Разработка системы семантического анализа слабо структурированной информации на основе онтологического подхода

В. В. Гаршина[1], И. В. Илларионов, Э. К. Алгазинов
Воронежский государственный университет
Воронеж, Россия
[1]garshina@cs.vsu.ru

Е. Н. Десятирикова[1], С. А. Чепелев, В. И. Акимов
Воронежский государственный технический университет
Воронеж, Россия
[1]science2000@ya.ru

Аннотация. **В работе приводится архитектура системы семантического анализа слабо структурированной информации на основе онтологической парадигмы; представлены технологии ее реализации. Особенностью является использование средств Tomita Parser для работы с русскоязычными источниками. Рассмотрен процесс интеграции онтологий разных уровней применительно к процессам управления предприятием на базе стандартов ISO серии 15926. Описано решение задачи реализации вывода на узлах онтологии.**

Ключевые слова: системы поддержки принятия решения; экспертные системы; онтологии; семантический анализ; анализ неструктурированной и слабоструктурированной информации

I. Введение

Слабоструктурированная информация широко используется процессами управления предприятием, но является слабо используемым источником бизнес-аналитики: Data Mining, смысловой обработки содержащихся в документах фактов (Text Mining) и вывода закономерностей на этих данных [1, 2]. Это связано с такими проблемами, возникающими при разработке подобных интеллектуальных аналитических систем, как: лингвистические трудности разработки систем разбора текстовых документов (парсеров); семантическое сопоставление структур данных, разных информационных систем, участвующих в процессе управления предприятием; смысловая интеграции разнородных по типу представления и источникам происхождения данных в единую семантическую модель предметной области [3]; реализация технологий обращения к данным, технологий обработки и аналитики с учетом семантики предметной области [4].

В статье приводится описание архитектуры технологической платформы для решения перечисленных проблем, реализованной в прототипе системы анализа контрактных отношений с поставщиками.

II. Материалы и методы

За основу смысловой аналитики разрабатываемой системы выбраны стандартизованные технологии SemanticWeb. Предлагаемая архитектура технологической платформы для системы, поддерживающей семантическую обработку и анализ слабоструктурированной информации, содержит сгруппированные по функциям компоненты. Ядром является *Онтологическая модель предметной области*, разработанная в соответствии с существующими стандартами предметной области и расширенная понятиями локальной области задач предприятия, и связанная с ней система наполнения, хранения и интеграции онтологий. Онтология расширяется базой правил-аксиом, обеспечивающих проведение выводов по онтологии (запросов), и базой логических правил, отвечающих за более высокие уровни семантической аналитики.

Первичные данные предметной области поступают в систему из различных источников: документы, файлы, таблицы, внешние информационные системы, письма, результаты аналитики (Data Mining, Big Data). Они проходят соответствующие процедуры обработки и интегрируются с онтологией. Варианты интеграции: наполнение онтологии структурированными данными; данными, полученными в результате обработки слабоструктурированной информации; расширение новыми концептами (расширение модели предметной области обучением на основе прецедентов) [5]. Выделение фактов из текстовых источников реализуется на основе парсеров естественного языка, библиотек грамматик, разработанных для локальных предметных областей, и генератора «общих» грамматик по онтологии.

Процесс интеграции данных, использующихся разными информационными системами предприятия (данных реляционной СУБД; локальных информационных систем; входных данных), может быть решен на основе преобразования данных в синтаксис RDF (Turtle) [6]. Онтология, интегрирующая эти данные, должна использовать существующие стандарты понятийного аппарата процесса управления предприятием – набор стандартов ISO 15926. Он представлен набором стандартных онтологий разного уровня, разрабатываемых для различных отраслевых применений [7].

В рамках реализации прототипа системы была проведена интеграция разработанной онтологии контрактных отношений с поставщиками к онтологии верхнего уровня стандарта ISO 15926 с использованием онтологического редактора Protege (методом ручного объединения). Таким

образом, была обеспечена структурная и лексическая стандартизация модели предметной области.

