

**И. А. Корсун<sup>1</sup>, Д. Е. Пальчунов<sup>1,2</sup>**

<sup>1</sup>Новосибирский государственный университет  
ул. Пирогова, 2, Новосибирск, 630090, Россия

<sup>2</sup>Институт математики им. С. Л. Соболева СО РАН  
пр. Академика Коптюга, 4, Новосибирск, 630090, Россия

*irina.korsun.nsu@gmail.com, palch@math.nsc.ru*

## **ТЕОРЕТИКО-МОДЕЛЬНЫЕ МЕТОДЫ ИЗВЛЕЧЕНИЯ ЗНАНИЙ О СМЫСЛЕ ПОНЯТИЙ ИЗ ТЕКСТОВ ЕСТЕСТВЕННОГО ЯЗЫКА \***

Статья посвящена разработке теоретико-модельных методов извлечения знаний из текстов естественного языка, а также методов извлечения из текстов определений понятий, полных относительно фиксированного контекста. Контекст задаётся либо объёмлющей онтологией, либо набором прецедентов предметной области; при этом полнота определения рассматривается относительно множества «интересующих» предложений. Извлечённые из текстов знания представляются в виде фрагментов атомарных диаграмм алгебраических систем. Разработаны и программно реализованы алгоритмы отображения бескванторных предложений логики предикатов в логику описаний (DL), а также их дальнейшего отображения в OWL. Это даёт возможность порождения новых знаний исходя из имеющихся знаний, уже содержащихся в онтологии, при помощи использования автоматических средств логического вывода.

*Ключевые слова:* онтология, теоретико-модельные методы, фрагменты атомарных диаграмм, определения понятий, извлечение знаний, порождение знаний, средства логического вывода.

### **Введение**

Статья посвящена разработке теоретико-модельных методов извлечения знаний из текстов естественного языка, а именно, знаний о смысле ключевых понятий заданной предметной области. Разрабатываются методы извлечения определений понятий, полных относительно некоторого фиксированного контекста. Исследованы разные способы определения относительной полноты данного определения понятия. Знания о смысле понятий извлекаются из набора текстов, написанных на естественном языке. Для этого используются разработанные ранее методы представления знаний, извлечённых из текстов, в виде набора конечных фрагментов атомарных диаграмм алгебраических систем, методы интеграции атомарных диаграмм и порождения таким способом нового онтологического знания, ранее в явном виде не сформулированного.

В настоящее время большое количество работ посвящено использованию логического вывода для работы с онтологиями. В [1] описана программная система, предназначенная для автоматизированной проверки согласованности онтологии экспертами предметной области.

---

\* Исследование выполнено при частичной финансовой поддержке РФФИ в рамках научного проекта № 14-07-00903\_a.

Система использует логический вывод как основу для выявления конфликтов. В [2] предложен подход для кризисного управления и реагирования, в котором используется онтология на языке OWL DL и машина логического вывода для построения цепочек рассуждений. В [3] представлена онтология, которая используется при разработке алгоритмов сравнения и анализа генома.

Одним из популярных современных средств сбора и хранения информации являются Wiki-системы [4]. На основе Wiki-систем разработаны программные системы для построения онтологий верхнего уровня (general-purpose ontology) [5], онтологий персоналий [6], программные системы, предоставляющие интерфейс на естественном языке для редактирования онтологий [7]. Wiki-системы используются в качестве среды для разработки онтологий [8], а также для создания корпуса текстов для построения онтологии конкретной предметной области [9]. Среди недостатков данного подхода следует отметить обеспечение лишь структурной целостности информации, без проверки ее семантической согласованности. Кроме того, используемые онтологии обладают достаточно низкой выразительностью.

Одним из важных направлений исследований является автоматизация построения онтологий из текстов, написанных на естественном языке. При этом используются подходы на основе лексико-синтаксических шаблонов [10], подходы на основе систем продукций [11], подходы на основе статистической информации [12].

Ряд работ посвящён использованию логик описаний (Description Logics, DL) и средств логического вывода для работы онтологическими моделями знаний [13]. В [14] логический вывод используется для обнаружения смысловых ошибок в текстах, представленных в Интернете. В [15] логический вывод используется для проверки наличия противоречий в онтологической модели.

Ранее в [16] нами был разработан теоретико-модельный подход к извлечению знаний из текстов естественного языка. В основе него лежит представление знаний при помощи конечных фрагментов атомарных диаграмм моделей. Были разработаны методы интерпретации различных частей речи и синтаксических связей с целью автоматического порождения сигнатуры модели. Были реализованы в виде программной системы методы автоматического построения атомарных предложений данной сигнатуры по тексту естественного языка [17].

В данной работе предложены алгоритмы отображения бескванторных предложений логики предикатов первого порядка сигнатуры, не содержащей функциональных символов, в логику описаний (DL), а также дальнейшего отображения множеств таких предложений в OWL. Это дало возможность порождения новых знаний исходя из имеющихся знаний, уже содержащихся в онтологии, при помощи использования автоматических средств логического вывода – ризонеров для логик описаний.

Алгоритмы трансляции бескванторных предложений логики предикатов в логику описаний (DL) и в OWL реализованы в виде программной системы. Эта система является модулем разрабатываемой информационной системы, осуществляющей порождение онтологии из фрагментов атомарных диаграмм, порождение новых знаний с помощью машины логического вывода, извлечение определений используемых в онтологии понятий в рамках данного контекста.

### **Теоретико-модельные методы извлечения и представления знаний**

Для представления знаний, извлечённых из текстов естественного языка, в качестве базовой конструкции мы используем атомарные предложения. Далее мы рассматриваем множества предложений логики предикатов первого порядка, каждое из которых является дизъюнкцией атомарных предложений и отрицаний атомарных предложений. При этом сигнатура рассматриваемых предложений не содержит символов функций, а состоит только из символов предикатов и констант.

Дадим необходимые определения и обозначения. Сведения по теории моделей можно найти в [18].

Модели  $\mathfrak{M} = \langle A; \sigma \rangle = \langle A; P_1, \dots, P_n, c_1, \dots, c_l \rangle$  рассматриваются в сигнатуре  $\sigma = \langle P_1, \dots, P_n, c_1, \dots, c_l \rangle$ , где  $A$  – основное множество модели,  $P_1, \dots, P_n$  – символы предикатов

и  $c_1, \dots, c_l$  – символы констант.  $S(\sigma)$  обозначает множество всех предложений сигнатуры  $\sigma$ , т.е. формул без свободных переменных. Запись  $\mathfrak{M} \models \varphi$  означает, что на модели  $\mathfrak{M}$  истинно предложение  $\varphi$ . Запись  $\Delta \vdash \varphi$  означает, что из множества формул  $\Delta$  выводима формула  $\varphi$ .

Для модели  $\mathfrak{M}$  сигнатуры  $\sigma$  мы обозначаем  $\sigma_A = \sigma \cup \{c_a \mid a \in A\}$ ; при этом считаем, что  $c_a \notin \sigma$  при  $a \in A$ . Через  $\mathfrak{M}_A$  мы обозначаем модель сигнатуры  $\sigma_A$ , обеднение которой до сигнатуры  $\sigma$  совпадает с  $\mathfrak{M}$  и значения констант  $c_a$  в обогащённой модели совпадают с самими элементами  $a$ , т.е.  $c_a^{\mathfrak{M}_A} = a$  при  $a \in \mathfrak{M}$ .

