Recherchebericht

1 Begriffe

1.1 Informatische Begriffe

Benutzeroberfläche[1],[2]

Eine Benutzeroberfläche oder auch grafische Benutzerschnittstelle oder GUI (graphical user interface) soll es Benutzern ermöglichen, auch ohne besondere Vorkentnisse mit einer Anwendungssoftware zu arbeiten. Das wird z.B. durch den Einsatz von grafischen Symbolen, Steuerelementen und Steuergeräten (z.B. Maus, Tastatur) erreicht.

Datenbank[3],[4]

Eine Datenbank ist ein elektronisches System zur effizienten, widerspruchsfreien, dauerhaften Speicherung und Verwaltung von großen Datenmengen, welche die Daten sowohl den Benutzern als auch Programmen auf Anfrage hin zur Verfügung stellt. Ein Datenbanksystem besteht aus Datenbank-Management-System (DBMS) und der eigentlichen Datenbank (Datenmenge). Datenbanken für Tripel heißen Triplestores, diese können entweder von Grund auf implementiert sein oder auf existierenden relationalen Datenbankmodellen aufbauend. Bei letzteren ist es nötig SPARQL Anfragen auf SQL Anfragen abzubilden.

Ontologie[5]

Eine Ontologie ist in der Informatik eine sprachlich gefasste und formal geordnete Darstellung einer Menge von Begrifflichkeiten und deren Beziehungen in einem bestimmten Gegenstandsbereich. Sie werden dazu genutzt, „Wissen“ in digitalisierter und formaler Form zwischen Anwendungsprogrammen und Diensten auszutauschen. Ontologien enthalten Regeln zu Schlussfolgerungen und zur Gewährleistung ihrer Gültigkeit. Wissen umfasst dabei sowohl Allgemeinwissen als auch Wissen über sehr spezielle Themengebiete und Vorgänge. Ontologien stehen eng mit der Idee des semantischen Webs im Zusammenhang. Im Vergleich zu einer Taxonomie, die nur eine hierarchische Untergliederung bildet, stellt eine Ontologie ein Netzwerk von Informationen mit logischen Relationen dar.

OWL[6],[7]

Die Web Ontology Language legt ihren Fokus auf die maschinengestütze Prozessierung von Informationen und nicht auf die Präsentation für Menschen. Mit OWL können die Bedeutung von Begriffen eines Vokabulars und die Beziehungen zwischen ihnen explizit beschrieben werden. OWL erweitert das RDF-/ RDFS-Vokabular und fügt eine formale Semantik hinzu. Auf mit OWL repräsentiertem Wissen kann maschinengestütztes logisches Schließen durchgeführt werden, um die Konsistenz dieses Wissens zu verifizieren oder implizites Wissen explizit zu machen.

Provenienz[8]

Bezeichnet Informationen über Entitäten, Aktivitäten und Personen, die an der Erzeugung von bestimmten Daten oder einem Ding beteiligt sind. Diese Informationen können benutzt werden um z.B. die Qualität oder Vertrauenswürdigkeit von Informationen zu bewerten.

PROV-O[9]

Die PROV Ontologie definiert, wie das Modell zur Repräsentation von Provenienzdaten mit OWL dargestellt werden kann. Sie besteht aus einer Menge von Klassen, Eigenschaften und Restriktionen. PROV-O soll die Repräsentation und den Austausch von Provenienzdaten, die von verschiedenen Anwendungen und unterschiedlichen Kontexten erzeugt wurden ermöglichen.

