



**Universität  
Zürich<sup>UZH</sup>**

*Institut für Computerlinguistik*

# Aspektextraktion in domänenspezifischen Rezensionen

Bachelorarbeit der Philosophischen Fakultät der Universität  
Zürich

Referent: Prof. Dr. M. Volk

Betreuer: Dr. M. Klenner

Verfasserin:

Martina Bachmann

Matrikelnummer 11-706-769

9. Juli 2014

## Zusammenfassung

Die Aspektextraktion ist ein wichtiger Bestandteil der aspektbasierten Sentimentanalyse und ist eines der Themen des *International Workshop on Semantic Evaluation 2014*. Ein *Aspekt* ist ein Begriff, der Teil einer Entität ist, für die anhand der Bewertungen aller ihrer Aspekte die Polarität ermittelt wird. Im Rahmen meiner Bachelorarbeit habe ich mich damit beschäftigt ein System zu entwickeln, das Aspekte in Rezensionen identifizieren kann. Die dabei verwendeten Rezensionen stammen von zwei verschiedenen Domänen: Restaurant und Laptop. Das Aspektextraktionssystem basiert auf der Idee, dass ein Aspekt jeweils dem einer bewerteten Nominalphrase innerhalb einer Rezension entspricht. Für das System habe ich zwei verschiedene Methoden getestet: Eine Methode erkennt Nominalphrasen nur dann als Aspektterme, wenn sich ein polares Wort direkt auf die Phrase bezieht; die zweite Methode identifiziert Phrasen als Aspekte, sobald im selben Satz ein polares Wort auftritt. Nebst diesem regelbasierten System, habe ich zusätzlich ein maschinelles Lernverfahren gewählt, welches wie ein Filter funktioniert. Es baut auf das Extraktionssystem auf und eliminiert vom System fälschlicherweise als Aspektterme identifizierte Phrasen. Die Evaluation hat gezeigt, dass die zweite Methode besser funktioniert als die erste. Des Weiteren hat das maschinelle Lernverfahren die Leistung des Systems verbessern können. Das System funktioniert insgesamt mit den Restaurantrezensionen besser als mit den Laptoprezensionen. In Anbetracht der Tatsache, dass für die Aspektextraktion eine einfache Idee verwendet wird, ist die Leistung des Systems relativ gut.

# Danksagung

Ich möchte mich bei allen, die mich während meiner Bachelorarbeit tatkräftig unterstützt und motiviert haben, recht herzlich bedanken.

Mein grösster Dank gilt meinem Betreuer Herrn Dr. Manfred Klenner für seine Bereitschaft, mich bei meiner Bachelorarbeit zu unterstützen. Ich möchte mich bei ihm für seine Hilfe und seine konstruktiven Anregungen bedanken, denn ich habe es immer sehr geschätzt, dass er mir jederzeit mit gutem Rat zur Seite gestanden hat.

Ebenfalls möchte ich mich bei Herrn Prof. Dr. Martin Volk bedanken, der sich als Referent zur Verfügung gestellt hat. Bedanken möchte ich mich an dieser Stelle auch ganz herzlich bei Nora Hollenstein für die gute Zusammenarbeit im Rahmen des SemEval Workshops und für ihre stetige Unterstützung und Hilfe im Zusammenhang mit der Sentimentanalyse-Pipeline und dem SemEval-Projekt.

Des Weiteren möchte ich mich bei den Veranstaltern des SemEval Workshops 2014 für die Organisation und die Zurverfügungstellung der Daten bedanken.

# Inhaltsverzeichnis

<b>Zusammenfassung</b>	<b>i</b>
<b>Danksagung</b>	<b>ii</b>
<b>Inhaltsverzeichnis</b>	<b>iii</b>
<b>Abbildungsverzeichnis</b>	<b>v</b>
<b>Tabellenverzeichnis</b>	<b>vi</b>
<b>Abkürzungen</b>	<b>vii</b>
<b>1 Einleitung</b>	<b>1</b>
<b>2 Aktueller Forschungsstand</b>	<b>4</b>
2.1 Extraktion basierend auf häufigen Nominalphrasen . . . . .	4
2.2 Extraktion basierend auf Relationen zwischen Meinungen und Aspekten	5
2.3 Extraktion basierend auf supervisiertem Lernen . . . . .	6
2.4 Extraktion basierend auf Topic Modelling . . . . .	7
<b>3 SemEval-Daten</b>	<b>8</b>
<b>4 Domänenspezifische Erweiterung der Pipeline</b>	<b>9</b>
4.1 Aufbau der Pipeline . . . . .	9
4.2 Lexikonerweiterung . . . . .	11
4.3 Verbmodellierung . . . . .	13
4.4 Auswirkung der Ressourcenerweiterungen . . . . .	14
<b>5 Aspektextraktion</b>	<b>17</b>
5.1 Aufbau des Extraktionssystems . . . . .	17
5.1.1 Methode 1: Polar markierte Nominalphrasen . . . . .	20
5.1.2 Methode 2: Nominalphrasen in polaren Sätzen . . . . .	21
5.2 Erweiterung durch ein maschinelles Lernverfahren . . . . .	22
<b>6 Evaluation</b>	<b>25</b>

6.1	Evaluationsmethode . . . . .	25
6.2	Resultate und Diskussion . . . . .	26
<b>7</b>	<b>Schlussfolgerung</b>	<b>31</b>
	<b>Bibliografie</b>	<b>33</b>
	<b>Selbständigkeitserklärung</b>	<b>35</b>
<b>A</b>	<b>Skripte</b>	<b>36</b>

# Abbildungsverzeichnis

1	Eintrag aus dem SemEval-Datenset . . . . .	8
2	Mögliche Dependenzmuster von Kompositakomponenten . . . . .	18
3	Dependenzen von Nominalphrasen mit Präpositionalkomponenten . .	19
4	Outputbeispiel des Aspektextraktionssystems . . . . .	20

# Tabellenverzeichnis

1	Polaritätstags im Lexikon . . . . .	12
2	Genauigkeit des SimpleLogistic Klassifikators . . . . .	16
3	Extrahierte Merkmale eines Aspekts pro Satz . . . . .	23
4	Resultate für das Restauranttestset mit dem regelbasierten System .	26
5	Resultate für das Laptotestset mit dem regelbasierten System . . . .	27
6	Resultate für das Restauranttestset mit dem maschinellen Lernver- fahren . . . . .	27
7	Resultate für das Laptotestset mit dem maschinellen Lernverfahren .	28

# Abkürzungen

CRF	Conditional Random Field
HMM	Hidden Markov Model
LDA	Latent Dirichlet Allocation
pLSA	Probabilistic Latent Semantic Analysis
PMI	Pointwise Mutual Information
SemEval	International Workshop on Semantic Evaluation
VISL	Visual Interactive Syntax Learning
XML	eXtensible Markup Language



# 1. Einleitung

Meinungen anderer haben einen hohen Stellenwert in der heutigen Gesellschaft. Im Zeitalter des Internets kann man ohne grossen Aufwand verschiedenste Dinge bewerten und seine Ansichten dazu kundgeben. Tagtäglich werden online unzählige Rezensionen über die neusten Filme, Bücher, Videospiele, Autos oder Computer verfasst. Solche Rezensionen sind vor allem aus wirtschaftlicher Sicht wichtig: Firmen interessieren sich zunehmend dafür, auf eine schnelle und effiziente Art und Weise an Kundenrückmeldungen zu ihren Produkten und Dienstleistungen zu gelangen, um diese gegebenenfalls den Bedürfnissen der Kunden anzupassen. Die sogenannte *Sentimentanalyse* ermöglicht ihnen dies. Die Sentimentanalyse ist ein Forschungsgebiet, welches sich damit beschäftigt, Meinungen, Stimmungen, Gefühle und Bewertungen zu analysieren, die sich unter anderem auf Produkte, Services, Organisationen, Individuen und Events beziehen [Liu, 2012, 7]. Mit Hilfe einer solchen Analyse lassen sich subjektive Informationen in Form von positiven und negativen Äusserungen in einem Text erkennen, was wiederum Rückschlüsse auf das Beschriebene zulässt.

Aufgrund des steigenden Interesses an der Sentimentanalyse beschäftigen sich zunehmend mehr Forscher mit diesem Thema. Auch am diesjährigen *International Workshop on Semantic Evaluation (SemEval)*<sup>1</sup>, bei dem jährlich verschiedene Aufgabenstellungen zum Thema Semantik veröffentlicht werden, war die Sentimentanalyse vertreten. Im Rahmen meiner Bachelorarbeit habe ich mich mit Aufgabe 4 befasst, welche das Thema *aspektbasierte Sentimentanalyse* behandelt<sup>2</sup>. Als Aspekt wird hier ein Begriff bezeichnet, der Teil einer übergeordneten Entität ist und – falls bewertet – zur Gesamtbewertung dieser Entität beiträgt. Liu [2012, 68] unterscheidet zwischen expliziten und impliziten Aspekten: Im Satz *das Frühstücksbuffet ist ausgezeichnet*, der so in einer Restaurantrezension stehen könnte, ist beispielsweise das *Restaurant* die Entität und das *Frühstücksbuffet* ein explizit genannter Aspekt und somit ein Teil der Entität. Hingegen der Satz *es war teuer* enthält eine implizite Referenz zu *Preis*, nicht aber eine explizite Nennung. Bei SemEval werden solche impliziten Referenzen nicht als Aspekte sondern als Aspektkategorien bezeichnet. Für die Aufgabe 4 des diesjährigen SemEval-Workshops sind Restaurant- und Lap-

---

<sup>1</sup>SemEval 2014: <http://alt.qcri.org/semeval2014/>

<sup>2</sup>Die Aufgabe 4 habe ich unter anderem in Zusammenarbeit mit Nora Hollenstein gelöst.

toprezensionen zur Verfügung gestellt worden. Die Aufgabe besteht aus vier Teilaufgaben: In der ersten Teilaufgabe ist es das Ziel, alle expliziten Aspekte zum Thema Restaurant und Laptop zu extrahieren; in der zweiten, die Polarität dieser Aspekte zu ermitteln. Die dritte Teilaufgabe verlangt, vordefinierte Aspektkategorien (*Preis, Service, Essen, Ambiente, Anekdote/Diverses*) zu identifizieren und die vierte wiederum, deren Polarität ausfindig zu machen. Aspektkategorien sind nur fünf vorhanden und sie sind somit eingeschränkter als die Aspekte. Mit Hilfe der identifizierten Aspekte lässt sich auf die Kategorien schliessen: *Fish* und *fruit* entsprechen beispielsweise der Kategorie *Essen*. Implizite Referenzen in einem Satz wie *es war teuer (Preis)* zählen ebenfalls zu den Aspektkategorien.

