Ein flexibles und modulares Anfragesystem für Textcorpora

Oliver Christ und Bruno M. Schulze*

Erscheint in: Tagungsberichte des Arbeitstreffens Lexikon + Text
17./18. Februar 1994, Schloß Hohentübingen
Niemeyer: Lexicographica Series Maior, Tübingen, Frühjahr 1995

Zusammenfassung

Wir beschreiben die Architektur und einzelne Komponenten eines modularen und erweiterbaren Corpusanfragesystems, das am Institut für maschinelle Sprachverarbeitung der Universität Stuttgart innerhalb des Projektes "Textcorpora und Werkzeuge zu ihrer Erschließung" (TC) entwickelt wurde.

Die Architektur des Anfragesystems unterstützt eine prinzipiell unbegrenzte Anzahl von Corpusannotationen unterschiedlicher Typen. Neben der im Corpus annotierten Information können externe Informationsquellen zur Evaluierung einer Corpusanfrage herangezogen werden, wie z.B. Online-Thesauri. Die für Corpusanfragen zur Verfügung stehenden Annotationen sowie die Art und Weise des Zugriffs auf die Informationsquellen können individuell für jedes Corpus deklariert werden.

Um die Mächtigkeit der Anfragemöglichkeiten zu illustrieren, werden einige Beispielanfragen zusammen mit Ausschnitten aus den berechneten Ergebnissen angegeben.

1 Einführung

Die Verfügbarkeit annotierter Corpora und von Werkzeugen, die Corpora mit linguistischer Information annotieren, führt zu einem erweiterten Corpusbegriff – neben dem eigentlichen Corpustext stehen Annotationen verschiedener Typen zur Verfügung, z.B. Angaben zur Wortklasse, morphosyntaktische Angaben, semantische Annotationen, Satz- und Absatzgrenzen etc. Außerdem stehen zunehmend – insbesondere für das Englische – Wissensquellen zur Verfügung, deren Informationen in Corpusanfragesystemen nutzbar sein sollten (z.B. maschinenlesbare Wörterbücher, on-line Thesauri wie WordNet [MBF+93], Morphologische Wissensbasen wie CELEX [BPvR93] etc.), um es dem Lexikographen oder Linguisten zu erlauben, eine Corpusanfrage so genau wie möglich auszudrücken, damit die Anfrage zu einer möglichst kleinen Menge spezieller, aber für das Problem relevanter Belege führt, anstatt zu einer großen Menge von Konkordanzen, die nachträglich manuell inspiziert und zusätzlich gefiltert werden müssen. Dabei soll es dem Anwender eines Anfragesystems

*Universität Stuttgart, Institut für maschinelle Sprachverarbeitung/Computerlinguistik, Azenbergstr. 12, D 70174 Stuttgart. Email: {oli,schulze}@ims.uni-stuttgart.de

verborgen bleiben, aus welchen Quellen oder auf welche Art und Weise die Information aus den zur Verfügung stehenden Informationsquellen extrahiert wird.

Neben der interaktiven Nutzung eines Corpusanfragesystems als Datenschnittstelle im linguistischen und lexikographischen Beschreibungsprozeß ist zusätzlich eine Nutzung durch NLP-Systeme oder NLP-Werkzeuge denkbar, wobei die aus einem Corpus gewonnene Information z.B. für Disambiguierungsaufgaben in Parsern oder in einem Generierungssystem zur Bestimmung lexikalischer Präferenzen herangezogen werden kann.

Die unterschiedlichen Anwendungssituationen und die unterschiedlichen Eigenschaften der Informationsquellen machen eine flexible und modulare Architektur des Anfragesystems nötig. Ausgehend von diesen Überlegungen haben wir die folgende Architektur entworfen und implementiert: um von den unterschiedlichen Zugriffsmethoden auf Wissensquellen zu abstrahieren, wird der Zugriff auf die Corpusinformationen zwischen einer "physikalischen Zugriffsebene" und einer "logischen Zugriffsebene" aufgeteilt. Die physikalische Zugriffsebene hat dabei die Aufgabe, eine einheitliche Schnittstelle zu den Corpusdaten zur Verfügung zu stellen und vollständig von unterschiedlichen Zugriffsmethoden zu abstrahieren. Die logische Zugriffsebene nutzt diese einheitliche Schnittstelle und implementiert einen Interpreter für eine Corpusanfragesprache, die es erlaubt, die Information aller zu einem Corpus annotierten Informationsquellen in Anfragen zu verwenden, ohne jedoch von individuellen Zugriffsmethoden abhängig zu sein. Innerhalb dieser Architektur können unterschiedliche Anwendungssituationen durch unterschiedliche Schnittstellen zur logischen Zugriffsebene implementiert werden, wobei es für Werkzeuge auch möglich ist, direkt über die physikalische Zugriffsebene auf die Corpusdaten zuzugreifen, sofern kein Zugriff über die Anfragesprache notwendig ist. Diese Gesamtarchitektur ist in Abbildung 1 dargestellt.

Die einzelnen Module werden in den folgenden Abschnitten detaillierter dargestellt: Abschnitt 2 beschreibt die Aufgaben der physikalischen Ebene, in Abschnitt 3 werden die logische Zugriffsebene und der darin implementierte Anfrageinterpreter CQP beschrieben. Um in einer interaktiven Nutzungssituation den Umgang mit den Werkzeugen zu erleichtern, wurde die graphische Benutzeroberfläche XKWIC entwickelt, die in Abschnitt 4 vorgestellt wird.

2 Die physikalische Zugriffsebene

Die Aufgabe der physikalischen Ebene ist es, eine einheitliche, von individuellen Zugriffsmethoden abstrahierende Schnittstelle zu den Dateien, Datenbanken oder anderen Informationsquellen zur Verfügung zu stellen, in denen Corpusinformationen abgelegt sind. Innerhalb dieser Ebene sind auch Methoden zur Erzeugung der verschiedenen Hilfsdateien (insbesondere Indizes) implementiert.

Der Zugriff auf Corpusinformationen erfolgt meist über die *Corpusposition*: das Corpus wird dabei als eine durchnumerierte Sequenz von Wörtern und anderen Informationen betrachtet.

