Institut für maschinelle Sprachverarbeitung
Pfaffenwaldring 5b
D-70569 Stuttgart

Bachelorarbeit Nr. 175

Vokabular-globale lexikalische Substitution in einem Vektorraummodell

Burak Erkus

Studiengang: Informatik Bachelor of Science

Prüfer: Prof. Dr. Sebastian Padó
Betreuer: Prof. Dr. Sebastian Padó

Beginn am: 20. September 2014
Beendet am: 20. März 2015

CR-Nummer: H.3.1, H.3.3, I.2.7

INHALTSVERZEICHNIS

INHALTSVERZEICHNIS ---- **III**

1 EINLEITUNG ---- **4**

1.1 MOTIVATION UND ZIELSETZUNG ---- 4
1.2 GLIEDERUNG ---- 6
1.3 GRUNDLAGEN DER LEXIKALISCHEN AMBIGUITÄT ---- 7
1.3.1 Ambiguität ---- *7*
1.3.2 WordNet ---- *8*
1.3.3 Disambiguierung ---- *10*

2 STAND DER FORSCHUNG ---- **12**

2.1 LEXIKALISCHE SUBSTITUTION ---- 12
2.2 SUPERVISED DISAMBIGUATION (ÜBERWACHTE SYSTEME) ---- 16
2.3 UNSUPERVISED DISAMBIGUATION (UNÜBERWACHTE SYSTEME) ---- 18
2.4 KNOWLEDGE-BASED DISAMBIGUATION (WISSENSBASIERTE SYSTEME) ---- 23

3 DATEN ---- **27**

3.1 COINCO – KORPUS ---- 27
3.2 GIGAWORD – KORPUS ---- 28
3.2.1 Annotated Gigaword – Korpus ---- *30*

4 REALISIERUNG DES ANSATZES ---- **33**

4.1 HERANGEHENSWEISE ---- 33
4.2 ALGORITHMUS ---- 35
4.3 EVALUIERUNG ---- 37

5 ZUSAMMENFASSUNG ---- **43**

ABBILDUNGSVERZEICHNIS ---- **44**

TABELLENVERZEICHNIS ---- **45**

LITERATURVERZEICHNIS ---- **46**

1 Einleitung

1.1 Motivation und Zielsetzung

Nach (Carstensen 2012) „befasst sich die Sprachtechnologie mit der Entwicklung marktreifer Anwendungen der maschinellen Sprachverarbeitung bzw. der Computerlinguistik“. Darunter ist beispielsweise zu verstehen, dass automatische Computersysteme in der Lage sind, Informationen auf Anfragen eigenständig durchzuführen und Antworten zu liefern. Die Hauptaufgabe besteht darin, Methoden und Verfahren zu entwickeln, durch die das System Funktionen des natürlichen Sprachverhaltens erkennt, erlernt und anwendet (Buechel 2010). Betrachtet man als klassisches Beispiel die maschinelle Übersetzung von einzelnen Wörtern, ganzen Texten oder Ausdrücken, so bereitet die *Ambiguität* der Wörter einer natürlichen Sprache Schwierigkeiten. Die eindeutige Interpretation sowie das Auflösen der Mehrdeutigkeit natürlichsprachlicher Begriffe wird durch eine Reihe von unterschiedlichen Faktoren beeinflusst, wie zum Beispiel von Sprache, Kontext, Sachverhalt, Thema, Satzzusammenhang oder Beziehung der Wörter im Text und im Satz. Natürlich betrifft es nicht nur das Problem der maschinellen Übersetzung, sondern auch andere Teilgebiete in der Computerlinguistik, wie Textklassifikationen, Spracherkennung, maschinelle Suche, automatische Datenextraktion aus Texten sowie Information Retrievel (Carstensen 2012).
Die Herausforderung für das Auflösen der Wortmehrdeutigkeit (engl. Word Sense Disambiguation, WSD[1]) und Präzisierung der Bedeutung besteht darin, dass automatische Systeme, die Semantik der von ihnen verarbeiteten Inhalte nicht richtig verstehen und bearbeiten können. Im Gegensatz zum Menschen fehlen maschinellen Computersystemen das gewisse Weltwissen und das situative Wissen[2], welches das Verständnis für den Inhalt und den kontextuellen Wortsinn der ambigen Wörter erschwert. Das menschliche Wesen hat die Fähigkeit, sich die lexikalischen Bedeutungen der Begriffe herzuleiten, welches bei maschinellen Systemen nicht der Fall ist.
Die ersten Gedanken im Bereich der Computerlinguistik oder Sprachtechnologie führte dazu, die natürliche Sprache maschinell zu verarbeiten. Heute ist dieser, auch maschinelle Sprachverarbeitung (natural language processing, NLP) genannter Bereich stark ausgewachsen und fortgeschritten. Trotz der rapiden Entwicklung der maschinellen Sprachverarbeitung und der natürlichsprachlichen Systeme (natural language systems, NLS) gibt es noch in vielen Bereichen der Informatik und der Computerlinguistik Lücken. Eines der Hauptprobleme der NLP Anwendungen ist, wie soeben genannt, die Mehrdeutigkeit vieler Begriffe (Carstensen 2012). Die jahrelange Forschung in der Sprachwissenschaft hat gezeigt, dass kein universeller Disambiguierungsalgorithmus existiert, der das WSD-Problem im Ganzen lösen und beheben kann. Stattdessen werden verschiedenen Methoden und Verfahren kombiniert und angewandt. Eine Verallgemei-

[1] WSD – Auflösung sprachlicher Mehrdeutigkeit auf der Wortebene
[2] Situatives Wissen - Wissen, welches aus Erlebnissen und gesammelten Erfahrungen gewonnen wird.

nerung einer solchen Methode ist bis heute noch nicht gelungen, da jede Sprache ihre eigene Struktur und Funktionsweise aufweist.
Der traditionelle Ansatz im Umgang mit der lexikalischen Mehrdeutigkeit liegt in der Wortbedeutungsdisambiguierung. Ziel dieses Ansatzes ist die Identifizierung der eindeutigen Bedeutung eines Wortes unter Berücksichtigung der Kontextinformationen. Neben den Kontextinformationen sind hierbei semantische Ressourcen für die Disambiguierung notwendig. Diese sind beispielsweise lexikalische Datenbanken wie *WordNet*[3] (Kapitel 1.3.2), welche semantische und lexikalische Beziehungsinformationen zwischen Wörtern enthält und insbesondere ein Bedeutungsinventar zur Verfügung stellt. Unter Zuhilfenahme dieses Inventars weist nun ein Disambiguierungssystem jedem ambigen Zielwort die passenden Bedeutung zu (Carstensen 2012). Eine alternative Herangehensweise ist die „*lexikalische Substitution* "(Kapitel 2.1), eine kontextbezogene Umschreibung, bei dem nur ein Wort ersetzt wird. Dieses ist ein relativ neues Paradigma der Bedeutungsbeschreibung, das eng mit WSD verwandt ist. Die bereits existierenden Modelle in der Bedeutungsbeschreibung haben gezeigt, dass des Rankings der möglichen Substitute nur diese betrachtet werden, die für das spezifische Zielwort vorgesehen sind. In Bezug dazu entstand die Motivation für diese Bachelorarbeit. Diese Arbeit soll genau diese Lücke schließen, indem alle Wörter im Vokabular gerankt werden.

In dieser Arbeit wird insbesondere das Ziel verfolgt, die vokabular-globale lexikalische Substitution mit dem Modell von (Thater et al. 2011) umzusetzen und zu ermitteln, um wie viel schlechter das Modell wird, wenn es *alle* potentiellen Substitute ranken soll. In der Arbeit von (Thater et al. 2011) werden drei unterschiedliche Verfahren zur Modellierung präsentiert. Die *No-Contextualization, Strict Contextualization* und *Similarity-Based-Contextualization.* Diese Bachelorarbeit wird das Modell mit dem *Similarity-Based-Contextualization* Verfahren untersuchen und anwenden. Das Modell wird mit den Daten aus dem Gigaword Korpora (Kapitel 3.2) trainiert und anschließend wird das Verhalten des Modells bei Anwendung auf den Stuttgarter Lexical Substitution Daten (CoInCo[4]-„Concept in Context") (Kapitel 3.1) mit allen potentiellen Substituten analysiert und ausgewertet. Das Modell wird in zweierlei Hinsicht auf dem CoInCo Datensatz quantitativ evaluiert - zuerst nur auf den für dieses Zielwort genannten Substituten und zusätzlich auf *allen* für beliebige Zielwörter genannten Substituten. Hier ist es wichtig zu sehen, wie sich das Modell verglichen zu den in Kremer at al. 2014 berichteten Zahlen (Generalized Average Precision, kurz GAP) verhält. Darüber hinaus wird die qualitative Evaluation der False Positives durchgeführt. Dabei ist entscheidend zu wissen, welche Wörter fälschlicherweise als Substitute vorausgesagt werden, obwohl sie es nicht sind und ob es sich eventuell um mögliche, aber nicht genannte Substitute handelt oder um echte Fehler.

[3] http://wordnet.princeton.edu/wordnet/
[4] CoInCo - http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/coinco.html (27.11.2014 17:00)

1.2 Gliederung

Die Bachelorarbeit ist in folgende Kapitel untergliedert:

Kapitel 1 – Einleitung: Dieses Kapitel beschreibt die Motivation dieser Bachelorarbeit und bietet einen Überblick hinsichtlich der Aufgabenstellung und der Ziele.

Kapitel 2 – Stand der Forschung: In diesem Kapitel wird zu Beginn der Begriff der *lexikalischen Substitution* detailliert mit aktuellen Arbeiten erläutert. Anschließend werden die grundlegenden Verfahren in der gegenwärtigen wissenschaftlichen Forschung eingeführt. Dabei werden drei allgemeine Ansätze der Bedeutungsbeschreibung und -unterscheidung näher erläutert.

Kapitel 3 – Daten: Erläutert die für diese Bachelorarbeit relevante Datensätze. Dabei wird insbesondere auf ihre Struktur und Größe eingegangen und inwiefern diese zum Einsatz kamen.

Kapitel 4 – Realisierung des Ansatzes: In diesem Kapitel wird die Umsetzung der lexikalischen Substitution mit dem unüberwachtem Modell von (Thater et al. 2011) umgesetzt und auf den CoInCo-Datensatz evaluiert. Zum Schluss werden die dabei resultierenden Ergebniss mit den in (Kremer et al. 2014) berichteten Zahlen verglichen.

1.3 Grundlagen der Lexikalischen Ambiguität

Dieses Kapitel soll zunächst einen Einblick in die Problematik der Bedeutungsbeschreibung geben und relevante Begriffe definieren, die einen Überblick über historische Entwicklungen vermitteln. Des Weiteren wird auf die Grundlage der praktischen Bedeutungsbeschreibung eingegangen. Dabei wird die lexikalischen Datenbank *WordNet* beschrieben.

1.3.1 Ambiguität

Nach Hadumod Bußmanns „Lexikon der Sprachwissenschaft“ wird Ambiguität wie folgt definiert:

> ***„Ambiguität*** *[lat. ambiguitás ›Doppelsinn‹. Auch: Amphibolie (veraltet), Mehrdeutigkeit]. Eigenschaft von Ausdrücken natürlicher Sprachen, denen mehrere Bedeutungen zukommen. Ambige Ausdrücke sind (isoliert betrachtet) semantisch unbestimmt und folglich präzisierungsbedürftig. Ambiguität zeichnet sich dabei gegenüber -> Vagheit*[5] *dadurch aus, dass das Präzisierungsspektrum als diskret wahrgenommen wird (Bsp. Bank: Lesart 1 = ›Geldinstitut‹, Lesart 2 = ›Sitzgelegenheit‹ usw.), während vage Ausdrücke (z.B. Farbadjektive, Gradadjektive) über ein kontinuierliches Präzisierungsspektrum verfügen.“* (Bußmann 2008)

Des Weiteren gliedert Hadumod Bußmann Ambiguität in vier Komponenten (Bußmann 2008):

- Lexikalische Ambiguität (Mehrfachbedeutung von Lexemen)
- Syntaktische Ambiguität
- Skopusambiguität
- Relationale Ambiguität

Für die Bedeutungsbeschreibung kommt jedoch nur die erste Komponente, die lexikalische Ambiguität in Frage. Bei dieser Art der lexikalischen Ambiguität wird in Homonymie und Polysemie unterschieden.

P. Edmonds definiert ein Spektrum von Unterschieden in der Wortbedeutung in Bezug auf die Granularität (Edmonds 2006). Edmonds gliedert die Ambiguität in eine Hierarchie von grobkörnig bis feinkörnig ein (siehe Abbildung 1).

[5] „pragmatische Unbestimmtheit“ (Bußmann 2008)

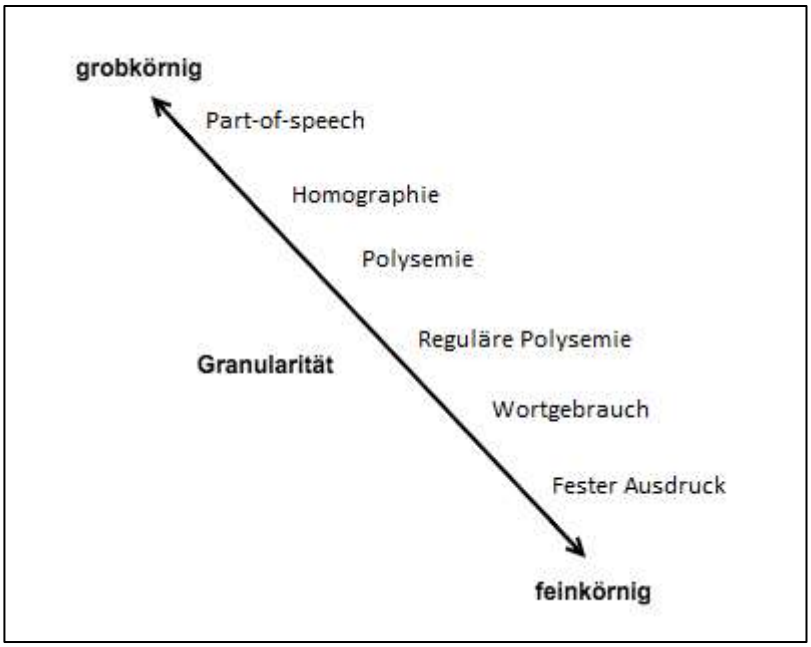

Abbildung 1: Wortbedeutung in Bezug auf Granularität (Edmonds 2006)

Im grobkörnigen Ende des Spektrums kann ein Wort eine kleine Anzahl von Bedeutungen haben die deutlich unterschiedlich sind. Je mehr man sich aber dem feinkörnigen Ende nähert, werden die grobkörnigen Bedeutungen eines Wortes in eine komplexe Struktur von miteinander zusammenhängenden Bedeutungen aufgelöst. Ein Wort besitzt genau dann eine Part-of-speech Ambiguität, wenn es in mehr als in einer Wortart auftritt. Beispielsweise ist „*sharp*" im Zusammenhang „*having a thin edge*" ein Adjektiv, in Bezug auf „*a musical notatiton*" ein Nomen, ein Verb bei „*to raise in pitch*" und ein Adverb bei „*exactly*". Part-of-speech Ambiguität weist nicht notwendigerweise die verschiedenen Bedeutungen eines Wortes auf. Dies kann aber durch „*Part-Of-Speech tagging*"[6] aufgelöst werden. In der Mehrzahl der WSD-Systeme wird „*Part-Of-Speech tagging*" als erster Schritt verwendet, sodass der WSD-Algorithmus die Wortartambiguität fokussiert (Edmonds 2006). Aus diesem Grund sind solche Aufgaben für Computersysteme schwieriger zu lösen, da sie das benötigte Hintergrundwissen nicht besitzen. Die Mehrdeutigkeit kann somit nur durch Kenntnis und Berücksichtigung der relevanten Hintergrundinformation und bestimmten Verfahren korrekt aufgelöst werden.

1.3.2 WordNet

Die externe Wissensquelle *WordNet* nimmt als Grundlage in der Bedeutungsbeschreibung ihren Platz ein. Viele Ansätze der Bedeutungsbeschreibung nutzen diese Quelle als Wortbedeutungsinventar. Sie dient in der Wortbedeutungsunterscheidung als alternative Datenquelle, die als Hilfsmittel zur Repräsentation der Bedeutung eines Wortes dient. WordNet ist eine umfangreiche lexikalische Datenbank der englischen Sprache, welche eine große Anzahl englischer Begriffe (Nomen, Verben, Adjektive und Adverbien) ent-

[6] part-of-speech tagging = Wortart-Annotierung bzw. die Zuordnung von Wörtern und Satzzeichen eines Textes zu Wortarten

hält. WordNet 3.0 umfasst insgesamt 155,287 Wörter. Außerdem beinhaltet sie zu jedem dieser Wörter eine Beschreibung ihrer Bedeutung, sowie semantische und syntaktische Relationen dieser Bedeutungsvarianten. Die vier unterschiedlichen Wortarten werden dabei auf der Grundlage ihrer Synonyme in eine Menge gruppiert. Diese Gruppierungen werden *Synsets* genannt. Die Synsets werden dabei durch semantische und lexikalische Beziehungen miteinander verknüpft. In WordNet 3.0 existieren 117,659 Synsets. Ein Wort-Synset Paar ist durch seine Wortart, Definition und einer Reihe von Beispielsätzen definiert. Die Mehrheit der Begriffe sind Substantive, die über 117,000 Begriffe und 82,000 der Synsets umfassen. Die Mehrheit der mehrdeutigen Wörter sind Verben, gefolgt von Adjektiven und Substantiven. Die durchschnittliche Anzahl der Synsets für ein Verb ist 2.17, während die durchschnittliche Anzahl der Synsets für ein Substantiv bei 1.24 liegt. Die Tabelle 1 zeigt die Aufteilung der Wörter, Synsets und Polysemie nach ihren Wortarten.