Предложение  $\varphi$  назовем атомарным, если  $\varphi = (c_1 = c_2)$  или  $\varphi = P(c_1, \dots, c_n)$ , где  $P, c_1, \dots, c_n \in \sigma_A$ .

В рамках данного подхода мы несколько изменим понятие атомарной диаграммы модели. А именно, атомарной диаграммой модели  $\mathfrak{M}$  сигнатуры  $\sigma$  назовем множество предложений  $AD(\mathfrak{M}) = \{\psi \in S(\sigma_A) \mid \mathfrak{M}_A \models \psi, \psi = \varphi \text{ или } \psi = \neg\varphi, \text{ а предложение } \varphi \text{ – атомарное}\}$ .

Обычно в атомарную диаграмму модели включают только атомарные предложения, истинные на этой модели. При этом, если атомарное предложение не входит в атомарную диаграмму модели, то оно, очевидно, является ложным на этой модели. Поэтому включать в атомарную диаграмму отрицания атомарных предложений, истинные на модели, в этом случае не имеет смысла.

Мы же включаем как атомарные предложения, так и их отрицания, истинные на модели. Это связано с тем, что мы рассматриваем не всю атомарную диаграмму целиком, а только её конечные подмножества – конечные фрагменты атомарной диаграммы. С практической точки зрения всю атомарную диаграмму модели невозможно рассматривать хотя бы потому, что она бесконечна (если сама модель не является конечной). Кроме того, извлекая знания из текстов естественного языка, мы практически никогда не имеем полную информацию о рассматриваемых объектах. Поэтому нам удобно рассматривать конечные фрагменты атомарной диаграммы, считая саму модель потенциально бесконечной: не ограничиваясь каким-то числом количество входящих в неё объектов.

Заметим, что такой подход даёт нам ряд дополнительных возможностей. Это связано, в частности, с тем, что в сигнатуре, содержащей только символы предикатов, любое подмножество основного множества модели образует её подмодель. И обратно, объединение основных множеств двух моделей одной и той же чисто предикатной сигнатуры (при условии, что эти множества не пересекаются) порождает новую модель данной сигнатуры, у которой две данные модели будут подмоделями. Если же основные множества этих двух моделей имеют непустое пересечение, для вложения их в общую надмодель необходимо и достаточно выполнение условия: подмодели этих двух моделей, образованные пересечением, совпадают.

Кроме того, мы можем заранее не фиксировать сигнатуру алгебраической системы, атомарную диаграмму которой собираем из фрагментов. Это позволяет нам добавлять фрагменты атомарной диаграммы, содержащие новые сигнатурные символы: новые предикаты и константы. При интеграции фрагментов атомарной диаграммы необходимо только контролировать непротиворечивость полученного фрагмента. Противоречие может возникнуть, поскольку фрагменты содержат не только атомарные предложения, но и их отрицания.

Для извлечения знаний из текстов естественного языка мы используем результаты наших исследований, начатых в [16]. В этой работе предложен теоретико-модельный подход к извлечению знаний из текстов, основанный на представлении знаний при помощи конечных фрагментов атомарных диаграмм моделей. В [16] разработаны методы интерпретации различных частей речи и различных синтаксических связей при помощи многоместных предикатов. Разработаны методы автоматического построения атомарных предложений на основе обработки предложений естественного языка. Для этого были использованы словарь номинализаций, содержащий более 8000 понятий и словарь валентностей для 2300 глаголов, созданные в рамках выполнения исследования.

Была разработана программная система [17], предназначенная для порождения фрагментов атомарных диаграмм моделей по текстам естественного языка. Программная система реализует разработанные методы и алгоритмы. Программная система может использоваться как в автоматическом, так и в автоматизированном режиме: автоматически построенный

фрагмент атомарной диаграммы модели может быть визуализирован, пользователь может его редактировать.

Фрагмент  $D$  атомарной диаграммы по существу является описанием некоторой ситуации, причём описанием частичным, а не полным. Описание ситуации – это элементы, представляемые константами сигнатуры  $\sigma_A$ : объекты, предметы, люди и т. д., а также свойства элементов, представляемые одноместными предикатами сигнатуры  $\sigma$ , и  $n$ -местные отношения, представляемые  $n$ -местными предикатами сигнатуры  $\sigma$ . Если известно, что данный набор элементов находится (или не находится) в данном отношении, фрагмент  $D$  содержит атомарное предложение – соответствующий предикат от констант, соответствующих этим элементам (или отрицание данного атомарного предложения).

Конечный фрагмент  $D$  атомарной диаграммы, являющийся конечным множеством атомарных предложений и отрицаний атомарных предложений, может быть представлен одним предложением  $\tau_D$  – конъюнкцией всех предложений, входящих в  $D$ . Мы можем рассмотреть несколько фрагментов  $D_1, \dots, D_n$  описывающих знания о ситуациях, извлечённых из разных текстов естественного языка. Если ситуации различны, но относятся к одной предметной области, мы можем их рассматривать как различные прецеденты данной предметной области. В таком случае знание, представленное этим набором прецедентов, формализуется дизъюнкцией  $(\tau_{D_1} \vee \dots \vee \tau_{D_n})$  соответствующих конъюнкций, формализующих знания о прецедентах. Любое бескванторное предложение сигнатуры  $\sigma_A$  может быть с точностью до эквивалентности представлено такой дизъюнкцией  $(\tau_{D_1} \vee \dots \vee \tau_{D_n})$ ; это показывает следующее утверждение.

**Предложение 1.** Для любого бескванторного предложения  $\varphi$  сигнатуры  $\sigma_A$  существует набор фрагментов атомарных диаграмм алгебраических систем  $D_1, \dots, D_n$  такой, что формулы  $\varphi$  и  $(\tau_{D_1} \vee \dots \vee \tau_{D_n})$  эквивалентны.

**Доказательство** непосредственно вытекает из известного факта, что для любой формулы логики высказываний существует эквивалентная ей формула, находящаяся в дизъюнктивной нормальной форме.

Таким образом, язык конечных фрагментов атомарных диаграмм является достаточно выразительным. Любое знание, извлечённое из текстов, сформулировать которое можно без использования кванторов всеобщности и существования, может быть выражено на языке конечных фрагментов атомарных диаграмм.

Более того, на языке конечных фрагментов атомарных диаграмм может быть выражено и любое знание, имеющее универсальную квантификацию – универсальные предложения, или  $\forall$ -предложения. Напомним, что предложение  $\varphi$  называется  $\forall$ -предложением если  $\varphi = \forall x_1 \dots \forall x_n \psi(x_1, \dots, x_n)$ , где  $\psi$  – бескванторная формула.  $\forall$ -предложение можно преобразовать в бескванторное предложение расширенной сигнатуры, заменив переменные, по которым идёт универсальная квантификация, на специальные новые константы, обозначающие «произвольный» объект.

В данной работе мы применим представленные выше теоретико-модельные методы для извлечения из текстов естественного языка знаний о смысле ключевых понятий предметной области, то есть извлечение из текстов частей определений понятий и интеграция извлечённых частей определений. Другими словами, речь идёт об извлечении из текстов естественного языка *онтологических знаний*, относящихся к заданной предметной области.