Innerhalb der physikalischen Ebene wird zwischen mehreren Attributtypen unterschieden:

- *Positionale Attribute* sind Attribute, bei denen jeder Corpusposition eine Zeichenkette zugeordnet ist. Insbesondere zählt die Sequenz der Wörter, aus denen das Corpus

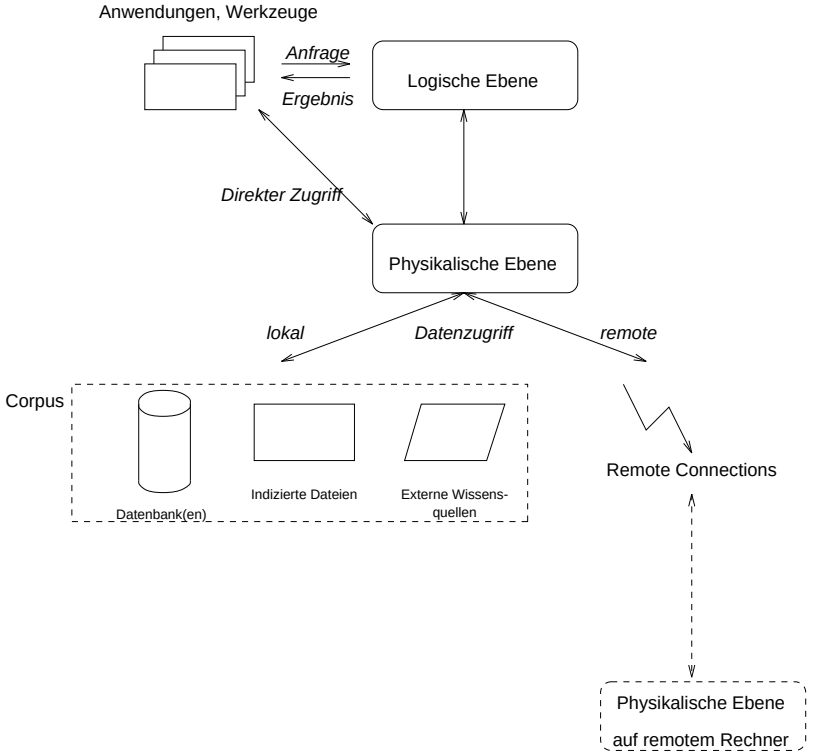

Abbildung 1: Die Gesamtarchitektur des Anfragesystems

aufgebaut ist, zu den positionalen Attributen.[1] Weitere Annotationen dieses Typs sind z.B. Wortklasseninformationen, Lemmata, morphosyntaktische Informationen etc. Die Anzahl der positionalen Attribute eines Corpus ist unbeschränkt. Abbildung

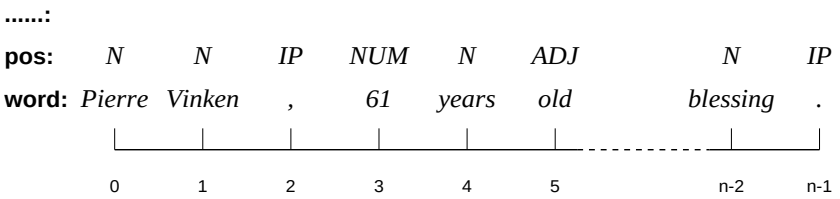

Abbildung 2: Positionale Attribute

2 illustriert die abstrakte Sicht auf positionale Attribute: jeder Corpusposition (im Bereich von 0 bis $n-1$, wobei n die Corpusgröße ist) und jedem positionalen Attribut (`word` und `pos`) sind Zeichenkettenwerte zugeordnet;

- *Strukturelle Attribute* enthalten Informationen über Satz- und Absatzgrenzen etc. Rekursive Strukturen, also z.B. NPs mit eingebetteten NPs, können gegenwärtig nicht repräsentiert werden. Strukturelle Attribute werden als Corpusintervalle re-

[1] Die Wortsequenz wird stets durch den Attributnamen `word` bezeichnet und muß immer annotiert sein. Weitere Annotationen neben der Wortsequenz sind optional und corpusspezifisch.

präsentiert, deren Anfangs- bzw. Endpunkte die Corpuspositionen sind, an denen die Struktur beginnt bzw. endet;

- *Bigram-Tabellen* enthalten zu einem deklarierten positionalen Attribut Informationen über die absolute Häufigkeit des adjazenten Auftretens zweier Werte aus dem Wertebereich dieses positionalen Attributs innerhalb einer bestimmten Fenstergröße. Zu jedem positionalen Attribut können solche Bigram-Tabellen definiert werden. Bigram-Tabellen können neben dem Zugriff durch die logische Zugriffsebene z.B. direkt von Werkzeugen zur automatischen Wortklassenzuweisung (Taggern) verwendet werden;

- *Mapping-Tabellen* sind zu jeweils zwei positionalen Attributen des Corpus deklariert und enthalten Informationen über die absolute Häufigkeit, mit der ein Wert des ersten Attributs und ein Wert des zweiten Attributs an der gleichen Corpusposition annotiert sind. Auch dieser Informationstyp kann in Taggern verwendet werden;

- *Alignment-Informationen* können zu zwei parallelen, alignten Corpora deklariert werden und enthalten Informationen über sich entsprechende, übersetzungsäquivalente Bereiche. Es ist in der gegenwärtigen Implementierung nicht möglich, zu einem Corpuspaar Alignment-Informationen auf mehr als einer Ebene zu definieren, also zusätzlich zu alignten Sätzen noch Informationen über alignte Phrasen oder Wörter abzulegen;

- *Dynamische* oder *virtuelle* Attribute schließlich binden externe Wissensquellen an das Corpus, ohne daß die durch diese Wissensquellen repräsentierten Informationen direkt annotiert wären ("virtuell"). Dieser Attributtyp verhält sich ähnlich wie eine Funktion: zu jedem solchen Attribut ist eine Argumenttypenliste und ein Rückgabewerttyp definiert sowie ein Kommando, das zur Berechnung des Attributwertes zum Zeitpunkt des Zugriffs mit entsprechenden aktuellen Parametern aufgerufen wird und einen Wert zurückliefert, der dann weiterverarbeitet wird ("dynamisch"). Indexdateien können für solche Attribute nicht erzeugt werden.[2]

Vor dem Bearbeitung mit dem Anfragesystem müssen Corpora aufbereitet werden. Die dabei notwendigen Schritte sind in erster Linie Zeichensatznormierung, Tokenisierung, Satzgrenzenerkennung und die Erzeugung von Indexdateien (vgl. [Chr94]). Der eigentliche Corpustext wird nach der Aufbereitung nicht mehr benötigt.

