LUIS – Ein natürlichsprachliches, universitäres Informationssystem

Toni Arnold, Simon Clematide, Roberto Nespeca, Jeannette Roth, Martin Volk

Institut für Informatik
Gruppe Computerlingustik
Winterthurerstr. 190
CH-8057 Zürich
{tarnold,siclemat,roberto,jroth,volk}@ifi.unizh.ch

1 Einleitung

Noch Mitte der 90er-Jahre war eine Studentin der Universität Zürich genötigt, den (u. U. weiten Weg) ins Hauptgebäude der Universität auf sich zu nehmen, wenn sie erfahren wollte, in welchem Gebäude und Raum eine Vorlesung stattfand. Dort kämpfte sie dann mit ihren Leidensgenossinnen und -genossen vor dem schwarzen Brett um einen Platz mit Sicht auf dasselbe – um vielleicht zu erfahren, dass sie noch gar nicht hätte anzureisen brauchen, weil ihre Vorlesungen erst in der kommenden Woche begannen.
Heute ist das zum Glück anders – Internet sei Dank. Die Studentin kann auf der Homepage der Universität das Online-Vorlesungsverzeichnis abrufen, dem auch kurzfristig bekannt werdende Informationen wie die Raumverteilungen bequem von zu Hause aus entnommen werden können. Mit dem elektronischen Publizieren des Vorlesungsverzeichnisses kommt man den Studierenden einen Schritt entgegen – doch von einem umfassenden Informationssystem kann noch nicht die Rede sein. Die Homepage der Universität Zürich enthält denn auch ergänzend ein breites, aber sehr heterogenes Angebot: Angefangen bei Informationen der Dekanate, Fakultäten und Beratungsstellen über das E-Mail- und Telefonverzeichnis, Gebäude- und Lagepläne, bis hin zur beliebten Rubrik ‚Universitäres Leben', wo Studierende erfahren, wie sie in Zürich ein Zimmer mieten können, welche speziellen Einkaufsmöglichkeiten ihnen geboten werden usw.
Auch eine FAQ, eine Sammlung häufig gestellter Fragen zum Studienbetrieb, steht auf dem WWW zur Verfügung.[1] Dort können Antworten gefunden werden auf Fragen wie: „Wie kann ich mich immatrikulieren?", „Wie hoch sind die Semestergebühren?", „Kann ich mit einem ausländischen Vorbildungsausweis an der Universität Zürich studieren?". In der FAQ kann sich der informationsbedürftige Student von Link zu Link bis zu seiner Frage durchklicken und erhält dort die Antwort – sofern er zuvor seine Frage auch wirklich gefunden hat. Und da liegt die Schwierigkeit: Wer seine Frage in der baumartig aufgebauten FAQ nicht findet, kommt auch zu keiner Antwort. Um diesem Dilemma Abhilfe zu schaffen, haben wir, die Computerlinguistik-Gruppe der Universität Zürich,

[1] Online abrufbar unter: http://www.ifi.unizh.ch/CL/UIS/Kanzlei/FAQ.

das webbasierte Projekt *Little University Information System*, kurz LUIS[2], ins Leben gerufen.

2 Die organisatorische Seite von LUIS

2.1 LUIS – Ein natürlichsprachlicher Informationszugang

LUIS ist ein *stichwortbasiertes Passagenretrieval-System*, das natürlichsprachliche Anfragen entgegennimmt und als „Antwort" Textstellen liefert, die möglichst gut zur Frage passen. Natürlichsprachlich heißt, dass die Frage in den gleichen Worten formuliert wird, wie man sie auch einem Menschen gegenüber formulieren würde. Der Vorteil des natürlichsprachlichen Ansatzes besteht darin, dass die Benutzer weder schwerfällige Stichwort-Recherchen mit Booleschen Operatoren, noch komplizierte Datenbank-Abfragesprachen wie SQL beherrschen müssen. Die Fragen können in vollständigen natürlichsprachlichen Sätzen gestellt werden, so wie das der menschlichen Kommunikation entspricht. LUIS ist verwandt mit Systemen wie *FAQ-Finder* [Bu97] und *AskJeeves*[3].