III. Семантический анализ слабоструктурированной информации на основе онтологии контрактных отношений CRM

Важной функцией любой информационной системы управления предприятием является ведение базы договоров с контрагентами (поставка продукции, выполнение работ, оказание услуг, аренда, и прочее). Учет и отслеживание истории выполнения договоров необходимы для планирования работы компании, контроля и анализа взаимоотношений по цепочкам CRM [8]. База договоров компании обширна, использование БД (которые лишь хранят их в удобной для поиска форме) не позволяет извлекать из них множество важных знаний. Решением проблемы может быть информационная система, обеспечивающая поиск и выделение фактов и знаний из документов компании, поддерживающая онтологическое представление и семантические технологии для их обработки. Также, ведение актуальной онтологической модели договорных взаимоотношений компании и аналитика на ней могут быть использованы для автоматизации всего комплекса CRM/SRM задач организации.

В качестве инструмента разработки онтологий был выбран свободно-распространяемый редактор Protégé 5.2.0, позволяющий использовать графическое редактирование, несколько встроенных резонеров для проверки построенной онтологии на предмет возможных противоречий.

Анализ базы договоров организации позволил выделить следующие понятия предметной области (классы онтологии) и связи между ними: Документы (договор, устав, доверенность); Действия (оплата, передача права, возложение обязанности); Лица (публичные и частные, юридические и физические); Объекты (материальные и

Рис. 1. Часть онтологии, описывающая участвующих в договоре лиц

нематериальные, движимость и недвижимость); Норма; Предмет; Представитель; Сторона. Объекты объединены в состав других объектов – родителей. В некоторых случаях в качестве значения свойства выставлен другой объект (например: Договор – имеет сторону – Лицо, поскольку стороной договора может быть частное/публичное лицо, юридическое/физическое лицо). Это позволяет использовать наследование в описании свойств (атрибутов) объектов. Также используются литералы – типизированные или не типизированные строки символов, например, наименование, дата, стоимость, организационно-правовая форма. Связь (свойство, предикат) соединяет объект с объектом (Object Properties) или объект с литералом (Data Properties). Пример части разработанной онтологии представлен на рис. 1.

IV. Этапы работы системы извлечения неструктурированной текстовой информации

Работа интеллектуальной системы состоит из нескольких этапов, которые можно представить на следующей схеме (рис. 2).

1) Этап нормализации исходного текста. Предполагается, что текстовый документ может быть представлен в разных форматах, поэтому выполняется конвертация в формат .txt, а также разбиение его на части с целью применения к ним различных подходов при анализе. Этап реализуется созданным приложением (платформа .NET, а язык – C#).

2) Этап извлечения фактов и данных (парсинг). Для дальнейшего наполнения созданной онтологии фактами необходим механизм извлечения данных из текста. Для русского языка был выбран *Tomita Parser*, свободно распространяемый инструмент от Yandex [9]. Он позволяет создать конфигурационные файлы (словари, грамматики) применительно к типам анализируемых текстов. Разрабатываемый набор файлов для работы парсера включает в себя: основной конфигурационный файл; словари (минимум один, корневой); файлы грамматик; файлы, описывающие типы извлекаемых фактов, если происходит извлечение фактов; файлы ключевых слов, если происходит определение новых ключевых слов.

Текст договора имеет структуру: начальная часть договора содержит название, место и дату заключения договора, а также сведения о его сторонах – их наименования, роли, представители. Далее текст договора также разделен на части (права и обязанности сторон, предмет договора, сроки, ответственность, заключительные положения и т.д.). В связи с этим, различные группы грамматик можно применять как к частям, так и к договору в целом. В текстах договоров, можно выделить устоявшиеся речевые обороты: «в лице», «на основании», «действующего на основании», «имеет право», «заключен на срок». Их можно использовать в соответствующих грамматиках для извлечения фактов и обращения к соответствующим элементам онтологии. Например, при наличии следующего исходного текста: “Общество с ограниченной ответственностью «Рога и копыта» в лице управляющего Бендера Остапа Ибрагимовича” для извлечения сведений о наименовании, организационно-правовой форме, сторонах договора, а также о представителе – физическом лице и его должности, можно написать грамматики в нотации *Tomita Parser*.