### **Полнота определений понятий относительно заданного контекста**

Для корректного извлечения знаний из текстов естественного языка необходимо решать проблему точного определения смысла, в котором в данном тексте используются понятия. Здесь возникает проблема полисемии или многозначности понятий, – в каком конкретно смысле понятие употребляется в данном тексте (в данном контексте).

Решение этой проблемы мы разделяем на две составляющие части, две подпроблемы.

1. Выделение набора различных определений данного понятия.

2. Определение по тексту / контексту того определения данного понятия, которое соответствует его употреблению в указанном месте.

При этом методы решения этих двух проблем взаимосвязаны.

Для выяснения точной семантики данного фрагмента текста естественного языка (например, точной семантики данного предложения естественного языка) и построения точного формального описания на языке логики предикатов или логики описаний необходимо иметь точные и полные определения входящих в текст понятий. Точность определений понятий обеспечивается тем, что мы формулируем их с помощью предложений логики предикатов или DL. Более сложной является проблема полноты определений понятий.

Очевидно, что с практической точки зрения мы почти никогда не можем дать полного определения данного понятия в абсолютном смысле. Тем не менее, для решения практических задач может быть достаточно *относительно полного* определения понятия, т. е. определения, полного относительно контекста рассмотрения этого понятия.

В этом параграфе мы дадим теоретико-модельную формализацию относительной полноты определений понятий и рассмотрим методы извлечения относительно полных определений понятий из текстов естественного языка.

Рассмотрим понятие  $P \in \sigma_A$  (для понятия  $c \in \sigma_A$  определения и рассуждения аналогичны). Пусть  $\varphi = \varphi(P)$  – некоторое (возможно неполное) определения понятия  $P$ ; запись  $\varphi(P)$  здесь означает, что символ  $P$  входит в предложение  $\varphi$ . Обозначим

$$\widehat{\varphi} = \{\psi \in S(\sigma_A) \mid \varphi(P) \vdash \psi\}.$$

$\widehat{\varphi}$  – это множество всех следствий  $\psi$  определения  $\varphi(P)$ ; заметим, что каждое такое следствие  $\psi$  является аналитическим предложением [19, 20].

Рассмотрим теперь несколько полисемичных определений  $\varphi_1, \dots, \varphi_n$  понятия  $P$ , то есть определений, описывающих разные смыслы, в которых понятие  $P$  может употребляться в разных контекстах. Обозначим  $D = \{\varphi_1, \dots, \varphi_n\}$ ,

$$\text{Ker}(D) = \{\psi \in S(\sigma_A) \mid \text{для любого } l \text{ выполнено } \varphi_l \vdash \psi\} = \bigcap_l \widehat{\varphi}_l \text{ и}$$

$$\text{Cov}(D) = \{\psi \in S(\sigma_A) \mid \text{для некоторого } l \text{ выполнено } \varphi_l \vdash \psi\} = \bigcup_l \widehat{\varphi}_l.$$

$\text{Ker}(D)$  назовём *ядром* множества определений  $D$ , а  $\text{Cov}(D)$  – *оболочкой* множества определений  $D$ . Заметим, что  $\text{Ker}(D) \subseteq \text{Cov}(D)$ .

$\text{Ker}(D)$  – это множество утверждений (свойств понятия  $P$ ), которые являются заведомо истинными вне зависимости от контекста и конкретного смысла понятия  $P$ , а  $\text{Cov}(D)$  – это множество утверждений (свойств понятия  $P$ ), которые могут быть истинными, если точный (конкретный) смысл понятия нам не известен. В общем случае множество  $\text{Cov}(D)$  может быть противоречивым.

В качестве примера рассмотрим три варианта  $\varphi_1, \varphi_2$  и  $\varphi_3$  смысла понятия «курица»: живая птица, замороженная тушка и курица-гриль. Во всех трёх случаях это объект материального мира, является видом птицы (соответственно, живой, замороженной и запечённой), имеет грудь, кости, ноги и крылья. Все эти свойства (записанные предложениями логики предикатов) входят во множество  $\text{Ker}(D)$ , где  $D = \{\varphi_1, \varphi_2, \varphi_3\}$ . В то же время множество  $\text{Cov}(D)$  содержит также утверждения, записанные в логике предикатов, что курица является живой, мороженной, запечённой, домашней птицей, полуфабрикатом, готовым блюдом. Каждое из этих утверждений может быть истинно, если контекст не может определить точный смысл данного понятия: например, во фразе «Мама купила курицу». С другой стороны, фраза «Мама купила живую курицу» точно определяет смысл  $\varphi_1$  данного понятия. Таким образом, в данном случае множество  $\text{Cov}(D)$ , очевидно, является противоречивым. Заметим также, что во втором случае фраза «Мама купила живую курицу», являющаяся контекстом употребления понятия «курица», однозначно задаёт определение этого понятия (по крайней мере, среди определений  $\varphi_1, \varphi_2$  и  $\varphi_3$ ). Далее мы дадим точную формулировку полноты определения понятия относительно фиксированного контекста.

Относительную полноту определения понятия мы будем рассматривать для двух вариантов фиксации контекста:

а) контекст определяется объемлющей онтологией;

б) контекст задаётся фрагментом атомарной диаграммы, построенным по тексту естественного языка, в котором данное понятие было использовано; при этом фиксируется специальное множество «интересующих» нас предложений.

Рассмотрим онтологию некоторой предметной области  $O$ . Будем считать, что онтология формально записана на языке логики предикатов первого порядка, т.е., что  $O \subseteq S(\sigma_A)$ . Пусть имеется набор полисемичных определений  $\varphi_1, \dots, \varphi_n$  понятия  $P$ , пусть  $D = \{\varphi_1, \dots, \varphi_n\}$ .

Определение  $\varphi_1$  назовём *полным относительно онтологии  $O$*  (среди определений  $\varphi_1, \dots, \varphi_n$ ), если  $O \cup \{\varphi_1\} \not\vdash$  и для любого  $l > 1$  выполнено  $O \cup \{\varphi_1\} \vdash \neg\varphi_l$ .

**Предложение 2.** Для онтологии  $O$  и множества определений  $D = \{\varphi_1, \dots, \varphi_n\}$  следующие условия эквивалентны:

- а)  $\varphi_1$  является полным относительно онтологии  $O$ ;
- б)  $O \cup \{\varphi_1\} \not\vdash$  и для любого  $l > 1$  найдётся предложение  $\psi \in \widehat{\varphi}_l$  такое, что  $O \cup \{\varphi_1, \psi\} \vdash$ ;
- в)  $O \cup \{\varphi_1\} \not\vdash$  и  $O \cup \{\varphi_1\} \cup \widehat{\varphi}_l \vdash$ .

Определение  $\varphi_1$  назовём *сильно полным относительно онтологии  $O$*  (среди определений  $\varphi_1, \dots, \varphi_n$ ), если для любого  $l > 1$  и для любого предложения  $\psi \in \widehat{\varphi}_l$  такого, что  $O \cup \{\varphi_1\} \not\vdash \psi$ , выполнено  $O \cup \{\varphi_1\} \vdash \neg\psi$ .

