

## Ассоциативная семантика ситуаций и сюжетов

**Аннотация.** В работе развивается тезис об ассоциативности восприятия текста, которая устанавливает связи между элементами высказываний по сходству атрибута, субъекта, отношения или путем выявления гомоморфизма графов. Предложен алгоритм сопоставления графов ситуаций, основывающийся на методе лучевого графа окрестности заданного радиуса. Семантическая модель текста собирается путем подстановки фрагментов из семантических моделей прецедентов. В эксперименте были использованы тексты логических задач для школьников младших и средних классов.

**Ключевые слова:** семантический анализ, ассоциативная семантика, семантическая модель, процессная онтология, прецедент, решатель логических задач.

### Введение

Системы поиска и аналитической обработки текстов, такие как Google Scholar, E-library, Scopus, Exactus, Mail.Ru, KM.Ru, УИС РОССИЯ и др., практически уже вошли в категорию гаджетов – предметов, облегчающих нам жизнь и без которых трудно обойтись. Они в разной степени успешно решают задачи семантического поиска, оценки качества текста, анализа связности с другими источниками и т.п., исходя из потребностей исследователей (лингвистов, филологов), издательств и авторов научно-технических публикаций.

Основу систем, анализирующих тексты, составляют методы выявления семантических отношений между объектами текста. Так, в русскоязычной системе Exactus [1] используется теория неоднородных семантических сетей [2] и на ее основе – реляционно-ситуационный метод [3], устанавливающий бинарные реляционно-ситуационные отношения между синтаксемами – синтактико-семантическими единицами текста [4] (порядка 100 видов синтаксем и несколько десятков видов отношений, по которым определяется сочетаемость синтаксем).

Концептуальные графы [5, 6] и нечеткие концептуальные графы [7, 8] широко используются для анализа текстов и представления знаний. При этом рассуждения моделируются с

помощью операций над графами, в основе которых лежит гомоморфизм графов [9, 10].

Гомоморфизм графа  $G$  графу  $H$  устанавливается путем сопоставления каждого узла первого графа узлу второго, соответствующему такому же или более частному понятию или отношению. Мера близости понятий и силы ассоциативной связи между ними вычисляют с помощью численных, логических, вероятностных, нечетких и лингвистических шкал. На практике поиск гомоморфизма в базе знаний для интерпретации текста – это весьма ресурсоемкая операция, поэтому необходимо максимально разгрузить ее от переборочных.

Стратегии и алгоритмы информированного (эвристического) поиска в графах хорошо представлены в работах [11, 12]. Широко применяется стратегия «лучевого» поиска, при которой выбор луча для дальнейшего пути осуществляется с помощью предпочитающих эвристик. Наиболее популярна стратегия «жадного» поиска по первому наилучшему совпадению узла. Здесь с помощью эвристической оценивающей функции выбирается «отправная» точка. Далее процесс поиска разворачивается в виде дерева состояний с различными оценками и способами возврата (backtraking) при неудачах и оптимизации поиска.

Для поддержки принятия решений, в первую очередь, нужно найти в базе знаний и опыта

аналогичные сюжеты или сценарии – модели реальных сюжетов. В сфере производства товаров сценариями являются технологические процессы, в сфере производства услуг обычно используется термин «регламент». Требуется определить сходство двух ситуаций по понятиям, свойствам, составам, внутренним и внешним связям. Очевидно, чем объемнее найденные прецеденты – факторные агрегаты, тем качественнее решение и меньше степень влияния субъективизма. Поиск аналогий обычно осуществляется волновым (вейвлет) методом или путем лучевого поиска с учетом предпочтений.

Целью данной работы является адаптация семантических моделей (СеМ) текстов к задачам поддержки принятия решений путем приближения грамматики СеМ к традиционным в информационных системах схемам BPMN управления бизнес-процессами. А также за счет алгоритма сборки СеМ текста путем сопоставления с семантическими моделями прецедентов с помощью метода «лучевой граф окрестности» и подстановки в формируемую СеМ элементов, которые не противоречат друг другу по принадлежности к понятиям из родовидовых деревьев процессно-ориентированной онтологии предметной области.

## 1. Процессуально ориентированная онтология предметной области

Процессуально ориентированная онтология (ПОО) характерна тем, что для удобства поддержки принятия решений делает акцент на процесс как систему взаимоотношений между субъектами-участниками.