2.2 Die Wissensbasis von LUIS

Drei Datenbasen dienen als Quellen, denen LUIS die Antwort-Textstellen entnimmt: Zum einen die bereits erwähnte *FAQ*, des Weiteren ein Glossar namens *Von A bis Z*[4], das studiumsspezifische Begriffe (Auslandsemester, Stipendienberatung, Zulassungsbedingungen usw.) erläutert, und eine *Sammlung von Web-Seiten*, vorwiegend der Universität Zürich, aber auch anderer fürs Studium wichtiger Institutionen (z.B. Berufs- und Studienberatung). Die Erstellung und Pflege dieser Datenbasen ist ein wichtiger Aufgabenbereich, um die Funktionstüchtigkeit und Nützlichkeit von LUIS zu gewährleisten.

2.2.1 Die FAQ

Die FAQ erstellten wir mit Hilfe der Verwaltungsmitarbeitenden der Universitätskanzlei (Studentensekretariat). Diese notierten die häufigsten Fragen, die die Studierenden an sie herantrugen, und deren Antworten. Dann sortierten wir die Fragen nach Themengebieten (Auditor-Status, Urlaub, Immatrikulation usw.). In einer späteren Phase erweiterten wir die Frage-Antwort-Sammlung mit Fragen zu Stipendien, zum Informatik-Angebot der Universität, zum Studieren mit Behinderung u.a.m. Weitere Fragen konnten dank einem Modul *Neue Fragen* ermittelt werden, das den Benutzenden seit Bestehen der FAQ ermöglicht, online Fragen zu stellen, die in der FAQ nicht vorhanden sind, und nach zwei

[2] Online abrufbar unter: http://www.ifi.unizh.ch/CL/UIS/LUIS
[3] Online abrufbar unter: http://www.askjeeves.com
[4] Online einsehbar unter: http://www.ifi.unizh.ch/CL/UIS/Studienfuehrer/VonAbisZ

bis drei Tagen von einer Mitarbeiterin der Kanzlei via E-Mail beantwortet zu erhalten. In einer Log-Datei wurden diese neuen Fragen seit Projektbeginn gesammelt und konnten bei der FAQ-Erweiterung genutzt werden. Das Themenspektrum verlangte die Kontaktaufnahme mit verschiedenen Beratungsstellen: Zur Kanzlei kamen die Stipendienberatung, die Informatikdienste, der Beratungsdienst für Behinderte und die Rektorenkonferenz der Schweizer Universitäten hinzu. Obwohl weitere Stellen der Universität Zürich Beratungen anbieten, schlugen sich nur diese vier in der FAQ-Erweiterung nieder. Viele zogen wir auf Grund ihrer Thematik nicht in Betracht (bspw. Medizinische Aidsberatung: zu klein und zu spezifisch ist dieser Themenbereich). Andere begründeten auf unsere Anfrage ihre Absage damit, dass die Fragen, mit denen Studierende an sie herantreten, nur individuell beantwortet werden könnten (bspw. Psychologische Beratungsstelle). Leider gab es auch eine Beratungsstelle, die vor der Arbeit zurückschreckte, die eine Aufstellung der häufigsten Fragen mit sich gebracht hätte – nicht einmal, als wir selbst einige Fragen und Antworten zusammenstellten und um Kontrolle baten, waren die Mitarbeitenden der betreffenden Beratungsstelle zur Zusammenarbeit bereit.

2.2.2 Das Glossar *Von A bis Z*

Erfreulicher zeigt sich die Zusammenarbeit mit der Berufs- und Studienberatung, die uns das von ihr erarbeitete Glossar mit rund 80 Begriffen und Erläuterungen (von A wie Akademischer Sportverband bis Z wie Zulassungsbedingungen) zur Verfügung stellt. Wir erhielten es in elektronischer[5] Form und bereiteten es so auf, dass via WWW die einzelnen Stichworte abgefragt werden können. Wichtig für LUIS war jedoch, dass wir das Glossar als Datenbasis zur Beantwortung von Fragen verwenden durften.

2.2.3 Weitere Web-Seiten

Die letzte Datenbasis, eine Hundertschaft von Web-Seiten der Universität und von verwandten relevanten Sites, war in ihrer Erstellung nicht minder aufwändig als die FAQ. Auch die Pflege derselben erfordert viel Zeit: Etwa halbjährlich muss das universitäre Web auf neue, für LUIS brauchbare Seiten hin durchforstet werden. Die Meldung von „toten“ Web-Seiten konnten wir automatisieren, eine automatische Kontrolle „toter“ Links in der FAQ und dem Glossar wird nächstens realisiert. Doch nicht nur *Links* müssen auf ihre Aktualität geprüft werden, auch muss bei der FAQ und dem Glossar dafür gesorgt werden, dass der *Inhalt* immer dem neuesten Stand entspricht. Diese Aufgabe können wir nicht selbst erfüllen, sondern sind auf die Zusammenarbeit mit den involvierten Beratungsstellen angewiesen. Es gilt somit, den Kontakt zu pflegen und diese zu bitten, uns Änderungen zu melden, die sich auf unsere Datenbasen auswirken.

