УДК 004.82 https://doi.org/10.15407/pp2019.02.038

К.А. Кудим, Г.Ю. Проскудина

МЕТОДЫ И СРЕДСТВА ИЗВЛЕЧЕНИЯ ДАННЫХ О ПЕРСОНАЛИЯХ ИЗ АВТОРЕФЕРАТОВ ДИССЕРТАЦИЙ

В работе рассмотрены подходы к решению задачи сбора и извлечения разрозненных данных о персоналиях из слабоструктурированных и неструктурированных документов, представленных в общедоступных каталогах авторефератов диссертаций. На языке PHP с применением XPath разработана система, которая позволяет автоматически собирать первичные документы из электронного каталога Национальной библиотеки Украины им. В.И. Вернадского, извлекать из этих документов данные и сохранять их в локальном хранилище. Для хранения выбрана модель данных RDF с учётом особенностей данных и возможностью последующего представления в семантической сети.

Ключевые слова: извлечение данных, слабоструктурированные документы, технология XPath, регулярные выражения, семантическая сеть.

Введение

В настоящее время в Интернете доступен большой объём разрозненной публичной информации о персоналиях, не представленной явно, а содержащийся в документах и электронных каталогах как дополнительные сведения. Например, сведения об учёных защитивших диссертацию могут быть найдены в авторефератах диссертаций.

В собранном виде эти данные могут представлять ценность как независимый ресурс. Однако сбор таких данных затруднён как из-за количества информации для обработки, так и по причине её слабой структурированности. Учитывая данное обстоятельство предлагается автоматизировать сбор и извлечение фактов о персоналиях из общедоступных источников, в частности из авторефератов диссертаций.

В качестве ресурса для реализации прототипа был выбран электронный каталог авторефератов диссертаций Национальной библиотеки Украины им. В.И. Вернадского.

1. Постановка задачи

Некоторое время, занимаясь созданием электронной большой Украинской Энциклопедии, в частности, Википедии и семантической Вики, мы обнаружили проект Викидата [1] и начали подробно изучать этот проект.

Что это такое в целом? Известно, что Википедия представляет собой большую коллекцию статей, которые наполняются пользователями в довольно произвольной форме. В настоящее время существует параллельный проект Викидата, где реализована попытка представить информацию Википедии только исключительно в структурированном виде. Причем так, чтобы ее можно было обрабатывать как машиной, так и человеком.

Почему такая проблема возникла? Дело в том, что в Википедии есть много языковых разделов (несколько десятков), и некоторая информация дублируется в каждом языковом разделе абсолютно идентично. Например, у каждой статьи есть список ссылок на ту же статью на других языках. И этот список в каждом языковом разделе будет один и тот же. Но в каждом языковом разделе, будут другие пользователи, которые редактируют эти списки. Так в Википедии появились боты, которые сами обновляют эти списки. А в проекте Викидата все решается просто – есть одно централизованное хранилище, которые хранит общие данные, а каждая языковая Википедия может вставлять эти данные на свои страницы. Можно сказать, что Викидата это база данных, где хранятся не просто статьи, а структурированные данные. Причем эти данные настолько подробные, что из них можно генерировать статьи, подобные википедийным.

На рис. 1 приведен пример статьи, которая сгенерирована по данным [2]. Это не статья, которую набирал человек. При

© К.А. Кудим, Г.Ю. Проскудина, 2019

этом ее сделали человеко-читаемой. Здесь есть краткая биография И.С. Баха, когда родился, когда умер, основные факты из биографии. Далее перечислены родители, дети. Весь этот сгенерированный контент берется из базы данных Викидата. Здесь также есть аудио, видео и графический контент, относящийся в биографии этого композитора, который берется из некоторых других централизованных источников информации соответствующего типа.