Метаонтология ПОО определяет следующие категории понятий: свойство, предмет, процесс, отношение, а также составные понятия - ситуация, сюжет, сценарий [13].

**Концепт-свойство** имеет вид:

$\langle \text{Свойство} \rangle ::= \langle \text{Имя} \rangle, [\langle \text{Комментарий} \rangle,] \langle \text{Тип значения} \rangle, [\langle \text{Способ} \rangle,]$

где: *Способ* – способ измерения свойства.

**Предметы** имеют состав (детали) и схему соединений, т.е. описываются геометрическими моделями: кинематическая схема, чертеж, карта, изображение, 3D - геометрическая модель, экранная форма:

$\langle \text{Предмет} \rangle ::= \langle \text{Имя} \rangle, \langle \text{Комментарий} \rangle, \langle \text{Список атрибутов} \rangle, [\langle \text{Список компонент} \rangle,$

$[\langle \text{Список отношений между компонентами} \rangle,] [\langle \text{Геометрическая модель} \rangle],$   
где  $\langle \text{Атрибут} \rangle ::= \langle \text{Имя} \rangle, \langle \text{Экземпляр-свойство} \rangle, [\langle \text{Диапазон изменения значений} \rangle, \langle \text{Ожидаемое значение} \rangle], [\langle \text{Единица измерения} \rangle,] [\langle \text{Экземпляр-метод} \rangle].$

**Процессы** имеют состав и алгоритм, т.е. их описание содержит вычислительные модели. Концепт процесса имеет следующий вид:

$\langle \text{Процесс} \rangle ::= \langle \text{Имя} \rangle, \langle \text{Комментарий} \rangle, \langle \text{Список атрибутов} \rangle, [\langle \text{Длительность} \rangle], [\langle \text{Вычислительная модель} \rangle], \langle \text{Список входных концептов} \rangle, \langle \text{Список выходных концептов} \rangle, [\langle \text{Состав подпроцессов} \rangle, \langle \text{Граф-схема} \rangle].$

Процесс порождает множество связей между объектами, поэтому характерной особенностью списка атрибутов процесса является то, что он, помимо общих характеристик, содержит атрибуты, описывающие **роли участников действия**:

- агент (исполнитель);
- бенефициант – заказчик, в чьих интересах выполняется действие, работа;
- реципиент – приемник действия (например, "Вася дает яблоко Кате", реципиент – Катя);
- предмет воздействия - исходный/результатирующий (в приведенном примере – яблоко);
- сцена действия;
- инструмент;
- коагент (соисполнитель);
- эффект и т.д.

$\langle \text{Отношение} \rangle ::= \langle \text{Имя} \rangle, \langle \text{Тип} \rangle, [\langle \text{Комментарий} \rangle,] \langle \text{Список атрибутов} \rangle, [\langle \text{Список объектов} \rangle, \langle \text{Тип значения} \rangle, \langle \text{Ожидаемое значение} \rangle, [\langle \text{Единица измерения} \rangle,] \langle \text{Экземпляр-метод} \rangle.$

Здесь *Экземпляр-метод* – это конкретный метод исчисления значения отношения.

В ПОО выделены следующие наиболее актуальные категории отношений, значения которых принадлежат разным классам:

- 1) *принадлежности* предмету, процессу, классу, множеству;
- 2) *родства* – степень удаленности друг от друга двух субъектов в родовидовом дереве;
- 3) *положения* субъекта в пространстве предмета, сцены относительно другого субъекта;
- 4) *следования* во времени относительно события, процесса;
- 5) *каузальные* – причинно-следственные связи;
- 6) *сравнения/сопоставления* значений свойств или математических моделей, в частности, графов;

7) *коммуникации* (толерантности) субъекта к другому субъекту или классу;

8) *сопутствия* – совместного существования разных сущностей;

9) *процессуальные* – ролевые отношения между участниками процесса;

10) *системные* – отношения на множестве субъектов, взаимодействующих между собой и оказывающих прямое или косвенное влияние друг на друга, например, в экологической системе.

**Ситуация** – это совокупность состояний предметов, процессов, отношений между ними, где **состояние** – это совокупность значений параметров предметов, процессов и отношений в некоторый момент или отрезок времени.

**Сюжет** – это упорядоченная во времени последовательность ситуаций и **Сценарий** (модель сюжета) – упорядоченная последовательность действий (орграф, диаграмма).

Сюжет описывает динамику бытовых, производственных и прочих отношений между субъектами.

