Projektdokumentation im Modul Semantic Web

Familienfreundlichste Stadtteile der Stadt Leipzig

Alexander Müller

10.07.2015

Recherchefragestellung: Welcher ist der familienfreundlichste Stadtteil Leipzigs auf Grund der Anzahl der vorhandenen Schulen, Kindergärten, Spielplätze, Schwimmbäder, Sporthallen und Sportplätze.

1 Inhaltliche Interpretation der Fragestellung

Leipzig ist eine wachsende Stadt und hat großes Interesse daran, dass junge Menschen gern in Leipzig leben und will diese bei einer Familiengründung unterstützen. Die Stadt Leipzig ist dabei sehr bemüht attraktiver für Familien zu werden und hat dazu bereits 2011 einen „Aktionsplan kinder- und familienfreundliche Stadt Leipzig 2011 bis 2015“[1] beschlossen.

Ein Rangliste der Stadtteile nach ihrer Familienfreundlichkeit könnte ein wichtiges Hilfsmittel sein, um zu bestimmen, in welchen Regionen der Stadt noch Handlungsbedarf besteht. Als Kriterium soll dabei die Anzahl der in einem Stadtteil vorhandenen Einrichtungen für Kinder verwendet werden. Die Einrichtungen für Kinder sind im Rahmen dieses Projektes auf Schulen (Grundschulen, Realschulen, Gymnasien), Kindergärten, Spielplätze, Schwimmbäder, Sporthallen und Sportplätze beschränkt.

[1] siehe `www.leipzig.de/jugend-familie-und-soziales/familienfreundliche-stadt/`

2 Relevante Datenquellen

Für die Auswertung der Fragestellung werden Lageinformationen zu sämtlichen Einrichtungen in Leipzig benötigt. Zusätzlich müssen diese dann den jeweiligen Stadtteilen der Stadt Leipzig zugeordnet werden können. Im Folgenden werden alle dafür relevanten Datenquellen aufgelistet und beschrieben.

2.1 Datenbasis des Leipzig Data Projekts

Die Leipziger Initiative für Offene Daten bemüht sich schon seit einigen Jahren um die Etablierung offener Daten in der Leipziger Region. Seit dem wurden verschiedene Daten systematisch erfasst, gepflegt und unter einer freien Lizenz veröffentlicht. Die Datenbasis des Leipzig Data Projekts besteht aus RDF-Wissensbasen für verschiedene Themenbereiche, welche in einem OntoWiki[2] öffentlich inspiziert werden können. Zusammen mit verschiedenen Tools wurden die Wissensbasen zu Verwaltung auch in einem Versionsverwaltungssystem im Turtle-Format[3] veröffentlicht. Außerdem wird ein Sparql Endpunkt[4] bereitgestellt, um Abfragen auf den Daten auszuführen. Für das Projekt wurden folgende Wissenbasen verwendet:

- Adressen in der Stadt Leipzig
- Schulen der Stadt Leipzig
- Stadtbezirke und Ortsteile

Außer diesen sind zum Beispiel Wissensbasen über sämtliche Bürgervereine, Polizeidirektionen, Seniorenbüros, Personen und Geo-Koordinaten für alle Adressen in Leipzig verfügbar.

Link	`http://leipzig-data.de/Data/`
Datenformat	RDF, OWL
Schnittstelle	SPARQL, Linked Data
Lizenz	CC0 1.0 Universal
Open Data	⋆⋆⋆⋆⋆

[2] siehe `http://leipzig-data.de/Data/`
[3] siehe `https://github.com/LeipzigData`
[4] siehe `http://leipzig-data.de:8890/sparql`

2.2 Adressdatenbank auf der Webseite der Stadt Leipzig

Die Stadt Leipzig bietet auf ihrer Webseite einen Service zur Recherche unterschiedlichster Einrichtungen in Leipzig. Die Datensammlung ist sehr umfangreich und lässt sich nach Themen sowie Gebieten filtern. Vorhandene Themen sind beispielsweise Gesundheit, Kinder, Jugend und Familie, Kulturvereine, Senioren, Sport, Migration, Integration und Interkulturelles, Wissenschaft, Wirtschaft, Kultur, Behinderung und Bildung mit jeweils mehreren unter Kategorien. Über diese Adresssuche wurden folgende Iformationen für das Projekt extrahiert:

- Kindergärten
- Spielplätze
- Schwimmhallen
- Sportplätze
- Sporthallen

Zu jedem gefunden Eintrag der Adresssuche existiert eine Webseite mit einer Detailansicht des Eintrages. Auf dieser Detailseite werden genauere Informationen zu der bestimmten Einrichtung angezeigt, beispielsweise Kurzbeschreibungen, Adressinformationen, Kontaktmöglichkeiten, Ansprechpartner, Angebote, Bilder oder Öffnungszeiten. Jedoch ist der Umfang der angegeben Informationen sehr unterschiedlich.

Link	`http://www.leipzig.de/suchergebnisse-adressdatenbank/`
Datenformat	HTML
Schnittstelle	HTTP (interne API)
Lizenz	urheber- bzw. leistungsrechtlich geschützt
Open Data	⋆⋆⋆⋆

3 Extraktion relevanter Daten und Import in einen Triplestore

3.1 Extraktion Leipzig Data Datenbasis

Die Extraktion der Daten erfolgt über das OntoWiki Leipzig Data Projekt. Dabei wurden die RDF-Daten der drei Wissensbasen (Adressen, Ortsteile, Schulen der Stadt Leipzig) im Turtle Format exportiert und lokal abgespeichert.

3.2 Extraktion www.Leipzig.de

Die Extraktion der Daten von der Webseite der Stadt Leipzig war um Einiges komplexer, da diese Informationen erst aus den HTML-Seiten extrahiert werden mussten und nicht bereits als Tripel vorgelegen haben. Zur Extraktion wurde ein Java-Programm geschrieben, welches mit Hilfe der Java-Bibliothek „JSOUP“[5] die Daten aus den HTML-Seiten extrahiert. Die Adresssuche auf der Webseite verwendet eine interne API, welche über Parameter der GET-Methode des Hypertext Transfer Protocols (HTTP) die Suchkriterien übergeben bekommt. Welche Parameter übergeben werden können, konnte durch die Analyse des Seitenquelltextes herausgefunden werden. Die folgende URL zeigt einen beispielhaften Aufruf der Suche, bei dem alle Kindergärten angezeigt werden sollen.

```
http://www.leipzig.de/suchergebnisse-dressdatenbank/
  ?tx_ewerkaddressdatabase_pi[showAll]=1
  &tx_ewerkaddressdatabase_pi[topics]=105
  &tx_ewerkaddressdatabase_pi[query]=
  &tx_ewerkaddressdatabase_pi[action]=list
  &tx_ewerkaddressdatabase_pi[controller]=Address
```

Über den Parameter „`tx_ewerkaddressdatabase_pi[topics]`“ wird eine eindeutige Nummer der gesuchten Kategorie übergeben, in dem oberen Beispiel steht die 105 für die Suchkategorie Kindergärten. Die für die Fragestellung relevanten Kategorien und deren Nummern wurden ebenfalls durch die Analyse des Quellcodes der Auswahlbox herausgefunden.

Kategorie	**ID**
Kindergärten	105
Spielplätze	111
Sporthallen	375
Sportplätze	377
Schwimmbäder	433, 435

Nach Aufruf einer Suchanfrage wird eine HTML-Seite mit einer Ergebnisliste zurückgegeben. Diese Liste enthält die Namen Link zu den Detailseiten der gefundenen Einträge.

Im zweiten Schritt des Java-Programms werden die Detailseiten einzeln aufgerufen, was ebenfalls mit JSOUP realisiert wurde. Da JSOUP das kompletten Document Object Model (DOM) der übergebenen URL lädt, kann über jedes Element der HTML-Seite genau adressiert und extrahiert werden. Die extrahierten Informationen aus den Detailseiten (Name, URL, Adresse, Stadtteil) werden bei jedem gefundenen Eintrag als Attribute in ein Objekt geschrieben und anschließend das Objekt in einer Liste gespeichert.

[5] siehe `http://jsoup.org`

Da diese Extraktion sehr zeitaufwendig ist, wird die Liste mit den Objekten für die weitere Verarbeitung im JSON-Format zwischengespeichert.

3.3 Erstellen der RDF-Triple und Import in den Triplestore

Die im Kapitel 3.2 extrahierten Daten wurden anschließend mit Hilfe des Jena-Frameworks[6] in RDF-Daten überführt und gespeichert. Dazu wurde ein Ontologie erstellt, welche aus jeweils einer Klasse für jeder der fünf Kategorien (Kindergärten, Spielplätze, Sporthallen, Sportplätze, Schwimmbäder) besteht. Dabei wurde der private Namespace `http://www.imn.htwk-leipzig.de/~amuelle3/Data/` mit dem Präfix *am* verwendet.

Die fünf Klassen ähneln sich sehr in ihren Eigenschaften. Alle Klassen besitzen die Datatype-Property *rdfs:label*, welche die genaue Beschreibung des Objektes enthält. Weiterhin besitzen alle fünf Klassen die Object-Property *ld:inOrtsteil*. Diese Property stammt aus der Leipzig Data Ontologie und der Präfix *ld* steht für folgenden Namespace: `http://leipzig-data.de/Data/Model/`. Diese Property hat als Range die Klasse *ld:Ortsteil* der Leipzig Data Ontologie.

Eine weitere Object-Property, welche bloß die Klasse Spielplatz nicht besitzt ist *ld:hasAddress*, ebenfalls aus der Leipzig Data Ontologie. Wie der Name vermuten lässt, wird mit dieser Property eine genaue Adresse mit dem Objekt verknüpft. Der Range der Property ist dabei die Klasse *ld:Adresse*. Spielplätze besitzen meist keine genaue Adresse, da sie in Parks oder freien Flächen in Wohngebieten liegen, besitzen somit diese Property nicht. Listing 1 und Listing 2 zeigen als Beispiel die Instanzen eines Kindergarten und eines Spielplatzes.

Listing 1: Beispielinstanz der Klasse Kindergarten

```
<http://www.imn.htwk-leipzig.de/~amuelle3/Data/
  Kindergarten/abenteuerland-kombinierte-tageseinrichtung>
      a           am:Kindergarten ;
      rdfs:label "Abenteuerland Kombinierte Tageseinrichtung" ;
      ld:hasAddress <http://leipzig-data.de/Data/
        04277.Leipzig.Heilemannstrasse> ;
      ld:inOrtsteil <http://leipzig-data.de/Data/Ortsteil/Connewitz> .
```

Listing 2: Beispielinstanz der Klasse Spielplatz

```
<http://www.imn.htwk-leipzig.de/~amuelle3/Data/
  Spielplatz/spielplatz-lene-voigt-park-maerchenplatz>
```

[6] siehe `https://jena.apache.org`

```
    a           :Spielplatz ;
    rdfs:label "Spielplatz Lene-Voigt-Park - Maerchenplatz" ;
    ld:inOrtsteil <http://leipzig-data.de/Data/Ortsteil/Reudnitz-Thonberg>
        .
```

Die erzeugten Tripel wurden abschließend als RDF-Dateien im Turtle-Format gespeichert und in den Triplestore „Fuseki“[7] des Apache Jena-Frameworks importiert. Dieser Triplestore dient als SPARQL-Endpunkt und besitzt eine grafische Oberfläche welche über den Browser aufgerufen werden kann.

4 Verlinkung von Ressourcen

Die Verlinkung der Ressourcen erfolgt über die *ld:hasAddress* und die *ld:inOrtsteil* Property der Leipzig Daten Ontologie. In Abbildung 1 ist das verwendete RDF-Schema abgebildet.

Um die Fragestellungen zu beantworten ist es notwendig von den jeweiligen Einrichtung eine Verbindung zu dem Stadtteil herzustellen, in dem sie sich befindet. In den Klassen für Schule, Kindergarten, Spielplatz, Schwimmbad, Sporthalle und Sportplatz existiert eine direkte Verbindung über die *ld:inOrtsteil* Property. Dagegen haben die Klassen der drei Schularten keine direkte Verknüpfung zu einem Stadtteil. Diese muss daher über die Adresse und die Property *ld:hasAddress* hergestellt werden, da jede Adresse einem Ortsteil zugeordnet ist. Diese sind bei allen Klassen gleich, bis auf die der Schulen, da bei diesem der Stadtteil über die Adresse verknüpft werden muss.

5 Anfrage an die Forschungswissensbasis

5.1 SPARQL-Anfrage

In einer SPARQL-Anfrage sollen nun alle Einrichtungen in den Stadtteilen gezählt werden. Dazu wird für jeden Klasse eine Abfrage benötigt, in der alle Objekte der Klasse mit dem dazugehörigen Stadtteil abgerufen werden. Anschließend wird über die Stadtteile gruppiert (`GROUP BY`) und dann die Anzahl mit `COUNT()` gezählt. In Listing 3 und Listing 4 ist der Unterschied zu sehen, ob die Zuordnung zu einem Stadtteil direkt über die Property*ld:inOrtsteil* oder erst über die Adresse hergestellt wird.

[7] siehe `https://jena.apache.org/documentation/fuseki2`

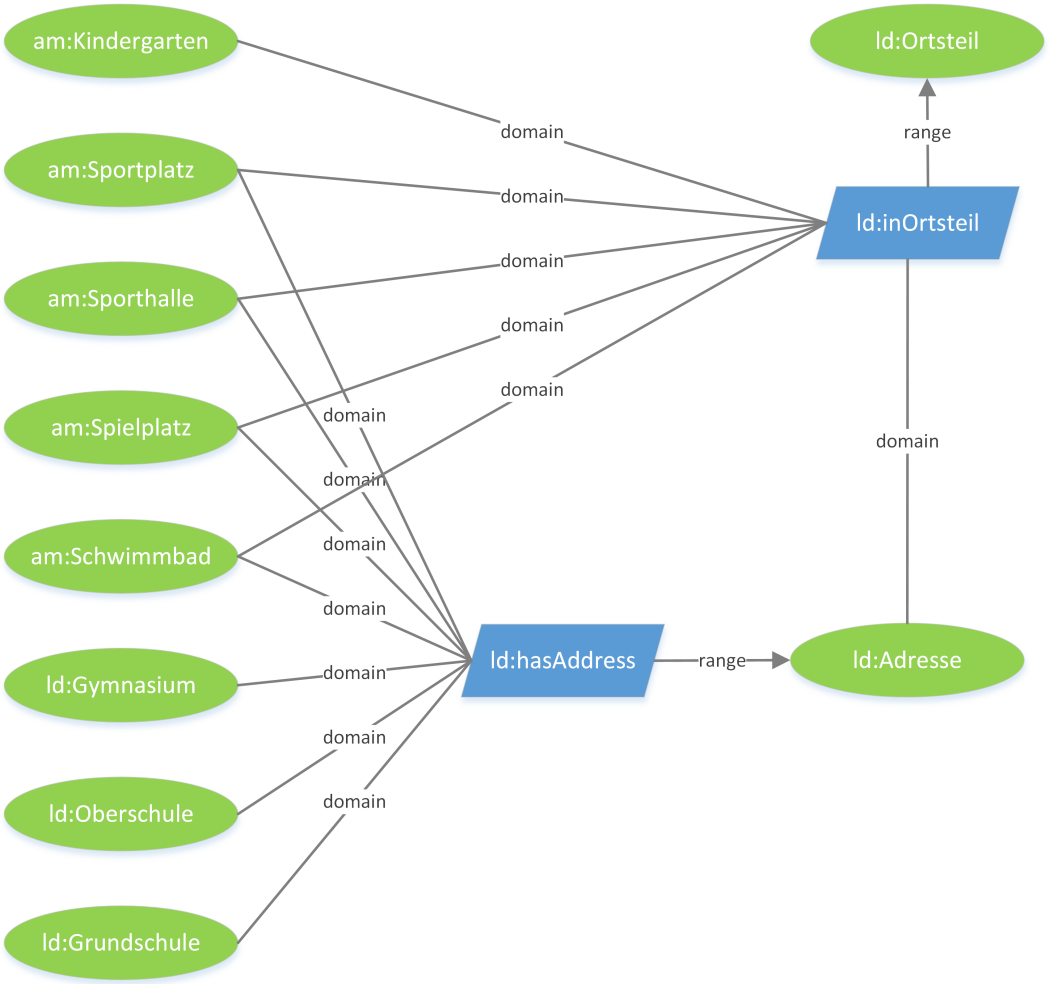

Abbildung 1: RDF-Schema

Listing 3: Abfrage direkt über ld:inOrtsteil

```
SELECT ?ort (count(?y) as ?kiga)
WHERE {
   ?y a am:Kindergarten .
   ?y ld:inOrtsteil ?ort .
}
GROUP BY ?ort
~
```

Listing 4: Abfrage über ld:hasAddress

```
SELECT ?ort (count(?g) as ?gym )
WHERE {
   ?g rdf:type ld:Gymnasium.
   ?g ld:hasAddress ?a.
   ?a ld:inOrtsteil ?ort.
}
GROUP BY ?ort
```

Um diese Zählung für alle Klassen gleichzeitig durchzuführen, werden diese einzelnen Abfragen in einer Abfrage geschachtelt. In der äußeren Abfrage werden dann zuerst alle

Ortsteile mit ihren Namen abgerufen. Da nicht jede Einrichtung in jedem Stadtteil vorhanden ist, dürfen nicht vorhandene Einträge auch nicht entfernt werden. In SQL würde das einem `LEFT OUTER JOIN` entsprechen. In SPARQL wird diese Funktionalität mit dem Schlüsselwort `OPTIONAL` eingesetzt. Der fehlende Wert bleibt somit in der entsprechenden Spalte leer und muss für eine spätere Auswertung mit 0 ersetzt werden. Durch die fehlenden Felder ist eine Aggregation der Werte in der äußeren Abfrage jedoch nicht mehr möglich. In Listing 5 ist eine gekürzte Version der finalen SPARQL-Abfrage an die Wissenbasis abgebildet.

Listing 5: SPARQL-Anfrage

```
PREFIX am: <http://www.imn.htwk-leipzig.de/~amuelle3/Data/Model/>
PREFIX ld: <http://leipzig-data.de/Data/Model/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema\#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns\#>

SELECT ?Ort ?grund ?gym ?ober ?kiga ?spielplatz ?schwimmbad ?sportplatz
    ?sporthalle
where
{
  ?ort a ld:Ortsteil .
  ?ort rdfs:label ?Ort .

  OPTIONAL {
    select ?ort (count(?g) as ?gym )
    where {
       ?g rdf:type ld:Gymnasium.
       ?g ld:hasAddress ?a.
       ?a ld:inOrtsteil ?ort.
    }
    GROUP BY ?ort
  }
  OPTIONAL {
    SELECT ?ort (count(?y) as ?kiga)
    WHERE {
       ?y a am:Kindergarten .
       ?y ld:inOrtsteil ?ort .
    }
    GROUP BY ?ort
  }
}
```

Die vollständige SPARQL-Abfrage kann in dem veröffentlichten GitHub-Repository[8] eingesehen werden.

[8] siehe `https://github.com/muexx/sematic-web`

5.2 Ergebnis der Anfrage

Das Ergebnis der Anfrage ist eine Tabelle mit allen Stadtteilen in der ersten Spalte und der jeweiligen Anzahl der vorhandenen Einrichtungen in den folgenden Spalten. Das Resultat wurde zur Auswertung als CSV-Datei exportiert und leere Spalten mit Null aufgefüllt. Anschließend wurde die Anzahl der Einrichtungen pro Stadtteil aufsummiert, um das Ergebnis auf die Fragestellung zu erhalten.

In Tabelle 1 sind die Stadtteile mit den meisten kinderfreundlichen Einrichtung zu sehen. Tabelle 2 zeigt die Stadtteile mit den wenigsten kinderfreundlichen Einrichtung.

Tabelle 1: familienfreundlichste Stadtteile

#	Stadtteil	Grund-schulen	Gym-nasien	Ober-schulen	Kinder-gärten	Spiel-plätze	Schwimm-bäder	Sport-plätze	Sport-hallen	Σ
1	Reudnitz-Thonberg	2	1	1	8	14	0	1	1	28
2	Stötteritz	1	1	0	8	13	2	1	2	28
3	Engelsdorf	1	1	0	6	15	0	1	0	24
4	Zentrum-Südost	2	1	1	12	7	1	0	0	24
5	Neustadt-Neuschönefeld	3	0	0	8	11	0	0	1	23
6	Paunsdorf	3	0	1	10	5	1	3	0	23

Tabelle 2: familienUNfreundlichste Stadtteile

#	Stadtteil	Grund-schulen	Gym-nasien	Ober-schulen	Kinder-gärten	Spiel-plätze	Schwimm-bäder	Sport-plätze	Sport-hallen	Σ
63	Grünau-Siedlung	0	0	0	0	1	0	0	0	1
62	Baalsdorf	0	0	0	1	2	0	0	0	3
61	Heiterblick	0	1	0	1	2	0	0	0	4
60	Miltitz	1	0	0	1	1	0	1	0	4
59	Meusdorf	1	0	0	1	2	0	0	1	5
58	Zentrum	0	0	0	0	2	0	0	3	5

6 Interpretation und Zusammenfassung

Wie aus den Ergebnistabellen abzulesen ist gibt es deutliche Unterschiede zwischen den Stadtteilen von Leipzig bei der Anzahl der kinderfreundlichen Einrichtungen. Dabei ist jede Kategorie der Einrichtungen gleich gewichtet, was eventuell nicht die Anforderungen der meisten Familien widerspiegelt. Aus dieser Überlegung heraus wäre es denkbar, die Ergebnisse zu personalisieren und den verschiedenen Kategorien eine andere Gewichtung zu geben. Zum Beispiel würde eine Familie mit Kindern, die bereits die Grundschule besuchen, der Kategorie der Kindergärten ein geringeres Gewicht geben, als der Kategorie der Gymnasien und Oberschulen, auf welche die Kinder in Zukunft gehen werden. Um eine solche Gewichtung zu realisieren, würde die Anzahl der entsprechenden Kategorien mit dem jeweiligen Gewicht multipliziert werden. Das würde das Gesamtergebnis jedes Stadtteils verändern und eine neue Sortierung erfordern.

Ob dies jedoch trotz dieser Anpassung als eindeutiges Maß der Kinderfreundlichkeit eines Stadtteils herangezogen werden kann, ist zu bezweifeln. Beispielsweise wurde bei der Fragestellung die Größe eines Stadtteiles komplett vernachlässigt. Auch die beschränkte Auswahl der acht verschiedenen Einrichtungen ist unzureichend, um die Fragestellungen zu beantworten. Mögliche Faktoren, die eventuell aussagekräftiger seien könnten, wäre der Altersdurchschnitt und der Anteil der bereits ansässigen Familien mit Kindern.