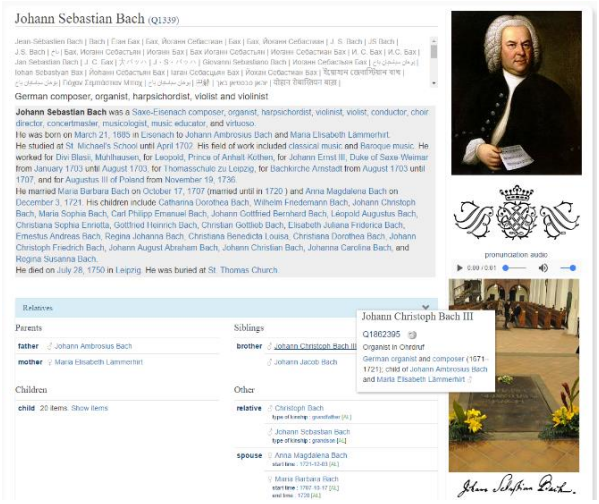


Рис. 1. Автоматически сгенерированная страница о И.С. Бахе

Викидата это – отдельный централизованный проект со своей администрацией и своими правилами. Например, на него распространяются правила википедийной *значимости*. Нельзя заносить в базу данных Викидата любые объекты подряд. Например, мы не можем заносить туда своих кандидатов наук, если это по мнению администраторов Викидата не выдающиеся ученые. Или какие-то значимые для нас статьи, которые не значимые для Википедии, также туда попасть не могут. Это первый момент, второй – это доступ. Для того, чтобы полноценно использовать базу Викидата, предлагается ее полностью скачать (а база данных Викидата предоставляет такую возможность) и установить себе. То есть централизованно базу данных Викидата использовать не эффективно.

Мы также задались вопросом, а можем ли мы найти в Интернете список всех ученых Украины, например, кандидатов и докторов наук? В настоящее время таких списков, централизованных нет, по крайней мере, мы не смогли их найти. И так, чтобы они были общедоступными. Некоторые ресурсы предоставляют списки докторов наук, а списков кандидатов наук вообще нет.

На основании вышеизложенного, по аналогии с проектом Викидата, принято решение о создании собственной централизованной базы данных Персоналии, руководствуясь следующими принципами:

- максимальное включение;
- публичные данные;
- источники информации: открытые, авторитетные и проверяемые;
- используется структурированное хранение (например, в формате RDF).

Таким образом в базу Персоналии, можно включать не только значимых персон, а *всех* персон, о которых есть публичная информация в Интернете, не нарушив ее приватности. Это могут быть ученые, кандидаты и доктора наук, это могут быть чиновники, врачи, учителя, сотрудники различных бюджетных организаций. Сюда можно добавлять всех, о ком, что-либо известно.

В Википедии считают, что, например, географический объект имеет право быть включенным в Википедию. Людей это не касается. И нередко в Википедии случаются диспуты по поводу персон, которые кому-то кажутся значимыми и их хотят включить в Википедию, а другой части сообщества эти персоны кажутся не значимыми, и они стараются удалять статьи о таких персонах. Нередко ведутся целые споры и пожизненные блокировки. Как говорят в Википедии, используется *имманентная значимость*.

Источники кого-то включить в базу данных Персоналии должны быть, как и в Википедии *авторитетными*, нельзя со своих слов кого-то включить. Обязательно должно быть подтверждение, и это должно быть доступное подтверждение. Информация должна размещаться на авторитетном ресурсе, принадлежащем организации к которой есть доверие всех участников.

Предполагается, что в базу данных Персоналии будет вноситься открытая и

достоверная информация обо всех людях Украины, но на первом этапе, ограничиться учеными Украины.

В итоге принято решение сохранить преимущество базы данных Викидата, чтобы хранилище оставалось структурированным, а не в виде хранилища документов. Поскольку к структурированному хранилищу можно осуществлять запросы.

2. Сбор данных

Далее возник вопрос наполнения. Естественно вручную не реально наполнять такую базу данных. В этом мы убедились, наполняя статьями нашу Библиотеку Периодических изданий[1]. Ручной ввод информации – весьма трудозатратное занятие. Даже когда доступны все материалы, введение в электронную библиотеку одного элемента данных занимает некоторое время (3–5 минут), умножая на то количество данных, которое надо ввести, получаем около тысячи часов. Поэтому нас будет интересовать именно *автоматическое наполнение*.