Определение  $\varphi_1$  назовём *абсолютно полным относительно онтологии  $O$*  (среди определений  $\varphi_1, \dots, \varphi_n$ ), если для любого предложения  $\psi \in S(\sigma_A)$  выполнено  $O \cup \{\varphi_1\} \vdash \psi$ , либо  $O \cup \{\varphi_1\} \vdash \neg\psi$ .

**Замечание 1.** Определение  $\varphi_1$  является абсолютно полным относительно онтологии  $O$  (среди определений  $\varphi_1, \dots, \varphi_n$ ) тогда и только тогда, когда множество предложений  $O \cup \{\varphi_1\}$  аксиоматизирует полную теорию сигнатуры  $\sigma_A$ .

Требования сильной полноты и абсолютной полноты определений понятий являются слишком «сильными» с практической точки зрения; они имеют в первую очередь теоретическое значение. В дальнейшем при разработке методов извлечения определений понятий из текстов естественного языка мы будем основываться на требовании относительной полноты определения. При этом с практической точки зрения полезной будет указанная выше эквивалентность пунктов (а) и (б) Предложения 2.

Рассмотрим теперь относительную полноту определения понятий в случае, когда контекст задаётся фрагментом атомарной диаграммы, построенным по тексту естественного языка, в котором данное понятие было использовано. Рассмотрим сразу общий случай, когда мы имеем несколько фрагментов атомарных диаграмм, извлечённых из текстов естественного языка, относящихся к одной предметной области.

Пусть  $F_1, \dots, F_m$  – фрагменты атомарных диаграмм алгебраических систем сигнатуры  $\sigma_A$ . Мы не требуем, чтобы  $F_1, \dots, F_m$  были фрагментами атомарной диаграммы некоторой единой алгебраической системы; мы считаем, что  $F_1, \dots, F_m$  представляют разные прецеденты одной и той же предметной области. Заметим, что если  $F_1$  и  $F_2$  являются фрагментами атомарной диаграммы одной алгебраической системы, то мы можем заменить их на один фрагмент атомарной диаграммы этой алгебраической системы – их объединение  $F = F_1 \cup F_2$ . Но поскольку  $F_1, \dots, F_m$  представляют прецеденты одной предметной области, на каждом из них истинна онтология этой предметной области. В частности, смысл принятий является одинаковым для всех фрагментов  $F_1, \dots, F_m$ .

Пусть имеется набор определений  $\varphi_1, \dots, \varphi_n$  понятия  $P$ , пусть  $D = \{\varphi_1, \dots, \varphi_n\}$ . Определение  $\varphi_1$  назовём *полным относительно набора прецедентов  $F = \{F_1, \dots, F_m\}$* , если:

- а) определение  $\varphi_1$  согласуется с набором прецедентов  $F$ , а именно, для любого  $i$  выполнено  $F_i \cup \{\varphi_1\} \not\vdash$ ;
- б) для любого  $l > 1$  найдётся  $i$  такое, что выполнено  $F_i \cup \{\varphi_1\} \vdash \neg\varphi_l$ .

**Замечание 2.** В общем случае два определения  $\varphi_1$  и  $\varphi_2$  (или даже более) из некоторого набора  $D = \{\varphi_1, \dots, \varphi_n\}$  могут быть полными относительно некоторого набора прецедентов  $F = \{F_1, \dots, F_m\}$ .

Это означает, что сами определения являются полными, но какое из них соответствует данной предметной области (данному множеству текстов естественного языка) по набору фрагментов атомарных диаграмм установить невозможно.

Заметим, что если бы вместо набора фрагментов атомарных диаграмм  $F = \{F_1, \dots, F_m\}$  мы рассматривали дизъюнкцию соответствующих конъюнкций

$$\bigvee_i \bigwedge_{\psi \in F_i} \psi,$$

то мы не смогли бы сформулировать предыдущее условие (б). Действительно, из  $F_i \cup \{\varphi_1\} \vdash \neg\varphi_l$  не следует, что  $((\bigvee_i \bigwedge_{\psi \in F_i} \psi) \& \varphi_1) \vdash \neg\varphi_l$ ; при этом утверждения  $F_i \cup \{\varphi_1\} \vdash \neg\varphi_l$  и  $((\bigwedge_{\psi \in F_i} \psi) \& \varphi_1) \vdash \neg\varphi_l$  равносильны. Аналогично, мы не смогли бы сформулировать следующее условие (с). Поэтому рассмотрение набора атомарных диаграмм как набора прецедентов предметной области (в рамках онтологии которой мы исследуем определения понятий) является принципиально важным.

**Предложение 3.** Определение  $\varphi_1$  из множества определений  $D = \{\varphi_1, \dots, \varphi_n\}$  является полным относительно набора прецедентов  $F = \{F_1, \dots, F_m\}$  тогда и только тогда, когда определение  $\varphi_1$  согласуется с набором прецедентов  $F$  и выполнено следующее условие:

с) для любого  $l > 1$  найдётся номер  $i$  и предложение  $\psi \in \widehat{\varphi}_l$  такие, что  $F_i \cup \{\varphi_1, \psi\} \vdash$ .

На Предложениях 2 и 3 основаны методы проверки относительной полноты определений понятий, а также разрабатываемые нами теоретико-модельные методы извлечения относительно полных определений понятий из текстов естественного языка.

Заметим, что, с множество предложений  $S(\sigma_A)$  сигнатуры  $\sigma_A$  бесконечно, в то время как для данного конечного фрагмента  $F$  атомарной диаграммы множество бескванторных предложений сигнатуры  $\sigma(F)$  этого фрагмента конечно, с точностью до эквивалентности. С точки зрения практических задач мы не можем обрабатывать бесконечное множество формул. Поэтому необходимо заменить бесконечные множества формул конечными; для этого мы зафиксируем специальное множество «интересующих» нас предложений.

Вместо бесконечного множества предложений  $S(\sigma_A)$  целесообразно рассматривать его конечное подмножество  $\Delta \subseteq S(\sigma_A)$  предложений, которые представляют интерес с точки зрения решаемых задач. В этом случае бесконечное множество предложений  $\widehat{\varphi}$  мы заменим на его конечное подмножество  $\widehat{\varphi} \cap \Delta$ . Таким образом, задача нахождения определения  $\varphi(P)$  понятия  $P$  трансформируется в задачу извлечения из текстов естественного языка предложений  $\psi \in \widehat{\varphi} \cap \Delta$ . При этом различные предложения  $\psi \in \widehat{\varphi} \cap \Delta$  могут извлекаться из разных текстов естественного языка.

При помощи описанных выше методов построения атомарных предложений по текстам естественного языка извлекаются фрагменты определений понятия. Затем атомарные предложения объединяются в единый фрагмент атомарной диаграммы. При этом проверяется непротиворечивость полученного фрагмента и его семантическая согласованность. Непротиворечивость проверяется при помощи трансляции бескванторных предложений логики предикатов в логику описаний ALCI, дальнейшей трансляции в OWL и применении средств логического вывода, как это будет подробно показано ниже. Семантическая согласованность проверяется при помощи эксперта в предметной области.

Далее проверяется относительная полнота полученных определений понятий. Для этого приведённые выше определения относительной полноты мы преобразуем, релятивизируя их к множеству интересующих нас предложений  $\Delta$ .

