgerung

Die Sentiment- Analyse ist vor allem ein sehr komplexes Thema. Auch bei Leuten ist die Erkennung von Gefühl, Meinung und persönlicher Stellung empfindliches Gebiet. Die verbale und non- verbale Kommunikation zwischen zwei Personen die miteinander sprechen, kann uns auch nicht mit Sicherheit sagen, welches Sentiment hinter der Aussage der sprechender Person ist. Die Tränen bedeutet nicht immer etwas trauriges; sie können auch Freudestränen sein. So ist es auch mit einer Aussage, ob verbal oder non- verbal, gesprochen oder geschrieben. Die tiefere Bedeutung einer menschlichen Aussage ist schwer zu erkennen, auch für Menschen. Das gleiche dem Computer beizubringen ist also eine große Herausforderung.

Wenn man aber die Herausforderung aufnimmt, erwartet es viel Arbeit. Einen Algorithmus für die Sentiment- Analyse kann man ohne die linguistische Vorbereitung nicht aufbauen, deshalb ist es wichtig die Disziplinen der Linguistik gut zu beherrschen. Bei der Methode grabucht man Wissen über die Syntax und Lexikologie, zu dem inhaltichen Teil und der Analyse des Sentiments. Ohne die vorbereiteten sprachlichen Informationen kann der Programmierer keinen Code schreiben, weil das Programm keine Datenbank mit gebrauchten Daten hat. Das heißt dann, dass man für die Sentiment- Analyse Fachleute aus mehreren Berichen braucht, wobei jede Persone eine wichtige Aufgabe hat.

Das Ziel der Arbeit war das Prozess der Sentiment- Analyse aus verschiedenen Aspekten, die mit verschiedenen Gebieten verbunden sind, anzuschauen. Nach dem vorläufigen Kenntnis und Erfahrung aus dem Studium, waren hier auch die zwei Bereiche verbunden- die Informatik und Germanistik. Obwohl es auf dem ersten Blick unmöglich scheint, haben die Bereiche viel gemeinsam. Die Kommunikation ist hier der Schlüssel. Die Germanistik beschäftigt sich mit der natürlicher Sprache und die Informatik mit der formalen, doch wenn man die Sprachen nebeneinander stellt, haben sie das gleiche Ziel- die Kommunikation.

Es ist vor allem interessant, wie der Mensch immer neue Wege findet mit Computer zu kommunizieren, ihn etwas beizubrignen oder so zu steuern, dass der Computer die Menschenarbeit

erleichtert, wenn auch nicht übernimmt. Die Sentiment- Analyse kann den Menschen sehr behilflich sein und für viele Bereiche eine Forderung bedeuten, doch so eine komplexe Methode verlangt viel Arbeit. Jetzt befindet sich die Menschheit erst am Anfang der Sentiment- Analyse und ihrer Bearbeitung, deshalb kann man noch Vieles von dieser Methode in der Zukunft erwarten.

