
Sentimentanalyse mit FrameNet

ABSCHLUSSBERICHT

[GITLAB]

Autoren

Nadia Arslan

Sanaz Safdel

Semester

SS 2019

Supervisor

Éva Mújdricza-Maydt

15. März 2020
Heidelberg

Inhaltsverzeichnis

1	Einführung	1
2	Motivation	1
3	FrameNet	2
4	SentiFrameNet	2
4.1	Der Opinion-Frame	2
4.2	Fazit	6
5	Das SentiProjekt	6
5.1	Analyse von FrameNet	7
5.1.1	Semantic Types	7
5.1.2	Lexical Units	9
5.1.3	Frames	9
5.1.4	Zusammenfassung	13
5.2	SentiFrames	14
5.3	SentiFrameelemente und der Opinion-Frame	14
5.4	Polarität durch SentiWordNet	16
5.5	Veranschaulichung der Opinion-Frames	16
6	Fazit	18
7	Ausblick	18

Literaturverzeichnis

1 Einführung

Dieser Abschlussbericht bezieht sich auf das Referat „SentiFrameNet - Die Verbindung von Framesemantik und Sentimentanalyse “ und beschreibt eine Analyse von FrameNet II (Ruppenhofer et al., 2006) bezüglich dieser Verbindung. Zur Sentimentanalyse mit FrameNet wird eine mögliche Implementierung vorgestellt und deren Anwendung besprochen. Außerdem wird eine weitere Resource, SentiWordNet 3.0 (Baccianella et al., 2010), herangezogen und geprüft, ob diese zur Verbesserung bzw. Erweiterung von SentiFrameNet (Ruppenhofer, 2013) genutzt werden kann.

2 Motivation

Die flache Sentimentanalyse, wie sie zum Beispiel von Pang et al. (2002) beschrieben wird, schreibt einem Ausdruck ein positives oder negatives Sentiment zu. Manchmal wird noch ein drittes Label, das neutrale Sentiment, verwendet. So ist das Sentiment, das in Satz 1 ausgedrückt wird, ein positives:

1. Mike admires Elfie. (Mike bewundert Elfie)

Das Wort „admire “ ist in diesem Fall der Sentimentträger und gilt als positiv. Satz 1 hat also ein positives Sentiment.

Auf dieser flachen Ebene der Analyse, Satz für Satz (bzw. Ausdruck für Ausdruck) auszuwerten, können wir nicht sagen wer wie über wen fühlt oder denkt. Wir beantworten nur das wie. Um all diese W-Fragen zu beantworten, macht es Sinn ein Sentiment mit Semantischen Rollen zu verknüpfen. Jetzt können wir sagen, dass der Agent aus Satz 1 (Mike) ein positives Sentiment gegenüber dem Patient des Satzes (Elfie) hat. In diesem Fall ist das Objekt von Mikes Bewunderung ein Mensch und wir können sogar fragen, wie dieser Mensch es findet von Mike bewundert zu werden. Wir würden vielleicht denken, Elfie freut sich darüber, aber im Kontext von Satz 2 bekommt man eventuell Zweifel.

2. Elfie blames Mike. (Elfie beschuldigt Mike)

Das Wort „blame “ ist in diesem Fall der Sentimentträger und gilt als negativ. Der Agent aus Satz 2 (Elfie) hat also ein negatives Sentiment gegenüber dem Patient des Satzes (Mike).

Auch hier könnten wir fragen, wie es Mike wohl findet von Elfie beschuldigt zu werden. Wahrscheinlich wird er dies gar nicht gut finden, mit Sicherheit können wir das jedoch erst sagen, wenn wir den größeren Kontext kennen.

Ruppenhofer (2012) schlagen eine tiefere Analyse von Ausdrücken der Meinung vor, in denen auch die Semantischen Rollen und Aspekte eines Ausdruckes berücksichtigt werden sollen. Ihr Ziel ist es FrameNet mit Informationen zur tieferen Sentimentanalyse anzureichern. Sie nennen ihre Version SentiFrameNet.

3 FrameNet

FrameNet (Baker et al., 1998) besteht aus semantischen Frames, die als Abbild einzelner Konzepte oder Vorgänge in der Welt fungieren. Einem Frame sind mehrere Frameelemente zugeordnet, welche auch als die Semantischen Rollen des Frames bezeichnet werden können. Neben diesen finden sich in einem Frame auch lexikalische Einheiten, die Lexical Units, Wörter, die mit dem Frame in Verbindung stehen. Die lexikalische Datenbank des Projekts enthält um die 800 Frames, 10.000 Lexical Units und mehr als 120.000 annotierte Beispielsätze. Das Projekt ist im International Computer Science Institute (ICSI) in Berkeley beheimatet und wurde vor allem von Charles J. Fillmore konzipiert.

4 SentiFrameNet

Im folgenden Kapitel wird eine vorgeschlagene FrameNet-Ergänzungen vonuppenhofer (2012) erläutert. Dies ist der Opinion-Frame, der Frame der Meinung, der es ermöglichen soll, Meinungen oder Gefühle unter Bezug auf Ursprung und Ziel festzuhalten.

Der vonuppenhofer (2012) vorgestellte Factuality-Frame, der Frame der Faktizität, der es ermöglichen soll, Weltwissen oder allgemein gültiges Wissen in sich zu binden, um dann darauf Bezug nehmen zu können, findet in dieser Arbeit keine weitere Erläuterung. Auch auf die Darstellung vonuppenhofer (2012) zur Eventstruktur wird in dieser Arbeit nicht weiter eingegangen.

4.1 Der Opinion-Frame

uppenhofer (2012) führen einen neuen Frame, den Opinion-Frame (siehe Abbildung 1) ein, dessen Frameelemente Ursprung (Source), Ziel (Target) und Polarität (Positivität oder Negativität) eines Sentiments festhalten können. Außerdem gibt es ein Frameelement, um die Intensität (Intensity) eines Sentiments

zu beschreiben. Im weiteren findet dieses jedoch keine weitere Beachtung, da solch eine Bewertung eine Skalierung oder Normierung von Sentimentintensivitäten voraussetzt, die über diese Arbeit hinaus gehen.