Рассмотрим набор определений  $\varphi_1, \dots, \varphi_n$  понятия  $P$ , пусть  $D = \{\varphi_1, \dots, \varphi_n\}$ . Для каждого  $l \leq n$  обозначим  $\Delta_l = \widehat{\varphi}_l \cap \Delta$ ; множество  $\Delta_l$  является конечным.

Для онтологии  $O$  релятивизируем пункт (б) Предложения 2 относительно множества предложений  $\Delta$ .

Определение  $\varphi_1$  будем считать полным относительно онтологии  $O$ , если  $O \cup \{\varphi_1\} \not\vdash$  и для любого  $l > 1$  найдётся предложение  $\psi_l \in \Delta_l$  такое, что  $O \cup \{\varphi_1, \psi_l\} \vdash$ . Из этого следует, что  $O \cup \{\varphi_1\} \vdash \neg\psi_l$ . Предложения  $\psi_1, \dots, \psi_n$  назовём предложениями, отделяющими смысл  $\varphi_1$  понятия  $P$  от других смыслов  $\varphi_1, \dots, \varphi_n$  этого понятия.

Таким образом, конъюнкция  $(\varphi_1 \& \neg\psi_1 \& \dots \& \neg\psi_n)$  по существу является полным явным определением понятия  $P$ , которое теперь можно включить в онтологии  $O$ .

Метод построения определения понятия  $P$ , полного относительно онтологии  $O$ , таким образом, заключается в нахождении предложений  $\psi_l \in \Delta_l$ , принадлежащих конечным множествам  $\Delta_l$ . Проверка противоречивости  $O \cup \{\varphi_1, \psi_l\} \vdash$  производится при помощи описанных

ниже методов трансляции таких предложений логики предикатов в логику описаний *ALCI*, дальнейшей трансляции в *OWL* и использовании ризонера *Hermit*.

Рассмотрим теперь набор прецедентов  $F = \{F_1, \dots, F_m\}$ . Пусть определение  $\varphi_1$  согласуется с набором прецедентов  $F$ , т.е., для любого  $i$  выполнено  $F_i \cup \{\varphi_1\} \neq \emptyset$ . Непротиворечивость проверяем указанным выше способом.

Релятивизируем пункт (с) Предложения 3 относительно множества предложений  $\Delta$ . Определение  $\varphi_1$  будем считать полным относительно набора прецедентов  $F = \{F_1, \dots, F_m\}$ , если для любого  $l > 1$  найдётся предложение  $\psi_l \in \Delta_l$  и номер  $i$  такие, что  $F_i \cup \{\varphi_1, \psi_l\} \vdash \psi_l$ . Это означает, в частности, что  $F_i \cup \{\varphi_1\} \vdash \neg\psi_l$ .

Как и для описанного выше случая онтологии  $O$ , предложения  $\psi_1, \dots, \psi_n$  отделяют смысл  $\varphi_1$  понятия  $P$  от других смыслов  $\varphi_1, \dots, \varphi_n$  этого понятия. При этом конъюнкция  $(\varphi_1 \& \neg\psi_1 \& \dots \& \neg\psi_n)$  является полным явным определением понятия  $P$  в контексте множества прецедентов  $F = \{F_1, \dots, F_m\}$ . Аналогично описанному выше, метод построения определения понятия  $P$ , полного относительно набора прецедентов  $F$ , заключается в нахождении указанного набора предложений  $\psi_l \in \Delta_l$ , принадлежащих конечным множествам  $\Delta_l$ .

### Трансляция фрагментов атомарных диаграмм в DL и OWL

В данном параграфе описан алгоритм порождения онтологии из фрагментов атомарных диаграмм. Разработанная программная система способна проверять готовую онтологию на непротиворечивость, а также обеспечивает порождение новых знаний посредством применения логического вывода, что позволяет пополнять онтологию новыми аксиомами.

Пользователь программной системы имеет возможность работать со знаниями, представленными в различных форматах: в виде бескванторных предложений логики предикатов, в виде формул логики описаний *ALCI* или *SROIQ*, а также на языке *OWL*.

Обрабатываемые системой файлы могут содержать:

- 1) фрагменты атомарных диаграмм, построенных по тексту на естественном языке;
- 2) выражения на языке логики описаний;
- 3) готовые онтологии в формате *OWL 2*.

Непосредственно порождение онтологии из фрагментов атомарных диаграмм осуществляется в три этапа:

- 1) трансляция фрагментов атомарной диаграммы модели в логику описаний;
- 2) переход из предложений на языке логики описаний в онтологию на языке *OWL*;
- 3) проверка полученной онтологии машиной логического вывода для поиска противоречий и пополнения новыми аксиомами.

#### Этап 1. Обработка атомарной диаграммы модели. Трансляция в DL

На первом этапе происходит трансляция бескванторных предложений логики предикатов в логику описаний. Рассматриваются предложения сигнатуры, содержащей только символы предикатов и констант (т. е. не содержащей функциональных символов).

Перед началом формализации выберем необходимый профиль *DL*. Для возможности дальнейшего расширения функционала, подходящим является логика описаний *SROIQ*, на которой основан язык *OWL 2*. Для целей данной работы нам достаточно использовать только фрагмент логики *SROIQ* – логику *ALCI*, не содержащую символа импликации, иерархии ролей и ограничений мощности.

При трансляции бескванторных предложений логики предикатов в логику описаний *ALCI* логические связки обрабатываются естественным образом; импликация выражается через дизъюнкцию и отрицание.

Нетривиальной является обработка многоместных предикатов. Для работы с такими предикатами мы используем технику представления многоместных предикатов через двухместные. А именно,  $n$ -местный предикат  $P(x_1, \dots, x_n)$  преобразуется в набор, содержащий  $n + 1$  двухместных предикатов. С помощью этого алгоритма мы решаем вопросы формализации ситуаций, в которых необходимо описать дополнительные уточняющие экземпляры отношений, а также представления отношений между тремя и более концептами.



$ALCI$ $Axiom ::= C \subseteq C \mid C(IName) \mid P(IName, IName)$ $C ::= CName \mid 1 \mid 0 \mid C \cap C \mid C \cup C \mid \neg C \mid \forall P.C \mid \exists P.C$ $P ::= PName \mid PName'$
--

Рис. 1. Профиль логики описаний – *ALCI*

## Этап 2. Трансляция из DL в OWL

При выполнении второго этапа вначале осуществим отображение синтаксиса логики описаний *ALCI* на синтаксис OWL (табл. 1).

Таблица 1

Соответствие синтаксиса OWL 2 и логики описаний *ALCI*

Синтаксис логики описаний <i>ALCI</i>	OWL синтаксис
$C \subseteq D$	SubClassOf(C D)
$C(a)$	ClassAssertion(C a)
$P(a, b)$	ObjectPropertyAssertion(P a b)
$C \cap D$	ObjectIntersectionOf(C D)
$C \cup D$	ObjectUnionOf(C D)
$\neg C$	ObjectComplementOf(C)
1	Owl:Thing
0	Owl:Nothing
$\exists P.C$	ObjectSomeValuesFrom(P C)
$\forall P.C$	ObjectAllValuesFrom(P C)
$P'$	ObjectInverseOf(P)

Далее для работы с выражениями языка логики описаний *ALCI* запускается соответствующий транслятор.