RDF[10]

Der Resource Description Framework ist ein Konzept zur Repräsentation von Informationen über Ressourcen (z.B. Dokumente, Personen, Gegenstände oder abstrakte Konzepte). Dabei werden die Informationen so dargestellt, dass sie verlustfrei zwischen Anwendungen ausgetauscht werden können. Insbesondere kann RDF dazu verwendet werden Daten im Web zu veröffentlichen und so zu verknüpfen, dass eine Person oder eine Anwendung durch das Verfolgen dieser Verknüpfungen Informationen aggregieren kann (Linked Data). Aussagen über Ressourcen sind Tripel der Form: Subjekt-Prädikat-Objekt. Dabei sind Subjekt und Objekt zueinander in Beziehung stehende Ressourcen. Das Prädikat beschreibt die Art der Beziehung. Eine Menge solcher Tripel kann als ein gerichteter bezeichneter Multigraph betrachtet werden.

RDFS[11]

RDF Schema ist eine semantische Erweiterung von RDF. Es erlaubt die Modellierung von Gruppen von Objekten und Beziehungen zwischen diesen Ressourcen. RDFS ähnelt objektorientierten Programmiersprachen wie JAVA, jedoch werden Klassen nicht bezüglich der Eigenschaften, die ihre Instanzen haben modelliert (z.B. Klasse Buch mit Eigenschaft Autor, deren Werte Personen sind), sondern es beschreibt Eigenschaften als Klassen von Ressourcen zugehörig (Eigenschaft Autor mit Domain Dokument und Range Person).

RDF Serialization[12],[13]

Sequenzielle Darstellung der RDF Graphen (strukturierte Daten). Dafür stehen eine Reihe von Formaten zur Verfügung, z.B. Turtle, JSON-LD, RDF/XML.

SPARQL[14],[15]

SPARQL ist eine Sammlung von Spezifikationen, die Sprachen und Protokolle zur Abfrage und Manipulation von RDF Graphen definieren. Mit SPARQL können Anfragen über verschiedene Datenquelle hinweg formuliert werden. SPARQL kann Graphmuster ihre Konjunktionen und Disjunktionen abfragen und unterstütz darüber hinaus Aggregation, Subqueries, Negationen, usw.

Synchronisation[16]

Im Allgemeinen bezeichnet Synchronisation das zeitliche Abstimmen von Vorgängen und sorgt dafür, dass Vorgänge gleichzeitig oder in einer festgelegten Reihenfolge ablaufen. Speziell in der Informatik versteht

man unter Synchronisation Verfahren, welche den gemeinsamen Zugriff von Prozessen auf geteilte Ressourcen regeln. Ferner stellt sie eine effiziente Zuteilung der Ressourcen sicher und das verarbeitete Daten in konsistenten Zuständen bleiben. In Datenbanksystemen geschieht die Synchronisation in der Regel durch ein Sperrverfahren, dem Locking.

Semantic Web[17]

Stellt eine Erweiterung des World Wide Webs dar. Ein Webdokument wird mit zusätzlichen Informationen verknüpft. So wird Maschinen ermöglicht vorher unstrukturierte Informationen in einen Kontext zu bringen. Beispielsweise kann der Begriff "Bremen" um die Information erweitert werden, ob ein Schiffs-, Familien- oder der Stadtname gemeint ist. Dies geschieht, indem Objekte von Interesse identifiziert werden und anschließend mit einer eindeutigen Adresse als Knoten angelegt werden. Jeder Knoten wiederum kann durch eindeutige Kanten dann mit anderen Knoten verbunden werden. So wird ein "Giant Global Graph" beschrieben. Zur Realisierung dienen Standards zur Veröffentlichung und Nutzung maschinenlesberer Daten, insbesondere RDF.

Weboberfläche[18]

Mittels einer Webschnittstelle kann eine Verbindung mit einem System über HTTP hergestellt werden. Somit kann mittels Webbrowser entweder eine GUI bedient werden oder es können mittels Webservice, Daten oder Anwendungen zwischen Systemen getauscht werden. Vorteil der Webschnittstelle ist die Unabhängigkeit von Plattformen und Programmiersprachen.