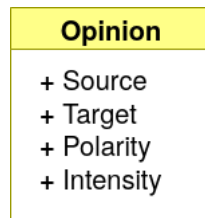


Abbildung 1: Der Opinion-Frame

Abbildung 2 zeigt die Sentimentträger aus Satz 1 und Satz 2, „admire“ und „blame“, die beide Lexical Units des Judgment-Frames sind.

Anhand von Constraints lassen sich Frameelemente unterschiedlicher Frames miteinander verbinden. Constraint (a) verbindet den Cognizer des Frames admire.Judgment mit der Source des Opinion-Frames. Constraint (b) verbindet den Evaluatee des Frames admire.Judgment mit dem Target des Opinion-Frames.

(a) `admire.Judgment.Cognizer = OpinionJudgment.Source`

(b) `admire.Judgment.Evaluatee = OpinionJudgment.Target`

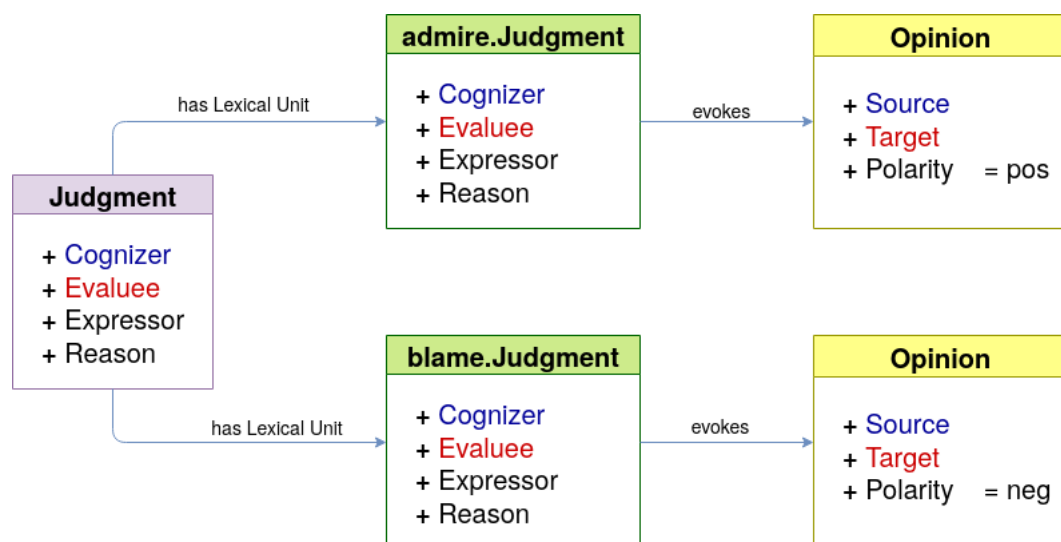


Abbildung 2: Frameanalyse von den Lexical Units admire.Judgment und blame.Judgment

(c) `admire.Judgment.Polarity = Opinionadmire.Polarity = positive`

(d) `blame.Judgment.Polarity = Opinionblame.Polarity = negative`

Die Ausprägung des Frameelements Polarity im Opinion-Frame ist positiv für admire und negativ für blame (siehe Constraints (c) und (d)).

Der Cognizer des Judgment-Frames hat im Falle von admire also eine positive Meinung von dem Evaluatee des Judgment-Frames, im Falle von blame eine negative.

An diesem Beispiel zeigt sich, dass Source und Target des Opinion-Frames mit Frameelementen des Judgment-Frames identifiziert werden können, dass die tatsächliche Evaluation des Sentiments aber von der jeweiligen Lexical Unit abhängt, die sich auf den Frame beziehen.

Anders verhält sich dies bei Satz 3 :

3. Will keeps bragging about his new bike.

Will gibt andauernd mit seinem neuen Fahrrad an.

Das Wort „brag“ steht im Bragging-Frame und trägt unterschiedliche Sentiments, je nachdem aus welcher Perspektive es betrachtet wird. Der Bragging-Frame enthält explizit Lexical Units, die eine Form von angeben darstellen.

Auf Grund der Definition des Bragging-Frames, gelten Polarity-Werte für den ganzen Frame und sind nicht mehr abhängig von den Sentiments der Lexical Units. Wer mit etwas angibt, findet diese Sache gut, hat also ein positives Sentiment. Abbildung 3 zeigt einen möglichen Opinion-Frame für Bragging. Zu diesem gehören Constraints (e) und (f).

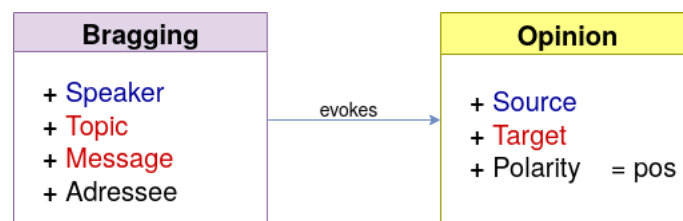


Abbildung 3: Der Bragging-Frame: Opinion 1

(e) `Bragging.Speaker = Opinion.Source`

(f) `Bragging.Topic | Bragging.Message = Opinion.Target`

Jemand, der das Angeben beobachtet, hat wahrscheinlich auch eine Meinung. Hier können verschiedene Szenarios angenommen werden. Man könnte annehmen, dass der Beobachter ein negatives Gefühl gegenüber dem Angeber hat. Dieses Gefühl kann sich aber auch auf die Sache beziehen, mit der angegeben wird oder sogar auf das Angeben selbst. Auch eine Kombination ist denkbar. Abbildung 4 zeigt diese möglichen Szenarios. Zu diesen gehören Constraints (g) bis (k).

