Das Structure Discovery Paradigma: Unüberwachte und Vorwissensfreie Automatische Sprachverarbeitung

Chris Biemann

Abteilung Automatische Sprachverarbeitung
Institut für Informatik
Fakultät für Mathematik und Informatik
Universität Leipzig
biem@informatik.uni-leipzig.de

Abstract: Das Ziel der hier zusammengefassten Dissertation [Bie07b] besteht darin zu zeigen, dass die Verarbeitung natürlichsprachlichen Textes auch ohne die Bereitstellung von sprachspezifischem Vorwissen möglich ist. Das hier eingeführte *Structure Discovery* Paradigma realisiert dies durch einen iterativen Prozess, welcher a) Struktur in unstrukturiertem Textmaterial erkennt und b) diese in demselben Material explizit macht. Spätere Iterationsschritte können unter Ausnutzung der dadurch vorliegenden Strukturinformation weitere, komplexere Strukturen erkennen.
Graphen bzw. Netzwerke stellen einen intuitiven Weg dar, um (sprachliche) Einheiten als Knoten und deren Verbindungen als Kanten zu kodieren. Hier werden sie durchgehend als Datenstruktur zur Repräsentation verwendet. Zunächst werden quantitative Betrachtungen von Sprachdaten, genauer: von Graphen, welche aus Sprachdaten erstellt wurden, durchgefürt. Um quantitative Eigenschaften echter Sprachdatengraphen besser als vorhergehende Modelle zu reproduzieren, wird ein generatives Modell entwickelt, welches gleichzeitig einen neuartigen Ansatz zur Erklärung von Sprachevolution darstellt.
Das Erkennen von Struktur erfordert das Abstrahieren von Einzelbeispielen und das Gruppieren von verschiedenen Einheiten. Entsprechend der hier gewählten Repräsentation wird der *Chinese Whispers* Algorithmus definiert, welcher eine Partition der Knoten eines Graphen liefert. Das extrem günstige Laufzeitverhalten dieses Algorithmus ermöglicht das effiziente Clustern von Graphen mit mehreren Millionen Knoten, wie sie in der Sprachverarbeitung üblicherweise auftreten.
Im praktischen Teil der Arbeit wird Structure Discovery anhand von drei Problemen exemplifiziert. Es gelingt, mehrsprachige Textsammlungen in ihre Einzelsprachen zu zerlegen. Das darauf aufbauende System zur automatischen Zuweisung von Wortarten schneidet in einer anwendungsbasierten Evaluation gleichwertig mit Wortartenprogrammen ab, welche durch maschinelles Lernen auf manuell erstellten Daten trainiert werden. Zum Abschluss wird das Erkennen und Auflösen von semantischer Mehrdeutigkeit diskutiert.
In der Arbeit wird gezeigt, dass es mit Structure Discovery möglich ist, Verarbeitungsschritte von Sprachdaten hervorzubringen, deren Qualität mit wissensintensiven Methoden vergleichbar ist. Die Vorwissensfreiheit des Paradigmas ermöglicht dieses Vorgehen für alle Spachen und Sachgebiete – es muss lediglich genügend Rohtext bereitgestellt werden.

1 Einführung

Die automatische Verarbeitung natürlicher Sprache gehört gleichzeitig zu den ältesten und zu den schwierigsten Teilgebieten der Informatik. Der Menschheitstraum, eine denkende Maschine zu konstruieren, ist eng mit dem automatischen Verstehen von Sprache verknüpft. Deren Komplexität ist es geschuldet, dass es auch nach einem Schock Jahren kein System gibt, das der menschlichen Sprachverarbeitungsleistung auch nur nahe kommt.

Bisher basierte die Automatische Sprachverarbeitung (ASV) immer auf *explizitem* oder *implizitem* linguistischen Wissen. Klassische computerlinguistische Ansätze befassen sich mit dem Erstellen von Regelsystemen, während das maschinelle Lernen durch manuelle Annotation gegebenes Wissen nutzt, um vergleichbare Markierungen mit Hilfe überwachten Lernens zu generieren. Im Gegensatz dazu erfolgt menschlicher (Erst-)spracherwerb größtenteils unüberwacht: Allein die Präsentation einer Vielzahl von Sprachbeispielen löst Lernprozesse aus, welche die für das Verstehen und Sprechen einer Sprache notwendigen Abstraktionen und Generalisierungen zur Verfügung stellen.