Представим онтологическую модель предметной области в виде графа  $GJ$  типа "джунгли" (jungle graph), в котором множество узлов, отображающих сущности предметной области и принадлежащих различным деревьям, связаны различными отношениями:

$$GJ = G(\mathcal{J}, \mathcal{R}),$$

где:  $\mathcal{J}$  - семейство родовидовых деревьев сущностей;  $\mathcal{R}$  - множество отношений между сущностями.

Заметим, что компонента  $\mathcal{R}$  отличает граф "джунгли" от графа "лес". Семейство деревьев сущностей:

$$\mathcal{J} = \langle GT^{ATR}, \{GT^{THING}\}, \{GT^{PROC}\}, GT^R, \{GT^{STORY}\}, \{GT^{SIT}\} \rangle,$$

где:  $GT^{ATR}$  – дерево свойств;  $\{GT^{THING}\}$  – множество деревьев предметов;  $\{GT^{PROC}\}$  - множество деревьев процессов;  $GT^R$  – дерево сущностей отношений;  $\{GT^{STORY}\}$  – множество деревьев сюжетов (story plot);  $\{GT^{SIT}\}$  – множество деревьев ситуаций.

Множество отношений между сущностями (концептами):

$$\mathcal{R} = \langle \{GT^{STR}\}, \{G^{ACT}\}, \{GT^{ROLE}\}, \{G^{TIME}\}, \{G^{QUASY}\}, \{OR\} \rangle,$$

где:  $\{GT^{STR}\}$  – множество деревьев структур сущностей (structure);  $\{G^{ACT}\}$  – множество

деревьев схем выполнения структурированных процессов;  $\{GT^{ROLE}\}$  – множество деревьев ролевых отношений между участниками процессов;  $\{G^{TIME}\}$  – множество деревьев графов временных отношений в сюжетах;  $\{G^{QUASY}\}$  – множество деревьев графов ситуаций (quasi random);  $\{OR\}$  – множество других отношений (other relation), сущности которых описаны в дереве  $GT^R$  отношений.

Собственно, граф  $GJ$  онтологической модели – это структурированное описание предметной области. На нем удобно строить процедуры задачно-ориентированного анализа и распознавания, автоматического синтеза планов решения задач.

## 2. Граф сюжета и лучевой граф окрестности заданного радиуса

**Определение 1.** Граф сюжета

$$G^u = (V^{u-1}, E^{u-1}, A, R),$$

где:  $u$  – порядок графа – уровень обобщенности информации;  $V^{u-1}$  – множество вершин  $v^{u-1}$ , отображающие подобъекты уровня  $u-1$ ;  $E^{u-1}$  – множество ребер  $e^{u-1} \in E^{u-1} = (v_i^{u-1}, R_{ij}, v_j^{u-1})$ , отображающее отношения между объектами;  $A = (AH, VH)$  – гиперграф атрибутов, указывающий принадлежность атрибутов из множества  $A$  некоторой вершине  $v \in V$ ;  $R$  – множество отношений между вершинами.

Путем смены уровня вершины или ребра графа  $G^u$  осуществляется развертка/свертка описания сюжета. Правила соединения вершин графа сюжета определяются **плекс-грамматикой**.

Для формализации семантической модели текста, описывающего ситуацию (СеМС), воспользуемся плекс-грамматикой [14, 15], где символы грамматических конструкций имеют не две (слева, справа), а  $N$  "точек примыкания".

Выделим 4 сорта элементов (деталей), из которых составляются модели ситуаций, и представим их в виде символов [16] (Рис. 1).

Процессы в точках примыкания соединяются с элементами, отвечающими на вопросы: (кто), {из чего}, {чем}, (где), (как), (что), (сколько), (кому), (для чего), (когда). Фигурные скобки обозначают, что несколько символов одного сорта может быть связано с данным символом. Например, для процесса может потребоваться несколько наименований материалов (из чего), инструментов (чем).