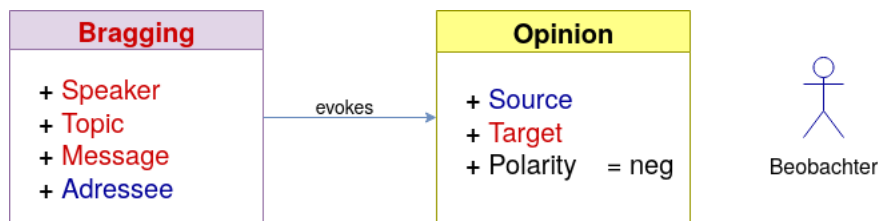


Abbildung 4: Der Bragging-Frame: Opinion 2

(g) `Bragging.Adressee | ?_s = Opinion.Source`

Konstraint (g) verbindet den Adressee des Bragging-Frames mit der Source eines Opinion-Frames. Es ist durchaus möglich, dass der Adressee des Angebens gleich der Beobachter ist. Denkbar wäre aber auch, dass es noch einen weiteren Beobachter gibt oder der Adressee wegfällt. Daraus folgt, dass die Source eines Opinion-Frames nicht immer gebunden sein muss, gleichwohl sie dennoch existiert. Ruppenhofer (2012) schlagen keine vollständige Notation dafür vor. Im Folgenden werden ungebundene Frameelemente mit „?_s“ (für ungebundene Source) und „?_t“ (für ungebundenes Target) benannt, um zu zeigen, dass sehr wohl eine Bindung zu einem unbekannten Element besteht.

(h) `Bragging.Speaker | Bragging.Topic | Bragging.Message = Opinion.Target`

(i) `?_t = Opinion.Target`

Konstraint (h) bindet die Frameelemente Speaker, Topic und Message des Bragging-Frames an das Target eines Opinion-Frames. Constraint (i) besagt, dass es im Bragging-Frame ein Element gibt, das das Target einer Meinung ist.

Eine Konstruktion, wie die von Constraint (j), erlaubt es, ganze Frames, oder auch nur einzelne Lexical Units an ein Element des Opinion-Frames zu binden. So könnte man sagen, dass jemand den Inhalt des ganzen Frames / der Lexical Unit bewertet.

(j) Bragging | Bragging.brag = Opinion.Target

Das Beispiel zeigt, dass der Bragging-Frame viele Frameelemente enthält, die das Target des negativen Opinion-Frames sein könnten. Mit Sicherheit lässt sich jedoch keines der Elemente zuordnen. Generell kann man aber sagen, dass Angeben eher eine negative Konnotation hat.

Um generelle Aussagen machen zu können, ist es sinnvoll noch eine weitere ungebundene Source, das „?_x“, einzuführen. Diese Variable wird gar nicht gebunden, sie kann vielmehr mit „generell“ übersetzt werden und wird im weiteren Projektverlauf noch veranschaulicht.

(k) ?_x = Opinion.Source

4.2 Fazit

Zusammenfassend kann man sagen, dass die Frameelemente eines Opinion-Frames, Source und Target, mehrfach an Frameelemente gebunden sein können, aber nicht gebunden sein müssen. Wenn Polarität vorhanden ist, lässt sie sich immer an Sentimentträgern (Lexical Units) bestimmen, kann aber auf ganze Frames übertragen werden. Ein Opinion-Event hat immer mehrere denkbare Opinion-Frames, die durch Constraints definiert werden können.

5 Das SentiProjekt

Das SentiProjekt beschreibt eine Analyse von FrameNet hinsichtlich der Sentimentanalyse und beschäftigt sich damit, Frames zu finden, die mit Opinion-Frames annotiert werden können, um dann die Sentiment tragenden Frameelemente dieser Frames zu identifizieren und durch Opinion-Konstraints miteinander zu verbinden. Um die Möglichkeiten zur Sentiment Analyse zu veranschau-

lichen, werden diese Constraints dann auf annotierte Beispielsätze aus FrameNet angewendet.

5.1 Analyse von FrameNet

Dieser Abschnitt gibt einen Überblick über die Elemente in FrameNet (Ruppenhofer et al., 2006), die zur tieferen Sentimentanalyse herangezogen werden können. Daten und Code befinden sich im Projektordner unter „FrameNetAnalyse/“.

5.1.1 Semantic Types

Manchen Frames, Frameelementen oder Lexical Units werden in FrameNet ein oder mehrere „Semantic Types“ zugeordnet. Ein Überblick aller Semantic Types in FrameNet gibt das Modul „/Code/get_semantic_types.py“. Es speichert alle vorhandenen Types dem Alphabet nach in der Textdatei „/Data/all_semantic_types.txt“.

Abbildung 5 zeigt die Anzahl der sieben häufigsten Semantic Types, mit denen Lexical Units annotiert sind. Beim Zählen wurde über alle Lexical Units in jedem Frame iteriert. Lexical Units, die in mehreren Frames stehen, wurden daher mehrfach gezählt.

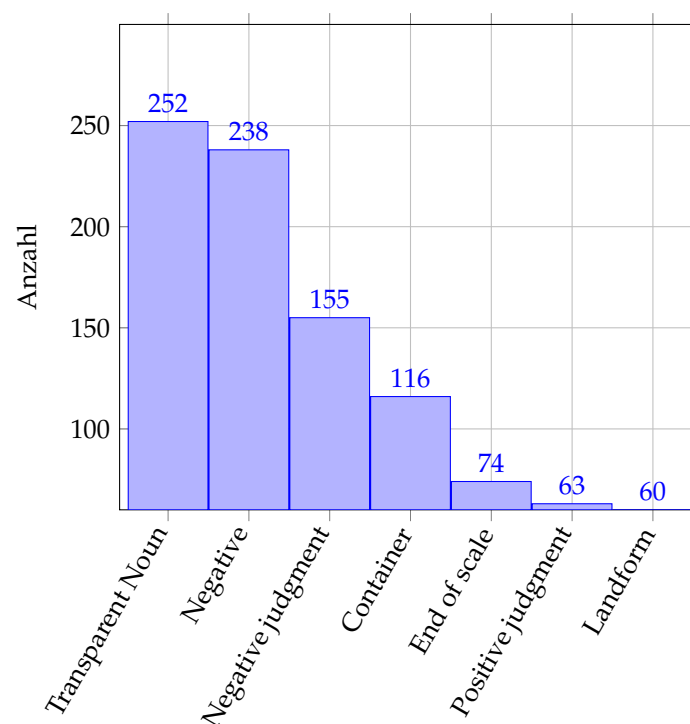


Abbildung 5: Verteilung Semantic Types in FrameNet

Die folgenden Semantic Types lassen sich für weitere Untersuchungen heranziehen:

Positive Judgment und Negative Judgment: Die semantischen Typen „Positive Judgment“ und „Negative Judgment“, können auf lexikalische Einheiten in einer Reihe von Frames angewendet werden, die die Meinung des Sprechers zu einer Situation anzeigen. Viele LUs beschreiben die positive oder negative Meinung eines impliziten oder expliziten Urteils, z.B. „like“ im Frame Experiencer_subj und „stingy“ im Frame Stinginess. Dieser Typ kennzeichnet solche LUs, damit sie trotz ihrer Unähnlichkeiten verglichen werden können.

