Nils Diewald, Maik Stührenberg, Anna Garbar, Daniela Goecke

Serengeti – Webbasierte Annotation semantischer Relationen

Der Artikel stellt zum einen ein Annotationsschema für semantische Relationen vor, das für die Beschreibung eines deutschsprachigen Korpus für Training und Evaluation eines Systems zur Anaphernauflösung entwickelt wurde, zum anderen wird das webbasierte Annotationstool Serengeti beschrieben, das zur Annotation anaphorischer Relationen im Projekt A2 „Sekimo“ eingesetzt wird.[1] Im Gegensatz zu anderen Annotationstools benötigt Serengeti keine lokale Installation, was den Einsatz an einer großen Anzahl von Rechnern erleichtert. Darüber hinaus implementiert Serengeti ein Mehrbenutzerkonzept, das sowohl Gruppen als auch einzelne Nutzer unterstützt und zugehörige Dateien und Annotationen verwaltet.

1 Einleitung

In der Computerlinguistik und Sprachverarbeitung werden in verschiedenen Bereichen große Korpora qualitativ hochwertig annotierter Texte benötigt. Deren wachsende Bedeutung für die empirische Forschung, Hypothesentests sowie Training und Evaluation von Algorithmen maschinellen Lernens wird allgemein anerkannt. Um sowohl an Qualität wie an Quantität bestmögliche Ergebnisse zu erzielen, sind neben Annotationsschemata mit strikter Taxonomie und möglichst eindeutiger Interpretation einfach handhabbare Werkzeuge zur Annotation und Organisation der Korpora nötig, da sich die Erstellung einer empirischen Basis gerade im Forschungsgebiet der Anaphernresolution aufgrund der manuellen Annotation als hochgradig aufwändig erwiesen hat. Darüber hinaus kann auf Grund von Formatinkompatibilitäten selten auf bereits vorhandene Korpora zurückgegriffen werden. Insbesondere für die Erstellung von Korpora längerer Texte nimmt somit der Aspekt der nachhaltigen Korpuserstellung eine entscheidende Rolle ein.

Der Artikel gliedert sich wie folgt: Zunächst wird das Projekt „Sekimo“ vorgestellt (Abschnitt 2), um im Folgenden auf die darin geleistete Korpusarbeit, insbesondere in Bezug auf das zu Grunde liegende Annotationsschema und das verwendete Annotationsformat, einzugehen (Abschnitt 3). Das Annotationswerkzeug Serengeti wird in Abschnitt 4 vorgestellt, Abschnitt 5 behandelt aktuelle Entwicklungen hinsichtlich des generischen Repräsentationsformats SGF. Der Artikel schließt mit einem Ausblick auf die Weiterentwicklung von Serengeti (Abschnitt 6).

[1] Die in diesem Artikel präsentierten Arbeiten wurden im Rahmen des Projekts A2 „Sekimo“ der von der Deutschen Forschungsgemeinschaft geförderten Forschergruppe 437 *Texttechnologische Informationsmodellierung* durchgeführt. Für das genannte Korpus wurden zum einen vom Projekt A2 gesammelte Texte und zum anderen vom Projekt C1 zur Verfügung gestellte Texte verwendet.

2 Das Projekt „Sekimo“

Das Projekt A2 „Sekimo“ befasst sich mit der Integration heterogener linguistischer Ressourcen zur texttechnologischen Modellierung, wobei der Begriff Heterogenität sich hierbei beispielsweise auf das Repräsentationsformat oder die Funktion bezieht. Anwendungsdomäne ist die automatische Analyse anaphorischer Relationen. Um heterogene Ressourcen nutzbar zu machen, kommt ein abstraktes Datenformat zum Einsatz (vgl. Simons et al., 2004), wobei im Rahmen des Projekts sowohl ein auf einer Prolog-Faktenbasis aufbauendes (vgl. Witt et al., 2005) als auch ein rein XML-basiertes Repräsentationsformat (vgl. Stührenberg et al., 2006) entwickelt wurde. Mechanismen zur Integration sind notwendig, da es schwierig ist, die Ausgaben verschiedener linguistischer Ressourcen miteinander zu kombinieren: abgesehen von der Problematik, dass die aus einem Verarbeitungsschritt resultierende Ausgabedatei in den seltensten Fällen als Eingabe für einen nachfolgenden Verarbeitungsschritt verwendet werden kann, ist die Unifikation verschiedener Ausgaben (d. h. die Zusammenführung in eine einzelne XML-Datei) – die erst eine Analyse von Beziehungen zwischen Ebenen ermöglicht – auf Grund von XML-Inkompatibilitäten (überlappenden Elementstrukturen) oftmals nicht möglich.

3 Annotation anaphorischer Relationen

Die Darstellung der Annotation anaphorischer Relationen im Projekt „Sekimo“ ist unterteilt in die Diskussion des Annotationsschemas und der formalen Repräsentation in Form des Annotationsformats. Das Korpus enthält 49 deutschsprachige Texte, die sowohl aus Fachliteratur als auch aus Tages- und Wochenzeitungen stammen, davon wurden 14 Texte vollständig in Bezug auf anaphorische Relationen annotiert. Für diese Texte wurden insgesamt 4323 anaphorische Relationen auf der Basis von 11740 Diskursentitäten (65203 Token) annotiert.

3.1 Das Annotationsschema

Zur Annotation anaphorischer Relationen existiert eine Reihe an Formaten, angefangen von UCREL (vgl. Fligelstone, 1992; Garside et al., 1997) über das SGML-basierte MUC Annotationsschema (vgl. Hirschmann, 1997) hin zu dem auf XML basierenden MATE/GNOME Schema (vgl. Poesio, 2004), um nur einige zu nennen. Das im Projekt „Sekimo“ verwendete Schema basiert auf einer Annotationsrichtlinie für Koreferenzstrukturen, die im Projekt B1 „HyTex“ erarbeitet wurde (vgl. Holler et al., 2004), und die eine Erweiterung bzw. Präzisierung des genannten MATE/GNOME Schemas für die Anwendungsdomäne Hypertextualisierung darstellt. Die Grundidee besteht darin, die Unterscheidung zwischen Kospezifikation (vgl. Sidner, 1979) und Koreferenz in der Annotation abzubilden (vgl. Holler-Feldhaus, 2004). Während zwei Ausdrücke nur dann koreferieren, wenn sie auf dieselbe Entität in der Welt verweisen, genügt für Kospezifikation, dass ein Ausdruck einen vorangegangenen Ausdruck sprachlich wieder aufgreift.

Für das Projekt A2 wurde dieses Schema erweitert, um die Annotation indirekter Anaphorik (Bridging-Relationen, vgl. Clark, 1977) zu erlauben, hier ist das Antezedens einer Anapher nicht explizit realisiert, sondern muss aus dem Kontext erschlossen werden. Sowohl bei Kospezifikation als auch bei indirekter Anaphorik besteht neben der textuellen Ebene die semantische Interpretation: sprachliche Ausdrücke führen neue Diskursentitäten (Diskursreferenten in der Terminologie von Karttunen, 1976) ein und können auf bereits eingeführte Diskursreferenten verweisen; zwischen Diskursreferenten können semantische Relationen bestehen. Ein weiteres Schema, das die Annotation mehrsprachiger Korpora (*cross-linguistic anaphoric annotation*) fokussiert, stellen Krasavina and Chiarcos (2007) vor. Im Gegensatz zu dem vorgestellten Schema wird hier jedoch keine explizite Unterscheidung von Koreferenz und Kospezifikation angenommen.

Das Annotationsschema liegt in Form eines Manuals für die Annotatoren (Goecke et al., 2007a) sowie als XML DTD und XML Schema (XSD) vor, wobei die technische Realisation die Basis für das Annotationswerkzeug Serengeti darstellt.