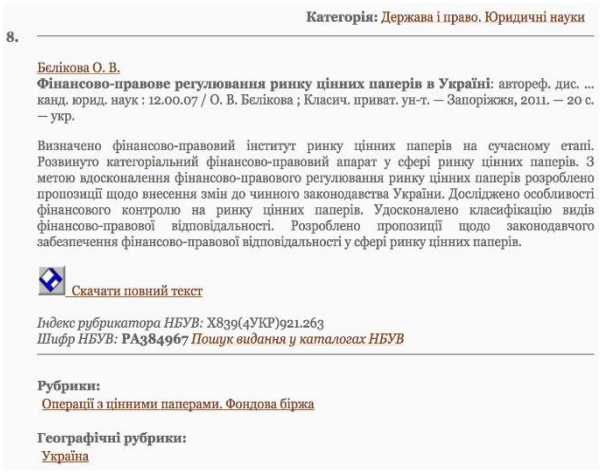
В качестве авторитетного источника (или исходных) данных был выбран сайт Национальной библиотеки Украины им. В.И. Вернадского, в части хранения авторефератов диссертаций[2]. На сегодня предоставляется открытый доступ к авторефератам всех диссертаций, защищенных в Украине за период 1998–2011 гг. Представлены они довольно просто: есть каталог и есть полные тексты авторефератов. Там имеется более 63 тыс. документов. И есть в каталоге некоторые об этих документах сведения, которые можно получить, не анализируя сами документы.

Осуществив навигацию по интерфейсу сайта Библиотеки им. В.И. Вернадского, было обнаружено, что у них есть навигация новых поступлений (рис. 2). Можно выбрать год и месяц и получить поступления не на этот месяц и год, а просто закачанные в этот месяц и год. Но для нас это неважно, главное, что мы получаем полный список авторефератов диссертаций.

Рис. 2. Навигация новых поступлений на сайте Национальной библиотеки Украины им. В.И. Вернадского

Был написан специальный скрипт, который эту навигацию делает автоматически. Начиная с 2007 года, когда они начали закачивать файлы, по 2012 год, обходит все месяцы. Таким образом были получены результаты поиска в html-формате, из которых следует уже извлекать первые данные.

На рис. 3. показано, как выглядит один элемент результата поиска. Это некоторая информация об автореферате диссертации.


Категорія: Держава і право. Юридичні науки

8.

Белікова О. В.
Фінансово-правове регулювання ринку цінних паперів в Україні: автореф. дис. ... канд. юрид. наук : 12.00.07 / О. В. Белікова ; Класич. приват. ун-т. — Запоріжжя, 2011. — 20 с. — укр.

Визначено фінансово-правовий інститут ринку цінних паперів на сучасному етапі. Розвинуто категоріальний фінансово-правовий апарат у сфері ринку цінних паперів. З метою вдосконалення фінансово-правового регулювання ринку цінних паперів розроблено пропозиції щодо внесення змін до чинного законодавства України. Досліджено особливості фінансового контролю на ринку цінних паперів. Удосконалено класифікацію видів фінансово-правової відповідальності. Розроблено пропозиції щодо законодавчого забезпечення фінансово-правової відповідальності у сфері ринку цінних паперів.

Скачати повний текст

Індекс рубрикатора НБУВ: Х839(4УКР)921.263
Шифр НБУВ: **РА384967** *Пошук видання у каталогах НБУВ*

Рубрики:
Операції з цінними паперами. Фондова біржа

Географічні рубрики:
Україна

Рис. 3. Библиографическая карточка автореферата диссертации

Для сбора первичных данных из электронного каталога используется поисковый робот (веб-краулер), который обходит страницы по навигационным ссылкам и сохраняет доступную информацию в локальном хранилище в необработанном виде. Поскольку поисковый робот не является универсальным и выборочно сохраняет

[1] http://dspace.nbuv.gov.ua
[2] http://nbuv.gov.ua/node/1539

только необходимую для последующей обработки информацию, то для каждого электронного каталога (источника данных) алгоритм обхода страниц по ссылкам должен быть адаптирован.

Локально копируется информация двух видов: HTML страницы и полные тексты авторефератов. Сохраненные HTML страницы используются поисковым роботом для дальнейшей навигации по страницам, а также для дальнейшей обработки, поскольку содержат также полезную информацию о документах. Полные тексты представлены файлами различных форматов, таких как PDF, RTF, DOC, которые сохраняются в оригинальном виде для последующей обработки.

После того, как все страницы результатов поиска были сохранены, были также закачаны все полные тексты. Их оказалось довольно много, как уже говорилось выше, более 63 тыс. авторефератов, что в общей сложности заняло 8.5 Гб.

Поисковый робот написан на PHP. Чтобы облегчить задачу адаптации программы под конкретный каталог, все вспомогательные функции вынесены из основной части. Обнаружение и извлечение ссылок для перехода на следующие страницы осуществляется с помощью языка XPath.

