Köln, den 17. September 2003

Studiengang Informationsverarbeitung
WS 2002/2003 – SS 2003
Sprachliche Informationsverarbeitung
Proseminar: ”Computerlinguistische Grundlagen”
bei Jürgen Hermes M.A.

Informationsextraktion

vorgelegt von

Fabian Steeg
Matrikelnummer 3598900
E-Mail: steeg@netcologne.de
Liebigstr. 43
50823 Köln

Inhaltsverzeichnis

1 Grundlegendes zur Informationsextraktion

1.1 Was ist Informationsextraktion?

Informationsextraktion (IE) kann aus zwei verschiedenen Perspektiven betrachtet werden. Einerseits als das Erkennen von bestimmten Informationen – so bezeichnet etwa Grishman IE als ”the automatic identification of selected types of entities, relations, or events in free text” (Grishman 2003) –, andererseits als das Entfernen der Informationen, die nicht gesucht werden. Letztere Sichtweise drückt etwa eine Definition von Cardie aus: ”An IE system takes as input a text and 'summarizes' the text with respect to a prespecified topic or domain of interest” (Cardie 1997). In diesem Sinne könnte man Informationsextraktion auch als gezielte Textzusammenfassung bezeichnen (vgl. Euler 2001a, 2001b und Abschnitt 4). Informationsextraktionssysteme sind also immer zumindest auf ein spezielles Fachgebiet, meist sogar auf bestimmte Interessengebiete (Szenarios) innerhalb eines allgemeineren Fachgebietes (Domäne) ausgerichtet. So wäre etwa in der Domäne 'Wirtschaftsnachrichten' ein mögliches Szenario 'Personalwechsel in einer Managementposition'. Eine weitergehende Einschränkung macht Neumann, wenn er schreibt, daß das Ziel der IE ”die Konstruktion von Systemen” sei, ”die gezielt domänenspezifische Informationen aus freien Texten aufspüren *und strukturieren* können [...]” (Neumann 2001, Hervorhebung von mir). In diesem Zusammenhang ist zu beachten, daß eine solche Einschränkung Konsequenzen für die technische Realisierung eines Informationsextraktionssystems hat. Ziel dieser Arbeit ist es, einen Überblick über Informationsextraktion als das zu verschaffen, was alle Definitionen gemeinsam haben, nämlich die Gewinnung von spezifizierten Informationen aus Texten sowie die wissenschaftliche Disziplin, die sich mit dieser Aufgabe beschäftigt.

1.2 Abgrenzung von Nachbargebieten

Abzugrenzen ist das eigenständige Forschungsgebiet der Informationsextraktion von verwandten Gebieten: Textzusammenfassung hat eine umfassende Zusammenfassung[1] des Inhaltes eines Textes zum Ziel. Textkategorisierung bedeutet das selbstständige Gruppieren von Texten, Textklassifikation das Einordnen von Texten in vorgegebene Gruppen. Mit Information Retrieval kann die Suche nach Dokumenten in einer Dokumentenmenge (Volltextsuche) oder auch – entsprechend der wörtlichen Bedeutung – die allgemeiner formulierte Aufgabe des Abrufs von Informationen gemeint sein (vgl. Strube et al. 2001). Data Mining bezeichnet ganz allgemein den “Prozeß, Muster in Daten zu erkennen” (Witten 2000:3).

1.3 Anwendungsmöglichkeiten

Generell sind zwei Arten der Anwendung von Informationsextraktion denkbar: Zum einen können die extrahierten Daten sofort für einen menschlichen Betrachter gedacht sein. In diesen Anwendungsbereich fällt etwa das von Euler (2001a) zu Testzwecken entwickelte System, das aus E-Mails extrahierte Informationen als SMS weiterleitet, oder ein System, das in einer Suchmaschine zu den Treffern extrahierte Informationen anzeigt, etwa die angebotenen Positionen in Stellenanzeigen. Zum anderen können die Daten für die maschinelle Weiterverarbeitung gedacht sein, sei es zur Speicherung in Datenbanken, zur Textkategorisierung oder -klassifikation

[1] Die umfassende automatische Textzusammenfassung ist insofern problematisch, als daß auch menschliche Leser bei der Aufgabe, das Wichtigste eines Textes zusammenzufassen, nie völlige Übereinstimmung erzielen werden, wenn nicht spezifiziert wurde, *inwiefern* die Informationen wichtig sein sollen.

oder als Ausgangspunkt für eine umfassende Textzusammenfassung. Bestehen die gesuchten Informationen aus mehreren Einzelinformationen, bestimmt das Anwendungsgebiet gewisse Ansprüche an das Informationsextraktionssystem. So müssen zu einer maschinellen Weiterverarbeitung die Informationen strukturiert vorliegen, während für eine Weiterverarbeitung direkt durch den Menschen auch ein unstrukturiertes Ergebnis genügen kann. Wenn die gesuchten Informationen nicht aus weiteren Einzelinformationen bestehen, wie bei der Erkennung von Eigennamen (s. Abschnitt 2), ist eine solche Unterscheidung überflüssig.

