Die Document Suite - XML-basierte Sprachverarbeitung als Basistechnologie für das 'Semantic Web'

Dietmar Rösner, Manuela Kunze
Otto-von-Guericke-Universität Magdeburg
Institut für Wissens- und Sprachverarbeitung
P.O.box 4120,
39106 Magdeburg, Germany
{*roesner, makunze*}*@iws.cs.uni-magdeburg.de*

Abstract: Die 'Document Suite' ist eine Sammlung von Werkzeugen für die verschiedenen Aufgaben beim Erschließen der Inhalte von elektronisch verfügbaren Dokumenten aus dem WWW oder aus anderen Dokumentbeständen. Bei ihrer Konzeption und Realisierung wurde konsequent darauf gesetzt, die Vorteile von XML und die der zugehörigen Formalismen und Werkzeuge auszunutzen: alle Module der 'Document Suite' erwarten XML-Dokumente als Eingabe und liefern ihre Resultate in XML-Format. Desweiteren werden alle Ressourcen (z.B. Lexika, Grammatikregeln, semantische Zuordnungen, Topic Maps, ..) einheitlich in XML kodiert.

1 Einleitung

In der Arbeitsgruppe 'Wissensbasierte Systeme und Dokumentverarbeitung' ist in den vergangenen Jahren – u.a. durch Arbeiten im Rahmen der DFG-Forschergruppe 'Workbench für die Informationsfusion'[DFG02] – die 'Document Suite' entstanden, eine Sammlung von Werkzeugen für die verschiedenen Aufgaben beim Erschließen der Inhalte von elektronisch verfügbaren Dokumenten [KR01a, KR01b].

Bei der Konzeption und Realisierung der 'Document Suite' wurde konsequent darauf gesetzt, die Vorteile von XML und die der zugehörigen Formalismen (z.B. XSLT [Sit02b]) und Werkzeuge (z.B. xt [Cla02]) auszunutzen: XML dient als vereinheitlichender Rahmen. So erwarten alle Module der 'Document Suite' XML-Dokumente als Eingabe und liefern ihre Resultate in XML-Format. Desweiteren werden alle Ressourcen (z.B. Lexika, Grammatikregeln, semantische Zuordnungen, Taxonomien, Topic Maps, ..) einheitlich in XML kodiert.

In diesem Beitrag geben wir einen Gesamtüberblick über die 'Document Suite' und ihre Funktionen, stellen dann einzelne Module vor und schließen mit einem Ausblick auf geplante nächste Schritte.

2 Sprachverarbeitung für das 'Semantic Web'

Das derzeit im WWW vorherrschende Markup ist HTML, semantisch und pragmatisch explizit ausgezeichnete Dokumente sind noch selten. Es ist daher lohnenswert, Werkzeuge zu entwickeln, die unausgezeichnete bzw. nur mit HTML ausgezeichnete Textdokumente analysieren und automatisch mit inhaltsbezogenem Markup anreichern können. Dadurch können solche Dokumente dann eine Rolle in einem 'Semantic Web' spielen.

Die entwickelten Techniken werden auch nützlich bleiben, wenn der Anteil an Dokumenten im Web zunehmen wird, die bereits bei der Erstellung explizites Markup erhalten. Es ist zu erwarten, dass die Granularität der XML-Elemente bei solchem Markup eher so sein wird, dass ganze Paragraphen oder Absätze von Fliesstext als Zeicheninhalt enthalten sein werden, eine durchgängige Auszeichnung grosser Dokumente bis zur Wort- bzw. Token-Ebene ist ohne Einsatz automatischer Verfahren zur Textanalyse und zur automatischen Annotation mit Markup praktisch nicht realisierbar. XDOC wird also für die Feinanalyse des Zeicheninhalts von XML-Elementen auch dann von Bedeutung bleiben, wenn das 'Semantic Web' – wie zu hoffen – zunehmend die derzeitige Generation des WWW ablösen wird.

3 Entwurfsprinzipien

Einige der Kriterien und Anforderungen, an denen sich die Arbeiten an der 'Document Suite' orientierten, legen die Verwendung von XML und seinen begleitenden Formalismen und Werkzeugen fast zwingend nahe. Andere Anforderungen sind zwar unabhängig von XML realisierbar, ihre Realisierung profitiert aber indirekt von der konsequenten Verwendung von XML, weil die dadurch frei gewordene Kapazität (z.B. durch die Nutzung von XML-Werkzeugen für viele Aufgaben) für die Lösung anderer Aufgaben eingesetzt werden kann (z.B. für Lernaufgaben und Aufbau und Nutzung von Wissensstrukturen).