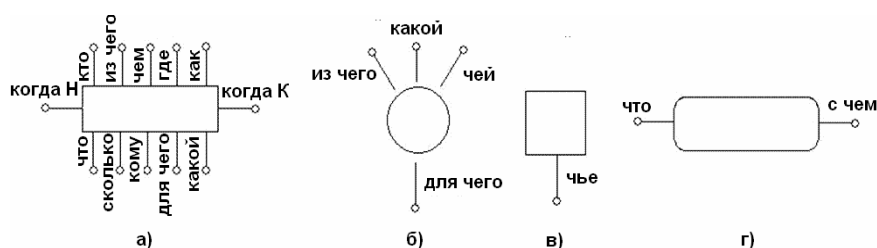


Рис. 1. Символы деталей семантической модели ситуации  
 а - процесс; б - предмет; в - свойство; г - отношение

Предметы в точках примыкания связываются с другими элементами в соответствии с вопросами: *{из чего}*, *{какой}*, *{чей}*, *{для чего}*. Предмет характеризуется свойствами, различными по физической природе (какой) и имеет несколько предназначений (для чего) – процессов, в которых он может быть использован.

Свойство связывается с одной сущностью, которую оно характеризует. Отношение связывает формулой две или более сущности.

Плекс-грамматику, порождающую множество семантических моделей ситуаций, представим как шестерку  $\langle V_T, V_N, R_S, R_O, Q, q_0 \rangle$ , где  $V_T$  – множество основных символов;  $V_N$  – множество производных (вспомогательных) символов;  $R_S$  – множество системных правил подстановки;  $R_O$  – множество предметных правил подстановки;  $Q$  – множество идентификаторов – меток точек примыкания;  $q_0$  – специальной (пустой) идентификатор. Основные элементы изображены на Рис. 1.

Идентификаторы точек примыкания для одного и того же элемента должны быть различными. В какой-то ПрО это множество может быть сокращено или расширено. Здесь множество идентификаторов точек примыкания приближено к вопросительным словам русского языка. Идентификатор  $q_0$  служит для заполнения места, не связанного ни с какой сущностью, т.е. является признаком неполноты СеМС.

Правила подстановки – это правила построения конструкций из символов и ранее определенных конструкций, т.е. из основных элементов и производных:

- свойства присоединяются к процессам (какой) и предметам (какой);
- предметы присоединяются к процессам в точках примыкания *кто*, *из чего*, *чем*, *где*, *что*, *кому*;

- отношения (зависимости) связывают формулами свойства или предметы, или процессы.

Минимальной конфигурацией ситуации можно считать  $Sit \rightarrow \langle p \rangle \langle s \rangle$ , где  $p$  – процесс;  $s$  – предмет – субъект, выполняющий процесс, например, *Имеется стол*.

**Системные правила подстановки** – это правила, универсальные для всех предметных областей, в то время как предметные правила задает эксперт или конечный пользователь (менеджер). Системные правила основаны на падежной грамматике Ч.Филмора [17], определяющей семантические валентности слов, т.е. их роли в предложении (глубинные падежи). Для этого используем правила плекс-грамматики. Они рекурсивно собирают модель ситуации из подходящих деталей:

- 1)  $\langle Situation \rangle \rightarrow \langle p \rangle \langle s \rangle$  (*кто* – для чего),
- 2)  $\langle Situation \rangle \rightarrow \langle a \rangle \langle Situation \rangle$  (*где* – какой) /  $\langle a \rangle \langle Situation \rangle$  (*чье* – как),
- 3)  $\langle Situation \rangle \rightarrow \langle s \rangle \langle Situation \rangle$  (*для чего* – что) /  $\langle s \rangle \langle Situation \rangle$  (*для чего* – из чего) /  $\langle s \rangle \langle Situation \rangle$  (*для чего* – чем) /  $\langle s \rangle \langle Situation \rangle$  (*для чего* – где) /  $\langle s \rangle \langle Situation \rangle$  (*для чего* – кому) /  $\langle s \rangle \langle Situation \rangle$  (*для чего* – из чего) /  $\langle s \rangle \langle Situation \rangle$  (*для чего* – чей),
- 4)  $\langle Situation \rangle \rightarrow \langle r \rangle \langle Situation \rangle$  (*что*, *с чем* –  $чье_{ij}$ ,  $чье_{kl}$ ) /  $\langle r \rangle \langle Situation \rangle$  (*что*, *с чем* –  $s_j, s_l, j \neq l$ ) /  $\langle r \rangle \langle Situation \rangle$  (*что*, *с чем* –  $p_j, p_l, j \neq l$ ) /  $\langle r \rangle \langle Situation \rangle$  (*что*, *с чем* –  $s_j, p_l$ ),
- 5)  $\langle Situation \rangle \rightarrow \langle p \rangle \langle r \rangle \langle Situation \rangle$  (*когда*  $H$ , *когда*  $K$  – что, *с чем* –  $p_j, p_s$ ),

где  $p$  – процесс,  $s$  – предмет;  $a$  – свойство;  $r$  – отношение.