Diese Arbeit zielt auf die Beantwortung folgender Fragen: Was kann in der ASV ohne Benutzung sprachspezifischen Wissens erreicht werden? Wieviel manuelle Arbeit ist für welches Maß an Verarbeitung wirklich notwendig?

Zu deren Beantwortung muss konsequenterweise ein Paradigmenwechsel stattfinden: Anstatt Maschinen die Verarbeitung von Sprache direkt beizubringen, werden sie hier mit der Fähigkeit ausgestattet, strukturelle Regularitäten in Textsammlungen[1] zu entdecken. Der Arbeitsaufwand wird von der manuellen Erstellung von Regeln oder Annotationen dahingehend verschoben, dass nun Prozesse definiert werden, welche notwendigerweise vorhandene Strukturen in Sprachdaten erkennen und ausnutzen können. Auf diese Weise wird ein Inventar von Mechanismen aufgebaut, welche – sobald sie auf einigen Datensätzen und Anwendungen validiert – dahingehend universell sind, dass sie auf neuen Daten mit ähnlicher Struktur ähnliche Resultate liefern. Diese enorme Erweiterung des sogenannten “acquisition bottleneck” ermöglicht eine vereinheitlichte Verarbeitung von Sprachdaten und stellt einen beschleunigten Zugang zu bisher nicht bearbeiteten Sprachen oder Domänen zur Verfügung.

In einer Zeit, in der Rechengeschwindigkeit, Speichermedien und die Verfügbarkeit elektronischer Texte ausreichenden Umfang für dieses Unterfangen erreicht haben, befinden wir uns erstmals in der Situation, dass uns Maschinen auf diese Weise die Hauptarbeit abnehmen können. Dem Problem der dünnen Datenlage (data sparseness) begegnen wir einfach dadurch, dass wir angesprochenen Prozessen mehr Rohdaten zur Verfügung stellen.

[1] In diesem Beitrag werden die Begriffe Textsammlung, Text, Textmenge und Sprachdaten austauschbar verwendet; hier wurden nur Betrachtungen zu geschriebenen, elektronisch verfügbaren Sprachdaten vorgenommen, da weitere Formate den Rahmen der Arbeit gesprengt hätten. Die Validität von Structure Discovery ist jedoch von der Art der Sprachdaten und deren Repräsentation unabhängig.

2 Structure Discovery Paradigma

In diesem Abschnitt werden Hauptaspekte der hier vorgestellten Dissertation zusammengefasst. Abschnitt 2.1 führt in das hier neu definierte Paradigma ein. Die Wahl der Graphrepräsentation wird in Abschnitt 2.2 motiviert. In Abschnitt 2.3 wird ein generatives Modell beschrieben, welches Zufallstext mit ähnlichen Eigenschaften wie natürliche Sprache liefert. Ein neues Graphclusteringverahren wird in Abschnitt 2.4 vorgestellt, welches dann im Abschnitt 2.5 in Structure Discovery Prozessen eingesetzt wird.

2.1 Einführung in Structure Discovery

Das Structure Discovery (SD) Paradigma ist ein Framework für das Lernen von strukturellen Regularitäten in großen Mengen natürlichsprachlichen Textes und für deren Explizitmachung in selbigem durch Selbstannotation. Im Kontrast zu den in der Automatischen Sprachverarbeitung vorherrschenden Paradigmen benötigt SD weder sprachspezifisches Vorwissen noch überwachtes Lernen und bleibt deshalb unabhängig von Sprache und Sachgebiet. Vielmehr meint das Arbeiten innerhalb dieses Paradigmas das Erstellen von Entdeckungsprozessen, welche von Rohtextmaterial ausgehend die vorliegenden Daten iterativ anreichern, wobei Selbstannotationen aus vorhergegangenen Iterationen verwendet werden können.
Das neue Paradigma gipfelt in der Vision der Structure Discovery Maschine (SDM), welche die Gesamtheit aller mit algorithmischen Mitteln erkennbaren Strukturregularitäten in Sprachdaten erkennt und explizit macht. In ihrer höchsten und reinsten Form erfolgt das Assemblieren der SDM nicht mehr manuell; die SDM konstruiert sich vielmehr selbst, hierbei verschiedene parametrisierbare Prozeduren kombinierend und den Grad der Generalisierung über die Daten optimierend.
Die Ausgabe einer SDM ist eine multidimensionale Annotation des Rohtextes mit Markierungen, welche strukturell gleichartige Phänomene als solche kennzeichnen. Einige dieser Annotationen sind unabhängig voneinander, jedoch hängen die meisten voneinander ab und bauen aufeinander auf. Dieser holistische Ansatz für datengetriebene Selbstannotation generiert eher über denn unter. Um nützliche von überflüssigen Annotationen zu unterscheiden, muss die Ausgabe der SDM in praktischen Anwendungen daraufhin getestet werden, welche Annotationen für welche Art von Aufgaben verwertbar sind. Abbildung 1 verdeutlicht beispielhaft die Art von Strukturinformation, auf welche hier abgezielt wird.