Die Arbeit an der 'Document Suite' orientiert sich u.a. an folgenden Entwurfszielen:

- Die Werkzeuge der 'Document Suite' sollen auf alltägliche, elektronisch (im WWW, aber auch in anderen Formaten) vorliegende Dokumente anwendbar sein.
- Die Werkzeuge sollen so robust wie möglich sein und mit unvollständiger Vorinformation ebenso wie mit fehlerbehafteten Eingaben umgehen können.
- Die einzelnen Werkzeuge sollen einerseits unabhängig verwendbar sein, sich andererseits aber flexibel kombinieren lassen.
  Die konsequente Verwendung von XML als einheitliches Format sowohl für die Quelldokumente (nach Vorverarbeitung), für alle Arten von Verarbeitungsergebnissen wie für alle Arten von Ressourcen (Lexika, Grammatiken, ...) ist die Basis für die erzielte Interoperabilität.
- Die Werkzeuge sollen nicht nur von den Entwicklern, sondern auch von Gebietsexperten ohne linguistische oder informatische Ausbildung nutzbar sein.

Auch dies läßt sich besonders gut mit XML und XSLT erreichen: XSL Stylesheets werden genutzt, um für unterschiedliche Zielgruppen unterschiedliche Präsentationen der systeminternen Daten und der Ergebnisse zu erzeugen; während Endbenutzer mit den internen Strukturen wenig anfangen können, sind diese für die Entwickler beim Debugging eine wichtige Hilfe.

4 Funktionen der 'Document Suite'

Klassifiziert man die verschiedenen Module der 'Document Suite' nach ihren Aufgaben, so erweist sich die folgende Einteilung als sinnvoll:

- Vorverarbeitung
- Strukturdetektion
- Zuordnung von Wortklassen (POS tagging)
- Syntaktisches Parsing
- Semantische Analyse
- Werkzeuge für die jeweilige Anwendung: z.B. Informationsextraktion, Wissensakquisition

4.1 Vorverarbeitung

Mit den Werkzeugen zur Vorverarbeitung werden Dokumente aus verschiedenen Quellen (neben WWW auch andere elektronisch verfügbare Dokumentbestände) und in verschiedenen Ausgangsformaten in das für die weitere Verarbeitung erforderliche XML-Format gebracht. Als eine Teilaufgabe gehört dazu auch die Behandlung von Sonderzeichen (für z.B. Umlaute, Apostrophe, ...).

Dieser Schritt entfällt, wenn das Dokument bereits in XML vorliegt.

4.2 Strukturdetektion

Liegt ein Dokument vor, bei dem die Struktur nur implizit vorhanden ist (wie es u.a. im WWW bei vielen nur in HTML ausgezeichneten Dokumenten der Fall ist), so ist es erforderlich, zumindest solche Strukturen zu detektieren und explizit zu machen, die für die weitere Verarbeitung wichtig sind.

So ist es z.B. für die linguistische Verarbeitung relevant, ob eine Einheit in einem Dokument ein Titel oder Untertitel sein kann. In einem solchen Fall wird der Parser auch

eine Nominalphrase akzeptieren (z.B. 'Technologien für das Semantic Web'), während er in einem Element vom strukturellen Typ 'Paragraph mit Fließtext' ganze Sätze erwarten wird.

In realistischen Texten ist es – anders als bei linguistischen Lehrbuch-Beispielen – nun aber nicht trivial, Satzgrenzen zu erkennen. Zwar helfen Interpunktionszeichen, aber ein Punkt kann eben nicht nur das Satzende markieren, sondern auch Teil einer Abkürzung sein (z.B. in 'z.B.' oder 'Dr.'), zu einer Zahl gehören (3.14) oder einer email-Adresse oder einem anderen gebietsspezifischen Bezeichner (z.B. Materialbezeichner, Enzymname, ...).

Als Ressourcen werden bei der Strukturdetektion daher ein Abkürzungsverzeichnis sowie verschiedene endliche Automaten für gebietsspezifische Bezeichner eingesetzt.

Beispiel 1 *Ergebnis einer Strukturdetektion*

```
Anwesend<IP>:</IP> <ABBR>Univ.-Prof.</ABBR>
<ABBR>Dr.</ABBR><ABBR>med.</ABBR> Dieter Krause<IP>, </IP>
Direktor des Institutes fuer Rechtsmedizin
```

(Bem.: mit IP werden Interpunktionen gekennzeichnet, mit ABBR Abkürzungen.)

Das folgende Beispiel ist ein Auszug aus dem genutzten XML-basierten Abkürzungsverzeichnis, zu jeder Abkürzung wird das vollständige Wort angegeben.

Beispiel 2 *Ausschnitt aus dem XML-basierten Abkürzungsverzeichnis*

```
<ABBREVIATION>
    <ABBREV>Abb.</ABBREV>
    <FULL>Abbildung</FULL>
</ABBREVIATION>
<ABBREVIATION>
    <ABBREV>Dr.</ABBREV>
    <FULL>Doktor</FULL>
</ABBREVIATION>
```

4.3 Zuordnung der Wortklassen

Als Vorstufe zur weiteren syntaktischen und semantischen Verarbeitung von textuellen Dokumenten ist es sinnvoll, den vorgefundenen, zusammenhängenden Zeichenketten – den Token – zunächst Wortklassen zuzuordnen (sog. part-of-speech- oder POS-Tagging). Dabei kann es sich sowohl um 'klassische' Wortklassen aus der Linguistik handeln (z.B. Nomen, Verb, Adjektiv, Artikel, Präposition und andere sog. lexikalische Kategorien) wie auch um nicht-lexikalische Kategorien, die dann meist gebietsspezifische Relevanz haben (z.B. Telefonnummern, email-Adressen, Materialkennungen, Substanzbezeichner, ...).