Первая составляющая транслятора – лексер. Задача лексера заключается в аналитическом разборе входной последовательности символов для получения на выходе последовательности символов, называемых лексемами языка. Лексемами языка называются те его понятия, которые определяются регулярной грамматикой, а также конечная совокупность служебных слов и символов языка.

Далее работу лексера принимает парсер (иначе – синтаксический анализатор). В ходе анализа исходный текст преобразуется в набор особых структур данных – деревьев, состоящих из соответствующих предикатов и констант. Такой способ представления хорошо подходит для дальнейшей обработки с последующей генерацией фрагмента онтологии.

## Этап 3. Использование машины логического вывода

После получения фрагмента онтологии ее необходимо обработать с помощью машины логического вывода для выявления наличия противоречий, а также для пополнения новыми аксиомами. Для осуществления данного этапа мы можем использовать любой ризонер,

имеющий возможность работать с логикой *SROIQ* – основой языка OWL 2. Исходя из произведенных исследований, а также результатов конкурса [21] для ситуаций, когда нам требуется высокая производительность, мы можем использовать Konclude [22], так как он занимает лидирующую позицию по скорости выполнения задач. Однако на начальных этапах разработки системы с точки зрения использования при реализации, удобной и также не уступающей в производительности является машина Hermit [23].

Приведем пример работы логического вывода для обработки онтологий в реальных ситуациях. Рассмотрим постановление правительства некоторых регионов, устанавливающее полный запрет на розничную продажу алкогольной продукции – слабоалкогольных напитков специального назначения – тонизирующих, в том числе энергетических. Мы имеем онтологию, содержащую следующие основные понятия: покупатель, продавец, нарушитель, напитки, содержащие тонизирующие компоненты – безалкогольные и слабоалкогольные напитки, а кроме этого включающую в себя информацию о продавцах данных товаров.

Дополним онтологию новыми аксиомами, после чего проверим ее ризонером (рис. 2, 3). Как результат его работы мы увидим расширение онтологии информацией о том, какие граждане являются нарушителями, а как следствие, увеличение значения некоторых метрик. А именно, появилось 10 новых аксиом.

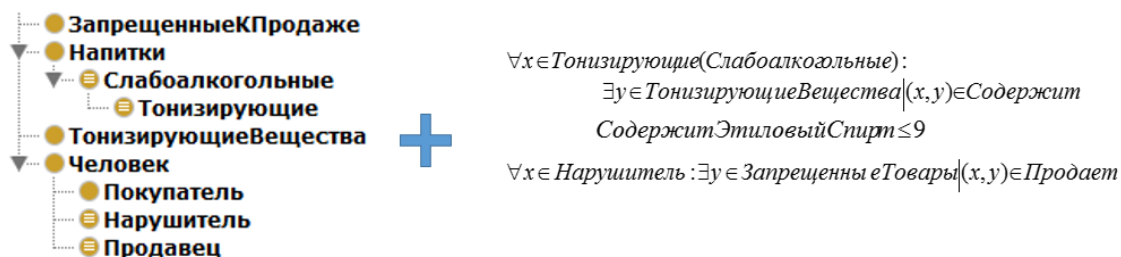


Рис. 2. Дополнение исходного фрагмента онтологии дополнительными аксиомами

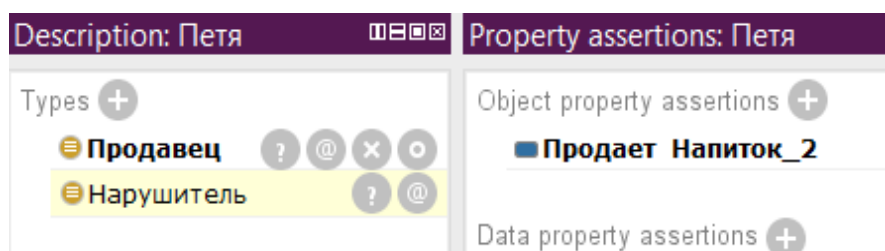


Рис. 3. Вывод информации о нарушителях после обработки ризонером

Благодаря интеграции информационной системы с машиной логического вывода загружаемая онтология проверяется на наличие противоречий с последующим информированием пользователя о наличии ошибок. Это также дает возможность увидеть неточности в формулировках документов.

### Программная реализация

Архитектуру системы и ее основные функции можно представить с помощью схем (рис. 4, 5).

Use-case диаграмма системы (рис. 6) демонстрирует реализованные на данный момент возможности пользователя.



