

## Проблема извлечения знаний в условиях динамически изменяющейся внешней среды на примере модели логического анализа текстов на естественном языке

В. А. КРАЙВАНОВА, Е. Н. КРЮЧКОВА

*Алтайский государственный технический университет, Барнаул, Россия*

e-mail: krayvanova@yandex.ru

Рассматривается математическая модель логического анализа фраз на естественном языке. Предлагаются механизмы адаптации и извлечения знаний для этой модели. Представлены результаты вычислительного эксперимента, демонстрирующие эффективность предложенных механизмов.

*Ключевые слова:* искусственный интеллект, естественный язык, извлечение знаний.

### Введение

Взаимодействие с компьютером на естественном языке (ЕЯ) не является универсальным, однако существует достаточно широкий круг задач, в которых пользователю проще сформулировать информацию в виде фразы на ЕЯ, чем в терминах формальных человеко-машинных интерфейсов. Примеры таких задач — интерфейсы к базам данных и системы управления так называемыми умными домами. Отметим, что эти задачи не требуют полноценного семантического анализа всех нюансов речи и зачастую достаточно буквального понимания фраз [1].

Наиболее распространенные подходы к анализу естественного языка основаны на синтаксическом и семантическом анализах и шаблонах [2]. Первый подход наиболее трудный, так как синтаксическое представление в очень малой степени отражает смысл запроса. Подход, основанный на семантике, кроме синтаксической информации, использует семантические словари. Применение этого подхода в связи с многозначностью ЕЯ приводит к комбинаторному взрыву. Кроме того, необходимые семантические словари — очень трудоемкая составляющая. Третий подход к анализу естественно-языковых запросов основан на шаблонах и наиболее прост с точки зрения программной реализации. Основной его недостаток состоит в необходимости предусмотреть все возможные способы выражений на естественном языке [2].

В настоящей работе рассматриваемый подход к пониманию ЕЯ основан на комбинации трех указанных подходов и заключается в следующем:

- все знания, используемые моделью, могут быть извлечены из синтаксической информации о фразе на ЕЯ;
- описание предметной области представляет собой семантический словарь упрощенной структуры;

— описание логических знаний системы основано на шаблонах и является множеством правил контекстной замены.

Как будет показано, данный подход позволяет значительно снизить трудоемкость составления базы знаний интеллектуальной системы, что дает возможность эффективно применять предлагаемую модель в человеко-машинных интерфейсах.

## 1. Модель логического анализа

Для моделирования понимания естественного языка необходимо задать следующие элементы:

- лексикон как множество известных в модели слов;
- формальное представление фразы ЕЯ на основе ее синтаксической структуры;
- представление команд системы управления;
- механизм анализа фраз.

В качестве элемента лексикона примем семантически неделимую единицу — слово. Пусть  $W$  — конечное множество слов предметной области, которые известны модели. Из множества  $W$  выделим множество  $N$  понятий естественного языка. Нечеткое отношение обобщения  $Gen : W \times N \rightarrow [0..1]$  определяет степень уверенности модели в том, что некоторое понятие  $p \in N$  обобщает слово  $w \in W$ . Нечеткое отношение синонимичности  $Syn : W \times W \rightarrow [0..1]$  определяет степень уверенности системы в том, что некоторое слово  $w_A \in W$  является синонимом к слову  $w_B \in W$ .

Формально фразу на естественном языке определим как функциональную форму, в которой в качестве функциональных символов выступают слова лексикона  $W$ , а в качестве типов аргументов — подчинительные синтаксические отношения. Пусть  $Rel$  — конечное множество типов аргументов модели. Функциональную форму  $\phi$  некоторой ЕЯ-фразы можно также представлять в виде дерева, вершины которого помечены словами  $w \in W$ , а ребра — типами аргументов  $r \in Rel$ . Корень дерева фразы  $\phi$  будем обозначать как  $Root(\phi)$ . Пусть  $u$  — узел помеченного дерева фразы. Тогда  $Word(u)$  — пометка узла  $u$ ,  $Relation(u)$  — пометка ребра, соединяющего узел  $u$  с родительским,  $Args(u)$  — множество дочерних узлов узла  $u$ ,  $Negation(u)$  — признак отрицания узла  $u$ .