Vom Typ Positive Judgment sind Lexical Units wie „praise.v“ im Frame Judgment, „like.v“ im Frame Experiencer_subj und „generous.a“ im Frame Frugality, vom Typ Negative Judgment die Lexical Units „criticize.v“, „hate.v“ und „stingy.a“

Negative: Lexical Units (LUs) werden mit dem Typ Negativ spezifiziert, wenn die Richtung der Abweichung in die negative Richtung geht. Z.B. Die LUs „cold“ im Frame Ambient_temperature oder „hate.v“ im Frame Experiencer_subject.

End Of Scale : Das „End Of Scale“ kennzeichnet LUs, die eine Instanz als einen maximalen Wert für ein Attribut charakterisieren. Solche LUs sind nicht verwendbar mit normalen Degree modifiers wie „very“. Z.B. „fabulous.a“ oder „bad.a“ im Frame Desirability.

Man kann an der Definition von Negativ und End of Scale schon sehen, dass diese Typen für LUs verwendet werden, deren frameale Semantik ein Degree-FE liefert. Alle solche LUs beschreiben ein bestimmtes Attribut einer Entität in einer bestimmten Richtung von dem Wert des Attributs.

Sentient und Human: Die Informationen von diesen Semantic Types sind in erster Linie dazu gedacht das Frame-Parsing und die automatische FE-Erkennung zu unterstützen. Diese Semantic Types sind dem Agenten zugeordnet, wie z.B. „person“, „individual“, „someone“, „somebody“, „mortal“ oder „human“. Das Wort Sentient kommt aus dem Lateinischen und bedeutet „Fühler“. Da die Eigenschaft zu fühlen oder eine Meinung zu haben die Frameelemente betrifft, die mit diesen Typen gekennzeichnet sind, eignen sie sich gerade für die Sentimentanalyse.

5.1.2 Lexical Units

Jedem Frame in FrameNet sind mehrere Lexical Units zugeordnet. Dies sind Worte, die in Verbindung mit dem Frame stehen.

Das Modul „/Code/get_lus_by_semtypes.py“ sucht alle Lexical Units, die zu einem der SentiSemantic Types gehören. Die 530 gefundenen Units sind in der json-Datei „/Data/lus_by_semtypes.json“ gespeichert und werden im Weiteren als Senti Lexical Units bezeichnet.

Tabelle 1 zeigt noch einmal explizit die Anzahl Lexical Units pro SentiSemantic Type.

SentiSemantic Type	Number of LUs
Positive Judgment	63
Negative Judgment	155
Negative	238
End Of Scale	74

Tabelle 1: Anzahl LUs pro SentiSemantic Type

5.1.3 Frames

Es gibt nur wenige Frames in FrameNet, denen einer der SentiSemantic Types zugeordnet wurde. Die Frames „Post lose possession“, „Pre getting“ und „Scarcity“ sind vom Semantic Type „Negative“, der Frame „Degree“ vom Type „End of scale“. Das Modul „/Code/get_frames_by_semType.py“ speichert diese Information unter „/Data/frames_by_semtypes.json“.

Abbildung 6 zeigt wie viele SentiLexicalUnits im jeweiligem Frame mit einem Semantic Type annotiert sind. Die Zahlen hierzu werden von dem Modul „/Code/get_frame_statistic.py“ ausgegeben.

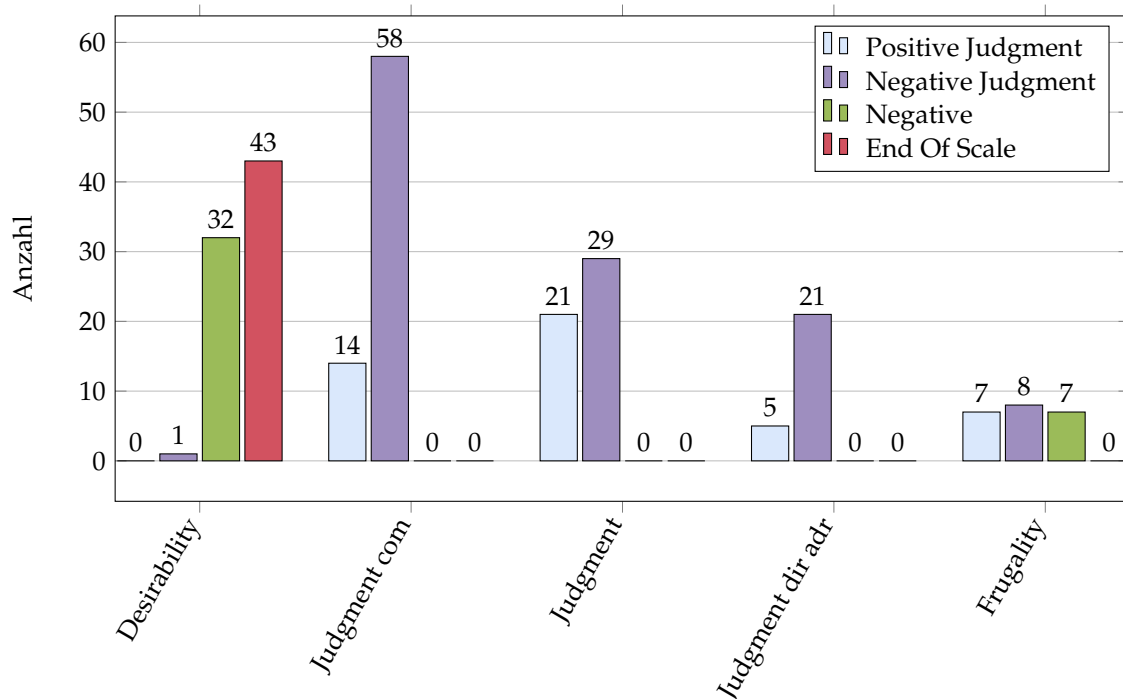


Abbildung 6: Häufigkeit Frames mit SentiSemantic Types annotierten LUs

Da das Wort Judgment (Beurteilung) sich hervorragend eignet um ein Opioneevent zu beschreiben und es in FrameNet Relation zwischen Frames gibt, gibt das Modul „/Code/get_judgment_frame_embedding.py“ die Judgment*-Frames in FrameNet aus, die zusammen eingebettet sind.