Für meine Bachelorarbeit habe ich mich vor allem mit Teilaufgabe 1, der Aspektextraktion, beschäftigt. Wie Liu [2012, 67] schreibt, bezieht sich eine Bewertung immer auf ein bestimmtes Objekt oder Thema, was gewöhnlich dem gesuchten Aspekt entspricht. Diese These setze ich auch in meiner Bachelorarbeit um: Ich gehe davon aus, dass in domänenspezifischen Texten wie Restaurant- und Laptoprezensionen die bewerteten Begriffe Aspektterminen entsprechen. Daraus schliesse ich, dass Nominalphrasen, auf die sich polare, d.h. positive oder negative, Begriffe beziehen, potentielle Aspektterme<sup>3</sup> sind. Um solche Aspekte zu erkennen, mache ich Gebrauch einer am Institut für Computerlinguistik in Zürich entwickelten Pipeline, die nebst linguistischer Analyse ein Polaritätslexikon und eine Verbressource verwendet. Meine Ziele für die Bachelorarbeit lauten demnach folgendermassen:

- Domänenspezifische Erweiterung des Polaritätslexikons und der Verbressource, um den Output der Pipeline zu verbessern
- Entwicklung eines regelbasierten Programmes zur Extraktion von Aspekten
- Erweiterung des Extraktionsprogrammes durch ein maschinelles Lernverfahren, welches als Filter für das Extraktionssystem verwendet wird

In meiner Arbeit gebe ich in einem theoretischen Teil in Kapitel 2 zuerst einen Überblick über den aktuellen Forschungsstand der aspektbasierten Sentimentanalyse. In Kapitel 3 präsentiere ich kurz die von SemEval zur Verfügung gestellten Daten, die ich in meiner Arbeit verwende. Anschliessend gehe ich in Kapitel 4 genauer auf die verwendete Pipeline ein und zeige auf, wie ich das Polaritätslexikon und die Verbressource erweitert habe, wie dadurch die Pipeline verbessert wird und inwiefern dies Auswirkungen auf die Aspekttermextraktion hat. In Kapitel 5 erläutere

---

<sup>3</sup>Wenn die Nominalphrase ein Kompositum (teilweise kombiniert mit einer Präpositionalphrase) ist, wird die ganze Phrase als Aspekt identifiziert; enthält die Phrase andere attributive Komponenten wie Adjektive, wird nur der Kopf der Phrase - das Substantiv oder das Kompositum - als Aspekt erkannt. Im Folgenden verwende ich den Begriff Nominalphrase für potentielle Aspektterme und beziehe mich damit jeweils auf beide Fälle.

ich mein Extraktionssystem und vergleiche die Leistung mit der SemEval-Baseline. Des Weiteren präsentiere ich ein maschinelles Lernverfahren, das auf dem Extraktionssystem aufbaut und evaluiere meinen Ansatz in Kapitel 6, bevor ich in Kapitel 7 ein Fazit ziehe.

## 2. Aktueller Forschungsstand

In den letzten Jahren haben sich einige Forscher mit aspektbasierter Sentimentanalyse beschäftigt und verschiedene Methoden zur Extraktion von Aspekten entwickelt. Dabei haben sich vier verschiedene Ansätze herauskristallisiert [Liu, 2012, 67]:

- (i) Die Ermittlung von Aspekten mit Hilfe von häufigen Nominalphrasen
- (ii) Die Ermittlung von Aspekten anhand von Relationen zwischen Bewertungen und Aspekten
- (iii) Die Ermittlung von Aspekten durch supervisiertes Lernen
- (iv) Die Ermittlung von Aspekten mit Hilfe von Topic Modelling

In den folgenden Unterkapiteln gehe ich genauer auf diese Ansätze ein und präsentiere die neuesten Entwicklungen dazu.

### 2.1. Extraktion basierend auf häufigen Nominalphrasen

Dieser Ansatz beschäftigt sich ausschliesslich mit explizit im Text erwähnten Aspekten [Liu, 2012, 68]. Hu und Liu [2004] haben eine Methode entwickelt, die mit Hilfe eines Wortarten-Taggers Nominalphrasen erkennt und deren Vorkommen zählt. Sie gehen dabei davon aus, dass Verfasser von Bewertungen ein ähnliches Vokabular verwenden, wenn sie über ein bestimmtes Thema schreiben. Dieses Vokabular entspricht den gesuchten Aspekttermen. Somit gilt: Je häufiger eine Nominalphrase erscheint, desto grösser ist die Wahrscheinlichkeit, dass sie ein Aspektterm ist. Hu und Liu's Algorithmus ignoriert demnach selten auftretende Nominalphrasen. Laut Liu [2012, 69] ist diese Methode zwar trivial, aber dennoch effizient. Für die SemEval-Aufgabe ist sie jedoch weniger effektiv, da besonders in den Restauranttexten viele verschiedene Gerichte erwähnt werden, aber nur wenige davon oft vorkommen.

Popescu und Etzioni [2005, 2] haben diesen Algorithmus weiterentwickelt: Für alle gefundenen Nominalphrasen wird die sogenannte *Pointwise Mutual Information*

(*PMI*) berechnet. Das heisst, es wird für jede Nominalphrase im Web überprüft, wie oft sie als Meronym mit der gegebenen Klasse vorkommt. Angenommen, die gegebene Klasse lautet *computer*, so wird ermittelt wie oft eine Nominalphrase in Verbindungen wie *computer has*, *of computer* oder *is a computer* erscheint. Das PMI Mass wird wie folgt berechnet [Popescu und Etzioni, 2005, 10]:

$$PMI(f, d) = \frac{Hits(d + f)}{Hits(d) * Hits(f)} \quad (2.1)$$

Das  $f$  steht hier für die Nominalphrase und  $d$  für eine obengenannte Verbindung. Je seltener eine Nominalphrase mit einer solchen Verbindung auftritt, desto kleiner wird der PMI Wert, was wiederum bedeutet, dass die Nominalphrase mit grosser Wahrscheinlichkeit kein Aspektterm ist [Liu, 2012, 70]. Mittels PMI können demnach gewisse Nominalphrasen als Aspektterme ausgeschlossen werden.

Eine weitere auf Nominalphrasen basierende Methode präsentieren Blair-Goldensohn et al. [2008]. Sie filtern wie Hu und Liu häufige Nominalphrasen heraus; diese dürfen jedoch aus maximal drei Wörtern bestehen und müssen in Sätzen, die eine positive oder negative Bewertung beinhalten, vorkommen oder in bestimmten syntaktischen Mustern auftreten [Blair-Goldensohn et al., 2008, 5]. Das heisst, eine Nominalphrase wird beispielsweise extrahiert, wenn sie auf ein Adjektiv folgt. In meiner Arbeit verfolge ich einen ähnlichen Ansatz wie Blair-Goldensohn et al., indem ich mich auf die Phrasen konzentriere, die in Sätzen mit subjektivem Inhalt auftreten.

Während die meisten Ansätze auf der absoluten Anzahl von Vorkommen einer Nominalphrase innerhalb eines Textes basieren, identifizieren Ku et al. [2006] Aspekte mit Hilfe des tf-idf-Mass. Sie nehmen an, dass ein Begriff, dann wichtig ist, wenn er nur in wenigen Paragraphen von vielen Dokumenten erscheint oder wenn er in vielen Paragraphen innerhalb von wenigen Dokumenten auftritt. Ist dies der Fall, ist der Begriff ein potentieller Aspektterm.

## 2.2. Extraktion basierend auf Relationen zwischen Meinungen und Aspekten

Dieser Ansatz macht Gebrauch von Wörtern, die eine positive oder negative Einstellung repräsentieren, um Aspekte zu identifizieren. Der Ansatz funktioniert, weil sich solche meinungsbezogenen Wörter normalerweise auf ein bestimmtes Objekt

beziehen, das meist einem Aspekt entspricht [Liu, 2012, 71]. Hu und Liu beispielsweise haben mit diesem Ansatz eine Methode entwickelt, die auch selten auftretende Aspektterme erkennt [Liu, 2012, 71]. Diese Methode identifiziert Sätze, in denen kein häufig auftretender Aspektterm gefunden wurde, die jedoch ein meinungsbezogenes Wort oder eine Phrase enthalten. Da Hu und Liu [2004, 3] davon ausgehen, dass Verfasser von Rezensionen nur wichtige Aspekte bewerten, vermuten sie, dass in solchen Sätzen auch Aspekte vorhanden sind, da eine Bewertung vorliegt. Ihre Methode extrahiert somit aus einem Satz mittels Abhängigkeitsrelationen eine Nominalphrase – der vermutete Aspektterm – auf die sich ein polares Wort bezieht.

Liu [2012, 71] weist darauf hin, dass gerade bei solchen Ansätzen ein Abhängigkeitsparser äusserst hilfreich sein kann, da er die Abhängigkeiten von ganzen Phrasen und nicht nur von einzelnen Wörtern ermittelt. Weil Aspekte Nominalphrasen mit mehreren Konstituenten sein können, wie zum Beispiel Komposita, ist ein solcher Parser für die Ermittlung der Relation zwischen einem polaren Wort und einem Aspekt geeignet. Aus diesem Grund machen einige Forscher [Kobayashi et al., 2006; Somasundaran und Wiebe, 2009] für die Identifizierung von Aspekten Gebrauch von Phrasenabhängigkeitsparsern.

### 2.3. Extraktion basierend auf supervisiertem Lernen

Supervisiertes Lernen scheint ein effektiverer Ansatz zu sein, da er häufig für die Informationsextraktion gewählt wird. Bei supervisierten Lernverfahren werden annotierte Daten benötigt, die für das Training verwendet werden. Gemäss Liu [2012, 72] bauen die meisten supervisierten Ansätze für die Aspektextraktion auf „*sequential learning*“ Methoden auf. Zu solchen Lernmethoden zählen unter anderem das *Hidden Markov Model (HMM)* oder das *Conditional Random Field (CRF)* Modell. Jin und Ho [2009] haben beispielsweise ein lexikalisches HMM, welches anhand von gelernten Mustern Aspektterme extrahiert, auf Kamerarezensionen angewendet. Das lexikalisierte Modell ist insofern anders als das herkömmliche HMM, weil es lexikalische Merkmale eingliedert. Dabei wird nicht nur die Wortart sondern auch die Wortform vorhergehender Wörter in Betracht gezogen. Auch Jakob und Gurevych [2010] haben einen supervisierten Ansatz entwickelt, bei dem sie eine CRF-basierte Methode verwenden. In ihrer Studie haben sie Texte aus verschiedenen Domänen – Film, Auto, Kamera und Webservice – verwendet, um so ein möglichst domänenunabhängiges System zu generieren. Auch für das Training haben Jakob und Gurevych [2010, 1037-1038] domänenunabhängige Merkmale verwendet: Dazu gehören unter anderem Wortarten, syntaktische Abhängigkeiten und Wortdistanzen.

## 2.4. Extraktion basierend auf Topic Modelling

Ein weiterer Ansatz basiert auf sogenannten *Topic Models*. Liu [2012, 73] definiert diese wie folgt:

Topic modelling is an unsupervised learning method that assumes each document consists of a mixture of topics and each topic is a probability distribution over words. A topic model is basically a document generative model which specifies a probabilistic procedure by which documents can be generated. The output of topic modelling is a set of word clusters.

Solche von Liu beschriebene Topic Models können als Sammlung von potentiellen Aspekten betrachtet werden. Beispiele von Verfahren, welche Topic Models berechnen, sind *Probabilistic Latent Semantic Analysis (pLSA)* und *Latent Dirichlet Allocation (LDA)*.