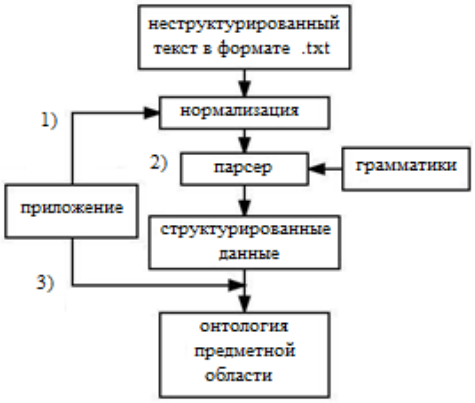

Рис. 2. Схема этапов

Общая грамматика, извлекающая данные в факт *ContractParty*-сторона договора (поля *Form* – организационно-правовая форма, *Title* – наименование, *RepresentativePosition* – должность, *RepresentativeFIO* – ФИО):

*S -> **OPF** interp (ContractParty.Form)*
__Title__<quoted> interp (ContractParty.Title) (',')
__InPersonOf Position__ interp (ContractParty.RepresentativePosition)
__FullName__ interp (ContractParty.RepresentativeFIO);

Заполнение полей факта обозначается ключевым словом *interp*. Выделенные жирным шрифтом фрагменты, в свою очередь, также являются грамматиками (вспомогательными). Грамматика организационно-правовой формы юридического лица:

OPF -> Word<kwtype=opf>;

Здесь *opf* это название пользовательского типа статей - словарь, содержащий перечисление всех возможных организационно-правовых форм юридических лиц и возможные их синонимы («ООО»; «Общество с ограниченной ответственностью»). Выражение *kwtype=opf* означает, что учитывается каждое слово (сочетание слов), хранящееся в словаре.

Грамматика наименования юридического лица:

Title -> Word<h-reg1> Word;*

Минимальное количество слов в наименовании – одно (первый терминал Word), которое пишется с большой буквы (помета *h-reg1*), также возможны еще ноль или более слов (оператор *). В основной грамматике Title применен с пометой *quoted*, означающей, что наименование располагается в кавычках.

Грамматика выражения, означающего, что какое-то юридическое лицо выступает в роли стороны договора в лице представителя – физического лица:

InPersonOf -> 'в' 'лицо';

Эта морфологическая лемма означает устойчивый набор слов, каждое из которых приведено в нормальную форму. В случае, если возможны несколько вариантов написания данной части текста, они так же прописываются один за другим. Например, можно добавить вариант *InPersonOf ->* 'представлять'.

Грамматика должности физического лица – представителя:
Position -> (Adj<gnc-agr[1]>) Noun<kwtype="должность", gnc-agr[1], rt>;
Position -> Noun<kwtype="должность", rt> Prep Noun;

Здесь показан случай двух вариантов грамматики, в обоих используется словарь должностей (управляющий, директор, председатель, представитель, глава и прочее). Первая грамматика состоит из необязательного прилагательного (*Adj* с оператором круглых скобок), а также существительного из словаря должностей. Помета *gnc-agr[1]* означает согласование двух слов по роду (*gender*), числу (*number*) и падежу (*case*). Вторая грамматика состоит из существительного из словаря должностей, а также предлога и существительного. Первая грамматика сработает на таких вариантах, как «исполнительный директор», «генеральный директор», «председатель». Вторая – определит факт «представитель по доверенности». При желании список грамматик можно расширять.