Nach der Aufbereitung muß ein Corpus *registriert* werden. Dabei wird in einer allgemein zugänglichen Datei festgehalten, welche Annotationen zu einem Corpus deklariert sind und wo innerhalb des Dateisystems die einzelnen Dateien abgelegt sind. Gleichzeitig wird durch die Registrierungsdatei ein symbolischer, eindeutiger Name für jedes Corpus definiert. Alle Werkzeuge greifen ausschließlich über diesen symbolischen Namen auf ein Corpus zu, so daß die Benutzer nicht wissen müssen, wo und auf welche Weise die Corpusdaten abgelegt sind.

[2] Der Begriff des *virtuellen* Attributs wurde von Herrn Prof. Rieger, Universität Trier, vorgeschlagen. Während durch *virtuell* die indirekte Kopplung der externen Informationsquelle an das Corpus charakterisiert wird, bezeichnet der Begriff *dynamisch* die Art und Weise des Zugriffs auf die Annotation und das Vorgehen bei der Evaluierung. Wir möchten am Begriff *dynamisch* festhalten.

```
NAME "Hansard corpus (Englischer Teil)"
ID      hansard-e
HOME /corpora/encoded/hansard-e

ATTRIBUTE word
ATTRIBUTE pos
ATTRIBUTE lemma

DYNAMIC ishuman(STRING):INT "/corpora/utils/cmd/wn-hypen '$1' human"

ALIGNED hansard-f            # franzoesischer Teil
```

Abbildung 3: Eine vollständige Registrierungsdatei mit mehreren Attributen

Abbildung 3 zeigt eine Registrierungsdatei für den englischen Teil der Hansard-Corpora.[3] Zu dem Corpus sind drei positionale Attribute (`word, pos, lemma`) definiert,[4] weiterhin ein dynamisches Attribut `ishuman`, das zum Zeitpunkt der Anfrageevaluierung für eine Argumentzeichenkette über einen Aufruf des Online-Thesaurus WordNet bestimmt, ob das Argument ein "human object" denotieren könnte. Das Corpus ist alignt zum französischen Teil der Hansard-Corpora.

Verschiedene Annotationen eines Corpus werden in verschiedenen Dateien gehalten, so daß ein Corpus auch nach der Registrierung erweitert, aktualisiert oder vervollständigt werden kann, ohne daß dafür Änderungen an bereits bestehenden Annotationen (Reindizierungen etc.) notwendig werden (sofern die Corpusgröße konstant bleibt).

Zu Testzwecken wurde innerhalb der physikalischen Ebene ein TCP/IP-basiertes Kommunikationsprotokoll definiert, das es ermöglicht, Corpora oder auch nur einzelne Annotationen verteilt auf verschiedenen Rechnern, die über das Internet erreichbar sind, abzulegen. Beim Zugriff auf extern abgelegte Corpusdaten wird eine Netzwerkverbindung aufgebaut und – falls der Benutzer mit den notwendigen Zugriffsrechten ausgestattet ist – die verlangte Information zurückgeliefert. Der netzwerkbasierte Corpusdatenaustausch ist zwar langsam, kann aber nützlich sein, wenn Anfragesysteme auf Rechnern zur Verfügung stehen sollen, die selbst zuwenig Speicherplatz für die lokale Installation der Corpora bieten oder nicht effizient genug sind. Der netzwerkbasierte Datenaustausch ist ausschließlich innerhalb der physikalischen Ebene implementiert, so daß der Zugriff auf externe Daten von Werkzeugen, die auf der physikalischen Ebene aufsetzen, transparent erfolgt.

Die wichtigste Anwendung, die auf der physikalischen Ebene aufbaut, ist die logische Zugriffsebene, die eine Corpusanfragesprache und einen entsprechenden Anfrageinterpreter implementiert. Andere Werkzeuge jedoch nutzen ausschließlich die Dienste der physikalischen Zugriffsebene, z.B. Werkzeuge zur Ausgabe von Frequenzlisten oder Mutual-Information-Werten.

[3] Die Hansard-Corpora sind Transkriptionen kanadischer Parlamentsdebatten und liegen parallel in Französisch und Englisch vor.

[4] Die Lemmatisierung des Corpus wurde mit Hilfe des in [KSZE92] beschriebenen Werkzeugs durchgeführt.

3 Die logische Zugriffsebene und die Corpusanfragesprache

Die logische Zugriffsebene nutzt die von der physikalischen Ebene bereitgestellten Datenzugriffsmethoden, um Corpusanfragen zu evaluieren und das Ergebnis zu präsentieren oder an andere Werkzeuge, die auf der logischen Ebene aufbauen, weiterzureichen.

Innerhalb der logischen Zugriffsebene ist ein Interpreter für eine Corpusanfragesprache implementiert. Mit Hilfe dieser Anfragesprache werden unter Einbeziehung der Corpusannotationen Bedingungen angegeben, die für eine Teilsequenz des Corpus gelten müssen, um in die Ergebnismenge der Anfrage übernommen zu werden.

Das zentrale Konstrukt der Anfragesprache ist der *Attributausdruck*. In einem solchen Ausdruck werden durch einen booleschen Ausdruck die Bedingungen beschrieben, die – wenn der Ausdruck für eine bestimmte Corpusposition evaluiert wird – die Werte der verschiedenen Corpusannotationen an dieser Corpusposition erfüllen müssen.