Topic Models weisen in Hinsicht auf die Aspektextraktion jedoch einige Nachteile auf. Gemäss Titov und McDonald [2008] basiert LDA auf der Verteilung von Topics innerhalb von verschiedenen Dokumenten, was bei Rezensionen zu einem bestimmten Thema – beispielsweise Laptops – wenig Sinn macht, da alle Rezensionen einen ähnlichen Inhalt aufweisen und über verschiedene Aspekte von Laptops berichten. Ein weiterer Nachteil ist die grosse Datenmenge, die nötig ist, um einen brauchbaren Output zu erzielen [Liu, 2012, 76]. Eine weitere Einschränkung liegt darin, dass mit Topic Models innerhalb eines Dokumentes häufig erwähnte Aspekte zwar gefunden werden, solche die in der Dokumentensammlung jedoch selten vorkommen, werden meist nicht identifiziert. Nichtsdestotrotz weist Liu [2012, 76] daraufhin, dass Topic Modelling ein «powerful and flexible modeling tool» ist, das sich – wenn weiter daran geforscht wird – als effizient für die Aspektextraktion erweisen wird.

Mein Aspektextraktionssystem baut insbesondere auf die in Kapitel 2.2 erläuterte Idee der Beziehung zwischen Meinungen und Aspekten auf. Des Weiteren ergänze ich das System mit einem supervisierten Lernverfahren, welches einem weiteren von Liu [2012] beschriebenen Ansatz entspricht. Topic Models sowie die Idee der Verwendung von häufigen Nominalphrasen hingegen sind nicht Teil meines Extraktionssystems.

### 3. SemEval-Daten

Die Organisatoren des SemEval-Workshops haben für die Aufgabe 4 Datensets aus zwei verschiedenen Domänen zur Verfügung gestellt: Das eine enthält Bewertungen von Restaurants, das andere besteht aus Laptoprezensionen. Zu Beginn des Workshops war für die Teilnehmer nur ein Trainingsset verfügbar; später haben die Veranstalter von SemEval ein Testset für die Evaluierung veröffentlicht (siehe Kapitel 6). Das Restaurantdatenset stammt von Ganu et al. [2009], die im Rahmen eines Sentimentanalyse-Projektes ein englisches Korpus aus mehreren tausend Rezensionen von *Citysearch New York* extrahiert haben. Das Trainingsset von Aufgabe 4 enthält rund 3000 Sätzen aus diesem Korpus. Auch das Laptoptrainingsset besteht aus etwas mehr als 3000 Sätzen, welche aus laptopbezogenen, englischen Kundenbewertungen extrahiert wurden.

Die Trainingssets liegen im XML-Format vor. Jeder Satz ist manuell von Experten mit den vorhandenen Aspekttermen und Aspektkategorien sowie deren Polaritäten annotiert worden. Im Gegensatz zum Restaurantdatenset enthält das Laptopdatenset keine Aspektkategorien. Diese benötige ich für mein Aspektextraktionsprogramm jedoch nicht. In den rund 3000 Trainingsätzen des Restaurantdatensets sind 1288 verschiedene Aspektterme vorhanden und insgesamt 3693 Aspektterme getaggt worden. Bei den Laptoptrainingsdaten erscheinen 1042 unterschiedliche Aspektterme und 2358 solche Terme sind annotiert worden. Abbildung 1 zeigt ein Beispiel eines annotierten Satzes aus dem Restaurantdatenset.

```
<sentence id="1429">
  <text>Sauce was watery and the food didn't have much flavor.</text>
  <aspectTerms>
    <aspectTerm term="Sauce" polarity="negative" from="0" to="5"/>
    <aspectTerm term="food" polarity="negative" from="25" to="29"/>
    <aspectTerm term="flavor" polarity="negative" from="47" to="53"/>
  </aspectTerms>
  <aspectCategories>
    <aspectCategory category="food" polarity="negative"/>
  </aspectCategories>
</sentence>
```

Abbildung 1.: Annotierter Beispielsatz aus dem Restauranttrainingsset



## 4. Domänenspezifische Erweiterung der Pipeline

Dieses Kapitel zeigt genauer auf, welche Ressourcen ich für die Aspektextraktion verwende und wie ich sie für diesen Zweck domänenspezifisch erweitert habe. In Kapitel 4.1 erläutere ich den Aufbau der Pipeline, die am Institut für Computerlinguistik in Zürich für die englische Sentimentanalyse entwickelt wurde und die ich nun für die Aspektextraktion benutze. Anschliessend thematisiere ich in Kapitel 4.2 und 4.3 die domänenspezifischen Anpassungen des Polaritätslexikons und der Verbressource bevor ich in Kapitel 4.4 die Auswirkungen dieser Änderungen präsentiere.

### 4.1. Aufbau der Pipeline

Die Pipeline bereitet grundsätzlich englische Texte so auf, dass sie anschliessend für die Sentimentanalyse weiterverwendet werden können. In einem ersten Schritt, wird der Input – in der SemEval-Aufgabe handelt es sich um einzelne Sätze – mit dem ClearNLP Dependenzparser<sup>1</sup> verarbeitet. Dabei wird der Input tokenisiert, mit den jeweiligen Wortarten getaggt, morphologisch analysiert und mit Dependenzen und semantischen Rollen angereichert.

In einem zweiten Schritt wird der Output des Dependenzparsers in das sogenannte *VISL*-Format umgewandelt. *VISL*<sup>2</sup> (*Visual Interactive Syntax Learning*) ist ein Forschungsprojekt, welches am Institut von Sprache und Kommunikation an der Universität von Süddänemark entstanden ist. Im Rahmen dieses Projektes wurde ein Parser namens *VISL-CG3* für Constraint-Grammatiken entwickelt. Dieser Parser erwartet als Input Text in einem sogenannten *Kohorten*-Format. Eine Kohorte setzt sich aus einer «Wortform zwischen spitzen Klammern [...] und den nachfolgenden zugehörigen Analysezeilen» zusammen [Rios und Clematide, 2010, 2]. Die Analyse-

---

<sup>1</sup>ClearNLP Dependenzparser: <http://clearnlp.wikispaces.com/>

<sup>2</sup>VISL: <http://beta.visl.sdu.dk/>

zeilen wiederum bestehen aus morphologischen und syntaktischen Tags. Beispiel 4.1 veranschaulicht das Kohorten-Format.

(4.1) ”<Restaurant>”

”Restaurant” Restaurant NN NEUT NOM SG

”Restaurant” Restaurant NN NEUT AKK SG

”Restaurant” Restaurant NN NEUT DAT SG

Der Output des ClearNLP Abhängigkeitsparsers wird, wie bereits erwähnt, in ein solches Kohorten-Format konvertiert. Normalerweise wird der VISL-CG3 Parser dazu verwendet, zu disambiguieren, welche Analysezeile im gegebenen Kontext die gewünschte ist. Da mit dem Abhängigkeitsparser jedoch bereits die linguistische Analyse durchgeführt worden ist und die passenden Analysekomponenten ermittelt worden sind, wird in jeder Kohorte nur jeweils eine Analysezeile aufgeführt. Das heißt, dass der Disambiguierungsschritt bereits mit dem Abhängigkeitsparser geschieht und die Analysen somit nur noch in das Kohorten-Format umgeformt werden müssen.

VISL-CG3 ermöglicht es, Regeln zu schreiben, die – basierend auf Kontextinformationen oder Relationen – auf spezifische Kohorten angewendet werden können. In der UZH-Pipeline für die Sentimentanalyse werden vier verschiedene Regeln auf den Input angewendet:

1. Alle Wörter, die im Input auftreten und im Polaritätslexikon vorhanden sind, werden markiert. (Das Polaritätslexikon wird in Kapitel 4.2 genauer erläutert)
2. Wenn Wörter wie *nie* oder *ohne* – sogenannte *Shifters*, welche die gegenteilige Polarität indizieren – auftreten, wird die gegenwärtige Polarität des Wortes, auf das sich der Shifter bezieht, invertiert.
3. Für jede Phrase wird anhand der polaren Kohorten, die durch Regel 1 markiert wurden, nach kompositionalem Prinzip [Moilanen und Pulman, 2007] die Polarität ermittelt und auf den Phrasenkopf übertragen.
4. Wenn Verben, die eine Polarität beim Subjekt, Objekt oder beim Komplement indizieren, im Input auftreten, wird die Kohorte des jeweiligen Subjekts, Objekts oder Komplements dementsprechend markiert. (Die Verbressource wird in Kapitel 4.3 genauer erläutert)

Die Polaritätsmarkierungen, die durch die Anwendung der oben genannten Regeln in Kraft treten, werden direkt in die Analysezeile integriert. Beispiel 4.2 zeigt eine Phrase mit modifizierten Kohorten. Das Wort *great* ist ein polares Wort und somit im Polaritätslexikon enthalten. Die Analysezeile wird dementsprechend laut Regel 1 mit einer Markierung (*A\_POS*) versehen. Dies hat einen positiven Effekt auf den

Kopf dieser Phrase, weshalb die Kohorte von *computer* mit *POS-NP-POLCHUNK-HEAD* markiert wird.

```
(4.2) "<great>"
      "great" great C_JJ JJ A_POS POS-NP-POLCHUNK-CHILD
      ADD:247:mark_child_pos #15->16
      "<computer>"
      "computer" computer C_NN NN POS-NP-POLCHUNK-HEAD #16->8
```

Da mein Programm zur Extraktion von Aspekten diesen VISL-Output verwendet, habe ich die Pipeline über alle Sätze aus den Restaurant- und Laptopdatensets laufen lassen. Anschliessend habe ich pro Satz den Output als neuer Knoten den SemEval-XML-Dateien hinzugefügt.

## 4.2. Lexikonerweiterung

Das Polaritätslexikon ist, wie in Kapitel 4.1 beschrieben, Teil der Sentimentanalyse-Pipeline für das Englische. Es besteht aus einer Liste von rund 4800 Wörtern, die Polarität, d.h. negative oder positive Gefühle oder Fakten, indizieren. Die Liste besteht aus den häufigsten polaren Adjektiven, Adverbien und Substantiven, die manuell annotiert wurden.

```
(4.3) gorgeously, A_POS,1,,adverb
      abusive,J_NEG,1,1,adjective
      unbelievably,INT,,adverb
```

Wie in Beispiel 4.3 ersichtlich, besteht jeder Eintrag aus den folgenden Komponenten: An erster Stelle steht das Lemma, gefolgt von einem Polaritätstag. Des Weiteren erscheint an dritter Stelle im Eintrag die Zahl 1, falls das Wort stark polar ist. An vierter Position ist mit der Zahl 1 das Aktiv und mit der Zahl 2 das Passiv markiert. An letzter Stelle folgt die jeweilige Wortart. Die dritte und vierte Position werden vor allem dann annotiert, wenn das Wort offensichtlich stark polar, aktiv oder passiv ist; ansonsten können diese Positionen auch leer bleiben.

Das im Eintrag an zweiter Stelle positionierte Polaritätstag kann verschiedene Formen annehmen. Es wird zwischen drei Hauptgruppen unterschieden: Shifters (SHI), intensivierende oder abschwächende Tags (INT,DIM) oder positive und negative Tags (POS,NEG). Als Shifter gekennzeichnete Wörter invertieren die Polarität ihres Bezugswortes. *Nor*, *never* und *absence* sind Beispiele solcher Shifters aus dem Polaritätslexikon. Intensivierende Wörter wie *increase* oder *absolutely* verstärken die Polarität in einem Satz, abschwächende Wörter hingegen führen zu einer Ab-

schwächung der Polarität, wie dies bei *hardly* und *limited* der Fall ist.