[5] Zuvor war das Glossar nur in gedruckter Form verfügbar im *Studienführer*, einer Informationsschrift der Studien- und Berufsberatung.

Wichtig ist, dass die informationstragenden Stellen sehen, dass ihre Beiträge zügig in das web-basierte Informationsangebot eingearbeitet werden und sich dadurch für sie eine Arbeitserleichterung ergibt. Ziel ist es, die Verwaltungs- und Beratungsstellen der Universität bei der Beantwortung repetitiver Fragen zu entlasten. Wir erhalten insbesondere von der Kanzlei die Rückmeldung, dass unser Angebot eine hochwillkommene und effiziente Entlastung bringt und die Studierenden besser informiert den Auskunftschalter aufsuchen.

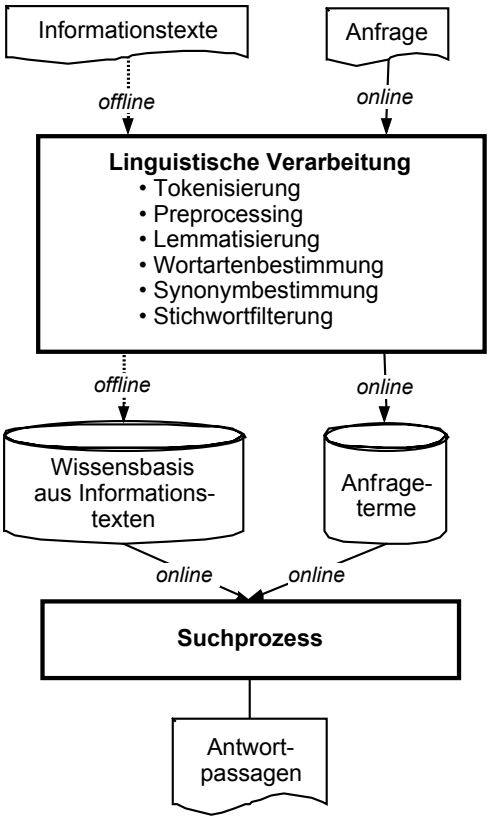

Abbildung 1: Architektur von LUIS

3 Die technische Seite von LUIS

Mit LUIS haben wir einen funktionierenden Prototypen erstellt, der letztlich zu einem Passagenretrieval-System erweitert werden soll, welches eine vertiefte syntaktische und semantische Analyse vornimmt. Der Ansatz lehnt sich an die Antwortextraktions-Systeme [Mo00] an, die in unserer Gruppe für die englische Sprache entwickelt wurden. Die folgende System-Übersicht beschreibt die Architektur von LUIS und das Zusammenspiel der verschiedenen Komponenten. Der Fokus wird dabei auf die linguistische Verarbeitung und den Suchprozess gelegt.

3.1 Überblick

Abbildung 1 zeigt den Ablauf und die Komponenten von LUIS. Ein wichtiges Merkmal ist die transparente und einheitliche linguistische Verarbeitung sowohl der Informationstexte wie der Anfragen, die von den Benutzenden gestellt werden. Die grosse Menge an Informationstexten kann dabei *offline* verarbeitet werden, im Gegensatz dazu wird jede Anfrage *online* verarbeitet. Da es sich dabei um einzelne Sätze handelt, bleibt die Verarbeitungszeit im Rahmen interaktiv nutzbarer Anwendungen. Der Suchprozess selbst muss selbstverständlich ebenfalls *online* erfolgen und liefert als Resultat eine nach Relevanz geordnete Liste von Passagen, welche die erfragte Information enthält. Die Informationstexte erfahren im Wesentlichen die gleiche Verarbeitung wie eine Anfrage. In der Folge gehen wir auf den typischen Ablauf beim Beantworten einer Frage durch das System ein.