Получить такое помеченное дерево фразы из предложения на ЕЯ можно, например, с помощью библиотек Dialing [3].

Команда системы управления представляется парой  $e = \langle \nu, Parameters \rangle$ , где  $\nu$  — функциональная форма, задающая определение команды на ЕЯ. Некоторые функциональные символы в  $\nu$  могут быть помечены не словами из  $W$ , а параметрами из множества  $Parameters$ .  $Parameters = \{p | p = \langle name_i, N_i \rangle, N_i \subseteq N\}$  — множество параметров команды, где  $name_i$  — имя параметра,  $N_i$  — множество понятий, обобщающих допустимые значения параметра.

В узлы дерева  $\nu$ , помеченные параметром  $name_i$ , может быть подставлено некоторое дерево  $\phi_i$  такое, что  $\exists n : Gen(Word(Root(\phi_i)), n) > 0, n \in N_i$ . Обозначим  $Value(p_i) = \phi_i$ .

Проверка наличия в некоторой фразе  $\phi$  команды  $e$  и вычисление параметров команды производятся с помощью операции унификации

$$\langle \gamma, Values \rangle = Unification(\phi, e), \quad (1)$$

результатом которой являются:

— уровень релевантности  $\gamma$ , определяющий степень уверенности системы в наличии команды;

— множество  $Values = \{ \langle p_i, \phi_i \rangle \}$ , где  $p_i$  — параметры команды  $e$ ,  $\phi_i$  — значения параметров, причем для каждого  $p_i$  должно существовать единственное значение  $\phi_i$ .

Задача логического анализа заключается в том, чтобы свести некоторую начальную фразу  $\phi$  к множеству фраз, определяющих команды и понятных объекту управления. Механизм преобразования фраз описывается конечным множеством правил контекстной замены  $Rules$ . В общем случае правило задается в виде  $r = \langle \nu \rightarrow \sigma, Parameters \rangle$ , где  $\nu$  и  $\sigma$  — соответственно гипотеза и следствие правила, представленные двумя суперпозициями, функциональные символы которых помечены как словами ЕЯ, так и параметрами из множества  $Parameters$ ;  $Parameters$  — множество параметров правила, аналогичное множеству параметров команды. Команда в этом случае является частным случаем правила — заключительным правилом, для которого следствие  $\sigma$  — пустое дерево.

Цепочка вывода из функциональной формы  $\phi$  представляет собой конечную последовательность функциональных форм  $\langle \phi_0, \phi_1, \dots, \phi_n \rangle$  таких, что  $\phi_0 = \phi$ , форма  $\phi_{i+1}$  непосредственно выводима из  $\phi_i$  для  $0 < i < n - 1$  с помощью некоторого правила  $r_i \in Rules$ , форма  $\phi_n$  получена из формы  $\phi_{n-1}$  с помощью заключительного правила  $r_{n-1} \in Rules$ .

На основе введенных определений задачу логического анализа функциональной формы  $\phi$  можно сформулировать следующим образом: найти множество  $E_\phi$  всех заключительных функциональных форм, выводимых из данной функциональной формы  $\phi$  с помощью цепочек вывода длиной не более  $Lenght$ . Очевидно, что при такой постановке задачи множество  $E_\phi$  конечно.

Если рассматривать данную модель с точки зрения практической применимости, возникает проблема, типичная для интеллектуальных систем: высокая трудоемкость построения базы знаний. Далее предложены механизмы пополнения логических знаний модели и оценка их эффективности.

## 2. Механизмы пополнения знаний

Важным свойством интеллектуальных моделей является адаптивность — способность сохранять свою адекватность в изменяющихся внешних условиях [4]. В современных условиях развития информационных технологий это особенно актуально, так как усложнение технических систем требует от субъекта управления все больших усилий [5].

Рассмотрим внешние для модели факторы, изменение которых ведет к снижению ее адекватности:

— функциональность объекта управления может активно изменяться и усложняться; это не должен приводить к усложнению модели управления;

— объектом управления могут пользоваться несколько субъектов управления, кроме того, язык каждого субъекта изменяется со временем.

Чтобы адаптироваться под изменившиеся внешние условия, модель должна быть дополнена механизмом получения информации об изменении и модификации имеющихся знаний. Возможными источниками информации об изменениях внешних условий для модели являются:

— модификация знаний экспертом по знаниям;

— фразы на ЕЯ, поступающие в модель для анализа.