Первое правило соединяет процесс и предмет через их точки примыкания "кто" (делает), "для чего" (предназначен) соответственно.

Второе правило добавляет к ситуации свойство, соединяя точку примыкания "чье" свойство с точкой примыкания "какой" предмета или процесса.

Третье правило присоединяет к ситуации новый предмет с помощью его точки примыкания "для чего" и точки примыкания процесса "что" или "из чего", или "чем", или "где", или "кому", или точек примыкания другого предмета "из чего", или "чей". Отношение как метод вычисления значений связывает значение неизвестного параметра одного объекта относительно значения однотипного параметра другого объекта. Напомним, что формулой любого отношения может быть константа.

Четвертое правило показывает, как отношение связывает  $i$ -тое свойство  $j$ -того объекта с  $k$ -тым свойством  $l$ -того объекта или связывает объекты в целом в случае, если это отношения родства, толерантности, сопутствия, сопоставления (сходства) и некоторые другие.

Пятое правило присоединяет к ситуации новый процесс. Правило показывает, что вначале нужно связать его во времени с какими-то имеющимися процессами с помощью отношения следования. Другие связи нового процесса построятся рекурсивно с помощью правил (2), (3), (4).

**Предметные правила подстановки** возникают следующим образом. В онтологии Про для всех точек соединения каждого процесса имеются множества взаимозаменяемых элементов (это знания экспертов). Тогда везде, где вместо идентификатора записан  $q_0$ , можно подставить элемент из соответствующего множества взаимозаменяемых, например, *экскаватор*  $\rightarrow$  *лопата*.

Пусть имеется некоторая обобщенная (производственная) ситуация  $SituationO$ . Обозначим  $\underline{S}_{q_i(p)}$  нечеткое множество взаимозаменяемых участников процесса  $p$  в точке примыкания  $q_i \in Q$ . Нечеткое множество  $\underline{S}$  – это множество пар  $\langle x, \mu(x) \rangle$ , где  $\mu(x):S \rightarrow M$ ,  $M = \{0, 1\}$  – функция, которая отображает  $x$  на единичный отрезок  $M$ , определяя степень принадлежности элемента  $x$  множеству  $S$ . Предметное правило подстановки имеет вид:

$$SituationO \wedge (q_i(p) = q_0) \wedge (\underline{S}_{q_i(p)} \neq \emptyset) \Rightarrow q_i(p) = \langle x, \mu(x) \rangle,$$

т.е. элемент  $x$  может быть подставлен вместо отсутствующего элемента со степенью адекватности  $\mu(x)$ .

На практике функция  $\mu(x)$  заменяется коэффициентом приоритета, задаваемым экспертом, а количество порождаемых моделей ситуаций ограничивается возможностями компьютера, либо система сама ранжирует СеМС и представляет пользователю несколько наилучших по критериям прагматики. Поскольку эти подстановки генерируются как цепочка следствий, то переборы в пределах множеств взаимозаменяемости постепенно сужаются.

Лучевой граф, введенный в работе [18], вершины и дуги которого нагружены пространственными характеристиками элементов *изображения объекта* и отношений между ними, удобен как инструмент ассоциативного поиска при формировании целостного образа ситуации из подобъектов. Для целей данной работы введем

**Определение 2.** Лучевой граф  $GC$  окрестности радиуса  $r$  с корневой вершиной  $v_c$  – это граф

$$GC = (v_c, r, VC, EC, A, R), v_c \in VC \subseteq V, EC \subseteq E,$$

где  $r$  – максимальная длина лучевой цепочки дуг, и для каждой вершины справедливо условие:

$$\forall i = [1 \dots N], \forall j = [1 \dots M_i], \forall k = [1 \dots M_i] j \neq k, \\ (Type R_j = Type R_k) \Rightarrow Volume R_j \neq Volume R_k,$$

где  $N$  – количество вершин графа;  $M_i$  – количество дуг, инцидентных  $i$ -той вершине, т.е. дуги, исходящие из одной вершины, различаются между собой либо типами отношений, либо их значениями.

Отметим, что, в отличие от лучевого поиска, задающего направление, где следует искать, лучевой граф окрестности (ЛГО) «узнает» окружающую окрестность целиком. Алгоритм, основанный на методе ЛГО является частным случаем «жадного» поиска. Он предъявляет повышенные требования к качеству «отправной точки». При этом глубина предварительного просмотра ограничена радиусом анализируемой окрестности.