Part-of-speech	# Word	# Synsets	# Word-Synset Pair	Average # Synsets
Noun	117,789	82,115	146,312	1.24
Verb	11,529	13,767	25,047	2.17
Adjective	21,479	18,156	30,002	1.40
Adverb	4,481	3,621	5,580	1.25
Total	155,287	117,659	206,941	1.52

Tabelle 1: Aufbau von WordNet[7]

Die semantischen Beziehungen, die Synsets verknüpfen, sind Hyperonyme (Oberbegriffe), Hyponyme (Unterbegriffe), Meronyme (Teilbegriffe) und Holonyme. Die Beziehung zwischen den Synsets tritt nur in ihren jeweiligen Wortarten auf. Bei zwei Wort-Synset Paaren mit verschiedenen Wortarten würde es keine semantische Beziehung zwischen ihnen geben. Dies schafft vier unterschiedliche Hierarchien innerhalb WordNet, eine für jede der vier Wortarten. Diese Hierarchien sind nicht miteinander verbunden.
Wie bereits erwähnt, wird die lexikalische Datenbank in einer Vielzahl von WSD-Methoden als Datenquelle eingesetzt, um Wortmehrdeutigkeiten zu disambiguieren. Beispielsweise wird dabei für eine Instanz eines Wortes unter Zuhilfenahme des Bedeutungsinventars, hier WordNet, die richtige Bedeutung für das Zielwort vorhergesagt. Ein solches „Wissen“ ist für die Bedeutungsbeschreibung grundlegender Bestandteil, welches nicht wegzudenken ist. Wissensressourcen liefern Daten, mit dem die wesentliche Bedeutung eines Wortes assoziiert werden kann (Navigli 2009). An dieser Stelle tritt der hauptsächliche Unterschied zwischen *WSD* und *lexikalische Substitution* ein. Die lexikalische Substitution ist, stellt eine andere Möglichkeit dar, die Auflösung der Wortmehrdeutig auf Wortebene zu beheben, ohne dabei vordefinierte und externe Wissensquellen zu benutzen. Dieses wird im Kapitel 2.1 detailliert beschrieben.

[7] http://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html

1.3.3 Disambiguierung

Um die Mehrdeutigkeit aufzulösen, muss primär das Wissen über mögliche Bedeutungen vorhanden sein. Dieses Wissen kann zum Beispiel aus Wörterbüchern oder Trainingsdaten entnommen werden (Manning & Schütze 1999). Die Disambiguierung ist „der Vorgang und das Ergebnis der Auflösung lexikalischer und struktureller Mehrdeutigkeit sprachlicher Ausdrücke durch den sprachlichen oder außersprachlichen Kontext“ (Bußmann 2008). Dabei „erfolgt in der Regel die sprachliche Disambiguierung auf der lexikalischen Ebene durch das Ausschließen von semantisch unverträglichen Lexemverbindungen“ (Bußmann 2008).

> *„Er sah das Schloss vor sich liegen* wird durch den Zusatz *und hob es auf* als ›Vorrichtung zum Schließen‹ im Unterschied zu ›Gebäude‹ disambiguiert.“ (Bußmann 2008)

Die Disambiguierung bei struktureller Mehrdeutigkeit „erfolgt durch explizite Ausformulierung der zugrunde liegenden Strukturen“.

> *„So sind die beiden Lesearten des Satzes -Die Wahl der Vorsitzenden fand Zustimmung- zu disambiguieren durch die Paraphrasen P_1 -Dass die Vorsitzenden gewählt wurden, fand Zustimmung- bzw. P_2 -Die Wahl, die die Vorsitzende getroffen hatte, fand Zustimmung“.* (Bußmann 2008)

Außerdem beschreibt (Bußmann 2008) die Disambiguierung durch den außersprachlichen Kontext. Diese Art von Disambiguierung wird durch eine Reihe von unterschiedlichen Faktoren beeinflusst, wie zum Beispiel von der Sprechsituation, dem Vorwissen, der Einstellung, den Erwartungen der Sprecher/Hörer sowie von der Mimik und Gestik. In der Arbeit von (Manning & Schütze 1999) wird beschrieben, dass sich die WSD auf die lexikalische Mehrdeutigkeit im sprachlichen Kontext beschränkt, obwohl unter einigen Umständen auch das Vorwissen über das Thema oder anderen Disambiguierungen im Text genutzt werden können.
In der Literatur wird zwischen den beiden Begriffen *Word Sense Disambiguation* und *Word Sense Discrimination* (Unterscheidung von Wortbedeutungen) differenziert. Zur besseren Verständlichkeit soll daher an dieser Stelle eine Abgrenzung beider Prozesse erfolgen:

- ***Word Sense Disambiguation*:** Nach (Schütze 1998), ist die Aufgabe von WSD, den verschiedenen Vorkommen von ambigen Wörtern eine Bedeutung zuzuordnen. Das Ziel dabei ist die Festlegung der Bedeutung eines ambigen Wortes in einem bestimmten Kontext. Pedersen und Mihalcea definieren WSD als Problematik der Auswahl einer Bedeutung für ein Wort aus einer Menge von vordefinierten Möglichkeiten bzw. eines expliziten Bedeutunginventars. Die verschiedenen Bedeutungen eines Wortes kommen in der Regel aus einem Wörterbuch oder Thesaurus. Die Repräsentation der Bedeutung eines Wortes erfolgt dabei entwe-

der in Bezug auf ein Wörterbuch oder auf eine Übersetzung in eine zweite Sprache (Pedersen & Mihalcea 2005). Der hauptsächliche Unterschied zur lexikalischen Substitution und zur Word Sense Discrimination ist dabei, dass das Bedeutungsinventar explizit festgelegt und vordefiniert ist.

- ***Word Sense Discrimination*:** Mit Hilfe der *Word Sense Discrimination* wird das gemeinsame Auftreten von Wörtern, je nach Sinngehalt in unterschiedliche Klassen gruppiert (im Sinne von „Clustering"). Dabei wird für jede Wortinstanz festgelegt, zu welcher Klasse sie zugeordnet wird. Die Elemente einer Klasse haben dieselbe Bedeutung (Schütze 1998). Die Bestimmung der Wortbedeutung erfolgt hierbei durch den Bezug auf den lokalen Kontext (Pedersen & Mihalcea 2005). Das bedeutet, dass das Bedeutungsinventar hierbei nicht a priori gegeben ist, sondern durch das Clustering der Daten entsteht.

2 Stand der Forschung

Zu Beginn dieses Abschnitts wird der Begriff der *lexikalischen Substitution (LexSub),* sowie wichtige Arbeiten der aktuellen LexSub-Forschung vorgestellt, die sich der Erzeugung der potentiellen Substitute widmen. Bisherige Arbeiten im Bereich der lexikalischen Substitution befassen sich entweder mit der Erzeugung und anschließendem Ranking der potentiellen Substitute oder nehmen sich nur der letzteren Aufgabe, dem Ranking, an. Nachfolgend werden drei allgemeine Methoden der Bedeutungsbeschreibung und -unterscheidung beschrieben. Dabei wird auf den Unterschied zu LexSub eingegangen.

2.1 Lexikalische Substitution

Die *lexikalische Substitution* bietet einen alternativen Ansatz zur Wortbedeutungsdisambiguierung. Die Instanzenbedeutung bei der *lexikalischen Substitution* wird nicht per Referenz auf Synsets repräsentiert, sondern als Menge an möglichen Substituten (McCarthy & Navigli 2009). Die grundlegende Gemeinsamkeit, die lexikalische Substitution und WSD miteinander verbindet, ist, dass beide sich mit der Identifizierung eines Substituts für ein bestimmtes Zielwort unter Berücksichtigung des Kontextes beschäftigen, ohne dabei die eigentliche Bedeutung des Satzes zu ändern. Der hauptsächliche Unterschied beider Ansätze liegt darin, dass die lexikalische Substitution das Bedeutungsinventar an Substituten und die Informationsquelle nicht fest vordefiniert, beziehungsweise explizit vorschreibt (McCarthy & Navigli 2009). Stattdessen werden Kommentatoren gebeten, für jede Instanz eines Wortes mehrere alternative Wörter oder Sätze aufzulisten, die für das Zielwort im jeweiligen Kontext als Substitut verwendet werden können (Kremer et al. 2014). Durch diese Eigenschaft wird das Problem der Granularität von Bedeutungsdifferenzierung überwunden (McCarthy & Navigli 2009), (Kremer et al. 2014). Anstatt Ontologien zu verwenden stellt sie eine *„bottom-up“* und korpusbasierte Charakterisierung der Wortbedeutung bereit (Kremer et al. 2014). Die Hauptaufgabe besteht in der Generierung und im Ranking der möglichen Substitute. Während der Generierung wird ein Inventar von möglichen Substituten für die jeweiligen Zielwörter erzeugt und ist daher für jedes Wort im Vokabular anwendbar (McCarthy & Navigli 2009). Das Ranking hingegen, befasst sich mit der Charakterisierung der Substitute in Bezug auf den Kontext und erstellt dabei eine Rangliste bezüglich der Güte der Substitute im jeweiligen Kontext.
Der traditionelle Ansatz, das Problem der lexikalischen Ambiguität zu beheben, wurde bisher mit der üblichen WSD-Methode (Kapitel 1.3.3 und 2.2) angegangen (McCarthy 2009), (Navigli 2009). Bei dieser Herangehensweise ist die Aufgabe ein Lemma-Level Klassifikationsproblem. Hierbei wird die Herausforderung durch sogenannte Trainingsklassifizierer überwunden, bei dem die Lemma-Instanzen mit ihren korrekten Bedeutungen gekennzeichnet sind. Dieser Ansatz hat jedoch grundlegende Probleme gezeigt (Kremer et al. 2014):

- Eine vollständige und konsistente Reiche von Bezeichnungen wurde vorausgesetzt
- WSD erfordert für jeden Satz und Lemma eine Annotation

Eine mögliche Lösung in Bezug auf diese Probleme wird in der Arbeit von (McCarthy & Navigli 2009) vorgestellt. Ihre Idee bestand in der Verwendung der *Lexikalischen Substitution.* Hierdurch wird vermieden, dass die Wortbedeutung durch eine einzelne Bezeichnung erfasst wird. Außerdem bietet sie eine Reihe von Vorteilen gegenüber der traditionellen WSD-Methode. In ihrer Forschungsarbeit „The English lexical substitution task" präsentieren sie den sogenannten *Gold Standard*[8]*,* der in der Linguistik als Datensatz bezeichnet wird, dessen Anreicherung mit zusätzlicher syntaktischer und semantischer Information (Annotation) von Experten für gut befunden wird und welcher sich folglich für Evaluationszwecke gut eignet. Diesen Datensatz haben sie bei *SemEval 2007*[9] für die lexikalische Substitution eingeführt. Ihr Ziel dabei war es, eine Evaluation bereitzustellen, bei dem der Bedeutungsinventar nicht vorverarbeitet oder vordefiniert ist. In ihrer Arbeit wird mittels lexikalischer Substitution ein passendes Substitut für das vorgesehene Zielwort in Abhängigkeit des Kontextes gefunden. Die Motivation ihrer lexikalischen Substitutionsaufgabe lag darin, die Wortbedeutung darzustellen, bei dem kein vordefiniertes Bedeutungsinventar verwendet wird. Die Lexikalische Substitution beinhaltete dabei zwei Prozesse:

- Das Auffinden einer Menge aller potentiellen Substitute für das Wort
- Das Auffinden des besten Substituts in Bezug auf den Kontext

Diesbezüglich hat man Annotatoren mit einbezogen, ein Substitut für das jeweilige Zielwort in Bezug zum Kontext zu finden, bei dem die Bedeutung so gut wie möglich zum Original bewahrt wird. Die Substitute enthalten dabei zusätzlich noch ein Gewicht, welches der Häufigkeit entspricht, wie oft dieses Substitut genannt wurde. Für die Annotation wurde keine vordefinierte Liste von Begriffen zur Verfügung gestellt. Die Berechnung der Übereinstimmung wurde anhand von zwei Maßnahmen durchgeführt.

- Übereinstimmung zwischen den Annotatoren
- Übereinstimmung mit der häufigsten Antwort

Die Daten, die (McCarthy & Navigli 2009) dabei verwendet haben, wurden aus dem „English Internet Corpus"[10] (EIC) entnommen, welcher von (Sharoff 2006) vorgefertigt wurde. Der EIC-Datensatz ist frei verfügbar und beinhaltet insgesamt 2010 Sätze. Davon sind 201 Zielwörter mit jeweils zehn Sätzen. In Abbildung 2 ist ein Auszug aus dem Gold-Standard-Datensatz für das Zielwort *charge* zu sehen. Im diesem Auszug sind fünf

[8] http://www.informatics.sussex.ac.uk.research/groups/nlp/mccarthy/task10index.html.
[9] Bei SemEval ehemals SENSEVAL, handelt es sich um eine Serie von fortlaufenden Workshops, welche sich mit der Evaluation von Systemen befassen, die sich mit der Interpreatation natürlcher Sprache befassen.
[10] http://corpus.leeds.ac.uk/

verschiedene Kontextsätze mit unterschiedlicher Bedeutung des Zielwortes aufgelistet, bei dem die jeweiligen Instanzen mit unterschiedlichen *ids* repräsentiert werden. Die potentiellen Substitute für das Zielwort im jeweiligen Kontextsatz sind in Tabelle 2 dargestellt.

```
<lexelt item="charge.v">
    <instance id="361">
        <context>Annual fees are <head>charged</head> on a pro-rata basis to
                correspond with the standardised renewal date in December .</context>
    </instance>
    <instance id="362">
        <context>Meanwhile , George begins obsessive plans for his
                funeral ... George , suspicious , <head>charges</head> to her
                room to confront them .</context>
    </instance>
    <instance id="363">
        <context>Pauline Gilmore , 32 , was <head>charged</head> with possessing
            a blast bomb , 14 bullets and 21 explosive pipe darts in a field at
            Drumcree on Wednesday morning .</context>
    </instance>
    <instance id="364">
        <context>Plug in you h 10 in the usb outlet and it will <head>charge</head>
                without the plug in adaptor .</context>
    </instance>
    <instance id="365">
        <context>U.S. Nevada trooper charged with reckless driving , manslaughter
                A state trooper was <head>charged</head> Monday with nine felony
                counts of reckless driving and involuntary manslaughter in a crash
                that killed four people .</context>
    </instance>
```

Abbildung 2: Kontextsätze für das Zielwort "charge"

Zielwort	Satz-ID	Wortart	Potentielle Substitute mit Häufigkeit
charge	**361**	**verb**	**levy 2, require 1; impose 1; demand 1;**
charge	**362**	**verb**	**run 2; rush 2; storm 1; dash 1;**
charge	**363**	**verb**	**indict 3; accuse of 2; accuse 1;**
charge	**364**	**verb**	**recharge 2 ; supply elevtricity 1; charge up 1;**
charge	**365**	**verb**	**indict 3; accuse of 2; accuse 1;**

Tabelle 2: SemEval 2007 - Gold Standard Substitute

Eine weitere lexikalische Substitutions-Aufgabe präsentieren (Kremer et al. 2014). Sie stellen den ersten großen englischen Korpus für die lexikalische Substitution aller Inhaltswörter dar, den sogenannten „all-words lexical substitution“ Korpus, welche unter *Concept in Context – CoInCo* (Kapitel 3.1) präsentiert wird. In ihrer Arbeit untersuchen sie die Art der Datensätze in der lexikalischen Substitution, indem sie diese mit den WordNet-Synsets (siehe 1.3.2) vergleichen. Die Besonderheit ihres Ansatzes liegt darin, dass sie alle Inhaltswörter betrachten und ersetzen. Der Hauptvorteil der Alle-Wörter-

Ersetzung ist, dass sie eine realistische Frequenzverteilung der Zielwörter und ihrer Bedeutung bereitstellt.
Sie verwenden diese, um empirisch:

- die Art der lexikalischen Substitute und
- die Art des Korpus (Wortbedeutung im Kontext)

zu bestimmen und zu untersuchen. Die Größe ihres Korpus, den sie hierbei zusammengestellt haben, bietet eine reichhaltige Quelle für die Untersuchung der Wortbedeutung. Die bisher vorhandenen Datensätze in der lexikalischen Substitution (McCarthy & Navigli 2009); (Sinha & Mihalcea 2014); (Biemann 2013);(McCarthy et al. 2013) sind entweder vergleichsweise zu klein oder lexikalische Beispieldatensätze. Die lexikalischen Beispieldatensätze bestehen aus Beispielsätzen für jedes Zielwort (siehe Abbildung 2). Dabei hat man nur für ein Zielwort in Bezug auf den jeweiligen Kontext Substitute generiert, wie in Tabelle 2 dargestellt. Im Vergleich zu (McCarthy & Navigli 2009) verwenden (Kremer et al. 2014) eine Teilmenge von „Manually Annotated Sub-Corpus[11]“ MASC (manuell-annotierten Teil-Korpus) für die Annotation. Dieser Datensatz zeichnet sich dadurch aus, dass er im Vergleich zu den bisherigen benutzen Datensätzen eine bessere Überdeckung der englischen Sprache sowie feinere Granularität aufweist. Der dabei verwendete Datensatz für lexikalische Substitution beinhaltet insgesamt Substitute für mehr als 30.000 Wörter. MASC ist ein Teil der Open American National Corpus[12], der in 19 Genres unterteilt ist und mit manuell erzeugter Annotation erstellt wurde, mit dem Ziel eines freizugänglichen Korpus. Man hat sich dabei auf zwei Genres für die Zielwörter spezifiziert, einmal auf *Zeitungsartikel (newspaper)* mit 18,942 Tokens und zum anderen auf *Romanliteratur (fiction)* mit 16,605 Tokens. Diese beiden Genres sind zum einen für die NLP relevant und bieten lange, einheitlichen Dokumente, die für die Annotation aller Wörter angemessen sind. Sie verwendeten die MASC Wortart Annotation, um alle Inhaltswörter (Verben, Nomen, Adjektive, und Adverbien) zu identifizieren, die insgesamt über 15.000 Zielwörter für die Annotation liefern. (Kremer et al. 2014) haben sich aus zwei Gründen für diesen Korpus entschieden:

- ihre Analysen können aus den bereits vorhandenen Annotationen profitieren
- sie können ihre Annotationen im Rahmen von MASC freigeben

(Kremer et al. 2014) sind der Annahme nachgegangen, dass durch diese Eigenschaften, ihr Korpus deutlich repräsentativer für einen Fließtext ist. Der dabei annotierte Datensatz umfasst insgesamt 168,143 Substitute für 15,732 Zielwörter in 2,482 Zielsätzen. Auf die Wortarten spezifiziert sind es 7,181 Nomen, 4,635 Verben, 2,487 Adjektive und 1,429 Adverbien. Wie oben erläutert, sind die Zielwörter in den beiden Genres Zeitungsartikel (8,103 Zielwörter in 990 Sätzen) und Romanliteratur (7,629 Zielwörter in 1,492 Sätzen) ausgeglichen. (Kremer et al. 2014) verwenden WordNet Version 3.1 als Quelle

[11] http://www.anc.org/data/masc/
[12] www.anc.org

sowohl für die lexikalischen Beziehungen als auch für Wortbedeutungen. (McCarthy & Navigli 2009) verwendeten in ihrer Arbeit WordNet Version 2.1. Die lexikalische Datenbank WordNet ist der de facto Standard in NLP und wird für die Bedeutungsbeschreibung sowohl auch für weitere Untersuchungen der Wortbedeutung verwendet (Navigli & Ponzetto 2012).
Diese Methode unterscheidet sich von (McCarthy & Navigli 2009) in zwei entscheidenden Punkten. Zu einem werden alle Instanzen jedes Zieles kommentiert und zum anderen werden alle Ziele unabhängig der Frequenz und der lexikalischen Mehrdeutigkeit miteinbezogen.

2.2 Supervised Disambiguation (Überwachte Systeme)

Die unterschiedlichen Methoden der Disambiguierung werden meist anhand ihrer verwendeten Daten und Quellen für Wortbedeutungsunterscheidung klassifiziert.

Die überwachten Systeme verwenden einen vordefinierten Bedeutungsinventar, wie beispielsweise WordNet, als Wissensquelle und manuell annotierte Trainingsdaten, in welchen eine Zuordnung zu den Klassen bereits vorgenommen wurde sowie die ambigen Wörter gekennzeichnet sind (McCarthy 2009). Ziel dabei ist es, dass der auf das Trainings-Korpus abgestimmte Algorithmus neue ambige Wörter erkennt und diese unter Berücksichtigung des Kontextes disambiguieren kann. Jedes Auftreten eines ambigen Wortes *w* wird mit der semantischen Bezeichnung gekennzeichnet. In der Regel erfolgt dies nach seiner kontextuellen Bedeutung (Manning & Schütze 1999). Die Instanzen in den Trainingsdaten werden manuell mit den entsprechenden Begriffen bzw. Bezeichnungen aus dem Bedeutungsinventar annotiert. Ein überwachter Lern-Algorithmus lernt aus diesen Begriffen den Kontext zu erkennen und dabei ein Modell zu erzeugen, welches verwendet wird, um automatisch Begriffe zum Zielwort in den Testdaten zuzuordnen. Das System lernt aus den Trainingsbeispielen (McCarthy 2009). Überwachte Lernmethoden erreichen generell eine sehr hohe Disambiguierungsgenauigkeit und übertreffen somit andere WSD-Methoden. Der Nachteil dieses Verfahrens ist jedoch, dass manuell annotierte Trainingsdaten für jedes zu disambiguierende Wort erforderlich sind. Deswegen ist das ein sehr arbeitsintensiver und zeitaufwendiger Prozess. Es gibt bereits Arbeiten, die versuchen, diese Trainingsdaten automatisch zu erstellen. Eine bekannte Methode wurde von (Yarowsky 1995) vorgestellt und eine weitere von (Carol Friedman 2008).
Abbildung 3 zeigt ein allgemeines Modell einer solchen überwachten WSD-Methode. In diesem Verfahren dienen die manuell annotierten Trainingsdaten als Eingabe in das „Evaluation Module“. Das „Evaluation Module“ separiert diese Daten jeweils in den Trainings- und Testteil. Die jeweiligen Informationen über die Wortbedeutungen, welche zu den ambigen Wörtern zuvor annotiert wurden, werden aus dem Testteil entfernt. Nachdem diese entfernt wurden, werden beide Datensätze zum „Vector Creation Program“ gesendet. An dieser Stelle tritt ein Unterschied zu LexSub auf. Bei der lexikalischen Sub-

stitution wird ein solches vordefiniertes Bedeutungsinventar und annotierte Trainingsdaten (siehe Abbildung 3) nicht explizit verwendet und ist somit für jedes Wort im gesamten Vokabular anwendbar.

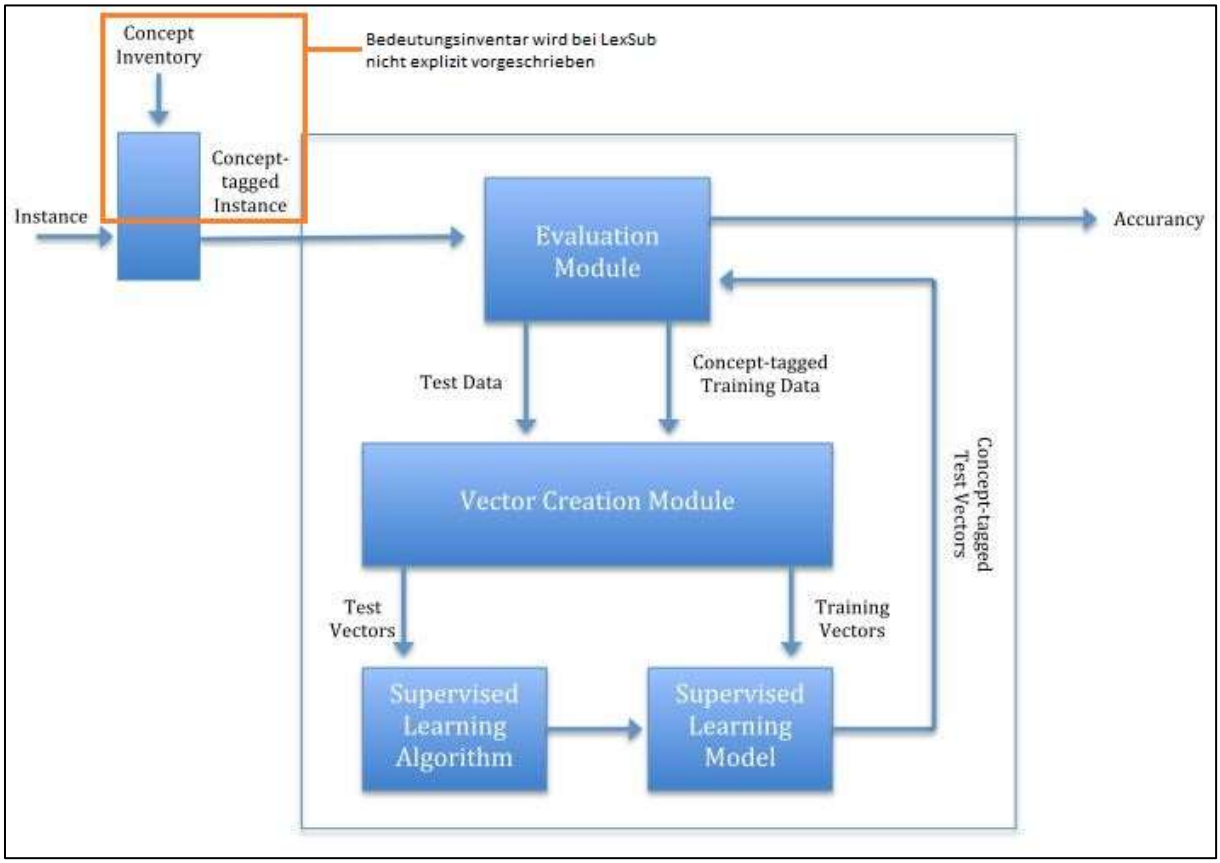

Abbildung 3: Supervised WSD Methode

Das „Vector Creation Module“ extrahiert die Features aus den Trainingsdaten und erzeugt für jede Instanz der manuell annotierten Trainingsdaten einen Trainingsvektor und einen Testvektor für jede Instanz der Testdaten. Der überwachte Lern-Algorithmus nimmt die Trainingsvektoren als Eingabe und lernt anschließend den Kontext, in welchem jede der möglichen Wortbedeutungen eingesetzt wird. Dieser Algorithmus erzeugt ein Model, welcher die Testvektoren als Eingabe erhält und jedem dieser Vektoren ihre entsprechende Bedeutung zuordnet. Das „Evaluation Program“ entnimmt diese Vektoren als Eingabe und berechnet die Genauigkeit dieses Models. Es gibt eine Vielzahl verschiedener überwachter Lern-Algorithmen, die in überwachten WSD Methoden verwendet wurden. Zum Beispiel *Support Vector Machines (SVMs), Klassifizieurng nach Bayes (Naives Bayes classifier)* oder *Informationstheoretische Annäherung (Informations Theory)* (McInnes 2009).

Bei der lexikalischen Substitution hingegen, wird nicht anhand des Bedeutungsinventars und der manuell annotierten Trainingsdaten wortspezifisches Disambiguieren erlernt, sondern es wird mit einem großen, nicht annotierten Korpus trainiert. Dabei werden syntaktische und semantische Features gesammelt und extrahiert, um einen Klassifizierer zu bilden. (Szarvas et al. 2013) präsentieren ein überwachtes lexikalisches Substitutionssystem, bei dem sie ein delexikalisiertes, vokabular-globales Klassifikations-

modell erstellen. Bezüglich ihrer Klassifikationsart kann diese lexikalische Substitution für jedes Wort im Vokabular angewendet werden, da sie nicht für jedes Zielwort einen separaten Klassifizierer verwenden. In diesem Ansatz wird anstelle der Erlernung von wortspezifischer-Substitutionspatterns, das Training eines globalen Modells zur lexikalischen Ersetzung auf delexikalisierten Features in den Vordergrund gestellt. Außerdem zeigen sie in ihrer Arbeit, dass sie im Gegensatz zu den üblichen überwachten Ansätzen auch potentielle Substitute für die Zielwörter produzieren können, die nicht in den Trainingsdaten enthalten sind. Diese erreichen sie durch die Verwendung von delexikalisierten Features aus verschiedenen Bereichen, einschließlich aus lexikalisch-semantischen Ressourcen wie WordNet (Fellbaum 1998), Gold-Standard-Datensatz (McCarthy & Navigli 2009),TWSI[13] (Biemann 2013), der Verteilungsähnlichkeit, N-Grammen und syntaktischen Features auf der Grundlage von großen, unkommentierten Korpora. Sie verwenden für ihr Modell zwei große freiverfügbare Datensätze. Zum einen den Gold Standard Datensatz von (McCarthy & Navigli 2009) und zum anderen einen ähnlichen, aber größeren Datensatz „TWSI“ von (Biemann 2013). Der TWSI-Datensatz von Biemann beinhaltet 24,647 Sätze für insgesamt 1,012 Zielwörter. Für die potentiellen Substitute wurde WordNet als Ressource verwendet.

2.3 Unsupervised Disambiguation (Unüberwachte Systeme)

Im Gegensatz zu der überwachten Modellierungsart verwenden die unüberwachten Systeme, keine externen Quellen und Daten wie z.B. maschinell lesbare Wörterbücher, Begriffshierarchien oder annotierte Texte (Agirre et al. 2006). Es liegen keine Hinweise auf die Bedeutungen der Wörter vor, da weder lexikalische Quellen, noch ein Trainings-Set oder Kollokationssequenzen[14] zur Verfügung stehen (Manning & Schütze 1999). Es wird lediglich mit den Rohdaten aus dem unvorbereiteten Korpus gearbeitet. Anhand dieser Methode soll das System die Klassen für die Daten, wie beim Clustering selbst erstellen und erlernen. Der Unterschied liegt darin, dass man beim überwachten Lernen die tatsächliche Wortbedeutung für jedes Wort in den Trainingsdaten kennt, während bei dem unüberwachten Lernen die Wortbedeutung in den Trainingsdaten nicht bekannt ist. Demnach kann man das unüberwachte Lernen als eine Art „Clustering-Aufgabe“ ansehen und das überwachte Lernen als eine Klassifikations-Aufgabe (Manning & Schütze 1999). Unsupervised Disambiguation stellt demnach eher einen ähnlichen Ansatz zur Word Sense Discrimination dar. Dabei werden die Instanzen eines bestimmten Zielwortes gruppiert, sodass alle Instanzen, die mit dem Zielwort in Verbindung gebracht werden, in derselben Klasse sind. Ein Ansatz hierzu wurde von (Schütze 1998) vorgestellt. Ein Vorteil der Gruppierung ist, dass keine große Menge von manuell annotierten Trainingsdaten erforderlich ist. Das Kennzeichnen der Instanzen in den Trainingsdaten geschieht mit Hilfe von bestimmten „clustering“ Algorithmen und nicht durch manuelle Annotation, wie bei den überwachten Methoden. Der Nachteil ist, dass für jedes Wort,

[13] TWSI = „Turk Bootstrap Word Sense Inventory“ Datensatz von (Biemann 2013) , https://www.lt.informatik.tu-darmstadt.de/de/data/twsi-turk-bootstrap-word-sense-inventory/

[14] Kollokation – das gehäufte benachbarte Auftreten von Wörter, gemeinsames Auftreten der Wörter

welches zu disambiguieren ist, Trainingsdaten erforderlich sind und im Vergleich zum überwachten Verfahren dieses keine hohe Disambiguierungsgenauigkeit besitzt (McInnes 2009).
Bei der lexikalischen Substitution werden die sogenannten *Cluster* nicht erstellt. Anstelle dieser *Cluster*, werden alle potentiellen Substitute betrachtet, die zuvor bei der Generierung der Substitute als richtige für die jeweiligen Zielwörter annotiert wurden. Anschließend wird zwischen diesen Substituten die Ähnlichkeit berechnet. Diese kann zum Beispiel mithilfe der *semantischen Ähnlichkeit* umgesetzt werden (siehe Kapitel 4.2). Dabei wird durch den Kontext der einzelnen Wörter der Wert der Kosinus-Ähnlichkeit berechnet. Der daraus resultierende Wert liegt zwischen 0.0 und 1.0. Je ähnlicher diese Wörter zueinander sind, desto größer ist der Wert. Diese Herangehensweise wird in Kapitel 4 detailliert erläutert.

