Zuerst ersch. in: Journal for Language Technology and Computational Linguistics ; 26 (2011), 1. - pp. 67-79

Heike Zinsmeister

Chancen und Probleme der Nutzung von Korpora, Taggern und anderen Sprachressourcen in Seminaren

1 Einleitung

Mit Korpora oder lexikalisch-semantischen Ressourcen zu arbeiten und dabei Programme zur Aufbereitung oder Analyse der Daten zu nutzen, gehört zum Alltag vieler Computerlinguisten. Computerlinguistische Studiengänge sollten daher ihren Studierenden nicht nur Wissen über Theorien und Algorithmen vermitteln – und eigene Programmierkenntnisse — sondern sie auch auf den Umgang mit vorhandenen Sprachressourcen vorbereiten. Hierbei sind nicht Ressourcen für das *E-Learning* gemeint, sondern Sprachressourcen, die unabhängig von einer Verwendung in der Lehre entwickelt wurden.

Beispiele für solche Sprachressourcen sind:

- Korpora wie das wortartengetaggte und diachron aufbereitete Kernkorpus des DWDS[1] oder die syntaktisch und koreferenzannotierte Tübinger Baumbank Deutsch / Zeitungssprache (TüBa-D/Z)[2].
- Lexikalisch-semantische Ressourcen wie GermaNet[3] oder FrameNet[4].
- Tagsets und Annotationsrichtlinien wie das Stuttgart-Tübingen-Tagset STTS (Schiller et al., 1999) für die Annotation von Wortarten im Deutschen oder die MATE-Annotationsrichtlinien zur Annotation von Koreferenz (Poesio, 2000).
- Programme für die manuelle Annotation wie EXMARaLDA ('EXtensible MARkup Language for Discourse Annotation', (Schmidt, 2004)) für die Mehrebenenannotation von multimodalen Korpora oder MMAX2 ('Multi-Modal Annotation in XML', (Müller and Strube, 2006)).
- Programme zur automatischen Annotation wie der TreeTagger (Schmid, 1997) für Wortartenannotation und syntaktisches Chunking oder der Stanfordparser (Rafferty and Manning, 2008) für syntaktische Konstituenten- und Dependenzannotation.
- Programme zu Indexierung, Recherche und Visualisierung wie die Corpus Workbench mit dem mächtigen Abfrageprogramm Corpus Query Processor CQP[5],

[1] DWDS-Korpora: `http://www.dwds.de/` [Letzter Aufruf der zitierten URLs: 23.03.2011].
[2] TüBa-D/Z: `http://www.sfs.uni-tuebingen.de/tuebadz.shtml`.
[3] GermaNet: `http://www.sfs.uni-tuebingen.de/GermaNet/`.
[4] FrameNet: `http://framenet.icsi.berkeley.edu/`.
[5] CQP: `http://cwb.sourceforge.net/index.php`.

ANNIS2[6] für die Abfrage und Darstellung von mehrebenenannotierten und multimodalen Daten oder TigerSearch[7] für die Abfrage syntaktisch annotierter Korpora.

In der Praxis sind Lehrende beim aktiven Einsatz von Sprachressourcen in Seminaren mit besonderen Anforderungen konfrontiert, die die Durchführung oftmals behindern oder sogar zum Scheitern verurteilen. Der vorliegende Beitrag stellt eine Art Checkliste für die Vorbereitung des Einsatzes von Ressourcen zusammen und diskutiert Probleme, die in Bezug auf einzelne Kriterien der Checkliste auftreten können. Abschließend werden aus dieser Diskussion Wünsche abgeleitet, wie Sprachressourcen bereitgestellt werden sollten, so dass sie in der Lehre leichter einsetzbar werden.

2 Didaktische Überlegungen

2.1 Zielgruppen

Stimmt es wirklich, dass der Einsatz von Sprachressourcen in der Lehre andere Anforderungen mit sich bringt, als die Verwendung von Sprachressourcen in Forschungsprojekten? In einem Szenario, in dem die Studierenden einen fundierten computationellen Hintergrund haben, ist der Unterschied vielleicht nicht groß. Diese Situation ist aber oft nicht gegeben, selbst in grundständigen Studiengängen der Computerlinguistik. Hinzu kommt, dass Seminare mit computerlinguistischen Inhalten nicht nur in den ausgewiesenen Studiengängen vertreten sind, sondern Bestandteil vieler sprachwissenschaftlicher und texttechnologischer Studiengänge sind. Zum Beispiel bietet der Bachelorstudiengang 'Sprachwissenschaft' an der Universität Konstanz[8] in Modul 4 unter 'Weiterführende Gebiete der Linguistik' neben Spracherwerb, Typologie und anderen Gebieten auch ein Seminar 'Computerlinguistik' an. Das Seminar führt Grundideen der Computerlinguistik ein und skizziert die Funktionsweisen von Applikationen wie Wortarten-Tagging, Maschineller Übersetzung, automatischer Grammatik- und Rechtschreibprüfung oder automatischer Textzusammenfassung. Die Studierenden erhalten dabei einen ersten Einblick in die Möglichkeiten (und Grenzen) der maschinellen Sprachverarbeitung. Interessierte Studierende können Computerlinguistik im Masterprogramm 'Speech and Language Processing' vertiefen, in dem Computerlinguistik einen der beiden möglichen Schwerpunkte bildet. Dort ist auch vorgesehen, dass die Studierende eine Programmiersprache erlernen. Dies gilt nicht für das Bachelorangebot.