Der Attributausdruck (1)

(1) `[word="ver.*" & pos != "FIN"]`

liefert "wahr" für alle Corpuspositionen, an denen der Wert des `word`-Attributs dem regulären Ausdruck `"ver.*"` genügt[5] und der Wert des `pos`-Attributs ungleich `"FIN"` ist.[6]

Eine Anfrage besteht im allgemeinen aus einem regulären Ausdruck über Attributausdrücke. Beispiel (1) ist also bereits eine Anfrage, allerdings muß innerhalb des Anfrageinterpreters jede Anfrage (und jedes Kommando) mit einem Strichpunkt ";" abgeschlossen werden. Als "Matchall"-Zeichen (entsprechend dem Punkt in regulären Ausdrücken über das Buchstabenalphabet) kann in der Anfragesprache der "leere" Attributausdruck `[]` verwendet werden.[7]

Bei der Evaluierung einer Anfrage extrahiert der Anfrageinterpreter alle Corpusintervalle, die dem regulären Ausdruck über Attributausdrücke genügen. Die Grenzen eines einzelnen Ergebnisintervalls sind dabei der Anfangs- und der Endpunkt der kürzesten den Anfragebedingungen genügenden Teilsequenz des Corpus, also Beginn und Ende der "matchenden" Sequenz. Das Ergebnis einer Anfrage ist dann eine Menge von Corpusintervallen, die aufgrund von Wiederholungsoperatoren und optionalen Teilausdrücken unterschiedliche Länge haben können.

Die folgenden Beispiele illustrieren einige Eigenschaften der Anfragesprache.

Die Anfrage (2)

[5] In Attributwerten können reguläre Ausdrücke über das Buchstabenalphabet gemäß dem POSIX-egrep-Standard verwendet werden.

[6] Die Menge der annotierbaren Wortklassen ("Tagset"), auf deren Werte in diesem Beispiel über den Attributnamen `pos` zugegriffen wird, ist corpusspezifisch und nicht vom Anfragesystem vorgegeben. Da die Menge der positionalen Attribute beliebig ist, können unterschiedliche Tagsets unter verschiedenen Attributnamen parallel annotiert werden.

[7] Wie in regulären Ausdrücken über Buchstaben wird direkte Adjazenz durch aufeinanderfolgende Attributausdrücke (bzw. reguläre Ausdrücke über Attributausdrücke) ausgedrückt. Weitere Konstrukte sind Optionalität (`?`), Disjunktion (`|`), Stern-Hülle (`*`), Plus-Hülle (`+`), Repetitionsintervalle ($\{n, m\}$) etc. Klammern können in der üblichen Weise verwendet werden.

(2) `[pos="SUB"] [pos="FIV"] [pos="PRP"] [pos="PER"] []* [pos="PTZ2"];`

sucht nach einer Sequenz von Nomen (`SUB`), finiter Verbform (`FIV`), Präposition (`PRP`), Personalpronomen (`PER`), beliebig vielen unspezifizierten Formen und einem abschließenden Partizip (`PTZ2`), wobei ein entsprechendes Tagset zugrundeliegt. Die folgenden Beispielbelege wurden durch Anfrage (2) aus einem Corpus aus einigen Verbmobil-Dialogen extrahiert, wobei der die Anfragebedingungen erfüllende ("*matchende*") Bereich kursiv dargestellt ist:

> denn die andern *Montage sind bei mir dann auch schon belegt* , aber ja
> Kongreß , der *Donnerstag ist bei mir Absolut ausgebucht* . also ja

Wenn in einem Attributausdruck nur auf das (Standard-)Attribut `word` Bezug genommen wird und der Operator "`=`" verwendet wird, können die umschließenden Klammern sowie die Attributgleichung `word =` weggelassen werden. Dadurch läßt sich ein Attributausdruck

```
[word="gehen|ging|gegangen"]
```

durch

```
"gehen|ging|gegangen"
```

abkürzen.

Auf Werte von dynamischen Attributen wird ähnlich zugegriffen:[8]

(3) `"kill.*" []? [pos="N.*" & ishuman(word)]`

Wie in der Registrierungsdatei in Abbildung 3 deklariert, liefert der Aufruf des dynamischen Attributes `ishuman` einen Zahlenwert zurück. Falls die rechte Seite der Attribut-Wert-Gleichung beim Aufruf eines zahlenwertigen dynamischen Attributs fehlt, liefert die Evaluierung des Teilausdrucks "wahr", falls der zurückgelieferte Zahlenwert ungleich 0 ist, und "falsch" sonst. Das Ergebnis der Beispielanfrage (3) sind Sequenzen, die aus einem Wort bestehen, das mit `kill` beginnt, gefolgt von einer optionalen, nicht weiter spezifizierten Form (beispielsweise "by"), schließlich gefolgt von einem Nomen, für das die WordNet-Konsultation "wahr" zurückliefert, wenn beim Aufruf der Wert des `word`-Attributes des Nomens übergeben wird. Beispielbelege für die Anfrage lauten:

> and Lane that he would *kill the Indian* . Three weeks later following his
> by the sheriff for *killing an old_man* named Asher_Jones , the warrant for

Es gibt einige vordefinierte Funktionen, die ähnlich wie dynamische Attribute verwendet werden, nicht aber dynamisch durch Zugriff auf externe Informationsquellen berechnet werden. Die Funktion "`f`" beispielsweise liefert die absolute Häufigkeit der Argumentzeichenkette im Corpus zurück:

[8] Da der online-Thesaurus WordNet, über den der Wert des dynamischen Attributs `ishuman` berechnet wird, nur für das Englische zur Verfügung steht, wurde die Anfrage auf einem englischen Corpus evaluiert.

(4) `"love.*" []? [pos="N.*" & f(word)>10 & ishuman(word)];`

In Beispiel (4) wird zusätzlich verlangt, daß die absolute Häufigkeit des Wertes des `word`-Attributs im Corpus größer ist als 10. Ähnliche Funktionen erlauben den Zugriff auf Bigram- und Mapping-Tabellen.

Auf strukturelle Attribute kann in einer SGML-ähnlichen Weise zugegriffen werden:

(5) `[pos="N.*"] [] <s> [pos="ART"]`

In Anfrage (5) wird verlangt, daß zwischen dem Nomen, einer beliebigen darauffolgenden Form (dem Satzendezeichen) und dem Artikel eine Satzgrenze liegt.