XPath является лаконичным языком для поиска узлов в XML документах, в частности им можно обрабатывать HTML страницы. Его использование позволяет сократить затраты на адаптацию алгоритма под разные ресурсы.

3. Обработка HTML страниц

Поскольку страницы, такие как html или xml, можно отнести к слабоструктурированным данным, где нет строгих типов, как в реляционных базах, и дерево может ветвиться очень по-разному. Извлечение нужных данных из корпуса таких страниц может осуществляться разными способами:

- с помощью регулярных выражений[3];
- навигацией по объектной модели документа (DOM);
- с помощью языка XPath;
- с использованием селекторов CSS.

Остановимся на них несколько подробнее. Как известно, регулярные выражения используются для формализации поисковых запросов в тексте. Регулярные выражения обладают известными недостатками. При достаточной сложности регулярного выражения (а при парсинге html возникают достаточно сложные регулярные выражения), они становятся очень трудно поддерживаемыми. Если страничка незначительно меняется, регулярное выражение приходится чуть ли не полностью переписывать. Их трудно понять другому человеку, не тому, кто их писал. И самое главное – они игнорируют структуру html, и рассматривают документ, просто как плоский текст. А html – структурированный текст, и почему бы этим не воспользоваться.

Есть и другая возможность разбора html, используя объектную модель документа, как это делают браузеры, которые используют структуру DOM для отображения документа. То есть на языке программирования входить в эту структуру и выбирать нужные узлы. Недостаток такого способа извлечения данных – слишком громоздко и зависимо от языка. Это также трудно поддерживать, поскольку опять же, если структура странички изменится, то придется переделывать и код программы.

Поэтому для решения поставленной задачи наиболее предпочтительным оказался язык для поиска выражений в xml-структурах. Называется этот язык XPath (икс-путь). Расшифровывается как xml path language, т. е. язык xml-путей. Он используется во многих стандартах xml. В отличие от регулярных выражений он позволяет писать запросы не как к плоскому тексту, а как к дереву. При этом запросы получаются достаточно простыми. Подмножества этого языка – селекторы CSS. Это тоже удобный инструмент, еще проще, чем язык XPath, но XPath – мощнее. Поэтому далее более подробно рассмотрим этот язык.

[3] Язык регулярных выражений, основанный на автоматах, для поиска подстрок в тексте.

Как уже было сказано выше, XPath – язык для навигации по xml-документу. Если в xml-документе необходимо найти какой-нибудь узел, то это можно сделать, либо обходом xml-дерева, либо просто воспользоваться готовым языком. Он позволяет небольшим выражением найти все узлы, которые соответствуют определенным критериям: по имени, по атрибуту узла и по другим условиям, по которым нам хотелось бы искать эти узлы.

XPath – весьма популярный язык, является стандартом консорциума W3C с 1999 года. Он лежит в основе многих других стандартов XML (рис. 4). Такие, например, как XQuery – очень мощный язык не только запросов к большим коллекциям xml-документов, но и есть полноценным языком программирования. В последние годы наблюдается перенос XQuery в языки программирования, на нем можно писать даже сайты, как на PHP, только в основе лежит XML.