Рис. 4. Схема взаимодействия клиента и WEB-контейнера

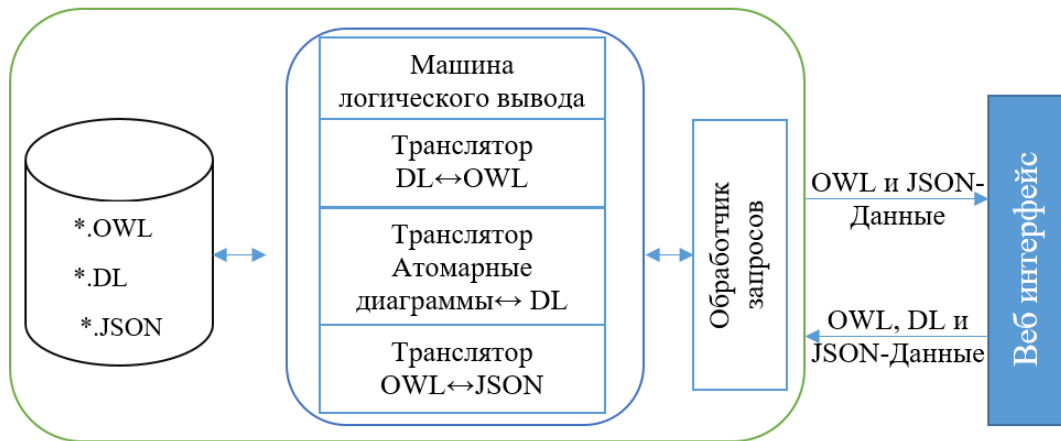


Рис. 5. Схема взаимодействия компонентов системы

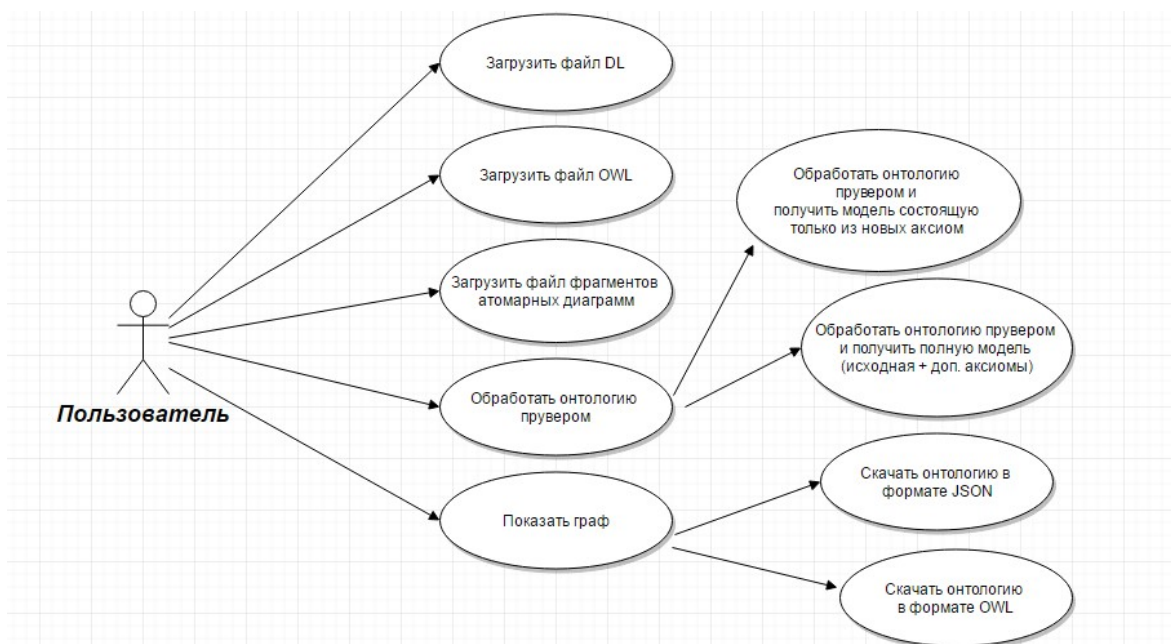


Рис. 6. Use-case диаграмма

В системе используются программные платформы *OWL API* [24] и *JENA* [25], которые позволяют работать не только с онтологиями, но и с машинами логического вывода, в частности с машиной Hermit.

### Заключение

В настоящей работе решается задача извлечения знаний о смысле понятий из текстов естественного языка. Исследована проблема полноты определений понятий относительно фиксированного контекста. Разработаны алгоритмы извлечения относительно полных определений понятий из текстов естественного языка.

Разработаны и программно реализованы алгоритмы отображения бескванторных предложений логики предикатов первого порядка, в частности, фрагментов атомарных диаграмм, на логики описаний.

Разработаны и реализованы в программной системе алгоритмы порождения онтологических знаний из фрагментов атомарных диаграмм алгебраических систем. Используемые программные платформы *OWL API* и *JENA* дают возможность работать не только с онтологиями, но и с машинами логического вывода. Благодаря этому программная система позволяет проверять онтологию на непротиворечивость, а также реализует порождение новых знаний посредством применения автоматических средств логического вывода. В частности, это даёт возможность автоматически пополнять онтологию новыми аксиомами.

Разработанная программная система может быть использована при создании систем поддержки принятия решений.

### Список литературы

1. Meilicke C., Stuckenschmidt H. A Reasoning-Based Support Tool for Ontology Mapping Evaluation. University of Mannheim, 2008.
2. Shen H., Hu J., Zhao J., Dong J. Ontology-based Modeling of Emergency Incidents and Crisis Management. Shanghai Jiao Tong University, Shanghai, 2010.
3. Flanagan K., Stevens R., Pocock M., Lee P., Wipat A. Ontology for genome comparison and genomic rearrangements. University of Newcastle upon Tyne, 2004.
4. Leuf B., Cunningham W. The Wiki Way: Quick Collaboration on the Web. Addison-Wesley Professional, 2001. 464 p.
5. Suchanek F.M., Kasneci G., Weikum G. YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia // Proceedings of the 16th International Conference on World Wide Web (Banff, Alberta, Canada, May 8–12, 2007). WWW '07. N. Y.: ACM Press, 2007. P. 697–706.
6. Shibaki Y., Nagata M., Yamamoto K. Constructing Large-Scale Person Ontology from Wikipedia // Proceedings of the 2nd Workshop on «Collaboratively Constructed Semantic Resources». Coling, 2010. P. 1–9.
7. Jie Bao, Paul R. Smart, Nigel R. Shadbolt, Dave Braines. A Controlled Natural Language Interface for Semantic Media Wiki Using the Rabbit Language // Workshop on Controlled Natural Language, 2009.
8. Hepp M., Bachlechner D., Siorpaes K. Harvesting Wiki Consensus – Using Wikipedia Entries as Ontology Elements // Proceedings of the First Workshop on Semantic Wikis – From Wiki to Semantics, Annual European Semantic Web Conference (ESWC 2006). 2006. P. 124–138.
9. Cui G. Y., Lu Q., Li W. J., Chen Y. R. Corpus Exploitation from Wikipedia for Ontology Construction // Proceedings of the Sixth International Language Resources and Evaluation (LREC 2008). Marrakech, 2008. P. 2125–2132.
10. Рабчевский Е. А. Автоматическое построение онтологий на основе лексико-синтаксических шаблонов для информационного поиска. // Тр. 11-й Всерос. науч. конф. «Электронные библиотеки: перспективные методы и технологии, электронные коллекции». Петрозаводск, 2009. С. 69–77.
11. Найханова Л. В. Методы и модели автоматического построения онтологий на основе генетического и автоматного программирования: Автореф. дис. ... д-ра техн. наук. Красноярск, 2008. 36 с.

12. *Мозжерина Е. С.* Автоматическое построение онтологии по коллекции текстовых документов. СПб., 2011.
13. *Тузовский А. Ф.* Работа с онтологической моделью организации на основе дескриптивной логики // Изв. Том. политехн. ун-та. 2006. Т. 309, № 7. С. 134–137.
14. *Gutierrez F., Dou D., Fickas S., Griffiths G.* Online Reasoning for Ontology-Based Error Detection in Text. University of Oregon, 2014.
15. *Hoehndorf R., Dumontier M.* A common layer of interoperability for biomedical ontologies based on OWL EL // Bioinformatics. 2011.
16. *Махасоева О. Г., Пальчунов Д. Е.* Автоматизированные методы построения атомарной диаграммы модели по тексту естественного языка // Вестн. Новосиб. гос. ун-та. Серия: Информационные технологии. 2014. Т. 12, вып. 2. С. 64–73.
17. *Махасоева О. Г., Пальчунов Д. Е.* Программная система построения атомарной диаграммы модели по тексту естественного языка. Свидетельство о государственной регистрации программы для ЭВМ № 2014619198, зарегистрировано 10.09.2014.
18. *Кейслер Г., Чэн Ч. Ч.* Теория моделей. М.: Мир, 1977. 615 с.
19. *Пальчунов Д. Е.* Моделирование мышления и формализация рефлексии I: Теоретико-модельная формализация онтологии и рефлексии // Философия науки. 2006. № 4 (31). С. 86–114.
20. *Palchunov D. E.* Virtual catalog: the ontology-based technology for information retrieval // Knowledge Processing and Data Analysis. Lecture Notes in Artificial Intelligence (LNAI). Springer-Verlag Berlin Heidelberg, 2011. Vol. 6581. P. 164–183.
21. ORE Live Competition. URL: [http://dl.kr.org/ore2015/vip.cs.man.ac.uk\\_8008/live.html](http://dl.kr.org/ore2015/vip.cs.man.ac.uk_8008/live.html).
22. *Steigmiller A., Liebig T., Glimm B.* Konclude: system description. Web Semantics: Science, Services and Agents on the World Wide Web, 27:78–85, 2014.
23. *Glimm B., Horrocks I., Motik B., Shearer R., Stoilos G.* A novel approach to ontology classification. J. of Web Semantics, 2011.
24. *Horridge M., Bechhofer S.* The OWL API: A Java API for OWL Ontologies // School of Computer Science. 2009.
25. *Carroll J. J., Dickinson I.* Jena: Implementing the Semantic Web Recommendations, 2004.