Das Beispiel 3 beinhaltet das Tagger-Ergebnis für den gebietsspezifischen Produktbezeichner 'Gussstueck EN 1982-CC333G-GS-XXXX'. Innerhalb des *PRODUCT*-Tags wird

die Produktionsmethode und das Material explizit angegeben. Desweiteren setzt sich dieser Tag aus den Elementen *NORM* (bestehend aus der Normart und einer Jahreszahl), *MAT-ID* (Kennzeichnung des Materials), *METHODE* (die Kurzform der Produktionsmethode) und *MODELLNR* zusammen.

Beispiel 3 *gebietsspezifische nonlexikalische Kategorie*

```
<PRODUCT Method="Sandguss" Material="CC333G">
    <N>Gussstueck</N>
    <NORM>
         <N>EN</N>
         <NR>1982</NR>
    </NORM>
    <IP>-</IP>
    <MAT-ID>CC333G</MAT-ID>
    <IP>-</IP>
    <METHODE>GS</METHODE>
    <IP>-</IP>
    <MODELLNR>XXXX</MODELLNR>
</PRODUCT>
```

Für das POS-Tagging deutscher Texte verwenden wir die Morphologiekomponente MORPHIX [FN88]. Diese Komponente zeichnet sich dadurch aus, dass die *geschlossenen* Wortklassen des Deutschen (d.h. Artikel, Präpositionen, Pronomen, Konjunktionen, usw.) sowie alle unregelmäßigen Verben in allen ihren Wortformen vollständig abgedeckt sind. Für die *offenen* Wortklassen (also Nomen, zusammengesetzte und regelmäßige Verben, Adjektive, Adverben) sind die Muster der möglichen Formen abgedeckt (linguistisch: Flexionsparadigmen). Um eine als Token auftretende Wortform aber korrekt analysieren zu können, muß ein Eintrag aus Grundform und Angaben u.a. zur Flexionsklasse im Lexikon von MORPHIX existieren. Zur Zeit beinhaltet das Lexikon für die MORPHIX-Komponente 9300 Einträge.

Die XDOC-Architektur macht es leicht, auch einen anderen Tagger für Deutsch einzubinden. So stehen verschiedene Anpassungen eines Brill-Tagger [Bri92] auch für Deutsch zur Verfügung [SK96]. In Subsprachen lassen sich diese aber nur mit gutem Ergebnis einsetzen, wenn sie mit ausreichend viel manuell aufbereiteten Material trainiert wurden. Diesen Aufwand wollten wir Nutzern von XDOC nicht aufbürden. Ein weiterer Nachteil der Brill-Tagger ist, dass Sie nur die aufgrund des verwendeten Hidden-Markov-Modell wahrscheinlichste POS-Klasse zurückgeben und nicht alle Alternativen.

Wenn im Tagger von XDOC ein Token in einem deutschen Text mit MORPHIX analysiert werden kann, wird die ermittelte Wortklasse als POS-Information verwendet. Diese Klassifikation ist, da sie nur das Token, aber noch keinen Kontext betrachtet, nicht immer eindeutig. Einige Beispiele: das Token ‘der’ kann sowohl ein Artikel sein (mit entsprechender Kombination der Werte für die Merkmale Kasus, Numerus und Genus) oder ein Relativpronomen, das Token ‘liebe’ kann eine Verbform zum Infinitiv ’lieben’ oder ein Adjektiv sein. Da nicht erwartet werden kann, dass das Lexikon jemals vollständig ist (z.B. Neologismen, produktive Kompositabildung, Fachtermini), muss mit der Situation umgegangen werden können, dass für einige Token keine MORPHIX-Analyse gefunden wird.

Hier werden zwei Techniken eingesetzt: Zunächst wird versucht, Heuristiken anzuwenden, die sich auf solche Aspekte des Token stützen, die mit Stringanalyse einfach festgestellt werden können (z.B. Gross-/Kleinschreibung, Endungen, ...), und welche zusätzlich ggf. die relative Position des Token in Bezug auf eine Satzgrenze mit einbeziehen. Führt dies zu einer Klassifikation, so wird das Token mit dem Bezeichner der zugehörigen POS-Klasse markiert, die verwendete Heuristik wird als Wert des Attribut SRC angegeben (vgl. Beispiel 4). Wenn keine Heuristik anwendbar ist, wird das Token als Element der Klasse 'unbekannt' eingestuft (Tag XXX).