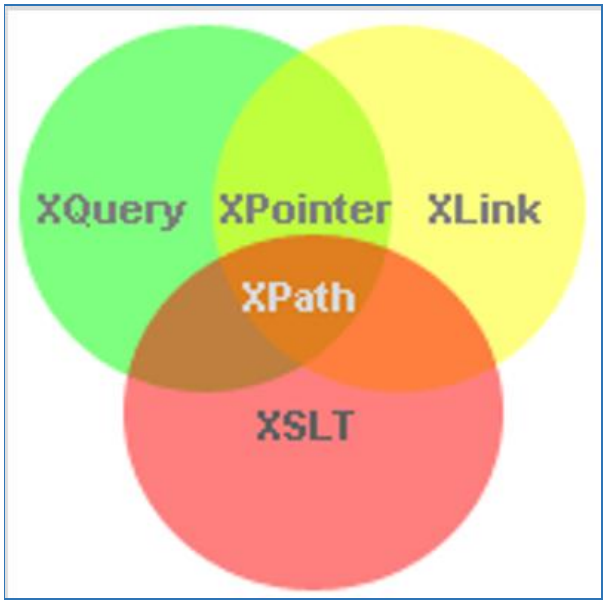


Рис. 4. Язык XPath как основа других XML стандартов консорциума W3C

Выражения XPath представляют собой некоторый путь. Например, на рис. 5. показаны html-узлы, где путь в корне *tr/td/a* – строка таблицы, потом ячейка таблицы, потом ссылка. То есть для того чтобы найти все ссылки документа, которые лежат в ячейках таблицы, можно написать такой простой запрос. Для сравнения можно представить, сколько бы занимал обход дерева на каком-нибудь языке программирования или каким бы регулярным выражением пришлось бы отыскивать все такие ссылки.

- Путь — /tr/td/a
  - Шаги — tr
    - Ось — **child::**tr
    - Проверка узла — child::**tr**
    - Предикат(ы) — [text()='a']
- Функции — position()
- Операции — |, //
- Контекст

Рис. 5. Основы языка XPath

Путь разбивается на шаги, каждый шаг представляет собой некий узел в дереве xml-разбора (в нашем случае html-разбора). Каждый шаг делится на ось, по которой происходит поиск. Ось это – либо дочерние узлы, либо родительские узлы, либо атрибуты, либо соседние с данным узлом узлы. Вторая часть шага – проверка узла, т. е. осуществляется поиск всех узлов, совпадающие с *tr*, строкой таблицы. В итоге третья часть шага – один или несколько предикатов, которые еще больше сужают область поиска. Например, если внутри узла у нас должен содержаться текст ‘a’, то можно написать следующий предикат: [text()=’a’], дополнительно фильтрующий множество найденных узлов.

Также в XPath определено много дополнительных функций, которые позволяют вычислять позицию этого узла, а также есть много строковых и арифметических функций. Есть также операции, которые позволяют объединять результаты, пропускать узлы в дереве, вплоть до дочерних. И что важно, XPath осуществляет поиск, начиная не обязательно от корня, а от любого узла в дереве, если ему задан контекст, в котором искать.

С помощью этих не хитрых возможностей можно делать весьма сложные запросы к html-документам. Вот несколько примеров. На рис. 6 приведены фрагменты

скрипта, который выбирает данные из html-страницы результатов поиска.

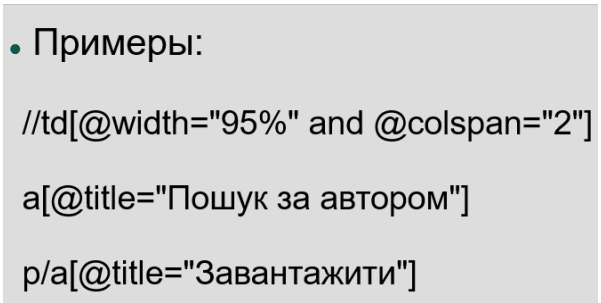


Рис. 6. Примеры поисковых выражений XPath

Вначале мы ищем все узлы на любом уровне дерева, которые являются ячейкой данных таблицы с атрибутами *width="95%" and* (здесь логическое “и”) *colspan="2"*. Так мы находим все ячейки в таблице, где находится один результат поиска. Дальше, используя этот узел как контекстный узел, мы ищем имя автора следующим образом. Находим ссылку (a) с атрибутом *title="Пошук за автором"* и таким образом найдем фамилию и инициалы автора. Если же мы найдем ссылку (a), которая является дочерним узлом параграфа (p) и имеет атрибут *title="Завантажити*", то мы найдем ссылку на полный текст.

Таким образом, сохранённые в локальном хранилище HTML документы из каталога электронной библиотеки уже содержат некоторую информацию о диссертациях: имя автора, название, научную степень. Эту информацию достаточно легко извлечь, используя XPath.

Для этого на языке PHP была написана программа, которая извлекает нужные данные из HTML страницы и сохраняет их в базу данных.

4. Обработка неструктурированных документов

Полные тексты авторефератов представлены в различных форматах, таких как RTF, PDF, DOC, DOCX (рис. 7). Поскольку в этих форматах зафиксировано форматирование и не хранится семантическая информация, то с ними можно работать только после извлечения текстовых данных. Для каждого формата необходима своя программа, преобразующая входной файл в текстовый формат.