Der grösste Teil der Lexikoneinträge weist jedoch ein positives oder negatives Tag auf. Diese Tags werden in drei verschiedene Unterkategorien unterteilt, die Martin und White [2003] in ihrer Appraisaltheorie definieren: Affect (F), Judgement (J) und Appreciation (A). Affect umfasst Wörter, die Gefühle wie beispielsweise Schock oder Freude ausdrücken [Martin und White, 2003, 35,42]. Judgement hingegen bezieht sich auf diejenigen Wörter, die Verhalten bewerten, das wir unter anderem bewundern, verachten oder loben [Martin und White, 2003, 42]. Hier kommt insbesondere die moralische Ebene ins Spiel: Wörter der Kategorie Judgement bewerten, ob ein Verhalten im Rahmen einer gegebenen Norm liegt oder davon abweicht [Martin und White, 2003, 35]. Die dritte Unterkategorie, Appreciation, beinhaltet Wörter, die weder emotional noch moralisch begründet Dinge bewerten. Dazu zählt beispielsweise *gutes Wetter* oder *Krankheit*. Für jede dieser drei Unterkategorien existieren sowohl positive als auch negative Begriffe, was zu sechs verschiedenen Kategorien führt, wie in Tabelle 1 ersichtlich ist. Somit ergibt sich ein fein-granuliertes Kategoriensystem mit neun verschiedenen Tags für die Annotation im Polaritätslexikon.

Tag	Kategorie	Beispiele aus Lexikon
F_POS	Affect positiv	compassion, amused
F_NEG	Affect negativ	bored, concern
J_POS	Judgement positiv	benevolence, carefully
J_NEG	Judgement negativ	unfair, abuse
A_POS	Appreciation positiv	greatly, charming
A_NEG	Appreciation negativ	crash, dreadful

Tabelle 1.: Polaritätstags im Lexikon

Für mein Extraktionssystem habe ich nun dieses Lexikon mit domänenspezifischen Wörtern ergänzt. Zu diesem Zweck habe ich die SemEval-Daten verwendet. Grund dafür ist, dass diese domänenspezifische Texte sind und somit erwartungsgemäss auch für die Domäne spezifische, polare Begriffe enthalten, die möglicherweise noch nicht im Lexikon enthalten sind. Daher habe ich die Sentimentanalyse-Pipeline auf alle Sätze im Restaurant- und Laptoptrainingsset angewendet. Aus dem erhaltenen VISL-Output habe ich anschliessend alle Adjektive, Substantive und Adverbien, die noch nicht im Lexikon vorhanden und mit keinem Polaritätstag markiert waren, mit einem Python-Skript extrahiert, manuell annotiert und dem Polaritätslexikon beigefügt. Die Anzahl der erweiterten Begriffe beläuft sich auf rund 150, wobei die Mehrheit Adjektive und negativ annotiert sind. Unter den erweiterten Begriffen

befinden sich teils domänenspezifische, wie zum Beispiel *yummy*, *oversalted* oder *bug*, teils jedoch auch domänenunabhängige Wörter, wie *stunning* oder *drawback*.

### 4.3. Verbmodellierung

Eine weitere Komponente der Sentimentanalyse-Pipeline ist eine Verbressource, in welcher englische Verben, die einen Einfluss auf ihre syntaktischen Argumente ausüben, modelliert sind. Die Verbressource ist im VISL-Format geschrieben und enthält zwei grundlegende Komponenten. Zum einen wird der Kasusrahmen von polaren Verben analysiert. Mit Kasusrahmen sind diejenigen syntaktischen Satzglieder gemeint, die von einem bestimmten Verb verlangt werden, damit es vollständig instanziiert werden kann. Das heisst, es wird ermittelt, ob ein Verb beispielsweise transitiv, intransitiv oder reflexiv ist. Zum Kasusrahmen gehören Konstrukte wie Akkusativ- oder Dativobjekte, Komplementsätze oder Subjekte.

Nebst der Ermittlung des Kasusrahmens wird analysiert, wie das jeweilige Verb die Polarität seines Kasusrahmens modifiziert. Dabei wird davon ausgegangen, dass das Verb mindestens einen sogenannten *Polaritätseffekt* oder eine *Polaritätserwartung* auf den Kasusrahmen ausübt. Der Polaritätseffekt (4.4) ist die Auswirkung, die ein Ereignis, welches vom polaren Verb indiziert wird, auf eine oder mehrere Komponenten im Kasusrahmen hat. Die Polaritätserwartung (4.5) hingegen ist eine Vermutung bezüglich der Polarität einer Komponente des Kasusrahmens in Anbetracht der gegebenen Polarität dieser Komponente und der Semantik des Verbs.

(4.4) Das Kind hilft kochen.

(4.5) Die Leute beklagen sich über das Essen.

Beispiel 4.4 illustriert den Polaritätseffekt: Das Verb *helfen* übt einen positiven Effekt auf das Subjekt aus. Das Subjekt *das Kind* weist in diesem Satz ursprünglich keine Polarität auf. Das Verb *helfen* ist jedoch in der Verbressource so modelliert, dass das Subjekt in seinem Kasusrahmen positiv markiert wird, was in diesem Beispiel zutrifft. Beispiel 4.5 hingegen veranschaulicht die Polaritätserwartung: Wiederum haben das Subjekt *die Leute* und das Präpositionalphrasen-Komplement *über das Essen* keine Polarität im Voraus. Das Verb *beklagen* lässt aber darauf schliessen, dass das Beklagte negativ ist, da sich Leute darüber beschweren. Somit wird die Präpositionalphrase mit einem negativen Tag versehen.

Grundsätzlich wird für jedes neue polare Verb, welches noch nicht in der Ressource modelliert ist, der Kasusrahmen und die möglichen Polaritätseffekte oder Erwartungen ermittelt. Mittels dieser Informationen wird das Verb anschliessend der entspre-

chenden Klasse – eine Gruppe von Verben, die denselben Kasusrahmen aufweisen und dieselben Effekte und Erwartungen ausüben – hinzugefügt oder es wird eine neue Klasse modelliert. Für alle Klassen werden in der Verbresource Regeln erstellt, die besagen, welche Komponente des Kasusrahmens welches Tag erhält. Mit dem VISL-CG3 Parser können diese Regeln danach auf die Daten angewendet werden.

Analog zum Polaritätslexikon, habe ich für die Verbresource alle als nicht-polar markierten Verben aus den beiden SemEval-Datensets extrahiert und daraus manuell die polaren Verben herausgefiltert. Um den Kasusrahmen sowie die Effekte und Erwartungen zu ermitteln, habe ich die jeweiligen Sätze, in denen die Verben erscheinen, genauer untersucht und konnte dadurch auf die entsprechende Klasse schließen. Anders als beim Polaritätslexikon habe ich die Verbresource bloss um rund 20 Verben erweitern können. Grund dafür ist, dass die meisten Verben in den Datensets nicht polar oder bereits in der Ressource modelliert sind. Die Erweiterungen waren jedoch meist domänenspezifisch, wie beispielsweise *overcook*, oder *overheat*.

## 4.4. Auswirkung der Ressourcenerweiterungen

Das Ziel der Ressourcenerweiterungen war es, die Sentimentanalyse-Pipeline respektive den VISL-Output für Texte aus den Domänen Restaurant und Laptop zu verbessern. Je mehr Wörter im Polaritätslexikon und in der Verbresource enthalten sind, desto mehr polare Wörter oder durch polare Wörter modifizierte Komponenten eines Satzes werden korrekt markiert. Dies wiederum bedeutet, dass der VISL-Output differenzierter ist und somit effizienter für weitere Verwendungen wie zum Beispiel der Aspekttermextraktion ist. Die folgenden Sätze veranschaulichen, inwiefern sich die Erweiterungen positiv auf den Output ausgewirkt haben:

### (4.6) Vor der Erweiterung des Polaritätslexikons:

```
"<The>" "The" the C_DT DT #1->2
"<food>" "food" food C_NN NN #2->3
"<is>" "is" be C_VBZ VBZ #3->0
"<yummy>" "yummy" yummy C_JJ JJ #4->3
```

### Nach der Erweiterung des Polaritätslexikons:

```
"<The>" "The" the C_DT DT #1->2
"<food>" "food" food C_NN NN POS_predicative #2->3
"<is>" "is" be C_VBZ VBZ verb_POS #3->0
"<yummy>" "yummy" yummy C_JJ JJ A_POS #4->3
```

**(4.7) Vor der Erweiterung der Verbressource:**

"<I>" "I" i C\_PRP PRP #1->3

"<will>" "will" will C\_MD MD #2->3

"<patronize>" "patronize" patronize C\_VB VB #3->0

**Nach der Erweiterung der Verbressource:**

"<I>" "I" i C\_PRP PRP **EFF\_NEG**-POLCHUNK-HEAD #1->3

"<will>" "will" will C\_MD MD #2->3

"<patronize>" "patronize" patronize C\_VB VB **FCLASS\_neg**  
**\_subj\_eff\_dobj verb\_NEG** #3->0

Beispiel 4.6 zeigt, welchen Effekt die Erweiterung des Polaritätslexikons auf den VISL-Output hat. Vor der Erweiterung ist *yummy* noch nicht im Lexikon enthalten und wird somit nicht als positiv markiert. Das Subjekt *food*, auf welches sich *yummy* bezieht erhält dementsprechend auch kein positives Tag. Der Satz würde somit als neutral betrachtet und nicht in die Aspektextraktion miteinbezogen werden, da nur polare Sätze beachtet werden (siehe Kapitel 5). Nach der Erweiterung wird *yummy* korrekterweise mit einem positiven Tag versehen. Das Verb *be* ist in der Verbressource so modelliert, dass wenn als Komplement ein polares Wort auftritt, das Subjekt mit derselben Polarität markiert wird. Dies trifft in diesem Fall zu: *Food* wird mit *POS\_predicative* getaggt.

In Beispiel 4.7 tritt das Verb *patronize* ('herablassend behandeln') auf, das vor der Modellierung noch kein negatives Tag aufweist. Nachdem ich das Verb der Verbressource hinzugefügt habe, wird es im VISL-Output mit einem negativen Tag (*verb\_NEG*) sowie der entsprechenden Klasse (*FCLASS\_neg\_subj\_eff\_dobj*) markiert. Das Verb übt einen Polaritätseffekt auf das Subjekt aus. Das Subjekt wird nun negativ markiert (*EFF\_NEG*), da es eine negative Handlung ausführt.