1.2 Linguistische Fachbegriffe

Affix[19]

An einen Stamm angefügtes Morphem (z.B. *re-* und *-ation* in *reconsideration*). Dem Stamm vorangestellte Affixe heißen **Präfix** (z.B. *re-*), dem Stamm angehängte Affixe heißen **Suffix** (z.B. *-ation*). Des Weiteren unterscheidet man noch Infixe und Zirkumfixe. **Infixe** treten im Wort auf. **Zirkumfixe** umschließen das Wort, d.h. sie sind mehrteilig und werden gleichzeitig an Anfang und Ende des Wortes angehangen.

Derivation[19]

Erzeugung von Lexemen aus anderen Lexemen (z.B. *select -> selector, selection; hot* und *dog -> hot dog*). Bei der Derivation wird im Allgemeinen die lexikalische Bedeutung oder die lexikalische Kategorie einesWortes geändert. Ihre Anwendung hängt nicht vom syntaktischen Kontext ab.

Inflektion[19]

Bildung grammatischer Formen aus einem Lexem (z.B. Vergangenheit, Gegenwart, Zukunft, Singular, Plural, Maskulinum, Femininum, Neutrum, usw.). Bei der Inflektion wird die lexikalische Bedeutung und die lexikalische Kategorie des Wortes nicht geändert. Im Allgemeinen hängt ihre Anwendung vom syntaktischen Kontext ab.

Lexem[19]

Ein Lexem ist ein Wort mit einem bestimmtem Klang und einer bestimmten Bedeutung (z.B. *dog* im Sinne von *Hund* und *dog* im Sinne von *jmd. an den Fersen kleben* sind zwei verschiedene Lexeme).

Morphem[19]

Die kleinste linguistische Einheit mit einer (konstanten) grammatischen Funktion (einschließlich Bedeutung). Bei einem Morphem kann es sich um ein Wort (z.B. *hand*) oder einen bedeutungstragenden Teil eines Wortes (z.B. *-ed* wie in *looked*) handeln. Diese sind dabei nicht weiter in kleinere bedeutungstragende Teile zerlegbar.

Stamm[19]

Basiseinheit, an die andere morphologische Strukturen angehangen werden (z.B. *consider* in *reconsideration*). Stämme können einfach oder komplex, d.h. zusammengesetzt sein. Ein einfacher Stamm wird auch als **Wurzel** bezeichnet (z.B. *disagree* ist Stamm von *disagreement*, aber die Wurzel ist *agree*).

2 Konzepte

Im Folgenden wird ein Überblick über von uns recherchierte Konzepte gegeben. Im Prozess des Recherchierens ist es jedoch unvermeidbar, dass man sich auch über Technologien informiert, die man später wieder verwirft, wenn sie nicht den Anforderungen genügen oder wenn man sich zwischen mehreren Alternativen entscheidet. Aus diesem Grunde sollte das vorliegende Kapitel nicht als Entscheidung für einen bestimmten Technologiestack betrachtet werden. Diese ist an vielen Stellen noch nicht endgültig gefallen, weshalb wir uns dazu entschieden haben, hier zunächst mehrere Möglichkeiten aufzuführen.

Apache Jena[20]

Apache Jena ist ein Java Framework für Semantic Web und Linked Data Anwendungen. Es umfasst:

- die RDF API mit der RDF Graphen eingelesen, erzeugt, bearbeitet und z.B. mit Turtle serialisiert werden können.
- die ARQ Engine, die u.a. SPARQL unterstützt.
- TDB ein Triplestore zum Persistieren der Daten.
- Fuseki ein SPARQL-Server, der als über HTTP erreichbarer SPARQL-Endpoint genutzt werden kann.
- die Ontology API für RDFS- und OWL-Unterstützung.
- die Inference API zur Implementierung eines reasoner.

Linked Data Reactor[21]

LD-R ist ein Java-Script Framework zur entwicklung flexibler und wiederverwendbarer User Interface Komponenten für Linked Data Anwendungen. Es baut auf Facebooks ReactJS Komponenten, der Flux Architektur, Yahoo!s Fluxible Framework und dem Semanic-UI Framework auf. Ziel ist es Benutzeroberfläche um die Fähigkeit zur Darstellung und Editierung von Linked Data zu erweitern.