1.4 Evaluationskriterien

Zur Bewertung (Evaluation) von Informationsextraktionssystemen werden die im Information Retrieval gebräuchlichen Kriterien Vollständigkeit (Recall) und Präzision (Precision) bzw. das aus diesen Werten ermittelte F-Maß verwendet. Bei einer Extraktion von Informationen der Art 'a' aus einer Menge von Informationen {aaaab} mit dem Ergebnis {aab} läge die Vollständigkeit bei $R = \frac{2}{4}$, da von den vier Informationen vom Typ 'a' in der Ausgangsmenge zwei extrahiert wurden, die Präzision würde $P = \frac{2}{3}$ betragen, da von drei Elementen in der Ergebnismenge zwei der gewünschten Art von Information entsprechen. Ein weiteres Kriterium zur Bewertung der Güte des Extraktes ist der Anteil der unerwünschten Informationen (Fall-out), hier $\frac{1}{3}$, da eine von drei extrahierten Informationen nicht vom gesuchten Typ ist. Das F-Maß erlaubt eine einheitliche Betrachtung der Gütekriterien Vollständigkeit und Präzision: $F = \frac{2RP}{R+P}$

2 Eigennamenerkennung und -klassifikation

2.1 Ziele

Eine vergleichsweise einfache Aufgabe für ein Informationsextraktionssystem ist die Eigennamenerkennung und -klassifikation. Das gewünschte Resultat einer solchen Erkennung und Klassifikation könnte etwa so[2] aussehen:

> *<Name Typ=Person> Stefan Ernst </Name>, Mitarbeiter der <Name Typ=Firma> IBM </Name>, fuhr nach <Name Typ=Ort> Köln </Name>.*

Ein solches Resultat eignet sich auch als erste Stufe der Extraktion eines Ereignisses (vgl. Abschnitt 3.3.1) und wird etwa im System FASTUS so verwendet (s. Appelt et al. 1993 und Abschnitt 3.5.1).

2.2 Vorgehensweise

Am Beginn der Informationsextraktion steht hier die Regel, die das Muster der zu extrahierenden Information beschreibt. Ein einfacher Ansatz zur Identifizierung von Eigennamen wäre etwa, alle Wörter mit großem Anfangsbuchstaben als Namen zu bewerten (Appelt und Israel 1999).
Zu einer auf die so erfolgte Identifizierung aufbauenden Kategorisierung – etwa in die Kategorie 'Firma' – könnte die Übereinstimmung einer Wortgruppe mit dem durch folgenden regulären Ausdruck beschriebenen Muster überprüft werden:

[2] Hier und in nachfolgenden Beispielen in der 'Standard Generalized Markup Language' (SGML)

```
name + ("ag"|"gmbh"|"gbr"|"& söhne"|"& co")
```

Auf diese Weise würde ein oder mehrere Namen, gefolgt von einer der genannten Zeichenketten als Name vom Typ 'Firma' erkannt werden. Dieses Beispiel deckt natürlich nicht alle Formen von Firmennamen ab, es soll lediglich zur Verdeutlichung einer grundsätzlichen Herangehensweise dienen. Als weiterer Ausbau könnte etwa eine Liste geläufiger Abkürzungen von Firmennamen eingebaut werden, um Kurzformen wie 'IBM' ebenfalls abzudecken, sowie ein Mechanismus, der dafür sorgt, daß Koreferenzen – etwa von 'IBM' und 'Big Blue' – erkannt werden. Für eine ausführlichere Darstellung solcher grundsätzlicher Überlegungen zur Eigennamenerkennung und -klassifikation siehe Grishman (2003).

3 Informationsextraktion mit strukturierter Ausgabe

3.1 Ziele

Die Entwicklung auf dem noch recht jungen Forschungsgebiet der Informationsextraktion wurde maßgeblich durch die 'Message Understanding Conferences' (MUC) vorangetrieben. Die sieben MUC wurden von 1987 bis 1997 von der 'Defense Advanced Research Projects Agency' (DARPA) – der zentralen Forschungs- und Entwicklungseinrichtung des US-amerikanischen Verteidigungsministeriums – veranstaltet. Vorgegebene Szenarios waren Nachrichten über nautische Operationen (MUC-1 1987 und MUC-2 1989), über terroristische Aktivitäten (MUC-3 1991 und MUC-4 1992), Joint Ventures und Mikroelektronik (MUC-5 1993), Personalwechsel in der Wirtschaft (MUC-6 1995), sowie über Raumfahrzeuge und Raketenstarts (MUC-7 1997) (Appelt und Israel 1999). Da zur gemeinsamen Evaluation ein standardisiertes Ausgabeformat notwendig war, verwendete man ab der zweiten MUC eine gemeinsame Ausgabeschablone (Template), weshalb nahezu alle[3] Informationsextraktionssysteme eine strukturierte Ausgabe der extrahierten Informationen leisten. Ein zu analysierender Text mit gesuchten Informationen sowie die daraus erstellte Schablone könnten etwa wie folgt aussehen:

> *Der Präsident nimmt es in diesen Januartagen auf sich, die Nation auf Krieg einzustimmen. Diesmal ist er in die weite Ödnis seines geliebten Texas gereist, zum 3. Gepanzerten Korps, das sich selbst 'America's Hammer' nennt.*
>
> (DIE ZEIT, 16. Januar 2003)

Domäne	Politische Nachrichten
Szenario	Besuch
Besucher	Der Präsident
Besuchter	3. Gepanzertes Korps *America's Hammer*
Zeit	Januar
Ort	Texas

[3] Eine Ausnahme hierzu bildet Euler (2001a, 2001b, 2002), s. dazu Abschnitt 4.