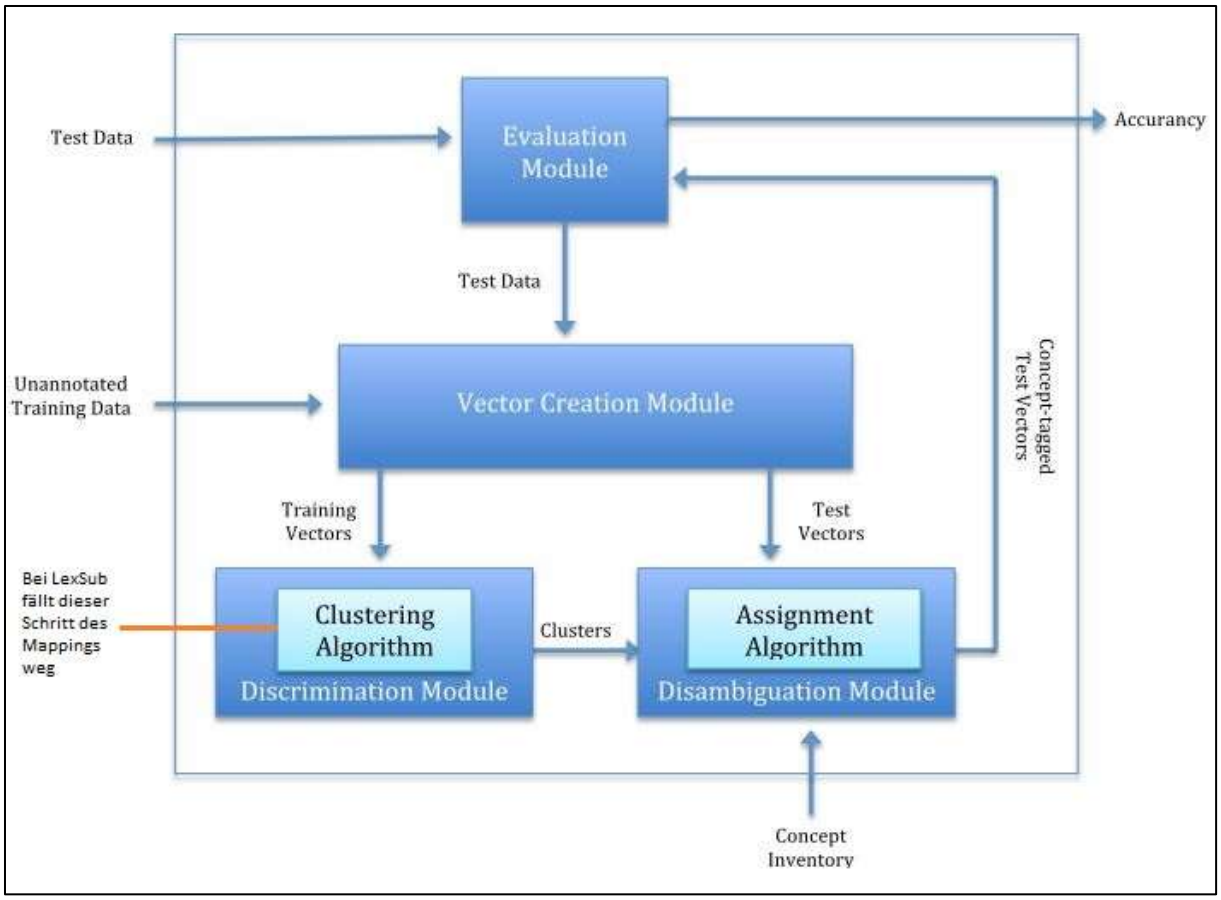

Abbildung 4: Unüberwachte WSD Methode

Abbildung 4 repräsentiert ein allgemeines Modell einer unüberwachten WSD Methode. Bei dieser Methode bekommt zu Beginn das „Evaluation Module“ die Testdaten als Eingabe. Möglicherweise wurden diese Daten für Evaluationszwecke annotiert. Wenn dies der Fall ist, werden diese Annotationen entfernt und so dem „Vector Creation Module“ zugesendet. Das „Vector Creation Module“ bekommt somit die Testdaten und eine Menge der nicht annotierten Trainingsdaten, die die Instanzen des Zielwortes enthalten, als Eingabe. Dann wird für jede Instanz der Trainings- und Testdaten ein Vektor erzeugt. Die erzeugten Trainingsvektoren werden anschließend zum „Discrimination Module“

gesendet und die Testvektoren zum „Disambiguation Module“. Im „Discrimination Module“ werden die Trainingsvektoren anhand eines „Clustering-Algorithmus“ gruppiert. Ein „Clustering-Algorithmus“ stellt die Vektoren in einem n-dimensionalen Raum dar und ordnet sie in sogenannte „Cluster“ (Gruppen) zusammen, wie in Abbildung 5 veranschaulicht. Es gibt einige unterschiedliche Typen von Clusteralgorithmen, wie zum Beispiel die agglomerierenden[15], trennenden oder partitionierenden Algorithmen.

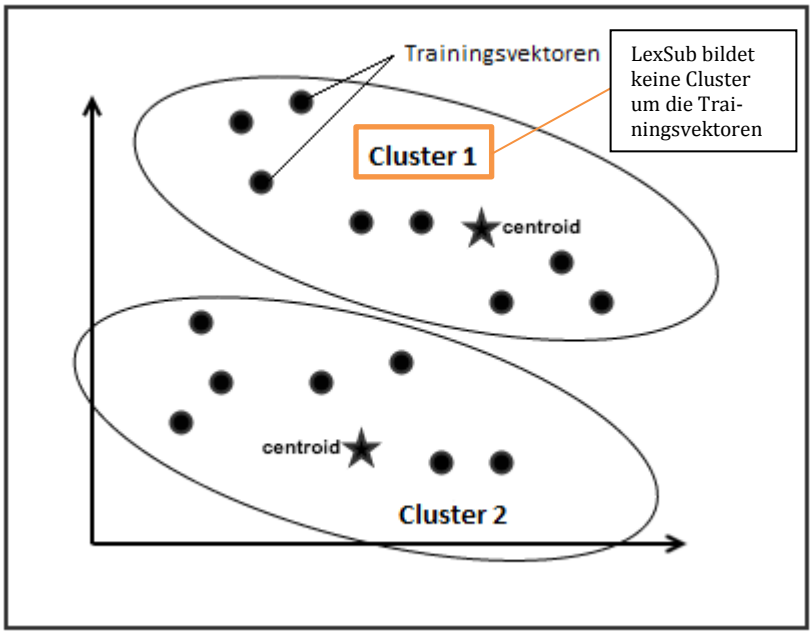

Abbildung 5: Beispiel des "Clustering"

Daraufhin werden die Cluster dem „Disambiguation Module“ übergeben. Anschließend weist der *„Assignment-Algorithmus“* (Zuweisungsalgorithmus) den einzelnen Clustern aus dem Bedeutungsinventar eine Bedeutung zu. Dies kann mit unterschiedlichen Methoden durchgeführt werden. Ein Ansatz, der durch (Wagstaff & Cardie 2000) beschrieben wurde, verwendet eine kleine Menge von annotierten Trainingsdaten, um die Zuweisungen zu den Clustern zu bestimmen. Weiter wird für jedes Clusters ein sogenannter Bedeutungsvektor erzeugt, indem der Zentroid des jeweiligen Clusters berechnet wird. Dies wird in Abbildung 5 durch einen Stern veranschaulicht. Ein Testvektor wird dann durch die Berechnung des Cosinus-Winkels zwischen seinem und jedem der möglichen Bedeutungsvektoren disambiguiert. Dies wird in Abbildung 6 dargestellt. Die Bedeutung, dessen Vektor den kleinsten Winkel zum Zielwort hat, wird diesem zugewiesen. Dies wird für jeden der Testvektoren durchgeführt und zum Schluss dem „Evaluation Module“ zurückgesendet, um die Genauigkeit des Systems zu bestimmen.
Im Gegensatz hierzu ist der Ablauf bei der lexikalischen Substitution nicht komplett identisch. Da bei der lexikalischen Substitution das Bedeutungsinventar nicht fest vorgeschrieben und verarbeitet ist, wird zuvor mit jedem Wort des gesamten Vokabulars anhand eines großen Korpus trainiert. Das bedeutet, dass dabei alle Substitute betrach-

[15] agglomerieren bedeutet anhäufen, ansammeln

tet werden und für jedes ein Vektor erzeugt wird. Daraufhin wird, wie bereits oben erwähnt, die Ähnlichkeit zwischen den Trainingsvektoren berechnet. Anhand dieser Ähnlichkeit werden in der Testphase die vorhergesagten Wörter für das zu substituierende Wort nach ihrer Ähnlichkeit geordnet. Aus diesem Grund entfällt hierbei der Schritt des Mappings. Die zuvor in der ersten Phase der LexSub-Aufgabe (Erzeugung der potentiellen Substitute) genannten richtigen Substitute, werden anschließend mit den Ergebnissen in der Liste der vorhergesagten Substitute für das zu ersetzende Zielwort verglichen. Hier geht es nicht wie in der allgemeinen WSD-Methode nur darum zu entscheiden, zu welcher Bedeutung das Zielwort zugewiesen wird, sondern wie sich richtig genannten Substitute zu den Ergebnissen nach dem Rankingverfahren verhalten. Für die lexikalische Substitution sind die unüberwachten Modelle von großer Bedeutung, da dabei keine abstrakten Bedeutungen für die ähnlichen Wörter vorausgesagt werden müssen und somit das Mapping jedes Mal entfällt. In Kapitel 4 wird die LexSub-Aufgabe mit dem unüberwachten Modell von (Thater et al. 2011) realisiert und evaluiert.

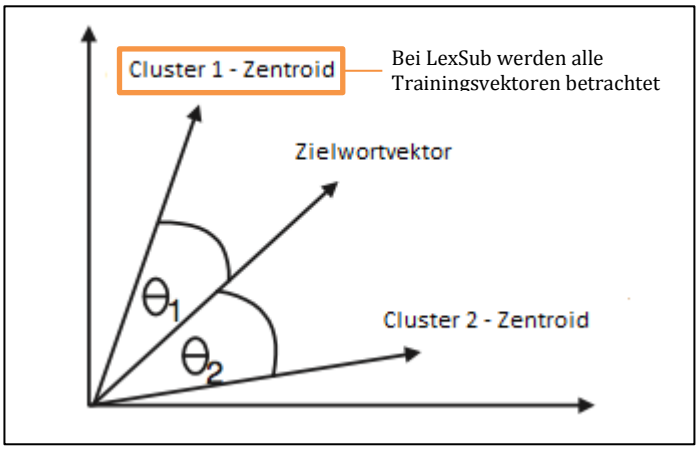

Abbildung 6: Beispiel des Zuweisungsalgorithmus

In der Fachliteratur existieren bereits Arbeiten, die auf den unüberwachten Ansatz basieren und als Verfahren für das Ranking in der lexikalischen Substitution verwendet werden. Die Gemeinsamkeit dieser Arbeiten besteht darin, dass Wörter durch Vektoren in einem Vektorraum dargestellt werden. In dem Ansatz von (Mitchell & Lapata 2008) beschreiben sie eine Herangehensweise, bei dem die Bedeutung einer Phrase und eines Satzes in einem Vektorraum repräsentiert wird. Zentraler Punkt dieses Ansatzes ist die Komposition von Vektoren. Die Komposition erfolgt dabei durch Addition und/oder Multiplikation. Sie untersuchen in ihrer Arbeit die systematische Kombination von verteilungsähnlicher Darstellung der Wortbedeutung mittels syntaktischer Struktur. Sie schlagen vor, die Bedeutung eines komplexen Ausdrucks, der aus zwei syntaktisch verwandten Wörter *w* und *w'* besteht durch einen Vektor darzustellen, indem die einzelnen Wortvektoren *w* und *w'* durch komponentenweise Multiplikation zusammengeführt werden. Die Ergebnisse in ihrer Arbeit zeigen, dass die multiplikativen Modelle genauere Repräsentation komplexer Ausdrücke liefern als additive Alternativen (Mitchell & Lapata 2008). In einem ähnlichen Ansatz präsentieren (Erk & Padó 2008) eine Methode der Berechnung einer Vektorraumdarstellung für die variierende Bedeutung eines Wor-

tes in unterschiedlichem Kontext. Sie schlagen eine strukturierte Vektordarstellung vor, bei der jedes Wort durch einen Standard *Kookkurrenzvektor*, sowie Vektordarstellungen für die selektiven Eigenschaften für Subjekt, Objekt und andere syntaktische Relationen charakterisiert wird. Die Kontextualisierung wird dabei durch die Kombination vom Basisvektor des Zielwortes mit den selektiven Eigenschaften des Subjekts und Objekts modelliert (Erk & Padó 2008). Diese beiden Ansätze lösen eine allgemeine Aufgabe der Repräsentation von Wort- und Phrasenbedeutung und können als das Standard WSD oder LexSub evaluiert werden. In der Arbeit von (Thater et al. 2010) schlagen sie einen ähnlichen Ansatz vor, bei der jedoch die Wortbedeutung durch eine systematische Kombination von erster und zweiter Ordnung Kontextvektoren modelliert wird. Die erste-Ordnung-Vektoren werden durch die jeweiligen Kontexte der Instanzen des Zielwortes aus den Trainingsdaten dargestellt. Die Dimension dieser Vektoren ist von der Anzahl der Kontexte abhängig. Die-zweite-Ordnung-Vektoren werden dargestellt, indem zuerst für jeden der Kontextwörter der Instanz ein erste-Ordnung-Vektor erstellt wird. Anschließend wird der zweite-Ordnung-Vektor dargestellt.
Die Grundlage beide Arten der Vektordarstellung zu konstruieren kann mithilfe von *Co-occurrence-Graphen* durchgeführt werden, welches in Abbildung 7 repräsentiert wird.

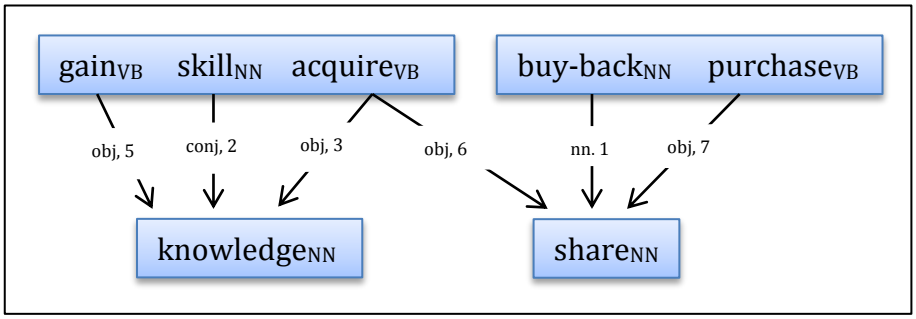

Abbildung 7: "Co-occurrence-Graph"

Im *Co-occurrence-Graphen* werden die Wörter mit Knoten repräsentiert und die möglichen Abhängigkeitsbeziehungen zwischen diesen werden durch Kanten dargestellt. Die Kanten werden zusätzlich mit der Gewichtung und der Beziehung, der zwischen diesen beiden Wörtern besteht beschriftet. In diesem Fall wird die Gewichtung mit der entsprechenden Häufigkeit angegeben. Die Kontextualisierung wird hierbei wie in der Arbeit von (Erk & Padó 2008) realisiert.
Zur besseren Verständlichkeit betrachtet man folgendes Beispiel für den Vektor des Nomen *knowledge*.

$$\langle 5_{(\mathrm{OBJ}^{-1},\mathrm{gain})}, 2_{(\mathrm{CONJ}^{-1},\mathrm{skill})}, 3_{\mathrm{OBJ}^{-1},\mathrm{acquire})}, \ldots \rangle$$

Angenommen man möchte für den Ausdruck *acquire knowlegde* den Bedeutungsvektor berechnen. Da Verben oftmals unterschiedliche syntaktische Nachbarn als Nomen aufweisen, ist es nicht einfach die erste-Ordnung Vektoren zu vergleichen. Zur Lösung dieses Problems haben (Thater et al. 2010) zusätzlich eine andere Art zur Erfassung von Vektorinformationen über alle Wörter eingeführt. Dieses erreichen sie in zwei Schritten.

In diesem Fall, wird ein solcher Pfad durch zwei Abhängigkeitsbeziehungen und aus zwei Wörtern charakterisiert. Sie verallgemeinern dieses, um unnötige Vektoren zu vermeiden und stellen den zweite-Ordnung-Vektor mit dem *mittleren Wort w'* dar. Der zweite-Ordnung-Vektor wird somit anhand der entsprechenden Dimensionen des Tripels (*r, r', w''*) mit den zwei Abhängigkeitsbeziehungen und einem Wort am Ende des Pfades dargestellt. Den zweite-Ordnung-Vektor für *acquire* würde man in diesem Fall so darstellen:

$$\langle 15_{(\mathrm{OBJ},\mathrm{OBJ}^{-1},\mathrm{gain})}, 6_{(\mathrm{OBJ},\mathrm{CONJ}^{-1},\mathrm{skill})}, 6_{(\mathrm{OBJ},\mathrm{OBJ}^{-1},\mathrm{buy-back})}, 42_{(\mathrm{OBJ},\mathrm{OBJ}^{-1},\mathrm{purchase})}, \dots \rangle$$

In diesem einfachen Beispiel sind die Werte, die Produkte der Kantengewichte von jedem Pfad. Nun kann man mit diesen beiden erste- und zweite-Ordnung-Vektoren die Interaktion von semantischen Informationen in komplexen Ausdrücken modellieren. Da die Wörter *acquire* und *knowledge* in einem bestimmten Verhältnis stehen, kann man den zweite-Ordnung-Vektor von *acquire* und den erste-Ordnung-Vektor von *knowledge* kontextualisieren. Diese Operation geschieht mithilfe von punktweiser Multiplikation. In diesem Beispiel erhält man somit einen neuen zweite-Ordnung-Vektor für *acquire* im Kontext mit *knowledge:*

$$\langle 75_{(\mathrm{OBJ},\mathrm{OBJ}^{-1},\mathrm{gain})}, 12_{(\mathrm{OBJ},\mathrm{CONJ}^{-1},\mathrm{skill})}, 0_{(\mathrm{OBJ},\mathrm{OBJ}^{-1},\mathrm{buy-back})}, 0_{(\mathrm{OBJ},\mathrm{OBJ}^{-1},\mathrm{purchase})}, \dots \rangle$$

Im Gegensatz zu den beiden obigen Ansätzen erzielt dieser bessere Ergebnisse, jedoch erfordert er mehr Aufwand aufgrund höherer Komplexität von zweiter Ordnung Kookkurrenzvektoren (Thater et al. 2010). Außerdem können unüberwachte Methoden in der Umsetzung der lexikalischen Substitution für das Ranking verwendet werden.