MMoOn[22]

Die Multilingual Morpheme Ontology ist ein RDF basiertes Modell zur Beschreibung der morphologischen Struktur natürlicher Sprachen. Es umfasst die Beschreibung von Elementen auf Wort-Ebene (z.B. Lexeme, Wortformen) sowie der Wortteil-Ebene (z.B. Wurzeln, Stämme, Affixe), Derivationen, Inflexionen und der Bedeutung aller linguistischen Elemente.
Das Modell besteht aus 2 Ebenen:

- eine **sprachunabhängige** Schema-Ebene (MMoOn Core), die linguistische Kategorien der Morphologie abbildet,
- eine **sprachspezifische** Schema-Ebene, in der die morphologischen Besonderheiten einer bestimmten Sprache repräsentiert werden.

Die sprachspezifische Schema-Ebene ist eine Instanz des MMoOn Core. Auf dem Modell lässt sich eine dritte Ebene aufbauen, die **MMoOn-Language-Inventories**, welche die sprachspezifischen Daten (Lexeme, Wortformen, Morphe, Morpheme) umfasst. Die Daten der MMoOn-Language-Inventories sind Instanzen der Klassen sprachspezifischen Schema-Ebene.

Onto-Wiki

Onto Wiki ist eine „semantic application“ bzw. ein Framework für die Nutzung von Applikationen im Semantik Web Kontext. Die Hauptfunktion liegt darin, Maschinen-Lesbare Daten in Form von RDF/XML, Notation3, Turtle und Talis(JSON) zu verwalten. Hierzu besitzt OntoWiki ein Userinterface.

Sesame[23],[24]

Sesame ist ein Java Framework zum Lesen, Speichern, Abfragen von und Schließen auf RDF-Daten. Es unterstützt die Verbindung zu SPARQL Endpoints und kann auch mit verschiedenen Drittanbieter Triplestores (z.B. Vituoso) verbunden werden. Sesame unterstützt alle bedeutenden Serialisierungsformate (u.a. Turtle). Das Framework besteht aus folgenden Komponenten:

- RDF Model definiert Interfaces und Implementierungen aller grundlegenden RDF Kategorien (URI, blank node, literal, statement)
- Rio (RDF I/O) stellt Parser zum Lesen von RDF Daten zu verfügung und Implementiert die Serialisierung
- Sail API (Storage and Inference) abstrahiert von Speicher- und Schluß-Details
- HTTP-Client verarbeitet die Kommunikation, die in HTTP-Server implementiert ist
- Repository API ist bietet abstrakte Methoden zur Verarbeitung von RDF-Daten, die die Entwicklung vereinfachen sollen
- HTTP-Server implementiert ein Protokoll um auf Sesame Repositories über HTTP zugreifen zu können

Virtuoso[25],[26]

Virtuoso Universal Server (Open Source Version: OpenLink Virtuoso) ist eine in C programmierte hybride Serverarchitektur, d.h. es bietet verschiedene Serverfunktionalitäten in einem System. Zum Funktionsumfang gehören:

- Verwaltung relationaler Daten (Columnstore, SQL, etc.)
- Verwaltung relationaler Graphen (Triplesore, SPARQL, etc.)
- Content Management (HTML, Turtle, etc.)
- Webdokument Server, File Server
- RDF-basierter Linked Data Server
- Web Application Server (phpBB3 (Board), Drupal, Wordpress, MediaWiki)

Zend Framework

Zend Framework ist ein Open Source Framework für die Entwicklung von Web-Anwendungen und Services mit PHP 5. Es bietet eine MVC-Implementation, eine Datenbankabstraktion und eine Form-Komponente, die HTML-Form-Darstellung, -Prüfung und -Filterung implementiert, damit

Entwickler alle diese Operationen konsolidieren können durch Verwendung objektorientierten Interfaces.