3.2 Vorgehensweise

Bei heutigen System werden meist Regeln zum Füllen eines einzelnen Feldes der Ausgabeschablone verwendet[4], allerdings wird auch an Systemen gearbeitet, die ganze Schablonen füllen (Neumann 2001). Die Regel entscheidet, welche Textpassagen extrahiert werden und – wenn die Regel eine komplette Schablone füllt – auf welches Feld der Schablone sie gehören. Bevor Regeln zum Füllen der Felder einer Schablone festgelegt werden können, muß zunächst das Format (d.h. die Felder der Ausgabeschablone) bekannt sein, schließlich sollen die Regeln festlegen, welches Feld mit welchem Teil des zu untersuchenden Textes gefüllt werden soll. Das Vorgehen gliedert sich also in drei Schritte: Am Anfang steht die Festlegung der Ausgabeschablone, dann müssen entsprechende Regeln formalisiert werden, die schließlich zum Füllen der Schablone mit Werten aus analysiertem Text verwendet werden. Zu den benötigten Regeln kann man auf verschiedene Arten gelangen.

3.3 Regelerstellung

3.3.1 Manuelle Regelerstellung

Die Muster der gesuchten Informationen können manuell entdeckt und formalisiert werden. Ein System, das für das Szenario ”Besuch” etwa *Der Präsident besucht die Truppen* als gesuchte Information erkennt und daraus die Ausgabeschablone

Besucher	Der Präsident
Besuchter	die Truppen

erstellt, könnte dies im einfachsten Fall mithilfe einer Regel, wie sie durch den regulären Ausdruck `name ”besucht” name` repräsentiert wird. Zur Identifikation eines Namens könnte zunächst eine Eigennamenerkennung (s. Abschnitt 2) durchgeführt werden. Diese Regel beschreibt allerdings nur einen winzigen Ausschnitt aus der Menge von Möglichkeiten, einen Besuch zu beschreiben – so werden hier nicht einmal verschiedene Formen des Verbs *besuchen* berücksichtigt. Das weitere Vorgehen bestünde nun prinzipiell in einer sukzessiven Verbesserung der Regel durch wiederholtes Testen und Evaluation der Ergebnisse.
Die Entdeckung der Muster ist in diesem Beispiel für jeden Entwickler eines Informationsextraktionssystems möglich, werden die Szenarios allerdings fachlich komplexer, muß der Entwickler Zugang zum benötigten Fachwissen bekommen, etwa durch Lexika oder Berater.

3.3.2 Automatische Regelerstellung aus annotiertem Text

Die Regeln können automatisch aus mit Anmerkungen versehenem (annotiertem) Text erstellt werden. Dabei werden dem System die gewünschte Ausgabeschablone sowie Text, in dem Wörter entsprechend den Werten der Ausgabeschablone markiert wurden, vorgegeben:

Besucher	
Besuchter	

> *Der <Besucher> Präsident </Besucher> besucht die <Besuchter> Truppen </Besuchter>.*

[4] Dies entspricht dem Vorgehen bei einer Eigennamenerkennung (s. Abschnitt 2), denn jede Regel beschreibt eine gesuchte Art von Information.

Die Ausgangsfrage für das automatische Lernen einer Regel zum Erkennen eines solchen Szenarios lautet nun: Welches Muster findet man an den Textstellen, die die relevanten Informationen (Besucher und Besuchter) ausmachen, an dem später in nicht-annotierten Texten das Szenario wiedererkannt werden kann? Der Umfang an Daten, in denen nach einem solchen Muster gesucht werden kann hängt von der Tiefe der in einem nächsten Schritt zu leistenden Analyse der Textpassage ab. Angenommen, das System verfügt über Möglichkeiten zur Analyse auf morphologischer, syntaktischer und semantischer Ebene, wäre etwa folgendes Analyseergebnis des Satzes denkbar:

```
<S>
 <NP Kasus=Nominativ>
  <Det> Der </Det>
  <N Numerus=Singular Typ=Person> Präsident </N>
 </NP>
 <VP>
  <V Numerus=Singular Stamm=”besuch”> besucht </V>
  <NP Kasus=Akkusativ>
   <Det> die </Det>
   <N Typ=Person> Truppen </N>
  </NP>
 </VP>
</S>
```