Im allgemeinen Fall wird bei Wiederholungsoperatoren bis ans Corpusende gesucht, falls die Evaluierung des regulären Ausdrucks nicht früher zu einem Ergebnis führt.[9] Um zu fordern, daß die Grenzen des Ergebnisintervalls innerhalb einer annotierten Struktur liegen oder um eine Beschleunigung der Anfrageevaluierung zu erreichen, kann in der Anfrage eine Suchgrenze angegeben werden. Diese Grenze kann entweder als maximale Zahl von Corpuspositionen angegeben werden oder es kann eine der deklarierten Strukturannotationen verwendet werden, wie das folgende Beispiel zeigt:

(6) `"ver.*" []* "\"uber" within s;`

Hier muß die gesamte Ergebnissequenz innerhalb eines Satzes liegen.[10]

Ein weiteres Konstrukt der Anfragesprache sind Referenzen auf Attributwerte an vorangegangenen Corpuspositionen. Dazu wird ein Attributausdruck mit einer Markierung versehen:

(7) `a:[pos="N.*"] ...`

In einem folgenden Attributausdruck kann dann auf Attributwerte der der Markierung zugeordneten Corpusposition Bezug genommen werden, wie in Beispiel (8):

(8) `a:[pos="N.*"] []* [pos="V.*" & num=a.num] within s;`

In dieser Anfrage wird also eine Übereinstimmung bezüglich des Numerus-Wertes des Nomens und des Verbs gefordert. Zusätzlich muß die gesamte Ergebnissequenz innerhalb eines Satzes liegen. Anfrage (9)

(9) `a:[pos="N.*"] ([]* [word=a.word] ){2} within s;`

[9] Reguläre Ausdrücke über Attributausdrücke werden mit einer *First-Match-Strategie* evaluiert. Um z.B. bei falschgeschriebenen oder im Corpus nicht auftauchenden Wörtern nicht bis ans Corpusende suchen zu müssen, ist intern eine feste Suchgrenze von 100 Corpuspositionen, gezählt vom Beginn des "Matches", festgelegt. Diese Grenze kann vom Benutzer beliebig verändert werden.

[10] Falls eine Eingabe von 8Bit-ASCII-Zeichen nicht möglich ist, können solche Zeichen, z.B. Umlaute, entweder in ihrer LaTeX-Umschreibung oder als Oktalcode eingegeben werden.

liefert Konkordanzen, bei denen das gleiche Nomen mindestens dreimal innerhalb eines Satzes auftritt.

Bei alignten, parallelen Corpora erlaubt es die Anfragesprache, zusätzliche Bedingungen über den alignten Teil des zu testenden Corpusintervalls zu formulieren. Anfrage (10)

(10) `"hot" :HANSARD-F "chaud";`

liefert Konkordanzen für *hot*, wobei im alignten, französischen Teil gleichzeitig das Wort *chaud* vorkommen muß. Die Negation ist ebenfalls möglich:

(11) `"hot" :HANSARD-F ! "chaud";`

Das Ergebnis sind Konkordanzen, in denen das Wort *chaud* im alignten Teil nicht vorkommt. Im allgemeinen Fall kann im Bedingungsteil über das alignte Corpus wiederum ein regulärer Ausdruck über Attributausdrücke verwendet werden.

Neben einigen Kommandos zur Corpus- und Dateiverwaltung unterstützt der Anfrageinterpreter *inkrementelle Anfragen*. Dabei ist es möglich, eine Anfrage nur auf dem Ergebnis einer früheren Anfrage zu evaluieren, was zu erheblichen Effizienzgewinnen führen kann. Beispielsweise können aus einem großen Corpus zuerst alle Sätze extrahiert werden, in denen eine bestimmte Wortform vorkommt. Anschließend wird nur die Menge dieser Sätze z.B. nach bestimmten syntaktischen Phänomenen untersucht. Zusätzlich stehen *Mengenoperationen* zur Verfügung, mit deren Hilfe der Durchschnitt, die Vereinigung und die Differenz von Anfrageergebnissen berechnet werden kann. Das Ergebnis einer solchen Mengenoperation wird wie ein durch eine Anfrage berechnetes Ergebnis behandelt, so daß darauf aufbauend weitere inkrementelle Anfragen oder Mengenoperationen durchgeführt werden können. Alle durch Mengenoperationen oder Suchanfragen entstandenen Ergebnismengen können statisch in Dateien abgelegt werden und stehen dadurch in nachfolgenden Sitzungen innerhalb des Anfrageprozessor wieder zur Verfügung. Eine Beschreibung der einzelnen Kommandos, der genauen Syntax der Corpusanfragesprache sowie ein einführendes Tutorial stehen in [SC94] zur Verfügung. Eine formale Darstellung des Entwurfs, der Semantik und der Implementierung der Anfragesprache sowie ein Vergleich mit anderen Corpusanfragewerkzeugen wird in [Sch94] gegeben.

Der Anfrageinterpreter kann im interaktiven Modus oder als Batch-Prozessor verwendet werden. Für den interaktiven Modus stehen sowohl eine kommandobasierte Benutzeroberfläche zur Verfügung als auch eine komfortable, graphische Benutzeroberfläche, die im nächsten Abschnitt vorgestellt wird.

4 Das Interaktionswerkzeug XKWIC

XKWIC (vgl. [Chr93]) ist eine auf OSF/Motif(tm) und dem X Window System basierende graphische Benutzeroberfläche, die neben den Kommandos, die über die kommandobasierte Benutzeroberfläche zum Anfrageprozessor zur Verfügung stehen, noch zusätzliche Kommandos und Funktionen anbietet:

- das Anfrageergebnis wird in einer KWIC-Konkordanz präsentiert, wobei der angezeigte Ausschnitt über *Scrollbars* verändert werden kann;
- das präsentierte Anfrageergebnis kann nach unterschiedlichen Kriterien sortiert werden;
- durch Selektion einer einzelnen Konkordanzzeile wird diese mit einem erweiterten, benutzerdefinierbaren Kontext angezeigt;
- das Anfrageergebnis oder ein selektierbarer Teil davon kann in einer Datei textuell abgespeichert werden;
- einzelne Konkordanzzeilen können manuell aus dem Anfrageergebnis gelöscht werden oder es können neue Ergebnismengen manuell durch Kopieren aus anderen Ergebnismengen erzeugt werden;
- bei parallelen, alignten Corpora kann durch Selektion einer Konkordanzzeile der entsprechende Bereich im alignten Corpus mitangezeigt werden.