2.4 Knowledge-Based Disambiguation (wissensbasierte Systeme)

Im Gegensatz zu den beiden vorherigen Modellierungsmethoden repräsentieren die wissensbasierten Methoden eine eigene Kategorie in WSD. Zusammen mit den korpusbasierten Methoden (überwachte und unüberwachte Methode) stellen sie eines der Hauptkategorien von Algorithmen dar, die für *„automatic sense tagging"* entwickelt wurden (Agirre et al. 2006). Alle wissensbasierten Ansätze setzen auf lexikalische Ressourcen, wie Wörterbüchern oder Thesauri und verwenden folglich keine Korpusdaten (Manning & Schütze 1999). Diese Methoden lernen anhand von strukturierten Daten[16], während die korpusbasierten aus Beispielinstanzen lernen (McInnes 2009). Die Leistung solcher wissensbasierten Verfahren wird in der Regel durch die korpusbasierte Alternative übertroffen. Aber sie haben dafür den Vorteil, dass sie eine größere Abdeckung aufweisen. Im Gegensatz zu korpusbasierten Techniken sind die wissensbasierten

[16] strukturierte Daten – vereinfacht kann man es sich wie eine Tabelle vorstellen. Es gibt Spaltenbezeichnungen (Datentypen) und die jeweils zugeordneten Eigenschaften.

Methoden für alle Wörter auf unbeschränkten Texten[17] anwendbar, für welche bereits annotierte Korpora existieren (Agirre et al. 2006). Das bedeutet, dass die *alle-Wörter-Disambiguierung* gegenüber *lexikalischer Disambiguierung* den Vorteil hat, dass nicht für jedes zu disambiguierende Wort die Trainingsdaten erforderlich sind. Lexikalische-Disambiguierungsmethoden können nur Wörter disambiguieren, für welche eine große Menge von Trainingsdaten existiert. Ein weiterer Vorteil der Alle-Wörter-Disambiguierungsmethode ist, dass sie skalierbar ist und daher in praktischen Fällen anwendbar, in welchen das ambige Wort im Voraus nicht bekannt ist, sowie die Trainingsdaten schwer zu erhalten sind. Der Nachteil dieser Methode ist die Sprach- und Domainabhängigkeit. Das bedeutet, dass die Wissensquellen in der entsprechenden Sprache und Domäne erforderlich sind. Außerdem ist die Disambiguierungsgenauigkeit gegenüber den überwachten Methoden nicht so hoch (Navigli 2009), (McInnes 2009).
Es existieren vier Haupttypen der wissensbasierenden Methoden(Agirre et al. 2006):

- **Lesk Algorithmus (1986)** - benutzt unmittelbar die Bedeutung und Definition der Wörter aus dem Wörterbuch. Die wahrscheinlichste Bedeutung für ein Wort in einem gegebenen Kontext wird anhand der kontextuellen Überlappung zwischen den Definitionen des jeweiligen Wortes aus dem Wörterbuch und der Kontextwörter, die das Zielwort umgeben, identifiziert.

Lesk-Algorithmus:

1. comment: Given: context c
2. **for all** senses sk of w **do**
3. score(sk) = overlap (Dk, Uv, inc Evj)
4. **end**
5. choose s' s.t.s' = argmax, score(sk)

- **Semantische Ähnlichkeit** - Das Maß der semantischen Ähnlichkeit wird über die semantischen Netzwerke[18] berechnet. Diese Kategorie beinhaltet Verfahren zur Ermittlung der semantischen Dichte bzw. der Distanz zwischen den Wortbedeutungen. Basierend auf der Größe des Kontextes, sind diese Maße wiederum in zwei Hauptkategorien unterteilt:
  - Methoden, welche sich auf den *lokalen Kontext* anwenden lassen und bei denen das semantische Ähnlichkeitsmaß verwendet wird, um Wörter zu disambiguieren, die durch *syntaktische Relation* oder *ihre Lokalität* verbunden sind.

[17] unbeschränkter Text – Text, der in seiner Länge nicht begrenzt ist
[18] Semantisches Netz – ist ein Modell von Begriffen und ihren Beziehungen zueinander. Es wird im Allgemeinen durch ein Graph repräsentiert. Die Knoten stellen dabei die Begriffe dar und die Beziehung zwischen ihnen wird durch die Kanten realisiert.

- Methoden, welche sich auf den *globalen Kontext* anwenden lassen und bei denen lexikalische Ketten[19] in Abhängigkeit vom semantischen Ähnlichkeitsmaß abgeleitet werden.

Der Einsatz der semantischen Ähnlichkeit wird in der Bedeutungsbeschreibung und –unterscheidung oftmals verwendet. Sie kann in allen drei Modellarten (überwacht, unüberwacht und wissenbasiert) eingesetzt werden. In den letzten Jahren gab es eine einige Forschungsarbeiten, bei dem dieser Ansatz verwendet wurde. In der Arbeit von (Agirre et al. 2009) wird ein wissensbasiertes Modell mit dem Einsatz von der lexikalischen Datenbank WordNet und ein überwachtes Modell verglichen, indem bei beiden Modellen die semantische Ähnlichkeit miteinbezogen wird. In Bezug auf die semantische Ähnlichkeit wird in beiden Modellen auf die Stärken und Schwächen eingegangen und es wird dabei ein mögliche Kombination beider Ansätze vorgestellt. Einen weiteren Ansatz hierzu hat (Ponzetto & Navigli 2010) vorgestellt. In ihrer Arbeit präsentieren sie eine Methode der automatischen Erweiterung von WordNet mit einer großen Menge von semantischen Relationen aus Wikipedia, einer enzyklopädischen Ressource. Weitere Arbeiten, die hierzu relevant sind: (Miller et al. 2012), (Mohler & Mihalcea 2009), (Navigli & Ponzetto 2012), (Navigli & Ponzetto 2010)

- **Selektionspräferenzen** - Selektive Präferenzen beinhalten Informationen über mögliche Relationen zwischen Wortkategorien. Sie werden als Mittel verwendet, um die möglichen Bedeutungen für ein Wort in einem bestimmten Kontext einzuschränken. Diese Informationen können auch in den anderen Modellarten verwendet werden. Man kann auch mögliche Relationen und Abhängigkeiten zwischen Wörtern in einem Satz oder Text extrahieren. Beispielsweise könnte der Ansatz von (Thater et al. 2010), welcher oben beschrieben ist, auch als eine Art der Selektionspräferenzansatz angesehen werden. (Thater 2010) betrachtet dabei die Abhängigkeitsbeziehung zwischen den Wörtern und ihrem dazugehörigen Kontext.
- **Heuristik für WSD** - Heuristische Methoden bestehen aus einfachen Regeln, die zuverlässig einer bestimmten Wortkategorie einen Sinn bzw. eine Bedeutung zuweisen können. Eine solche Heuristik, die oft als Grundlage in der Evaluation vieler WSD-Systeme verwendet wird, ist die „*most-frequent-sense*“ (häufigste Bedeutung) Heuristik. Zwei andere mögliche Heuristiken wären beispielsweise „*one-sense-per-discourse“* und „*one-sense-per-collocation*“[20], die bei den wissensbasierten Methoden ihre Verwendung finden.

Die Abbildung 8 veranschaulicht ein allgemeines Modell der wissensbasierte WSD Methoden. In dieser Methode erhält das „Evaluation Module“ die Testdaten als Eingabe. Möglicherweise wurden die Instanzen in den Tesdaten bereits die entsprechende Be-

[19] Lexikalische Kette - eine lexikalische Kette ist ein „Faden“ der Bedeutung, der sich durch einen ganzen Text zieht. Diese Kette enthält Begriffe, die miteinander in Relation stehen.
[20] Vertiefend (Agirre et al. 2006)

deutug zugewiesen. Diese Annotationen werden entfernt und so dem „Vector Creation Module“ zugesendet.

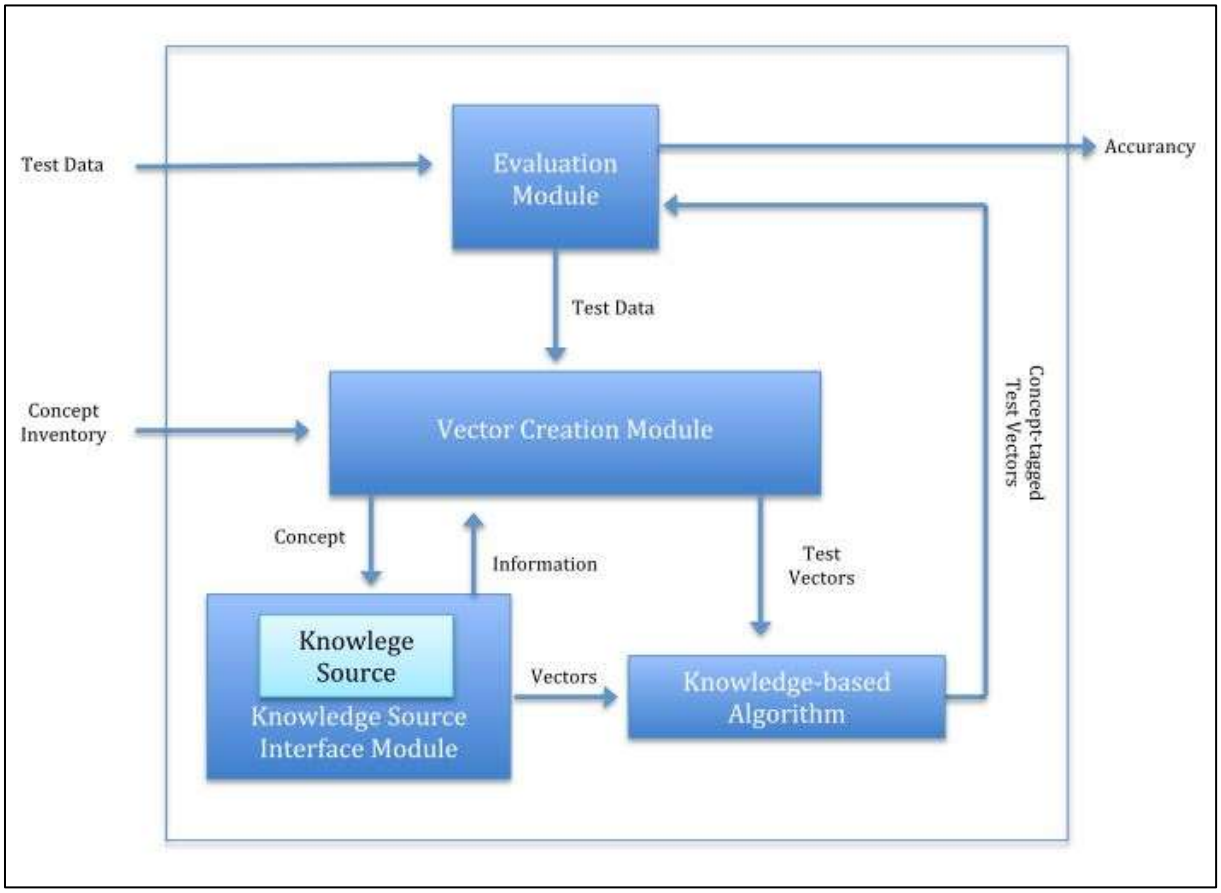

Abbildung 8: Wissensbasierte WSD-Methode

Unter Verwendung von Informationen aus der Wissensquelle (Knowledge Source) wird für jede Instanz ein Testvektor erstellt. Diese Information wird über die Schnittstelle (Knowledge Source Interface Module) erhalten. Die Testvektoren werden anschließend an den wissensbasierten Algorithmus gesendet, der die Informationen aus den Vektoren verwendet, um die entsprechende Bedeutung des Zielwortes zu bestimmen. Es gibt verschiedene Arten von wissensbasierten Methoden. Sie alle beruhen aber auf manuell erstellten Wissensquellen wie Wörterbücher, Thesauri oder lexikalischen Datenbanken. Die nun gekennzeichneten Testvektoren werden zurück zum „Evaluation Module“ gesendet und die Genauigkeit der Methode wird bestimmt (McInnes 2009). Weitere wissensbasierte Arbeiten die für WSD und lexikalische Substitution relevant sind: (Miller et al. 2012), (Mohler & Mihalcea 2009), (Navigli & Ponzetto 2012), (Navigli & Ponzetto 2010), (Sinha & Mihalcea 2014), (Bär et al. 2012), (Hassan et al. 2007).

3 Daten

Im diesem Kapitel werden für diese Bachelorarbeit relevante Datensätze näher erläutert. Insbesondere wird auf ihren Aufbau und Größe eingegangen und inwiefern sie zum Einsatz kamen. Die hierbei verwendeten Datensätze sind:

- CoInCo-Korpus („*Concepts in Context*“)
- Gigaword

3.1 CoInCo – Korpus

Der CoInCo-Korpus ist ein relativ großer englischsprachiger und alle Wörter umfassender Datensatz -“all-words lexical substitution Korpus“-. Dieser wurde auf Grundlage der newswire- und fiction-Teile des frei zugänglichen MASC-Korpus veröffentlicht. Als ein Korpus für lexikalische Substitution umfasst er 35,000 Tokens laufenden Text, bei dem für alle 15,000 Tokens von Inhaltswörtern mit Crowdsourcing[21]-Methoden mindestens 6 Synonyme generiert wurden (Abbildung 9).

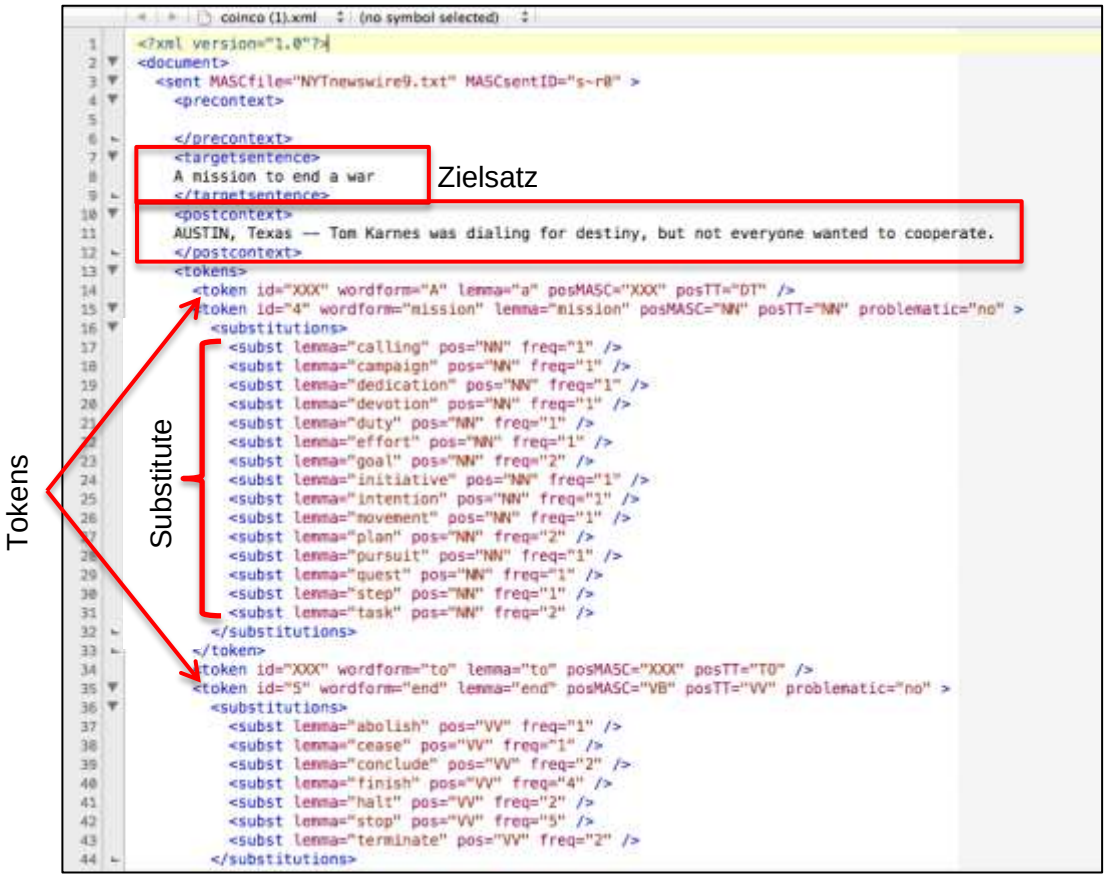

Abbildung 9: Ausschnitt aus dem CoInCo - Datensatz

[21] Crowdsourcing ist ein Prozess, in diesem Fall die Annotation, bei dem eine Gruppe von freiwilligen oder bezahlten Teilnehmern die Auslagerung der Aufgaben durchführt

Dabei wurde allen Teilnehmern der Satz sowie zwei weitere Sätze Diskurskontext (precontext und postcontext) zur Verfügung gestellt. In Abbildung 9 wird in Zeile 8 der Zielsatz (targetsentence) *„A mission to end a war"* und in Zeile 11 der dazugehörigen Diskurskontext (postconetxt) illustriert. Daraufhin wird ab Zeile 13 der Satz in Tokens segmentiert. Jedes Wort im Satz entspricht einem Token, welche mit einer eigenen ID gekennzeichnet wird. Wie man deutlich in Zeile 14 sehen kann, bekommt das erste Wort *„A"* keine Token-ID, weil dafür keine Substitute generiert wurden. In Zeile 15 jedoch hat das Word *"mission"* die Token-ID 4 und es wurden hierfür 15 Substitute generiert, die durch "*substitutions"* gekennzeichnet sind. In den fortlaufenden Zeilen wird dasselbe weitergeführt, indem man für die Begriffe *„end"* und *„war"* Substitute generiert. Des Weiteren wird für jedes Token seine Wortform, das dazugehörige Lemma sowie die Wortart (kurz pos) angegeben. Außerdem wird für jedes potentielle Substitut das Lemma, die Wortart und die Frequenz (freq) angegeben, sowie wie viele Annotatoren dieses Substitut in Bezug auf den Kontext als geeignet sahen.