Da mit Hilfe der domänenspezifischen Erweiterungen der VISL-Output verbessert wurde, haben sich dementsprechend auch die Resultate der verschiedenen Teilaufgaben der SemEval-Aufgabe 4 positiv verändert. Die Ergänzungen waren nicht nur spezifisch für Teilaufgabe 1 vorgesehen, sondern es war generell auch das Ziel, den Output für alle Teilaufgaben zu verbessern. Ich habe das Lexikon und die Verbressource erweitert, bevor ich mich mit der Teilaufgabe 1 befasst habe. Den Einfluss der Erweiterungen konnte ich anhand der Resultate der Teilaufgabe 2 messen<sup>3</sup>. Für die Teilaufgabe 2 ist ein maschinelles Lernverfahren gewählt worden, weshalb sich die Verbesserung anhand der Genauigkeit des Simple Logistic Klassifikators mit 5-facher Kreuzvalidierung erkennen lässt. Genauigkeit bezieht sich in diesem Fall darauf, wie vielen Aspekttermen die richtige Polarität zugewiesen wurde. Wie in

---

<sup>3</sup>Nora Hollenstein hat sich gleichzeitig mit der Teilaufgabe 2 – der Polaritätserkennung von Aspekttermen – beschäftigt, für welche die Erweiterung ebenfalls von Nutzen war.

Tabelle 2 ersichtlich ist, hat die Erweiterung die Genauigkeit des Klassifikators auf dem Restaurantdatenset um 2.8% von 57.7% auf 60.5% verbessert, beim Laptopdatenset um 2.9% auf 52.7%. Diese Resultate zeigen auf, dass ein korrekter und detaillierter VISL-Output einen positiven Einfluss auf weitere Anwendungen hat.

<b>Trainingsset</b>	<b>vorher</b>	<b>nachher</b>	<b>Veränderung</b>
Restaurant	57.7%	60.5%	+2.8%
Laptop	49.8%	52.7%	+2.9%

Tabelle 2.: Genauigkeit des SimpleLogistic Klassifikators vor und nach der Ressourcenerweiterung



# 5. Aspektextraktion

Im Folgenden präsentiere ich das System, welches ich implementiert habe, um Aspekte in den SemEval-Daten zu erkennen und zu extrahieren. Das System baut auf den folgenden drei Hypothesen auf:

1. Eine Nominalphrase, die bewertet wird ist ein Aspekt.
2. Eine Nominalphrase ist ein Aspekt, wenn die Phrase in einem polaren Satz auftritt.
3. Maschinelles Lernen kann als Filter verwendet werden, um zu generelle Aspekt-Kandidaten zu klassifizieren

In einem ersten Teil in Kapitel 5.1 erkläre ich, wie das System aufgebaut ist. Dabei gehe ich genauer auf die erste (Kapitel 5.1.1) und die zweite Hypothese (Kapitel 5.1.2) ein. Anschliessend in Kapitel 5.2 erläutere ich, wie ich die dritte Hypothese umgesetzt habe.

## 5.1. Aufbau des Extraktionssystems

Das Extraktionssystem habe ich in einem Python-Skript (Appendix A.1) implementiert. Es besteht aus verschiedenen Methoden zur Verarbeitung der SemEval-Daten. Als Input erwartet das System eine XML-Datei, auf welche die Sentimentanalyse-Pipeline angewendet worden ist. Dies bedeutet, dass die Datei pro Knoten einen Satz aus einer Rezension sowie den VISL-Output dieses Satzes enthält. Die Datei soll ähnlich wie in Abbildung 1 (siehe Kapitel 3) strukturiert sein; die Aspektterm- und Aspektkategorieknoten sollen jedoch nicht vorhanden sein, dafür ein Knoten für den VISL-Output.

Das Skript liest die Datei ein und iteriert über die einzelnen Sätze. Dabei werden pro Satz alle vorhandenen Nominalphrasen und somit potentielle Aspektterme identifiziert. Die zu erkennenden Aspekte weisen verschiedene Formen auf: Dies können einerseits einzelne Substantive wie *food* oder *graphics* sein oder komplexere Nominal-

phrasen<sup>1</sup>. Letztere können aus mehreren Substantiven bestehen, wie beispielsweise die Komposita *battery life* oder *potato spinach gnocchi*. Auch erkannt werden Nominalphrasen mit einer vorhergehenden Nationenbezeichnung. Damit sind Ausdrücke wie *Italian cuisine* oder *Japanese food* gemeint. Die komplexen Nominalphrasen enthalten ausserdem manchmal auch Präpositionalphrasen. Beispiele dafür sind *chicken with avocado*, *chicken special with Edamame Puree* oder *support for games*. Die in den SemEval-Daten zu erkennenden Aspekte sind meist explizit; falls implizite Aspekte vorkommen werden diese durch die Aspektkategorie – eine allgemeinere Aufgliederung – indiziert. Dies ist jedoch Bestandteil der Teilaufgabe 3, die in dieser Arbeit nicht behandelt wird.

Für die Erkennung von Aspekten erweist sich der VISL-Output als dienlich. Er enthält unter anderem Informationen zu den Wortarten und den Abhängigkeiten zwischen den einzelnen Wörtern. Bei der genaueren Analyse des VISL-Outputs der komplexen Nominalphrasen, habe ich zwei verschiedene Muster des ClearNLP Dependenzparsers erkennen können. Abbildung 2 zeigt die beiden möglichen Muster.

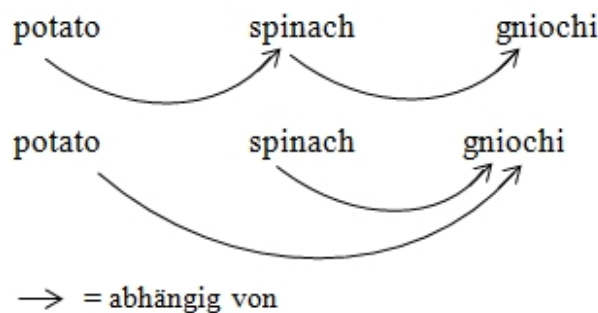


Abbildung 2.: Mögliche Abhängigkeitsmuster von Kompositakomponenten

In gewissen komplexen Phrasen ist die jeweilige Konstituente direkt von der nächsten Konstituenten innerhalb der Phrasen abhängig, d.h., dass die darauffolgende Komponente der Kopf der vorhergehenden bildet. Bei Komposita bedeutet dies, dass die letzte Konstituente dieser Reihe, die ein Substantiv ist, dem Nominalphrasenkopf entspricht und somit das Ende der Phrase markiert, sofern keine Präpositionalphrase folgt, die sich darauf bezieht. Beim zweiten Abhängigkeitsmuster beziehen sich alle Konstituenten direkt auf den Kopf.

Bei Nominalphrasen mit integrierter Präpositionalphrase sind die Abhängigkeiten wie

<sup>1</sup>Grundsätzlich erkennt das System ganze Komposita und bei Nominalphrasen mit attributiven Komponenten den Kopf der Nominalphrasen. Beim Gold Standard entsprechen die Aspekte teilweise nur den Köpfen, teilweise jedoch auch den vollständigen Nominalphrasen. Diese Inkonsistenz stellt für mein regelbasiertes System ein Problem dar, wie in Kapitel 6 erläutert wird.

in Abbildung 3. Das Substantiv oder das Kompositum in der Präpositionalphrase ist abhängig von der Präposition. Diese wiederum weist eine Abhängigkeit vom Kopf der Hauptnominalphrase auf. Die Nominalkomponenten innerhalb der Hauptnominalphrasen können eines der beiden Muster von Abbildung 2 enthalten.

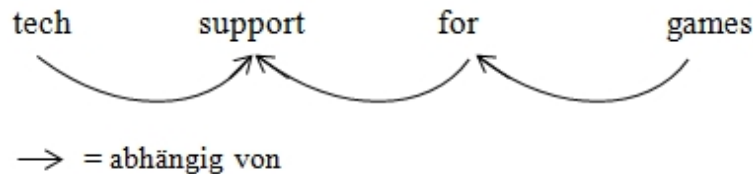


Abbildung 3.: Abhängigkeiten von Nominalphrasen mit Präpositionalkomponenten

Beim Iterieren über die Sätze überprüft das Python-Skript, ob ein Substantiv oder eine Nationenbezeichnung vorhanden ist. Ist dies der Fall, prüft das Skript das darauffolgende Wort, welches entweder ein weiteres Substantiv oder eine Präposition sein muss. Gleichzeitig wird mit Hilfe der Abhängigkeiten untersucht, ob sich die verschiedenen Elemente aufeinander beziehen, respektive ob sie zusammen wie in Abbildung 2 und 3 eine Nominalphrase bilden. Folgt auf ein Substantiv weder eine Präposition noch ein anderes Substantiv, wird das einzelne Substantiv als Aspekt identifiziert.

Wenn in einem Satz ein solches Nominalphrasenmuster erkannt wird, überprüft das Skript in einem weiteren Schritt, ob der momentan bearbeitete Satz polare Elemente enthält. Dieser Schritt wird durchgeführt, weil ich davon ausgehe, dass Aspekte, die in einer Rezension auftreten, bewertet werden. Zur Ermittlung der Polarität habe ich deshalb folgende zwei Methoden verwendet, die in Kapitel 5.1.1 und 5.1.2 genauer erläutert werden:

- Methode 1: Eine Nominalphrase wird dann als Aspekt erkannt, wenn sich ein polares Element innerhalb des Satzes direkt auf die Phrase bezieht.
- Methode 2: Eine Nominalphrase wird dann als Aspekt erkannt, wenn ein polares Element im Satz gefunden wird. Das Element muss sich nicht direkt auf die Nominalphrase beziehen.

Das Skript identifiziert mit Hilfe einer dieser beiden Methoden potentielle Aspektterme. Wenn ein Aspektterm erkannt wird, wird er als Attribut eines neuen Aspektterm-Knotens dem aktuellen Satz hinzugefügt und zusätzlich in einer Liste abgespeichert. Ausserdem werden nebst dem Term die Anfangs- und Endposition des Terms innerhalb der Textzeichenkette als *from*- und *to*-Attribute dem Knoten hinzugefügt, sowie ein leeres *polarity*-Attribut, welches für die Evaluation mit dem SemEval-

Evaluationsskript enthalten sein muss (siehe Kapitel 6).

Wenn das Skript über alle Sätze iteriert und alle möglichen Aspekte identifiziert und getaggt hat, wird anschliessend noch einmal über alle Sätze iteriert. Bei dieser zweiten Iteration werden nur jene Sätze, die noch kein Aspektterm-Knoten aufweisen, in Betracht gezogen. Hierbei handelt es sich um neutrale Sätze, die keine polaren Elemente enthalten oder bei denen polare Elemente fälschlicherweise nicht markiert wurden. Gründe dafür können fehlende Lexikon- oder Verbressourceneinträge sein oder ein fehlerhafter Output des ClearNLP Dependenzparsers, was dazu führt, dass die VISL-Regeln, die polare Elemente markieren, nicht ausgeführt werden können. Aus diesem Grund wird in allen Sätzen, die noch kein Aspektterm-Knoten enthalten, überprüft, ob ein Aspektterm aus der zuvor erstellten Liste im jeweiligen Satz enthalten ist. Dabei wird nach dem Prinzip vorgegangen, dass die längst mögliche Zeichenkette und somit keine Teilzeichenkette als Aspektterm erkannt wird. Das folgende Beispiel erläutert dieses Prinzip: Ein Satz enthält den Aspekt *lobster risotto*. Die Liste der identifizierten Aspektterme beinhaltet unter anderem die Zeichenketten *lobster*, *risotto* und *lobster risotto*. Da der Satz alle drei Zeichenketten enthält, würden alle als potentielle Aspektterme gehandelt und getaggt werden. Um dies zu verhindern, überprüft das Skript für jedes gefundene Element, ob es eine Teilzeichenkette eines anderen gefundenen Elementes ist. In diesem Beispiel sind *lobster* und *risotto* Teilzeichenketten von *lobster risotto*, deshalb wird nur Letzteres als Aspektterm-Knoten dem Satz hinzugefügt.