Unten werden die Frames und die dazugehörigen Frameelemente, die in Abbildung 7 sind definiert.

Judgment: Der Judgment-Frame ist geeignet, wenn ein Erkenner (Cognizer) eine Beurteilung über einen Evaluatee (ist die Person, die bewertet wird) hat. Diese Beurteilung kann positiv (z.B. respektieren) oder negativ (z.B. verurteilen) sein, und diese Information wird in den semantischen Typen Positiv oder Negativ Judgment auf den Lexikalischen Einheiten dieses Frames aufgezeichnet. Es kann einen bestimmten Grund (Reason) für die Beurteilung des Cognizers geben, oder es kann eine Fähigkeit oder Rolle vorhanden sein, in der der Evaluatee beurteilt wird. Anhand dieser Definition, und auch Abbildung 7, kann man sehen, dass die Frameelemente Cognizer, Evaluatee, Expressor und Reason für den Judgment-Frame definiert wurden. Frameelement Ausdruck (Expressor) ist das Körperteil oder die Handlung eines Körperteils, die das vom Cognizer getroffene Urteil vermittelt. Z.B. Sie betrachtete ihn mit einem dankbaren Blick. In diesem Beispiel wird Blick mit dem Frameelement Expressor annotiert.

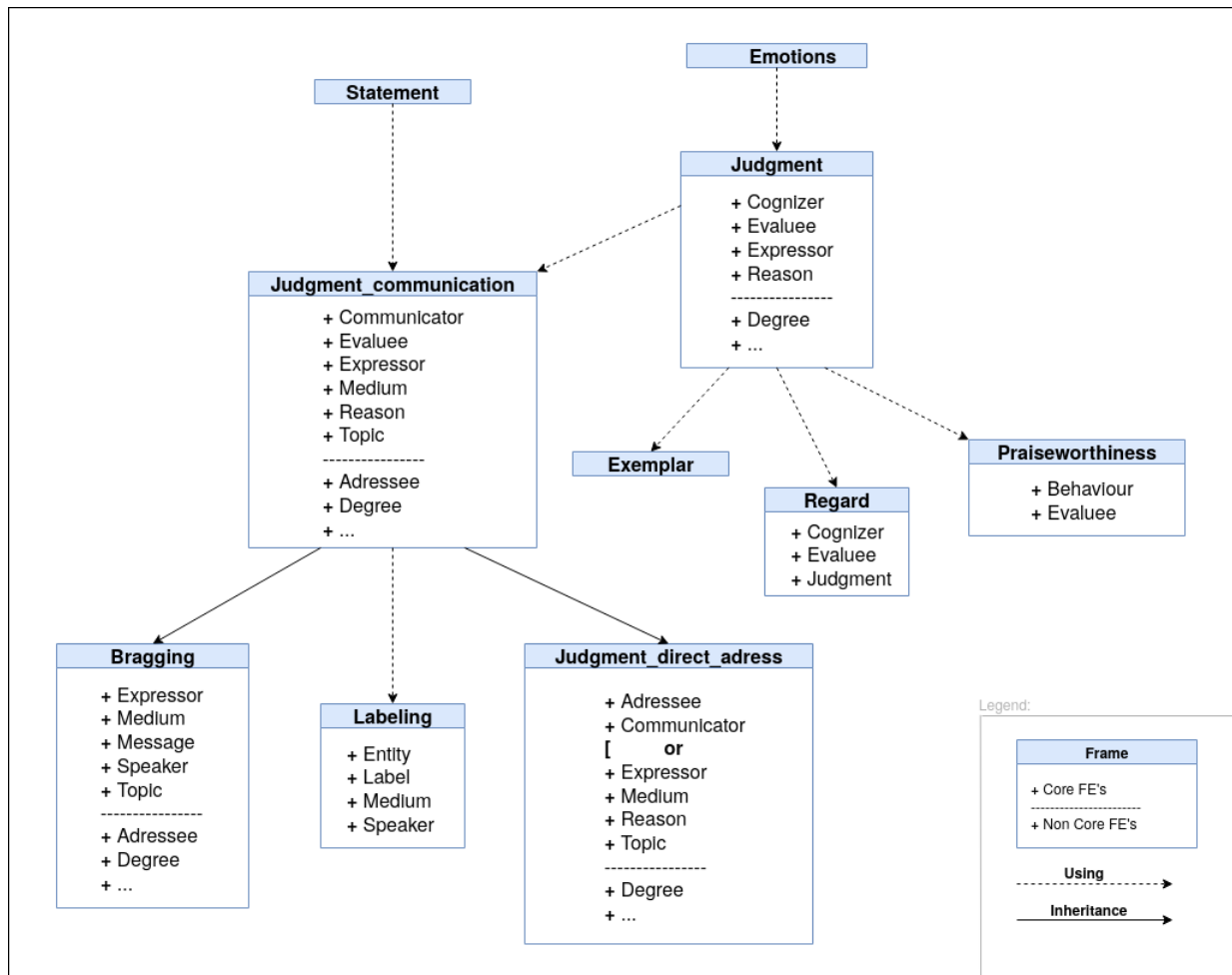


Abbildung 7: Frameembedding für Judgment*-Frames

Judgment_communication: In diesem Fall teilt ein Kommunikator (Communicator) einem Evaluatee eine Beurteilung über einen Wertgegenstand mit. Diese Beurteilung kann positiv (z.B. Lob) oder negativ (z.B. Kritik) sein, was durch die den lexikalischen Einheiten beigefügten semantischen Typen Positiv oder Negativ Judgment angezeigt wird.