3 Aspekte

3.1 Ist-Zustand[22]

In den letzten Jahren vollzog sich ein rapider Wandel hinsichtlich lexikalischer Ressourcen im Semantic Web. Während in vielen Fachgebieten linguistische Informationen bereits maschinenlesbar sind, ist dies im Bereich der Morphologie weitestgehend noch nicht erreicht. Morphologische Daten sind bisher entweder nicht vorhanden oder liegen verborgen in semi-strukturierten Dokumenten. Konkret bedeutet das, dass es derzeit keine geeignete Plattform gibt, die von Linguisten gemeinsam genutzt werden könnte, um morphologische Daten zu sammeln und zu kategorisieren.

3.2 Zielsetzung

Ziel dieses Projektes ist es, eine derartige Online-Plattform in Form einer Web-Applikation zu konzipieren und umzusetzen. Diese soll Sprachwissenschaftlern die Möglichkeit bieten, ihre Forschungsergebnisse festzuhalten, wiederzuverwenden und anderen zur Verfügung zu stellen. Gleichsam sollen auch sie von der Vorarbeit anderer profitieren können, indem sie deren gesammelte Informationen auswerten.
Im Speziellen muss das vollendete Projekt folgende Funktionalitäten unterstützen: Man muss neue Einträge erstellen, vorhandene Daten durch eine eingebaute Suchfunktion finden, anzeigen und bearbeiten können sowie vorhandene Datenlücken auffinden und aufgelistet bekommen. Weiterhin müssen Mechanismen entwickelt werden, die die dauerhafte Konsistenz der genutzten RDF-Datenbank gewährleisten und Redundanz vorbeugen (Duplikate erkennen/vermeiden). Um einen sinnvollen Workflow zu gewährleisten, ist dafür eine Benutzerkontensteuerung nötig. Das bedeutet, dass Linguisten sich zunächst registrieren müssen und danach Zugriff sowie die Berechtigung zur Änderung der Daten erhalten. Andernfalls könnte nicht zurückverfolgt werden, wer wann welche Änderungen vorgenommen hat. Zur Verwaltung der User-Accounts und Überwachung des Systems soll außerdem ein Administrator-Account eingerichtet werden.
Die große Herausforderung liegt darin, die Oberfläche so intuitiv wie möglich zu gestalten, denn Linguisten besitzen meist keine bzw. nur wenig fundierte informatische Kenntnisse. Die Bedienung sollte ohne Einarbeitung offensichtlich sein und keine Spielräume für Verwirrung bieten.

3.3 Gestaltungsspielräume

Unter Umständen können darüber hinaus - je nach zeitlichen Gegebenheiten - von uns folgende Funktionalitäten integriert werden:
Es wäre gut, die Definitionen des verwendeten Vokabulars (rdfs:comment) in der Web-Oberfläche direkt einsehen zu können. Dies könnte beispielsweise durch Tooltips oder ein integriertes Glossar realisiert werden.
Eine weitere nützliche Funktion wäre die Möglichkeit, in User-Accounts eigene Projekte anlegen zu können, falls jemand umfangreichere Daten hat und diese selbst wieder verwenden möchte. Dazu kämen jedoch weitere Aufgaben wie die Ansicht der selbst angelegten Dateneinträge in Listenform sowie die Möglichkeit des Downloads der Projektdaten in verschiedenen Formaten.
Darüber hinaus wäre es wünschenswert, eine übergreifende Datenset-History zu haben, die für alle sichtbar ist und in der man sieht, wer wann was im Datenset über die Web-Oberfläche geändert hat.

Auch diverse Social-Media Funktionen könnten nützlich sein, um die Möglichkeit der direkten Kommunikation der Linguisten untereinander zu ermöglichen. Somit könnte wissenschaftliche Zusammenarbeit möglich werden, ohne im realen Leben in der Nähe zu wohnen oder sich überhaupt zu kennen.