Im Anschluss an die zweite Iteration, wird bei jedem Satz der VISL-Output-Knoten entfernt. Die XML-Datei entspricht nun dem Format, welches für die Evaluation verwendet wird. Abbildung 4 zeigt einen Eintrag aus der Datei.

```
<sentence id="1094:1">
  <text>Nice screen, keyboard works great!</text>
  <aspectTerms>
    <aspectTerm term="screen" from="5" to="11"/>
    <aspectTerm term="keyboard" from="13" to="21"/>
  </aspectTerms>
</sentence>
```

Abbildung 4.: Outputbeispiel des Aspektextraktionssystems

### 5.1.1. Methode 1: Polar markierte Nominalphrasen

Wie in Kapitel 5.1 erwähnt, habe ich zwei verschiedene Methoden gewählt, um Köpfe gefundener Nominalphrasen in einem Satz als Aspektterme klassifizieren zu können.

Die Methode 1 entspricht meiner ursprünglichen Idee: Diejenigen Elemente in einer Rezension, die positiv oder negativ markiert sind und somit bewertet werden, sind die gesuchten Aspektterme. Bei dieser Methode werden demnach Nominalphrasen nur als Aspekt erkannt, wenn der Kopf der Phrase selbst kein Polaritätstag (*A\_POS*, *J\_NEG* etc.) aufweist und im selben Satz ein Element auftritt, das als polar markiert ist und sich direkt auf die Nominalphrase bezieht.

```
(5.1) "<Great>" "Great" great C_JJ JJ A_POS COORD_CHILD_POS
      #1->2
      "<Performance>" "Performance" performance C_NNP NNP
      COORD_HEAD_POS #2->0
      "<and>" "and" and C_CC CC COORD_CHILD_POS #3->2
      "<Quality>" "Quality" quality C_NNP NNP COORD_CHILD_POS #4->2
      "<.>" "." . C_ . COORD_CHILD_POS #5->2
```

Im Beispielsatz 5.1 sind zwei Substantive in einer koordinierten Phrase enthalten: *performance* und *quality*. Der Satz weist ausserdem das positive Adjektiv *great* auf. Anhand des Musters *#Zahl->Zahl* am Ende jeder Analysezeile kann erkannt werden, welche Konstituente des Satzes von welcher abhängig ist. Die erste Zahl entspricht der Position der jeweiligen Konstituenten innerhalb des Satzes; die zweite Zahl verweist auf den Kopf der Konstituenten. Wie im Beispiel ersichtlich ist, bezieht sich das Adjektiv auf das erste Element der koordinierten Phrase, die dadurch positiv markiert wird. Somit werden die beiden Substantive bei der Anwendung von Methode 1 als Aspektterme erkannt.

### 5.1.2. Methode 2: Nominalphrasen in polaren Sätzen

Im Gegensatz zu Methode 1, werden in Methode 2 sämtliche Köpfe von Nominalphrasen als potentielle Aspekte identifiziert, falls im selben Satz irgendeine polare Konstituente auftritt. Wiederum gilt auch bei dieser Methode die Regel, dass der Kopf der Nominalphrase kein polares Wort sein darf, da Aspektterme gewöhnlich nicht polar sind. Das folgende Beispiel zeigt einen Satz aus dem Restaurantdatenset, aus dem mit Hilfe von Methode 2 ein Aspektterm extrahiert wird:

```
(5.2) "<The>" "The" the C_DT DT _ #1->3
      "<wait>" "wait" wait C_NN NN #2->3
      "<staff>" "staff" staff C_NN NN #3->4
      "<was>" "was" be C_VBD VBD #4->0
      "<loud>" "loud" loud C_JJ JJ A_NEG COORD_HEAD_NEG #5->4
      "<and>" "and" and C_CC CC COORD_CHILD_NEG #6->5
```

```
"<inconsiderate>" "inconsiderate" inconsiderate C_JJ JJ J_NEG  
COORD_CHILD_NEG #7->5  
"<.>" ". " . C_ . #8->4
```

Unter Verwendung von Methode 1, würde der Aspektterm *wait staff* nicht identifiziert werden, da sich keine polare Konstituente im Satz direkt auf die Phrase bezieht. Aus diesem Grund habe ich die Methode 2 implementiert. Ein weiterer Grund für die zusätzliche Methode ist, dass der ClearNLP Parser-Output nicht immer fehlerfrei ist. Wenn der Parser falsche Abhängigkeiten ermittelt und sich dadurch ein polares Element fälschlicherweise nicht auf eine entsprechende Nominalphrase bezieht, wird die Phrase bei Methode 2 trotzdem als Aspekt identifiziert. Dies trifft in Beispiel 5.2 zu und somit wird *wait staff* als Aspekt extrahiert.

Im Vergleich zu Methode 1, werden in Methode 2 mehr Aspekte identifiziert, da die Methode ein vergleichsweise gieriger Algorithmus ist: Alle potentiellen Aspektterme aus Methode 1 werden auch automatisch mit der Methode 2 erkannt. Umgekehrt ist dies nicht der Fall.

## 5.2. Erweiterung durch ein maschinelles Lernverfahren

Nebst dem in Kapitel 5.1 beschriebenen, regelbasierten System, habe ich maschinelles Lernen verwendet, um die Aspektextraktion zu verbessern. Das maschinelle Lernverfahren ist jedoch nicht ein neues, unabhängiges System, sondern baut auf das regelbasierte auf. Das Ziel dieser Erweiterung ist es, die Leistung des regelbasierten Systems zu erhöhen.

In einem ersten Schritt habe ich ein Python-Skript (Appendix A.2) geschrieben, welches Merkmalsvektoren für die einzelnen Aspekte erstellt. Diese Merkmalsvektoren werden anschliessend für das Training eines Klassifikators benötigt. Als Input erwartet das Skript eine XML-Datei, die ein ähnliches Format aufweist wie der Output des regelbasierten Systems. Der Unterschied ist jedoch, dass der Knoten mit dem VISL-Output noch vorhanden sein muss, d.h., die Datei enthält die einzelnen Sätze, sowie Aspektterm- und VISL-Knoten. Das Skript liest die XML-Datei ein und iteriert über jeden Satz. Dabei extrahiert es die getaggten Aspektterme sowie die entsprechenden Merkmale für den Merkmalsvektor. In Tabelle 3 sind alle elf Merkmale aufgelistet, die zur Erstellung des jeweiligen Vektors herausgefiltert werden.

Nebst diesen elf Merkmalen habe ich noch weitere getestet, wie beispielsweise die Anzahl Negationen oder die Anzahl positiver oder negativer Prädikate, die einen

Merkmal	Datentyp	Beschreibung
form	Array von Zeichenketten	Wortform des entsprechenden Aspekts
pos_in_Sent	binär	Wenn eine positive Konstituente im Satz enthalten ist, erhält dieses Merkmal den Wert 1, ansonsten 0
neg_in_Sent	binär	Wenn eine negative Konstituente im Satz enthalten ist, erhält dieses Merkmal den Wert 1, ansonsten 0
num_adj	Integer	Anzahl der Adjektive im Satz
pos_adj	Integer	Anzahl der positiven Adjektive im Satz
neg_adj	Integer	Anzahl der negativen Adjektive im Satz
num_pos	Integer	Anzahl der positiven Konstituenten im Satz
num_neg	Integer	Anzahl der negativen Konstituenten im Satz
strong	Integer	Anzahl der stark polaren Konstituenten im Satz
aktiv	Integer	Anzahl aktiven Konstituenten im Satz
passiv	Integer	Anzahl passiven Konstituenten im Satz

Tabelle 3.: Extrahierte Merkmale eines Aspekts pro Satz

Effekt auf das Subjekt ausüben. Diese haben sich jedoch nach mehrfachen Testdurchläufen mit dem Klassifikator als weniger diskriminierend erwiesen.

Nachdem das Skript alle Aspektterme und deren Merkmale extrahiert hat, wandelt es diese in Merkmalsvektoren um und schreibt sie in eine *arff*-Datei. Das *arff*-Format ist insofern wichtig, als es für die Nutzung von *weka* [Hall et al., 2009] verlangt wird. *Weka* ist eine Sammlung von Algorithmen für das maschinelle Lernen, die ich für meine Arbeit verwendet habe.

Bevor nun ein Klassifikator von *weka* trainiert und zum Klassifizieren verwendet werden kann, muss zuerst ein Trainings- und ein Testset kreiert werden. Das Trainingsset muss Vektoren der positiven wie auch der negativen Klasse enthalten, damit der Klassifikator mittels dieser Vektoren Regeln zur Klassifizierung lernen kann. Die positive Klasse (*yes*) beinhaltet alle korrekten Aspektterme; die negative Klasse (*no*) bezieht sich auf alle Nominalphrasen, die fälschlicherweise als Aspektterme identifiziert werden. Für die positive Klasse werden alle Aspektterme und deren Merkmale aus dem manuell annotierten Restaurant- respektive Laptoptrainingsset extrahiert und in Merkmalsvektoren umgewandelt. Um Beispiele für die negative Klasse zu erhalten, lasse ich mein regelbasiertes System alle potentiellen Aspektterme in den SemEval-Trainingssets ermitteln und gleiche die gefundenen Nominalphrasen mit den vorannotierten Aspekten im Trainingsset ab. Falls eine Phrase vom regelbasierten System gefunden wurde, jedoch nicht im Trainingsset annotiert ist, wird für diese Phrase von jedem Satz, in der sie auftritt, ein Merkmalsvektor erstellt. Die Merkmalsvektoren bilden Beispiele für die negative Klasse. Beispiel 5.3 zeigt mögliche

Merkmalsvektoren des Laptoptrainingssets, wobei die Merkmale in derselben Reihenfolge auftreten, wie sie in Tabelle 3 aufgelistet sind.

(5.3) applications,1,1,6,4,0,23,3,2,1,0,yes  
college,1,0,3,3,0,9,0,0,0,0,no

Um die Testdaten zu erstellen, wende ich das regelbasierte System auf ein Testset an, in diesem Fall auf die von SemEval zur Verfügung gestellten Restaurant- und Laptopstestsets. Das System extrahiert alle möglichen Aspekte und fügt sie dem Testset hinzu. Dieses Testset wird anschliessend dem Skript (Appendix A.2) übergeben, welches alle Merkmale extrahiert und wiederum eine *arff*-Datei erstellt. Anstatt jedoch wie beim Trainingsset die Klassen *yes* und *no* anzugeben, wird anstelle der Klasse ein *?* eingefügt, da die Klasse vom Klassifikator zuerst ermittelt werden muss.