4. I commended his handling of the crisis.

In Satz 4 zum Beispiel gehört Frameelement Communicator zu „I“, welcher die Sprache in schriftlicher oder gesprochener Form verwendet, um seine Bewertung über den Evaluatee, in diesem Fall „his handling of the crisis“ einer anderen Person zu vermitteln. Hier ist „commended“ Sentimentträger und teilt eine positive Meinung mit.

Judgment_direct_adress: In diesem Frame beurteilt ein Kommunikator (Communicator) den Empfänger (Addressee) und teilt diese Beurteilung dann direkt dem Empfänger (Addressee) mit. Die Beurteilung wird aus einem bestimmten Grund (Reason) oder zu einem bestimmten Thema (Topic) abgegeben.

Zu diesem Frame gehört das Frameelement Communicator, der teilt dem Addressee eine Beurteilung mit. Man kann in Satz 5 sehen, dass „The principal“ in der Rolle des Communicators ist, Addressee in diesem Beispiel „the student“ trifft, der wird durch den Communicator beurteilt und erhält die Nachricht der Zustimmung oder Ablehnung. Frameelement Reason gehört in Satz 5 zu dem Ausdruck „for cheating on the exam“. Topic ist das Thema, auf Grund dessen der Addressee beurteilt wird. Die Frameelemente Medium und Expressor gehören auch zu dem Judgment_direct_adress-Frame. Medium identifiziert die abstrakte Instanz, in der die Beurteilung vermittelt wird. Hier ist die Lexical Unit „rebuke.v“ Sentimentträger, welche den Judgment_direct_adress-Frame evoziert und die negative Meinung von „principal“ im Satz 5 ausdrückt.

5. The principal rebuked the student for cheating on the exam.

Labeling: Der Labeling-Frame ist geeignet, wenn ein Sprecher (Speaker) in einem Ausdruck eine Bezeichnung (Label) verwendet, um auf eine Instanz (Entity) zu referieren. Die Bezeichnung (Label) beschreibt die Entität und drückt eine Beurteilung aus. Die Frameelemente Speaker, Label, Entity und Medium gehören zu diesem Frame.

6. Some labeled him a womanizer.

Im Satz 6 wird das Wort „some“ mit Frameelement Speaker annotiert, der das Label „womanizer“ für Entity „him“ verwendet hat, um ihn zu beurteilen.

Bragging: Der Bragging-Frame enthält die Frameelemente Sprecher (Speaker), wenn der Sprecher spricht über ein Thema (Topic), das eine Handlung des Sprechers oder eine Entität oder ein Sachverhalt sein kann. Alternativ kann

ein Nachricht (Message) Frameelement verwendet werden, um direkt zu beschreiben, was der Sprecher sagt. In diesem Fall können anstatt des Sprechers die Frameelemente Medium oder Expressor zugeordnet werden. Das Frameelement Empfänger (Addressee) gehört auch zu dem Bragging-Frame.

Regard: Dieser Frame wird verwendet, wenn ein Erkenner (Cognizer) eine Beurteilung (Judgment) über einen Evaluatee hat, die zum Ausdruck bringt, wie hoch oder niedrig die Wertschätzung des Bewerter ist. Hier gehören die Frameelemente Cognizer, Evaluatee und Judgment zu dem Frame Regard. In Satz 7 wurde das Wort „what“ in diesem Kontext mit Frameelement Beurteilung (Judgment), „you“ mit Erkenner (Cognizer), „of him“ mit dem Evaluatee annotiert.

7. What do you think of him as a linguist?

Praiseworthiness: Der Praiseworthiness-Frame hat die Funktion, wenn das Verhalten (Behavior) eines Evaluatees in Form der Reaktion von jemandem beschrieben wird, wie „praise“, „blame“ oder „admiration“. Die Wörter Verhalten (Behavior) und Evaluatee sind Frameelemente vom Praiseworthiness-Frame. Z.B. in Satz 8 wird Anhand der Definition von diesem Frame „of you“ mit Frameelement Evaluatee und „to help that person“ mit Frameelement Verhalten (Behavior) annotiert.

8. It was honorable of you to help that person.

5.1.4 Zusammenfassung

Zusammenfassend kann man sagen, dass es in FrameNet Semantische Typen gibt, die für die Sentimentanalyse benutzt werden können. Diese kennzeichnen sowohl Lexikalische Einheiten (z.B. positive Judgment), als auch Frameelemente (z.B. Sentient). Daraus ergibt sich, dass es auch zur Sentimentanalyse geeignete Lexical Units und Frameelemente in FrameNet gibt. Auch ganze Frames eignen sich, so kann man sagen, dass z.B. alle Judgment*-Frames eine Meinung beschreiben können. In manchen Frames können positive, negative oder beide Meinungen beschrieben sein.

5.2 SentiFrames

Als SentiFrames bezeichnen wir Frames in FrameNet, die mit Opinion-Frames annotiert werden können. Einige geeignete Frames wurden bereits benannt. Darunter waren die Judgment*-Frames, der Bragging-Frame und der Regard-Frame. Ein Ziel des Projektes ist es diese SentiFrames zu identifizieren. Sie wurden zum einen während des Projektverlaufs ergänzt, zum anderen wurde nach SentiFrames mit der Endung „-ing“ gesucht. Die Idee dahinter ist, zunächst „verbale“ Frames, wie z.B. Bragging, zu finden, um dann in einem zweiten Schritt verwandte Frames in FrameNet via Frame-Relationen zu suchen. Eine Liste der 46 gefundenen SentiFrames findet sich unter „/Data/Senti_Frames.txt“.

Am Beispiel von Satz 3 wurde festgestellt, dass es Frames gibt (z.B. der Bragging-Frame), die nach ihrer Definition implizit Sentiment in sich tragen. Eine Liste der 25 SentiFrames mit impliziter Frame-Polarität findet sich unter „/Data/Senti_Frames_mit_Framepolarity.txt“.

In dieser Liste befinden sich zum Beispiel „Fear“, „Trendiness“, „Desiring“ oder „Warning“.