2.2 Einsatzszenarien

Es sind drei allgemeine Szenarien für den Einsatz von Sprachressourcen in der Lehre denkbar[9]: 'Im Rampenlicht', 'Hinter den Kulissen' und als 'Schattenspiel'. Im er-

[6] ANNIS2: `http://www.sfb632.uni-potsdam.de/d1/annis/`.
[7] TigerSearch: `http://www.wolfganglezius.de/doku.php?id=cl:tigersearch`.
[8] url: `http://ling.uni-konstanz.de/pages/allgemein/bachelor.html`.
[9] Die hier verwendete Bühnen-Metaphorik geht auf Guy Aston zurück, der für den Einsatz von Korpora in der Fremdsprachendidaktik eine 'on stage'-Nutzung von einer 'behind the scene'-Nutzung unterscheidet (vgl. Aston (2000)).

sten Szenario stehen die Ressourcen dahingehend 'im Rampenlicht', dass sie von den Studierenden selbst aktiv im Seminar oder in Übungen genutzt werden. 'Hinter den Kulissen' werden Ressourcen verwendet, wenn die Lehrenden sie für die Erstellung ihrer Lehrmaterialien verwenden, die Nutzung selbst aber in der Lehre nicht thematisieren. Ein Beispiel wäre, die TüBa-D/Z auszuwerten, um interessante Beispiele für eine Seminarstunde zu indirekten Anaphern ('Bridging') zu finden. Ein anderes Beispiel wäre, den TreeTagger auf unterschiedlichen Trainingsdaten zu trainieren, um realistische Zahlen für eine Übungsaufgabe zur Evaluierung von Wortartentaggern zu erhalten. Das dritte Szenario ähnelt einem 'Schattenspiel'. Die Ressourcen werden im Unterricht zwar thematisiert und, zum Beispiel, auf Folien im Unterrichtsvortrag vorgestellt, aber von den Studierenden selbst nicht aktiv genutzt – auch nicht in seminarbegleitenden Übungen. Die Studierenden erhalten dabei ein rein passives Wissen über die Ressourcen.

Untersuchungen zur Lernpsychologie zeigen, dass aktives Lernen den Lernprozess am Besten unterstützt (vgl. z.B. Winteler (2004), Kapitel 10). Umgangssprachlich ausgedrückt, man lernt etwas am Besten, indem man es *tut*. 'Begreifen' hat sehr viel mit 'Greifen' d.h. mit aktivem Handeln zu tun. Aus didaktischer Sicht ist die aktive Verwendung von Sprachressourcen im 'Rampenlicht' daher einer passiven Vorstellung in 'Schattenspielen' vorzuziehen.

2.3 Lernziele

Der Einsatz von Sprachressourcen in der Lehre kann zweierlei Lernzielen dienen. Zum einen kann das Ziel eine *Methoden- bzw. Ressourcenkompetenz* sein, die die Lernenden in die Lage versetzt, die verwendeten Ressourcen selbstständig für eigene Zwecke einzusetzen. Zum Beispiel, mit welchen Kommandozeilen-Befehlen der TreeTagger auf einem Korpus trainiert werden kann, oder mit welcher Maus- und Tastenkombination in MMAX2 eine Koreferenzrelation zwischen zwei 'Markables' markiert wird bzw. welches linguistische Phänomen in der TüBa-D/Z als 'Bridging' annotiert ist. Zum anderen kann die Ressource nur als *Mittel zum Zweck* eingesetzt werden, wenn das eigentliche Lernziel das Verstehen einer (computer-)linguistischen Fragestellung ist. Zum Beispiel kann der TreeTagger in einem Szenario eingesetzt werden, in dem das eigentliche Lernziel die Evaluierung von statistischen Programmen ist, und die Studierenden Kompetenzen im Umgang mit Akkuratheitsbestimmung, Kreuzvalidierung und Konfusionsmatrizen erwerben sollen. Wenn das eigentliche Lernziel Koreferenz und Anaphorik im Deutschen ist, können Studierende in den anaphorischen Annotationen der TüBa-D/Z interessante Beispiele mit Kontext finden und mit dem Programm MMAX2 das Phänomen mit Hilfe von manueller Annotation kennenlernen (vgl. Zinsmeister (2011)).

In der realen Umsetzung findet man Mischformen der beiden Lernziele. Das scheint auch sinnvoll zu sein, weil die eigene Erfahrung zeigt, dass die Studierenden praktische Hürden besser tolerieren, wenn das Lernziel ausdrücklich (auch) die Ressourcenkompetenz selbst ist. Soll die Ressource nur Mittel zum Zweck sein, wird eher ein reibungsloser Ablauf erwartet und nur wenig Aufwand für das Erlernen der Handhabung akzeptiert.