2.2 Graphen und Topologie von Netzwerken

Kurz vor der Jahrtausendwende erlebte das sich mit Zufallsgraphen befassende Teilgebiet der Graphentheorie verstärktes fachübergreifendes Interesse, getrieben von der Entdeckung, dass viele natürliche und künstliche Graphen (wie z.B. das Internet, soziale Netzwerke, metabolische Netzwerke u.v.m.) zwei Eigenschaften gemeinsam haben: Die

```
<sentence lang=12, subj=34.11>
 <chunk id=c25>
   <word POS=p3 m=0.0 s=s14>In=creas-ed</word>
  <MWU POS=p1 s=s33>
    <word POS=p1 m=5.1 s=s44>interest</word>
    <word POS=p1 m=2.12 s=s106>rate-s</word> </MWU> </chunk>
 <chunk id=c13>
  <MWU POS=p2>
    <word POS=p2 m=17.3 s=74>lead</word>
    <word POS=p117 m=11.98>to</word> </MWU> </chunk>
 <chunk id=c31>
    <word POS=p1 m=1.3 s=33>investment-s</word>
    <word POS=p118 m=11.36>in</word>
    <word POS=p1 m=1.12 s=33>bank-s</word> </chunk>
 <word POS=p298> . </word>
</sentence>
```

Abbildung 1: Einige Structure Discovery Beispielannotation für den engl. Satz “Increased interest rates lead to investments in banks”. Die unterstrichenen Markierungen werden im Abschnitt 2.5 genauer beschrieben. Legende: *lang* bezieht sich auf die Erkennung der Sprache, *POS* referenziert die Wortart und *m* markiert verschiedene Bedeutungen.

Kleine-Welt-Eigenschaft (small world property) sowie Skalenfreiheit (siehe [Bar03] zur Einführung). Daran anschließende Arbeiten brachten Netzwerkgeneratoren hervor, welche verschiedenen in der Natur beobachtbaren Grapheigenschaften Rechnung tragen und deren Ursprung durch Angabe eines solchen Generators erklären.
Skalenfreie Netzwerke weisen keine typische Anzahl von Verbindungen pro Knoten auf, ihr Verlinkungsgrad folgt keiner Skala. Die Verteilung der Knotengrade folgt einem Potenzgesetz. Die Kleine-Welt-Eigenschaft bezeichnet das Phänomen, dass überraschend kurze Wege zwischen allen Knoten des Netzwerkes existieren, obwohl die Gesamtzahl an Kanten vergleichsweise gering ist.
Auch lexikalische Netzwerke besitzen diese Eigenschaften [ST05], welche die seit [Zip49] bekannte Omnipräsenz von Potenzgesetz-Verteilungen in Sprachdaten erklären. Dies heißt mit anderen Worten, dass die meisten sprachlichen Einheiten selten und mit wenigen verbunden sind, während andere extrem konnektives Verhalten an den Tag legen.
Das insbesondere für semantische Räume gern verwendete Vektorraummodell (vergleiche [Sch93]) ist in seiner reinen From für solch stark verzerrt verteilte Daten ungeeignet, weswegen häufig rechenintensive Dimensionsreduktionsmethoden zum Ausgleich von Effizienz- und Effektivitätsproblemen eingesetzt werden. Die Graphrepräsentation stellt hier eine günstige Alternative dar, da das Äquivalent von Nullwerten weder repräsentiert noch verarbeitet werden muss.