В рассматриваемом контексте можно сформулировать следующие задачи:

- пополнение лексикона новыми словами и достраивание отношений обобщения и синонимичности в процессе логического анализа;
- достраивание множества правил при пополнении лексикона;
- формирование принципиально новых логических правил.

## 2.1. Расширение множества правил при пополнении лексикона

Лексикон модели является более простым и прозрачным уровнем знаний модели. Построение множества  $W$  и отношений  $Syn$  и  $Gen$  вручную — менее трудоемкая задача, чем построение множества правил  $Rules$ . Рассмотрим возможности модификации множества правил на основе изменений, внесенных в лексикон экспертом по знаниям.

Логические знания близких предметных областей имеют схожее внутреннее строение. Правила имеют ту же структуру, отличаясь лишь на лексическом уровне. Таким образом, при задании некоторой ассоциации между уже известным лексиконом  $W$  и лексиконом  $W_{new}$  новой предметной области можно расширить множество правил на новую предметную область [6].

Тема  $T \subseteq W$  — это поименованное подмножество слов лексикона, принадлежащих к одной предметной области. Обозначим как  $Name(T)$  — название темы, представляющее собой некоторое словосочетание. В отличие от отношения обобщения, которое определяет для данного слова понятие на более высоком уровне абстракции, принадлежность к теме фиксирует раздел предметной области, к которой относится слово. Так, понятие ЖИВОТНОЕ обобщает слова КОРОВА и СОБАКА, однако слово СОБАКА относится к теме КИНОЛОГИЯ, а слово КОРОВА — к теме РАЗВЕДЕНИЕ КРУПНОГО РОГАТОГО СКОТА.

Слово может принадлежать к нескольким темам или не принадлежать ни к одной из них. Рассмотрим механизм перестройки базы правил при пополнении лексикона новой темой.

Для решения поставленной задачи введем на лексиконе системы отношение ассоциации. Пусть  $T$  и  $G$  — некоторые темы. Два слова  $t \in T$  и  $g \in G$  связаны отношением ассоциации  $t = Association(g)$ , если слово  $t$  относится к словосочетанию  $name_T$  так же, как слово  $g$  относится к  $name_G$ . Отношение ассоциации симметрично. На множествах слов  $A_T \subseteq T$  и  $A_G \subseteq G$  установлено отношение ассоциации  $A_T \sim A_G$ , если

- $\forall t \in A_T \exists g \in A_G : t = Association(g)$ ;
- при замене в произвольной фразе  $\phi$  всех слов  $t \in A_T$  на соответствующие им слова  $g \in A_G$  получаемая при этом фраза  $\phi'$  остается осмысленной при условии, что  $\forall w \in T : \exists u \in \phi : Word(u) = w \rightarrow w \in A_T$ .

Отношение ассоциации строится экспертом в процессе описания лексикона предметной области. На рис. 1 приведен пример построения такого отношения. Отношением аналогии могут быть связаны только слова, отнесенные к некоторым темам.

Состояние знаний модели описывается парой  $\langle W, Rules \rangle$ , где  $W$  — лексикон модели,  $Rules$  — множество правил модели. Пусть  $\langle W_0, Rules_0 \rangle$  — начальное состояние знаний модели. Рассмотрим изменение множества правил при добавлении в лексикон новой темы  $T_N$ , связанной отношениями ассоциации с уже имеющимися в лексиконе темами. Для этого введем обозначения:  $A_N \subseteq T_N$  — множество слов и понятий новой темы, связанных отношением ассоциации с другими словами;  $AssThemes(A_N) = \{T | \exists A \subseteq T : A \sim A_N\}$  — множество тем, со словами из которых множество  $A_N$  связа-