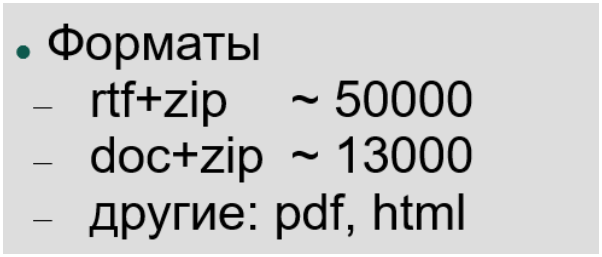


Рис. 7. Корпус текстов авторефератов диссертаций с точки зрения их форматов

Поскольку большинство файлов выбранного библиотечного ресурса представлено в формате RTF, в первую очередь была разработана программа для преобразования этого формата в обычный текст. И следующий модуль с помощью регулярных выражений извлекает из текста целевые сущности.

Формат RTF – довольно простой. Для работы с ним был взят готовый парсер, с последующей его доработкой. Преимуществом данного формата для нас явилось то, что внутреннее его представление не бинарное, а в символах ASCII. Вся структура документа состоит из двух сущностей: это – *управляющие слова*, которые начинаются с обратного слэша и *группы*, которые получаются обрамлением фигурными скобками. Благодаря группам, удалось сразу удалить много “лишнего” из этого формата: всякие бинарные данные, картинки, шрифты, палитры цветов. Все остальное, что не является управляющими словами и группами, это символы. В случае кириллицы мы там символов не увидим, поскольку кириллические символы являются управляющими словами, и тут еще возникает задача перекодировки символов. То есть готовый парсинг нуждался в усовершенствовании, для того чтобы не приходилось после него еще и преобразовывать кодировку. После rtf-парсинга на выходе получаем простой неразмеченный текст.

После преобразования текст поступает на вход алгоритму, распознающему сущности, представляющие интерес для целей системы: имена научных руководителей, имена оппонентов, научные степе-

ни, названия организаций, в которых проводилась подготовка и защита диссертации, дата защиты, номер специальности. Всё, что пока удалось извлечь, сохраняется в базе данных.

Здесь еще предстоит дальнейшая работа над rtf-документом по извлечению именованных сущностей из плоского текста без структуры. Здесь нужно отметить, что решение этой задачи находится на начальной стадии. В настоящее время мы извлекаем из плоского текста регулярными выражениями, используя некоторые признаки. Например, порядок следования слов, фамилия обычно идет с инициалами или три слова с заглавными буквами; заголовок – все заглавные буквы; специальность – последовательность чисел с точками; для УДК тоже есть свой символьный формат или шаблон и т. д.

Пример регулярного выражения, извлекающего из плоского текста, имя автора, название диссертации, УДК и специальность:

/^.*?([$upper]+[$lower]* [$upper]+[$lower]* [$upper]+[$lower]*) ([Уу][ ]?[Дд][ ]?[Кк][:]?[ ]?[-–0-9.'“”»«:;\/\+\[\]\(\)ЄС]+) (.*?) ([0-9][ ]?[0-9][ ]?\.[ ]?[0-9][ ]?[0-9][ ]?\.[ ]?[0-9][ ]?[0-9][.]?[-]?) /

Стартовая программа прошла тестирование вначале на одном автореферате, а затем регулярные выражения были настроены на большее число авторефератов (в экспериментальном корпусе их было 300, выборка за месяц).