Ausgangspunkt für die Annotation anaphorischer Relationen ist eine mehrstufige Vorverarbeitung, die verschiedene heterogene linguistische Ressourcen integriert. Zunächst werden Texte mit einer logischen Dokumentstruktur versehen, die u. a. Absätze, Sprachinseln, Abbildungen und Listen markiert, zusätzlich werden durch den funktional-dependenziellen Parser-Tagger Machinese Syntax der Firma Connexor Oy morphologisch-syntaktische Informationen auf Wortebene annotiert. Um Primärdatenidentität für die Unifikation mit der nicht-annotierten Referenz zu gewährleisten, werden die Originalannotationen des Parser-Taggers für die nachfolgenden Annotationsschritte modifiziert.[2] Der Begriff „Primärdatenidentität“ bezeichnet die Identität der zu Grunde liegenden Texte auf der Ebene der Zeichen. Im folgenden Schritt werden diejenigen Elemente, die Teil einer semantischen Relation sein können, so genannte *Markables*, identifiziert (vgl. Müller and Strube, 2001).[3] Im Projekt „Sekimo“, das die Detektion anaphorischer Relationen behandelt, dienen alle sprachlichen Ausdrücke, die einen Diskursreferenten im Sinne von Kamp and Reyle (1993) in die Diskurs- bzw. Textrepräsentation einführen, als relevante Diskursentitäten und somit als Markables. Die Identifikation erfolgt automatisch auf Basis der vom Machinese Syntax annotierten Wortformen. Zunächst werden einfache Diskursentitäten markiert, d. h., Diskursentitäten, die durch eine einfache NP realisiert sind. Aufbauend auf diesen können auch komplexe Diskursentitäten (also NPs mit Präpositionalphrase oder NPs mit NP als Prämodifizierer) annotiert werden. NPs mit Relativsatz werden nicht als komplexe Diskursentitäten markiert. Jede Diskursentität ist mit einem dokumentweit eindeutigen Identifikator versehen. Die zu untersuchenden semantischen Relationen, die zwischen Diskursentitäten bestehen, klassifizieren wir als Relationen der Kospezifikation (direkter Anaphorik) und indirekter Anaphorik (Bridging-Relationen, vgl. Clark, 1977; Vieira

[2] Eine Unifikation ist in diesem Fall problemlos möglich, da es zwischen der logischen Dokumentstruktur und der Annotation durch den Machinese Syntax nicht zu Überlappungen kommen kann.

[3] Die hier vorgestellte Version von Serengeti verwendet Texte mit vorannotierten Markables; eine derzeit in der Erprobung befindliche Fassung unterstützt bereits den Einsatz unannotierter Texte und das Hinzufügen von Markables während der Annotation.

and Poesio, 2001). Beide Typen können jeweils in weitere sekundäre Relationstypen unterteilt werden. Die direkte Anaphorik wird im Annotationsschema unterteilt in die Untertypen *ident*, *namedEntity*, *propName*, *synonym*, *hyperonym*, *hyponym*, *addInfo*, *paraphrase*. Der Wert *ident* wird vergeben, wenn sich ein Pronomen auf eine NP oder eine NP auf eine rekurrente NP bezieht. Kospezifikation zwischen einer NP, die nicht vom Typ *namedEntity* ist, und sich auf eine NP vom Typ *namedEntity* bezieht, wird mit dem entsprechenden sekundären Relationstyp ausgezeichnet. Der Wert *propName* wird vergeben, wenn die anaphorische Diskursentität ein Eigenname ist, und auf eine nominale Bezugsgröße im vorangegangenen Kontext verweist. Synonymiebeziehungen werden als solche markiert, wenn sich die Kopfnomen von Anapher und Antezedens in einer solchen befinden. Dabei ist zu beachten, dass im Projekt „Sekimo“ ein weiter Begriff der Synonymie verwendet wird, also der Kontext im Text entsprechend berücksichtigt wird, und auch Abkürzungen als Synonyme der jeweiligen Langform im Text ausgezeichnet werden. Hyperonymie und Holonymie zwischen den Kopfnomen von Anapher und Antezedens wird durch den entsprechenden sekundären Relationstyp ausgezeichnet. Bei den beiden Typen *addInfo* und *paraphrase* wird unterschieden, ob die kospezifierte NP neue oder zusätzliche Informationen einführt, bzw. die Anapher das Antezedens umschreibt.

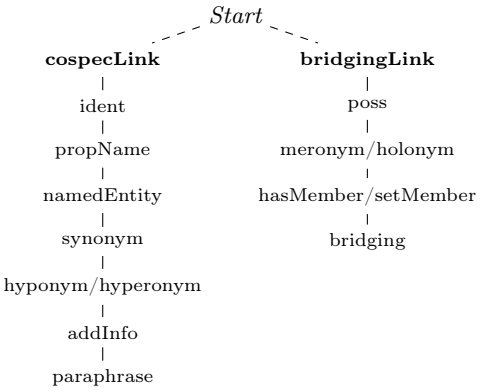

Abbildung 1: Der Entscheidungsbaum im Annotationsverlauf.

Analog zur Kospezifikation modelliert das vorliegende Annotationsschema auch Bridging-Relationen präziser. Dabei wurden die einzelnen Relationstypen so gewählt, dass sie durch linguistische Ressourcen (wie z. B. GermaNet, vgl. Goecke et al., 2007b) beschreibbar sind. Die Untertypen sind im Einzelnen: *poss*, *meronym*, *holonym*, *hasMember*, *setMember*, *bridging*. Als *poss* werden solche Relationen indirekter Anaphorik markiert, deren phorischer Ausdruck explizit durch ein Possessivpronomen oder eine Genitiv-NP besitzanzeigend markiert ist. Stehen Kopfnomen von Anapher und An-

tezedens in einer Meronymierelation, wird der entsprechende Wert genutzt, analog dazu die Holonymierelation. Davon abzugrenzen sind die Relationstypen *hasMember* und *setMember*. Ersterer liegt vor, wenn die Anapher eine Menge beschreibt, und das Antezendes ein Element dieser Menge; *setMember* wird als Relationstyp verwendet, wenn der phorische Ausdruck Element einer durch die Bezugsgröße beschriebenen Menge ist. Sollte keiner der genannten Relationstypen zutreffen, wird die allgemeine Relation *bridging* verwendet (z. B. Torte – Hochzeit). Eine weitere Unterteilung hinsichtlich Schema- oder Skriptbasierter Inferenz wird nicht vorgenommen. Zur Hilfestellung der Annotatoren bei der Entscheidung für einen Relationstyp wurde der in Abbildung 1 dargestellte Entscheidungsbaum entwickelt. Dabei können die Annotatoren den Entscheidungsbaum sequentiell überprüfen, d. h., nachdem sie die Entscheidung für Kospezifikation oder indirekte Anaphorik getroffen haben, können die einzelnen Subtypen nacheinander geprüft werden, wobei am Ende der Liste die allgemeinen Subtypen stehen, die nur gewählt werden sollten, sofern keiner der vorherigen Relationstypen als angemessen angesehen wurde. Darüber hinaus wurden Relationstypen definiert, die für Relationen gelten, deren Antezedentien durch nicht-nominale Einheiten eingeführt werden (z.B. Ereignisse, Fakten, Propositionen), und die wir der Terminologie von Asher (1993) folgend als *abstract event anaphora* bezeichnen. Für die Annotation von *abstract event anaphora* wurden drei Subtypen definiert: der Relationstyp *abstrProp* beschreibt anaphorische Relationen deren Antezdens durch eine Proposition eingeführt wird, Antezedentien des Relationstyps *abstrEvType* werden durch Ereignisse eingeführt und *abstrCluster* beschreibt diejenigen Relationen, deren Antezedens durch eine Summe von Propositionen bzw. durch einen Textabschnitt eingeführt wird.