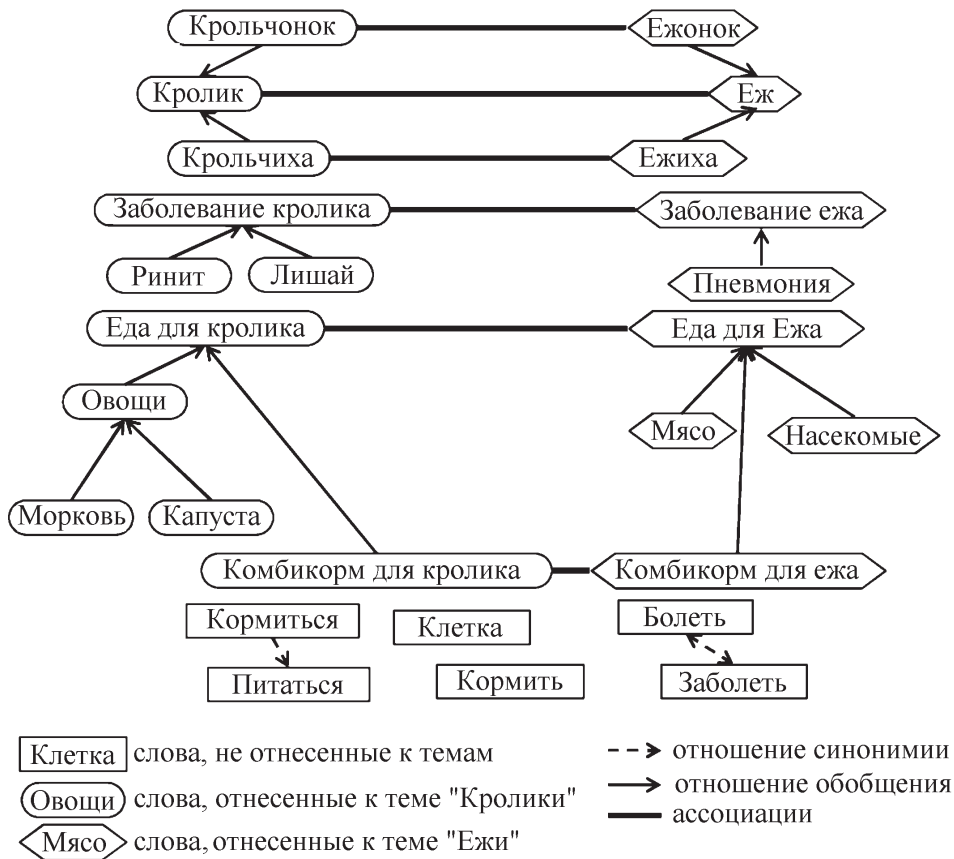


Рис. 1. Отношение ассоциации

но отношениями ассоциации;  $Nodes_T(r)$  — множество узлов правила  $r$ , помеченных словами из темы  $T$ . Тогда  $\langle W_{new}, Rules_{new} \rangle$  — новое состояние знаний модели, где  $W_{new} = W_0 \cup T_N$ ,  $Rules_{new} = Rules_0 \cup Rules_A$ .  $Rules_A$  — множество всех новых правил  $r_A = \langle \nu_A \rightarrow \sigma_A, Parameters_A \rangle$ . Эти правила получены из уже имеющихся правил  $r = \langle \nu \rightarrow \sigma, Parameters \rangle \in Rules_0$  таких, что  $\exists T \in AssThemes(A_N) : Nodes_A(r) \neq \emptyset$  и  $Nodes_A(r) = Nodes_T(r)$ . Преобразование осуществляется следующим образом:

- деревья  $\nu_A$  и  $\sigma_A$  получены из деревьев  $\nu$  и  $\sigma$  соответственно заменой пометок узлов  $u \in Nodes_a(r)$  на ассоциированные с ними слова  $Association(Word(u))$  из темы  $T_N$ ;
- параметр  $p_{Ai} \in Parameters_A$  совпадает с  $p_i \in Parameters$ , если  $p_i = \langle name, N \rangle$  и  $\nexists w \in N : w \in A$ ;
- если для некоторого параметра  $p_i = \langle name, N \rangle \in Parameters \exists w \in N : w \in A$ , то параметр  $p_{Ai} = \langle name, N_A \rangle \in Parameters_A$  получен из  $p_i$  следующей заменой множества допустимых значений:  $N_A = N_i \setminus A \cup \{w_A | w_A = Association(w), w \in N \cap A\}$ .

Предложенный механизм дает возможность автоматизировать рутинные операции пополнения знаний в пределах одной предметной области, однако не позволяет получать концептуально новые правила.