Um den POS-Tagger schnell und einfach zu halten, wird die Disambiguierung zwischen mehreren möglichen POS-Klassen für ein Token und das Ableiten einer möglichen POS-Klasse für ein unbekanntes Token aufgrund von Kontext auf die syntaktische Analyse verlagert. Damit folgen wir dem Prinzip, dass bei der isolierten Anwendung von Modulen (hier: POS-Tagging) Resultate mit sog. Übergenerierung dann akzeptiert werden können, wenn in nachfolgenden Schritten mehrdeutige Ergebnisse gefiltert werden (hier: durch den syntaktischen Kontext).

Beispiel 4 *unbekannte Token mit Heuristik als Nomen klassifiziert*

```
<NP TYPE="COMPLEX" RULE="NPC3" GEN="FEM" NUM="PL" CAS="_">
  <NP TYPE="FULL" RULE="NP1" CAS="_" NUM="PL" GEN="FEM">
       <N SRC="UNG">Blutanhaftungen</N>
  </NP>
  <PP CAS="DAT">
    <PRP CAS="DAT">an</PRP>
    <NP TYPE="FULL" RULE="NP2" CAS="DAT" NUM="SG" GEN="FEM">
      <DETD>der</DETD>
      <N SRC="UC1">Gekroesewurzel</N>
    </NP>
  </PP>
</NP>
```

4.4 Syntaktisches Parsing

Für das syntaktische Parsing wird ein Chart-Parser verwendet, der auf Grammatiken arbeitet, die kontextfreie Regeln enthalten, die mit Merkmalsstrukturen annotiert sind. Der Vorteil eines Chart-Parser ist, dass die Mehrfachanalyse erfolgreich verarbeiteter Teilstrukturen vermieden wird und alle – auch partiellen – Analyseergebnisse in einer kompakten Datenstruktur erhalten bleiben (siehe z.B. [GM89]). Für XDOC wurde eine Version eines Chart-Parser implementiert, der mit einer Bottom-up-Strategie arbeitet. Im Beispiel 5 ist die XML-Darstellung der Grammatikregel abgebildet, die im Beispiel 4 zur Anwendung kam. Die Attribute des Tags *RULE* beinhalten die Bezeichnung der Regel, die Kategorie der Regel (z.B. NP für *noun phrase* oder S für *sentence*) und strukturspezifische Informationen (Art der Struktur). Nach dem *RULE*-Tag folgt die Auflistung der Elemente der rechten Seite der Regel. Der *ELEMENT*-Tag kann einen Operator für das Element beinhalten (z. B. optional) sowie auch Literale (z.B. '?' für Fragesätze). Unter *EXAMPLE* wird

ein typisches Beispiel für die Regel angegeben.

Beispiel 5 *Darstellung einer Grammatikregel in XML*

```
<RULE RULE-CAT="NP" TYPE="COMPLEX" RULE="NPC3">
        <ELEMENT TYPE="FULL">NP</ELEMENT>
        <ELEMENT OP="*">PP</ELEMENT>
        <ELEMENT>PP</ELEMENT>
        <EXAMPLE>
            <STRING>die Katze auf dem Baum im Garten</STRING>
        </EXAMPLE>
</RULE>
```

Auch bei 'lexikalischen Lücken' soll die Verarbeitung in der 'Document Suite' robust sein. Daher werden nicht nur eindeutig und komplett spezifizierte Token, sondern auch die folgenden Varianten als Eingaben für den syntaktischen Parser zugelassen:

- Token mit mehreren möglichen POS-Klassen,
- Token ohne POS-Klasse (d.h. Klasse 'unbekannt' mit Tag 'XXX'),
- Token mit POS-Klasse, aber ohne oder mit nur partieller Merkmalsinformation.

Der letzte Fall kann auftreten, da einige der beim POS-Tagging verwendeten Heuristiken zwar erlauben, die Wortklasse zu bestimmen (z.B. Nomen), aber nicht ausreichen, um das Flexionsparadigma aus dem einzelnen Token zu bestimmen (man beachte, das es ca. zwei Dutzend verschiedene Flexionsparadigmen für die Deklination von Nomen im Deutschen gibt).

Für jede Eingabe versucht der syntaktische Parser alle vollständigen Analysen zu finden, welche die jeweilige Eingabe überspannen. Kann er keine solche komplette Analyse erreichen, wird versucht, maximale partielle Resultate zu Strukturen zu kombinieren, die dann die gesamte Eingabe überspannen.

Eine erfolgreiche Analyse stützt sich möglicherweise auf eine Hypothese über die Wortklasse eines anfänglich noch unklassifizierten Token (mit Tag XXX). In der XML-Repräsentation des Parsingergebnis wird die Hypothese angezeigt (Merkmal AS) und kann ausgenutzt werden, solche Klassifikationen aufgrund kontextueller Information zu lernen.