Eine Liste der 21 SentiFrames ohne impliziter Frame-Polarität findet sich unter „/Data/Senti_Frames_ohne_Framepolarity.txt“. Bei diesen Frames muss die Polarität an den jeweiligen Lexical Units aufgelöst werden.

In dieser Liste befinden sich unter anderem die Judgment*-Frames.

5.3 SentiFrameelemente und der Opinion-Frame

Um die SentiFrames mit ihren möglichen Opinion-Frames zu verbinden, wurden für jeden SentiFrame Constraints erstellt. Diese Constraints folgen einer anderen Notation als in Kapitel 4, denn sie bilden die Source-Target-Beziehungen ausgesuchter Frameelemente innerhalb eines SentiFrames ab. Diese Notation wurde der von FrameNet angepasst.

Ein Constraint ist ein Tripel, (Source, Polarity, Target), und erlaubt es zwei Elemente mit einer Relation zu verlinken. Es sind verschiedene Relationen denkbar, diese Arbeit beschränkt sich auf die Relation Polarität, welche zwei Werte, positiv und negativ, annehmen kann. Constraint (1) gehört beispielsweise zum Bragging-Frame:

(1) <Source=Bragging.Speaker - Polarity=pos -> Target=Bragging.Topic>

Das Constraint besagt, dass der Speaker des Bragging-Frames eine positive Haltung zum Topic des Bragging-Frames hat.

Konstraint (m) kommt aus dem Judgment-Frame und besagt, dass der Cognizer eine Haltung gegenüber dem Topic des Frames hat. Ob diese Meinung positiv oder negativ ist, bleibt vorerst offen.

(m) <Source=Judgment.Cognizer - Polarity=?_p -> Target=Judgment.Topic>

Da die Semantik der SentiFrames (insbesondere jene mit Frame-Polarität) und deren Frameelemente unterschiedlich sind, war es nicht möglich alle Constraints automatisch zu generieren.

Zunächst wurden alle Frameelemente aller SentiFrames ermittelt, die die Source einer Meinung sein können. Da die Eigenschaft der Meinungshabung nur auf Menschen oder ähnliches zutrifft, wurden zunächst alle Frameelemente mit dem Semantic Type „Sentient“ oder „Human“ ausgesucht. Die Liste wurde ergänzt durch wenige weitere Elemente wie z.B. „Survivor“ aus dem Surviving-Frame. All diese Frameelemente können auch das Target einer Meinung sein. Hinzu kommen nun auch alle Frameelemente über die man eine Meinung haben kann. Die Liste dieser Elemente wurde weitestgehend per Hand erstellt. Die vollständigen Listen finden sich unter „/Data/source_elements.txt“ und „/Data/target_elements.txt“ .

Das Modul „/Code/mk_constraints.py“ erstellt dann alle möglichen Constraints eines Frames. Dazu wird jede mögliche Source einmal mit jeder anderen Source und einmal mit jedem möglichen Target verknüpft. Außerdem werden ungebundene Constraints hinzugefügt, die es ermöglichen sollen implizite Elemente auszudrücken. Diese Elemente werden durch „?_s“ oder „?_x“ für eine ungebundene Source und „?_t“ für ein ungebundenes Target dargestellt. Die Polarität wird mit „?_p“ angegeben und bleibt zunächst ungebunden.

Konstraint (n) bis (p) aus dem Bragging-Frame sagen zum Beispiel aus, dass der Speaker des Frames Etwas toll findet (n), dass Jemand das Topic des Frames toll findet (o) und dass Angeben generell etwas Schlechtes ist (p).

(n) <Source=Bragging.Speaker - Polarity=pos -> Target=?_t>

(o) <Source=?_s - Polarity=pos -> Target=Bragging.Topic>

(p) <Source=?_x - Polarity=neg -> Target=Bragging>

Die automatisch erstellten Constraints wurden im Nachhinein überarbeitet. Hierbei wurden die Polaritätswerte der Frames mit Frame-Polarität gesetzt und

unsinnige Constraints gelöscht. Die Constraints aller SentiFrames sind einsehbar unter „/Data/ Constraints/ “.

5.4 Polarität durch SentiWordNet

Da es in FrameNet keine durchgehende Annotation der Lexical Units bezüglich Polarität gibt, wurde die Ressource SentiWordNet 3.0 (Baccianella et al., 2010) herangezogen.

SentiWordNet besteht aus einer Liste von Synsets und schreibt jedem dieser Synsets einen positiven, einen negativen und einen neutralen Score zu. Dabei stellt ein Synset eine Menge von Wörtern mit gleicher Bedeutung dar. Ein Wort mit mehreren Bedeutungen kann daher in mehreren Synsets auftauchen.

Um die Polarität der Lexical Units aus SentiFrames ohne Framepolarität zu bestimmen, wurde ein Modul geschrieben, das das gesuchte Wort in SentiWordNet findet und entweder als positiv, negativ oder neutral einordnet. Hierzu wurden die positiven und negativen Scores aller gefundenen Synsets addiert und das Label mit dem höheren Score (bei Gleichstand das Label neutral) zurückgegeben.

5.5 Veranschaulichung der Opinion-Frames

Um veranschaulichend zu testen, wie gut die geschriebenen Constraints funktionieren, wurden annotierte Sätze aus FrameNet bestimmt, welche sich zur Sentimentanalyse eignen. Dabei wurde darauf geachtet, dass die Sätze mit einem SentiFrame annotiert sind und mindestens drei annotierte Frameelemente aufweisen.

Das Modul „/Code/get_annotated_senti_sentences_from_fn.py“ filtert solche Sätze und speichert sie unter „/Data/annotated_sents_from_FraNet.txt“.

Das Modul „/Code/map_constraints_to_annotation.py“ geht jeden dieser Sätze durch und erstellt, mithilfe der geschriebenen Constraints, ein Opinion-Mapping. Die Ergebnisse sind zu finden unter „/Data/Opinion_mapping/“.

Das Modul „/Code/try_me.py“ gibt einen interaktiven Einblick über die Ergebnisse, die dieses Projekt hervorgebracht hat.