3.2 Linguistische Verarbeitung

3.2.1 Tokenisierung und Preprocessing

Die Aufgabe dieser Verarbeitungsstufen ist es, aus einem Strom von einzelnen Zeichenkodes Wortformen und spezifische Texteinheiten (Adressen, Datums- und Zeitabgaben usw.) zu bilden, Sätze und Abschnitte zu segmentieren sowie Überschriften zu erkennen. Obwohl die Informationstexte im HTML-Format vorliegen – im Gegensatz zur Anfrage, die als Reintext eingetippt wird – muss sich der Tokenizer nicht mehr mit der Kodierung und Struktur von HTML-Dokumenten beschäftigen. Diese werden in einer Vorverarbeitung zu einem „flachen“ Format normalisiert, wo von der Dokumentstruktur nur noch die Abschnitts- und Überschriftssegmentierung bleibt.
Wegen der Mehrdeutigkeit von Punktzeichen (Satzendepunkt und/oder Abkürzungspunkt, Auslassungspunkt) wird dem ersten Tokenisieren ein *Preprocessing* nachgeschaltet, das auf Grund von Abkürzungslisten Punktzeichen desambiguieren kann. Die Abkürzung ‚usw.’ würde vom Tokenizer nicht als eine Wortform erkannt, sondern in ‚usw‘ und ‚.‘ aufgeteilt. Das *Preprocessing*-Modul fasst diese beiden Teile zusammen und

expandiert die Abkürzung gleichzeitig, damit bei der Suche die Erwähnung von ‚dipl.' und ‚diplomiert' in Zusammenhang gebracht werden kann.
Daneben erkennt, markiert und normalisiert das Preprocessing spezifische Texteinheiten wie Datums-, Zeit- und Währungsangaben, E-Mail-Adressen, Telefon- und Faxnummern usw. Bei der späteren Suche werden solche Texteinheiten analog zu Fragewörtern wie ‚Was', ‚Wann', ‚Wo' und ‚Wie viel' behandelt. Das Fragewort ‚Wann' wird als unspezifizierte Datumsangabe repräsentiert.
Das *Preprocessing* basiert auf einem *Pattern-Matching*-Ansatz, der diverse gelochte Formraster zur Verfügung stellt und prüft, ob die Lücken mit den erkannten *Tokens* korrekt gefüllt werden können. Für die Raster ist ein eigenes Textformat[6] entwickelt worden, damit das Ergänzen oder Modifizieren dieser Muster ohne Programmierkenntnisse erfolgen kann. Die entsprechenden Programm-Stückchen werden automatisch aus dem Format erzeugt.

3.2.2 Lemmatisierung

Die beim Tokenisieren erkannten Wortformen werden im nächsten Schritt lemmatisiert. Wir verwenden dafür das kommerziell erhältliche Morphologieanalyse-Programm *Gertwol*, das Wortformen auf die Stammform (Lemma) und die entsprechenden morphosyntaktischen Merkmale reduziert. Die Stammform enthält Informationen über Präfigierung und Suffigierung, zudem werden Komposita erkannt und die Kompositionsgrenzen markiert.
Obwohl *Gertwol* – insbesondere dank der Kompositionsanalyse – eine gute Abdeckung erreicht, waren wichtige Fachtermini, die in den Texten der Universitätsadministration vorkommen, für *Gertwol* nicht analysierbar. Um diese Lücken aufzufüllen, haben wir manuell ein universitätsspezifisches Lexikon erstellt, das mit dem gleichen Format wie *Gertwol* operiert.
Zweck der Lemmatisierung ist es, dass bei der Suche Erwähnungen des gleichen Inhalts in unterschiedlichen Wortformen (‚immatrikuliert' vs. ‚immatrikulieren') entdeckt werden. In einem weiteren Schritt planen wir, nominalisierte Verben in Bezug zu deren verbalen Formen zu bringen (‚Immatrikulation' vs. ‚immatrikuliert'). Bereits realisiert, d.h. bei der Suche berücksichtigt, ist die Kompositionsauflösung, so dass der Wortteil ‚Diplom' in ‚Diplomprüfung' auffindbar ist.

3.2.3 Wortartenbestimmung

Die Verarbeitung mit *Gertwol* liefert für jede Wortform immer dieselben Analysen – meist sind es mehrere verschiedene. Bei der Wortartenbestimmung mit einem statistischen Tagger hingegen nimmt man Bezug auf den konkret vorhandenen Satz und dessen Wortfolge und liefert die dazu am wahrscheinlichsten passenden Wortarten zurück. Die Wortartenbestimmung kann deshalb bis zu einem gewissen Grad die von *Gertwol* gelie-

[6] Das Format ist von Toni Arnold, dem Hauptprogrammierer von LUIS, entwickelt und implementiert worden.