2.3 Generierung von Kookkurrenznetzwerken

Kookkurrenz ist ein wichtiger Baustein für das Betrachten von Wörtern in ihrem Kontext, insbesondere wenn keine sprachspezifischen Vorverarbeitungsschritte zur Verfügung stehen. Zwei Wörter kookkurrieren, falls sie zusammen in einer Informationsseinheit auftre-

ten. Hier werden zwei Arten von Informationseinheiten verwendet: die Satzebene und die Nachbarschaftsebene; die korrespondierenden Kookkurrenzen bezeichnet man als Satzkookkurrenzen bzw. Nachbarschaftskookkurrenzen [Qua98].
Um zufallsbedingte von inhaltlich motivierten Kookkurrenzen zu trennen, wird ein Signifikanzmaß angewendet. Dieses entscheidet auf Basis der Einzelwortfrequenzen und deren gemeinsamer Auftretenshäufigkeit, mit welcher Signifikanz die Hypothese zurückgewiesen werden kann, dass die beiden Wörter unabhängig voneinander auftreten. Der einer Textsammlung zugeordnete Kookkurrenzgraph ist die Gesamtheit aller Kookkurrenzen, wobei Wörter als Knoten und Signifikanzwerte als Kantengewichte aufgefasst werden[2].
Durch Betrachtung von Kookkurrenznetzwerken für viele verschiedene Sprachen wird hier empirisch nachgewiesen, dass sowohl Nachbarschafts- als auch Satzkookurrenzgraphen skalenfreie Kleine-Welt Graphen sind. Der Exponent der Knotengradverteilung ist nahe 2, was mit den existierenden Netzwerkgeneratoren nicht oder nur durch den Verlust anderer Eigenschaften produziert werden kann.
Um diese Diskrepanz auszuräumen, wird ein Generator benötigt, der dem Charakter von Sprache als linearer Symbolfolge Rechnung trägt. Genauso wie Kookkurenzgraphen aus einer realen Textsammlung generiert werden, ist es möglich, Netzwerke aus Zufallstext zu generieren.
Existierende Zufallstextmodelle stellen sich als ungeeignet für die Reproduktion der Eigenschaften von Kookkurrenzgraphen und anderen Eigenschaften natürlicher Sprache heraus, weswegen ein neues Zufallstextmodell entwickelt wird, siehe [Bie07a]. Dieses Zufallstextmodell führt erstmals den Satz als Einheit in Zufallstext ein. Es basiert auf der Annahme, dass Wortfolgen mit höherer Wahrscheinlichkeit generiert werden, je öfter sie bereits generiert wurden. Realisiert wird der Generator als Zufallsbewegung (random walk) auf einem gerichteten Wortnachbarschaftsgraph zwischen Satzanfang und Satzende, wobei der Graph durch die Produktion neuer Wörter erweitert wird.
Ein Vergleich mit natürlichsprachlichem Text zeigt, dass die Verteilungen der Wortlängen, Satzlängen, Wortfrequenzen und der Knotengrade in Kookkurrenzgraphen vom Zufallstextmodell reproduziert und somit völlig emergent erklärt werden. Dessen einfachen Aufbau eingedenk, stellt dieses Zufallstextmodell eine plausible Erklärung für eine Reihe von Sprachuniversalien zu Verfügung, ohne auf linguistische Konzepte aus der Syntax oder Semantik zurückzugreifen.

2.4 Graphclustering für Structure Discovery

Um Struktur auf unüberwachte Weise in Sprache zu entdecken, müssen sprachliche Einheiten durch Ähnlichkeitsmaße miteinander in Beziehung gesetzt werden. Clusteringmethoden können diese dann in Cluster gruppieren, was Abstraktion und Generalisierung realisiert. Insbesondere im Kontext der Sprachverarbeitung stößt die Anwendung der meisten herkömmlichen Clusteringverfahren auf folgende Probleme:

- Vorgegebene Anzahl von Clustern. Dies ist z.B. für das Finden verschiedener Wort-

[2]Eine Open-Source-Implementierung des Autors zur Erstellung von Kookkurrenzgraphen aus Texten verschiedener Formate ist verfügbar auf http://wortschatz.uni-leipzig.de/~cbiemann/software/TinyCC2.html

bedeutungen inakzeptabel: Wüsste man die Anzahl der Bedeutungen vorher, bräuchte man nicht mehr zu clustern.