*Материал поступил в редколлегию 03.08.2016*

**I. A. Korsun<sup>1</sup>, D. E. Palchunov<sup>1,2</sup>**

<sup>1</sup> *Novosibirsk State University  
1 Pirogov Str., Novosibirsk, 630090, Russian Federation*

<sup>2</sup> *Institute of Mathematics SB RAS  
4 Academician Koptuyug Ave., Novosibirsk, 630090, Russian Federation*

*irina.korsun.nsu@gmail.com, palch@math.nsc.ru*

## **MODEL-THEORETIC METHODS OF EXTRACTION OF KNOWLEDGE ON THE MEANING OF CONCEPTS FROM THE NATURAL LANGUAGE TEXTS**

The paper is devoted to the development of model-theoretic methods of knowledge extraction from the natural language texts and, in particular, methods of extraction of concept definitions which are complete relative to the fixed context. The context may be determined by ambient ontology as well as by a set of precedents. The completeness of definitions is considered modulo a special set of "interesting" sentences. The extracted knowledge is represented in the form of fragments of atomic diagrams of algebraic systems. Algorithms of mapping some quantifier-free sentences of predicate logic onto Description Logics (DL) as well as of their further mapping onto OWL are developed and implemented. It gives a possibility to generate new knowledge based on existing knowledge that already contained in the ontology by using automated logical reasoning.

*Keywords:* domain ontology, model-theoretic methods, fragments of atomic diagrams, concept definitions, knowledge extraction, knowledge generation, automated logical reasoning.

## References

1. Meilicke C., Stuckenschmidt H. – A Reasoning-Based Support Tool for Ontology Mapping Evaluation // University of Mannheim, 2008.
2. Shen H., Hu J., Zhao J., Dong J. – Ontology-based Modeling of Emergency Incidents and Crisis Management // Department of Management Information Systems, Shanghai Jiao Tong University, Shanghai, 2010.
3. Flanagan K., Stevens R., Pocock M., Lee P., Wipat A. – Ontology for genome comparison and genomic rearrangements // School of Computing Science, University of Newcastle upon Tyne, 2004.
4. Leuf B., Cunningham W. The Wiki Way: Quick Collaboration on the Web. – Addison-Wesley Professional, 2001. – 464 p.
5. Suchanek F.M., Kasneci G., Weikum G. YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia // Proceedings of the 16th International Conference on World Wide Web (Banff, Alberta, Canada, May 8–12, 2007). WWW '07. – NY: ACM Press, 2007. – P. 697–706.
6. Shibaki Y., Nagata M., Yamamoto K. Constructing Large-Scale Person Ontology from Wikipedia // Proceedings of the 2nd Workshop on «Collaboratively Constructed Semantic Resources». – Coling, 2010. – P. 1–9.
7. Jie Bao, Paul R. Smart, Nigel R. Shadbolt, and Dave Braines. A Controlled Natural Language Interface for Semantic Media Wiki Using the Rabbit Language. In Workshop on Controlled Natural Language, 2009.
8. Hepp M., Bachlechner D., Siorpaes K. Harvesting Wiki Consensus – Using Wikipedia Entries as Ontology Elements // Proceedings of the First Workshop on Semantic Wikis – From Wiki to Semantics, Annual European Semantic Web Conference (ESWC 2006). – 2006. – P. 124–138.
9. Cui G.Y., Lu Q., Li W.J., Chen Y.R. Corpus Exploitation from Wikipedia for Ontology Construction // Proceedings of the Sixth International Language Resources and Evaluation (LREC 2008). – Marrakech, 2008. – P. 2125–2132.
10. Rabchevskii, E.A. Automatic ontology construction based on lexical and syntactic patterns for information retrieval // Proceedings of the 11th Scientific Conference "Digital Libraries: Advanced Methods and Technologies, Digital Collections" - Petrozavodsk, 2009. - P. 69- 77.
11. Naikhanova, L.V., Methods and models of automatic construction of ontologies based on genetic and automata programming, Extended Abstract of Doctoral (Techn.) Dissertation, Krasnoyarsk, 2008. - 36 p. (in Russian).
12. Mozzherina, E.S., Automatic construction of ontologies from collections of text documents // Saint-Petersburg State University. 2011. (in Russian).
13. Tuzovsky A.F. Work with an ontological model of organization on the basis of Description Logic // Bulletin of the Tomsk Polytechnic University. – 2006. – V. 309. – № 7. – P. 134–137
14. Gutierrez F., Dou D., Fickas S., and Griffiths G. – Online Reasoning for Ontology-Based Error Detection in Text // Computer and Information Science Department University of Oregon, 2014.
15. Hoehndorf, R., Dumontier, M. A common layer of interoperability for biomedical ontologies based on OWL EL // Bioinformatics, 2011.
16. Makhasoeva O.G., Palchunov D.E. Semi-automatic methods of a construction of the atomic diagrams from natural language texts // Vestnik NSU: Information Technologies - 2014. Vol. 12, No. 2, p.64–73. - ISSN 1818-7900 (in Russian).
17. Makhasoeva O.G., Palchunov D.E. Program system for the construction of the atomic diagram of a model from natural language texts. (in Russian). Certificate of the State Registration of the computer program. No. 2014619198, registered 10.09.2014.
18. Keisler G., Cheng C. C. Model theory. - M.: Mir, 1977. - 615 p.
19. Palchunov D. E. Modelirovanie myshleniya i formalizaciya refleksii. I: Teoretiko-model'naya formalizaciya ontologii i refleksii [Modeling of reasoning and formalization of reflec-

tion I: Model theoretical formalization of ontology and reflection]. *Filosofiya nauki*, 2006, no. 4 (31), p. 86–114. (in Russian).

20. Palchunov D.E. Virtual catalog: the ontology-based technology for information retrieval. // In: *Knowledge Processing and Data Analysis. Lecture Notes in Artificial Intelligence (LNAI)*, Springer-Verlag Berlin Heidelberg, Volume 6581, 2011, p. 164–183.

21. ORE Live Competition. URL: [http://dl.kr.org/ore2015/vip.cs.man.ac.uk\\_8008/live.html](http://dl.kr.org/ore2015/vip.cs.man.ac.uk_8008/live.html).

22. A. Steigmiller, T. Liebig, and B. Glimm. Konclude: system description. *Web Semantics: Science, Services and Agents on the World Wide Web*, 27:78–85, 2014.

23. Glimm, B., Horrocks, I., Motik, B., Shearer, R., Stoilos, G.: A novel approach to ontology classification. *J. of Web Semantics*, 2011.

24. Horridge M., Bechhofer S. *The OWL API: A Java API for OWL Ontologies* //School of Computer Science. 2009.

25. Jeremy J. Carroll, Dickinson I.: *Jena: Implementing the Semantic Web Recommendations*, 2004.