ferten Ergebnisse desambiguieren – z.B. die unwahrscheinlichere Lesart von ‚einen' als Verb von der Verwendung als unbestimmter Artikel.
Für LUIS verwenden wir den *TreeTagger*[7]. Damit ein statistischer Tagger verwendet werden kann, muss er zuerst über einem geeignet annotierten Korpus trainiert werden [VS98]. Wir haben ein für unseren Anwendungsbereich typisches Textkorpus mit gut 80'000 Wortformen zusammengestellt und die Wortarteninformation manuell eingefügt bzw. korrigiert. Bei den Wortarten haben wir uns möglichst nah an das STTS (Stuttgart-Tübingen-Tagset) gehalten, das für das Deutsche am weitesten verbreitet ist. Damit können wir nun eine automatische Wortartenbestimmung machen, die den Eigenheiten unserer Texte angepasst ist und hinreichend robust läuft.
Mit Hilfe der Wortartenbestimmung ist es möglich, abgetrennte Verbpräfixe, die oft mit Präpositionen formgleich sind, in die Verblemmas zu integrieren. So kann der oberflächliche Unterschied zwischen ‚meldet an' und ‚anmeldet' aufgehoben werden.

3.2.4 Synonymbestimmung

Damit die Ausbeute (*recall*) bei der Suche nicht zu klein ist, d.h. damit genügend relevante Passagen von LUIS zurückgeliefert werden, haben wir die Bestimmung von synonymen Lemmas eingebaut. Für den englischen Sprachraum wird in der Computerlinguistik für die Lösung diese Problems meist (vgl. etwa [VH00]) auf den elektronisch verfügbaren *WordNet*-Thesaurus[8] zurückgegriffen. Der Ansatz von WordNet wurde mittlerweile auf viele europäische Sprachen portiert (*EuroWordNet,* siehe [Vo98]), im Rahmen des *GermaNet*-Projekts [HF97] ist an der Universität Tübingen auch ein Thesaurus für die deutsche Sprache entstanden .
Leider sind die allgemeinen Synonymbeziehungen von Nomen, wie sie im *GermaNet* erscheinen, ungeeignet für unsere Zwecke. Viele Einträge fehlen, die in unserem Anwendungsbereich relevant sind, oder die Synonymklassen sind zu gross für unsere spezifischen Bedürfnisse. Deshalb haben wir einen anwendungsspezifischen Thesaurus (*UniNet)* erstellt mit gut 20'000 Einträgen, der Synonym- und Hyponymbeziehungen für universitätsspezifische Termini einhält; so ist z.B. die Information, dass sich die Wörter ‚Studentenausweis'. ‚Legitimationskarte' und ‚Legi' auf die gleiche Sache beziehen, im *UniNet* festgehalten. Im Aufbau, Format und der gewählten Struktur folgt das *UniNet* dabei dem Beispiel von *WordNet* bzw. *GermaNet* und kann problemlos integriert werden.
Die Verben sind weniger anwendungsgebunden und für deren Synonymiebeziehungen greifen wir auf die Daten von *GermaNet* zurück.

[7] Detaillierte Informationen über den TreeTagger sind erhältlich unter:
http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger-de.html
[8] Weitere Informationen über *WordNet* sind erhältlich unter: http://www.cogsci.princeton.edu/~wn.

3.2.5 Stichwortfilterung

In der Wissensbasis werden letztlich neben den Textsegmentierungsinformationen (Sätze, Abschnitte, Kapitel, Überschriften) nur Informationen im Zusammenhang mit inhalttragenden Lemmas (Nomen, Verben und Adjektive) eingetragen, sowie die spezifischen Texteinheiten.
Um den Benutzenden wenig aussagekräftige Antwortstellen zu ersparen, werden bestimmte Stichworte in einer Stoppwortliste aufgesammelt, so dass deren Einträge später beim Suchprozess gezielt ignoriert werden können. Wir gehen davon aus, dass die Häufigkeit eines Wortes umgekehrt proportional zu seinem Informationsgehalt steht. Falls ein Wort in den von uns durchsuchten Texten häufiger vorkommt als ein bestimmter Grenzwert, findet es automatisch Eingang in die Stoppwortliste. Da einzelne Wörter (‚immatrikulieren') zwar häufig vorkommen, aber in unserem Anwendungsbereich trotzdem stark inhaltstragend sind, werden diese manuell in eine Antistoppwortliste eingetragen und nie herausgefiltert.

3.3 Der Suchprozess

Jede Anfrage wird der gleichen linguistischen Verarbeitung unterzogen wie die Informationstexte. D.h. sie wird auf ihre in der Verarbeitung gewonnenen inhaltstragenden Stichwörter und spezifischen Texteinheiten reduziert, welche unter dem Begriff *Suchset* zusammengefasst sind. Beim Suchprozess geht es darum, möglichst kleine Textstellen zu finden, die zu diesem Suchset passen.