3.2 Das Annotationsformat

Wie das MATE/GNOME-Schema ist das hier vorgestellte Annotationsformat XML-basiert und verwendet Standoff-Annotationen (vgl. Thompson and McKelvie, 1997), d. h. prinzipiell kann die Annotation unter Verwendung eines beliebigen XML-Editors durchgeführt werden. Listing 1 zeigt ein einfaches Beispiel aus dem Korpus, das mehrere Annotationsebenen enthält: die logische Dokumentstruktur (Element `para`), die Satzsegmentierung und Tokenisierung aus der Machinese Syntax-Ausgabe (Elemente `sentence` und `token`) sowie die Detektion der Diskursentitäten. Das Element `de`, das relevante Diskursentitäten markiert, trägt die drei obligatorischen Attribute `deID`, `deType` und `headRef`. Mittels `deID` kann über einen dokumentweit eindeutigen Wert jede Diskursentität identifiziert werden, `deType` gibt den Typ der Diskursentität (im Beispiel *namedEntity* oder *nom*) an und `headRef` referenziert das Kopfnomen der zu Grunde liegenden Nominalphrase (über XML ID/IDREF-Konstrukte). Token können Kindelemente der Elemente `de`, `sentence` oder `text` sein. Die diskurssemantischen Beziehungen werden als Kinder des Elements `standoff` gespeichert, hier finden sich auch weitere Informationen der Parser/Tagger-Ausgabe, die aus Gründen der Übersichtlichkeit ausgelagert wurden (Element `token_ref`).

Mittels des Elements `bridgingLink` wird indirekte Anaphorik zwischen zwei Diskursentitäten annotiert. Die in Abschnitt 3.1 genannten Subtypen werden dabei als Wert des Attributs `relType` spezifiziert, anaphorisches Element und Antezedens bzw. Antezedentien anhand ihrer dokumentweit eindeutigen ID in den Attributen `phorIDRef` und `antecedentIDRefs` referenziert. Analog dazu dient das Element `cospecLink` der Auszeichnung von Kospezfikation.

Listing 1: Das Annotationsformat für anaphorische Relationen.

```
<chs>
    <text>
        <para>
            <sentence>
                <de deID="de8" deType="namedEntity" headRef="w36">
                    <token ref="w36">Maik</token>
                </de>
                <token ref="w37">hat</token> <token ref="w38">kein</token>
                <token ref="w39">eigenes</token> <token ref="w40">Fahrrad</token>,
                <token ref="w42">und</token>
                <de deID="de10" deType="namedEntity" headRef="w43">
                    <token ref="w43">Marie</token>
                </de>
                <token ref="w45">fährt</token> <token ref="w46">nicht</token>
                <token ref="w47">in</token>
                <de deID="de11" deType="nom" headRef="w49">
                    <token ref="w48">den</token>
                    <token ref="w49">Urlaub</token>
                </de>.
            </sentence>
            <sentence>
                <de deID="de12" deType="nom" headRef="w53">
                    <token ref="w52">Zwei</token>
                    <token ref="w53">Kinder</token>
                </de>,
                <de deID="de13" deType="nom" headRef="w56">
                    <token ref="w55">eine</token>
                    <token ref="w56">Gemeinsamkeit</token>
                </de>:
            </sentence>
        </para>
    </text>
    <standoff>
        <token_ref id="w36" head="w37" pos="N" syn="@NH" depV="subj" morph="MSC␣SG␣
            NOM"/>
         [...]
        <semRel>
            <bridgingLink relType="hasMember" antecedentIDRefs="de8␣de10" phorIDRef="
                de12"/>
        </semRel>
    </standoff>
</chs>
```

Ambiguität wird durch die Definition mehrerer `cospecLink`- bzw. `bridgingLink`-Elemente realisiert, im Falle multipler Antezedentien wird auf mehrere Diskursentitäten in Antezedensposition verwiesen. Im Beispiellisting 1 besteht eine indirekt anaphorische Relation vom Typ *hasMember* zwischen den Antezedentien *Maik* (`de8`) und *Marie* (`de10`) und dem phorischen Ausdruck *Zwei Kinder* (`de12`).

Die Speicherung der semantischen Relationen als Standoff-Annotation am Ende der XML-Instanz ist der Tatsache geschuldet, dass die einzelnen Annotationsebenen jeweils durch eine entsprechende linguistische Ressource erstellt werden. Das allerdings erschwert die Verwendung eines einfachen XML-Editors zur Annotation, da oft zwischen verschiedenen Stellen im Dokument hin und her gewechselt werden muss. Aus diesem Grund wurde der Einsatz eines geeigneten Annotationswerkzeugs untersucht.

4 Serengeti

4.1 Annotationswerkzeuge

Für die Annotation unimodaler Daten sind in den letzten Jahrzehnten zahlreiche Werkzeuge entwickelt worden, die es dem Benutzer erleichtern, Texten Informationen hinzuzufügen. Neben für einen sehr begrenzten Einsatzbereich konzipierten Programmen, wie dem RST-Tool zur Erstellung von RST-Bäumen (vgl. O'Donnell, 1997), gibt es viele Annotationstools, die allgemeiner gestaltet sind und sich zur Beschreibung verschiedener semantischer Relationen in Texten eignen. Auch für die Annotation von Koreferenz existieren bereits spezialisierte Werkzeuge, wie XANADU (vgl. Garside and Rayson, 1997) oder der Coreferential Link Annotator (CLinkA, vgl. Orăsan, 2000). Die Vorteile, die eine Spezialisierung bietet, wie die optimierte Benutzerführung und die zielgerichtete Visualisierung, bringen jedoch auch Einschränkungen mit sich. So ist bei CLinkA das Annotationsschema direkt im Programm integriert und nicht erweiter- oder benutzerdefinierbar. Zugleich kann kein bereits auf einer anderen Beschreibungsebene annotierter Text verarbeitet werden. Dadurch lässt sich das Programm nur für sehr wenige Aufgaben einsetzen. Diese Beschränkungen führten zur Entwicklung des generalisierten Perspicuous and Adjustable Links Annotator (PALinkA, vgl. Orăsan, 2003), ein Werkzeug, das sich für unterschiedliche Schemata und Dokumente konfigurieren lässt. Solcherart generalisierte Annotationswerkzeuge haben den Vorteil, schnell an neue Aufgaben angepasst werden zu können. So muss der Benutzer nicht für jede neue Aufgabe den Umgang mit einem anderen Programm erlernen.

Zur Generalisierung verfolgen die Programme verschiedene Ansätze. MMAX (vgl. Müller and Strube, 2001), ein sehr populäres Tool zur Annotation von Koreferenz- und Bridgingbeziehungen, bietet eine große Funktionsvielfalt mit umfangreichen Möglichkeiten zur Definition des eigenen Annotationsschemas. WordFreak (vgl. Morton and LaCivita, 2003) hingegen bietet nur ein Basissystem, das durch Plugins, etwa die Integration automatischer Tagger, in seiner Funktionalität beliebig erweiterbar ist und es dem Nutzer erlaubt, sich seine persönliche Annotationsumgebung einzurichten. Weitere Annotationssysteme, wie GATE (vgl. Cunningham et al., 1996, 2002), bestehen

nicht aus einem einzigen Programm sondern aus einem Baukastensystem mit definierten Schnittstellen, durch die einzelne Programmmodule miteinander verbunden werden können.