3 Kriterien für den Einsatz von Sprachressourcen 'im Rampenlicht'

Dieser Abschnitt stellt eine Liste von Kriterien für die Nutzung von Sprachressourcen in der Lehre vor. Die Zusammenstellung ist als eine Art Checkliste aufgebaut. Mögliche Ausprägungen der Kriterien werden in Fragen formuliert. Die Kriterien ergeben sich aus der Art des Lernziels und (impliziten) Modellen der Rahmenbedingungen, des Nutzers und der Ressource selbst. Das Ressourcen-Modell beschreibt externe Merkmale der Ressource sowie deren Eigenschaften in konkreten Verwendungskontexten.

Die Auswahl der Kriterien orientiert sich an Evaluationskriterien für sprachverarbeitende Software (vgl. EAGLES (1996), basierend auf dem allgemeinen Qualitätsmodell für Software ISO 9126)[10] und ist um Kriterien für Korpusressourcen und für die Implementierung im Lehrkontext erweitert. Ein Anspruch auf Vollständigkeit besteht nicht.

3.1 Checkliste

Lernziel

- Ist die Kenntnis der Ressource und ihrer Nutzung das Lernziel oder nur ein Mittel zum Zweck?

Rahmenbedingungen

- Zeit: Wie viel Zeit steht für die Einarbeitung und Nutzung der Ressource sowie die Auswertung der Ergebnisse zur Verfügung?
  Soll die Ressource nur innerhalb einer Seminarsitzung genutzt werden, seminarbegleitend (während mehrerer Sitzungen) oder zusätzlich zur Lehrveranstaltung in Seminar- oder Studienarbeiten?

- Ort: An welchem Ort soll die Ressource genutzt werden?
  Im Universitäts-Pool, auf privaten Rechnern an der Universität oder auf privaten Rechnern zuhause?

- Hardware: Welche Hardware steht zur Verfügung?
  Eine homogene Plattform auf Poolrechnern oder eine Vielzahl von Betriebssystemen auf den Privatrechnern der Studierenden?

- Sozialform: In welcher sozialen Form soll die Ressource genutzt werden?
  In Einzelarbeit oder in Gruppenarbeit?

[10] Vielen Dank an Stefanie Dipper, die mich auf diese Qualitätskriterien hinwies, siehe auch Dipper et al. (2004).

Nutzer

- Motivation: Kennen die Studierenden Forschungsfragen, die einen Einsatz der Ressource motivieren? Sind die Studierenden durch eigene (Forschungs-)Interessen motiviert, die Ressource kennenzulernen? Wissen die Studierenden um die Berufsrelevanz der Ressource?
- Technisches Know-How: Sind die Studierenden mit der zu nutzenden Hardware vertraut? Sind die Studierenden bereits mit der Ressource selbst vertraut? Sind die Studierenden mit ähnliche Ressourcen vertraut? Besitzen die Studierenden Programmierkenntnisse (z.B. um die Ressource zu installieren und zu bedienen)?

Verfügbarkeit

- Bereitstellung: Kann man die Ressource online nutzen? Kann die Ressource heruntergeladen werden? Ist die Prozedur, wie man die Ressource erhalten kann transparent?
- Lizenzierung: Muss die Ressource lizenziert werden? Gibt es Gruppen- oder nur Einzellizenzen?
- Kosten: Ist die Ressource für die Lehre kostenfrei?
- Externer Support: Erhalten die Nutzer der Ressource Unterstützung von den Entwicklern oder anderen Experten?

Funktionalität

- Tauglichkeit /Angemessenheit: Bietet die Ressource für das Lernziel relevante Merkmale oder Inhalte bzw. erzeugt sie relevante Analysen?
- Vertrautheit: Entspricht die Ressource in ihrer Funktionalität bekannten Standards oder Konventionen (z.B., dass Copy&Paste mit bekannten Tastenkombinationen möglich ist oder dass bei der Darstellung von Dependenzannotationen die Pfeilspitze einer Kante auf das jeweilige Regens zeigt).

Verwendbarkeit

- Verständlichkeit: Ist der Aufbau der Ressource intuitiv und leicht verständlich? Ist die Ressource gut dokumentiert? Sind ggfs. Eingabe- und Ausgabeformate bekannt?
- Lernbarkeit: Kann die Verwendung der Ressource leicht erlernt werden? Gibt es eine Lernanleitung?
- Operabilität: Ist die Ressource leicht zu handhaben? Zum Beispiel, gibt es intuitive Hilfefunktionen?
- Attraktivität: Macht es Spaß die Ressource zu nutzen?

Effizienz

- Zeit: Wie lang dauert die Nutzung / Ausführung des Programms (z.B. Dauer der Auswertung einzelner Korpussuchanfragen oder Trainingszeiten für ein statistisches Übersetzungsmodell).
- Ressourcennutzung: Beansprucht die Ressource große Rechnerkapazitäten? Benötigt sie Zugriff auf andere Ressourcen?

Portabilität

- Adaptierbarkeit: Kann die Ressource auf verschiedenen Plattformen genutzt werden? Benötigt die Ressource spezifische Systemvoraussetzungen bzw. vorhandene Software (z.B. bestimmte Libraries)?
- Installierbarkeit: Ist die Ressource leicht zu installieren, so dass auch Nutzer ohne Programmierkenntnisse eine Installation durchführen können?
- Ko-Existenz: Kann die Ressource mit anderen Ressourcen gemeinsam genutzt werden oder kann es zu Störungen kommen?