3.3.1 Erstellung der Suchdatenbasen

Um die Effizienz zu erhöhen, wird eine zweistufige Suchstrategie verwendet. In einer ersten Stufe wird die ganze Wissensbasis auf die Elemente im Suchset durchsucht und die gefundenen Stellen werden in kleineren Subdatenbasen abgelegt. Ziel dieses Vorgangs ist es, nur diejenigen Datensätze der Wissensbasis in einem zweiten Schritt weiter zu bearbeiten, in denen die gesuchten Begriffe auch vorkommen. Es werden parallel vier Subdatenbasen aus den jeweils relevanten Textstellen erstellt. Der *ersten* Subdatenbasis liegt ein Suchset zugrunde, das alle in der Anfrage vorkommenden Wörter (ohne Synonyme und Kompositazerlegung) beinhaltet. Die *zweite* Subdatenbasis enthält Textstellen mit nominalen Synonymen aus dem *UniNet*-Thesaurus. Ein Beispiel wären die synonymen Nomen ‚Studentenausweis' und ‚Legitimationskarte'. Die *dritte* Subdatenbasis wird erzeugt auf Grund der verbalen Synonyme aus *GermaNet*. Die *vierte* Subdatenbasis berücksichtigt die Bestandteile von nominalen Komposita in der Frage.

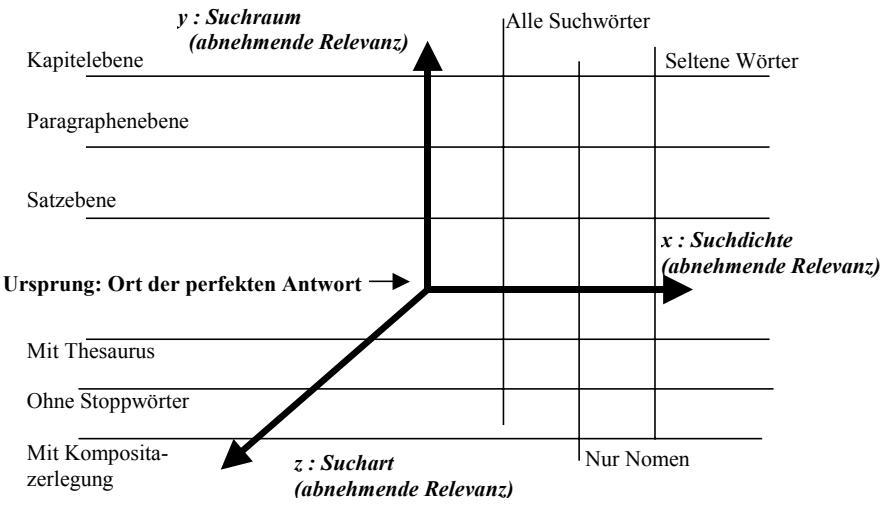

Abbildung 2: Suchuniversum von LUIS

3.3.2 Antwort-Lokalisation

Der LUIS-Suchstrategie liegen zwei Prinzipien zugrunde. Das Verfahren kann am übersichtlichsten in einem dreidimensionalen Koordinatensystem dargestellt werden (siehe Abbildung 2).

- *Abnehmende Relevanz*: Je weiter sich die Suche vom Ursprung des Koordinatensystems entfernt – das die „perfekte Lösung" symbolisiert – desto weniger aussagekräftig sind die gefundenen Resultate. Dieses Prinzip gilt für alle drei Vektoren.
- *Best-Bet-Strategie*: Sobald die Anforderungen einer Suchkombination *vollständig* erfüllt sind, d.h. mindestens eine Textpassage gefunden wurde, in der alle Suchausdrücke vorhanden sind, wird die Suche beendet.