Jeder dieser Generalisierungsansätze bedarf eines unterschiedlich großen Aufwands in Bezug auf die Konfiguration der Annotationsumgebung, um den eigenen Anforderungen zu genügen. Entweder müssen zu Beginn detaillierte Einstellungen bezüglich des Eingabe- und Ausgabeformats sowie des Annotationsschemas vorgenommen oder verschiedene Zusatzpakete zum Kernprogramm installiert werden. Da an der Erstellung eines Annotationskorpus unter Umständen viele Annotatoren beteiligt sind, die diese Vorgaben zu Beginn umsetzen müssen, kann der Konfigurationsaufwand erheblich sein. Ändert sich während der Arbeit das Annotationsschema – was insbesondere während der Entwicklungs- und Evaluationsphase nicht selten vorkommt – muss jeder Annotator diese Änderungen an seinem Programm vornehmen. Um dies zu vereinfachen, ist ein Konzept für kollaboratives Arbeiten vonnöten. Ein webbasiertes System beschränkt die Installation und Konfiguration der Umgebung auf einen Computer, den Web-Server. Beliebig viele Annotatoren können auf diese Umgebung zugreifen, ohne selbst aufwändige Konfigurationen vornehmen zu müssen; Änderungen des Annotationsschemas oder der Programmumgebung müssen nicht an mehreren Systemen vorgenommen werden. Auch für die Korpushaltung ist ein zentrales Konzept von Vorteil, um ortsungebunden und jederzeit auf das Korpus zugreifen zu können. Zudem ermöglicht es Annotatoren, zeitgleich an identischen Dokumenten zu arbeiten, unabhängig von ihrem Standort. Das Annotation Graph Toolkit (AGTK, vgl. Maeda et al., 2001; Ma et al., 2002) bietet die Möglichkeit, durch ein Client-Server-Modell mit mehreren Annotatoren an einer Annotation zu arbeiten. Das Annotat wird hierbei zentral in einer Server-Datenbank verwaltet, die Annotationsumgebungen selbst bleiben allerdings lokal installiert.

Da im Projekt „Sekimo“ neben der Korpuserstellung viel Wert auf die Evaluation des Annotationsschemas gelegt wurde, verfolgen wir bei der zentralen Korpusverwaltung den Ansatz, mehreren Personen zu ermöglichen, unabhängig voneinander ein Dokument zu annotieren und erst in einem weiteren Schritt durch Vergleich und Unifikation von Annotationen – anstelle von gemeinschaftlicher Annotation – eine verbindliche Fassung (*Gold Standard*) zu erstellen. Auf diese Weise kann eine Evaluation des Schemas durch einen Inter-Annotator-Vergleich stattfinden.

4.2 Architektur

Serengeti ist eine webbasierte Client-Server-Applikation für den Mozilla Firefox Browser[4] zur Annotation semantischer Relationen in Texten. Die hier vorgestellte Version des Programms ist noch weitgehend spezialisiert, mit einem im Vergleich zu den vorangehend vorgestellten Systemen geringen Funktionsumfang, und wird aktuell zu einem konfigurier- und erweiterbaren System ausgebaut.

[4] Serengeti unterstützt Firefox ab Version 1.5; Die Browsersoftware ist frei verfügbar unter `http://www.mozilla.com/firefox/`.

Auf Client-Seite, zur Darstellung des grafischen Benutzerinterfaces, werden bewährte Web-Technologien verwendet (XHTML, CSS, Javascript), auf Serverseite wird Perl eingesetzt.

Die Kommunikation zwischen Client und Server wird dabei aufgabenbedingt unterschiedlich gelöst: Lade- und Speicheroperationen mit geringem Datentransfer (etwa dem Empfang der Dokumentlisten oder dem Speichern erstellter Relationen) werden mittels einer AJAX-Engine durchgeführt (Asynchronous JavaScript and XML, Garrett, 2005), während für Operationen mit umfangreichem Datentransfer, wie dem Rendern der Dokumente, auf das klassische, synchrone Modell in Verbindung mit eingebetteten Frames zurückgegriffen wird (s. Abb. 2).[5]

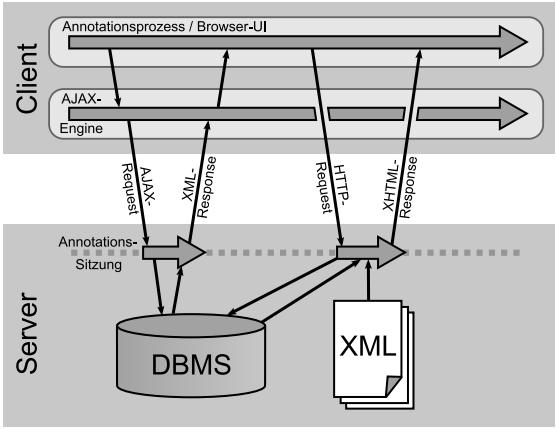

Abbildung 2: Client-Server-Kommunikation (vgl. Garrett, 2005)

Die zu annotierenden Dokumente sind als XML-Instanzen auf dem Server gespeichert und werden beim Aufruf in ein XHTML-Dokument transformiert. Die Annotationen, Projekt- und Benutzerdaten werden von einer MySQL-Datenbank verwaltet.
Die verteilte Architektur erlaubt es, Korpus- und Benutzerverwaltung serverseitig zu realisieren und die technischen Anforderungen auf Clientseite niedrig zu halten. Haben mehrere Personen ein und dasselbe Dokument annotiert, ermöglicht die zentrale Korpushaltung einen Vergleich und eine gesteuerte Unifikation beider Annotationen durch den Projektleiter.

4.3 Annotation mit SERENGETI

Nach der Anmeldung auf der Webseite[6] werden im oberen Teil der SERENGETI-Oberfläche zwei Menüs in Form von Auswahllisten eingeblendet (Gruppen- und Dokument-Menü, s. Abb. 3), durch die der Benutzer die Möglichkeit hat, das Annotationsprojekt sowie das zu annotierende Dokument auszuwählen.

Nach dem Laden des Dokuments kann unmittelbar mit der Annotation begonnen werden. Im oberen Abschnitt der grafischen Oberfläche, dem Text-Fenster, wird der zu annotierende Text visualisiert, der untere Abschnitt teilt sich in das Relations-Fenster auf der linken und das Editier-Formular auf der rechten Seite. Der Text wird mit Formatierungen bezüglich Paragraphen, Listen, Tabellen und nicht-textuellen Elementen

[5] Um Datenverlust zu vermeiden, sind während des Datentransfers allerdings keine weiteren Benutzeraktionen erlaubt, was dem „klassischen“ asynchronen AJAX-Ansatz widerspricht.
[6] Eine Demo-Installation ist unter `http://coli.lili.uni-bielefeld.de/serengeti/` zu finden.

dargestellt. Zudem sind alle Markables im Text durch Unterstriche markiert und mit ihrer eindeutigen ID ausgezeichnet, repräsentiert durch anklickbare Boxen, die es dem Annotator ermöglichen, die an einer semantischen Relation beteiligten Markables per Mausklick auszuwählen (oder gegebenenfalls zu verwerfen).

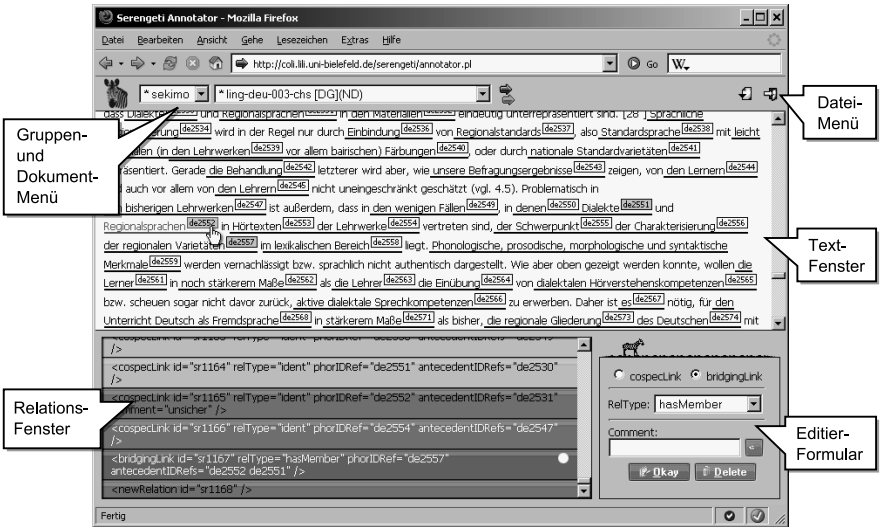

Abbildung 3: SERENGETI Hauptfenster

Für die Definition einer semantischen Relation werden im Rahmen des „Sekimo"-Projekts das anaphorische Element und ein oder mehrere Antezedentien markiert. Für die Annotation anaphorischer Relationen muss *zuerst* die Anapher und anschließend ein Antezedens bzw. mehrere Antezedentien ausgewählt werden. Um zwischen den beiden Typen der Diskursentitäten im Text zu unterscheiden, werden diese verschiedenfarbig (rosa – Anapher, blau – Antezedentien) dargestellt. Im nächsten Schritt wird die zwischen den beiden Diskursentitäten bestehende Relation definiert. Alle annotierten Beziehungen werden im Relations-Fenster als XML-Elemente gelistet. Am Anfang ist diese Liste bis auf einen gelben Balken leer, der die sogenannte `newRelation` enthält. Sie verbindet noch keine DEs und ist mit einem weißen Punkt versehen, der die aktuell ausgewählte Relation markiert.