- Balancierte Clustergrößen. Einige, z.B. im Bereich VLSI[3] eingesetzte Verfahren zielen auf Partitionen ungefähr gleicher Größe, was für Sprachdaten mit ihren zahlreichen Potenzgesetz-verteilten Größen ungeeignet ist.
- Zu hohes Laufzeitverhalten. Für das Clustern von Kookkurrenzgraphen mit i.A. einigen Millionen Knoten wird ein effizientes Verfahren benötigt.

Zur Umgehung dieser Probleme wurde das Graphclusteringverfahren *Chinese Whispers* (CW) [Bie06a] für ungerichtete, gewichtete Graphen entwickelt[4]. CW ist definiert in Abbildung 2 und weist den Knoten des zu clusternden Graphen Klassen zu, welche als verschiedene Cluster interpretiert werden.

CW(graph $G(V, E)$**):**

for all $v_i \in V$ **do**
  $class(v_i) = i$
end for
for it=1 to Anzahl-Iterationen **do**
  for all $v \in V$, zufällige Reihenfolge **do**
    $class(v)$=vorherrschende Klasse in $neigh(v)$
  end for
end for
return Partition P gegeben durch Klassen $class$

Abbildung 2: Chinese Whispers Algorithmus

Hiebei ist die Nachbarschaft $neigh(v)$ eines Knotens v definiert als alle Knoten, mit denen v eine Kante teilt. Die vorherrschende Klasse in dieser Nachbarschaft ist die Klasse derjenigen Knotengruppe, welche die höchste Summe der Kantengewichte zu v aufweist. CW macht keine Annahmen über die Anzahl oder die Größenverteilung der Cluster. Die Laufzeit verhält sich linear in der Anzahl der Kanten des Graphen, was das untere Limit für Methoden, welche den gesamten Graphen betrachten, darstellt. CW arbeitet asynchron und ist leicht parallelisierbar. Hervorzuheben ist ferner, dass veränderte Klassen eines Knotens sofort und nicht erst in der nächsten Iteration zur Verfügung stehen. Dieser nichtdeterministische und formal nicht konvergierende Algorithmus ist parameterfrei und speziell zum Clustern von Kleine-Welt Graphen geeignet, da in diesen die Anzahl Kanten und der Durchmesser vergleichsweise gering ist. Dies wiederum wirkt sich günstig auf die Laufzeit aus.
Erweiterungen des Basisalgorithmus werden diskutiert, um Quasideterminismus, Fuzzy Clustering, feinere Granularität sowie ein flach hierarchisches Clustering zu erreichen. Im folgenden praktischen Teil wird CW für mehrere Structure Discovery Prozesse eingesetzt.

[3] Eine Aufgabe in Very Large Scale Integration ist die (balancierte) Verteilung von elektronischen Schaltungen auf mehrere Chips

[4] Open-source Implementierung auf http://wortschatz.informatik.uni-leipzig.de/~cbiemann/software/CW.html

2.5 Drei Structure Discovery Prozesse

In diesem Abschnitt wird mit drei aufeinander aufbauenden Prozessen das Structure Discovery Paradigma veranschaulicht.

2.5.1 Identifikation von Sprachen

Ein wichtiger Vorverarbeitungsschritt für die meisten Sprachanwendungen ist die Bestimmung der Sprache eines Textes. Die Betrachtung von Satzkookkurrenzgraphen multilingualer Textsammlungen zeigt, dass zwischen Wörtern derselben Sprache viel mehr signifikante Kookkurrenzen auftreten als zwischen Wörtern verschiedener Sprachen. Der Kookkurrenzgraph zerfällt jedoch nicht in nichtverbundene Komponenten, da manche Wörter in mehreren Sprachen auftreten, wie z.B. alle Eigennamen, aber auch häufige Wörter wie 'die' (deutscher Artikel, niederländisches Pronomen, englisches Verb oder Nomen, ...).
Dies ausnutzend wird der Satzkookkurrenzgraph einer multilingualen Textsammlung mit CW geclustert. Die Anzahl übereinstimmender Wörter zwischen Einzelsätzen und diesen Clustern ist dann ein Maß für die Zugehörigkeit von Textstücken zu Sprachen, welche durch die Cluster repräsentiert werden. Auch ohne die beteiligten Sprachen zu kennen, erreicht das Verfahren [BT05] ebenso nahezu 100%ige Genauigkeit wie bekannte, überwachte Verfahren[5]. Selbst Verunreinigungen von wenigen Promille in monolingualen Korpora können auf Satzbasis entdeckt werden. Insbesondere hier erweisen sich CWs Parameterfreiheit sowie die fehlenden Annahmen über Clustergrößenverteilungen als nützlich.