Quellen

[1] Hartmut Ernst, Jochen Schmidt, Gerd Beneken (Autoren) (2015). Grundkurs Informatik
[2] https://de.wikipedia.org/wiki/Grafische_Benutzeroberfl%C3%A4che#cite_ref-2 (abgerufen am 05.01.2016)
[3] https://de.wikipedia.org/wiki/Datenbank (aberufen am 05.012016)
[4] https://en.wikipedia.org/wiki/Triplestorehttps://en.wikipedia.org/wiki/Triplestore (abgerufen am 06.01.2016)
[5] https://de.wikipedia.org/wiki/Ontologie_%28Informatik%29 (abgerufen am 05.01.2016)
[6] Hitzler, P., Krötzsch, M., Parsia, B., Patel-Schneider, P. F., Rudolph, S. (Hrsg.) (2012). *OWL 2 Web Ontology Language Primer.* W3C. http://www.w3.org/TR/2012/REC-owl2-primer-20121211/ (abgerufen am 06.01.2016)
[7] McGuinness, D. L., van Harmelen, F. (2004). *OWL Web Ontology Language Overview.* W3C. http://www.w3.org/TR/2004/REC-owl-features-20040210/ (abgerufen am 06.01.2016)
[8] Groth, P., Moreau, L. (Hrsg.). (2013). *PROV-Overview.* W3C. http://www.w3.org/TR/2013/NOTE-prov-overview-20130430/ (abgerufen am 03.01.2016)
[9] Lebo, T., Sahoo, S., McGuinness, D. (2013) *PROV-O: The PROV Ontology.* W3C, http://www.w3.org/TR/2013/REC-prov-o-20130430/ (abgerufen am 05.10.2016)
[10] Schreiber, G., Raimond, Y. (Hrsg.) (2014). *RDF 1.1 Primer.* W3C. http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/ (abgerufen am 03.01.2016)
[11] Brickley, D., Gutha, R.V. (Hrsg.) (2014). *RDF Schema 1.1.* W3C. http://www.w3.org/TR/2014/REC-rdf-schema-20140225/ (abgerufen am 05.01.2016)
[12] Schreiber, G., Raimond, Y. (Hrsg.) (2014). *RDF 1.1 Primer.* W3C. http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/ (abgerufen am 03.01.2016)
[13] https://de.wikipedia.org/wiki/Serialisierung (abgerufen am 03.01.2016)
[14] The W3C SPARQL Working Group (2013). *SPARQL 1.1 Overview.* W3C. http://www.w3.org/TR/2013/REC-sparql11-overview-20130321/ (abgerufen am 06.01.2016)
[15] Garlik, S.H., Seaborne, A. (Hrsg.) (2013). *SPARQL 1.1 Query Language.* W3C. http://www.w3.org/TR/2013/REC-sparql11-query-20130321/ (abgerufen am 06.01.2016)
[16] https://de.wikipedia.org/wiki/Synchronisation (abgerufen am 05.01.2016)
[17] https://de.wikipedia.org/wiki/Semantic_Web (abgerufen am 05.01.2016)
[18] https://de.wikipedia.org/wiki/Webschnittstelle (abgerufen am 05.01.2016)
[19] Aronoff, M., Fudeman, K. (2011). *What is Morphology?* Chichester [u.a.], Wiley-Blackwell
[20] http://jena.apache.org (abgerufen am 05.01.2016)
[21] http://ld-r.org/ (abgerufen am 05.01.2016)
[22] http://mmoon.org/ (abgerufen am 04.01.2016)
[23] http://rdf4j.org/about.docbook?view (abgerufen am 05.01.2016)
[24] http://rdf4j.org/sesame/2.7/docs/users.docbook?view (abgerufen am 05.01.2016)
[25] http://virtuoso.openlinksw.com/ (abgerufen am 06.01.2016)
[26] http://virtuoso.openlinksw.com/dataspace/doc/dav/wiki/Main/ (abgefuen am 06.01.2016)
[27] www.zend.com (abgerufen am 07.01.2016)