Zusätzlich wird innerhalb von XKWIC eine einfache *Query-History* unterstützt. Diese Query-History besteht aus einer Liste, in der der Anfragetext und der Name des zugrundeliegenden Suchcorpus aller eingegebenen Corpusanfragen festgehalten wird. Eine in der Liste gespeicherte Anfrage kann durch Selektieren erneut ausgeführt werden oder der Anfragetext kann in das Anfrageeingabefenster übernommen werden, um erst nach eventuellen Änderungen ausgeführt zu werden. Die Query-History kann in einer Datei abgespeichert werden und steht somit in nachfolgenden Sitzungen wieder zur Verfügung.

Abbildung 4 zeigt XKWIC nach der Bearbeitung einer Anfrage. Der obere Bereich des Fensters dient zur Eingabe einer Anfrage und des Suchcorpus. Die Menge der verfügbaren Suchcorpora wird durch Selektion des mit dem Fragezeichen versehenen Buttons rechts neben dem Eingabefenster für das Suchcorpus in einem separaten Dialogfenster angezeigt.

Der mittlere Teil des Fensters dient zur Anzeige des Anfrageergebnisses, wobei der angezeigte Ausschnitt durch den Scrollbar rechts des Fensters in der üblichen Art und Weise verändert werden kann. Beim Selektieren einer Konkordanzzeile wird diese invers unterlegt und mit einem benutzerdefinierbaren, erweiterten Kontext im untersten Fenster dargestellt.

In XKWIC kann momentan bei der Anzeige einer Konkordanzzeile lediglich die Sequenz der Wortformen ausgegeben werden. Die Ausgabe der Werte anderer positionaler oder struktureller Attribute, die einem Corpus zugeordnet sind (z.B. POS-Tags), wird derzeit noch nicht unterstützt.

Innerhalb von XKWIC wurde ein auf dem *X Windows Inter-Client Communications Protocol* basierendes Kommunikationsprotokoll implementiert, das es ermöglicht, Anfragen von anderen Programmen an XKWIC zu senden, um sie dort evaluieren und präsentieren zu lassen. Dadurch wurde z.B. die Anbindung des Lisp-basierten Typed Feature Systems (TFS, [Eme94]) an XKWIC möglich. Im Rahmen des DELIS-Projektes ([HE94]), in dem unter anderem ein interaktives System zur corpusbasierten Lexikonentwicklung implementiert wird,

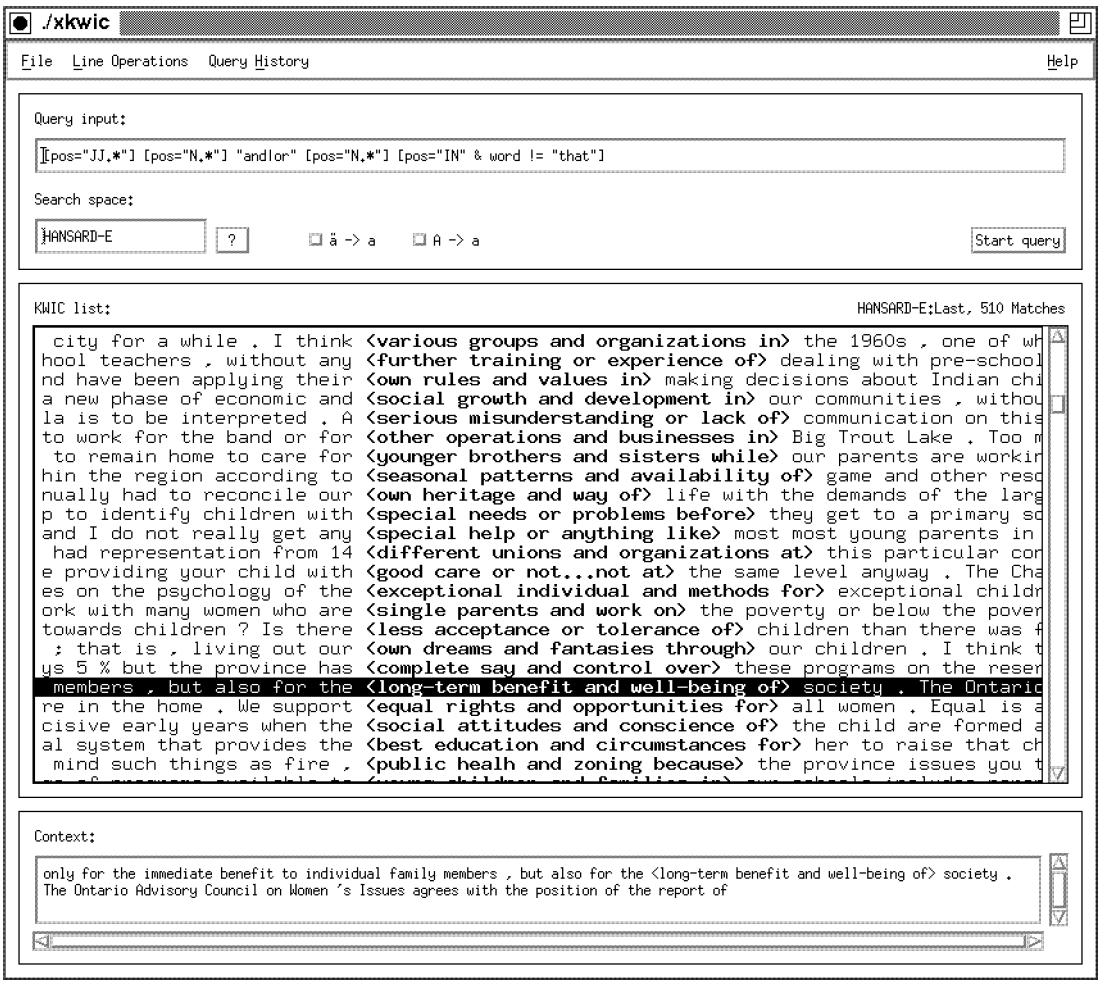

Abbildung 4: Das Präsentations- und Interaktionswerkzeug XKWIC

werden dabei in TFS spezifizierte syntaktische Informationen über Lemmata in Corpusanfragen an ein mit speziellen Informationen annotiertes Corpus abgebildet. Diese Corpusanfragen werden dann über das Kommunikationsprotokoll an XKWIC zur Evaluierung und Präsentation gegenüber dem Benutzer geschickt.