Nachdem die Trainings- und Testdateien für *weka* erstellt worden sind, kann ein Klassifikator darauf angewendet werden. *Weka* stellt verschiedene Klassifikatoren zur Verfügung. Für meine Arbeit habe ich diverse solcher Algorithmen – unter anderem den *SimpleLogistic*, *J48*, *NaiveBayes* und den *IBk* – auf den vorhandenen Daten angewendet. Dabei hat sich herausgestellt, dass der *NaiveBayes* auf den Daten die besten Resultate erzielt. Infolgedessen habe ich diesen Klassifikator auf dem erstellten Trainingsset mit 5-facher Kreuzvalidierung trainiert. Mit dem Restauranttrainingsset hat der *NaiveBayes* eine Genauigkeit von 88.9% und mit dem Laptoptrainingsset 90.0% erreicht.

Anschliessend habe ich den trainierten Klassifikator die Testdatei klassifizieren lassen. Er weist jedem Aspekt entweder die Klasse *yes* oder *no* zu. Mit Hilfe dieser Klassifizierung entfernt nun ein weiteres Python-Skript (Appendix A.3) die vom regelbasierten System fälschlicherweise annotierten Aspektterme – diejenigen, die vom Klassifikator in die negative Klasse eingeteilt wurden – aus der XML-Datei.

Das maschinelle Lernverfahren weist somit ein Filterfunktion auf und versucht zu generelle Aspekt-Kandidaten zu eliminieren.



## 6. Evaluation

Im folgenden Kapitel evaluiere ich das implementierte Aspektextraktionssystem und vergleiche es mit der von SemEval zur Verfügung gestellten Baseline. Zuerst erläutere ich meine Vorgehensweise bei der Evaluation des Systems in Kapitel 6.1. Anschließend in Kapitel 6.2 präsentiere und analysiere ich die Resultate.

### 6.1. Evaluationsmethode

Im Rahmen des SemEval-Workshops hat zuerst eine Trainingsphase stattgefunden, in der die Teilnehmer mittels verfügbaren Trainingsdaten ihre Systeme aufbauen und trainieren konnten. Nach dieser Trainingsphase haben die SemEval-Organisatoren die Testdaten für die Restaurant- und Laptopdomänen veröffentlicht. Die zwei Testsets sind wie bereits das Trainingsset im XML-Format und enthalten je 800 Sätze, aus denen das System alle Aspekte extrahieren soll. Um die Testdateien vom System verarbeiten lassen zu können, habe ich sie zuerst durch die Sentimentanalyse-Pipeline laufen lassen, die den benötigten VISL-Output produziert und in die Testdateien einfügt hat. Im Anschluss daran habe ich das implementierte Aspektextraktionssystem auf die Dateien angewandt und den Output für die Evaluation verwendet.

Die Verantwortlichen des SemEval-Workshops haben ein Evaluationskript (Appendix A.4) zur Verfügung gestellt, welches ich für die Evaluierung verwende. Bei der Evaluation kann in einem ersten Schritt mit Hilfe eines zur Verfügung gestellten XML Schemas überprüft werden, ob die zu evaluierende Datei die erwünschte Struktur aufweist. Wenn dies zutrifft, kann in einem nächsten Schritt das Evaluationskript angewendet werden. Das Skript berechnet anhand der übergebenen Datei die Präzision, den Recall und das F-Mass.

Damit die Teilnehmer des SemEval-Workshops die Leistung ihrer Systeme einschätzen können, haben die Organisatoren ein Baseline-System<sup>1</sup> veröffentlicht, mit dem die Systeme verglichen werden können. Das Baseline-System basiert auf einer einfachen

---

<sup>1</sup>Baseline-System:<http://alt.qcri.org/semeval2014/task4/data/uploads/baselinesystemdescription.pdf>

Methode: Es sammelt alle Aspektterme, die in der Trainingsdatei vorhanden sind, und speichert sie in einer Dictionary-Datenstruktur ab. Um Aspektterme zu identifizieren wird schlicht überprüft, ob ein Term im Dictionary vorhanden ist. Dieses System erkennt dadurch nur Aspektterme, die im Trainingsset aufgetreten sind.

## 6.2. Resultate und Diskussion

Wie in Kapitel 5 beschrieben, habe ich zwei verschiedene Methoden in meinem regelbasierten System verwendet, um Nominalphrasen als Aspektterme zu identifizieren. Die erste Methode erkennt die Phrasen als Aspekte, wenn sich eine polare Konstituente direkt darauf bezieht; die zweite Methode identifiziert eine Phrase als Aspektterm, sobald eine polare Konstituente im Satz auftritt. Die Tabelle 4 zeigt auf, wie gut das Aspektextraktionssystem mit den beiden Methoden im Vergleich zur Baseline auf den Restaurantdaten funktioniert.

	Baseline	Methode 1	Methode 2
<b>Präzision</b>	62.7%	46.7%	44.9%
<b>Recall</b>	37.7%	59.2%	69.8%
<b>F-Mass</b>	47.1%	52.2%	54.6%

Tabelle 4.: Resultate für das Restauranttestset mit dem regelbasierten System

Das System erzielt mit beiden Methoden bessere F-Mass Werte als die Baseline: Mit der Methode 1 übertrifft der Systemoutput die Baseline um 5.1%, mit der Methode 2 um 7.5%. Unter der Verwendung von Methode 2 erreicht das System somit das beste Resultat auf den Restauranttestdaten. Beim Betrachten der Tabelle 4 fällt auf, dass das System im Vergleich zur Baseline jeweils einen höheren Recall als Präzision aufweist. Ein Grund für den tiefen Recall der Baseline liegt möglicherweise darin, dass die Baseline auf einem einfachen Verfahren basiert und Nominalphrasen nur als Aspekte erkennt, wenn diese bereits im Trainingset vorgekommen sind. Das Testset scheint somit viele Aspektterme zu enthalten, die im Trainingsset nicht aufgetreten sind und somit von der Baseline nicht identifiziert werden können. Die beiden implementierten Methoden hingegen erkennen insgesamt mehr Aspekte. Insbesondere die Methode 2 erzielt einen hohen Recall von rund 70%. Dies liegt daran, dass die Methode „gierig“ ist, da sie alle Nominalphrasen in einem polaren Satz als Aspekte identifiziert. Die Methode findet so zwar mehr Aspekte, jedoch auch viele Nominal-

phrasen, die nicht als Aspekte erfasst werden sollten. Dies macht sich in der tiefen Präzision bemerkbar. Die Methode 1 hingegen identifiziert weniger Aspekte; von denen, die erkannt werden, sind aber eine grössere Anzahl echte Aspekte.

In Tabelle 5 sind die Resultate für das Laptotestset aufgeführt.

	<b>Baseline</b>	<b>Methode 1</b>	<b>Methode 2</b>
<b>Präzision</b>	47.4%	17.5%	18.9%
<b>Recall</b>	21.8%	36.0%	43.6%
<b>F-Mass</b>	29.8%	23.5%	26.4%

Tabelle 5.: Resultate für das Laptotestset mit dem regelbasierten System

Die Resultate fallen im Vergleich zum Restauranttestset für alle Systeme relativ schlecht aus. Mein implementiertes System erzielt mit beiden Methoden ein schlechteres Ergebnis als die Baseline. Der Recall ist zwar in beiden Fällen besser, die Präzision ist jedoch so tief, dass die Leistung des Systems insgesamt ernüchternd ist. Nachdem ich die Laptotestdatei analysiert habe, sind mir die folgenden möglichen Ursachen für die niedrigen Ergebnisse aufgefallen: Die Sätze der Laptotestprezensionen sind komplexer als diejenigen des Restauranttestsets. Dies hat zu einer erhöhten Fehlerrate im VISL-Output und somit zu einer niedrigen Leistung des Aspektextraktionssystems geführt. Ausserdem sind im Goldstandard des Laptotestsets teilweise auch Verben wie *work* oder *configure* als Aspektterme annotiert. Das Aspektextraktionssystem identifiziert jedoch momentan nur Nominalphrasen.

In einem weiteren Schritt habe ich – wie in Kapitel 5.2 beschrieben – das System mit einem maschinellen Lernverfahren erweitert. In Tabelle 6 wird das Ergebnis der Erweiterung für das Restauranttestset veranschaulicht.

	<b>Methode 1</b>	<b>Methode 2</b>
<b>Präzision</b>	60.9%	58.4%
<b>Recall</b>	58.2%	68.3%
<b>F-Mass mit masch. Verfahren</b>	59.5%	63.0%
<b>F-Mass mit regelbas. System</b>	52.2%	54.6%
<b>Veränderung</b>	+7.3%	+8.4%

Tabelle 6.: Resultate für das Restauranttestset mit dem maschinellen Lernverfahren

Mit dem maschinellen Lernverfahren hat sich das System noch einmal deutlich verbessert. Das F-Mass des Systems ist mit der maschinellen Erweiterung unter Verwendung der Methode 1 um 7.3% auf 59.5% gestiegen; mit der Methode 2 sogar um 8.4% auf 63%. Beim Betrachten der Tabelle fällt auf, dass das maschinelle Lernverfahren die Präzision beider Methoden stark verbessert hat, nur der Recall ist minim gesunken. Mit Hilfe dieses Verfahrens konnte somit erfolgreich eine Grosszahl von Nominalphrasen, die vom regelbasierten System fälschlicherweise als Aspekte identifiziert worden sind, ausgeschlossen werden. Der nur leicht niedrigere Recall zeigt, dass der NaiveBayes Klassifikator nur wenige echte Aspektterme der negativen Klasse zugeordnet hat.

Ein ähnliches Bild zeigt sich beim Laptoptestset, wie die Tabelle 7 illustriert.

	<b>Methode 1</b>	<b>Methode 2</b>
<b>Präzision</b>	22.6%	54.2%
<b>Recall</b>	32.4%	36.4%
<b>F-Mass mit masch. Verfahren</b>	26.6%	43.5%
<b>F-Mass mit regelbas. System</b>	23.5%	26.4%
<b>Veränderung</b>	+3.1%	+17.1%

Tabelle 7.: Resultate für das Laptoptestset mit dem maschinellen Lernverfahren

Auch mit diesem Testset erzielt die Methode 2 die besten Ergebnisse. Wiederum verbessert das maschinelle Lernverfahren die Präzision bei beiden Methoden, bei Methode 1 um 5.1% und bei Methode 2 um gute 35.3%. Das maschinelle Lernverfahren hat jedoch einen grösseren Einfluss auf den Recall des Laptoptestsets, als dies bei den Restaurantrezensionen der Fall ist: Mit Methode 1 verringert sich der Recall um 3.6% und mit Methode 2 um 7.2%. Der NaiveBayes Klassifikator scheint in diesem Fall mehr echte Aspektterme in die negative Klasse einzuteilen. Dies kann wiederum auf den fehlerhaften VISL-Output zurückgeführt werden. Insgesamt erzielt das maschinelle Lernverfahren jedoch Verbesserungen mit beiden Methoden, wobei das F-Mass unter Anwendung von Methode 2 um gute 17.1% auf 43.5% verbessert wird. Dies ist im Vergleich mit der Baseline, die ein F-Mass von 29.8% erreicht, ein relativ gutes Ergebnis.