Das Editier-Formular im rechten unteren Bereich des Fensters hält spezielle Optionen für die Erstellung und Bearbeitung von Relationen bereit. Im Falle der Annotation anaphorischer Relationen nach dem „Sekimo"-Annotationsschema (vgl. Abschnitt 3.1) soll zunächst der primäre Relationstyp bestimmt werden. Hier wird zwischen Kospezifikation und indirekter Anaphorik unterschieden. Dabei wird im Relations-Fenster nach Selektion des primären Typs der Elementname der `newRelation` zu `bridgingLink`

bzw. `cospecLink` geändert. Anaphern und Antezedentien werden durch die Attribute `phorIDRef` und `antecedentIDRefs` kodiert. Je nach ausgewähltem primären Typ ändert sich das Set der sekundären Relationstypen (vgl. Abb. 1), das im Editier-Formular als Auswahlliste dargestellt und dessen Wert vom Attribut `relType` übernommen wird.

Zusätzlich können Kommentare im Attribut `comment` gespeichert werden. Dies ist hilfreich, wenn der Annotator einen Vermerk zur annotierten Relation machen möchte. Solche Relationen werden grün eingefärbt. Nachdem eine Relation annotiert wurde, wird dies mit der Schaltfläche „Okay“ bestätigt. Bei vollständigen Relationen ohne Kommentare ändert sich die Farbe von Gelb auf Blau und eine neue `newRelation` wird im Relations-Fenster angelegt, welche als Nächstes bearbeitet werden kann. Ist nicht entscheidbar, auf welches Antezedens sich eine Anapher bezieht, können mehrere Relationen mit dieser Anapher definiert werden. Fehlerhafte Relationen können mit Hilfe des „Delete“-Buttons gelöscht werden. Diese Relationen werden zunächst rot hinterlegt und erst nach dem Speichern der Annotation endgültig aus der Liste entfernt. Unvollständig annotierte Relationen (z.B. in Bezug auf die teilnehmenden Diskursentitäten) werden in der Liste orangefarbig hervorgehoben und können später korrigiert werden.

Mit Hilfe des Datei-Menüs (s. Abb. 3) können Annotationen verwaltet werden, etwa durch Speichern, Drucken oder Exportieren. Die „View“-Option ermöglicht dem Annotator, abgeschlossene (d. h. weder kommentierte noch unvollständige) Relationen im Relations-Fenster auszublenden, um eine bessere Übersicht über die Annotation zu erhalten. Zum anderen können diejenigen Diskursentitäten, die bereits anaphorische Verwendung gefunden haben, durch die farbliche Hervorhebung ihrer ID-Boxen im Text-Fenster angezeigt werden.

4.4 Annotationsvergleich mit SERENGETI

Im so genannten *Consensus*-Modus besteht für bestimmte Mitglieder der Annotationsgruppe (die *Consensus-User*) die Möglichkeit, die Qualität der Annotationen mittels Inter-Annotator-Agreement zu verifizieren. Gleichermaßen lässt sich so auch das Schema überprüfen. Dieses Vorgehen hilft, die besten Annotationsergebnisse zu erzielen. Dabei werden zwei Annotationen im Relations-Fenster gleichzeitig dargestellt, wobei Relationen nach bestimmten Kriterien, etwa ihren anaphorischen Elementen, sortiert werden. In beiden Annotationen identische Relationen werden grau hinterlegt; in ausschließlich einer Annotation vorkommende erscheinen nur auf der entsprechenden Seite.

Falls Relationen lediglich ein Element (eine DE oder den Relationstyp) gemeinsam haben, werden sie einander gegenüber gestellt, wobei die Unterschiede hervorgehoben werden (s. Abb. 4: das anaphorische Element ist in beiden An-

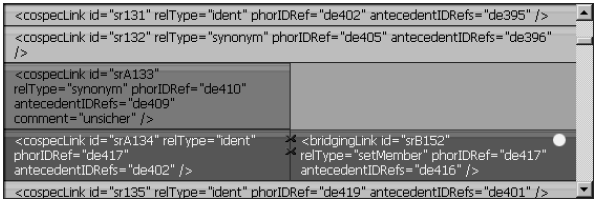

Abbildung 4: *Consensus*-Modus

notationen gleich, der Relationstyp und das Antezedens unterscheiden sich). Die im *Consensus*-Modus dargestellten Relationen können wie Relationen im Annotations--Modus bearbeitet (d. h. entfernt, geändert oder bestätigt) und die Annotation ebenso gespeichert werden. Ist eine Vergleichs-Annotation widerspruchsfrei, kann diese an weiteren Vergleichen teilnehmen.

5 SGF – Sekimo Generic Format

Neben der Verwendung innerhalb des „Sekimo"-Projekts wird SERENGETI auch für andere Anwendungen eingesetzt: abgesehen von der Möglichkeit, die aktuelle Version zur Annotation von lexikalischen Ketten einzusetzen (für eine prototypische Implementation vgl. Stührenberg et al., 2007), werden Teile der Architektur im Rahmen einer Kooperation mit der Universität Essex für die Projekte „AnaWiki" (vgl. Poesio and Kruschwitz, 2008) und die „AnaphoricBank"[7] genutzt und erweitert (s. Abschnitt 6). Unter anderem zu diesem Zweck wurde das *Sekimo Generic Format* (SGF) als Austauschformat für eine generalisierte Version von SERENGETI entwickelt. Ein weiteres Einsatzgebiet ist die weitergehende Analyse von Zusammenhängen zwischen einzelnen Elementen verschiedener Annotationsebenen. Dazu können unterschiedliche Architekturen eingesetzt werden. Der bisher im Projekt „Sekimo" verfolgte Weg (vgl. Abschnitt 3.2) war der Einsatz einer Standoff-Annotation sowie die Verwendung einer Prolog-Faktenbasis (vgl. Witt et al., 2005). Dabei erlaubt die Prolog-Faktenbasis die Analyse von Beziehungen zwischen Elementen verschiedener Annotationsebenen und ermöglicht so Aufschlüsse über mögliche Zusammenhänge zwischen linguistischen Merkmalsstrukturen (vgl. Lüngen et al., 2008). Hierzu müssen die XML-Instanzen in das Prolog-Format überführt werden. Unifikationen, die in der Prolog-Faktenbasis durchgeführt werden, nutzen zum XML-Export *Milestones* bzw. *Fragments* (vgl. Sperberg-McQueen and Burnard, 2002; Witt, 2002; DeRose, 2004), um überlappende Elemente auszuschließen.