Будем называть *общей гомоморфной частью*  $\tau(G_1')$  графов  $G_1$  и  $G_2$  часть  $G_2'$ , определенную на множестве  $V_2'$  и состоящую из всех

ребер  $\tau(E_1) = E_2 = (g_{2i}, g_{2j}), g_{2i}, g_{2j} \in V_2' \subseteq V_2$ , для которых существуют соответствующие ребра

$$E_1 = (g_{1i}, g_{1j}) = \tau^{-1}(E_2) = (\tau^{-1}(g_{2i}), \tau^{-1}(g_{2j})), g_{1i}, g_{1j} \in V_1.$$

В графе  $G_1$ , при этом, однозначное отображение  $\tau$  существует тогда и только тогда, когда вершины  $g_{2i}, g_{2j}$  и соответствующие им прообразы  $g_{1i}, g_{1j}$  совпадают в  $n$ -мерном пространстве  $A'$  анализируемых признаков с точностью, заданной  $A''$ .

Алгоритм сопоставления графа  $G_1$  известному прецедента с графом  $G_2$  анализируемого текста заключается в следующем:

1. Поиск корневой вершины в графе  $G_2$ . Сравнивается корневая вершина  $v_0 \in G_1$  со всеми вершинами графа  $G_2$ , принадлежащими тем же классам и подклассам процессно-ориентированной онтологии (ПОО) по значениям атрибутов, заданных прилагательными, числительными, причастиями или иными средствами языка.

2. При совпадении вершин-субъектов  $v_{01} \in G_1$  и  $v_{j2} \in G_2$  сравниваются между собой инцидентные ребра-отношения также по принадлежности к классам ПОО и значениям атрибутов, а затем – вершины, отображающие вторые субъекты этих отношений.

Если заданный радиус окрестности лучевого графа больше 1, то далее процесс сопоставления распространяется волновым способом.

### 3. Выявление противоречий в семантических конструкциях

В результате морфологического анализа текста, осуществляемого на основе словаря А.А.Зализняка [19], мы получаем таблицу, содержащую все возможные леммы и словоформы для разбираемого слова, а также морфологические признаки этих словоформ.

Синтаксический анализ производится с помощью правил продукционного типа. В ходе его определяются:

- члены предложения (подлежащее, сказуемое, обстоятельство, дополнение и т.д.);
- вид предложения (повествовательное, вопросительное, восклицательное) – определение эмоциональной окраски предложения по знаку в конце предложения («?», «!», «.»);

- сложность предложения (простое, сложно-сочиненное, сложноподчиненное).

Задачами семантического анализа являются:

- установление ролей субъектов в описываемой ситуации;
- идентификация и слияние характеристик одних и тех же субъектов, найденных в разных предложениях, в частности, обозначенных местоимениями;
- сокращение семантической многозначности слов.

Главными ролями являются действие и актер. Действие обычно выражено глаголом, актер – именем существительным или личным местоимением. Актер – главное действующее лицо. Остальные роли являются менее важными:

- 1) объект – объект действия (дополнение);
- 2) реципиент – приемник действия, т.е. то, на что направлено действие;
- 3) инструмент, используемый при совершении действия;
- 4) сцена – место, где происходит действие (обстоятельство места);
- 5) время действия (обстоятельство времени);
- 6) атрибут – относительная или качественная характеристика, выраженная прилагательным, наречием, причастием или числительным (определение);
- 7) вводное слово – особая роль, отведенная для так называемых вводных слов, т.е. слов, стоящих в начале предложения и имеющих лишь функцию эмоциональной окраски предложения (к ним обычно относятся междометия и союзы);
- 8) слово-связка – особая роль, передающая связи между простыми предложениями (обычно выражается союзами, наречиями и знаками препинания);

9) количество – число предметов или выполняемых действий, выражается числительным.

**Сборка семантической конструкции.** Детали семантической конструкции, в принципе, можно выделить, пользуясь априорными знаниями. Для этого используется описанная выше плекс-грамматика, порождающая грамматически правильные структуры понятий: предмет и его свойства; процесс, его свойства и участники; отношения между предметами и процессами. При этом конструкции (сборки) получают грамматически правильными, но не всегда имеют смысл, поскольку тезаурус естественного языка ограничен и многие слова имеют не-

сколько значений. Поэтому требуется база опыта, в данном случае в виде онтологии ситуаций, сценариев, сюжетов. Кроме того, поскольку алгоритм процесса (например: *Пилить, Конать, Строить*) существенно зависит от ситуации и участников, в частности, от актора, инструмента, бенифицианта, каждый процесс в онтологии (как и в толковых словарях) содержит *множество возможностей* в виде списка ссылок на классы ситуаций. Эти связи формально можно представить как *бинарные отношения* «быть прецедентом». Также предметы: каждое понятие содержит ссылки на понятия процессов, в которых они могут быть использованы.