Mein Aspektextraktionssystem hat in den meisten Fällen – ausser beim Laptoptestset ohne die Erweiterung mit dem maschinellen Lernverfahren – die Baseline klar übertroffen. Dennoch scheint grosses Verbesserungspotential vorhanden zu sein, denn das System erreicht maximal einen Wert von 63% für das F-Mass. Einen star-

ken Einfluss auf das Resultat hat sicherlich der VISL-Output. Das Polaritätslexikon sowie die Verbressource, die noch längst nicht ausgeschöpft sind, könnten weiter ergänzt werden, was den VISL-Output und somit das Resultat positiv beeinflussen würde. Die Erweiterungen sind jedoch nur dann von Nutzen, wenn die Leistung des Dependenzparsers ausreichend ist. Bei komplexen Sätzen produziert der Parser jedoch nicht immer einen korrekten Output; Fehler treten insbesondere bei den Wortarten in der jeweiligen Analysezeile auf. Dies wirkt sich auf das Extraktionssystem aus, denn wenn Substantive nicht als solche markiert sind kann das System sie nicht als Aspektterme erkennen. Umgekehrt verfälschen auch Wörter, die fälschlicherweise als Substantive erkannt worden sind, das Resultat, weil sie vom System für potentielle Aspekte gehalten werden. Solche Wörter sind meist ambig und können mehreren Wortarten zugeordnet werden, wie beispielsweise *light*, welches ein Adjektiv oder ein Substantiv sein kann.

Die Verbesserung des VISL-Outputs hat einen positiven Einfluss auf den regelbasierten Teil des Systems. Um die Leistung des maschinellen Lernverfahrens zu erhöhen, könnten weitere Merkmale für die Merkmalsvektoren getestet werden. In meiner Arbeit habe ich eine ausgewählte Anzahl von Merkmalen verwendet, wobei sich insbesondere das erste Merkmal – die Form des Aspektterms – als effektiv für die Klassifikation erwiesen hat. Die Anzahl der Merkmale kann jedoch noch erweitert werden und dies könnte – möglicherweise auch mit einem anderen Klassifikator – zu einem besseren Ergebnis führen.

Nebst den erwähnten Verbesserungsmöglichkeiten, muss jedoch auch beachtet werden, dass die Annotationen des Goldstandards, auf welchem das Aspektextraktionssystem evaluiert wird, teilweise Fehler aufweisen, respektive manchmal inkonsequent sind. Im Satz *we like to add our fresh salsa into it* ist beispielsweise im Goldstandard der ganze Ausdruck *fresh salsa* als Aspekt annotiert, hingegen in *the dinner menu offers a variety of great entrees, including fresh seafood and huge steaks* wird nur das Substantiv *seafood* ohne das Adjektiv *fresh* als Aspekt identifiziert. Mein implementiertes System ermittelt den Aspektterm im zweiten Satz korrekt; im ersten wird jedoch nur *salsa* erkannt, was das Resultat verschlechtert. Solche Unregelmässigkeiten im Goldstandard tragen dazu bei, dass die Leistung des Systems nicht höher ist.

Insgesamt haben für Aufgabe 4 des Workshops 28 Teilnehmer ein Resultat mit dem Restaurantdatenset eingereicht. Die F-Masse der Teilnehmer liegen in einem Bereich zwischen 38% - 84%. Mit der Erweiterung des maschinellen Lernverfahrens und unter der Verwendung der Methode 2 erzielt mein System 63%. Dieses Ergebnis ist eher im hinteren Bereich der Rangliste einzuordnen, denn rund die Hälfte aller Teilnehmer haben mit ihren Ansätzen ein F-Mass zwischen 70% und 80% erreicht. Ähnlich ist das Resultat für das Laptopdatenset. Hier haben 27 Personen oder Teams teilgenommen. Die F-Masse ihrer Ansätze liegen zwischen 24% und 75%. Bei diesem

Datenset liegt mein bestes Resultat von 43.5% ebenfalls im hinteren Bereich der Rangliste. Ein wahrscheinlicher Grund für dieses Ergebnis ist, dass mein System ein relativ tiefer Recall aufweist, da es auf Regeln basiert und somit unter anderem auch wegen Unregelmässigkeiten im Gold Standard nicht viele Aspekte identifizieren kann.

## 7. Schlussfolgerung

Im Rahmen des SemEval-Workshops 2014 habe ich mich in meiner Bachelorarbeit mit der Aspektextraktion beschäftigt. Das Ziel der Arbeit war es, ein System zu implementieren, das aus Rezensionen der Domänen Restaurant und Laptop automatisch Aspektterme identifiziert. Dabei habe ich drei grundsätzliche Ideen verfolgt: Der erste Grundgedanke war, dass Aspektterme denjenigen Nominalphrasen in einem Satz entsprechen, die bewertet werden, wie dies auch Hu und Liu [2004] vorgeschlagen haben. Die zweite Idee war, dass alle Nominalphrasen Aspektterme sind, sobald sie in einem polaren Satz auftreten, unabhängig davon, ob die Nominalphrasen direkt polar bewertet werden. Die dritte Idee, die ich verfolgt habe, war, dass ein maschinelles Lernverfahren als Filter für zu generelle Aspekt-Kandidaten verwendet werden kann.

Um Bewertungen in einem Satz erkennen zu können, habe ich die vom Institut für Computerlinguistik Zürich entwickelte Sentimentanalyse-Pipeline verwendet. Die Pipeline liest jede Rezension ein, parst sie mit dem ClearNLP Abhängigkeitsparser und wandelt den Output des Parsers ins VISL-Format um. In einem weiteren Schritt werden unter anderem ein Polaritätslexikon und eine Verbressource verwendet, um im geparsten Output polare Wörter und deren Effekte und Erwartungen auf andere Elemente innerhalb des Satzes zu markieren. Als Teil meiner Arbeit habe ich dieses Lexikon und die Verbressource domänenspezifisch erweitert, was eine Verbesserung des Outputs und somit einen positiven Einfluss auf die Leistung des Aspektextraktionssystems zur Folge hatte.

Bei der Implementierung des Aspektextraktionssystems, welches anhand des Pipeline-Outputs potentielle Aspektterme ermittelt, habe ich zwei verschiedene Methoden getestet: Bei Methode 1 identifiziert das System nur Phrasen als Aspektterme, auf die sich ein polares Wort direkt bezieht; bei Methode 2 wird eine Phrase als Aspekt erkannt, sobald im selben Satz ein polares Wort auftritt. Beide Methoden haben für das Restaurantdatenset bessere Resultate als die SemEval-Baseline erzielt. Beim Laptopdatenset hingegen war die Präzision der beiden Methoden so tief, dass die Baseline nicht übertroffen werden konnte. Erst mit der Erweiterung eines maschinellen Lernverfahrens hat das System deutlich bessere Ergebnisse erreicht, denn dieses zusätzliche Verfahren erkennt viele vom System fälschlicherweise als Aspektterm

identifizierte Phrasen und eliminiert sie. Die Resultate haben gezeigt, dass die Methode 2 in beiden Domänen besser funktioniert als Methode 1, nicht zuletzt auch weil der Pipeline-Output Fehler aufweist und sich polare Wörter somit nicht immer direkt auf den Aspektterm beziehen. Grundsätzlich war die Aspektextraktion bei der Restaurantdomäne erfolgreicher als bei der Laptopdomäne.

Mein Aspektextraktionssystem hat sich als ein erfolgreicher Ansatz erwiesen, dennoch scheint noch einiges Verbesserungspotential vorhanden zu sein, zumindest für die vorgestellten Domänen. In einer weiteren Untersuchung wäre es interessant zu sehen, wie gut das System auf anderen Domänen funktioniert. Das Polaritätslexikon sowie die Verbressource müssten dabei wiederum mit domänenspezifischen Wörtern ergänzt werden, um die Leistungen in den verschiedenen Domänen vergleichen zu können.

Die vorliegende Arbeit hat gezeigt, dass mit einer einfachen Idee und guter linguistischer Analyse Aspektextraktion möglich ist. Die Ideen für weitere Ansätze auf diesem Gebiet sind bestimmt noch nicht vollständig ausgeschöpft.



# Bibliografie

- Sasha Blair-Goldensohn, Kerry Hannan, Ryan McDonald, Tyler Neylon, George A Reis, und Jeff Reynar. Building a sentiment summarizer for local service reviews. In *Proceedings of WWW Workshop on NLP in the Information Explosion Era*, 2008.
- Gayatree Ganu, Noemie Elhadad, und Amélie Marian. Beyond the stars: Improving rating predictions using review text content. In *Proceedings of the 12th International Workshop on the Web and Databases*, 2009.
- Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, und Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.
- Minqing Hu und Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM, 2004.
- Niklas Jakob und Iryna Gurevych. Extracting opinion targets in a single-and cross-domain setting with conditional random fields. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1035–1045, 2010.
- Wei Jin und Hung Hay Ho. A novel lexicalized hmm-based learning framework for web opinion mining. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 465–472, 2009.
- Nozomi Kobayashi, Ryu Iida, Kentaro Inui, und Yuji Matsumoto. Opinion mining on the web by extracting subject-attribute-value relations. In *Proceedings of AAAI-CAAW 06*, 2006.
- Lun-Wei Ku, Yu-Ting Liang, und Hsin-Hsi Chen. Opinion extraction, summarization and tracking in news and blog corpora. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, volume 100107, 2006.

- Bing Liu. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167, 2012.
- James R Martin und Peter R White. *The Language of Evaluation: Appraisal in English*. Palgrave Macmillan, 2003.
- Karo Moilanen und Stephen Pulman. Sentiment composition. In *Proceedings of the Recent Advances in Natural Language Processing International Conference*, pages 378–382, 2007.
- Ana-Maria Popescu und Oren Etzioni. Extracting product features and opinions from reviews. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, 2005.
- Annette Rios und Simon Clematide. Constraint grammar mit visl. Technical report, Institute of Computational Linguistics, University Zürich, 2010.
- Swapna Somasundaran und Janyce Wiebe. Recognizing stances in online debates. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pages 226–234, 2009.
- Ivan Titov und Ryan T McDonald. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, volume 8, pages 308–316, 2008.

# Selbständigkeitserklärung

Hiermit versichere ich, dass ich die vorgelegte Arbeit selbständig und ohne unerlaubte Hilfsmittel verfasst habe. Andere als die angegebenen Hilfsmittel habe ich nicht verwendet.

Ort/Datum: Villmergen, 9. Juli 2014

Unterschrift:

A handwritten signature in black ink, appearing to read "M. Bachmann", is centered on the page. The signature is written in a cursive style with a horizontal line at the end.

# A. Skripte

## (A.1) *aspect\_extraction.py*

In diesem Skript ist das regelbasierte System mit Methode 1 und 2 implementiert. Der Input und der Output entspricht einer XML-Datei.

## (A.2) *feature\_extraction.py*

Das Skript extrahiert die Merkmale für den jeweiligen Aspektterm und erstellt die Merkmalsvektoren. Als Input erwartet es eine XML-Datei; als Output gibt es eine arff-Datei zurück.

## (A.3) *add\_aspects.py*

Das Skript eliminiert die vom maschinellen Verfahren als Nicht-Aspektterme klassifizierte Phrasen. Der Input und der Output entspricht einer XML-Datei.

## (A.4) *semeval\_base.py*

Das Skript ist von SemEval zur Verfügung gestellt worden und evaluiert die mit Aspekten angereicherte XML-Datei. Dabei wird die Präzision, der Recall und das F-Mass berechnet.