3.2 Diskussion der Anforderungen

Installierbarkeit und **Portabilität** können vernachlässigt werden, wenn die Ressource nur auf Poolrechnern genutzt werden soll und das technische Know-How der Lehrenden oder des lokalen technischen Supports ausreichend ist. Wird erwartet, dass die Studierenden die Ressource auch auf eigenen Rechnern verwenden, werden die beiden Eigenschaften zu entscheidenden Kriterien. Als ideale Lösung bietet sich hier eine **Online-Nutzung** von Ressourcen an. Doch auch diese kann zu Problemen in der Umsetzung führen. Bei WLAN-Nutzung, zum Beispiel, kann eine zu große Gruppe die lokalen Kapazitäten überfordern. Auch auf der Ressourcenseite kann es zu Engpässen kommen, auf die man als externer Nutzer keinen Einfluss hat, wie vorübergehende Ausfälle wegen Wartungsarbeiten. Ein anderes Problem mit reinen Onlineressourcen besteht dann, wenn die Ressource ein sprachverarbeitendes Programm ist und die Studierenden 'private' Sprachdaten, z.B. personenbezogene Texte wie E-Mails oder Blogs, weiterverarbeiten. In diesem Fall ist es aus Datenschutzgründen besser, das verarbeitende Programm ist lokal installiert.

Aus Progammierersicht ist es effizient, wenn ein Programm auf bereits vorhandene Programme und Module zurückzugreift. Dies macht die Installation von Ressourcen jedoch umständlicher und auch schwieriger, weil dann ggf. externe Ressourcen mitinstalliert werden müssen, die eventuell nicht gut dokumentiert sind oder bei denen sich die Verwendung auf verschiedenen Betriebssystemen unterscheidet. Für den Einsatz in der Lehre sind 'Download and Run'-Ressourcen, bei denen die Ressource als nutzungsfähiges **Gesamtpaket** bezogen werden kann, einem flexibel kombinierbaren Modul vorzuziehen.

Die Nutzung eines **Pools** hat den Vorteil, dass die Hardware potenziell homogen ist und die Installation von den Lehrenden durchgeführt werden kann. Klare Nachteile sind die eingeschränkte zeitliche Verfügbarkeit eines Pools und dass es immer Unterschiede zwischen den Privatrechnern der Studierenden und den Poolrechnern geben wird (Betriebssystem, Tastatur, Hilfsprogramme, usw.).

Der Spaßfaktor bei der Nutzung einer Ressource als Ausdruck ihrer **Attraktivität** ist nicht zu unterschätzen. Wann macht es Spaß, mit einer Ressource umzugehen? Eine graphische Oberfläche spielt sicher eine große Rolle. Wenn die Ressource, zum Beispiel, ein Korpus mit Abfrageprogramm ist, dann macht es mehr 'Spaß' damit zu arbeiten, wenn die Suchergebnisse in einer ansprechenden Form optisch dargestellt werden. In TigerSearch kann man, zum Beispiel, einstellen, ob man nur den übereinstimmenden Teilbaum angezeigt haben möchte oder den ganzen Satzgraphen – oder den Satzgraphen mit dem Teilbaum farblich hervorgehoben (in einer Wunschfarbe). Zusätzlich kann der Nutzer wählen, ob und wie viele Kontextsätze angezeigt werden sollen (bis zu drei Kontextsätze).

Indirekt entsteht auch Spaß, wenn der Aufbau einer Ressource oder ihre Handhabung vertraut ist. Der Spaß wird einem vergällt, wenn Erwartungen an einen Inhalt immer wieder enttäuscht werden oder wenn ansonsten automatische Handhabungen ins Leere laufen. Hier spielen manchmal Kleinigkeiten eine große Rolle, zum Beispiel, ob es eine Copy&Paste-Funktionalität gibt, die mit 'normalen' Tastenkombinationen zu bedienen ist. Eine Ressource, die allgemeinen **Konventionen** entspricht, ist leichter zu nutzen als eine sehr individuell gestaltete Ressource, und ist daher in der Lehre vorzuziehen.

Spaß entsteht aber nicht nur, wenn alles bekannt ist und reibungslos abläuft. Im Gegenteil, Spaß entsteht gerade auch dann, wenn ein Prozess des Erkenntnisgewinns stattfindet – wenn man etwas Neues entdeckt oder ein Problem lösen kann, insbesondere dann, wenn am Ende ein 'greifbares' Resultat entsteht.[11]

Für den Einsatz von Sprachressourcen in der Lehre bietet sich als soziale Form die **Gruppenarbeit** an. Probleme können so leichter behoben werden. Vorwissen und Fähigkeiten der Gruppenmitglieder können sich ergänzen. Selbst das gemeinsame Lästern über die Aufgabe oder Probleme bei der Umsetzung tragen zum 'Spaß' bei und können sich indirekt positiv auf das Lernergebnis auswirken.[12]

Ein sehr wichtiges Entscheidungskriterium für die Nutzung einer Ressource in der Lehre ist das Vorhandensein von **Dokumentationen**. Eine gute Dokumentation ist mehrteilig und erfüllt verschiedene Anforderungen: eine allgemeine Beschreibung der Ressource, problem-spezifische Hilfestellung, Beispiele (Beispieldateien bei Programmen oder Suchanfragen und Annotationsbeispiele bei Korpora) und eine detaillierte Nutzungsanleitungen (ein Schritt-für-Schritt-Tutorium oder Annotationsrichtlinien), vgl. auch Dipper et al. (2004).