## 2.2. Достраивание лексикона

Достраивание отношений  $Gen$  и  $Syn$  основано на анализе позиции в дереве фразы, на которой встретилось незнакомое слово. Введем обозначения:

—  $AssGen_0$  — начальный уровень достоверности предположения о существовании отношения обобщения между словами;

—  $AssGen$  — уровень подтверждения достоверности предположения о существовании отношения обобщения;

—  $AssSyn_0$  — начальный уровень достоверности предположения о существовании отношения синонимичности между словами;

—  $AssSyn$  — уровень подтверждения достоверности предположения о существовании отношения синонимичности.

Для модифицирования отношения  $Gen$  модель строит предположение о том, что слово, подставляемое в параметр, соответствует данному параметру. Например, модель знает, что завявшие растения нужно поливать, а также, что растения — это фиалки и герань. Модель получает от пользователя фразу “Фикус в спальне совсем завял”. Слово ФИКУС встретилось на месте переменной, описываемой понятием РАСТЕНИЕ, поэтому можно предположить, что ФИКУС обобщается понятием РАСТЕНИЕ (рис. 2). Предположение будет подтверждаться, если слово ФИКУС снова встретится на месте понятия РАСТЕНИЕ, т. е. уверенность модели в наличии отношения обобщения между словами РАСТЕНИЕ и ФИКУС будет расти. Если, несмотря на сделанное предположение, в результате дальнейшего вывода не будет получено ни одной заключительной функциональной формы, то предположение считается неверным и уровень уверенности модели в наличии отношения снижается.

Пусть  $\langle \gamma, Values \rangle = Unification(\phi, r)$  — результат унификации гипотезы некоторого правила  $r$  и функциональной формы  $\phi$ , и  $\exists p_i : \tau(p_i) < 1$ , где  $\tau(p_i)$  — уровень достоверности унификации параметра. Тогда, если уровень релевантности результата  $\gamma$  больше некоторого порогового значения  $\gamma_0$ , то отношение  $Gen$  изменяется следующим образом:

$Gen_{old} \rightarrow Gen_{new}$  :

$Gen_{new}(w, n) = Gen_{old}(w, n), w \neq w_i$  или  $n \notin N_i$ ;

$Gen_{new}(w_i, n) = \begin{cases} AssGen_0, & Gen_{old}(w_i, n) = 0, \\ \min(Gen_{old}(w_i, n) + AssGen, 1), & Gen_{old}(w_i, n) > 0, \end{cases} \quad n \in N_i,$

где  $w_i = Word(Root(Value(p_i)))$ .

Отношение синонимичности достраивается на основе предположения, что два унифицируемых узла помечены синонимами. В примере на рис. 3 слово НАБРАТЬ встретилось на месте слова НАПЕЧАТАТЬ, и поэтому модель предполагает, что НАБРАТЬ —

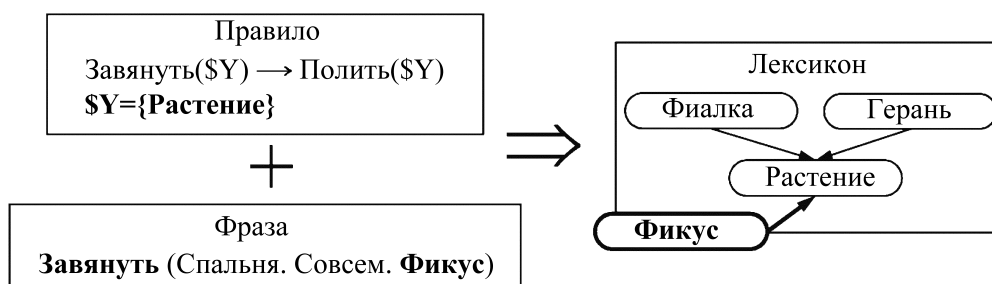
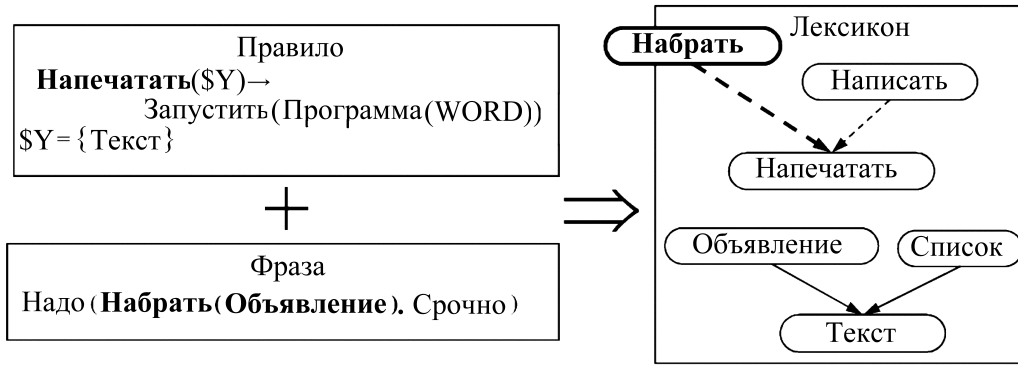