Modelliert man die Konstituenten des Satzes als Objekte mit Zugriffmöglichkeiten auf die zugehörigen Daten, wäre folgende (in Java formulierte) Funktion zum Extrahieren der gesuchten Information denkbar:

```
void extrahiereInformation(Satz satz, Schablone ergebnis){
  if(satz.np.istNominativ()
  && satz.vp.v.hatStamm(“besuch”)
  && satz.vp.np.istAkkusativ()
  && satz.vp.np.n.istPerson()){
    ergebnis.bestimmeAlsBesucher(satz.np);
    ergebnis.bestimmeAlsBesuchter(satz.vp.np);
  }
}
```

Die Funktion überprüft, ob der zu analysierende Satz den Kriterien aus dem annotierten Korpus entspricht, d.h. ob die Regel hier angewendet werden kann. Ist dies der Fall, werden die entsprechenden Konstituenten des Satzes den entsprechenden Feldern der Schablone zugeordnet. Wirkliche Beschreibungen von Vorgängen wie einem Besuch, etwa das Beispiel aus Abschnitt 3.1 (*Der Präsident nimmt es in diesen Januartagen auf sich, die Nation auf Krieg einzustimmen. Diesmal ist er in die weite Ödnis seines geliebten Texas gereist, zum 3. Gepanzerten Korps, das sich selbst ’America’s Hammer’ nennt.),* sind ungleich komplexer und schwieriger zu modellieren (vgl. Abschnitt 3.4).

3.3.3 Automatische Regelerstellung aus nicht-annotiertem Text

Bei Ansätzen, automatisch aus nicht-annotiertem Text Regeln zu erstellen (s. Grishman et al. 2000), werden dem System zwei oder drei Regeln vorgegeben. Nun sucht das System in einem Trainingskorpus nach Dokumenten, in denen Muster, die diesen vorgegebenen Regeln entsprechen, besonders häufig vorkommen[5]. In der Annahme, daß diese Dokumente insgesamt relevante Informationen enthalten, werden weitere in diesen Dokumenten enthaltenen Muster extrahiert und in Form von Regeln vom System verwendet.

3.3.4 Vor- und Nachteile von automatischer und manueller Regelerstellung

Sowohl die manuelle als auch die automatische Regelerstellung haben Vor- und Nachteile: Die manuelle Erstellung ist aufgrund der sukzessiven Anpassung der Regeln an das Szenario sehr zeit- und arbeitsaufwändig – dafür bietet sich die Möglichkeit, ein sehr gut angepaßtes System zu entwickeln und damit ein hohes F-Maß zu erreichen. Außerdem lassen sich Regeln für Szenarios entwickeln, für die es keine Korpora gibt, oder diese (etwa aus Kostengründen) nicht zur Verfügung stehen. Allerdings sind solche Systeme auf das Szenario beschränkt, für das sie entwickelt wurden und eine Anpassung an veränderte Erfordernisse kann unmöglich sein: Appelt und Israel (1999) nennen etwa ein System zur Eigennamenerkennung, das so realisiert wurde, daß es Wörter in Großschreibung als Eigennamen einstuft. Hier kann eine Implementierung von Texten in Kleinschrift eine komplette Neuentwicklung erfordern, während ein System, das seine Regeln automatisch lernt, einfach einen Korpus in Kleinschrift bekommen könnte. Automatische Regellernsysteme haben also den Vorteil, auf neue Domänen oder Szenarios portierbar zu sein, allerdings kann der Annotierungsaufwand je nach gewähltem Verfahren sehr hoch sein, etwa wenn nicht länger nur Namen von Politikern sondern auch von politischen Institutionen von einem System zur Namensklassifikation als zum Typ 'politisch' zugehörig erkannt werden sollen, und so im kompletten Trainingskorpus die Anmerkungen erweitert werden müßten. Der Aufwand kann je nach neuem Einsatzgebiet bei manueller Regelerstellung viel geringer sein, da hier die Regel direkt bearbeitet werden kann. Zur Qualität der Ergebnisse bemerkt Grishman, daß automatische Lernverfahren bei der Verarbeitung von Texten mit regelmäßiger, sich wiederholender Struktur gute Ergebnisse liefern, in Bereichen mit größerer linguistischer Variation aber hinter Systemen mit manuell erstellten Regeln zurückbleiben (Grishman 2003).

3.4 Regelbasierte Extraktion

Die grundsätzliche Vorgehensweise bei der eigentlichen Extraktion besteht darin, die zu bewertende Textstelle linguistisch so tief zu analysieren, daß die verfügbare Regel anwendbar ist. Um den Satz *Der Präsident besucht die Truppen* mit der in Abschnitt 3.3.2 beschriebenen Regel als relevant erkennen zu können, müßte die linguistische Analyse etwa die Erkennung von Phrasen, eine Stammformenreduktion und die Ermittlung des Numerus umfassen. Zudem müßte ein Mechanismus zur Ermittlung der semantischen Kategorien von *Präsident* und *Truppen* gegeben sein, etwa durch Zugriff auf ein Lexikon oder eine vorherige Eigennamenerkennung und -klassifikation. Handelt es sich dagegen um ein reales Beispiel – wie *Der Präsident nimmt es*