Beispiel 6 *durch kontextuelle Bedingungen unbekanntes Token als Adjektiv klassifiziert und Merkmale abgeleitet*

```
<NP TYPE="COMPLEX" RULE="NPC3" GEN="MAS" NUM="SG" CAS="NOM">
  <NP TYPE="FULL" RULE="NP3" CAS="NOM" NUM="SG" GEN="MAS">
    <DETI>kein</DETI>
    <XXX AS="ADJ">ungehoeriger</XXX>
    <N>Inhalt</N>
  </NP>
  <PP CAS="DAT">
```

```
      <PRP CAS="DAT">in</PRP>
      <NP TYPE="FULL" RULE="NP2" CAS="DAT" NUM="SG" GEN="FEM">
        <DETD>der</DETD>
        <N SRC="UC1">Mundhoehle</N>
      </NP>
    </PP>
  </NP>"
```

Auf eine ähnliche Weise kann aufgrund von Kongruenzbedingungen (Beispiel: in einer Nominalphrase müssen z.B. Artikel, eventuell vorhandenes Adjektiv und Nomen in den Werten der Merkmale Kasus, Numerus und Genus übereinstimmen) aus den bekannten Merkmalswerten für Token aus geschlossenen Wortklassen (z.B. Artikel, Präpositionen) auf unterspezifizierte Merkmale von Token aus offenen Wortklassen (z.B. Adjektiv, Nomen) geschlossen werden. So läßt sich, wenn entsprechende Wortformen vorliegen, aus erfolgreichen syntaktischen Analysen das Flexionsparadigma vorher unbekannter Wörter aus den kontextuellen Restriktionen ableiten. Im Beispiel konnten etwa die Merkmalswerte für das unbekannte Wort 'Mundhoehle' (u.a. der Wert FEM für Genus) aus den Merkmalen des Artikels in Verbindung mit den Kasusrestriktionen der Präposition 'in' abgeleitet werden.

Die in der 'Document Suite' verwendeten Grammatiken sind modular organisiert. Gruppen von Regeln können flexibel aktiviert und deaktiviert werden. Dies wird ausgenutzt, um den Konventionen unterschiedlicher Subsprachen Rechnung tragen zu können. Je nach Gebiet können Subsprachen syntaktische Konstruktionen präferieren, die in der Standardsprache ungewöhnlich oder gar ungrammatisch sind.

4.5 Semantische Analyse

Innerhalb der semantischen Analyse wird zunächst auf Tokenebene den Token eine Bedeutung zugeordnet und diese dann über die Strukturanalyse genauer spezifiziert. Damit ist es durch weitere Analysen möglich, komplexere Inhalte aus den Dokumenten zu erschließen.

Die bisher beschriebenen Module sind Hilfsmittel für die zentrale Aufgabe der semantischen Analyse und die je nach Aufgabenstellung sich daran anschließende Weiternutzung der erschlossenen Inhalte.

Derzeit werden hauptsächlich drei Techniken bei der semantischen Analyse verwendet.

Semantisches Taggen

Für das semantische Taggen wurde ein semantisches Lexikon erstellt (derzeit ca. 700 Einträge), in welchem die Semantik eines Wortes (s. Beispiel 7) und sofern erforderlich, der Kasusrahmen für Nomen und Verben (demnächst auch für Präpositionen), beschrieben wurde.

Bei der Erstellung des Lexikons wurde überlegt, die Ressourcen von GermaNet [GPS02]

einzubinden. Da aber nur teilweise die fachspezifischen Begriffe in GermaNet abgedeckt wurden und Kasusrahmeninformation für Nomen nicht unterstützt werden, wurden eigene Ressourcen angelegt. Neben den Kasusrahmen wurden auch Informationen zu Part-of-Relationen in die semantische Analyse integriert.

Beispiel 7 *Auszug aus dem semantischen Lexikon für Adjektive*

```
<PROPERTY>
    <WORD>dunkelrot</WORD>
    <CATEGORY>color</CATEGORY>
</PROPERTY>
```

Der Kasusrahmen beschreibt mögliche Valenzen eines Tokens mit den jeweiligen semantischen und syntaktischen Formen. Im Beispiel 8 ist der Kasusrahmen für das Wort *Fertigen* zu sehen. Der Tag *ROLE* beschreibt die Semantik des Wortes, während über die Tags *RELATION* mögliche Valenzen (mit ihrer syntaktischen und semantischen Form) beschrieben werden.

Beispiel 8 *Auszug aus dem semantischen Lexikon für Nomen*

```
<CASE-FRAME>
    <WORD>fertigen</WORD>
    <DESCRIPTION>produzieren</DESCRIPTION>
    <ROLE>process</ROLE>
    <RELATIONS>
        <RELATION>source</RELATION>
        <SEMANTIC>material</SEMANTIC>
        <SYNTACTIC>
             <TYPE cas="dat" prp="aus">PP</TYPE>
        </SYNTACTIC>
    </RELATIONS>
    <RELATIONS>
        <RELATION>result</RELATION>
        <SEMANTIC>object</SEMANTIC>
        <SYNTACTIC>
            <TYPE cas="gen">NP</TYPE>
            <TYPE cas="acc" prp="von">PP</TYPE>
        </SYNTACTIC>
    </RELATIONS>
</CASE-FRAME>
```

Wie bei dem morphologischen Taggen ist es auch hier möglich, mehrere Interpretationen eines Tokens zu erhalten. Diese Ambiguitäten werden durch die nachfolgende Analyse aufgelöst. Die Auszeichnung der Ergebnisse erfolgt ebenfalls in der XML-Notation.