2.5.2 Zuweisen von Wortarten

Einsprachigen Text voraussetzend, wird im folgenden die Zuweisung von Wortarten realisiert, also: die/*Artikel* Markierung/*NomenSg* von/*Präposition* Wörtern/*NomenPl* im/*Präposition* Text/*NomenSg* mit/*Präposition* syntaktischen/*Adjektiv* Kategorien/*NomenPl*.
Herkömmliche, linguistisch motivierte Wortartenmarkierer (Part-of-Speech Tagger) benötigen mehrere zehntausend von Hand annotierte Sätze zum Training, welche aufgrund hoher Kosten bisher nur für eine Handvoll Sprachen zur Verfügung stehen. Ferner gestaltet sich die Domänenanpassung schwierig, z.B. ist die Qualität eines auf Zeitungstext trainierten und auf Emails angewendeten Wortartenmarkierers unbefriedigend, was eine unüberwachte Variante auch für ressourcenreiche Sprachen motiviert.
Die Erstellung eines Inventars für Wortarten wird hier in zwei Schritten realisiert. Häufige, hochfrequente Wörter werden aufgrund der Ähnlichkeit ihrer Zweiwortkontexte geclustert, mittel- und niederfrequente Wörter erfahren eine Gruppierung aufgrund gemeinsamer Nachbarschaftskookkurrenzen.
Im Gegensatz zu in der Literatur bekannten unüberwachten Wortartenmarkierern, welche das Inventar nur aus hochfrequenten Wörtern konstruieren, bringt die hier vorgestellte Methode [Bie06b] weitaus größere Lexika hervor und kann aufgrund der Effizienz der

[5] Open-sorce Implementierung auf http://wortschatz.uni-leipzig.de/~cbiemann/software/langSepP.html

beteiligten Komponenten weitaus mehr Text zum Erstellen des Wortartenmarkierungsmodelles verwenden. Dies wirkt sich direkt auf die Konsistenz der Markierungen aus. Weitere Alleinstellungsmerkmale sind die selektive Aufnahme bei der Erstellung des Inventars (d.h. unter anderem, dass nicht alle Wörter über einer bestimmten Häufigkeit ins Lexikon aufgenommen werden müssen) und die Aufnahmen von Mehrdeutigkeiten in das Lexikon, um Phänomenen wie in 'Er verteidigte/*Verb* die Burg' gegenüber 'Die verteidigte/*Adjektiv* Burg' Rechnung zu tragen.
Die von dieser Methode automatisch induzierten Inventare sind normalerweise feingliedriger als diejenigen in der Linguistik gebräuchlichen. Als Beispiel zeigt Abbildung 3 einen Ausschnitt mit der Unterscheidung zwischen weiblichen und männlichen Vornamen[6]. Die Evaluation als Komponente für eine Reihe von Aufgaben aus dem maschinellen

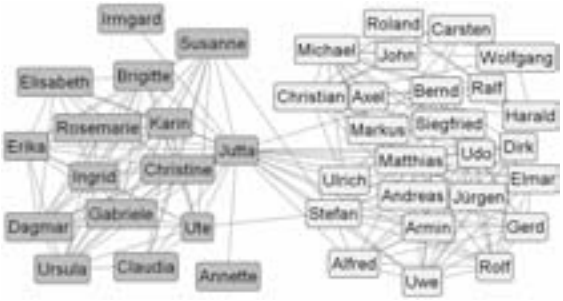


Abbildung 3: Ausschnitt aus dem Graph häufiger Wörter mit Unterscheidung zwischen Vornamen verschiedenen Geschlechts als Wortarten. Graustufen geben die Klassen des mit CW geclusterten gesamten Graphen wieder.