3.3.3 Merkmalsdefinitionen des LUIS-Suchuniversums

Die in Abbildung 2 dargestellten Vektoren sind folgendermaßen definiert:

- Vektor *X* definiert das Merkmal der *Suchdichte* innerhalb der erstellten Subdatenbasen. LUIS kennt die Möglichkeit, beim Scheitern einer ersten Suche die Inhalte der Suchsets zu variieren. Vorgegeben sind jeweils drei Szenarien. Zuerst wird versucht, mit allen Suchwörtern ein Resultat innerhalb der Subdatenbasen zu erzielen; scheitert dies, dann werden neue Suchsets erzeugt, die nur noch die Nomen der gefilterten Suchwörter berücksichtigen; scheitert dies erneut, werden aufgrund der eruierten Häufigkeit der gefilterten Suchwörter nur noch die seltensten Suchwörter im Suchset belassen, und nur noch mit diesen wird ein weiterer Versuch unternommen.
- Vektor *Y* steht für das Merkmal des *Suchraums*. Werden alle Ausdrücke eines Suchsets in einem Satz gefunden, dann wird das Suchresultat als relevanter eingestuft als wenn die gesuchten Ausdrücke in einem Paragraphen oder sogar nur in einem Kapitel verstreut gefunden werden.
- Vektor *Z* steht für das Merkmal der *Sucharten* und nimmt dabei implizit eine Gewichtung der einzelnen Sucharten vor. Die Ergebnisse einer Suche mit einem Suchset, das die ursprünglich in der Ausgangsfrage vorgefundenen und später als Suchwörter gefilterten Ausdrücke enthält, werden als relevanter eingestuft als eine Suche, die zusätzlich synonyme Ausdrücke verwendet. Dahinter rangieren noch die Sucharten, bei denen die Entfernung der Stoppwörter einerseits und die Suche aufgrund von Komposita andererseits versucht wird.

Aufgrund der genannten Merkmale und Prinzipien lässt sich die Suchstrategie zusammenstellen. Es können verschiedene Kombinationen aus den drei Merkmalsbereichen für die Definitionen von Suchsets hintereinander implementiert werden, wobei zwingend aus den ersten beiden Merkmalsbereichen (*X*, *Y*) je ein Merkmal gewählt werden muss. Das Einbeziehen der dritten Dimension (*Z*) ist optional. Je weniger Raum im Koordinatensystem belegt wird, desto aussagekräftiger fallen die Ergebnisse aus. Die Reihenfolge der gewählten Definitionen für die Suchsets – sprich die Suchstrategie – lässt sich über einer Liste ohne Programmierkenntnisse steuern und anpassen.

3.3.4 Sortierung und Gewichtung der gefundenen Passagen

LUIS kennt drei Sortierkriterien. Zuerst wird anhand der Anzahl Treffer innerhalb eines gegebenen Quelldokuments eine erste Rangliste erstellt.[9] In einem zweiten Schritt werden Textstellen, die aus den FAQ- und Glossar-Datenbasen stammen, bei gleicher Anzahl Treffer nach vorne gerückt. Dieser Gewichtung liegt die Idee zu Grunde, dass im Vergleich zu anderen verfügbaren Quellen in den erwähnten Dokumentsammlungen die konziseren Informationen enthalten sind. Drittens werden Treffer, die in Kapitelüberschriften enthalten sind, nach vorne gerückt.

[9] Dabei werden Treffer im ganzen gefundenen Quelldokument gezählt, d.h. hier werden die Parameter Satz, Paragraph und Kapitel nicht unterschieden.

3.4 WWW-Anbindung

Die Kernmodule von LUIS – die linguistische Verarbeitung und der Suchprozess – wurden in der logischen Programmiersprache Prolog[10] implementiert. Der Weg vom Abschicken der Anfrage aus der WWW-Schnittstelle bis zum Kernmodul sowie die Anzeige der Resultate übernehmen Perl-Skripte (*cgi*-Schnittstelle). Die gefundenen Textpassagen werden nicht isoliert angezeigt, sondern im Dokument angesprungen und farblich markiert. Dies bedingt, dass wir für alle Informationstexte lokale Kopien unterhalten. Damit diese Kopien mit den sich ständig verändernden Daten auf dem Web synchronisiert sind, wird jede Nacht automatisch eine aktuelle Wissensbasis erzeugt.