[11] Dies entspricht dem Ansatz des Forschenden Lernens (vgl. Huber (2004), S. 33, nach Reiber (2007), S. 10).

[12] Das gemeinsame Lästern weckt zumindest passive Teilnehmende auf und löst Emotionen aus, die wiederum grundsätzlich lernfördernd sind. Idealerweise rückt es auch die Handhabung der Ressource ins Zentrum der Aufmerksamkeit.

Die Problem-orientierte Dokumentation kann eine einfache Auflistung im Sinne von häufig gestellten Fragen (FAQs) sein. Bei Korpusressourcen entspricht dies auch der Diskussion von schwierigen Annotationsentscheidungen. Diese Art der Informationspräsentation ist besonders zugänglich für weniger geübte Nutzer, die in einem fortlaufenden Erklärungstext Mühe haben, die Informationen zu finden, die für ihre Fragestellungen relevant sind. Daher ist die **FAQ-Liste** ein wichtiger Bestandteil für denn Einsatz von Sprachressourcen in der Lehre.

Die Bereitstellung von konkreten **Anwendungsbeispielen** erspart den Lehrenden viel Vorbereitungszeit. Dies schließt eine genaue Beschreibung des Formats der Eingabe- und Ausgabedateien von Programmen ein aber auch des Datenformats von Korpusressourcen. Im Idealfall entsteht eine Sammlung von Unterrichtsentwürfen und Sammlungen von Übungsaufgaben, in denen die Ressourcen zum Einsatz kommen. Neben der eigentlichen Dokumentation sind vollständige Beispieldateien sehr hilfreich, mit denen ein Programm unmittelbar getestet werden kann. Im Unterricht können sie als Editiergrundlage für die weitere Arbeit der Studierenden dienen. Die Beispielanwendung dokumentiert idealerweise vollständige Befehle, beim Einsatz von GUIs auch in bildlicher Form (durch Screenshots oder Videotutorien).

Für den Einsatz in der Lehre ist es ebenfalls sehr hilfreich, wenn die Dokumentation auf **Veröffentlichungen** zur Ressource verweist, ebenso auf solche Arbeiten, bei denen die Ressource in irgendeiner Form zum Einsatz kam. In jedem Fall sollte eine einschlägige Referenz angegeben sein, mit welcher die Ressource zitiert werden kann. Im Unterrichtszenario ist dies vor allem dann relevant, wenn das Programm von den Studierenden selbstständig genutzt werden soll z.B. in Abschlussarbeiten.

Die Dokumentation von annotierten Korpora oder lexikalischen Ressourcen beschreibt zusätzlich zu einer möglichen Nutzungsbeschreibung den Inhalt und ggfs. den Entstehungsprozess der Ressource. Bei einem annotierten Korpus kann dies durch die Veröffentlichung der **Annotationsrichtlinien** ('Guidelines') geschehen. Annotationsrichtlinien beinhalten Definitionen der analysierten linguistischen Phänomene, Klassifikationen der Phänomene in abgrenzbare Untertypen und eine exhaustive, eindeutige Zuordnung von Klassen zu Annotationsetiketten (Labeln bzw. 'Tags').

Ein einfaches Annotationsbeispiel sind die verschiedenen Lesarten von *es* (adaptiert von Boyd et al. (2005) und Naumann (2006)). Jede Lesart bildet eine Klasse und wird mit einem spezifischen Etikett versehen:

1. *Nominale Anapher:*
   Das Baby liegt in der Wiege. Es schläft ruhig.
2. *Abstrakte Anapher:*
   Die Benzinpreise steigen wieder. Es ist unglaublich.
3. *Korrelat:*
   Es ist gut, dass Peter kommen konnte.
4. *Wetterverb / Prädikativ der Zeit, des Orts, etc.:*
   ... weil es regnete / ... weil es schon drei Uhr war.
5. *Vorfeld-Es:*
   Es wurde bis zum Morgen getanzt.

Die Annotationsrichtlinien legen auch fest, welche sprachlichen Einheiten überhaupt mit Etiketten markiert werden dürfen ('Markables'). Im Fall der Annotation von *es* als nominale Anapher könnte festgelegt werden, dass nur Nominalphrasen als markierbare Einheiten für die Antezedenten gelten (aber keine Sätze oder ähnliches).

Ein wichtiger Bestandteil von Annotationsrichtlinien sind die Umsetzungskriterien (die Operationalisierungen), wann ein bestimmtes Markable mit einer bestimmten Etikette versehen werden darf. Je konkreter diese Operationalisierungen sind, desto besser können die Nutzer die vorliegenden Annotationen der Daten nachvollziehen. Eine robuste Form der Umsetzung besteht darin, linguistische Tests für das Phänomen zu spezifizieren, z.B. wird im Entscheidungsbaum in Abb. 1 ein Paraphrasierungstest mit *nämlich* vorgeschlagen, um das Antezedenz einer Anapher zu identifizieren[13].