Zusätzlich zu der Möglichkeit, innerhalb von XKWIC den alignten Teil einer einzelnen Konkordanzzeile anzeigen zu lassen, erlaubt das Programm `print-aligned` die textuelle Ausgabe eines vollständigen Anfrageergebnisses zusammen mit der jeweils jeder einzelnen Konkordanzzeile zugeordneten Übersetzung. Die Ausgabe von `print-aligned` für die Anfrage

(12) `"den" ".*(tag|woch)" [pos="INF"]`

lautet ausschnittsweise

> vmob-d: und ich seh grade , die Freitage danach hab ich auch mehr Platz . vielleicht sollten wir dann *den Freitag festhalten* für dieses Seminar .
>
> vmob-e: And I see that I have more time on the subsequent Fridays. Maybe we should settle on Friday then , for this seminar.

> vmob-d: halt ich fest , notier ich eben , dann muß ich die Wochenplanung vielleicht mal auf *den Mittwoch verschieben* ,
>
> vmob-e: I'm noting that down , then I'll have to move the scheduling of the week to Wednesday.

5 Effizienz der Anfrageevaluierung

Die Effizienz der Anfrageevaluierung hängt wesentlich von der Komplexität des Anfrageausdrucks ab. Dabei ist im allgemeinen der erste Teilausdruck der Anfrage von entscheidender Bedeutung. Wird bei großen Corpora innerhalb des ersten Teilausdrucks ein regulärer Ausdruck auf der Attributwertebene verwendet, so kann die Evaluierung der Anfrage beträchtliche Zeit in Anspruch nehmen. Enthält der erste Ausdruck jedoch keine regulären Ausdrücke, ist der Zugriff auf die Corpusdaten durch die verwendeten Indizierungstechniken effizient genug für die interaktive Anwendung.

Die einfache Anfrage (13)

(13) `"Bre.*zel";`

benötigt zur Evaluierung auf einem 36 Millionen Wörter umfassenden deutschen Zeitungscorpus ca. 17 Sekunden.[11] Das Ergebnis der Anfrage (14)

(14) `"Brezel";`

[11] Außer "Brezel" (23 Vorkommen) werden im Suchcorpus durch den verwendeten regulären Ausdruck nur noch die Schreibungsvariante "Bretzel" und der Eigenname "Brentzel" (die jeweils nur einmal auftreten) beschrieben. Alle Zeitmessungen wurden auf einer 2-Prozessor SparcStation 10 (Betriebssystem SunOS 4.1.3) unter normaler Belastung durchgeführt.

wird dagegen in 0.7 Sekunden berechnet, wobei die absolute Häufigkeit des Wortes im Corpus auch ohne die Verwendung regulärer Ausdrücke Auswirkungen auf die Effizienz hat. So benötigt die Evaluierung der Anfrage (15)

(15) `"Kohl";`

ca. 9 Sekunden und liefert 4812 Vorkommen.

Das Ergebnis der Suchanfrage (16)

(16) `"gehen|ging|gegangen";`

wird in ca. 224 Sekunden berechnet. Innerhalb der Anfragesprache stehen jedoch zwei weitere Möglichkeiten zur Verfügung, um die Anfrage zu formulieren:

(17) `"gehen" | "ging" | "gegangen";`

(18) `[word="gehen" | word = "ging" | word = "gegangen"];`

In (17) liegen disjunktiv verknüpfte Attributausdrücke vor, so daß die Disjunktion auf der Ebene des endlichen Automaten behandelt wird. In (18) wird nur ein Attributausdruck verwendet, wobei die Disjunktion innerhalb des booleschen Ausdrucks bearbeitet wird. Das ursprüngliche Beispiel (16) behandelt die Disjunktion auf der Ebene regulärer Ausdrücke in Attributwerten.

Die beiden Anfragen (17) und (18) werden erheblich schneller berechnet (in beiden Fällen ca. in 28 Sekunden), da keine regulären Ausdrücke innerhalb der Attributwerte verwendet werden. Alle drei Anfragen berechnen natürlich das identische Ergebnis (15837 Vorkommen). Durch interne Datencaching-Mechanismen des Betriebssystems kann allerdings die Zeitmessung erheblich gestört werden: so wird das Ergebnis von Anfrage (18) bei einer sofortigen erneuten Ausführung in ca. 1 Sekunde berechnet.

Der erste Teilausdruck einer Anfrage (ebenso der 1. Ausdruck eines disjunktiv verknüpften weiteren Teilausdrucks) ist deshalb von für die Effizienz entscheidender Bedeutung, da durch ihn die "Startpositionen" für die Evaluierung des aus dem regulären Anfrageausdruck gebildeten endlichen Automaten festgelegt werden. Liegen diese Startpunkte fest, wird der Automat für jeden dieser Startpunkte angestoßen und überprüft, ob die in den Attributausdrücken formulierten Bedingungen an den jeweils betrachteten Positionen erfüllt werden. Dabei ist klar, daß einfachere und "kürzere" Automaten auch schneller evaluiert werden können.

Dynamische Attribute sind aufgrund der mit jeder Wertberechnung verbundenen Shell-Aufrufe sehr ineffizient, weshalb sie in den Beispielen nur jeweils innerhalb der letzten, mit anderen Attribut-Wert-Gleichungen konjunktiv verknüpften Attribut-Wert-Gleichung verwendet wurden, da durch die vorausgehenden Attribut-Wert-Gleichungen innerhalb des gleichen Attributausdrucks die Anzahl der Aufrufe eines dynamischen Attributs stark eingeschränkt werden kann.