Рис. 2. Достраивание отношения  $Gen$

Рис. 3. Достраивание отношения *Syn*

синоним к НАПЕЧАТАТЬ. Рассмотрим унификацию узлов  $u \in \phi$  и  $u_r \in \nu$ . Здесь  $\phi$  — анализируемая фраза,  $\nu$  — гипотеза правила  $r$ . Введем обозначения:

$$\begin{aligned} w &= Word(u); \\ w_r &= Word(u_r); \\ \langle \gamma, Values \rangle &= Unification(\phi, r). \end{aligned}$$

Тогда, если  $\gamma > \gamma_0$ , то отношение *Syn* изменяется следующим образом:

$$\begin{aligned} Syn_{old} &\rightarrow Syn_{new} : \\ Syn_{new}(w_1, w_2) &= Syn_{old}(w_1, w_2), w_1 \neq w \text{ или } w_2 \neq w_r; \\ Syn_{new}(w, w_r) &= \begin{cases} AssSyn_0, & Syn_{old}(w, w_r) = 0, \\ \min(Syn_{old}(w, w_r) + AssSyn, 1), & Syn_{old}(w, w_r) > 0. \end{cases} \end{aligned}$$

Уверенность модели в сделанном предположении накапливается в процессе логического анализа. Если из результата  $\phi_{new}$  применения правила  $r$  к функциональной форме  $\phi$  не было получено ни одной заключительной функциональной формы, то все предположения, сделанные при формировании  $\phi_{new}$ , считаются неподтвержденными и отношения на лексиконе возвращаются в состояния  $Gen_{old}$  и  $Syn_{old}$ .

Построение многочисленных предположений может привести, во-первых, к ошибочным результатам, во-вторых, к неоправданному разрастанию графа вывода и комбинаторному взрыву. Поэтому ограничим минимальную степень достоверности результата унификации некоторым значением  $\gamma_0$  и минимальную степень достоверности цепочки вывода некоторым значением  $\gamma_{rez0}$ .

### 2.3. Извлечение правил

Правило контекстной замены может быть описано фразой на естественном языке. Рассмотрим в качестве объекта управления саму модель. Множество параметров для заключительных правил в данном случае состоит из двух элементов: \$HYPOTHESIS (гипотеза найденного правила) и \$CONSEQUENCE (следствие найденного правила). В указанные параметры могут быть подставлены любые функциональные формы. Результатом логического анализа в этом случае являются пары пропозициональных (т. е. не содержащих параметров) функциональных форм  $\langle \phi_{hyp}, \phi_{con} \rangle$ .

Рассмотрим механизм выделения параметров. Пусть по завершении вывода в параметры подставлены функциональные формы

$$Values = \{\langle \$HYPOTHESIS, \phi_{hyp} \rangle, \langle \$CONSEQUENCE, \phi_{con} \rangle\}.$$

Тогда новое правило будет иметь вид

$$r = \langle \phi'_{hyp} \rightarrow \phi'_{con}, Parameters \rangle,$$

где  $\phi'_{hyp}$  и  $\phi'_{con}$  получены из  $\phi_{hyp}$  и  $\phi_{con}$  заменой некоторых слов на параметры. Множество параметров строится следующим образом:

$$\begin{aligned} Parameters = \{ & p_i | p_i = \langle name_i, N_i \rangle : \\ & \exists u \in \phi_{hyp} : \exists w \in W : Gen(w, Word(u)) > 0, \\ & N_i = \{Word(u)\}, \\ & N_i \neq N_j \text{ при } i \neq j\}. \end{aligned}$$

Узел  $u$  из  $\phi_{hyp}$  или из  $\phi_{con}$  помечается параметром  $p_i = \langle name_i, N_i \rangle$ , если  $Word(u) \in N_i$ .