3.2 Gigaword – Korpus

Der englische Gigaword[22]-Korpus wurde von der Linguistic Data Consortium (LDC) produziert. Dieser Korpus ist ein sehr umfassendes Archiv aus englischsprachigen Textdaten von Nachrichtenagentur, die über mehrere Jahre von der LDC erarbeitet wurden. Sieben verschiedene internationale Quellen der englischsprachigen Nachrichtenagentur sind hierbei vertreten:

- Agence France Press, Englisch Service (afp_eng)
- Associated Press Worldstream, English Service (apw_eng)
- Central News Agency of Taiwan, English Service (cna_eng)
- Los Angeles Times/Washington Post Newswire Service (ltw_eng)
- Washington Post/Bloomberg Newswire Service (wpb_eng)
- The New York Times Newswire Service (nyt_eng)
- The Xinhua News Agency English Service (xie)

Jeder Datei-Name besteht aus einem drei-Buchstaben-Präfix, gefolgt von einem sechsstelligen Datum (Jahr und Monat, in dem die Daten-Inhalte von der jeweiligen Nachrichtenagentur zugestellt wurden). Alle Textdaten werden in SGML Form präsentiert, mit einer sehr einfachen minimalen Markup-Struktur. Der gesamte Text besteht aus einem druckbaren ASCII und Leerzeichen. Die Markup-Struktur, die für alle Daten gleich ist, kann wie folgt zusammengefasst werden (siehe Abbildung 10):

- Die Titelzeile (Überschrift) ist optional – nicht alle DOCs haben einen

[22] English Gigaword – https://catalog.ldc.upenn.edu/LDC2003T05 (27.11.2014 17:00)

- Die Datumszeile ist optional – nicht alle DOCs haben einen
- Paragraphenmarkierungen werden nur verwendet, wenn der DOC-Type gleich “story“ ist.

Agentur-Quelle, Datumsangabe und Typangabe

Überschrift

Paragraph

```
<DOC id="LTW_ENG_20081201.0001" type="story" >
<HEADLINE>
Road Map in Iraq: When Mr. Obama Takes Office, a Sovereign Iraqi Government and a U.S.
 Withdrawal Timetable Will Be in Place
</HEADLINE>
<TEXT>
<P>
The following editorial appeared in Sunday's Washington Post:
</P>
<P>
Barack Obama recently reiterated his campaign promise to order up a
plan for the withdrawal of U.S. forces from Iraq. But the Iraqi parliament
has beaten him to it. Its ratification Thursday of a new bilateral military
agreement with the United States not only establishes a timetable for the
redeployment of American troops but delimits the missions they can undertake
between now and the end of 2011. Mr. Obama has always said that his strategy
was aimed at forcing Iraqi leaders to take responsibility for their country
and its security. In adopting and ratifying the accord, the government and
parliament have taken a major step toward that goal.
</P>
<P>
By now Mr. Obama and most other opponents of the military surge launched
by President Bush nearly two years ago have acknowledged its success in
greatly reducing violence around Iraq. The completion of the Status of
Forces Agreement and the accompanying strategic partnership accord with the
United States shows how far the political system also has come. Two years
ago, Mr. Bush's national security adviser wrote a memo questioning whether
Prime Minister Nouri al-Maliki was able or even willing to assert his
authority. But Mr. Maliki has been both skillful and forceful in extracting
concessions from the Bush administration -- such as the 2011 withdrawal date
-- and winning agreement from Sunni as well as Shiite and Kurdish
legislators.
</P>
```

Abbildung 10: Ausschnitt aus Gigaword-Datensatz LTW_ENG[23]

Statistiken bezüglich der Datenmengen für jede Quelle werden nachfolgend zusammengefasst. Man muss beachten, dass die Spalte *„Totl-MB“* die Menge an Daten aufzeigt, die man erhält, wenn die Daten entpackt werden. Die beträgt insgesamt ungefähr 26 Gigabyte. Die Spalte *„Gzip-MB“* zeigt die Größe der Daten in komprimierter Form. Die Zahlen in der *„K-Wrds“* Spalte stellen die Anzahl der Leerzeichen-Tokens, nachdem alle SGML-Tags eliminiert wurden. Die Tabelle 3 zeigt die entsprechende Auflistung der sieben Ressourcen.

[23] https://catalog.ldc.upenn.edu/desc/addenda/LDC2009T13.html

Source	#Files	Gzip-MB	Totl-MB	K-Wrds	#DOCs
afp_eng	146	1732	4937	738322	24779624
apw_eng	193	2700	7889	1186955	3107777
cna_eng	144	86	261	38491	145317
ltw_eng	127	651	1694	268088	411032
nyt_eng	197	3280	8938	1422670	1962178
wpb_eng	12	42	111	17462	26143
xin_eng	191	834	2518	360714	1744025
TOTAL	**1010**	**9325**	**26348**	**4032686**	**9876086**

Tabelle 3: Gigaword: Größe der sieben Daten[24]

3.2.1 Annotated Gigaword – Korpus

Napoles et al. erstellten mehrere Annotations-Ebenen auf dem englischen Gigaword v.5 Korpus, um es sinnvoll als standardisierten Korpus für Wissensextraktion benutzen zu können. Die meisten bestehenden umfangreichen Arbeiten basieren auf uneinheitliche Korpora, die häufig von Wissenschaftlern unabhängig neu annotiert werden mussten. Ihr Ziel dabei war es, einer größeren Gruppe von Wissenschaftlern eine reichhaltige Ressource für Wissenserwerbszwecke zur Verfügung zu stellen, die sonst vielleicht nicht die Möglichkeit hätten, eine solche Ressource auf ihre eigene Weise zu erstellen.
Gigaword ist derzeit der größte verfügbare Korpus von englischen Nachrichtendokumenten. Die neueste Ergänzung, Gigaword v.5, (Parker et al. 2011) beinhaltet fast 10 Millionen Dokumente aus sieben Nachrichtenagenturen, mit insgesamt mehr als 4 Milliarden Wörtern. Diese Ressource wird (Napoles et al. 2012):

- eine konsistente Datenmenge bereitstellen, mit der die Forscher Ergebnisse vergleichen können
- die Verdoppelung der Annotation von verschiedenen Forschergruppen vermeiden

Im Vergleich zu vorherigen annotierten Korpora ist der „Annotated Gigaword-Korpus“ eine größere Ressource, auf formal bearbeiteten Material, der zusätzlich Ebenen der Annotation hat und den aktuellen Stand der Technik in der Textverarbeitung wiederspiegelt. Insbesondere weist die Sammlung folgende Eigenschaften für den englischen Gigaword v.5 auf:

- Tokenisierte und segmentierte Sätze,
- „Treebank-style[25]“
- Syntaktische Abhängigkeitsbäume,
- „named entities“ (Eigennamen, Personen, Organisationen, Orte und Zeitangaben)
- Ko-Referenz Ketten in Dokumenten

[24] https://catalog.ldc.upenn.edu/LDC2011T07
[25] treebank = geparstes Korpus, bei dem jeder Satz eines Textkorpus mit syntaktischer Struktur annotiert wird. Die syntaktische Struktur wird als Baumstruktur dargestellt. (http://en.wikipedia.org/wiki/Treebank)

Abbildung 11 zeigt ein Dokument mit seiner zugehörigen ID, sowie einer Typzuordnung “story“ und zwei geparste Paragraphen, die mit “P“ gekennzeichnet sind.

Agentur-Quelle, Datumsangabe und Typangabe

geparster Paragraph

```
<DOC id="AFP_ENG_19940512.0003" type="story" >
<HEADLINE>
( (S (NP (NNS Tributes)) (VP (VBP pour) (ADVP (RB in)) (PP (IN for) (NP (JJ late) (NNP British)
 (NNP Labor) (NNP Party) (NN leader))))))
</HEADLINE>
<DATELINE>
( (NP (NP (NP (NNP UNDATED)) (, ,) (NP (NNP May) (CD 12))) (PRN (-LRB- -LRB-) (NP (NNP AFP))
 (-RRB- -RRB-))))
</DATELINE>
<TEXT>
<P>
( (S (NP (NNS Tributes)) (VP (VBD poured) (ADVP (IN in) (PP (IN from) (PP (IN around) (NP (DT the)
 (NN world))))) (NP (NNP Thursday)) (PP (TO to) (NP (NP (DT the) (JJ late) (NNP Labor) (NNP Party)
 (NN leader) (NNP John) (NNP Smith)) (, ,) (SBAR (WHNP (WP who)) (S (VP (VBD died)
 (ADVP (RBR earlier) (PP (IN from) (NP (NP (DT a) (JJ massive) (NN heart) (NN attack))
 (VP (VBN aged) (S (NP (CD 55))))))))))))) (. .)))
</P>
<P>
( (S (PP (IN In) (NP (NNP Washington))) (, ,) (NP (DT the) (NNP US) (NNP State) (NNP Department))
 (VP (VBD issued) (NP (NP (DT a) (NN statement)) (VP (VBG regretting) (NP (`` ``) (NP (DT the)
 (JJ untimely) (NN death)) ('' '') (PP (IN of) (NP (DT the) (JJ rapier-tongued) (JJ Scottish)
 (NN barrister) (CC and) (NN parliamentarian)))))))) (. .)))
</P>
```

Abbildung 11: Ausschnitt aus dem annotierten Gigaword-Datensatz AFP_ENG

Abbildung 12 zeigt einen Teil der Tokenisierung des ersten Paragraphen. Nach Angabe der Satz-ID (sentence id) beginnt die Tokenisierung . Die Tokens werden nacheinander mit der entsprechenden IDs aufgezählt. Zusätzlich wird für jedes Token seine Länge und seine Wortart (POS) angegeben, sowie ob es sich um ein “named entity“ (NER) handelt.

Länge

```
<sentences>
  <sentence id="1">
    <tokens>
      <token id="1">
        <word>Tributes</word>
        <lemma>tribute</lemma>
        <CharacterOffsetBegin>0</CharacterOffsetBegin>
        <CharacterOffsetEnd>8</CharacterOffsetEnd>
        <POS>NNS</POS>
        <NER>O</NER>
      </token>
      <token id="2">
        <word>poured</word>
        <lemma>pour</lemma>
        <CharacterOffsetBegin>9</CharacterOffsetBegin>
        <CharacterOffsetEnd>15</CharacterOffsetEnd>
        <POS>VBD</POS>
        <NER>O</NER>
      </token>
        ...
```

Abbildung 12: Ausschnitt einer Tokenisierung eines Paragraphe

```
    <parse>(ROOT (S (NP (NNS Tributes)) (VP (VBD poured) (ADVP (IN in) (PP (IN from)
(PP (IN around) (NP (DT the) (NN world))))) (NP (NNP Thursday)) (PP (TO to)
(NP (NP (DT the) (JJ late) (NNP Labor) (NNP Party) (NN leader) (NNP John) (NNP Smith))
 (, ,) (SBAR (WHNP (WP who)) (S (VP (VBD died) (ADVP (RBR earlier) (PP (IN from)
(NP (NP (DT a) (JJ massive) (NN heart) (NN attack)) (VP (VBN aged) (S (NP (CD 55))))))))))))
 (. .)))</parse>
      <basic-dependencies>
        <dep type="nsubj">
          <governor>2</governor>
          <dependent>1</dependent>
        </dep>
        <dep type="root">
          <governor>0</governor>
          <dependent>2</dependent>
        </dep>
        <dep type="advmod">
          <governor>2</governor>
          <dependent>3</dependent>
        </dep>
        <dep type="prep">
          <governor>3</governor>
          <dependent>4</dependent>
        </dep>
          ...
```

Abbildung 13: Abhängigkeiten der Wörter im Datensatz

Die Abhängigkeiten zwischen den einzelnen Begriffen werden in Abbildung 13 veranschaulicht. Die Abkürzungen, die hinter *type* stehen, geben die Art der Abhängigkeit an und die zwei Zahlen bei *governor* und *dependet* zwischen welchen zwei Begriffen diese Abhängigkeit besteht. Aus der Abbildung 13 kann man entnehmen, dass zwischen dem Begriff 2 (*poured*) und dem Begriff 1 (*Tributes*) eine *Nominale-Subjekt* Beziehung (engl. nominal subject, kurz nsubj) besteht.

4 Realisierung des Ansatzes

Für die Realisierung dieses Ansatzes in dieser Bachelorarbeit, wurden Wort– und Kookkurrenzlisten verwendet, die aus den CoInCo-Korpus (Kapitel 3.1) und dem englischen Gigaword-Korpus (Kapitel 3.2.1) entnommen worden sind. Das realisierte Modell basiert auf der Arbeit von (Thater et al. 2011).

4.1 Herangehensweise

In Kapitel 2.3 wurde bereits detailliert anhand eines Beispiels ein unüberwachtes Modell in Bezug auf Bedeutungsbeschreibung beschrieben und dessen Unterschied zur LexSub diskutiert. In diesem Kapitel wird ein weiteres unüberwachtes Modell von (Thater et al. 2011) präsentiert und anschließend als LexSub-Aufgabe evaluiert. Wie bereits in Kapitel 2.1 erläutert, befasst sich die LexSub-Aufgabe mit der Generierung und anschließendem Ranking der potentiellen Substitute. Dieses Modell von (Thater et al. 2011) wird mit dem Verfahren der semantischen Ähnlichkeit umgesetzt und dabei auf das Ranking der möglichen Substitute für das zu ersetzende Zielwort angewandt. In ihrem Modell beschreiben sie die Darstellung einer kontextuellen Wortbedeutung mit Vektoren. Diese Vektoren werden bezüglich der Wörter im syntaktischen Kontext modifiziert. Die Kontextualisierung eines Vektors wird dabei durch Neugewichtung seiner Komponenten realisiert, basierend auf der verteilten Information der Wörter im Kontext. Der kontextuelle Einfluss auf die Bedeutung eines Zielwortes wird in diesem Ansatz durch Vektor-Komposition modelliert. Das heißt, dass die Bedeutung des Wortes *w* im Kontext *c* durch einen Vektor dargestellt wird und dann diese beiden Vektoren mit komponentenweiser Multiplikation oder Addition kombiniert werden. Dieses Modell erlaubt die Berechnung der Vektordarstellung für einzelne Wörter, bei dem die genaue Bedeutung des Wortes durch den Satzkontext charakterisiert wird.
Als kleine Einführung betrachte man folgendes Beispiel für das Verb *charge* (Abbildung 14). Die Bedeutung des Verbes im Ausdruck *charge a fee* hat eine finanzielle Bedeutung. Im Gegensatz hierzu hat das Verb im Ausdruck *charge a battery* eine ganz andere Bedeutung. Mit dieser Methode kann man nun aus dem Basisvektor des Zielwortes einen kontextualisierten Vektor ableiten, indem man seine Komponenten in Bezug zum Kontext neu gewichtet. Die Dimensionen des Basisvektors und auch des kontextulaisierten Vektors zeigen nun die gemeinsam auftretenden Wörter in bestimmten syntaktischen Beziehungen. Dies wird in Abbildung 14 bis Abbildung 16 repräsentiert.