Beispiel 9 *Ergebnisse des semantischen Taggers*

```
<MULT>Fertigen</MULT>
```

```
<PROPERTY TYPE="zustand">fester</PROPERTY>
<CONCEPT TYPE="object">Koerper</CONCEPT>
<MULT>aus</MULT>
<PROPERTY>formlosem</PROPERTY>
<CONCEPT TYPE="object material">Stoff</CONCEPT>
```

Analyse von Kasusrahmen

Mit Hilfe der im Kasusrahmen beschriebenen Valenzen wird in dem zu analysierenden Text nach verknüpfbaren Konzepten gesucht. So liefert uns die Auswertung der Phrase *Fertigen fester Koerper aus formlosem Stoff* die im Beispiel 10 dargestellte Ergebnis-Struktur.

Fertigen wurde als Prozess (Attribut *TYPE*) erkannt und das Ergebnis des Prozesses (Relation *RESULT*) sind *feste Koerper*. Das Ausgangsmaterial für den Prozess (Relation *SOURCE*) wird durch die Präpositionalphrase *aus formlosem Stoff* beschrieben. Die syntaktischen Valenzen wurden durch den Tag *FORM* beschrieben. So könnte es sich bei der Relation *RESULT* syntaktisch gesehen um eine Nominalphrase im Genitiv oder um eine Präpositionalphrase im Akkusativ mit der Präposition *von* handeln.

Beispiel 10 *Analyse des Kasusrahmen*

```
<CONCEPT TYPE="process">
    <WORD>Fertigen</WORD>
    <DESC>Schaffung von etwas</DESC>
    <SLOTS>
        <RELATION TYPE="RESULT">
            <ASSIGN_TO>OBJECT</ASSIGN_TO>
            <FORM>N(gen, fak) P(akk, fak, von)</FORM>
            <CONTENT>fester Koerper</CONTENT>
        </RELATION>

        <RELATION TYPE="SOURCE">
            <ASSIGN_TO>MATERIAL</ASSIGN_TO>
            <FORM>P(dat, fak, aus)</FORM>
            <CONTENT>aus formlosem Stoff</CONTENT>
        </RELATION>

        <RELATION TYPE="INSTRUMENT">
            <ASSIGN_TO>PROCESS</ASSIGN_TO>
            <FORM>P(akk, fak, durch)</FORM>
            <CONTENT></CONTENT>
        </RELATION>
    </SLOTS>
</CONCEPT>
```

Analyse spezifischer syntaktischer Strukturen

Bei dieser Methode wird der Fokus auf spezifische syntaktische Strukturen gesetzt. In der Subsprache der forensischen Medizin findet man häufig Strukturen wie z.B. *Leber dunkelrot*. Diese Strukturen sind innerhalb der deutschen Sprache nicht typisch und die verwendete Grammatik mußte um diese subsprachenabhängigen Strukturen erweitert werden. Semantisch können wir diese und andere spezifische syntaktische Strukturen auswerten.

Beispiel 11 *Beschreibung der syntaktischen Formen der Relation* has

```
<RELATION>
    <TYPE>has</TYPE>
    <DESC>has_attribute</DESC>
    <FILLER>ADJ</FILLER>
    <PRESENTATION>
        <ELEMENT>N</ELEMENT>
        <ELEMENT>ADJ</ELEMENT>
    </PRESENTATION>
    <SYNTACTIC>
        <STRUCTURE>NP1</STRUCTURE>
        <STRUCTURE>NP2</STRUCTURE>
        <STRUCTURE>NP3</STRUCTURE>
        <STRUCTURE>MA1</STRUCTURE>
    </SYNTACTIC>
</RELATION>
```

Die im Beispiel 11 vorgestellte Datenstruktur beschreibt die semantische Interpretation der syntaktischen Strukturen NP1, NP2, NP3 (Regeln für Varianten von Nominalphrasen) und MA1 (Regel für die 'medical-attribute'-Struktur), festgelegt über den Tag *SYNTACTIC*, als *has*-Relation. Der Tag *TYPE* spezifiziert die generalisierte Relation, die abgeleitete Relation ergibt sich aus der übergeordneten Relation und der Semantik des Tokens, das über den Tag *FILLER* definiert wurde. Die Reihenfolge der Attribute des Prädikats ergibt sich aus der Reihenfolge der Elemente aus dem Tag *PRESENTATION*.