Грамматика имени физического лица следующая:
Initial -> Word<wff=/[А-Я]\./>;
Initials -> Initial<h-reg1> Initial<h-reg1>;
FullName -> Initials Word<gram="фам"> |
Word<gram="фам"> Initials |
Word<gram="фам", gnc-agr[1]> Word<gram="имя", gnc-agr[1], gnc-agr[2], rt> Word<gram="отч", gnc-agr[2]>;

Результат – нетерминал *FullName* – возможен из трех вариантов, разделенных оператором “|” (дизъюнкция).

В первом варианте используется вспомогательный нетерминал *Initials* (инициалы), состоящий из двух инициалов, каждый из которых представлен заглавной буквой русского алфавита (с помощью пометы *wff* задается регулярное выражение), к которым присоединяется слово с пометой *gram="фам",* означающее фамилию. Во втором варианте терминалы и нетерминалы расположены в другом порядке. В третьем случае ФИО записывается полностью, следовательно, представляет три слова, согласованных между собой. Грамматика сработает на следующие варианты написания ФИО: О.И. Бендер, Бендер О.И., Бендер Остап Ибрагимович.

3) Этап перенесения полученных фактов в онтологию. Формируются триплеты из полученных фактов и заносятся в онтологию. Реализуется приложением (платформа .NET, а язык - C#), с использованием библиотек для работы с XML и OWL файлами.

4) Решение задачи реализации вывода на узлах онтологии через запросы SPARQL [10] с использованием предикатов русского языка. Привязку концептов, используемых в онтологии, к языкам можно проводить через присвоение классам, атрибутам классов (слотам), отношениям (связям между концептами) меток Label с атрибутом-указателем на язык ('en' 'fr' ...). Метки задаются в редакторе Protege в момент создания Label. Используя Label с атрибу-

том-указателем языка, можно проводить отображение онтологии (граф и классы), соответствующее концептам (классам и атрибутам классов онтологии). Также плагин Protégé Ontograph позволяет отображать онтологию по Label, то есть видеть ее на других ассоциированных с ней языках. Механизм связи Label с указателем языка используется в проекте для нахождения соответствий полученных текстовых фактов из неструктурированных документов на основе грамматик *Tomita Parser* с концептами (классами, атрибутами и отношениями), заданными в онтологии предметной области. Также через обращение к Label проводится вывод заключений на онтологии (через SWRL TAB), используя лексику русского языка.

5) Пример запроса поиска (SPARQL) по русскому значению предиката

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX contractsbase:
<http://www.semanticweb.org/alex/ontologies/2017/10/ContractsBase#>
SELECT ?org ?title
WHERE
{ ?org rdf:type contractsbase:Organization.
      ?org ?type ?title.
      ?type rdfs:subPropertyOf contractsbase:hasText.
      ?type rdfs:label ?rusLabel.
      filter(str(?rusLabel)="имеет наименование")}
```

Запрос реализуется следующим образом: из онтологии выбираются все сущности (?org), имеющие тип contractbase:Organization, затем, выбираются все текстовые (contractbase:hasText) свойства этих сущностей (?propName). Положим, у этих тестовых свойств есть label (rdfs:label). Обозначим его ?rusLabel. Значение этого ?rusLabel равно “имеет наименование”. Для вывода самих наименований нужна еще одна переменная – ?title. В результате, по онтологии выведутся все организации с их наименованиями, с которыми наша компания взаимодействовала.

6) Пример запроса поиска (SPARQL) по русскому значению объекта

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX contractsbase:
<http://www.semanticweb.org/alex/ontologies/2017/10/ContractsBase#>
SELECT ?org ?title
WHERE
{ ?org rdf:type contractsbase:Organization.
      ?org ?type ?title.
      ?type rdfs:subPropertyOf contractsbase:hasText.
      filter(str(?title)="ООО Рога и копыта")}
```