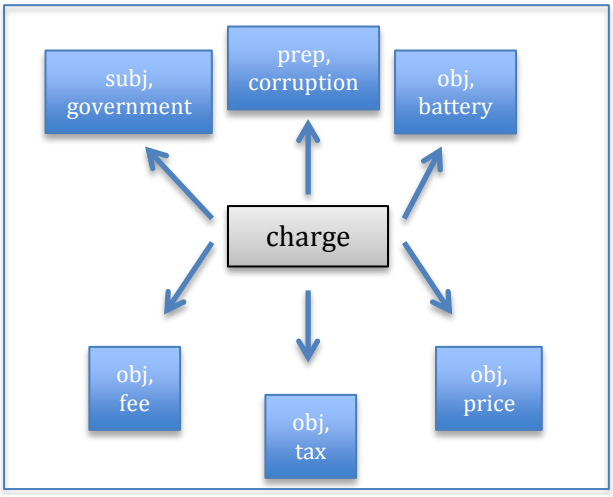

Abbildung 14: Basisvektor von "charge"

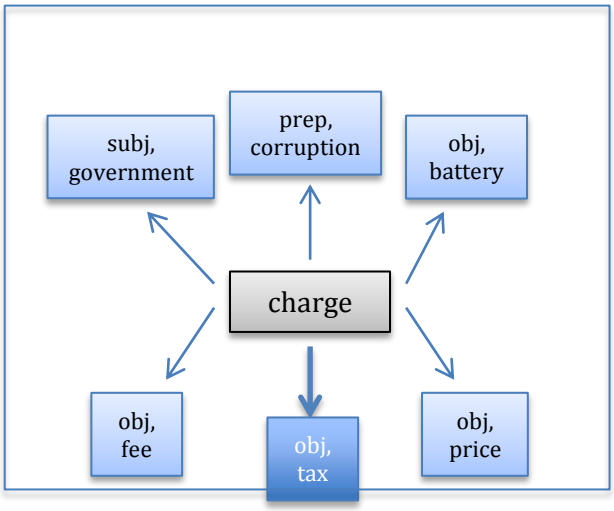

Abbildung 15: Strikte Kontextualisierung

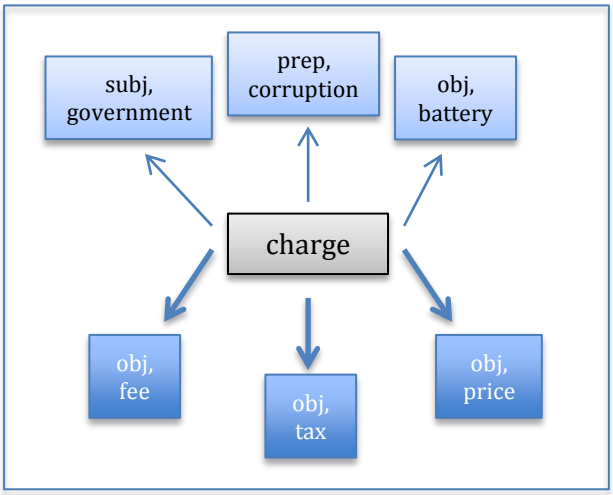

Abbildung 16: Kontextualisierung nach semantischer Ähnlichkeit

Den einfachsten Weg, den Vektor eines Wortes zu seinem Kontext anzupassen, erreicht man dadurch, dass man nur die Dimensionen zu seinem syntaktischen Nachbarn beibehält. Somit erhält man einen dünnbesetzten Vektor mit vielen Nullwerten. Beispielweise wäre die Kontextualisierung des Vektors *charge* im Ausdruck *charge a tax* für alle Komponenten null, bei denen $r \neq obj$ oder $w \neq tax$. Dieser Vektor hat in diesem Fall dann nur einen Nicht-Null-Eintrag und zwar für das Kontextwort *tax* selbst. Dies zeigt Abbildung 15. Durch den Einsatz von Informationen über die semantische Ähnlichkeit der Kontextwörter erzielt man wesentlich bessere Ergebnisse. Das bedeutet, dass man nicht nur die Dimensionen der Kontextwörter selbst betrachtet bzw. berücksichtigt, sondern auch alle Dimensionen der Wörter, die ähnlich zu den Kontextwörter sind. Der Vektor *charge* im Ausdruck *charge a tax* beinhaltet somit Nicht-Null-Einträge in den Dimensionen, bei dem alle Wörter ähnlich zu *tax* sind. In diesem Zusammenhang wären es die Kontextwörter *fee* und *price* (Abbildung 16*).*

4.2 Algorithmus

Dieses Kapitel behandelt die formale Beschreibung des Algorithmus von Thater et al. 2011.
Es wird angenommen das *W* eine Menge von Wörtern ist und *R* eine Menge von syntaktischen Relationen. Die Menge *R* beinhaltet zusätzlich Abhängigkeitsbeziehungen wie SUBJ (Subjekt) oder OBJ (Objekt). Die Bedeutung eines Wortes $w \in W$ wird durch einen Vektor im Vektorraum V repräsentiert. Der Vektorraum *V* wird durch die Menge von Basisvektoren $\{e(r, w') \mid r \in R, w' \in W\}$ aufgespannt (1). Ein solcher Vektor bezeichnet die Zusammenhangsstärke zwischen *w* und jedem mit *w* in Relation stehendem Kontextwort *w'*. Insbesondere assoziiert man ein Wort $w \in W$ mit einem Vektor $v(w) \in V$ durch

$$\boldsymbol{v}(w) \;\coloneqq \sum_{r \in R, w' \in W} f(w, r, w') \cdot \boldsymbol{e}_{(r,w')} \;, \tag{1}$$

wobei f eine Funktion darstellt, die dem Abhängigkeitstripel *(w, r, w´)* ein Gewicht zuweist. Im einfachsten Fall könnte das Gewicht, die Häufigkeit der auftretenden Relation *r(w, w')* in einem Korpus von Abhängigkeitsbäumen darstellen. (Thater et al. 2011) verwendet hingegen die *„pointwise mutual information"* (Church & Hanks 1990):

$$PMI(w, r, w') = log \frac{p(w, w' \mid r)}{p(w, \bullet \mid r)\; p(\bullet, w' \mid r)} \tag{2}$$

Die Punkte (•) stehen hierbei für die Marginalisierung über den relevanten Variablen.
Das Auftreten eines Wortes w im Kontext eines anderen Wortes w_c wird durch die syntaktische Relation r_c verbunden. Dabei erhält man eine kontextualisierte Version *v(w)* durch Neugewichtung der Vektorkomponenten.

$$v_{r_c, w_c}(w) \;\coloneqq\; \sum_{r \in R,\; w' \in W} \alpha_{r_c, w_c, r, w'} \cdot f(w, r, w') \cdot \boldsymbol{e}_{(r,w')} \tag{3}$$

Die Gewichte $\alpha_{r_c,w_c,r,w'}$ messen den Grad, bei dem die Vektordimension *(r, w')* mit dem zu berücksichtigenden Kontext *(r_c, w_c)* kompatibel sind. Man betrachtet drei alternative Definitionen dieser Gewichte, entsprechend der Beispiele in Abbildung 14 bis Abbildung 16.

- **No contextualization:** $\alpha_{r_c,w_c,r,w'} := 1$

  In diesem Fall stimmt die Definition von $v_{r_c,w_c}(w)$ mit der Definition von $\boldsymbol{v}(w)$ überein.

- **Strict contextualization:**

  $\alpha_{r_c,w_c,r,w'} := \delta_{r_c,r}\delta_{w_c,w}$

  $$= \begin{cases} 1 & \text{if } r_c = r \text{ and } w_c = w' \\ 0 & else \end{cases}$$

  In diesem Fall behalten wir nur die Dimensionen (r_c, w_c), die durch den Kontext zugelassen werden und setzen alle andere auf 0.

- **Similarity-based contextualization:**

  $\alpha_{r_c,w_c,r,w'} := \delta_{r_c} \cdot sim(w_c, w')$

  $$= \begin{cases} sim(w_c, w') & \text{if } r_c = r \\ 0 & else \end{cases}$$

  In diesem Fall wird dies verallgemeinert und es werden alle Wörter w‘ zugelassen, die semantisch ähnlich zum Kontextwort w_c sind. Die Ähnlichkeit zwischen w_c und *w‘* wird mit dem Cosinus Winkel zwischen seinem Basisvektor *v(w_c)* und *v(w‘)* berechnet. Dabei berücksichtigen sie mehrere Kontextwörter für ein gegebenes Wort *w.* Aus den gegebenen Kontextwörtern $w_1, \ldots, w_n$ und ihren zugehörigen Relationen $r_1, \ldots, r_n$ leiten sie einen kontextualisierten Vektor für das Wort *w* ab. Dieser entsteht durch die Vektoraddition $v_{r_i w_i}$ *für alle* $(1 \leq i \leq n)$:

  $$v_{r_1 w_1,\ldots,r_n w_n}(w) := \sum_{i=1}^{n} v_{r_i,w_i}(w) \qquad (4)$$

  Der daraus resultierende Vektor $v_{r_1 w_1,\ldots,r_n w_n}(w)$ ist somit die vollständig kontextualisierte Vektordarstellung des Wortes *w,* welches Informationen über alle Kontextwörter beinhaltet.

4.3 Evaluierung

Ansätze zur lexikalischen Substitution wurden in der Regel mit dem SEMEVAL 2007 Datensatz (McCarthy & Navigli 2009) evaluiert. Ein Vergleich der Ergebnisse auf einem größeren Datensatz von Arbeiten, die sich mit *lexikalischer Substitution* befassen, bieten (Kremer et al. 2014) in ihrer Arbeit. Hierfür verwenden sie den CoInCo Datensatz, bei dem alle Wörter der vier Wortarten (Nomen, Verben, Adjektive, Adverben) substituiert wurden. Die Tabelle 4 zeigt die Ergebnisse, die (Kremer et al. 2014) bei der Anwendung der drei Paraphrasen Ranking Modellen (unüberwachte Modelle) von (Erk & Padó 2008)(EP08), (Thater et al. 2010) (TFP10) und (Thater et al. 2011) (TFP11) auf ihren eigenen Korpus MASC LexSub (CoInCo) und dem weiteren Datensatz SEMEVAL 2007 (McCarthy & Navigli 2009) erhielten. Das Modell von (Erk&Pado) und (Thater 2010) wurde in Kapitel 2.3 bereits kurz eingeführt. Die Grundidee dieser drei Modelle ist, dass die Bedeutung eines Zielwortes in einem spezifischen Kontext durch die Modifizierung seines Basisvektors repräsentiert werden kann. Dies geschieht, indem die Wörter, die in einem direkten syntaktischen Kontext zum Zielwort stehen, dazu verwendet werden, die Bedeutung des Zielwortes näher zu spezifizieren. Sie verwendeten den englischen Gigaword Korpus, um Kookkurrenzen zu sammeln. Es wurden dabei drei Varianten der einzelnen Modelle betrachtet:

- **syntactically structured:** hierbei werden Kookkurrenzen zwischen dem Abhängigkeitstripel *(w, r, w')* und zusätzlich die syntaktische Information gespeichert.

- **syntactically filtered:** dabei werden ebenfalls anhand des Abhängigkeitstripels *(w, r,w')* Kookkurrenzen gesammelt, aber die syntaktische Information wird nicht als Vektor dargestellt.

- **bag-of-words:** verwenden ∓ 5 Wörter

Corpus		Syntactically structured			Syntactically filtered		Bag of words		random
		TFP11	TFP10	EP08	TFP11/EP08	TFP10	TFP11/EP08	TFP10	
MASC LexSub	context	**47.8**	46.0	47.4	47.4	41.9	46.2	40.8	33.0
	baseline	46.2	44.6	46.2	45.8	38.8	44.7	37.5	
SemEval 2007	context	**52.5**	48.6	49.4	50.1	44.7	48.0	42.6	30.0
	baseline	43.7	42.7	43.7	44.4	38.0	42.7	35.8	

Tabelle 4: Vergleich drei Paraphrasen-Ranking-Modellen auf verschiedenen Korpora

Die Tabelle 4 zeigt die Ergebnisse, die sie nach ihrem Ranking mit der Auswertung des GAP (*Generalised Average Precision*)-Wertes (siehe Formel (8)) erhielten (Kishida 2005), (Thater et al. 2010). Das Ranking haben sie mit dem Standardverfahren durchgeführt, in dem nur alle genannten Substitute für alle Instanzen des Testwortes betrachtet wurden. Die berichteten Zahlen aus der Tabelle 4 werden mit den Ergebnissen der Evaluierungsphase dieser Arbeit verglichen. Es wird betrachtet, wie sich das Modell verhält, wenn es

die Menge aller genannten Substitute für alle Zielwörter als Substitutionskandidaten verwendet und rankt.
In der Evaluierungsphase dieser Arbeit wurde der große, englischsprachige, alle Wörter umfassende CoInCo-Datensatz (Kapitel 2.1 und 3.1) und der annotierte englische Gigaword Korpus (Kapitel 3.2.1) verwendet. Die Hauptaufgabe bestand im Ranking *aller* möglichen Substitute. Das Ziel dabei war es, für jedes Zielwort die richtigen Substitute höher zu ranken als alle anderen und zu beobachten, wie viel schlechter das Modell dabei wird. Zu Beginn wurden alle Zielwörter und alle ihre dazugehörigen Substitute aus dem CoInCo-Datensatz gefiltert. Dabei wurden lediglich alle unterschiedlichen Wörter betrachtet. Diese waren insgesamt 20.749 unterschiedliche Instanzen. Anhand dieser Instanzen wurde aus dem annotierten Gigaword Korpus für jede Instanz sein Vorkommen mit einem Kontextwort und seiner entsprechenden Relation gesammelt. Zusätzlich wurden höhere Frequenz und PMI-Werte für die Berechnung der Ähnlichkeit verwendet, um den Rechenaufwand für die Kontextualisierung der Vektoren zu reduzieren. Das Abhängigkeitstripel *(w, r, w')* musste mindestens fünfmal vorkommen und der PMI-Wert musste mindestens 2 betragen. Durch die Verwendung dieser beiden Bedingungen verringerte sich die Anzahl der Instanzen auf 7894. Anschließend wurde für jede der 7894 Instanzen die semantische Ähnlichkeit mit jedem anderen berechnet. Zum Schluss wurden anhand der semantischen Ähnlichkeitswerte zwei Experimente durchgeführt, um dabei für die 5450 Testwörter, die Vorhersage des Modells anhand des GAP-Werts zu berechnen.
Im ersten Experiment wurde das Standardverfahren, wie bei (Kremer et al. 2014), durchgeführt. Dabei wurden nur die annotierten Substitute für alle Instanzen des Zielwortes betrachtet und anhand dieser das Ranking durchgeführt. Im zweiten Experiment wurde diese Herangehensweise erweitert, indem für das Ranking *alle* annotierten Substitute aller Zielwörter verwendet wurden. Die daraus resultierende Ranking-Liste wurde anhand der GAP-Formel ausgewertet. Die GAP-Formel beurteilt, wie präzise die vorhergesagten Ergebnisse unter Berücksichtigung der jeweilig annotierten Substitute aus dem Gold Standard sind. Dabei wird jedes Substitut, welches richtig gerankt wurde „*belohnt*", indem sein Gewicht aus dem Gold Standard mitberücksichtigt wird (siehe Formel (5)). Das Gewicht entspricht dabei der Häufigkeit der annotierten Substitute. Die GAP-Werte erstrecken sich zwischen dem Wert 0 und 1. Der Wert 1 bedeutet, dass ein perfektes Ranking vorliegt, in dem alle richtigen Substitute die falschen und alle höhergewichteten Substitute die niedriger-gewichteten übertreffen.

$$p_i = \frac{\sum_{k=1}^{i} x_k}{i} \tag{5}$$

Dabei ist P_i der jeweilige Präzisionswert an der Stelle i.

$$R' = \sum_{i=1}^{R} I(y_i)\, \bar{y}_i \tag{6}$$

Die Variable x_i entspricht in diesem Fall dem Gewicht des idealen bzw. annotierten Substituts, falls es in dem *Gold Standard (CoInCo)* als genanntes Substitut auftaucht. Andernfalls ist der Wert 0. Die Variable $\bar{y}_i$(Formel (6)) ist das Durchschnittsgewicht der ideal gerankten Liste $y_1, \ldots, y_i$ aus den annotierten Substituten im Gold Standard.

$$I(x_i) = \begin{cases} 1\ wenn\ x_i > 0 \\ 0, andernfalls \end{cases} \tag{7}$$

$$GAP = \frac{\sum_{i=1}^{n} I(x_i) p_i}{R'} \tag{8}$$

Zur besseren Verständlichkeit werden die zwei Experimente anhand eines Beispiels für das Testwort ***agreement*** verdeutlicht. Dieses kommt in den folgenden Sätzen als Zielwort vor:

*„First of America, which now has 45 banks and $12.5 billion in assets, announced an **agreement** to acquire the Peoria, Ill., bank holding in January.*

Die hierzu in CoInCo annotierten Substitute mit der jeweiligen Frequenz in Klammern sind: *deal (2), pact (1), buy (1), buyout (1), conclusion (1), offer (1), settlement (1).*

*The offer is being launched pursuant to a previously announced **agreement** between the companies.*

Im Vergleich hierzu, wurden im zweiten Satz für das Zielwort folgende Substitute annotiert: *arrangement (2), accord (1), contract (1), deal (1), pact (1), resolution (1), understanding (1)*. Die linke Spalte der Tabelle 5 zeigt nun, die für die erste Instanz des Zielwortes annotierten Substitute, welche nach ihrem Gewicht sortiert sind.