Die Analyse des oben genannten Beispiels liefert uns folgendes Ergebnis, das als Topic Map [Spe01] mit XML kodiert ist:

Beispiel 12 *Auswertung der syntaktischen Struktur* Leber dunkelrot

```
has_color(Leber,dunkelrot)
<topicmap>
    <topic id="Leber">
        <instanceof><topicRef xlink:href="#organ"></instanceof>
        <basename>
            <basenamestring>Leber</basenamestring>
        </basename>
    </topic>

    <topic id="dunkelrot">
        <instanceof><topicRef xlink:href="#color"></instanceof>
```

```
        <basename>
            <basenamestring>dunkelrot</basenamestring>
        </basename>
    </topic>

    <association>
        <instanceof><topicRef xlink:href="#has_color"></instanceof>
        <member>
            <rolespec><topicRef xlink:href="#organ"></rolespec>
            <topicref xlink:href="#Leber">
        </member>
        <member>
            <rolespec><topicRef xlink:href="#color"></rolespec>
            <topicref xlink:href="#dunkelrot">
        </member>
    </association>
</topicmap>
```

Als Ergebnis erhalten wir das Prädikat *has-color*. Der Name des Prädikates setzt sich aus dem übergeordneten Relationstyp *has* und der Kategorie des Adjektivs zusammen. Die Reihenfolge der Attribute des Prädikats wurde innerhalb der Ressourcen für diese Analyse festgelegt. Dasselbe Resultat wird auch als Topic Map ausgezeichnet. Innerhalb der Topic Map wurden zwei Topics definiert (*Leber* und *dunkelrot*) und durch den *association*-Tag wird die Relation *has-color* beschrieben.
Die Notation der Ergebnisse kann jederzeit entsprechend dem Anwendungsgebiet angepasst werden, denkbar wäre auch ein KIF-basierter Ansatz bzw. auch ein RDF-basierter Ansatz [MS99] (in [LD01] wurde die Integration von Topic Maps in RDF beschrieben).

Diese Methode kann auch auf andere interessante Strukturen ausgeweitet werden, z.B. Nominalphrasen bestehend aus Adjektiv und Nomen oder auch komplexere Strukturen, wie z.B. den Gleichstellungsnominativ (*Urformen ist Fertigen fester Koerper aus formlosem Stoff.*). In dem Beispiel 10 zur Kasusrahmenanalyse wurde die Phrase *fester Koerper* als die linguistische Struktur des Rollenfüllers zur Relation *RESULT* übernommen. Durch die Analyse der syntaktischen Struktur erhalten wir die Beschreibung der Phrase z.B. als Topic Map.

Beispiel 13 *semantische Auswertung der Phrase* fester Koerper

```
<topicmap>
    <topic id="Koerper">
        <instanceof><topicRef xlink:href="#object"></instanceof>
        <basename>
            <basenamestring>Koerper</basenamestring>
        </basename>
    </topic>

    <topic id="fest">
        <instanceof><topicRef xlink:href="#zustand"></instanceof>
        <basename>
            <basenamestring>fest</basenamestring>
        </basename>
    </topic>
```

```
    <association>
        <instanceof><topicRef xlink:href="#has_zustand"></instanceof>
        <member>
            <rolespec><topicRef xlink:href="#object"></rolespec>
            <topicref xlink:href="#Koerper">
        </member>
        <member>
            <rolespec><topicRef xlink:href="#zustand"></rolespec>
            <topicref xlink:href="#fest">
        </member>
    </association>
</topicmap>
```

Der nächste Schritt wird die Definition eines einheitlichen Formats für die Präsentation (z.B. RDF) der Ergebnisse der semantischen Analysen sein und die Verschmelzung der beiden letztgenannten Techniken.

Die Wahl des semantischen Repräsentationsformalismus erfolgt anwendungsunabhängig. Da die Dokumentensuite verschiedene Aufgabenstellungen erfüllen soll, wird ein internes, auf XML basierendes Format gewählt, so daß es über Transformationswerkzeuge möglich ist, die anwendungsspezifische Darstellung der Ergebnisse zu generieren.

5 Verwandte Arbeiten

Die Arbeiten an der 'Document Suite' konnten von verwandten Projekten profitieren, bei denen aber englische Dokumente im Zentrum standen:

Bei GATE [Sit02a, CW88] wurde zum ersten Mal die Idee des Hintereinanderschaltens ('piping') einfacher Module, um so komplexe Funktionalität zu realisieren, in konsequenter Weise auf die Dokumentverarbeitung mit linguistischen Mitteln angewendet. Das Projekt LT XML [Gro99] hat Pionierarbeit mit XML als Datenformat für die linguistische Verarbeitung geleistet.

SMES [NBB+97] ist als Umgebung für das Verarbeiten deutschsprachiger Texte konzipiert und implementiert worden. Ein Nachteil von SMES war, dass kein einheitlicher Formalismus verwendet wurde und die Benutzer daher mit unterschiedlichen Repräsentationsformaten in den verschiedenen Modulen konfrontiert waren.

6 Zusammenfassung

Die 'Document Suite' wird aktuell angewendet bei Arbeiten zur

- Wissensakquisition aus technischer Dokumentation über Gießereitechnologie,
- semantischen Analyse von Obduktionsprotokollen,
- Extraktion von Firmenprofilen aus WWW-Seiten.