На рис. 8 показан фрагмент таблицы базы данных, где можно увидеть

yearmonth	id	author	name	link	fio	udc	nazvanie	speciality
201002	1	Дідора В.Г.	Агроекологічне обгрунтування технології виробництв...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	ВІ КТОР ГРИГОРОВИЧ	УДК 633.521:581.5/47	АГРОЕКОЛОГІ ЧНЕ ОБГРУНТУВАННЯ ТЕХНОЛОГІЇ ВИРО БН И...	06.01.09
201002	2	Сердюченко О.В.	Адміністративно-правові засади забезпечення енерге...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	ЕРДЮЧЕНКО Олексій Володимирович	УДК 347:620.9(477)	АДМІНІСТРАТИВНО-ПРАВОВІ ЗАСАДИ ЗАБЕЗПЕЧЕННЯ ЕНЕР Г...	12.00.07
201002	3	Гридасов Ю.В.	Адміністративно-правові засоби захисту права власн...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	ГРИДАСОВ ЮРІЙ ВОЛОДИМИРОВИЧ	УДК 342.951	АДМІНІСТРАТИВНО-ПРАВОВІ ЗАСОБИ ЗАХИСТУ ПРАВА ВЛАСН...	12.00.07
201002	4	Фурман О.А.	Активізація навчально-пізнавальної діяльності майб...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	УРМАН Олена Андріївна	УДК 378.851	АКТИВІЗАЦІЯ НАВЧАЛЬНО-ПІЗНАВАЛЬНОЇ ДІЯЛЬНОСТІ МАЙБ...	13.00.02
201002	5	Гаврилків В.М.	Алгебро-топологічні структури на суперрозширеннях	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	Гаврилків Володимир Михайлович	УДК 51 2.53	АЛГЕБРО-ТОПОЛОГІЧНІ СТРУКТУРИ НА СУПЕРРОЗШИРЕННЯХ	01.01.0 6
201002	7	Репетєва О.В.	Антигіпоксичні, антиоксидантні та антиексудативні ...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	РЕПЕТЄВА ОЛЕНА ВАЛЕРІЇВНА	УДК : 615.275.4:547.	АНТИГІПОКСИЧНІ, А НТИОКСИДАНТНІ ТА АНТИЕКСУДАТИВНІ...	14.03.05
201002	8	Нестерук С.В.	Багатоцільова антенна решітка з цифровим керування...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	Нестерук Сергій Володимирович	УДК 621.396.677	БАГАТОЦІЛЬОВА АНТЕННА РЕШІТКА З ЦИФРОВИМ КЕРУВАННЯ...	05.12.0 7
201002	9	Ткач Є.В.	Банківське кредитування інноваційного розвитку: вза... взаємозв'язок жанрів: новітн...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	Ткач Євген Вікторович Василівна	УДК 336.77:334.012.6 8	БАНКІВСЬКЕ КРЕДИТУВАННЯ ІННОВАЦІЙНОГО РОЗВИТКУ: ВЗАЄМОЗВ'ЯЗОК ЖАНРІВ: НОВІТН...	08.00.08
201002	14	Воліков І.О.	Вибір варіанта запобіжної аналгезії в абдомінальні...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	ВОЛІКОВ ІГОР ОЛЕКСАНДРОВИЧ	УДК: 617.55-089:616.	ВИБІР ВАРІАНТУ ЗА ПОБІЖНОЇ АНАЛГЕЗІЇ В АБДОМІНАЛЬН...	14.01.30
201002	15	Бегма Н.А.	Використання побічних продуктів переробки зерна ку...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	Бегма Наталія Андріївна	УДК 636.4.087.74	Використання побічних продуктів переробки зерна ку...	06.02.02
201002	16	Костащук В.І.	Використання та охорона мінерально-сировинних ресу...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	Костащук Володимир Іванович	УДК 911.3: 330.15 (4	В икористання та охорона мінерально-сировинних рес...	11.00. 02
201002	21	Василенко І.В.	Виховання громадянськості студентів засобами самов...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	ВАСИЛЕНКО ІРИНА ВІТАЛІЇВНА	УДК : 37.0 34:37.3:3	ВИХОВАННЯ ГРОМАДЯНСЬКОСТІ СТУД ЕНТІВ ЗАСОБАМИ САМО...	13.00.07
201002	22	Дарчик С.В.	Візія національно-визвольної війни українського на...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	ДАРЧИК Світлана Василівна	УДК 930.2 (477) "1	ВІЗІЯ НАЦІОНАЛЬНО-ВИЗВОЛЬНО Ї ВІЙНИ УКРАЇНСЬКОГО Н...	07.00.06
201002	25	Олексін М.І.	Вплив дисперсії і нелінійних ефектів оптичного вол...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	Олексін Михайло Іванович	УДК 621.391 .6 : 535	ВПЛИВ ДИСПЕРСІЇ І НЕЛІНІЙНИХ ЕФЕКТІВ ОПТИЧНОГО ВОЛ...	05.12.02
201002	29	Красівська В.В.	Вплив на гемостаз паталогічних інгібіторів зсіданн...	http://irbis-nbuv.gov.ua//cgi-bin/irbis64r_81/cgii...	СІВСЬКА Валерія Валеріївна	УДК : 616.15-06+616.	ВПЛИВ НА ГЕМОСТАЗ ПАТОЛОГІЧНИХ ІНГІБІТОРІВ ЗСІДАНН...	14.01.31
201002	30	Плахотнюк	Вплив стану молочної залози	http://irbis	ПЛАХОТНЮК ІГОР	УДК 619:618.11	ВПЛИВ СТАНУ МОЛОЧНОЇ	16.00.07