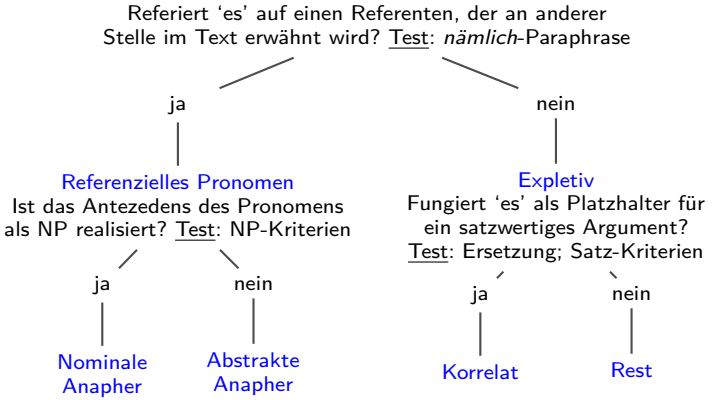

Abbildung 1: Annotationsrichtlinien: Entscheidungsbaum linguistischer Tests.

Neben Definitionen und Tests enthalten die Richtlinien konkrete Annotationsbeispiele: unkontroverse Fälle, die die Analyse veranschaulichen, und Problemfälle, anhand derer die Grenzen der Analyse klar gemacht werden können. Die Richtlinien sollten auch die Annotationsentscheidungen der Problemfälle dokumentieren, so dass die Nutzer nachvollziehen können, welche konkreten Analysen sie in den Daten erwarten.[14]

Sowohl bei Programmen, mit denen bestehende Ressourcen ausgewertet werden, als auch bei solchen, die unmittelbar Daten manipulieren, sollte eine Anleitung vorhanden sein, die angibt, wie das **Analyseergebnis** gesichtet und ggfs. weiterverarbeitet werden kann. Zum Beispiel, wie bestimmte automatische Analyseergebnisse anhand eines Goldstandards evaluiert werden können oder wie die manuelle Annotation von mehreren

[13] Der Paraphrasierungstest zu Beispiel (1): *Es, nämlich das Baby, schläft ruhig.*

[14] Wie man den Annotationsvorgang als didaktisches Mittel in der Lehre einsetzen kann, wird in Zinsmeister (2011) diskutiert.

Annotatoren verglichen werden kann. Bei Korpusressourcen und anderen Daten benötigt man Angaben zu Programmen für die Suche und Visualisierung. Manche Ressourcen stellen APIs für die weitere Prozessierung der (Ausgabe-)Daten zur Verfügung z.B. für eine Konversion in andere Dateiformate. Hier gelten bereits genannte Argumente: Bei Ressourcennutzung auf Privatrechnern sind viele Studierende mit der Verwendung von APIs überfordert. Auch bei einer Poolnutzung ist ein Gesamtpaket leichter zu handhaben.

Bei der Entscheidung, eine bestimmte Ressource in der eigenen Lehre einzusetzen, spielt es eine Rolle, ob man die Ressource auch in der eigenen Forschung verwendet. Es fällt ebenfalls leichter, Ressourcen zu nutzen, die in anderen lokalen Projekten entwickelt oder verwendet werden. Die Wahl zwischen zwei alternativen Ressourcen kann im Zweifelsfall entscheiden, ob man die Entwickler einer Ressource persönlich kennt. Dies alles deutet darauf hin, dass Lehrende Zugriff auf persönliches **Expertenwissen** zu einer Ressource suchen, konkrete Personen, die direkt ansprechbar sind.[15]

In der Lehre ist es ganz entscheidend, die **Motivation**, warum eine bestimmte Ressource genutzt werden soll, zu kommunizieren – was leider immer wieder vernachlässigt wird. Den Studierenden soll das Lernziel bewusst sein und welche Rolle die Ressource in Bezug darauf spielt. Auch wenn das unmittelbare Lernziel eine reine Methoden- oder Ressourcenkompetenz ist, sollten die Studierenden wissen, für welche weiterführenden Forschungsfragen oder Anwendungen die Kompetenz relevant ist. Darüber hinaus sollte man klarstellen, in wie weit die Ressource mittelfristig für das Studium relevant ist, zum Beispiel in anderen Seminaren oder für die studentische Abschlussarbeit. Schlussendlich sollte auch die langfristige Relevanz überprüft werden. Ist das Wissen um die Ressource auch auf Gebiete außerhalb des Studiums übertragbar? Hat es Berufsrelevanz – innerhalb und außerhalb der Forschung?

4 Wunschliste

Aus der Diskussion der Anforderungen oben leiten sich eine Reihe von Wünschen ab, wie man die Bereitstellung von Sprachressourcen für einen Einsatz in der Lehre optimieren könnte. Die Wünsche betreffen nicht den Aufbau der Ressourcen (außer beim Unterpunkt ‘Portabilität‘), sondern ihre Nutzbarmachung für die Lehre.

Bereitstellung Es gibt eine zentrale Webseite mit Links auf Sprachressourcen, die auch als zentrale Informationsstelle dient. Sie bietet Anleitungen im Sinne von Dokumentationen und Expertenrat (eine Art ‘Hotline’). Neben den Ressourcen selbst sammelt sie Forschungsfragen und zeigt Anwendungsbeispiele auf.