Abbildung 8 und 9 zeigen beispielhaft eine leicht veränderte Ausgabe des Moduls „/Code/try_me.py“ für „brag.v“ und „admire.v“. Im oberen Teil der Ausgabe werden die Informationen zum Satz angezeigt, darunter sind alle Constraints aufgelistet, die dieser Satz anspricht. Einmal das Originalkonstraint und einmal das auf die Frameelemente des Satzes gemappte Constraint.

```
#####
lu: brag.v
frame: Bragging
text: Luke bragged to friends : ' All I want is to kill a copper .
Target: bragged
FE: { 'Speaker': 'Luke', 'Addressee': 'to friends',
      'Message': '' All I want is to kill a copper .' }
=====
1.) ?_x neg Bragging
   _GENERAL — negative -> Bragging
-----
2.) Bragging.Addressee neg Bragging.Speaker
   to friends — negative -> Luke
-----
3.) Bragging.Speaker pos Bragging.Message
   Luke — positive -> ' All I want is to kill a copper .
-----
4.) ?_s neg Bragging.Speaker
   _SOMEONE — negative -> Luke
-----
5.) ?_s pos Bragging.Message
   _SOMEONE — positive -> ' All I want is to kill a copper .
-----
6.) Bragging.Speaker pos ?_t
   Luke — positive -> _SOMETHING
#####
```

Abbildung 8: Ein mögliches Opinion-Mapping für brag.v

```
#####
lu: admire.v
frame: Judgment
text: Sometimes Mike admired Elfie for her virtue.
Target: admired
FE: { 'Cognizer': 'Mike', 'Evalued': 'Elfie', 'Reason': 'for her virtue' }
=====
1.) ?_x ?_p Judgment.?_lu
   _GENERAL — positive -> Judgment.admire.v
-----
2.) Judgment.Cognizer ?_p Judgment.Evalued
   Mike — positive -> Elfie
-----
3.) Judgment.Cognizer ?_p Judgment.Reason
   I — positive -> for her virtue
-----
4.) Judgment.Cognizer ?_p ?_t
   Mike — positive -> _SOMETHING
-----
5.) ?_s ?_p Judgment.Reason
   _SOMEONE — positive -> for her virtue
-----
6.) ?_s ?_p Judgment.Evalued
   _SOMEONE — positive -> Elfie
#####
```

Abbildung 9: Ein mögliches Opinion-Mapping für admire.v

6 Fazit

Zusammenfassend lässt sich sagen, dass die erstellten Opinion-Mappings in den meisten Fällen zutreffend sind. Constraint 2.) in Abbildung 8 zeigt, dass es Constraints im Datensatz gibt, die Aussagen treffen, denen man nicht mit Sicherheit zustimmen kann. Es kann durchaus sein, dass seine Freunde Luke gut für seinen Ausruf finden, auch wenn er damit angibt. Das ist abhängig von dem Inhalt des Ausrufs und der Meinung von Mike's Freunden dazu. Um sichere Aussagen zu treffen, fehlt hier mehr Kontext.

Abbildung 9 hat zum Beispiel kein Constraint, dass Elfies Haltung, die des Evaluatee, beschreibt. Dass sie wohl eine Meinung hat, haben wir schon in Kapitel 2 vermutet, aber wie diese Meinung ist, kann man nach wie vor erst im weiteren Kontext sagen.

Die Annotation der Polarität der Lexical Units mit Hilfe von SentiWordNet hat weites gehend gut funktioniert, obwohl keine Wortdisambiguierung vorgenommen wurde. Dem ein oder anderen Wort wird ein anderes Sentiment zugeordnet als erwartet, so wird zum Beispiel „critic.n“ als positiv gelabelt, was nicht falsch ist, jedoch nicht in jedem Kontext stimmt. Ebenso verhält es sich mit „cool.a“, welches positiv gelabelt wird, aber, je nach Kontext, auch eine negative Haltung beschreiben kann.

7 Ausblick

Es gibt viele Möglichkeiten die beschriebene Arbeit auszubauen. Zum einen ist die Liste der SentiFrames ist noch nicht vollständig. Dies könnte man noch verbessern, welches auch das Schreiben weiterer Constraints nach sich ziehen würde.

Zum anderen kann zum Beispiel das Opinion-Frameelement „Intensity“, welches hier keine Beachtung fand, mit diversen vorhandenen Frameelementen (Degree oder Manner) oder auch ganzen Frames (Judgment_of_intensity) in FrameNet in Verbindung gebracht werden. Auch wäre denkbar die Scores aus SentiWordNet zu benutzen, um eine Art Skalierung abzuleiten.

In FrameNet gibt es nicht nur Framerelationen, es gibt auch Relationen zwischen Frameelementen verschiedener Frames. Dies wäre ein anderer Ansatzpunkt der Erweiterung. So könnte man zum Beispiel prüfen, ob sich Sentiments von Frame zu Frame übertragen lassen. Die Frames Attack und Defending sind beispielhaft für solch eine Erweiterung.

Was bisher völlig außer Acht gelassen wurde, ist die Temporalität, der zeitli-

che Aspekt, von Aussagen. Eine Theorie zur Einbettung in Opinion-Eventstrukturen würde es vielleicht ermöglichen, zeitliche Abläufe zu modellieren und Sentiment auch Kontext übergreifend binden zu können.

Literaturverzeichnis

Baccianella, S., A. Esuli, and F. Sebastiani

2010. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, volume 10, Pp. 2200–2204.

Baker, C. F., C. J. Fillmore, and J. B. Lowe

1998. The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, Pp. 86–90. Association for Computational Linguistics.

Pang, B., L. Lee, and S. Vaithyanathan

2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, Pp. 79–86. Association for Computational Linguistics.

Ruppenhofer, Josef und Rehbein, I.

2012. Semantic frames as an anchor representation for sentiment analysis. In *Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis*, Pp. 104–109. Association for Computational Linguistics.

Ruppenhofer, J.

2013. Extending framenet for sentiment analysis. *Veredas-Revista de Estudos Linguísticos*, 17.

Ruppenhofer, J., M. Ellsworth, M. Schwarzer-Petruck, C. R. Johnson, and J. Scheffczyk

2006. Framenet ii: Extended theory and practice.