Das im folgenden vorgestellte alternative *Sekimo Generic Format* hingegen ist vollständig XML-basiert und somit unabhängig von Zwischenformaten und erlaubt für den gesamten Prozess der Verarbeitung die Nutzung von XML-Software. Grundlage dazu ist das Konzept des *Annotation Graph* (vgl. Bird and Liberman, 1999, 2001), der einen Zeit- bzw. Zeichenstrahl als Basis für die Alignierung von Annotationen an die zu annotierenden Daten nutzt[8] – im Gegensatz zum *OHCO*-Modell (vgl. Renear et al., 1996), das eine geordnete Hierarchie aus verschachtelten Elementen modelliert, die sich als Baum darstellen lässt. Der Grund für den Einsatz eines graphenbasierten Modells liegt in der Problematik, mittels OHCO-basierter Inline-Annotation multiple Annotationen im Sinne einer Markup-Unifikation miteinander in Beziehung zu setzen, da es hier zu Überlappungen zwischen Elementen aus verschiedenen Ebenen kommen kann, die in XML nicht gestattet sind. Entsprechende Arbeiten dazu finden sich neben SGF in den Standardisierungsbestrebungen des ISO/TC 37/SC4 mit dem *Linguistic Annotation*

[7] `http://www.anaphoricbank.org`

[8] Das Konzept des Annotation Graph nutzt gelabelte azyklische Digraphen zur Darstellung linguistischer Annotationen.

Framework (LAF) und dem *Graph-based Format for Linguistic Annotations* (GraF; vgl. Ide, 2006; Ide and Suderman, 2007; Ide and Romary, 2007) sowie auf nationaler Ebene unter anderem im Kooperationsprojekt C2 der SFBs 441, 538 und 632, „Nachhaltigkeit linguistischer Daten“ (vgl. u. a. Dipper et al., 2006; Wörner et al., 2006; Eckart, 2006; Teich and Eckart, 2007; Witt et al., 2007).

Das Konzept der Datenhaltung von SGF sieht vor, alle zu einem Primärdatum zugehörigen Annotationen in einer Instanz zu speichern (im Gegensatz zu den anderen genannten Architekturen). Eine SGF-Instanz kann sowohl im Dateisystem (als Datei), in einer nativen XML-Datenbank, als auch in einer relationalen Datenbank oder einem hybriden Datenbanksystem gespeichert werden.[9]

Prinzipiell ist das Format sowohl zur Speicherung von textuellen als auch multimodalen Primärdaten nebst Annotationen geeignet und kann damit zur Analyse beliebiger linguistischer Phänomene herangezogen werden.[10] SGF ist vollständig XML-Schema--basiert und nutzt XML Namespaces (vgl. Bray et al., 2006) zur Trennung der einzelnen Annotationsebenen. Eine SGF-Instanz besteht immer aus dem *Base Layer* mit dem Namespace `http://www.text-technology.de/sekimo` und dem Präfix `base`, der grundlegende Funktionalitäten, Elemente und Attribute zur Verfügung stellt. Darüber hinaus kann eine beliebige Anzahl an Annotationsebenen, die jeweils eigenen XML-Namespaces zugeordnet werden, durch die `import`-Funktionalität in das Basis-Schema integriert werden (vgl. Thompson et al., 2004). Zur Validierung der jeweiligen Annotationsebenen können die ursprünglichen Dokumentgrammatiken (sofern sie als XSD vorliegen) genutzt werden, da das Basis-Schema sowohl für Metadaten als auch für Kindelemente des `layer`-Elements Konstrukte aus anderen Namensräumen zulässt. Listing 2 zeigt die nach SGF konvertierte Beispielannotation aus Listing 1.

Das Wurzelelement `corpus`, das mit einer eindeutigen ID und dem Korpustyp (*text* oder *multimodal*) versehen ist, umfasst ein oder mehrere `corpusdata`-Elemente. Im Kindelement `primaryData` können textuelle Primärdaten direkt gespeichert werden (bei kürzeren Texten, als Inhalt des `textualContent`-Elements) oder es wird mittels des Attributs `uri` des `location`-Elements auf eine externe Datei referenziert. Die Attribute `start` und `end` speichern den Wert des ersten bzw. letzten Zeichens (sofern es sich um einen Text handelt, sonst die Start- und Endzeit) der Primärdaten. Dabei wird jedes Zeichen, also auch Whitespaces (Leerzeichen, Umbrüche, Tabstops etc.) gezählt, empfehlenswert ist daher eine vorherige Normalisierung der Primärdaten in Bezug auf solche Zeichen. Es besteht die Möglichkeit, mittels des optionalen Elements `checksum` eine Prüfsumme für die Primärdaten zu speichern (im Listing 2 nicht gezeigt), die gewährleistet, dass externe Ressourcen auf dem gleichen Eingabetext arbeiten. Optionale Metadaten (Element `meta`, im Beispiel nicht enthalten) können dem gesamten Korpus

[9] Aktuelle Entwicklungsstufen von Serengeti nutzen ein SGF-API (Application Programming Interface), dem die Abbildung von SGF auf ein relationales Datenbanksystem (z.B. MySQL) zu Grunde liegt.

[10] Bei multimodalen Primärdaten wird an Stelle des Zeichenstrahls ein Zeitstrahl zur Alignierung der Annotationen genutzt. Die Verwendung multipler Primärdaten (z. B. einer Video- und einer Audiospur) ist möglich, allerdings muss ein Primärdatum ausgewählt werden, das den Zeitstrahl vorgibt.

Listing 2: SGF-Instanz (Ausschnitt)

```
<base:corpus  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://www.text-technology.de/sekimo"
    xmlns:base="http://www.text-technology.de/sekimo">
    <base:corpusData  xml:id="c1" type="text" sgfVersion="1.0">
        <base:primaryData  start="0" end="100" xml:lang="de">
            <base:location  uri="c1.txt"/>
        <base:primaryData>
        <base:segments>
            <base:segment  xml:id="seg1" start="0" end="100"/>
            <base:segment  xml:id="seg2" start="0" end="67"/>
            <base:segment  xml:id="to1" type="char" start="0" end="4"/>
            <base:segment  xml:id="to13" type="char" start="68" end="72"/>
            <base:segment  xml:id="seg5" type="seg" segments="to13␣to14"/>
        </base:segments>
        <base:annotation>
            <base:level  xml:id="doc" priority="0">
                <base:layer  xmlns:doc="http://www.text-technology.de/sekimo/doc">
                    <doc:text  base:segment="seg1">
                        <doc:para  base:segment="seg1"/>
                    </doc:text>
                </base:layer>
            </base:level>
        </base:annotation>
        <base:annotation>
            <base:level  xml:id="cnx" priority="0">
                <base:layer  xmlns:cnx="http://www.text-technology.de/cnx">
                    <cnx:sentence  id="w35" base:segment="seg2">
                        <cnx:token  base:segment="to1" xml:id="w36" head="w37" pos="N"
                              syn="@NH"
                                                depV="subj" morph="MSC␣SG␣NOM"/>
                    </cnx:sentence>
                </base:layer>
            </base:level>
        </base:annotation>
        <base:annotation>
            <base:level  xml:id="de" priority="1">
                <base:layer  xmlns:chs="http://www.text-technology.de/sekimo/chs">
                    <chs:de  base:segment="to1" deID="de8" deType="namedEntity"
                         headRef="w37"/>
                    <chs:de  base:segment="to7" deID="de10" deType="namedEntity"
                         headRef="to1"/>
                    <chs:de  base:segment="seg5" deID="de12" deType="nom" headRef="w53
                         "/>
                </base:layer>
            <base:level  xml:id="chs" priority="1">
                <base:layer  xmlns:chs="http://www.text-technology.de/sekimo/chs">
                    <chs:semRel>
                        <chs:bridgingLink  xml:id="sr1" relType="hasMember" phorIDRef=
                             "de12"
                                                antecedentIDRefs="de8␣de10"/>
                    </chs:semRel>
                </base:layer>
            </base:level>
        </base:annotation>
    </base:corpusData>
</base:corpus>
```

(als Kindelement von `corpus`), einzelnen Korpuseinträgen (unterhalb von `corpusData`) oder einer Annotationsebene (als Kindelement von `level`) zugeordnet werden, im Projekt „Sekimo“ werden hierzu Metadaten der „Open Language Archives Community“ (vgl. Simons and Bird, 2003) verwendet.