[5] Dies ist eine Volltextsuche. Hier zeigt sich, wie eng die Informationsextraktion mit ihren Nachbargebieten verbunden ist, ob wie hier diese als Hilfsmittel einsetzend oder als Vorverarbeitung für diese, wie in Abschnitt 1.3 erwähnt.

in diesen Januartagen auf sich, die Nation auf Krieg einzustimmen. Diesmal ist er in die weite Ödnis seines geliebten Texas gereist, zum 3. Gepanzerten Korps, das sich selbst 'America's Hammer' nennt. – ist jedoch weiteres nötig. Hier sind vor allem zwei Probleme zu nennen: Zum einen die Auflösung anaphorischer Ausdrücke[6] – hier etwa pronominale Koreferenzen (*Der Präsident, er*) und Eigennamen-Koreferenzen (*3. Gepanzertes Korps, America's Hammer*) – zum anderen das Zusammenführen partieller Instanzen, wie sie hier bei einer Analyse einzelner Sätze entstehen würden:

Szenario	
Besucher	Der Präsident
Besuchter	
Zeit	Januar
Ort	

Szenario	Besuch
Besucher	er
Besuchter	3. Gepanzertes Korps America's Hammer
Zeit	
Ort	Texas

Nach der Analyse des ersten Satzes wäre die Information unvollständig, während die Schablone für den zweiten Satz im Feld 'Besucher' lediglich einen referenzierenden Wert hat. Dies sind nur einige der Schwierigkeiten auf dem Gebiet der Informationsextraktion, weitere ergeben sich etwa aus morphologischen Herausforderungen (z.B. die Zerlegung von Komposita, wie sie hier nötig wäre, um aus *Januartagen* im ersten Satz den Zeitpunkt 'Januar' zu erkennen) und auf dem Gebiet der Pragmatik, z.B. lediglich indirekt geäußerte Informationen zu erkennen – etwa daß im Bereich 'Terminabsprachen' *Mir ist etwas dazwischen gekommen* eine Absage bedeutet.

3.5 Beispiele und Evaluation

3.5.1 Gemeinsamkeiten der Systeme

Die im Rahmen der MUC entwickelten und evaluierten Systemen haben neben der Ausgabeschablone auch die grundsätzliche Anforderung gemeinsam, große Textmengen schnell verarbeiten zu können. Systeme, die hohe F-Werte liefern, aber lange für die Analyse brauchen, waren im Rahmen der Domänen sämtlicher MUC (verschiedenen Arten von Nachrichten) unbrauchbar. Da selbst im Rahmen der heutigen, unvollständigen Kenntnisse der Funktionsweise natürlicher Sprache eine komplette linguistische Analyse hier zu aufwändig wäre, gehen Informationsextraktionssysteme üblicherweise einen Kompromiß bezüglich der Komplexität der Analyse zugunsten der Verarbeitungsgeschwindigkeit ein. So hat etwa die Verwendung von endlichen Automaten hier eine wahre "Renaissance" (Neumann 2001) erfahren. So verwenden etwa die Systeme FASTUS[7] und ANNIE[8] kaskadierte endliche Automaten zur Analyse. Viele Systeme haben einen modularen Aufbau, bei dem einzelne, domänenunabhängige linguistische Analyseschritte voneinander – und diese von den domänenspezifischen Modulen – getrennt sind. FASTUS etwa bestand 1993 aus Modulen zur Erkennung von komplexen Wörtern und Eigennamen, von einfachen und komplexen Phrasen, zur Erkennung der gesuchten Ereignisse und zum Zusammenführen von partiellen Ergebnissen (s. Appelt et al. 1993). Im

[6] Die Auflösung anaphorischer Ausdrücke ist ein Thema für sich, s. hierzu Mitkov (2003).

[7] Eine veränderte Abkürzung für 'Finite State Automaton Text Understanding System', entwickelt von SRI International.

[8] Eine Komponente der 'General Architecture for Text Engineering' (GATE), die an der University of Sheffield entwickelt wurde.

Gegensatz dazu verfügt ANNIE etwa über Module zum Zerlegen des Textes in Wörter (Tokenizer), zum Zurückführen von Wörtern auf ihre Grundformen (Lemmatisierer) und zur Wortartenbestimmung mittels eines Part-of-Speech-Taggers sowie eines semantischen Taggers[9]. Dies zeigt andeutungsweise, wie stark der Umfang der linguistischen Analyse von System zu System variiert.

3.5.2 Evaluation

Im Bereich der Eigennamenerkennung wurden im Rahmen der MUC Ergebnisse von F-Werten bis zu 90% erreicht. Bei den evaluierten Szenarios, die Ereignisse beschreiben, wurden dagegen nie Werte jenseits der 60% erreicht (Appelt und Israel 1999, Grishman 2003). Dies könnte darauf hindeuten, daß eine seichte Sprachanalyse, wie sie aus oben beschriebenen Gründen verwendet wurde, nicht ausreicht, um solch komplexe sprachliche Konstrukte zu erkennen, allerdings schreibt Grishman, daß auch die Verwendung von ”more powerful analysis techniques – what could be described as a ’deep understanding’ model” (Grishman 2003) lediglich an ausgewählten Beispielen gute Leitungen bringt, nicht aber bei der allgemeinen Anwendung.