Durch Verwendung weiterer Indexdateien und verschiedener Optimierungstechniken bei der Anfrageevaluierung könnte die Effizienz des Anfragesystems erheblich gesteigert werden.

Dies bezieht sich insbesondere auf Mechanismen, die die Evaluierung regulärer Ausdrücke in Attributwerten beschleunigen könnten.

Für den Benutzer des Anfragesystems ergibt sich die Folgerung, daß bei großen Corpora reguläre Ausdrücke am Anfang eines Anfrageausdrucks vermieden werden sollten oder in eine Aufzählung wie bei (17) oder (18) umgeformt werden sollten. Bei kleineren Corpora ist eine Umformung nicht notwendig. Ein weiterer Effizienzgewinn ergibt sich, wenn das Corpus lemmatisiert vorliegt und so z.B. die Aufzählung konjugierter Formen eines Verbs überflüssig wird.

6 Zusammenfassung

Mit dem hier vorgestellten System steht ein Corpusanfragesystem zur Verfügung, das den effizienten Zugriff auf große, annotierte Textcorpora ermöglicht. Durch die Aufteilung des Gesamtsystems in unterschiedliche Module wird auf der Seite der Werkzeuge eine hohe Flexibilität erreicht, was die Entwicklung spezieller Werkzeuge, die Corpusinformation benötigen, auf der Grundlage bestehender Module und Werkzeuge stark vereinfacht. Auf der Seite der Corpusdaten wird durch unterschiedliche Attributtypen, die Trennung der verschiedenen einem Corpus annotierten Attribute und durch die Aufteilung der Informationen in verschiedene Dateien ebenfalls eine hohe Modularität erreicht, so daß einzelne Annotationen eines Corpus hinzugefügt, erweitert, aktualisiert oder korrigiert werden können, ohne daß Änderungen an bestehenden Annotationen notwendig werden. Über netzwerkbasierten Corpusdatenaustausch ist auch eine über verschiedene Rechner verteilte Datenrepräsentation möglich.

Beim Entwurf der Anfragesprache wurde versucht, auf bekannte Beschreibungselemente wie reguläre Ausdrücke, Attribut-Wert-Gleichungen und boolesche Ausdrücke zurückzugreifen, was das Erlernen der Anfragesprache und den Umgang mit den Anfragewerkzeugen erleichtert. Durch die graphische Benutzeroberfläche XKWIC zum Anfrageprozessor wurde eine zusätzliche Benutzerfreundlichkeit erreicht.

Die Unterstützung inkrementeller Anfragen und die Möglichkeit, Anfrageergebnisse durch Mengenoperationen kombinieren zu können, kann zu erheblichen Effizienzgewinnen bei der Anfrageevaluierung führen. Außerdem wird dadurch die Entwicklung einer Anfragestrategie unterstützt: der Anwender des Systems kann im Vorfeld der eigentlichen Corpusarbeit bereits einen Teil der Vorgehensweise planen, mit der durch schrittweise verfeinerte, aufeinander aufbauende inkrementelle Anfragen aus zuerst großen Belegmengen nach und nach detaillierte, kleine Belegmengen gefiltert werden.

Ein weiterer Vorteil des vorgestellten Systems ist es, externe linguistische Wissensquellen über zum Zeitpunkt der Anfrageevaluierung berechnete Attributwerte an ein Corpus binden zu können, beispielsweise Thesauri, Terminologiedatenbanken o.ä. Die durch diese Wissensquellen angebotene Information kann dann in Corpusanfragen verwendet werden.

Die Möglichkeit, auf einfache Weise in annotierten Corpora zu suchen, erlaubt eine wesentlich detailliertere Spezifikation der im Corpus zu suchenden Belege, woraus eine kleinere Anzahl, dafür aber unter Umständen umso relevanterer Belege resultiert. Insgesamt kann

durch detailliertere Anfragemöglichkeiten die Zahl der eventuell manuell zu filternden Belege stark reduziert werden, was im linguistischen Beschreibungsprozeß insbesondere bei großen Corpora positive Auswirkungen auf die Effizienz hat, mit der eine auf einer Konkordanz basierende Beschreibung erstellt werden kann.

Literatur

[BPvR93] R. H. Baayen, R. Piepenbrock, H. van Rijn. The CELEX Lexical Database (CD-ROM). Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA, 1993.

[Chr93] Oliver Christ. *The Xkwic User Manual.* Institut für maschinelle Sprachverarbeitung, Universität Stuttgart, 1993.

[Chr94] Oliver Christ. *The IMS Corpus Workbench Technical Manual.* Institut für maschinelle Sprachverarbeitung, Universität Stuttgart, 1994.

[Eme94] Martin Emele. TFS – The Typed Feature Structure Representation Formalism. Erscheint in: *Proceedings of the International Workshop on Sharable Natural Language Resources (SNLR)*, 1994.

[HE94] Ulrich Heid, Martin Emele. DELIS: Tools for Corpus Based Lexicon Building. Erscheint in: *Proceedings of KONVENS '94*, 1994.

[KSZE92] Daniel Karp, Yves Schabes, Martin Zaidel, Dania Egedi. A Freely Available Wide Coverage Morphological Analyzer for English. In *Proceedings of COLING '92, Nantes*, 1992.

[MBF+93] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, Katherine Miller. Introduction to WordNet: An On-line Lexical Database. Technischer Bericht, Cognitive Science Laboratory, Princeton University, 1993.

[SC94] Bruno M. Schulze, Oliver Christ. *The CQP Users's Manual.* Institut für maschinelle Sprachverarbeitung, Universität Stuttgart, Version 1.0d, Mai 1994.

[Sch94] Bruno M. Schulze. Entwurf und Implementierung eines Anfragesystems für Textcorpora. Diplomarbeit Nr. 1059, Institut für maschinelle Sprachverarbeitung (IMS) and Institut für Informatik, Universität Stuttgart, Januar 1994.