Рис. 8. Фрагмент результирующей таблицы извлечения данных

информацию, полученную в результате парсинга html-страницы (авторефераты за месяц-год) и текстов самих авторефератов, где:

yearmonth – значение года месяца поступления автореферата;

id – идентификатор автореферата;

author – фамилия и инициалы автора авторефераты, извлеченные из html-страницы запроса;

name – название автореферата (диссертации), извлеченное из html-страницы запроса;

link – адрес ссылки, где находится полный текст автореферата;

fio – фамилия имя и отчество. извлеченные из текста автореферата;

udc – значение классификатора УДК, извлеченное из текста автореферата;

nazvanie – название автореферата (диссертации), извлеченное из текста автореферата;

speciality – значение классификатора специальность, извлеченное из текста автореферата.

В дальнейшем нам нужно научиться извлекать из плоского текста персоны и организации. Можно использовать готовые библиотеки. Есть желание их сравнить и реализовать методы самостоятельно. При извлечении именованных сущностей можно использовать такие ресурсы (мы называем их Словарями), как Википедия и Викидата для верификации этих именованных сущностей. Поскольку, например, организации уже есть как в Википедии, так и в базе данных Викидата.

5. Хранение данных

Набор атрибутов персоналий не фиксирован и может неограниченно расширяться, поэтому необходима модель данных, учитывающая эту особенность. Поскольку конечной целью является представление данных в Интернете, желательна модель данных легко совместимая с семантической сетью. Подходящей моделью данных является RDF.

Выводы

В работе рассмотрены подходы к решению задачи сбора и извлечения разрозненных данных о персоналиях из слабоструктурированных и неструктурированных документов, представленных в общедоступных каталогах авторефератов диссертаций.

На языке PHP с применением XPath разработана система, которая позволяет автоматически собирать первичные документы из электронного каталога Национальной библиотеки Украины им. В.И. Вернадского, извлекать из этих документов данные и сохранять их в локальном хранилище.

Для хранения выбрана модель данных RDF с учётом особенностей данных и возможностью последующего представления в семантической сети.

Литература

1. Проскудина Г.Ю., Кудим К.А. О технологии использования внешних данных при создании и редактировании энциклопедических текстов. *Проблеми програмування.* 2017. № 1. С. 67–82.
2. Markus Krötzsch, Michael Günther, Markus Damm & Georg Wild (2016). SQID - Searching, Querying, and Interacting with Data. [online – https://tools.wmflabs.org/sqid].

References

1. PROSKUDINA G. & KUDIM K. (2017). About technologies of use of external data on creating and editing of encyclopedic text. *Problems in programming*. [online – pp.isofts.kiev.ua] (1). P. 67–82. (in Russian). Available from: http://pp.isofts.kiev.ua /ojs1/article/view/223 [Accessed 6/02/2017].
2. MARKUS KRÖTZSCH, MICHAEL GÜNTHER, MARKUS DAMM & GEORG WILD (2016). SQID - Searching, Querying, and Interacting with Data. [online – https://tools.wmflabs.org/sqid]. (in English). Available from: http://pp.isofts.kiev.ua/ojs1/ article/view/145 [Accessed 6/06/2017].

Получено 10.05.2019

Об авторах:

Кудим Кузьма Алексеевич,
младший научный сотрудник.
Количество научных публикаций в
украинских изданиях – 16.
Количество научных публикаций в
зарубежных изданиях – 2.
http://orcid.org/0000-0001-9483-5495,

Проскудина Галина Юрьевна,
научный сотрудник.
Количество научных публикаций в
украинских изданиях – 29.
Количество научных публикаций в
зарубежных изданиях – 15.
http://orcid.org/0000-0001-9094-1565.

Место работы авторов:

Институт программных систем
НАН Украины,
03187, Киев-187,
проспект Академика Глушкова, 40.
Тел.: (044) 526 3559.
E-mail: guproskudina@gmail.com,
kuzmaka@gmail.com