В режиме сборки семантической модели текста в качестве прецедентов служат словарные статьи – варианты применения слова или словосочетания в различных текстах.

С целью ускорения поиска нужного варианта перед записью в онтологию знаний о предметной области граф прецедента перестраивается таким образом, чтобы первой была вершина, назначенная экспертом в качестве ключевой (опорной, отправной). При этом вершины (субъекты ситуации) и ребра (отношения) снабжены ссылками на узлы (классы) соответствующих родовидовых деревьев онтологии.

Очевидно, такая процедура вносит в базу знаний субъективизм эксперта, но для алгоритма сопоставления не важно, какая вершина будет первой. Просто процесс несколько замедлится. Естественно, со временем акценты в знаниях о прецедентах могут меняться.

Таким образом, сборка непротиворечивой семантической конструкции – это подбор вариантов толкования слов, словосочетаний, выражений текста таких, что полученная конструкция удовлетворяет какому-то определению или понятию онтологии предметной области текста либо непосредственно совпадает с одним из экземпляров понятия, описывающего какой-либо класс ситуаций или сюжетов (пока абстрагируемся от проблемы формирования новых понятий).

При необходимости, граф анализируемого текста может быть дополнен информацией из найденного прецедента в незанятых точках примыкания модели (см. выше).

**Пример 1.** *Time Flies Like An Arrow* [20, 21]. Рассмотрим два варианта семантических конструкций, исходя из полученных ролей слов в ситуации (сюжете).

1. Актор: *муха*; свойства: *временная*; субъект: *стрела*; отношение сопоставления: *подобна*. Здесь предикаты: *Иметь Свойство (Муха, Временная) = False; Похожи (Муха, Стрела) = True* (с низким сходством лучевых графов предметов).

2. Актор: *время*; действие: *летит*; субъект: *стрела*; отношение сопоставления: *подобно*. В этой семантической конструкции предикаты:

*Актор (Летит, Время) = True;*

*Похожи (Прецедент (Время, Летит), Прецедент (Стрела, Летит)) = True* (с высоким сходством по атрибуту скорости процесса).

Таким образом, выигрывает второй вариант.

**Пример 2.** *"I must fly now. She hates waiting around."*

"Fly": 1. Имя существительное – летающее насекомое.

2. Глагол:

- летать каким-либо воздушным видом транспорта;

- пилотировать, управлять каким-либо летательным аппаратом;

- летать, разлетаться самостоятельно под воздействием какой-то силы или по инерции;

- переносное – двигаться быстро.

Как имя существительное "fly" противоречит структуре простого предложения, так как следует за "must". В значении глагола можно отбросить третий вариант, потому что имеется атрибут долженствования. "Wait around" (устойчивое словосочетание) – слоняться в ожидании чего-либо. Тогда, с учетом второго предложения "она очень не любит ждать", для глагола "fly" остается единственный вариант – двигаться быстро (торопиться).

**Пример 3.** Задача про миссионеров и каннибалов [22].

*Три миссионера и три каннибала должны переплыть на другой берег в лодке, в которой могут поместиться только двое. Миссионеры должны соблюдать осторожность, чтобы количество каннибалов не превысило количество миссионеров на каком-либо берегу. Как переплыть реку?*

Семантическая модель текста задачи показана на Рис.2, данные на входе решателя – на Рис.3, фрагмент графа состояний задачи – на Рис.4.

На Рис.4 узлы отображают состояния: простой кружок – промежуточное, зачеркнутый – неудовлетворительное, квадрат – целевое состояние. Ребра представляют действия и гипотезы.