7.1 Quellenverzeichnis

1. Jason Chuang, Jean Wu, Richard Socher, Rukmani Ravisundaram und Tayyab Tariq(August 2013): Sentiment Analysis. https://nlp.stanford.edu/sentiment/index.html (21.4.2018)
2. Sentiment Analysis with Python NLTK Text Classification. http://text-processing.com/demo/sentiment/ (3.9.2018)
3. Martina Bürge(12. Jun 2013): Was ist eigentlich…: Social Sentiment Analysis?. https://bernet.ch/blog/2013/07/12/was-ist-eigentlich-social-sentiment-analysis/ (4.9.2018)
4. Universität Leipzig. Institut für Informatik: Digitale Informationsverarbeitung. http://www.informatik.uni-leipzig.de/~der/Vorlesungen/DIV/sprachen.pdf (12.5.2018)
5. IJCSI International Journal of Computer Science Issues, Volume 13, Issue 5 (September 2016): Unigram Polarity Estimation of Movie Reviews using Maximum Likelihood. http://www.ijcsi.org/papers/IJCSI-13-5-120-124.pdf (21.8.2018)
6. Waltinger, Ulli (11. November 2012): German Polarity Clues http://www.ulliwaltinger.de/ (21.8. 2018)
7. Tele- akademie. Paul Watzlawick - Die fünf Axiome der Kommunikation. http://www.ratgeber-tele-lernen.de/kommunikation/content/watzlawick_/ (25.8.2018)
8. Historisch-Kulturwissenschaftliche Informationsverarbeitung: Praktische Anwendung von Grammatiken für Compiler. http://hki.uni-koeln.de/archive/hki2016/wisem-2011/basisinformationstechnologie-i/theoretische-informatik-i/praktische-anwendung-von-grammatiken-fuer-compiler.html (15.8.2018)
9. Bauer, Anatoli (13. Oktober 2017): DIGITALE & ANALOGE KOMMUNIKATION – BEISPIEL AUS DEM ALLTAG https://uni-24.de/digitale-analoge-kommunikation-beispiel-aus-dem-alltag-tz24/ (23.8.2018)
10. Schanze, Robert (07. Februar 2017): Giga. Unterschied zwischen analog und digital? – Einfach erklärt https://www.giga.de/extra/ratgeber/specials/unterschied-zwischen-analog-und-digital-einfach-erklaert/ (23.8.2018)
11. Universitat Pompeu Fabra: IULA Penn Treebank https://repositori.upf.edu/handle/10230/20049 (29.8.2018)
12. MathWorks: Deep Learning. Drei Dinge, die Sie wissen sollten. https://ch.mathworks.com/de/discovery/deep-learning.html (1.9.2018)
13. Datenbanken verstehen für Anfänger und Profies: Data Mining – Was ist Data Mining? http://www.datenbanken-verstehen.de/business-intelligence/data-mining-grundlagen/data-mining/ (1.9.2018)
14. Natural Language Processing APIs and Python NLTK Demos: Python NLTK Sentiment Analysis with Text Classification Demo. http://text-processing.com/demo/sentiment/ (3.9.2018)
15. Intentex (2018): Sentiment Text Analysis Software. https://app.intencheck.com/analyze/ (3.9.2018)

16. Farooq, Umar; Mansoor, Hasan; Nongaillard,Antoine; Ouzrout, Yacine; Qadi, Muhammad Abdul (2016): ResearchGate. Negation Handling in Sentiment Analysis at Sentence Level. https://www.researchgate.net/publication/314424838_Negation_Handling_in_Sentiment_Analysis_at_Sentence_Level (6.9.2018)
17. Hutson, Matthew (24. Mai 2018): Science. How researchers are teaching AI to learn like a child. http://www.sciencemag.org/news/2018/05/how-researchers-are-teaching-ai-learn-child (12.9.2018)

7.2 Literaturverzeichnis

1. Srbljić, Siniša (2003): Jezični procesori 1. *Uvod u teoriju formalnih jezika, automata i gramatika.* Zagreb: Element
2. Srbljić, Siniša (2003): Jezični procesori 2. Analiza izvornog i sinteza ciljnog programa. Zagreb: Element
3. Duden. Deutsches Universalwörterbuch. 6., überarbetete und erweiterte Auflage 2007. Mannheim u. a.: Dudenverlag
4. Petrović, Velimir (2001): Einführung in die Linguistik für die Germanisten. Osijek: Sveučilište Josipa Jurja Strossmayera
5. Dovedan, Zdravko (2003): Formalni jezici: sintaksna analiza, Zagreb: Zavod za informacijske studije Odsjeka za informacijske znanosti
6. Haep, Sebastian (30. Mai 2017): MASTERTHESIS. Sentiment Analyse von informellen Kurztexten im Unternehmenskontext. Dortmund: Fachhochschule. TECHNISCHE HOCHSCHULE KÖLN
7. Watzlawick, Paul; Beavin, Janet H; Jackson, Don D (1996): Menschliche Kommunikation, Formen, Störungen, Paradoxien. Achte unveränderte Auflage. Bern, Stuttgart, Toronto: Verlag Hans Huber
8. Endres-Niggemeyer, Brigitte; Herrmann, Thomas; Kobsa, Alfred; Dietmar Rösner (1989): Interaktion und Kommunikation mit dem Computer: Jahrestagung der Gesellschaft für Linguistische Datenverarbeitung (GLDV). Ulm, 8.-10. März 1989 Proceedings: Haugender, Hans: Wie sag`ich`s dem Computer? oder How to Do Things? With Words! Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong: Springer-Verlag