Testwort: „agreement“	
annotierte Substitute für die erste Instanz	alle Substitute für alle Instantzen des Testwortes
deal (2)	pact
pact (1)	resolution
buy (1)	contract
buyout (1)	arrangement
conclusion (1)	deal
offer (1)	conclusion
settlement (1)	settlement
	buyout
	understanding
	offer
	buy
	accord

Tabelle 5: Beispiel für das erste Experiment für Testwort *„agreement“*

Die rechte Spalte in der Tabelle 5 wurde anhand der semantischen Ähnlichkeit zwischen dem Testwort *agreement* und den möglichen Substitutionskandidaten aller Instanzen des Testwortes gerankt. Somit kann man den GAP-Wert für die erste Instanz des Testwortes berechnen, welcher sich wie folgt für das Beispiel ergibt:

$$GAP = \frac{1 + \frac{3}{5} + \frac{4}{6} + \frac{5}{7} + \frac{6}{8} + \frac{7}{10} + \frac{8}{11}}{2 + \frac{3}{2} + \frac{4}{3} + \frac{5}{4} + \frac{6}{5} + \frac{7}{6} + \frac{8}{7}} = 0{,}538 = 53{,}8\%$$

Das Ergebnis sagt aus, dass die Liste der vorhergesagten Substitute unter Berücksichtigung der jeweilig annotierte Substitute für das Testwort aus dem Gold Standard, 53,8% präzise sind. Diese Berechnung kann man analog für die zweite Instanz des Testwortes durchführen und auf dieselbe Art den GAP-Wert berechnen. Nachdem man für alle Testwörter den GAP-Wert errechnet hat, kann man den Durchschnitt berechnen und somit bestimmen, wie präzise das Verfahren beim Ranking ist. Nach der Evaluierung des ersten Experiments hat man einen gesamten GAP-Wert von 0,3623 erhalten, welcher einer 36,23% Präzision entspricht. Interessant wird es im zweiten Experiment, wenn man nicht nur alle Substitute für alle Instanzen des Testwortes betrachtet, sondern die Menge aller Substitute für alle Zielwörter. Das bedeutet, dass die rechte Spalte in der Tabelle 5 mit allen potentiellen Substitutionskandidaten für jedes Testwort anhand der semantischen Ähnlichkeit gerankt wird.
Das zweite Experiment wird wiederrum anhand des ersten Beispiels erklärt, um den Unterschied zu verdeutlichen. Die Zahl in den Klammern bei den Substituten in der linken Spalte der Tabelle 6 ist weiterhin das Gewicht. Zur Vereinfachung des Beispiels entsprechen die Zahlen in den Klammern für *alle Substitutionskandidaten* in der rechten Spalte der jeweiligen Position nach dem Ranking. Diese wurden ebenso nach der semantischen Ähnlichkeit gerankt, diesmal aber unter Berücksichtigung *aller Substitute*.

Testwort: „agreement“	
annotierte Substitute	Menge *aller* Substitute für *alle* Zielwörter
deal (2)	pact (27)
pact (1)	deal (1221)
buy (1)	conclusion (1255)
buyout (1)	settlement (1343)
conclusion (1)	buyout (2033)
offer (1)	offer (3940)
settlement (1)	buy (5377)

Tabelle 6: Beispiel für das zweite Experiment für das Testwort „*agreement*"

Wie man deutlich sehen kann, gibt es Wörter die höher gerankt sind als die im Gold Standard annotierten. Dadurch, dass *alle* Substitute betrachtet werden, wirkt sich das Ranking sehr stark auf die Position der annotierten Substitute in der Liste der vorhergesagten aus. Das bedeutet, dass der $p_i - Wert$ und folglich auch der GAP-Wert kleiner wird. Die mittels semantischer Ähnlichkeit besser gerankten Wörter sind False Positive

Ergebnisse. Das bedeutet, es werden fälschlicherweise Substitute vorausgesagt, die keine sind, wie zum Beispiel *curve*, *workshop, meeting, editorial,* welche auf die ersten vier Positionen nach semantischer Ähnlichkeit zum Zielwort *agreement* gerankt wurden. Die falsch vorgeschlagenen Substitute drängen die richtigen Substitute in der Rankingliste nach unten und verschlechtern mit ihrem Rang folglich den GAP- Wert.
Im Gegensatz dazu, können mittels semantischer Ähnlichkeit Wörter vorgeschlagen werden, die richtige Substitute sind, die aber a priori nicht genannt wurden. Jedoch war dem nicht so, sondern es handelte sich um „echte Fehler". Somit errechnet sich der GAP-Wert für das erste Testwort in Bezug auf *alle Substitutionskandidaten* wie folgt:

$$GAP = \frac{\frac{1}{27} + \frac{3}{1221} + \frac{4}{1255} + \frac{5}{1343} + \frac{6}{2033} + \frac{7}{3940} + \frac{8}{5370}}{2 + \frac{3}{2} + \frac{4}{3} + \frac{5}{4} + \frac{6}{5} + \frac{7}{6} + \frac{8}{7}} = 0{,}0054 = \ 0.54\%$$

Wie man deutlich sehen kann, hat sich das Ergebnis gegenüber dem ersten Experiment drastisch verschlechtert. Die Präzision ist um genau 53,3% gesunken. Das Ergebnis sagt aus, dass die Liste aller vorhergesagten Substitute unter Berücksichtigung der jeweilig annotierte Substitute für das Testwort aus dem Gold Standard, nur 0,5% präzise sind. Nachdem für alle Testwörter der GAP-Wert errechnet wurde, hatte man für die Evaluierung des zweiten Experiments einen gesamten GAP-Wert von 0,028 erhalten, welches einer 2,8% Präzision entspricht. Im Gegensatz zum Standardverfahren, liefert das Modell von (Thater et al. 2011) beim Ranking *aller Substitute* von *allen Zielwörtern* kein gutes Ergebnis für die Präzision. Die Präzision hat sich insgesamt um 33,43% verschlechtert. Die Tabelle 7 zeigt den Vergleich des Rankingmodells mit den Ergebnissen aus (Kremer et al. 2014) hinsichtlich des prozentualen GAP-Wertes. Die Abkürzung (TFP11-2) in der orange markierte Spalte repräsentiert die Ergebnisse dieser Arbeit unter Verwendung des Modells von (Thater et al. 2011).

Corpus		Syntactically structured			Syntactically filtered		Bag of words		TFP11-2
		TFP11	TFP10	EP08	TFP11/EP08	TFP10	TFP11/EP08	TFP10	
MASC LexSub	Ranking nach Experimet 1	**47.8**	46.0	47.4	47.4	41.9	46.2	40.8	**36,23**
	Ranking nach Experiment 2	-	-	-	-	-	-	-	**2,8**

Tabelle 7: Vergleich der GAP-Ergebnisse aus (Kremer et al. 2014) und dieser Evaluierung

Der Unterschied zwischen den Ergebnissen des Rankings nach dem Standardverfahren, welches in Tabelle 7 dargestellt wird, liegt in der Extraktion der Kookkurrenzen aus dem Gigaword-Korpus. Für diese Arbeit hat man gegenüber (Kremer et al. 2014) das Modell mit insgesamt 12 XIN-Dateien aus dem kompletten Jahr 2010 trainiert. Dadurch kann es zu unterschiedlichen Frequenzen und Wortarten kommen. Außerdem wirkt sich das Training des Modells auf das Endergebnis aus, da das Verfahren durch das Training mit mehreren Dokumenten genauer wird. Das Hauptinteresse dieser Arbeit lag darin, herauszufinden, um wie viel schlechter das Modell wird, wenn es alle potentiellen Sub-

stitute aller Zielwörter ranken soll. Dies wurde mit dem zweiten Experiment bewiesen. Das Modell hat sich in Bezug auf das Ergebnis von (Kremer et al. 2014) um genau 45% und in Bezug auf das Ergebnis im ersten Experiment dieser Arbeit um 33,43% verschlechtert. Mögliche Gründe für die Verschlechterung sind beispielsweise die Extraktion der Kookkurrenzen aus Gigaword, bei dem False Positive Ergebnisse vorausgesagt werden und es zusätzlich viel mehr Wörter anhand der semantischen Ähnlichkeit ranken muss. Die semantische Ähnlichkeit zwischen den potentiellen Substituten wurde diesbezüglich auch anhand der Kookkurrenzen aus Gigaword berechnet. Dabei kann es dazu kommen, dass Wörter, die überhaupt keinen Zusammenhang besitzen, durch Vergleich ihrer Kontexte als ähnlich dargestellt werden.

5 Zusammenfassung

Diese Bachelorarbeit befasste sich mit der vokabular-globalen lexikalischen Substitution in einem Vektorraumodell. Im Gegensatz zu vielen bereits existierenden Arbeiten aus dem NLP, die sich mit der Auflösung der Wortmehrdeutigkeit (WSD) beschäftigen, stand der Fokus dieser Arbeit auf der Evaluierung einer *lexikalischen Substitution.* Dabei spezifizierte man sich auf das Ranking der Substitute. Bisherige Ansätze in diesem Bereich haben gezeigt, dass bezüglich des Rankings der potentiellen Substitute nur diejenigen betrachtet werden, die für das spezifische Zielwort annotiert sind.
Ziel dieser Arbeit war es, zu untersuchen, inwieweit sich das Rankingverfahren verschlechtert, wenn alle Wörter im Vokabular bzw. alle annotierten Substitute aller Zielwörter gerankt werden. Dies wurde anhand des unüberwachten Modells von (Thater et al. 2011) realisiert und auf einem großen, alle Wörter umfassenden Datensatz evaluiert.
Die Ergebnisse der Evaluation zeigten, dass sich die Präzision beim Ranking aller möglichen Substitute im Vergleich zum Standardverfahren erheblich verschlechtert. Es wurde im Gegensatz zu 36,23% lediglich eine *Generalized Average Precision* von 2,8% erreicht. Wie zu erwarten war, verringerte sich die Präzisionsrate, indem die Menge der zu rankenden Substitute vergrößert wurde. Nach Anwendung des Modells von (Thater et al. 2011) auf beiden Rankingexperimenten, konnte man feststellen, dass das Verfahren einen großen Unterschied zwischen beiden Evaluierungen zeigt.

Eine mögliche Verbesserung erscheint sinnvoll, indem eine Vorfilterung unter dem Vokabular der potentiellen Substitute durchgeführt wird. Dabei würden nur diejenigen Substitute verwendet werden, die tatsächlich ähnlich zueinander sind. Somit würde sich die Anzahl der potentiellen Substitute verringern. Dies könnte beispielsweise unter Verwendung eines separaten Bedeutungsinventars umgesetzt werden. Diese Umsetzung muss aber anhand einer neuen Evaluierung realisiert werden.

Abbildungsverzeichnis

Abbildung 1: Wortbedeutung in Bezug auf Granularität (Edmonds 2006) 8
Abbildung 2: Kontextsätze für das Zielwort "charge" 14
Abbildung 3: Supervised WSD Methode 17
Abbildung 4: Unüberwachte WSD Methode 19
Abbildung 5: Beispiel des "Clustering" 20
Abbildung 6: Beispiel des Zuweisungsalgorithmus 21
Abbildung 7: "Co-occurrence-Graph" 22
Abbildung 8: Wissensbasierte WSD-Methode 26
Abbildung 9: Ausschnitt aus dem CoInCo - Datensatz 27
Abbildung 10: Ausschnitt aus Gigaword-Datensatz LTW_ENG 29
Abbildung 11: Ausschnitt aus dem annotierten Gigaword-Datensatz AFP_ENG 31
Abbildung 12: Ausschnitt einer Tokenisierung eines Paragraphe 31
Abbildung 13: Abhängigkeiten der Wörter im Datensatz 32
Abbildung 14: Basisvektor von "charge" 34
Abbildung 15: Strikte Kontextualisierung 34
Abbildung 16: Kontextualisierung nach semantischer Ähnlichkeit 34

Tabellenverzeichnis

Tabelle 1: Aufbau von WordNet 9
Tabelle 2: SemEval 2007 - Gold Standard Substitute 14
Tabelle 3: Gigaword: Größe der sieben Daten 30
Tabelle 4: Vergleich drei Paraphrasen-Ranking-Modellen auf verschiedenen Korpora 37
Tabelle 5: Beispiel für das erste Experiment für Testwort „*agreement*“ 39
Tabelle 6: Beispiel für das zweite Experiment für das Testwort „*agreement*" 40
Tabelle 7: : Vergleich der GAP-Ergebnisse aus (Kremer et al. 2014) und dieser Evaluierung 41

Literaturverzeichnis

Agirre, E. et al., 2009. A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 19–27.

Agirre, E., Edmonds, P. & others, 2006. Word Sense Disambiguation: Algorithms And Applications.

Bär, D. et al., 2012. Ukp: Computing semantic textual similarity by combining multiple content similarity measures. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*. Association for Computational Linguistics, pp. 435–440.

Biemann, C., 2013. Creating a system for lexical substitutions from scratch using crowdsourcing. *Language Resources and Evaluation*, 47(1), pp.97–122.

Buechel, G. Prof. Dr. phil., 2010. Maschinelle Verarbeitung natürlicher Sprache - Coputerlinguistik. Available at: www.nt.fh-koeln.de/fachgebieteinf/buechel/DBW8.pdf.

Bußmann, H., 2008. Lexikon der Sprachwissenschaft. Vierte, durchgesehene und bibliographisch ergänzte Auflage unter Mitarbeit von Hartmut Lauffer.

Carol Friedman, J.-W.F. und, 2008. Word Sense Disambiguation via Semantic Type Classification. In AMIA Annual Symposium Proceedings Archive. pp. 177–181.

Carstensen, K.-U., 2012. Sprachtechnologie - Ein Überblick. Available at: http://kai-uwe-carstensen.de/Publikationen/Sprachtechnologie.pdf.

Church, K.W. & Hanks, P., 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1), pp.22–29.

Edmonds, P., 2006. Lexical disambiguation. , Elsevier Encyclopedia of Language and Linguistics(2. Ausgabe), pp.607–630.

Erk, K. & Padó, S., 2008. A structured vector space model for word meaning in context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pp. 897–906. Available at: http://dl.acm.org/citation.cfm?id=1613831.

Fellbaum, C., 1998. *WordNet*, Wiley Online Library.

Hassan, S. et al., 2007. Unt: Subfinder: Combining knowledge sources for automatic lexical substitution. In *Proceedings of the 4th International Workshop on Semantic Evaluations*. Association for Computational Linguistics, pp. 410–413.

Kishida, K., 2005. *Property of average precision and its generalization: An examination of evaluation indicator for information retrieval experiments*, National Institute of Informatics Tokyo, Japan.

Kremer, G. et al., 2014. What Substitutes Tell Us – Analysis of an “All-Words” Lexical Substitution Corpus. In *Proceedings of EACL*. Gothenburg, Sweden.

Manning, C.D. & Schütze, H., 1999. *Foundations of statistical natural language processing*, MIT press.

McCarthy, D., 2009. Word sense disambiguation: An overview. *Language and Linguistics compass*, 3(2), pp.537–558.

McCarthy, D. & Navigli, R., 2009. The English lexical substitution task. *Language resources and evaluation*, 43(2), pp.139–159. Available at: http://link.springer.com/article/10.1007/s10579-009-9084-1.

McCarthy, D., Sinha, R. & Mihalcea, R., 2013. The cross-lingual lexical substitution task. *Language resources and evaluation*, 47(3), pp.607–638.

McInnes, B.T., 2009. *Supervised and Knowledge-based Methods for Disambiguating Terms in Biomedical Text using the UMLS and MetaMap*. University of Minnesota, Thesis.

Miller, T. et al., 2012. Using Distributional Similarity for Lexical Expansion in Knowledge-based Word Sense Disambiguation. In *COLING*. pp. 1781–1796.

Mitchell, J. & Lapata, M., 2008. Vector-based Models of Semantic Composition. In *ACL*. Citeseer, pp. 236–244.

Mohler, M. & Mihalcea, R., 2009. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 567–575.

Napoles, C., Gormley, M. & Van Durme, B., 2012. Annotated gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*. Association for Computational Linguistics, pp. 95–100. Available at: http://dl.acm.org/citation.cfm?id=2391218.

Navigli, R., 2009. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2), p.10.

Navigli, R. & Ponzetto, S.P., 2010. BabelNet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the association for computational linguistics*. Association for Computational Linguistics, pp. 216–225.

Navigli, R. & Ponzetto, S.P., 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193, pp.217–250.

Parker, R. et al., 2011. English gigaword fifth edition, june. *Linguistic Data Consortium, LDC2011T07*.

Pedersen, T. & Mihalcea, R., 2005. Advances in word sense disambiguation. In *Tutorial, Conf of ACL*.

Ponzetto, S.P. & Navigli, R., 2010. Knowledge-rich word sense disambiguation rivaling supervised systems. In *Proceedings of the 48th annual meeting of the association for computational linguistics*. Association for Computational Linguistics, pp. 1522–1531.

Schütze, H., 1998. Automatic word sense discrimination. *Computational linguistics*, 24(1), pp.97–123. Available at: http://dl.acm.org/citation.cfm?id=972724.

Sharoff, S., 2006. Open-source corpora: Using the net to fish for linguistic data. *International journal of corpus linguistics*, 11(4), pp.435–462. Available at: http://www.ingentaconnect.com/content/jbp/ijcl/2006/00000011/00000004/art00004.

Sinha, R. & Mihalcea, R., 2014. Explorations in lexical sample and all-words lexical substitution. *Natural Language Engineering*, 20(01), pp.99–129.

Szarvas, G. et al., 2013. Supervised All-Words Lexical Substitution using Delexicalized Features. In *HLT-NAACL*. pp. 1131–1141.

Thater, S., Fürstenau, H. & Pinkal, M., 2010. Contextualizing semantic representations using syntactically enriched vector models. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 948–957. Available at: http://dl.acm.org/citation.cfm?id=1858778.

Thater, S., Fürstenau, H. & Pinkal, M., 2011. Word Meaning in Context: A Simple and Effective Vector Model. In *IJCNLP*. pp. 1134–1143.

Wagstaff, K. & Cardie, C., 2000. Clustering with instance-level constraints. *AAAI/IAAI*, 1097.

Yarowsky, D., 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, pp. 189–196. Available at: http://dl.acm.org/citation.cfm?id=981684.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich und sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift