УДК 004.6+ 004.4

Ресурсы и инструменты для преподавания методов и средств Semantic Web

Апанович З.В. (Институт систем информатики СО РАН, Новосибирский государственный университет)

За последние годы создано значительное количество структурированных данных как научными так и коммерческими организациями. На базе этих данных разрабатываются многочисленные приложения, что делает более важным и нужным знакомство с этим направлением современных ИТ-специалистов. В данной работе обсуждаются как важные аспекты эволюции направления Semantic Web, так и опыт преподавания методов и средств Semantic Web.

Ключевые слова: *Открытые Связанные Данные, RDF, RDFS, OWL, SPARQL.*

1. Введение

С момента возникновения идеи Semantic Web в 2001 это направление стремительно развивалось. За последние годы размер Облака Связанных Данных[1] значительно возрос, и в настоящее время насчитывает порядка 10 000 наборов данных, содержащих в общей сложности более 150 миллиардов триплетов. К наиболее часто используемым наборам связанных данных относятся [1]:

- Europeana, объединяющая метаданные для цифровых объектов из музеев, архивов и аудиовизуальных архивов по всей Европе;
- Linked Data Service Библиотеки Конгресса США, использующая более 50 словарей;
- OCLC WorldCat Linked Data, каталог из более чем 370 миллионов библиографических записей, экспериментально доступных в форме Связанных данных;
- VIAF, Виртуальный международный авторитетный файл OCLC, объединяющий более 40 авторитетных файлов из разных стран и регионов. К этим наборам данных осуществляется более 100 000 запросов в день.

[1] http://linkeddata.org/

Значительное количество структурированных данных, не являющихся открытыми, создано также коммерческими организациями. Идея Semantic Web была подхвачена гигантами Интернета, такими как Google, Microsoft, Facebook, в результате чего возникли огромные графы знаний, такие как Google Knowledge graph[2] , Microsoft Satori[3], извлекающие информацию из источников данных различной природы, и позволяющие значительно улучшить качество поиска информации. В таблице 1 [2] приведена информация о размерах наиболее известных в настоящее время графах знаний. Экземпляры (instances) означают экземпляры классов (концепты А-боксов), под количеством фактов понимается количество триплетов RDF, количество типов - это количество различных типов или классов, определенных в схеме, количество отношений – количество различных отношений, определенных в схеме.

Помимо этого, публикация структурированных данных в HTML контенте коммерческих Web-сайтов стала мэйнстримом. Множество коммерческих сайтов встраивают структурированные данные в свои html-страницы при помощи таких форматов как RDFa и JSON-LD и таких словарей как schema.org и GoodRelations, рассчитывая на улучшение видимости их сайтов. В настоящее время более 540 миллионов HTML страниц имеют встроенные структурированные описания данных. На Рис. 1 показан график, отражающий рост использования классов словаря schema.org коммерческими Интернет-сайтами для публикации структурированных данных о товарах и услугах [3].

Таблица 1 [2]. Объемы общеизвестных графов знаний

Название	Кол-во экземпляров	Кол-во фактов	Кол-во типов	Кол-во отношений
DBpedia (English)	4 806 150	176 043 129	735	2 813
YAGO	4 595 906	25 946 870	488 469	77
Freebase	49 947 845	3 041 722 635	26 507	37 781
Wikidata	15 602 060	65 993 797	23 157	1,673
NELL	2 006 896	432 845	285	425

[2] http://googleblog.blogspot.co.uk/2012/05/introducing-knowledge-graph-things-not.html
[3] http://blogs.bing.com/search/2013/03/21/understand-your-world-with-bing/

OpenCyc	118 499	2 413 894	45 153	18 526
Google's Knowledge Graph	570 000 000	18 000 000 000	1 500	35 000
Google's Knowledge Vault	45 000 000	271 000 000	1 100	4 469
Yahoo! Knowledge Graph	3 443 743	1 391 054 990	250	800

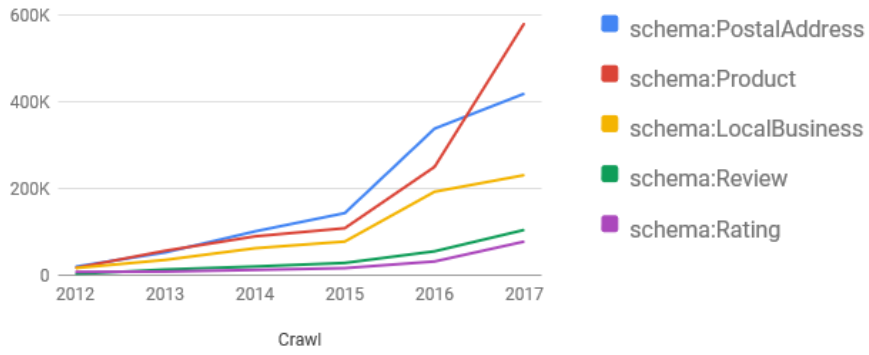

Рис. 1. Рост использования классов словаря schema.org коммерческими Интернет-сайтами для публикации структурированных данных о товарах и услугах [3]

Еще одним немаловажным аспектом является то, что направление Semantic Web предлагает действенные подходы к решению проблемы разнородности данных, являющейся одним из современных вызовов направления Big Data.

Все выше перечисленные факты делают более важным и нужным знакомство с этим направлением современных ИТ-специалистов. В Новосибирском университете на ММФ и Факультете Информационных Технологий читается спецкурс «Принципы, методы и средства

связывания данных в приложениях Semantic Web»[4], предназначенный для магистрантов, специализирующихся в области разработки программного обеспечения. В данной работе обсуждаются как важные аспекты эволюции направления Semantic Web, так и методы преподавания дисциплины «Принципы, методы и средства связывания данных в приложениях Semantic Web».

Работа выполнена при финансовой поддержке РАН (проект № 15/10).

2. Знакомство с принципами связанных данных на примере существующих наборов данных

Идея о том, что Интернет – это не только связанные документы, но и связанные данные, была высказана Тимом Бернесом Ли еще в 1994 году[5]. Затем в 2001 году появился термин «Semantic Web» [4], который обозначал расширение Всемирной паутины, позволяющее людям использовать общий контент, преодолевая границы отдельных приложений и веб-сайтов. Первоначальная идея состояла в том, что люди публикуют данные в Интернете, онтологии используются для того, чтобы все используемые термины понимались одинаково, а затем люди создают интеллектуальные приложения на основе доступных данных.

Реальный количественный скачок в развитии этого направления начался в 2006 году, когда были сформулированы принципы связанных данных[6], и началась деятельность, направленная на создание Облака Связанных данных. Поэтому знакомство с топологией Облака связанных данных является первым шагом при знакомстве с Semantic Web. Последняя визуализация этого набора, сделанная в феврале 2017 года, находится по адресу http://lod-cloud.net/. В облако входят наборы данных, разбитые по следующим тематикам: кросс-доменные, такие как DBPedia и Wikidata, географические (Geonames), библиотечные (Europeana, Worldcat, VIAF), наборы, посвященные наукам о жизни (Bio2RDF, Uniprot), лингвистические (BabelNet), средства массовой информации (BBC, New York Times), социальные сети. Хотя правительственные данные и упоминаются в этом облаке, основная информация о них находится на отдельных порталах. С момента своего создания в 2007 году, Облако Связанных данных значительно расширилось, и в настоящее время, его элементы рассредоточены по нескольким каталогам, таким как Datahub.io, publicdata.eu, data.gov, open.canada.ca. Все эти наборы данных используют CKAN платформу с открытым исходным кодом, которая является по факту стандартом для Открытых данных. Имеется

[4] http://fit.nsu.ru/data_/docs/mag/program/2013-2015/SWeb.pdf.
[5] http://www.w3.org/Talks/WWW94Tim/
[6] http://www.w3.org/DesignIssues/LinkedData.html

специальный набор данных LODStats [5], который поддерживает статистику о текущем состоянии облака данных.

Все наборы облака LOD созданы в соответствии с базовыми принципами Linked Data:

использовать URIs для определения сущностей;

- использовать HTTP URIs таким образом, чтобы на эти сущности можно было ссылаться и чтобы они могли быть найденными человеком и программным клиентом;
- при разыменовании URI, предоставлять полезную информацию о соответствующей сущности, используя такие стандарты, как RDF и SPARQL;
- при публиковании данных в веб включать в описание ссылки на другие наборы данных.

Наборы данных, удовлетворяющие всем принципам связанных данных, соответствуют пятизвездочной модели связанных данных[7].

В контексте изучения принципов Связанных Данных очень важным является осознание того, что URI (IRI) являются, прежде всего, инструментом *именования* (создания уникальных глобальных идентификаторов) как для информационных объектов, таких как HTML-страницы, так и для объектов реального мира. Эти объекты реального мира могут иметь описания как в человеко-читаемом формате, таком как страницы HTML, так и в формате, понятном компьютеру (различные синтаксические формы RDF), и что выбор подходящего описания объекта реального мира осуществляется при помощи такой процедуры, как *обсуждение контента*. Все эти понятия легко продемонстрировать на примере произвольного набора данных, хранящегося в облаке связанных данных. Например, объект реального мира Москва имеет URI <dbpedia.org/resource/Moscow> в наборе данных Dbpedia.org, html-страница этого ресурса имеет URI <http://dbpedia.org/page/Russia>, а URI <http://dbpedia.org/data/Moscow> соответствует файлу в формате RDF/XML. Получение наиболее релевантного ресурса осуществляется при помощи механизма обсуждения контента. Поэтому одно из первых практических заданий предлагаемых студентам, состоит в самостоятельном ознакомлении с произвольным набором, входящим в Облако Связанных данных, и составлении краткого отчета, отвечающего на такие вопросы как: По каким правилам формируются URI данного набора данных? Какая онтология используется для структурирования этого набора данных? С какими наборами данных связан данный набор?

[7] http://5stardata.info

Какие способы доступа имеются к данному набору данных? В соответствии с какими лицензиями он используется? И т.д.

3. Знакомство с моделью данных RDF

Модель данных RDF является стандартом, разработанным W3C, и предназначена для интегрированного представления информации, которая происходит из нескольких источников и гетерогенно структурирована (представлена с использованием различных схем).

В этой модели граф RDF является множеством триплетов вида <subject, predicate, object>, где субъект и предикат всегда являются URI, а объект может быть как URI, так и литералом (строкой). При этом предикат указывает на отношение, существующее между субъектом и объектом. Благодаря тому, что URI является глобальным уникальным идентификатором, отдельные триплеты с одинаковыми субъектами могут «склеиваться» образуя ориентированный граф с помеченными ребрами. Условно, предикаты, используемые для описания связанных данных можно разбить на три группы:

Предикаты отношений используются для связывания сущностей из разных наборов данных. Именно благодаря этим предикатам осуществляется связь между наборами, разнесенными по разным адресам. Например, следующий триплет представляет факт, состоящий в том, что Михаил Васильевич Ломоносов является автором полного собрания сочинений:

<http://www.worldcat.org/oclc/2899645> *schema:creator* <http://viaf.org/viaf/46896023>.

При этом URI субъекта этого триплета принадлежит пространству имен набора данных Worldcat, URI объекта принадлежит пространству имен набора данных VIAF, а предикат schema: creator описан в словаре schema.org.

Предикаты идентичности на уровне экземпляра указывают на URI, используемые другими источниками данных для идентификации одних и тех же объектов реального мира или абстрактных понятий (*owl:sameAs* и *rdfs:seeAlso*). Свойство *owl:sameAs* используется для выражения того, что два URI соответствуют одному и тому же объекту реального мира. Например, следующий триплет утверждает, что субъект и объект данного триплета соответствуют одному и тому же объекту реального мира:

<http://www.geonames.org/524894> *owl:sameAs* <http://dbpedia.org/resource/Moscow>. Свойство *rdfs:seeAlso* показывает, что более актуальную информацию про данный субъект можно найти, пройдя по URI объекта триплета. Автоматическое создание триплетов этого типа называется «установлением идентичности сущностей». Linked Data опирается на

решение проблемы идентичности сущностей в эволюционной и распределенной манере. Эволюционность состоит в том, что со временем создается все больше связей *owl:sameAs*, а распределенность связана с тем, что этим могут заниматься параллельно много поставщиков и потребителей данных.

Словарные предикаты связывают данные с описаниями словарных терминов, которые используются для представления данных, а также эти определения с определениями связанных терминов в других словарях. Словарные связи делают данные само-описываемыми и позволяют приложениям Linked Data понимать и интегрировать данные, описанные при помощи разных словарей. Например, предикат *rdf:type* позволяет описать принадлежность ресурса классу выбранного словаря (онтологии). Стало быть, триплет <http://dbpedia.org/resource/Moscow> *rdf:type dbo:Place* связывает сущность Moscow из набора данных dbpedia.org с классом *dbo:Place*, описанным в онтологии http://dbpedia.org/ontology. Имеется также большая группа предикатов, предназначенная для описания связей между классами и свойствами разных словарей (онтологий). Свойства из словаря RDFS *rdfs:subPropertyOf* и *rdfs:subClassOf* могут быть использованы, для того чтобы декларировать отношения между двумя свойствами или двумя классами из разных словарей. Словарь OWL предоставляет предикаты *owl:equivalentClass* и *owl:equivalentProperty* для утверждений, что два класса или два свойства, имеют один и тот же смысл.

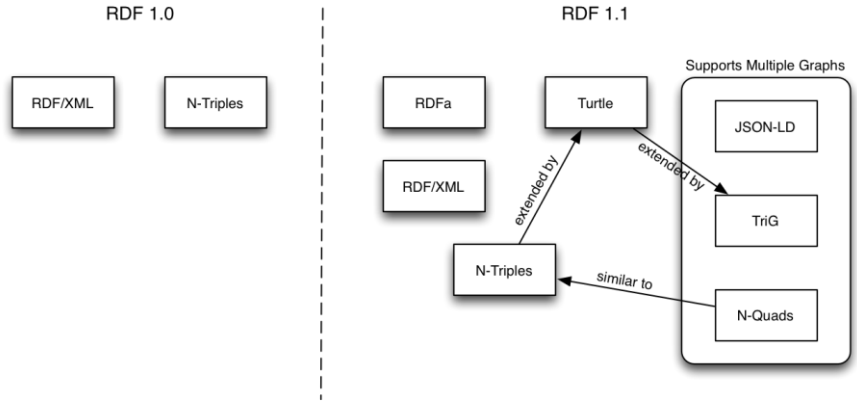

Рис. 2. Увеличение множества синтаксических форм для модели данных RDF1.1 по сравнению с моделью RDF1.0 [6]

Чтобы опубликовать граф RDF в Интернете, он сначала должен быть сериализован при помощи одной из синтаксических форм RDF. Следует заметить, что развитие проекта по созданию и практическому использованию облака Связанных данных оказало большое влияние на разработку новых версий всех основных форматов, используемых в Semantic

Web. В 2012 году появился OWL2, в 2013 году появился SPARQL 1.1, а в 2014 году – RDF 1.1. На Рис 2. показаны синтаксические формы RDF 1.0 и RDF 1.1. Одним из существенных отличий RDF 1.1 по сравнению с RDF 1.0 является появление понятия *RDF Dataset*, определяемого как множество RDF-графов. Помимо таких форм как RDF/XML, Turtle и N-Triples, появились три синтаксические формы, ориентированные на описание нескольких именованных графов Trig, N-Quads и JSON-LD. Также форматы RDFa и JSON-LD позволяют встраивать структурированные данные в HTML- страницы. Поскольку вновь появившиеся форматы активно используются в реальных приложениях Semantic Web, в курсе демонстрируются все указанные варианты синтаксических форм [7], а также осуществляется знакомство с инструментами, позволяющими конвертацию данных между этими представлениями. Помимо этого, рассматриваются такие понятия модели RDF, как пустые узлы, типизированные литералы, абсолютный и относительный URI, идентификатор фрагмента, понятие базы и т.д. Все основные примеры рассматриваются параллельно в форматах RDF/XML и Turtle.

Задание, позволяющее быстро познакомиться с простейшими этапами создания Связанных данных, выглядит следующим образом. Студентам предлагается ознакомиться с основными классами и свойствами словаря FOAF (Friend Of A Friend)[8], а затем создать описание собственной персоны в формате RDF/XML при помощи приложения FOAF-a-matic[9]. После этого они должны вручную добавить к сгенерированному файлу в формате RDF/XML пять синтаксически правильных триплетов, использующих в качестве предикатов различные свойства, описанные в разных наборах данных облака LOD. Студенты должны проверить синтаксическую правильность полученного файла при помощи валидатора RDF[10].

После знакомства с основными понятиями RDF проводится вводная лекция по SPARQL, знакомящая с его основными конструкциями. Эта вводная часть воспринимается студентами достаточно легко, поскольку SPARQL имеет много схожих черт с SQL. На последующих занятиях теоретические вопросы Связанных данных рассматриваются параллельно с углубленным изучением SPARQL 1.1. и использованием запросов SPARQL для ознакомления с понятиями Связанных данных. В частности, знакомство с форматами RDFS и OWL сопровождается знакомством с классами и свойствами конкретных словарей, описанными в облаке связанных данных. Их исследование базируется на запросах SPARQL 1.1.

---

[8] http://xmlns.com/foaf/spec/
[9] http://www.ldodds.com/foaf/foaf-a-matic.html
[10] http://www.w3.org/RDF/Validator/

4. Языки описания словарей RDFS и OWL и словари, используемые в облаке связанных данных

RDF Schema (RDFS) и язык описания онтологий Web Ontology Language (OWL) являются наиболее известными языками описания словарей (онтологий) в области Semantic Web. RDFS – это минимальный язык описания онтологии, построенный поверх RDF, который позволяет определить:

- классы индивидуальных ресурсов;
- свойства, связывающие два ресурса;
- иерархии классов;
- иерархии свойств;
- ограничения на область определения и область значений свойств (*rdfs:domain*, *rdfs:range*).

OWL обеспечивает разработчиков онтологий гораздо более сложными и полезными конструкциями, чем RDFS, благодаря чему популярность OWL постоянно растет. Как и в RDFS, основными элементами OWL являются классы, свойства и индивиды, которые являются членами классов. Свойства OWL являются бинарными отношениями, среди них принято выделять *owl:ObjectProperty* и *owl:DataTypeProperty*. Свойства типа *owl:ObjectProperty* связывают двух индивидов, тогда как *owl:DataTypeProperty* связывают индивида с литералом.

В курсе по Semantic Web делается акцент на том, что стандарты RDFS и OWL, связаны между собой, что основной синтаксической формой OWL2 является RDF/XML, и, стало быть, к схемам, описанным в форматах rdfs и owl, можно писать запросы SPARQL. Например, в том, что *owl:Class* является подклассом *rdfs:Class*, а *owl:DatatypeProperty* и *owl:ObjectProperty* являются подклассами *rdf:Property* легко убедиться, при помощи простого запроса SPARQL:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:< http://www.w3.org/2000/01/rdf-schema#>
SELECT ?s ?o
FROM <http://www.w3.org/2002/07/owl>
WHERE  {?s rdfs:subClassOf ?o }
```

Одним из важнейших свойств Semantic Web является возможность интегрировать данные из различных источников, описанных при помощи разных словарей (схем, онтологий). Словарь состоит из классов, свойств и типов данных, которые определяют смысл данных. RDF словари сами выражаются и публикуются в соответствии с принципами

связанных данных. В настоящее время в Интернете опубликовано значительное количество словарей и онтологий. Например, набор данных LODStats дает информацию о 2593 словарях, имеющихся в Облаке Связанных данных[11]. Web of Data применяет двойственный подход к такой разнородности данных. Во-первых, рекомендуется уменьшать уровень разнородности за счет использования терминов из широко используемых словарей. Во-вторых, в случае использования проприетарных терминов, они должны быть как можно более *само-описываемыми*, то есть каждый словарный термин должен быть связан со своим описанием. Кроме этого, рекомендуется публиковать файлы соответствий между терминами из разных словарей в виде RDF.

Таким образом, словари являются важной частью облака связанных данных, и знакомство с наиболее популярными словарями необходимо в рамках курса по Semantic Web [8]. Для того чтобы иметь возможность повторно использовать имеющиеся словари, надо знать, где их найти. Большое количество словарей может быть обнаружено при помощи поискового движка Watson[12]. Более 628 различных словарей (данные указаны на январь 2018 года) представлены в специализированном наборе данных Linked Open Vocabularies (LOV) [9].

Набор данных LOV строит экосистему словарей, поддерживающую их повторное использование. Он подсчитывает популярность каждого термина словаря, а также устанавливает отношения между словарями, используя словарь VOAF. На уровне словарей LOV извлекает три типа информации для каждой версии словаря: метаданные, входные связи/входные словари, выходные связи/выходные словари, связанные с каждым словарем. Метаинформация, определенная внутри словаря, предоставляет контекст и полезные данные о словаре. Для этого обычно повторно используются такие словари, как DublinCore[13], описывающий создателей, публикаторов, и других персон, внесших вклад в создание и развитие словаря, CreativeCommons[14], описывающий лицензии словаря и др. Входные связи позволяют в явном виде указать, что термины данного словаря ссылаются на термины других словарей. Аналогичным образом выходные связи позволяют в явном виде указать, что термины других словарей ссылаются на термины данного словаря.

В словаре VOAF введены такие типы отношений между словарями как *voaf:metadata*, *voaf:specializes*, *voaf:extends*, *voaf:hasEquivalencesWith*, *voaf:generalizes*,

[11] http://stats.lod2.eu/vocabularies
[12] http://watson.kmi.open.ac.uk/WatsonWUI/
[13] http://dublincore.org/documents/dc-rdf/
[14] https://creativecommons.org/ns

voaf:hasDisjunctionsWith. Также используется отношение импорта между словарями *owl:import.* На Рис. 3 показано изображение входных и выходных связей словаря schema.org с другими словарями LOV. Такое изображение генерируется автоматически для каждого словаря, включенного в LOV, при помощи запросов SPARQL.

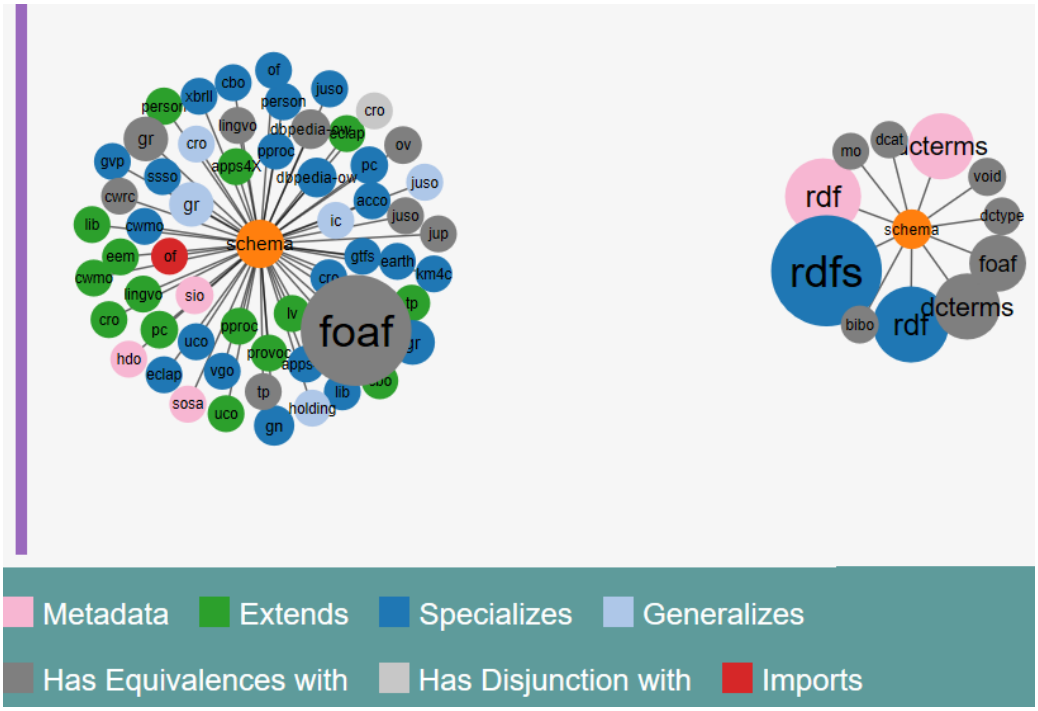

Рис. 3. Связи словаря schema.org с другими словарями LOV [9].

Для того чтобы существующий словарь был включен в набор данных LOV, должны выполняться следующие требования:

— словарь должен быть написан на RDF (RDFS, OWL/RDF) и быть разыменовываемым;
— словарь должен быть синтаксически правильным;
— все словарные термины (классы, свойства и типы данных) словаря должны иметь свойство *rdfs: label;*
— словарь должен ссылаться и повторно использовать релевантные существующие словари;
— словарь должен предоставлять некоторые метаданные о самом словаре (по крайней мере, название).

Для того чтобы поддержать повторное использование словарей, LOV поддерживает поиск словарей на основе терминов (класс, свойство, тип данных) или предметной области. Он ранжирует все термины входящих в него словарей на основе количества использований каждого термина в наборах данных облака LOD. Так самыми популярными терминами являются *rdf:type* (рейтинг1.000), *rdfs:label* (рейтинг1.000), *owl:sameAs* (рейтинг 0.5401).

Помимо того, что LOV является весьма полезным вспомогательным средством при поиске онтологий, он является также прекрасным инструментом для использования в образовательных целях. В частности, благодаря имеющейся конечной точке SPARQL, можно знакомить учащихся со многими вопросами устройства онтологий, как это будет показано в разделе 6.

Большим достоинством LOV является то, что он хранит локальные копии онтологий, с которыми можно знакомиться разными способами. Так, например, страничка LOV, посвященная словарю schema.org, указывает локальное URI этого словаря (schema.org). Но для того, чтобы получить описание этого словаря в формате RDFS, необходимо извлечь его из RDFa описания HTML страницы[15] при помощи инструмента RDFa Distiller and Parser[16]. Все словари, знакомство с которыми осуществляется в данном курсе (VOID, Dublin Core, FOAF, goodRelations, schema.org, dbpedia.org/ontology, owl, rdfs), присутствуют в наборе данных LOV.

5. Запросы SPARQL – важный инструмент работы с наборами данных

Запросы SPARQL [10] позволяют не только получить исчерпывающую информацию о любом незнакомом наборе данных, но также копировать данные, создавать новые данные, конвертировать данные, осуществлять контроль качества данных, не только проверяя выполнение таких ограничений как корректность типов данных, но и соответствие бизнес правилам.

При знакомстве с новым набором данных, прежде всего, надо понять, что это за данные. Самый первый запрос SPARQL, который следует задать к незнакомому набору данных это

```
SELECT * WHERE {  ?s ?p ?o . }#LIMIT 50
```

Поскольку это запрос к графу по умолчанию, а многие хранилища триплетов хранят данные в именованных графах, вторым уместным запросом будет запрос, который покажет URI всех именованных графов, входящих в данный набор:

```
SELECT DISTINCT ?g WHERE {  GRAPH ?g {?s ?p ?o } }
```

В частности, при запуске этого запроса на конечной точке SPARQL словаря LOV выдается список всех словарей (онтологий) имеющихся в настоящий момент в этом наборе данных, а также URI всех словарей, по которым можно строить запросы SPARQL к любому

[15] https://schema.org/docs/schema_org_rdfa_html

[16] http://www.w3.org/2007/08/pyRdfa

из описанных словарей. На примере этого набора данных достаточно просто проиллюстрировать запросами SPARQL занятия, посвященные именованным графам, языкам описания RDFS и OWL. Например, запуск следующего запроса на конечной точке http://lov.okfn.org/dataset/lov/sparql позволяет студентам быстро почувствовать разницу в доступе к именованным графам и графам по умолчанию:

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT DISTINCT ?g ?s  ?o
FROM  NAMED <http://www.w3.org/ns/mls>
FROM  <http://purl.org/vocab/aiiso/schema>
WHERE {{?s rdfs:label ?o.}
UNION
{GRAPH ?g {?s rdf:type ?o}}
}
```

Также, в курсе рассматриваются такие типы запросов, как: «Какие классы декларированы в данном наборе данных? Сколько экземпляров имеется у каждого класса заданного набора данных? Какие классы используют определенное свойство? Какие свойства декларированы? Какие значения имеет данное свойство? Какие классы используются?». В контексте RDF вопрос про то, какие классы используются, следует понимать как синоним вопроса « у какого класса имеются экземпляры»?

Следует отметить, что как классы, так и свойства в словарях часто декларируются неявно, как подклассы или подсвойства других классов. Поэтому для полного ответа на вопрос обо всех декларированных классах необходимо проверить также наличие триплетов, имеющих в качестве предиката *rdfs:subClassOf*, а также поискать транзитивное замыкание по этому предикату (rdfs:subClassOf+).

Таким образом, все статистические данные, которые обычно сопровождают набор данных в облаке связанных данных, можно вычислить самостоятельно, грамотно пользуясь запросами SPARQL. Ответы на все выше перечисленные базовые вопросы студенты учатся находить для произвольного набора данных, который они видят первый раз в жизни.

Помимо вопросов, направленных на исследование незнакомого набора данных, имеется большая группа запросов, предназначенных для работы с наборами данных на протяжении всего их жизненного цикла. Поэтому в режиме практических занятий рассматриваются следующие вопросы:

— как при помощи запроса CONSTRUCT объединить данные из нескольких разрозненных наборов данных,
— как сконструировать новые данные на основе имеющихся,
— как преобразовать данные, описанные при помощи одной онтологии в данные, основанные на другой онтологии,
— как при помощи запросов SPARQL можно автоматизировать решение задачи идентификации сущностей в разных наборах данных (генерацию отношений *owl:sameAs*),
— как устанавливать соответствие между классами и свойствами разных онтологий.

Опыт показал, что благодаря знакомству с языком SQL, студенты легко справляются с формой запроса SELECT, при этом у них возникают проблемы с применением таких форм, как ASK и CONSTRUCT. Поэтому данные формы запросов требуют большего количества упражнений. Еще один вопрос, требующий внимательного рассмотрения, связан с построением запросов к отдельным файлам при помощи ключевого слова FROM или же доступа к удаленной конечной точке при помощи ключевого слова SERVICE. Наконец, в режиме практических занятий следует явно показывать, что запросы SPARQL будут формулироваться по-разному, в зависимости от типа конечной точки SPARQL (специализированной, такой как dbedia.org/sparql, или же точки произвольного доступа, такой как http://uriburner.com/sparql).

6. Жизненный цикл связанных данных и инструменты, используемые на разных этапах жизненного цикла.

Процесс предоставления связанных данных часто характеризуется как *жизненный цикл*, в котором данные создаются, повторно используя другие источники данных, преобразуются, а затем делаются доступными для использования. Вновь созданные данные могут стать одним из ряда источников данных, используемых при подготовке дальнейших данных. Таким образом, создание и публикация связанных данных имеет циклический характер. Известно несколько разных моделей жизненного цикла связанных данных, наиболее подробная модель жизненного цикла [11] включает такие этапы как извлечение данных, хранение и поддержка запросов, авторская разработка, связывание данных, семантическое обогащение, анализ качества, эволюция и исправление ошибок, поиск, просмотр и разведка. В упрощенной формулировке жизненного цикла связанных данных можно выделить три основные стадии. Эти стадии также могут быть сопоставлены с принципами связанных данных.

Создание связанных данных: Извлечение данных, создание HTTP URI в качестве имен, и выборе словарей для описания предикатов и для классификации сущностей. Этот этап относится к принципам связанных данных 1 и 2, поскольку предполагает нахождение или создание HTTP URI имен для вещей.

Связывание данных: Поиск и выражение связей между сущностями-синонимами из разных наборов данных. Этот этап относится к принципу связанных данных 4, так как предполагает обеспечение ссылок на другие сущности.

Публикация связанных данных: Создание метаданных о наборе данных и обеспечение доступа к набору данных. Относится к принципу связанных данных 3 в том, что обеспечивает возвращение полезной информации о наборе данных.

Инструменты, применяемые на каждой стадии жизненного цикла, быстро эволюционируют. Каждый год появляются новые инструменты, а некоторые инструменты устаревают и перестают использоваться. Поэтому эта часть курса нуждается в ежегодном обновлении. Тем не менее, есть некоторые базовые инструменты, знакомство с которыми важно для понимания курса.

Наиболее распространенные форматы, на основе которых осуществляется генерация данных RDF – это таблицы или табличные данные, реляционные базы данных и текстовые данные. Для разных видов данных существуют различные стратегии и инструменты, позволяющие преобразовать их в RDF. Реляционные БД обычно отображаются в RDF при помощи правил установления соответствия и инструментов, таких как D2R [12]. В этих случаях правила отображения пишутся вручную, что весьма несложно, так как схема реляционной БД обычно явно задана. В рамках данного курса, студенты знакомятся с языком R2RML, который является рекомендацией W3C для определения отображения между реляционными базами данных и связанными данными. Он позволяет определить шаблоны, по которым названия таблиц отображаются в названия классов выбранной онтологии, первичные ключи таблицы отображаются в глобальные URI в выбранном пользователем пространстве имен, а названия столбцов таблиц отображаются в предикаты выбранной пользователем онтологии. Полу-структурированные данные, такие как веб – таблицы, обычно существуют в больших количествах, но без явной семантики. Их автоматическое преобразование в RDF является не простой задачей [13]. В рамках данного курса, студенты знакомятся с простым инструментом OpenRefine[17], который позволяет решать эту задачу в интерактивном режиме. Ряд инструментов существует для поддержки извлечения данных из

[17] http://openrefine.org/

свободного текста, включая Open Calais[18] и DBpedia Spotlight [14]. На этапе связывания данных решается задача установления связей идентичности между отдельными индивидами, а также установления словарных связей между свойствами и классами, описанных в разных онтологиях словарях. Для автоматизированного решения первой задачи часто используется инструмент SILK [15], с которым осуществляется знакомство в данном курсе. Дополнительно, в курсе изучается вопрос, как триплеты с предикатом *owl:sameAs* могут быть созданы в результате логического вывода. Генерация триплетов с предикатом *owl:sameAs* при помощи запросов SPARQL рассматривается на практических занятиях. Что касается задачи создания словарных связей, то автоматическое решение этой задачи остается активной областью исследования, знакомиться с которой более уместно в форме научных семинаров[19]. В рамках данного курса демонстрируется, как можно решать эту задачу при помощи запросов SPARQL.

После того, как наборы данных RDF был созданы и взаимосвязаны, процесс публикации включает следующие задачи:

1. Создание метаданных для описания набора данных;
2. Предоставление доступа к набору данных;
3. Валидация набора данных.

Метаданные создаются в соответствии с такими словарями как VOID и Dublin Core. Оба эти словаря представлены в наборе данных LOV. В качестве основной формы предоставления связанных рассматриваются каталог DataHub, основанный на платформе с открытым исходным кодом CKAN, а также хранилища триплетов, такие как Jena[20] и Virtuoso[21].

В качестве инструментов валидации данных можно использовать RDF Triple-Checker[22], который помогает находить опечатки и распространенные ошибки в данных RDF, OntOlogy Pitfall Scanner![23] , который помогает обнаружить некоторые наиболее распространенные ошибки, возникающие при разработке онтологий, а также инструмент RDFAlerts[24].

7. Приложения Связанных данных

[18] http://www.opencalais.com/opencalais-demo/
[19] http://oaei.ontologymatching.org/
[20] jena.apache.org
[21] https://virtuoso.openlinksw.com/
[22] http://graphite.ecs.soton.ac.uk/checker/
[23]http://oops.linkeddata.es/ OOPS!
[24] http://swse.deri.org/RDFAlerts/

Приложения связанных данных являются тем, что подтверждает ценность этого направления. Среди существующих приложений можно выделить следующие три группы.

1) *Навигаторы Связанных данных.* Разыменовывают URI, чтобы получить описание ресурса. Типичным примером навигатора Связанных данных являются DBPedia.org.

2) *Поисковые системы связанных данных.* Позволяют посылать запросы к связанным данным. В отличие от обычных поисковых систем, которые в основном рассматривается как средство для локализации человеко-читаемого контента, семантическая поисковая система используется для поиска онтологий, словарей и документов RDF. К таким системам относятся Watson и LOV. Системы Семантического поиска, встроенные в поисковые системы Google и Bing, опираются на внутренние Графы Знаний и позволяют помимо поиска по ключевым словам выдавать дополнительную информацию о сущностях, отображенных на эти Графы Знаний. Поисковые системы, основанные на Графах Знаний, все чаще используются в промышленности. Появляется все больше коммерческих приложений, использующих интеграцию данных о продуктах, услугах, предложениях работы и т.д. [16, 17].

3) *Приложения Связанных данных*, ориентированные на конкретную предметную область. Эти приложения создаются для решения конкретного круга проблем в указанном домене. Подавляющее большинство приложений Связанных данных попадают в эту третью категорию.

Примером приложения семантических технологий являются многочисленные разделы портала bbc.com. Использование семантических веб-технологий не видно пользователю, который взаимодействует с порталом как с обычным веб-сайтом. Для управления контентом этого сайта реализована специальная архитектура, которая называется «Динамическая Семантическая Публикация» и направлена на автоматизацию агрегации или публикацию взаимосвязанного контента в рамках портала BBC[25].

Еще одно интересное приложение, которое рассматривается в курсе - Open Pharmacology Space[26] - семантическая исследовательская среда для фармакологии. Исследование и разработка новых лекарств требует от ученых извлечения знаний из множественных источников информации от баз данных белков и химических соединений до моделей биологических путей. Интеграция данных в таких системах является серьезной проблемой. Например, источник ChemSpider содержит информацию о химических соединениях, где они

[25] http://www.bbc.co.uk/blogs/bbcinternet/2012/04/sports_dynamic_semantic.html

[26] http://www.openphacts.org/open-phacts-discovery-platform

были получены, CheMBL дополняет эту информацию данными о биоактивности молекул, похожих на лекарства, Drugbank дает информацию о клиническом использовании лекарств, содержащих указанную молекулу. Поскольку эти источники по-разному подходят к представлению структуры молекулы, они могут выдавать разные соединения для одного и того же химического препарата. Open Pharmacology Space [17] идентифицирует сущности, описанные в разных источниках данных, и увязывает их между собой.

Еще один интересный проект GRAVITATE[27] направлен на создание инструментов, позволяющих археологам реконструировать разрушенные культурные объекты, части которых хранятся в разных коллекциях. Помимо примеров конкретных приложений, в курсе рассматривается также архитектура приложений, работающих со Связанными данными.

8. Заключение

В данной работе кратко представлены основные части курса, который все еще находится в процессе развития, поскольку в процессе развития находится научное направление Semantic Web. С одной стороны, данный аспект повышает актуальность представляемого курса, с другой стороны, создает трудности в его преподавании, поскольку каждый год курс приходится обновлять. Наблюдается процесс интеграции знаний из Облака Открытых Связанных данных с данными, встроенными в html- страницы, методы машинного обучения используются для улучшения качества данных Semantic Web, и одновременно методы Semantic Web используются для поддержки открытия новых знаний [18]. Это значит, что потребность в специалистах, способных работать в этом направлении, будет и дальше возрастать.

Список литературы

1. Smith-Yoshimura K. Analysis of International Linked Data Survey for Implementers// D-Lib Magazine. July/August 2016. Vol. 22, №7/8.
2. Paulheim H., Automatic Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods// Semantic Web. 2016.
3. Bizer Ch., Meusel R., Primpeli A. Web Data Commons - RDFa, Microdata, and Microformat Data Sets, 2018 http://webdatacommons.org/structureddata/
4. Berners-Lee T., Hendler J., Lassila O. The Semantic Web: Scientific American: Feature Article May 2001

[27] http://gravitate-project.eu/

5. Ermilov I., Lehmann J., Martin M., Auer S. LODStats: The Data Web Census Dataset //ISWC 2016, pp. 38-46.
6. Wood D., 3 Round Stones Inc. What's New in RDF 1.1 URL:http://www.w3.org/TR/rdf11-new/ (дата обращения 05.01.2018)
7. Schreiber G., Raimond Y., RDF 1.1 Primer, W3C Working Group Note 24 June 2014 URL: http://www.w3.org/TR/rdf11-primer/ (дата обращения 05.01.2018)
8. d'Aquin M., Noy N. F. Where to publish and find ontologies? a survey of ontology libraries// Web Semantics:Science, Services and Agents on the World Wide Web, 2012. P. 96 – 111.
9. Vandenbusschea P-Y, Atemezingb G. A., Poveda-Villalónc M., Vatantd B., Linked Open Vocabularies (LOV): a gateway to reusable semantic vocabularies on the Web// Semantic Web, 2014. P. 1–5.
10. DuCharme B. Learning SPARQL, Second Edition, O'Reilly Media, Inc. URL http://it-ebooks.info/book/2574/ (дата обращения 25.12.2017).
11. Auer S., Lehmann J., A-C. Ngonga Ngomo, Zaveri A.Introduction to Linked Data and its Lifecycle on the Web// Reasoning Web. Semantic Technologies for the Web of Data - 7th International Summer School 2011, Galway, Ireland, August 23-27, 2011, Tutorial Lectures http://www.jens-lehmann.org/files/2013/reasoning_web_linked_data.pdf (дата обращения 05.01.2018).
12. Spanos D.-E., Stavrou P., Mitrou N., Bringing relational databases into the semantic web: A survey, Semant. Web. 2012. Vol. 3, №2. P. 169–209. http://www.semantic-web-journal.net/sites/default/files/swj121_1.pdf (дата обращения 05.01.2018).
13. Mulwad V., Finin, T. Joshi A., Semantic message passing for generating linked data from tables, in: Proceedings of the 12th International Semantic Web Conference, Springer, 2013.
14. Mendes P.N., Jakob M., García-Silva A., Bizer C., Dbpedia spotlight: Shedding light on the web of documents// Proceedings of the 7th International Conference on Semantic Systems, I-Semantics'11, ACM, New York, NY,USA, 2011, P. 1–8.
15. Isele R., Jentzsch A., Bizer Ch. Silk Server - Adding missing Links while consuming Linked Data// 1st International Workshop on Consuming Linked Data (COLD 2010), Shanghai, November 2010.
16. Ristoski P., Mika P., Enriching Product Ads with Metadata from HTML Annotations// ESWC 2016, LNCS 9678, 2016. P. 151–167.
17. Gray A. J.G., Groth P., Loizou A., Askjaer S., Brenninkmeijer C., Burger K., Chichester C., Evelo C. T., Goble C., Harland L., Pettifer S., Thompson M., Waagmeester A., Williams A. J. Applying linked data approaches to pharmacology: Architectural decisions and implementation//Semantic Web . 2014. P. 101–113.
18. Ristoski P. Paulheim H. SemanticWeb in data mining and knowledge discovery: A comprehensive survey//Web Semantics: Science, Services and Agents on theWorldWideWeb. 2016. P.1–22.