Bei den Arbeiten an und mit der ‘Document Suite’ werden auf der Basis von XML als vereinheitlichendem Rahmen Techniken aus Dokumentverarbeitung, Computerlinguistik und Wissensverarbeitung für die verschiedenen Teilaufgaben bei einer inhaltsorientierten Verarbeitung von Dokumenten kombiniert. Werkzeuge dieser Art sind derzeit bei vorwiegend nur HTML-strukturierten Web-Dokumenten eine unverzichtbare Basistechnologie für das ‘semantic web’. Auch wenn zukünftig vermehrt bereits in XML strukturierte WWW-Dokumente vorliegen werden, werden Techniken der Analyse von Texten weiterhin erforderlich sein, da auf absehbare Zeit die Strukturierung eher die Makrostruktur auszeichnen wird und inhaltliche Analysen innerhalb von Textblöcken daher weiterhin erforderlich bleiben dürften.

Durch die explizite Trennung der Werkzeuge und Ressourcen ist es möglich, einige der hier vorgestellten Funktionen z.B. auch für Analysen englischsprachiger Dokumente dadurch zu nutzen, daß lediglich Ressourcen ausgetauscht werden. Die XML-Auszeichnung der Ressourcen erleichtert einen solchen sprachabhängigen Austausch [KX02].

Verfügbarkeit der Document Suite

Die Funktionalität der Document suite steht für interessierte Benutzer für Testzwecke zur Verfügung unter **http://lima.cs.uni-magdeburg.de:8000/**

Danksagung

Wir danken den anonymen Gutachtern für Ihre Hinweise und Fragen, die uns angespornt haben bei dem hoffentlich geglückten Versuch, unsere Arbeiten und unsere Ergebnisse noch nachvollziehbarer darzustellen.

Literatur

[Bri92] E. Brill. A simple rule-based part-of-speech tagger. In *Proceeding of the Third Conference on Applied Natural Language Processing*, pages 152–155, 1992.

[Cla02] J. Clark. http://www.jclark.com, 2002.

[CW88] H. Cunningham and Y. Wilks. GATE - a General Architecture for Text Engineering. *Proccedings of COLING-96*, 1988. http://gate.ac.uk.

[DFG02] DFG-Projekt: Workbench für die Informationsfusion. http://fusion.cs.uni-magdeburg.de, 2002.

[FN88] W. Finkler and G. Neumann. MORPHIX: A fast Realization of a classification-based Approach to Morphology. In H. Trost, editor, *Proc. der 4. Österreichischen Artificial-Intelligence Tagung, Wiener Workshop Wissensbasierte Sprachverarbeitung*, pages 11–19. Springer Verlag, August 1988.

[GM89] G. Gazdar and C. Mellish. *Natural Language Processing in LISP: An Introduction to Computational Linguistics*. Addison-Wesley, Reading, 1989.

[GPS02] GermaNet-Project-Site. http://www.sfs.nphil.uni-tuebingen.de/lsd/, 2002.

[Gro99] Language Technology Group. LT XML version 1.1. *http://www.ltg.ed.ac.uk/software/xml/*, 1999.

[KR01a] M. Kunze and D. Rösner. Eine XML-basierte Werkbank für das Document Mining. In Henning Lobin, editor, *Sprache und Texttechnologie in digitalen Medien; Proceedings der GLDV-Frühjahrstagung 2001*, pages 131–140. Gesellschaft für linguistische Datenverarbeitung (GLDV) e.V., http://www.gldv.org, 2001.

[KR01b] M. Kunze and D. Rösner. An XML-based Approach for the Presentation and Exploitation of Extracted Information. In *International Workshop on Web Document Analysis*, September 2001.

[KX02] M. Kunze and C. Xiao. Presentation of Multilingual Resources for Natural Language Processing. *Proceedings of STAIRS 2002*, erscheint July 2002.

[LD01] M.S. Lacher and S. Decker. On the integration of Topic Maps data with RDF data. *Proceedings of SWWS 2001: The First Semantic Web Working Symposium*, pages 331–344, July 2001.

[MS99] Resource Description Framework (RDF) Model and Syntax Specification. http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/, 1999.

[NBB+97] G. Neumann, R. Backofen, J. Baur, M. Becker, and C. Braun. An information extraction core system for real world german text processing. In *5th International Conference of Applied Natural Language*, pages 208–215, March 1997.

[Sit02a] GATE Site. http://gate.ac.uk, 2002.

[Sit02b] XSL Site. http://www.w3.org/Style/XSL, 2002.

[SK96] H. Schmid and A. Kempe. Tagging von deutschen Korpora mit HMM, Entscheidungsbäumen und neuronalen Netzen. In H. Feldweg and E.W. Hinrichs, editors, *Lexikon und Text: Wiederverwendbare Methoden und Ressourcen zur linguistischen Erschliessung des Deutschen*. Niemeyer, Tübingen, 1996.

[Spe01] XML Topic Maps (XTM) 1.0 Specification. http://topicmaps.org, 2001.