4 Zusammenfassung und Ausblick

Die Antworten von LUIS sind im Vergleich zu den herkömmlichen, nur mit nackten Stichworten operierenden Suchmaschinen differenzierter und qualitativ besser. Die erhöhte Präzision ist einerseits in den ausgesuchten Wissensquellen begründet und andererseits in der linguistischen Verarbeitungstiefe (Reduktion auf Grundformen und Synonymgewinnung durch den UniNet-Thesaurus).
LUIS ist kein Dokumentenretrievalsystem wie herkömmliche Suchmaschinen, sondern ein Antwort-Extraktionssystem, das innerhalb eines Dokumentes die relevanten Stellen markiert und somit eine schnelle Orientierung des Benutzers erlaubt.
Jedoch stößt man auch mit LUIS an die Grenzen, die einem im Kern stichwortbasiert operierenden Systems gesetzt sind.[11] Trotz des Einsatzes spezifischer Thesauri sind die Resultate von den *gewählten Wörtern* in der Frage abhängig und weniger vom *erfragten Inhalt*. Dies bedeutet, dass die Benutzenden bessere Resultate erhalten, wenn ihre Anfragen nahe an der verwendeten Sprache in den Informationstexten ist.
Für eine Weiterentwicklung wünschenswert wäre die Integration von Modulen, die auf semantisch-logischen Problemlösungen basieren und sowohl den Kontext in der Wissensbasis als auch den Inhalt eines Fragesatzes analysieren und entsprechend verarbeiten könnten, wie das im Projekt *ExtrAns* realisiert ist [Mo00].
Eine Implementierung auf Grund der Erfassung von Fragentypologien und die Erarbeitung von logischen Formeln hierzu wären interessante Ansätze, die aufwändig und anspruchvoll zu verwirklichen sind. Gerade für Systeme aber, die einen expliziten Themenbereich haben wie LUIS, wäre ein solcher Ansatz nützlich.
Bisher jedoch hat sich der kombinierte Einsatz von unseren Systemen mit zunehmendem Grad an Flexibilität bewährt. Ein Benutzer sucht in einem ersten Schritt Informationen

[10] Informationen über das von uns verwendete Prolog sind erhältlich unter: http://www.sics.se/sicstus.html.
[11] Vergleiche mit herkömmlichen Suchmaschinen im WWW sind nur bedingt aussagekräftig, weil die Größenordnungen ganz anders sind. Suchmaschinen, wie z.B. http://www.search.ch, die auf Suchanfragen Tausende von Treffern orten, sind für gezielte Suchen nur beschränkt brauchbar. Vergleiche hierzu auch die offizielle Suchmaschine der Universität Zürich: http://www.search.unizh.ch.

im festen Format der FAQ und des Glossars *Von A bis Z*. Wird er nicht fündig, so kann er seine Frage an LUIS stellen. Wenn LUIS keine Antwortstelle lokalisieren kann, gibt es die bedeutungstragenden Wörter der Anfrage an die allgemeine Suchmaschine der Universität Zürich weiter. Obwohl das Fernziel natürlich darin besteht, möglichst viele Anfragen von LUIS in korrekter Weise beantworten zu lassen, erlaubt die aktuelle Hybrid-Lösung bereits eine große Entlastung der Universitätsadministration zu vergleichsweise geringen Kosten. Abfragen sind von jedem ans Internet angeschlossenen Computer möglich, zusätzlich können sie auch an *Info-Säulen* getätigt werden, die an ausgewählten Standorten innerhalb der Universität Zürich aufgestellt sind.
Die Weiterentwicklung von LUIS soll nun vor allem diejenigen linguistischen Verfahren einbeziehen, die derzeit in der Computerlinguistik-Gruppe entwickelt werden. Dazu gehört die automatische Erkennung von Phrasen (Nominal- und Präpositionalphrasen) und Teilsätzen (*clauses*). Durch diese verbesserte Strukturierung der Texte erwarten wir eine weitere Erhöhung der Präzision.
Darüber hinaus sind wir an Praxiserfahrungen interessiert und bieten LUIS zur Portierung an andere Hochschulen an.

Literaturverzeichnis

[Bu97] Burke, R. D. et. al.: Question answering from frequently-asked question files: Experiences with the FAQ Finder system. The University of Chicago, Computer Science Department, Technical Report TR-97-05, 1997.

[HF97] Hamp, B.; Feldweg, H.: GermaNet - a lexical-semantic net for German. In (Vossen, P. et. al. Hrsg.): Proc. of the ACL/EACL-97 Workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications, 1997; S. 9–15.

[Mo00] Mollá Aliod, D. et. al.: ExtrAns, an Answer Extraction System. In: Traitement Automatique de Langues (T.A.L.), Special issue on Information Retrieval oriented Natural Language Processing, 2000.

[VS98] Volk, M.; Schneider, G.: Comparing a statistical and a rule-based tagger for German. In: Proc. of KONVENS-98, Bonn, 1998; S. 125–137.

[VH00] Voorhees, E. M., Harman, D. K. (Hrsg.): Proceedings of the 8th Text Retrieval Conference, NIST, 2000.

[Vo98] Vossen, P. (Hrsg.): EuroWordNet: A Multilingual Database with Lexical Semantic Networks. Kluwer Academic Publishers, 1998.