Da Annotationen am Zeichenketten- bzw. Zeitstrahl aligniert werden, werden für jede Annotationsebene Segmentierungen vorgenommen (`segments`). Dabei sollten neue Segmente (`segment`) nur dann hinzugefügt werden, wenn ein Element mit den entsprechenden Start- und Endpositionen nicht bereits durch Annotationen einer anderen Ebene eingeführt wurde. Da jedes Segment durch das Attribut `xml:id` eindeutig identifizierbar ist, kann im Anschluss der Segmentierung entsprechend darauf verwiesen werden. Eine Besonderheit stellt das Segment 'seg5' in Zeile 13 in Listing 2 dar: es besteht aus zwei Segmenten, womit eine hierarchische Beziehung zwischen Segmenten kodiert werden kann, die auch überlappende Segmente erlaubt. Jedes `corpusData`-Element kann eine Reihe von `annotation`-Kindelementen beinhalten. Dabei steht jedes `annotation`-Element für eine Annotationseinheit, innerhalb derer eine oder mehrere Annotationsebenen stehen dürfen – wobei das Element `level` die konzeptuelle Ebene der Annotation und das Element `layer` die XML-Realisierung speichert. Die Unterscheidung wird deutlich beim Vergleich der Annotationen, die jeweils die Ebene *doc* (logische Dokumentstruktur) bzw. *cnx* (Parser/Tagger-Ausgabe) beinhalten, mit der Annotation, die sowohl die Ebene *de* (Ebene der Diskursentitäten) als auch *chs* (Ebene der semantischen Relationen) beinhaltet.

Innerhalb eines `layer`-Elements sind die modifizierten Annotationen aus dem Ursprungsdokument enthalten. Dabei werden die Elemente wie folgt geändert: Elemente mit textuellem Inhalt (`PCDATA`) werden in leere Elemente überführt, Elemente mit gemischtem Inhaltsmodell werden zu reinen Container-Elementen (d. h. ohne *mixed content*). Elemente, deren Inhaltsmodell bisher nur aus anderen Elementen bestand, bleiben unverändert. So bleibt insbesondere die Hierarchiebeziehung zwischen Elementen einer Annotationsebene weiterhin direkt kodiert. Die Attribute bleiben ebenfalls unverändert – allerdings wird jedem Element das Attribut `segment` aus dem *Base Layer* hinzugefügt. Die im Listing 1 noch vorhandene Auslagerung der Token-Informationen mittels `token_ref` ist unnötig.

Relationen zwischen Elementen verschiedener Annotationsebenen lassen sich durch XPath- bzw. XQuery-Ausdrücke (vgl. Berglund et al., 2007; Boag et al., 2007) identifizieren. Im Verbund mit einer nativen XML-Datenbank oder einem hybriden Datenbanksystem lassen sich entsprechend umfangreiche Abfragen realisieren – aber auch auf Dateiebene lassen sich solche mit geeigneten XQuery-Prozessoren wie z. B. Saxon[11] durchführen. Eine ausführlichere Darstellung des Formats inklusive Evaluation ist in Stührenberg and Goecke (2008) gegeben.

[11] `http://saxon.sourceforge.net` bzw. `http://www.saxonica.com`

6 Zusammenfassung und Ausblick

Die in diesem Artikel vorgestellte Version des webbasierten Annotationssystems SERENGETI bietet bereits eine Reihe hilfreicher Werkzeuge zur Annotation semantischer Relationen und grenzt sich aufgrund seiner Architektur von vergleichbaren Werkzeugen ab. Das zu Grunde liegende Annotationsschema hat sich als sinnvolle Basis für die bisherige Annotationsarbeit erwiesen.

Im Zuge der aktuellen Generalisierung, zu der die Nutzung einer auf dem *Sekimo Generic Format* beruhenden Datenbank ebenso gehört wie die Möglichkeit, Markables während der Annotation hinzuzufügen und zu editieren, werden sowohl auf Client- als auch auf Serverseite Schnittstellen für Plugins etabliert. Diese erlauben eine Erweiterung der Werkzeugpalette sowie die Anpassung der Arbeitsumgebung an die Erfordernisse weiterer Annotationsaufgaben. Hierbei wird es möglich sein, beliebige Typen von Relationen und Markables für neue Annotationsprojekte zu definieren und für beliebige SGF-Layer Transformationsfilter zu ergänzen, die die HTML-Ausgabe steuern. Des Weiteren sind zusätzliche Funktionen für den Inter-Annotator-Vergleich geplant, etwa die automatische Berechnung von Übereinstimmungswerten.

Literatur

Asher, N. (1993). *Reference to abstract objects in discourse*, volume 50 of *Studies in Linguistics and Philosophy*. Kluwer Academic Publishers, Dordrecht, London, Boston.

Berglund, A., Boag, S., Chamberlin, D., Fernández, M. F., Kay, M., Robie, J., and Siméon, J. (2007). XML Path Language (XPath). Version 2.0. W3C Recommendation, World Wide Web Consortium.

Bird, S. and Liberman, M. (1999). Annotation graphs as a framework for multidimensional linguistic data analysis. In *Proceedings of the Workshop "Towards Standards and Tools for Discourse Tagging"*, pages 1–10. Association for Computational Linguistics.

Bird, S. and Liberman, M. (2001). A Formal Framework for Linguistic Annotation. *Speech Communication*, 33(1–2):23–60.

Boag, S., Chamberlin, D., Fernández, M. F., Florescu, D., Robie, J., and Siméon, J. (2007). XQuery 1.0: An XML Query Language. W3C Recommendation, World Wide Web Consortium.

Bray, T., Hollander, D., Layman, A., and Tobin, R. (2006). Namespaces in XML 1.0 (2nd Edition). W3C Recommendation, World Wide Web Consortium.

Clark, H. (1977). Bridging. In Johnson-Laird, P.N. & Wason, P., editor, *Thinking: Readings in Cognitive Science*, pages 411–420. Cambridge University Press, Cambridge.

Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). GATE: An Architecture for Development of Robust HLT Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, pages 168–175. ACL.

Cunningham, H., Wilks, Y., and Gaizauskas, R. J. (1996). GATE – a General Architecture for Text Engineering. In *Proceedings of the 16th Conference on Computational Linguistics*, Copenhagen. COLING.

DeRose, S. J. (2004). Markup Overlap: A Review and a Horse. In *Proceedings of Extreme Markup Languages*.

Dipper, S., Hinrichs, E., Schmidt, T., Wagner, A., and Witt, A. (2006). Sustainability of Linguistic Resources. In Hinrichs, E., Ide, N., Palmer, M., and Pustejovsky, J., editors, *Proceedings of the LREC 2006 Satellite Workshop on "Merging and Layering Linguistic Information"*, Genua.

Eckart, R. (2006). Towards a modular data model for multi-layer annotated corpora. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 183–190, Sydney, Australia. Association for Computational Linguistics.

Fligelstone, S. (1992). Developing a Scheme for Annotating Text to Show Anaphoric Relations. In Leitner, G., editor, *New Directions in English Language Corpora: Methodology, Results, Software Developments*, pages 153–170. Mouton de Gruyter, Berlin.

Garrett, J. J. (2005). *AJAX: A New Approach to Web Applications*. Adaptive Path LLC. Online: `http://www.adaptivepath.com/publications/essays/archives/000385.php`.

Garside, R., Fligelstone, S., and Botley, S. (1997). Discourse Annotation: Anaphoric Relations in Corpora. In Garside, R., Leech, G., and McEnery, A., editors, *Corpus Annotation: Linguistic Information from Computer Text Corpora*, pages 66–84. Addison-Wesley Longman, London.

Garside, R. and Rayson, P. (1997). Higher-level annotation tools. In Garside, R., Leech, G., and McEnery, A., editors, *Corpus Annotation: Linguistic Information from Computer Text Corpora*, pages 179–193. Addison-Wesley Longman, London.

Goecke, D., Stührenberg, M., and Holler, A. (2007a). Koreferenz, Kospezifikation und Bridging: Annotationsschema. Interne Reports der DFG-Forschergruppe 437 "Texttechnologische Informationsmodellierung".