4 Informationsextraktion durch gezielte Zusammenfassung

4.1 Ziele

Ist eine strukturierte Ausgabe nicht erforderlich, ist ein anderer Ansatz möglich: Die Informationsextraktion durch gezielte Zusammenfassung (s. Euler 2001b). Hierbei würde aus dem Beispiel in Abschnitt 3.1 statt der Schablone ein gekürzter Text, etwa in folgender Form: ”Präsident [...] Texas gereist [...] zum 3. Gepanzerten Korps [...]”. Denkbar wäre auch, daß lediglich die zwei Sätze, aus denen der Beispieltext besteht, aus dem kompletten Artikel extrahiert werden. Eine solche Ausgabe, versehen mit Hinweisen über Art und Umfang der Auslassungen, kann für bestimmte Bereiche (s. Abschnitte 1.3 und für eine konkrete Anwendung 4.5) ausreichend sein.

4.2 Die semantische Richtschnur: Wortlisten

Ausgangspunkt für die Informationsextraktion durch gezielte Zusammenfassung stellen Wortlisten dar, in denen allen Wörtern ein Gewicht entsprechend Ihrer Bedeutung für das Szenario, das erkannt werden soll, zugeordnet ist. Solche Wortlisten können etwa aus lexikalisch-semantischen Netzen gewonnen werden oder – ähnlich den Regeln bei der Extraktion mit strukturierter Ausgabe – automatisch aus annotiertem Text. Eine solche Liste stellt die semantische Richtschnur dar, die von der Auswahl der relevanten Sätze bis zu einer eventuellen Kürzung dieser eine ”gewisse semantische Orientierung” (Euler 2001b) bietet.
Zum automatischen Erstellen von Wortlisten müssen im Lernkorpus relevante Informationen gekennzeichnet werden, allerdings reicht es hier bereits, ganze Sätze oder sogar Dokumente[10] zu kennzeichnen, da die Struktur der zu extrahierenden Informationen – etwa welches Wort bei einem Besuch den Besucher und welches den Besuchten bezeichnet – für das Vorgehen des

[9] Für weitere Informationen zu den Modulen von ANNIE siehe Cunningham et al. (2003).

[10] Zu Ergebnissen mit Korpora, in denen Sätze oder ganze Dokumente gekennzeichnet wurden siehe Abschnitt 4.5.

Systems nicht relevant ist. Wollte man etwa Informationen über Raketenstarts (das Szenario von MUC-7) extrahieren, könnte ein Ausschnitt aus dem annotierten Korpus so aussehen:

> *Ein weiterer Fehlstart hätte möglicherweise sogar das Ende für die Fertigung großer Trägerraketen in Westeuropa bedeuten können. <Raketenstart> Die insgesamt 15. Ariane-5 verließ die Startrampe 3A in Kourou um 0:52 MEZ und verschwand Sekunden später in einer dichten Wolkenschicht. </Raketenstart> Eine gute halbe Stunde später hatte die Ariane-5 ihren Zielorbit erreicht...*
>
> (Aus einer Meldung auf <http://www.vfr.de/>)

Anhand eines in dieser Art gekennzeichneten Korpus erstellt das System eine Rangliste der wichtigsten[11] Wörter für das Szenario. Euler (2001b) hat vor der Gewichtung der einzelnen Wörter eine Stammformenreduktion durchgeführt, ohne diese habe er "keine guten Ergebnisse erzielt". An dieser Stelle wäre auch eine weitergehende linguistische Vorverarbeitung denkbar, etwa die schon in Abschnitt 3.4 erwähnte Zerlegung von Komposita. Die am höchsten bewerteten Wörter im Korpus (bei Euler 10%) bilden nun die für das Szenario relevante Wortliste.

4.3 Extraktion durch gezielte Zusammenfassung

4.3.1 Ermittlung relevanter Sätze

Bei der eigentlichen Extraktion bekommt jedes Wort im zu untersuchenden Text den Wert, der ihm in der Liste zugeordnet ist, bzw. keinen Wert, wenn es nicht in der Liste steht. Im nächsten Schritt werden die Werte der Wörter eines Satzes addiert und dieser Wert durch die Anzahl der Wörter im Satz geteilt, um unterschiedlich lange Sätze gleich zu behandeln. Nun kann durch Versuche ein Schwellenwert bestimmt werden, den ein Satz erreichen muß, um als Treffer zu gelten bzw. für eine weitere Kürzung interessant zu sein.

4.3.2 Satzkürzungen

Ist eine weitere Kürzung der Ergebnisse gewünscht, können die Sätze in mehreren Stufen gekürzt werden. In einer ersten Stufe werden "übliche Abkürzungen" (Euler 2001b) eingeführt sowie Grußformeln, Anreden und "meistens inhaltsleere Wörter wie *naja* und *überhaupt*" (Euler 2001b) gestrichen. Auf der nächsten Stufe werden Artikel, Adverbien und Adjektive gestrichen, auf der Stufe mit der maximalen Kürzung schließlich ganze Phrasen, außer Verbalphrasen. Phrasen, die Wörter aus der Liste enthalten sowie Wörter, die in der Liste enthalten sind, werden nie gestrichen, wodurch die Wortliste auch hier eine Art semantische Richtschnur bildet. Die beschriebenen Satzkürzungen erfordern zumindest teilweise eine linguistische Analyse der zu kürzenden Sätze, etwa zur Phrasenstrukturermittlung und Wortartenbestimmung. Hierzu verwendete Euler das sprachverarbeitende System MESON des Deutschen Forschungszentrums für Künstliche Intelligenz (DFKI).