[15] In der Praxis wenden sich Nutzer eher an lokale IT-Spezialisten, die selbst oftmals keine Computerlinguisten sind, als an die Entwickler der Ressourcen selbst, vgl. die Diskussion beim D-Spin-Workshop zu eHumanities und Sprachressourcen am 17.01.2011 an der Berlin-Brandenburgischen Akademie in Berlin.

Verfügbarkeit Lehrrelevante Sprachressourcen sind online-nutzbar und stehen auch für einen Download zur Verfügung. Lizenzen werden als Gruppenlizenz für einen ganzen Lehrstuhl vergeben. Die Ressourcen sind für die Lehre kostenfrei. Ein Expertennetzwerk steht für externen Support zur Verfügung.

Verwendbarkeit Die Ressource ist gut dokumentiert. Es gibt eine ausführliche Beschreibung, problem-spezifische Hilfestellung im Sinne von FAQs, Beispieldateien bzw. Annotationsbeispielen sowie eine detaillierte Nutzungsanleitung. Weitere Meta-Dokumentationen benennen Veröffentlichungen zur Ressource sowie Veröffentlichungen, zu der die Ressource beigetragen hat. Zusammen mit der Ressource werden weiterführende Programme zur Verfügung gestellt (siehe Meta-Ressourcen unten).

Portabilität Die Ressource ist leicht zu installieren. Sie wird als plattformunabhängiges Gesamtpaket angeboten.

Dokumentation Wie die Ressourcen selbst sind die Dokumentationen sowohl online zugänglich und damit von den Studierenden immer abrufbar, als auch in der Form eines Dokuments herunterladbar (z.B. als ein pdf-Dokument mit aktiven Links), so dass die Studierenden sie auch lokal speichern und offline nutzen können.

Meta-Ressourcen Bei Programmen werden Konvertierungsprogramme für das geforderte Eingabeformat bereitgestellt, zum Beispiel eine einfache Umformatierung in ein Ein-Wort-Pro-Zeile-Format wie beim TreeTagger. Ebenso wird eine Konvertierung der Enkodierung ermöglicht, zum Beispiel, wenn ein Programm nicht Unicode-basiert arbeitet. Bei Korpora und anderen Daten wird ein Programm zur Recherche und zur Visualisierung der Annotationen angeboten. Unterrichtsbezogene Meta-Ressoucen ergänzen die Bereitstellung: Forschungsfragen, in denen die Ressource eine Rolle spielt, Entwürfe von Lehreinheiten und Übungsaufgaben mit der Ressource. Allgemein – ggf. ganz unabhängig von einzelnen Programmen – kann man in einem Webformular die Bewertung von Annotationen gegen einen beliebigen Goldstandard evaluieren bzw. die Inter-Annotatoren-Übereinstimmung berechnen, sowie sich Konfusionsmatrizen anzeigen lassen, die darstellen, welche Annotationsetiketten wie oft miteinander verwechselt wurden. Ebenso kann man über ein allgemeines Webformular Kodierungskonversionen online durchführen (z.B. utf-8 nach iso-latin-1).

An wen richten sich die genannten Wünsche? Die Entwickler der Ressourcen sind damit sicher überfordert. Die Umsetzung sollte vielmehr an zentralen Stellen geschehen, an denen Ressourcen und lehrrelevante Meta-Ressourcen nachhaltig bereitgestellt werden können.

5 Zusammenfassung und Ausblick

Mehrfach war angeklungen, dass die Online-Nutzung von Ressourcen eine gewünschte Option für die Lehre sei. Diese Idee war früh in der *Linksammlung zu computerlinguistische Online-Ressourcen* an der Universität Zürich umgesetzt worden.[16] Der Online-Service *WebLicht*[17] des Projekts D-Spin geht noch einen Schritt weiter. Es bietet ein online nutzbares Portal zur Verarbeitung von Sprachdaten an. Nutzer laden ihre Daten in das System und stellen sich per Drag&Drop eine Verarbeitungsabfolge aus verschiedenen Programmen zusammen, zum Beispiel aus einem Textsegmentierer, einem Wortartentagger und einem syntaktischen Parser. Der angereicherte Text wird anschließend wieder auf den lokalen Rechner gespeichert. Ressourcenentwickler sind aufgerufen, als Partner ihre Programme in Weblicht einzuspeisen. Eine andere Art von Online-Sammlung stellt die *Studienbibliographie Computerlinguistik* dar.[18] Thematisch strukturiert nennt sie vorwiegend Literatur, weist aber auch auf relevante Sprachressourcen hin. Das *Portal Computerlinguistik*[19], ein Gemeinschaftsprojekt der Gesellschaft für Sprachtechnologie und Computerlinguistik (GSCL) und der Computerlinguistik-Sektion der Deutschen Gesellschaft für Sprachwissenschaft (DGFS-CL), ist noch im Aufbau. Es hat das Ziel, Informationen und Ressourcen zur Sprachverarbeitung mit einem Schwerpunkt auf der Verarbeitung des Deutschen bereitzustellen. Einen speziellen Support für die Nutzung der Ressourcen in der Lehre gibt es bisher nicht.