Lernen von Sprachannotationen in [BGG07] verdeutlicht, dass die hier vorgestellte Wortartenmarkierungsmethode in den meisten Anwendungen den gleichen Beitrag leistet wie ein traditionelles System. Somit wird der manuelle Erstellungsaufwand für neue Sprachen und Textsorten überflüssig, wenn nur ausreichend Rohtext zur Verfügung steht. Mit einer Open-Source Implementierung wurden Wortartenmarkierer für 14 Sprachen vorberechnet[7]. Das Zuweisen von Wortarten klärt grammatische Mehrdeutigkeiten, lässt jedoch verschiedene Bedeutungen von Wörtern unberührt. Dies ist Thema des folgenden Abschnitts.

2.5.3 Finden von Wortbedeutungen

Auch für das Finden von verschiedenen Bedeutungen eines Wortes, z.B. 'Bank' als Geldinstitut oder Sitzmöbel, erweist sich ein Clustering des Kookkurrenzgraphen als nützlich. Wörter, die signifikant mit *einer* Bedeutung des Zielwortes auftreten (z.B. 'Park' mit 'sitzen'), kommen signifikant häufiger miteinander vor als Wörter, die mit verschiedenen

[6]Datenbasis: 10 Millionen Sätze aus dem Projekt Deutscher Wortschatz, siehe http://www.wortschatz.uni-leipzig.de

[7]Software und Modelle auf http://wortschatz.uni-leipzig.de/~cbiemann/software/unsupos.html

Bedeutungen assoziiert sind (z.B. ‘Park’ und ‘Geld’). Dies nutzt ein Wortbedeutungsinduktionsverfahren aus, welches zunächst die Nachbarschaft eines Zielwortes aus dem Kookkurrenzgraphen extrahiert und diese dann mit CW clustert. Eine pseudowortbasierte Evaluation zeigt vergleichbare Präzision zu einem dezidiert für Wortbedeutungsinduktion entwickelten Verfahren [Bor07], jedoch eine höhere Abdeckung, siehe auch [Bie06a].

3 Zusammenfassung und kritische Würdigung

Die hier zusammengefasste Arbeit markiert einen Paradigmenwechsel in der Automatischen Sprachverarbeitung: Maschinen wird nicht mehr explizit oder implizit beigebracht, Sprache zu strukturieren, sondern sie werden vielmehr mit der Möglichkeit ausgestattet, diese Strukturen selbst zu entdecken.
Die Repräsentation der Daten wird intuitiv und einheitlich mit Graphen realisiert. Die Analyse der Makrostruktur der aus Sprachdaten erstellten Graphen führt zur Entwicklung eines Zufallstextmodelles, welches quantitative Eigenschaften natürlicher Sprache besser approximiert als vorangegangene Modelle. Bestechend durch seine Einfachheit, leistet dieses Modell einen Beitrag zur Erklärung von Sprachevolution.
Ein Graphclusteringverfahren minimaler Berechnungskomplexität wird entwickelt, welches die für die Strukturentdeckung nötigen Abstraktions- und Generalisierungsprozesse realisiert. Der Basisalgorithmus, welcher nichtdeterministisch Graphen partitioniert, wird in Richtung Quasideterminismus, Fuzzy Clustering und flach hierarchisches Clustering erweitert. Im praktischen Teil der Arbeit wird das Structure Discovery Paradigma anhand dreier Aufgaben aus der ASV illustriert: Identifikation verschiedener Sprachen, Zuweisung von Wortarten und dem Finden von Wortbedeutungen. Die für die Ergebnisse dieser Arbeit relevante Software steht weitgehend als Open-Source Software auf der Homepage des Autors zu Verfügung.
Als Gesamtergebnis lässt sich festhalten, dass unüberwachte und vorwissensfreie Automatische Sprachverarbeitung im Structure Discovery Paradigma erfolgreich in der Lage ist, Systeme hervorzubringen, welche den mit traditionellen Methoden erstellten nichts nachstehen. Im Unterschied zu diesen benötigen Structure Discovery Prozesse jedoch keine manuell annotierten Trainingsdaten oder elaborierte Regelsysteme, sondern lediglich eine große Menge von Rohtexten. Deshalb stellt Structure Discovery nicht nur eine Alternative für ressourcenarme Sprachen dar, sondern ermöglicht auch die schnellere Entwicklung neuer Anwendungen.