[11] Euler erreichte die besten Ergebnisse über die Worthäufigkeit. Vergleichsverfahren waren Information Gain, G^2-Statistik und SVM-Gewichte (s. Euler 2001b).

4.4 Vor- und Nachteile der Extraktion durch gezielte Zusammenfassung

Der wichtigste Unterschied zum Verfahren mit strukturierter Ausgabe in Form von Ausgabeschablonen ist die Tatsache, daß die Ergebnisse nicht strukturiert sind. Wenn die Ergebnisse sofort von einem Menschen verarbeitet werden sollen ist dies aber nicht notwendigerweise ein Nachteil, sondern kann den Anforderungen angemessen sein. Bei den Ergebnissen der gezielten Zusammenfassung wird die Verknüpfung der Elemente eines Vorganges – etwa von *Ariane-5, Kourou* und *verließ* zu dem Vorgang "Raketenstart" – vom Menschen vorgenommen. Der menschliche Rezipient erkennt hier aufgrund seines Hintergrundwissens, daß *Ariane-5* die Rakete ist – das System speichert lediglich, daß diese Wörter etwas mit einem Raketenstart zu tun haben. Die Vorteile, die durch diese Einschränkung gewonnen werden, sind der geringe Annotierungsaufwand beim Erstellen des Korpus (und daher zumindest theoretisch eine hohe Portabilität auf neue Szenarios) sowie die Möglichkeit der Anpassung von Vollständigkeit und Präzision an die konkreten Erfordernisse – auf Ebene der Satzauswahl durch den Schwellenwert, ab dem ein Satz als Treffer ausgezeichnet wird und auf Ebene der Satzkürzung durch die gewählte Kürzungsstufe.

4.5 Beispiel und Evaluation

Euler hat das beschriebene Verfahren anhand eines E-Mail-to-SMS-Service getestet. Der Dienst soll E-Mails, die Terminabsprachen (Ankündigungen, Verschiebungen, Ab- und Zusagen) enthalten, kürzen und sie als SMS an ein Mobiltelefon verschicken. Der verwendete Korpus umfaßte 560 deutschsprachige E-Mails mit insgesamt ca. 45000 Wörtern. Der Korpus bestand zur Hälfte aus E-Mails, die Informationen vom gesuchten Szenario 'Terminabsprache' enthielten und zur anderen Hälfte aus beliebigen E-Mails. "Etwa 13%" (Euler 2001b) der Sätze des Korpus wurden als terminbezogen gekennzeichnet. Für das Training wurden 90% des Korpus verwendet, zum Testen die übrigen 10%. Beim reinen Satzfiltern ohne weitere Kürzungen erreichte Euler F-Werte bis zu 81,2% mit der beschriebenen Markierung relevanter Sätze im Trainingskorpus und bis zu 76,5%, wenn relevante Dokumente (d.h. E-Mails) gekennzeichnet wurden. Eine Evaluation der in Abschnitt 4.3.2 beschriebenen Satzkürzungen durch Befragung der Empfänger der SMS-Nachrichten ergab, daß Informationen wie der Zeitpunkt des Termins und sein Status, d.h. ob er "ausfällt oder verschoben wird etc." "am besten erhalten" (Euler 2001b) bleiben. Die Art des Termins gehe oft verloren, wäre aber für einen Empfänger mit Hintergrundwissen rekonstruierbar gewesen[12](Euler 2001b). Euler kommt bezüglich der Kürzungsstufen zu folgendem Fazit: "Bei mittlerer Kürzung, die nicht mehr die Phrasenentfernung durchführt, sollte in den meisten Fällen Halt gemacht werden" (Euler 2001b). Später hat er die Stufen der Satzkürzung weiter ausdifferenziert (s. Euler 2002). Die guten Ergebnisse nach F-Maß selbst beim Kennzeichnen ganzer Dokumente im Trainingskorpus lassen einfach handhabbare und leistungsfähige Systeme denkbar erscheinen. So könnte im E-Mail-Client jede E-Mail auf Knopfdruck einer bestimmten Kategorie zugeordnet werden und so für verschiedenen Bereiche Wortlisten mit geringem Aufwand erstellt werden. Denkbar ist für Euler (2001b) sogar eine ganz allgemeine Kennzeichnung der Nachrichten in 'interessant' und 'uninteressant' und somit ein auf das Interesse des Benutzers abgestimmtes System. Allerdings merkt Euler an, daß das Verfahren "noch für andere Domänen und andere Sprachen getestet werden" (Euler 2001b) muß.