Запрос реализуется следующим образом: из онтологии выбираются все сущности *(?org)* имеющие тип *contractbase:Organization*. Положим, у организации *(?org)* есть свойство *(?propName),* имеющее значение *?value.* Это свойство – текстовое *(contractbase:hasText),* значение этого свойства – *«ООО Рога и копыта».* Что это за свойство? Вывод*: “имеет наименование”.*

Заключение

Можно утверждать, что онтологический подход является эффективным и реализуемым для следующих задач: - семантическая интеграция информационных систем, участвующих в процессе управления предприятием;

- смысловая интеграция разнородных по типу представления и источникам происхождения данных;
- разработка аналитических и интеллектуальных систем, учитывающих семантику предметной области.

В статье показаны подходы к решению этих задач через использование стандартов и технологий разработки семантических систем и посредством интеграции open source информационных программных продуктов.

Список литературы

[1] Volkova V.N., Vasiliev A.Y., Efremov A.A., Loginova A.V. “Information technologies to support decision-making in the engineering and control”, in Proc. of 2017 20th IEEE SCM, St. Petersburg, Russia; 24-26 May 2017, pp.727-730. DOI: 10.1109 /SCM.2017.7970704.

[2] Volkova V.N., Efremov A.A., Loginova A.V., Leonova A.E. “The conception of information system for support decision-making under conditions of territoriale-distributed databases”, in Proc. of 2017 IEEE 2nd CTS 2017, St. Petersburg, Russia, 25-27 October 2017, pp. 79-82. DOI: 10.1109/CTSYS.2017.8109493

[3] Chernenkaya L.V., Desyatirikova E.N., Belousov V.E., Chepelev S.A., Sergeeva S.I., Slinkova N.V., “Optimal planning of distributed control systems with active elements”, in Proc. of 2017 IEEE 2nd CTS 2017; St. Petersburg; Russian Federation; 25-27 October 2017, pp.39-42 DOI: 10.1109/CTSYS.2017.8109482

[4] Zegzhda P.D., Zegzhda D.P., & Stepanova T.V. “Approach to the construction of the generalized functional-semantic cyber security model”, *Automatic Control and Computer Sciences*, vol.49, issue 8, 2015, pp. 627-633. DOI:10.3103/S0146411615080192

[5] Mager V.E., Belousov V.E., Desyatirikova E.N., Polukazakov A.V., Ivanov S.A., Pocebneva I.V. “Information processing algorithm at creation of optimum structure of the self-adjusted technical system in quality parameters”, in Proc. of 2017 IEEE 2nd CTS 2017; St. Petersburg; Russian Federation; 25-27 October 2017, pp.118-121. DOI: 10.1109/CTSYS.2017.8109503

[6] Gorshkov S. (2013) Applied Semantics for Integration and Analytics. In: Klinov P., Mouromtsev D. (eds) Knowledge Engineering and the Semantic Web. KESW 2013. Communications in Computer and Information Science, vol 394. Springer, Berlin, Heidelberg

[7] Igamberdiev M., Grossmann G., Selway M. et al. Softw Syst Mode, 2018, 17: 269. https://doi.org/10.1007/s10270-016-0520-6

[8] Desyatirikova E.N., Belousov V.E., Zolotarev V.N., Lavlinskaia O.Yu., “Design process of software quality management”, in Proc. of 2017 IEEE IT and QM and IS; St. Petersburg; Russia; 24-30 September 2017 , pp.496-499. DOI: 10.1109/ITMQIS.2017.8085870

[9] Dubov M., Mirkin B., Shal A.: Automatic russian text processing system. Open Systems DBMS, 2014, **22**(10), pp.15–17.

[10] Gorshkov S. Access Control, Triggers and Versioning over SPARQL Endpoint. In: Klinov P., Mouromtsev D. (eds) Knowledge Engineering and the Semantic Web. KESW 2014. Communications in Computer and Information Science, vol 468. Springer, Cham.