Описанный механизм позволяет пополнять знания модели как на начальном этапе построения базы знаний, так и в процессе эксплуатации. Поскольку извлечение знаний для модели требует более глубокого анализа текста, чем просто буквальное понимание, рассмотренный механизм имеет ограничения:

- правило должно описываться одним предложением;
- при формировании параметров не учитываются синонимы и местоимения.

### 3. Вычислительный эксперимент

В данном разделе представлена оценка эффективности предложенного механизма для извлечения знаний из необработанного текста (самостоятельная автономная работа) и адаптированного текста (гибридная человеко-машинная система).

Эффективность модели извлечения знаний проанализирована в серии вычислительных экспериментов. Исходные данные для экспериментов — тексты советов по различным тематикам, взятые из сети Интернет. Рассмотрим три предметные области:

- 1 — советы по ведению домашнего хозяйства;
- 2 — советы по уходу за комнатными растениями;
- 3 — советы начинающим программистам.

Тексты для каждой из предметной области составлены на основе нескольких сайтов. В первой предметной области правила сформулированы наиболее четко, в третьей — четко сформулированные правила фактически отсутствуют.

В связи с особенностями представления информации в Интернет исходные тексты прошли предварительную техническую подготовку (удаление нетекстовых объектов (рисунков, таблиц) и исправление орфографических и синтаксических ошибок), которая позволяет избежать ошибок на этапе синтаксического анализа. Авторские формулировки полностью сохранены. Далее под необработанным текстом подразумевается текст, прошедший предварительную техническую подготовку.



Роль человека в гибридной человеко-машинной системе извлечения знаний для предложенной модели логического анализа заключается в адаптации необработанного текста (устранение местоимений, упрощение формулировок, выделение неявных определений в отдельные предложения). При адаптации фразы, не содержащие правил, сохраняются в тексте, чтобы смоделировать поведение модели в процессе эксплуатации.

Для оценки результатов извлечения знаний введем следующие обозначения:

$R_{expert}$  — количество правил, обнаруженных экспертом. Учитываются только правила, полностью содержащиеся в одном предложении, например, “Чтобы сервировочные ножи блестели, их нужно почистить” — это правило, а конструкция “Для той же цели подойдет разбавленный настой чая” — нет;

$R_{auto}$  — количество правил, извлеченных автоматической системой;

$R_{relevant}$  — количество релевантных правил, извлеченных автоматической системой.