[12] Zur Rolle des Hintergrundwissens vgl. Abschnitt 4.4.

5 Schlußbetrachtung

In der Informationsextraktion bietet sich auf mehreren Ebenen ein vielfältiges Bild. Informationsextraktionssysteme können für verschiedene Aufgabenbereiche von der automatischen Analyse von Stellenanzeigen bis zur Vorbereitung einer allgemeinen Textzusammenfassung eingesetzt werden. Entsprechend diesen Anforderungen können die Systeme strukturierte oder unstrukturierte Ergebnisse liefern. Weiter können die Systeme völlig unterschiedliche linguistische Tiefe aufweisen, von der Extraktion durch gezielte Zusammenfassung mit reiner Satzfilterung, wo lediglich semantische Orientierung in Form der Wortliste gegeben ist, bis hin zu Systemen mit Analysemodulen für sämtliche Ebenen der Sprache. In einigen Bereichen führt unser mangelndes Verständnis für die Funktionsweise natürlicher Sprache zu einer Stagnation der Entwicklung, doch da Informationsextraktion eine eingeschränktere Aufgabe als ein komplettes Textverständnis darstellt, sind vielfach im Sinne eines ”appropriate language engineering” (Grishman 2003) den Anforderungen angemessene Lösungen (vielleicht auch gerade in Verbindung mit den Nachbargebieten) möglich. Als Beispiel hierfür möge das von Euler (2001a, 2001b, 2002) entworfene Verfahren dienen, das im Unterschied zu den die IE dominierenden Systemen lediglich unstrukturierte Ergebnisse liefert. Dafür erreicht es hohe Leistung nach F-Maß und verlangt lediglich einen geringen oder gar minimalen Annotierungsaufwand des Trainingskorpus, was eine hohe Portabilität auf neue Domänen und Szenarios bedeuten könnte, etwa in Form einer Erstellung von Wortlisten *en passant* bei einer Textklassifikation.

Literatur

Appelt, Douglas; John Bear, Jerry Hobbs, David Israel, Megumi Kameyama, Mark Stickel, Mabry Tyson (1993) *FASTUS: A Cascaded Finite-State Tranducer for Extracting Information from Natural-Language Text,* Sri International. 8. September 2003 <http://www.ai.sri.com/~appelt/fastus-schabes.html/>.

Appelt, Douglas & David Israel (1999) *Introduction to Information Extraction Technology. A Tutorial Prepared for IJCAI-99,* SRI International. 8. September 2003 <http://www.ai.sri.com/~appelt/ie-tutorial/>.

Cardie, Claire (1997) "Empirical Methods in Information Extraction" in *AI Magazine*, Vol. 18, 4, 65-68. 8. September <http://citeseer.nj.nec.com/cardie97empirical.html>.

Cunningham, Hamish; Diana Maynard, Kalina Bontcheva, Valentin Tablan, Cristian Ursu, Marin Dimitrov (2003) *Developing Language Processing Components with GATE (a User Guide),* University of Sheffield. 8. September 2003 <http://gate.ac.uk/sale/tao/tao.pdf/>.

Euler, Timm (2001a) *Informationsextraktion durch Zusammenfassung maschinell selektierter Textsegmente,* Universität Dortmund. 8. September <http://www-ai.cs.uni-dortmund.de/dokumente/euler2001a.pdf/>.

— (2001b) *Informationsextraktion durch gezielte Zusammenfassung von Texten,* Universität Dortmund. 8. September <http://www-ai.cs.uni-dortmund.de/events/fdml2001/FGML2001-Euler-Paper.pdf/>.

— (2002) "Tailoring Text using Topic Words: Selektion and Compression" in *Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA),* IEEE Computer Society Press. 8. September 2003 <http://www-ai.cs.uni-dortmund.de/dokumente/euler2002.pdf/>.

Grishman, Ralph; Silja Huttunen, Pasi Tapanainen, Roman Yangarber (2000) "Unsupervised Discovery of Scenario-Level Patterns for Information Extraction" in *Proceedings of the Conference on Applied Natural Language Processing ANLP-NAACL2000,* Seattle. 282-289. 8. September 2003 <http://citeseer.nj.nec.com/yangarber00unsupervised.html/>.

Grishman, Ralph (2003) "Information Extraction" in Mitkov, Ruslan et al., *The Oxford Handbook of Computational Linguistics,* Oxford University Press. 545-559.

Mitkov, Ruslan (2003) "Anaphora Resolution" in Mitkov, Ruslan et al., *The Oxford Handbook of Computational Linguistics,* Oxford University Press. 267-283.

Neumann, Günter (2001) "Informationsextraktion" in Carstensen, Kai-Uwe et al. *Computerlinguistik und Sprachtechnologie. Eine Einführung,* Heidelberg, Berlin: Spektrum. 448-455.

Strube, Gerhard u.a. (Hrsg.) (2001) *Digitales Wörterbuch der Kognitionswissenschaft,* Klett-Cotta.

Witten, Ian & Eibe Frank (2000) *Data Mining - Praktische Werkzeuge und Techniken für das maschinelle Lernen,* Hanser.