Goecke, D., Stührenberg, M., and Wandmacher, T. (2007b). Extraction and representation of semantic relations for resolving definite descriptions. extended abstract. In Mönnich, U. and Kühnberger, K.-U., editors, *OTT'06. Ontologies in Text Technology: Approaches to Extract Semantic Knowledge from Structured Information*, volume 1-2007 of *Publications of the Institute of Cognitive Science (PICS)*, pages 27–32. Institute of Cognitive Science, Osnabrück.

Hirschmann, L. (1997). MUC-7 Coreference Task Definition (version 3.0). In Hirschman, L. and Chinchor, N., editors, *Proceedings of Message Understanding Conference (MUC-7)*.

Holler, A., Maas, J.-F., and Storrer, A. (2004). Exploiting Coreference Annotations for Text-to-Hypertext Conversion. In *Proceedings of the 4th International Conference on Language Resources and evaluation (LREC 2004)*, volume II, pages 651–654, Lisbon, Portugal.

Holler-Feldhaus, A. (2004). Koreferenz in Hypertexten: Anforderungen an die Annotation. *Osnabrücker Beiträge zur Sprachtheorie (OBST)*, pages 9–29.

Ide, N. (2006). ISO/TC 37/SC4 N311: Linguistic Annotation Framework. Technical report, ISO/TC 37/SC4.

Ide, N. and Romary, L. (2007). Towards International Standards for Language Resources. In Dybkjaer, L., Hemsen, H., and Minker, W., editors, *Evaluation of Text and Speech Systems*, pages 263–284. Springer.

Ide, N. and Suderman, K. (2007). GrAF: A Graph-based Format for Linguistic Annotations. In *Proceedings of the Linguistic Annotation Workshop*, pages 1–8, Prague, Czech Republic. Association for Computational Linguistics.

Kamp, H. and Reyle, U. (1993). *From Discourse to Logic*. Kluwer: Dordrecht.

Karttunen, L. (1976). Discourse Referents. *Syntax and Semantics: Notes from the Linguistic Underground*, 7:363–385.

Krasavina, O. and Chiarcos, C. (2007). PoCoS - Potsdam Coreference Scheme. In *Proceedings of the Linguistic Annotation Workshop*, pages 156–163, Prague, Czech Republic. Association for Computational Linguistics.

Lüngen, H., Bärenfänger, M., Goecke, D., Hilbert, M., and Stührenberg, M. (2008). Anaphoric relations as cues for rhetorical relations. erscheint in LDV Forum – Zeitschrift für Computerlinguistik und Sprachtechnologie.

Ma, X., Haejoong, L., Bird, S., and Maeda, K. (2002). Models and Tools for Collaborative Annotation. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, Paris. European Language Resources Association.

Maeda, K., Bird, S., Ma, X., and Lee, H. (2001). The Annotation Graph Toolkit: Software Components for Building Linguistic Annotation Tools. In *HLT '01: Proceedings of the first international conference on Human language technology research*, pages 1–6, Morristown, NJ, USA. Association for Computational Linguistics.

Morton, T. and LaCivita, J. (2003). WordFreak: An Open Tool for Linguistic Annotation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 17–18, Edmonton, Canada.

Müller, C. and Strube, M. (2001). Annotating Anaphoric and Bridging Relations with MMAX. In *Proceedings of the 2nd SIGdial Workshop on Discourse and Dialogue*, pages 90–95, Aalborg, Denmark.

O'Donnell, M. (1997). RST-Tool: An RST Analysis Tool. In *Proceedings of the 6th European Workshop on Natural Language Generation*, Duisburg, Germany.

Orăsan, C. (2000). CLinkA a Coreferential Links Annotator. In *Proceedings of the Second International Conference on Language Resources and Evaluation*, pages 491–496. LREC.

Orăsan, C. (2003). PALinkA: A highly customisable tool for discourse annotation. In *Proceedings of the Fourth SIGdial Workshop on Discourse and Dialogue*, Sapporo, Japan.

Poesio, M. (2004). The MATE/GNOME Scheme for Anaphoric Annotation, Revisited. In *Proceedings of SIGDIAL*, Boston.

Poesio, M. and Kruschwitz, U. (2008). ANAWIKI: Creating anaphorically annotated resources through web cooperation. Submitted to LREC 2008.

Renear, A., Mylonas, E., and Durand, D. (1996). Refining our notion of what text really is: The problem of overlapping hierarchies. *Research in Humanities Computing. Selected Papers from the ALLC/ACH Conference, Christ Church, Oxford, April 1992*, 4:263–280.

Sidner, C. (1979). *Towards a computational theory of definite anaphora comprehension in English discourse*. PhD thesis, MIT.

Simons, G. and Bird, S. (2003). *OLAC Metadata*. OLAC: Open Language Archives Community.

Simons, G., Lewis, W., Farrar, S., Langendoen, T., Fitzsimons, B., and Gonzalez, H. (2004). The Semantics of Markup. In *Proceedings of the ACL 2004 Workshop on RDF/RDFS and OWL in Language Technology (NLPXML-2004)*, Barcelona.

Sperberg-McQueen, C. and Burnard, L., editors (2002). *TEI P4: Guidelines for Electronic Text Encoding and Interchange*. published for the TEI Consortium by Humanities Computing Unit, University of Oxford, Oxford, Providence, Charlottesville, Bergen.

Stührenberg, M. and Goecke, D. (2008). SGF – an integrated model for multiple annotations and its application in a linguistic domain. In *Proceedings of Balisage: The Markup Conference*.

Stührenberg, M., Goecke, D., Diewald, N., Cramer, I., and Mehler, A. (2007). Web-based Annotation of Anaphoric Relations and Lexical Chains. In *Proceedings of the Linguistic Annotation Workshop (LAW)*, pages 140–147, Prag. Association for Computational Linguistics.

Stührenberg, M., Witt, A., Goecke, D., Metzing, D., and Schonefeld, O. (2006). Multidimensional Markup and Heterogeneous Linguistic Resources. In *Proceedings of the 5th Workshop on NLP and XML (NLPXML-2006): Multi-Dimensional Markup in Natural Language Processing*, pages 85–88.

Teich, E. and Eckart, R. (2007). An XML-based data model for flexible representation and query of linguistically interpreted corpora. In *Datenstrukturen für linguistische Ressourcen und ihre Anwendungen. Data Structures for Linguistic Resources and Applications. Proceedings of the Biennial GLDV Conference 2007*, Tübingen.

Thompson, H. S., Beech, D., Maloney, M., and Mendelsohn, N. (2004). XML Schema Part 1: Structures (2nd Edition). W3C Recommendation, World Wide Web Consortium.

Thompson, H. S. and McKelvie, D. (1997). Hyperlink semantics for standoff markup of read-only documents. In *Proceedings of SGML Europe '97: The next decade – Pushing the Envelope*, pages 227–229, Barcelona.

Vieira, R. and Poesio, M. (2001). An empirically based system for processing definite descriptions. *Computational Linguistics*, 26(4):539–593.

Witt, A. (2002). *Multiple Informationsstrukturierung mit Auszeichnungssprachen. XMLbasierte Methoden und deren Nutzen für die Sprachtechnologie*. Dissertation, Universität Bielefeld.

Witt, A., Goecke, D., Sasaki, F., and Lüngen, H. (2005). Unification of XML Documents with Concurrent Markup. *Literary and Lingustic Computing*, 20(1):103–116.

Witt, A., Schonefeld, O., Rehm, G., Khoo, J., and Evang, K. (2007). On the Lossless Transformation of Single-File, Multi-Layer Annotations into Multi-Rooted Trees. In *Proceedings of Extreme Markup Languages*, Montréal, Québec.

Wörner, K., Witt, A., Rehm, G., and Dipper, S. (2006). Modelling Linguistic Data Structures. In *Proceedings of Extreme Markup Languages*, Montréal, Québec.