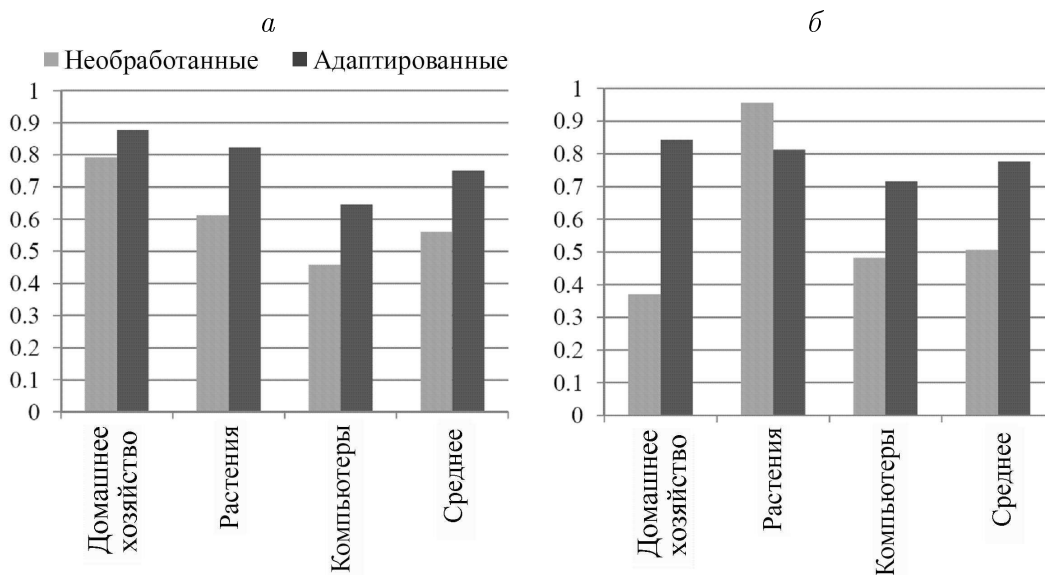


Рис. 4. Точность (а) и полнота (б) результатов анализа

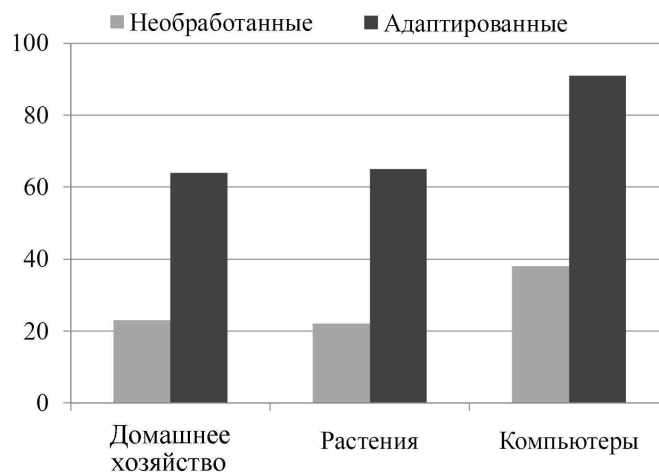


Рис. 5. Количество извлеченных правил

Качество извлеченных знаний оценивалось по двум параметрам:

—  $Precision = R_{relevant}/R_{auto}$  — точность;

—  $Recall = R_{relevant}/R_{expert}$  — полнота.

Как видно из рис. 4, поведение модели на необработанных текстах существенно зависит от типа текста, в то время как полнота результатов анализа адаптированных текстов от типа текста зависит незначительно и в среднем составляет 0.77, что намного выше аналогичного показателя для необработанных текстов. Точность анализа адаптированных текстов зависит от сложности текста, но при этом заметно выше точности для необработанных текстов.

Важным количественным показателем в эксперименте является  $R_{relevant}$  — число полученных релевантных правил. Использование гибридного подхода, т. е. предварительной адаптации текста, в сравнении с необработанными текстами позволяет извлечь примерно в 2.6 раза больше правил (рис. 5).

Таким образом, вычислительный эксперимент демонстрирует адекватность построенной модели. Используемые механизмы извлечения знаний позволяют модели адаптироваться к изменяющимся внешним условиям. Поскольку модель рассчитана на буквальное понимание отдельных фраз, самоприменение модели при извлечении правил из текста требует участия эксперта.

Предложенная математическая модель реализована в виде ядра программной архитектуры AL System [7].

## Список литературы

- [1] Крайванова В.А., Крючкова Е.Н. Технология AL SYSTEM в человеко-машинных интерфейсах на естественных языках // Сб. тр. VII Всерос. научно-практ. конф. студентов, аспирантов и молодых ученых “Молодежь и современные информационные технологии”. Ч. 1. Томск, 2009. С. 289–290.
- [2] Найханова Л.В., Евдокимова И.С. Методы и алгоритмы трансляции естественно-языковых запросов к базе данных в SQL-запросы. Улан-Удэ: Изд-во ВСГТУ, 2004. 248 с.
- [3] САЙТ рабочей группы “Автоматическая обработка текстов” [Электронный ресурс] / Режим доступа: <http://aot.ru/>
- [4] Венда В.Ф. Системы гибридного интеллекта: эволюция, психология, информатика. М.: Машиностроение, 1990. 448 с.
- [5] WANT R., PERING T., TENNENHOUSE D. Comparing autonomic and proactive computing // IBM Systems J. 2003. Vol. 42, No. 1. P. 129–135.
- [6] Крайванова В.А., Крючкова Е.Н. Методы организации базы знаний системы логического анализа текстов на естественном языке // Перспективные технологии искусственного интеллекта: Сб. тр. научно-практ. конф. Пенза, 2008. С. 202–207.
- [7] Крайванова В.А. Разработка адаптивных логических приложений AL System. Свидетельство о государственной регистрации программы для ЭВМ № 2009610579.

Поступила в редакцию 13 мая 2009 г.