Über den deutschsprachigen Tellerrand hinaus ist die *Open Language Archives Community* (OLAC) zu nennen, die eine weltweite virtuelle Bibliothek von Sprachressourcen aufbaut[20] sowie die Web-Links der Linguist List[21].

Müssen Lehrende einen computerlinguistischen Hintergrund besitzen, um Sprachressourcen in der Lehre einsetzen zu können? Auch in rein sprachwissenschaftlichen Seminaren können Studierende von Sprachressourcen wie Online-Korpora oder Online-Parsern profitieren, zu deren Nutzung man keine eigentliche computationelle Kompetenz benötigt. Die Ressourcen können helfen – wie von der Lernpsychologie gefordert – Sprachdaten und Analysen 'greifbar' zu machen.

Danksagung

Vielen Dank an die anonymen Gutachter für Korrekturen und hilfreiche Kommentare. Diese Arbeit ist gefördert aus dem Europäischen Sozialfonds in Baden-Württemberg.

[16] Züricher Sammlung zu Online-Demos: `http://kitt.cl.uzh.ch/kitt/cltools`

[17] WebLicht (`https://weblicht.sfs.uni-tuebingen.de/`) ist offen für Mitglieder der 'Clarin Identity Federation'. Viele Universitäten und andere Hochschule besitzen diesen Status.

[18] Studienbibliographie CL: `http://www.coli.uni-saarland.de/projects/stud-bib/`.

[19] Portal CL: `http://www.computerlinguistik.org/portal/portal.html?s=Home`.

[20] OLAC: `http://www.language-archives.org/`.

[21] Linguist List: `http://linguistlist.org`, zum Beispiel unter 'Education' die Links 'Software' und 'Linguistic Exercises and Aids'.

Literatur

Aston, G. (2000). Learning English with the British National Corpus. In Battaner, M. and López, C., editors, *VI jornada de corpus lingüístics*, pages 25–40. Barcelona.

Boyd, A., Gegg-Harrison, W., and Byron, D. (2005). Identifying non-referential it: A machine learning approach incorporating linguistically motivated patterns. In *Proceedings of the ACL Workshop on Feature Engineering for Machine Learning in Natural Language Processing*, pages 40–47, Ann Arbor, Michigan. Association for Computational Linguistics.

Dipper, S., Götze, M., and Stede, M. (2004). Simple annotation tools for complex annotation tasks: an evaluation. In *Proceedings of the LREC Workshop on XML-based Richly Annotated Corpora*, pages 54–62, Lisbon.

EAGLES (1996). Evaluation of natural language processing systems. Final report. EAGLES DOCUMENT EAG-EWG-PR.2. http://issco-www.unige.ch/projects/ewg96/ewg96.html.

Huber, L. (2004). Forschendes Lernen: 10 Thesen zum Verhältnis von Forschung und Lehre aus der Perspektive des Studiums. *Die Hochschule*, 2:29–49.

Müller, C. and Strube, M. (2006). Multi-level annotation of linguistic data with MMAX2. In Braun, S., Kohn, K., and Mukherjee, J., editors, *Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods*, pages 197–214. Peter Lang, Frankfurt a.M.

Naumann, K. (2006). *Manual of the Annotation of in-document Referential Relations*. University of Tübingen.

Poesio, M. (2000). Coreference. In Mengel, A., Dybkjaer, L., Garrido, J., Heid, U., Klein, M., Pirrelli, V., Poesio, M., Quazza, S., Schiffrin, A., and Soria, C., editors, *MATE Deliverable D2.1. MATE Dialogue Annotation Guidelines*, pages 134–187.

Rafferty, A. and Manning, C. D. (2008). Parsing three German treebanks: Lexicalized and unlexicalized baselines. In *Proceedings of the Workshop on Parsing German*, pages 40–46, Columbus, Ohio. Association for Computational Linguistics.

Reiber, K. (2007). Grundlegung: Forschendes Lernen als Leitprinzip zeitgemäßer Hochschulbildung. *Tübinger Beiträge zur Hoschuldidaktik: Forschendes Lernen als hochschuldidaktisches Prinzip – Grundlegung und Beispiele*, 3(1):6–12.

Schiller, A., Teufel, S., Stöckert, C., and Thielen, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS. Technical report, Institut für maschinelle Sprachverarbeitung, Stuttgart.

Schmid, H. (1997). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *New Methods in Language Processing*, pages 154–164. UCL Press, London.

Schmidt, T. (2004). Transcribing and annotating spoken language with EXMARaLDA. In *Proceedings of the LREC-Workshop on XML based richly annotated corpora, Lisbon 2004*, Paris. ELRA.

Winteler, A. (2004). *Professionell lehren und lernen. Ein Praxisbuch.* Wissenschaftliche Buchgesellschaft.

Zinsmeister, H. (2011). Exploiting the 'annotation cycle' for teaching linguistics. Vortrag beim Workshop Corpora in Teaching Languages and Linguistics. Berlin. http://www.linguistik.hu-berlin.de/institut/professuren/korpuslinguistik/events-en/CTLL/ctll_abstracts/ctll_zinsmeister.