Literatur

[Bar03] Albert-László Barabási. *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life*. Plume Books, 2003.

[BGG07] Chris Biemann, Claudio Giuliano und Alfio Gliozzo. Unsupervised Part-of-Speech Tagging Supporting Supervised Methods. In *Proceedings of Recent Advances in Natural Language Processing (RANLP-07)*, Borovets, Bulgaria, 2007.

[Bie06a] Chris Biemann. Chinese Whispers – an Efficient Graph Clustering Algorithm and its Application to Natural Language Processing Problems. In *Proceedings of TextGraphs: the Second Workshop on Graph Based Methods for Natural Language Processing*, Seiten 73–80, New York City, 2006.

[Bie06b] Chris Biemann. Unsupervised Part-of-Speech Tagging Employing Efficient Graph Clustering. In *Proceedings of the COLING/ACL-06 Student Research Workshop*, Sydney, Australia, 2006.

[Bie07a] Chris Biemann. A Random Text Model for the Generation of Statistical Language Invariants. In *Human Language Technologies 2007: Proceedings of the Main Conference (HLT-NAACL-07)*, Seiten 105–112, Rochester, New York, 2007.

[Bie07b] Chris Biemann. *Unsupervised and Knowledge-free Natural Language Processing in the Structure Discovery Paradigm*. Dissetation, Universität Leipzig, Deutschland, November 2007.

[Bor07] Stefan Bordag. *Elements of Knowledge-free and Unsupervised Lexical Acquisition*. Dissertation, Universität Leipzig, Deutschland, 2007.

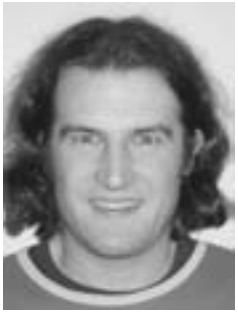
[BT05] Chris Biemann und Sven Teresniak. Disentangling from Babylonian Confusion – Unsupervised Language Identification. In *Proceedings of Computational Linguistics and Intelligent Text Processing, 6th International Conference (CICLing-05)*, Springer LNCS, Seiten 773–784, Mexico D.F., Mexico, 2005.

[Qua98] Uwe Quasthoff. Deutscher Wortschatz im Internet. *LDV Forum*, 15(2):4–23, 1998.

[Sch93] Hinrich Schütze. Word space. In S. Hanson, J. Cowan und C. Giles, Hrsg., *Advances in Neural Information Processing Systems 5*. Morgan Kaufmann Publishers, 1993.

[ST05] Mark Steyvers und Joshua B. Tenenbaum. The Large-Scale Structure of Semantic Networks: Statistical Analyses and a Model of Semantic Growth. *Cognitive Science*, 29(1):41–78, 2005.

[Zip49] G. K. Zipf. *Human Behavior and the Principle of Least-Effort*. Addison-Wesley, Cambridge, MA, 1949.



Chris Biemann studierte Informatik an der Unitersität Leipzig und der Vrijen Universiteit Amsterdam. Nach dem Abschluss mit Diplom in linguistischer Informatik im Jahre 2003 war er am Lehrstuhl für Automatische Sprachverarbeitung in Leipzig tätig und absolvierte Forschungsaufenthalte in Südkorea und Norwegen. Seine im November 2007 angenommene Dissertation bereitete er im Rahmen des Graduiertenkollegs Wissensrepräsentation an der Universität Leipzig vor und war gleichzeitig Mitarbeiter im Projekt Deutscher Wortschatz. Chris Biemann ist Mitglied zahlreicher internationaler Programmkommittees und begutachtet Beiträge für international rennommierte Fachzeitschriften. Er organsierte mehrere Workshops im Rahmen von internationalen Top-Konferenzen der Bereiche Computerlinguistik und Maschinelles Lernen. Seit Januar 2008 arbeitet er als Research Scientist in San Francisco bei Powerset, einer semantischen Suchmaschine.