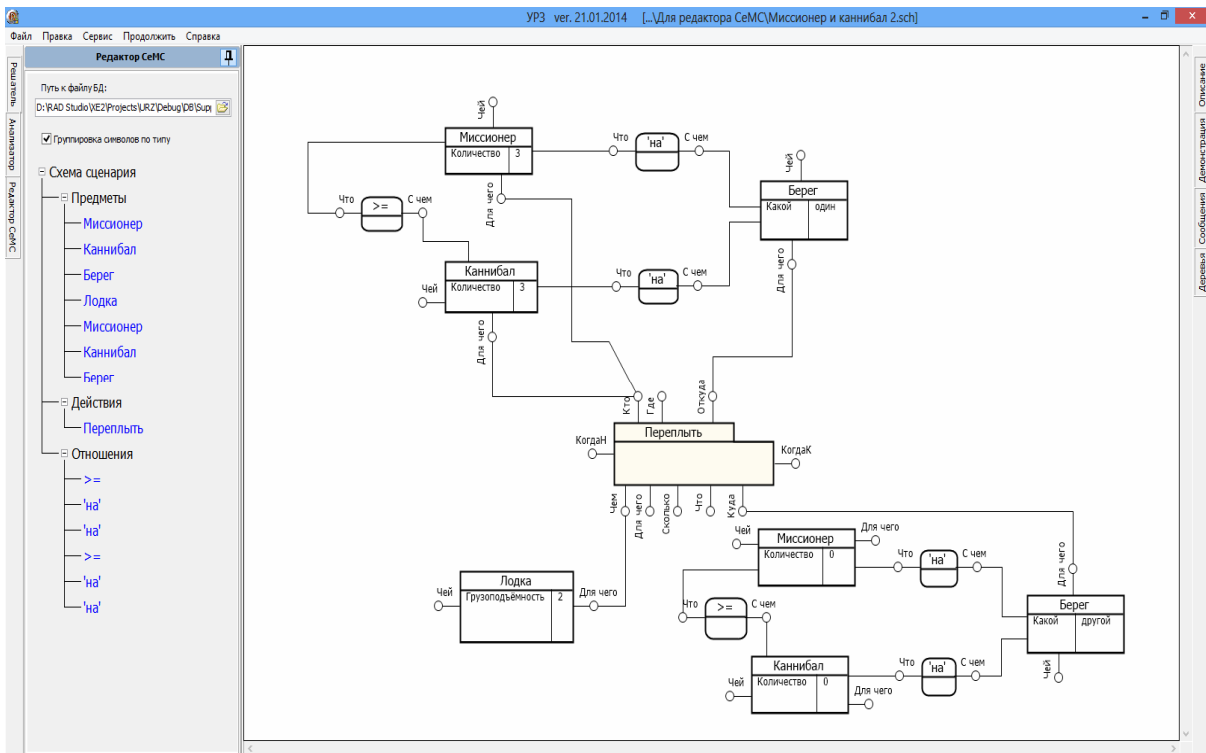


Рис. 2. Семантическая модель текста задачи о миссионерах и каннибалах

Исходная ситуация						Целевая ситуация					
???	1 Имя	2 Какой	3 Сколько	4 Место	5 Грузоподъемность	???	1 Имя	2 Какой	3 Сколько	4 Место	5 Грузоподъемность
1	Берег	Один				1	Берег	Один			
2	Берег	Другой				2	Берег	Другой			
3	Лодка			Берег [Какой=Один]	2	3	Лодка			Берег [Какой=Другой]	2
4	Миссионер		3	Берег [Какой=Один]		4	Миссионер		3	Берег [Какой=Другой]	
5	Каннибал		3	Берег [Какой=Один]		5	Каннибал		3	Берег [Какой=Другой]	

Действия						
???	1 Имя	2 Кто	3 НаЧем	4 Откуда	5 Куда	6 Операция
1	Переплывать	Несколько Миссионер И/ЛИ Несколько Каннибал	Лодка	Берег ?	Противоположно Откуда	Переслать

Ограничения					
1	Количество Каннибал [Место = Берег [Какой=\$X]]	<=	Количество Миссионер [Место = Берег [Какой=\$X]]		
2	ИЛИ				
3	Количество Миссионер [Место = Берег [Какой=\$X]]	=	0		

Рис.3. Данные на входе решателя

тезы, в результате которых преобразуются состояния; штрих-пунктиром изображаются действия, которые не удовлетворяют ограничениям на действия.

Таким образом, результаты решателя подтверждают, что машина правильно поняла текст задачи.

## Заключение

Поиск прецедентов в базе знаний необходим как для уточнения семантики текстовых описаний ситуаций и сюжетов, так и для заимствования опыта в ходе принятия решений.

Обычно процесс представляется как черный ящик с множеством входов и выходов. Описанная



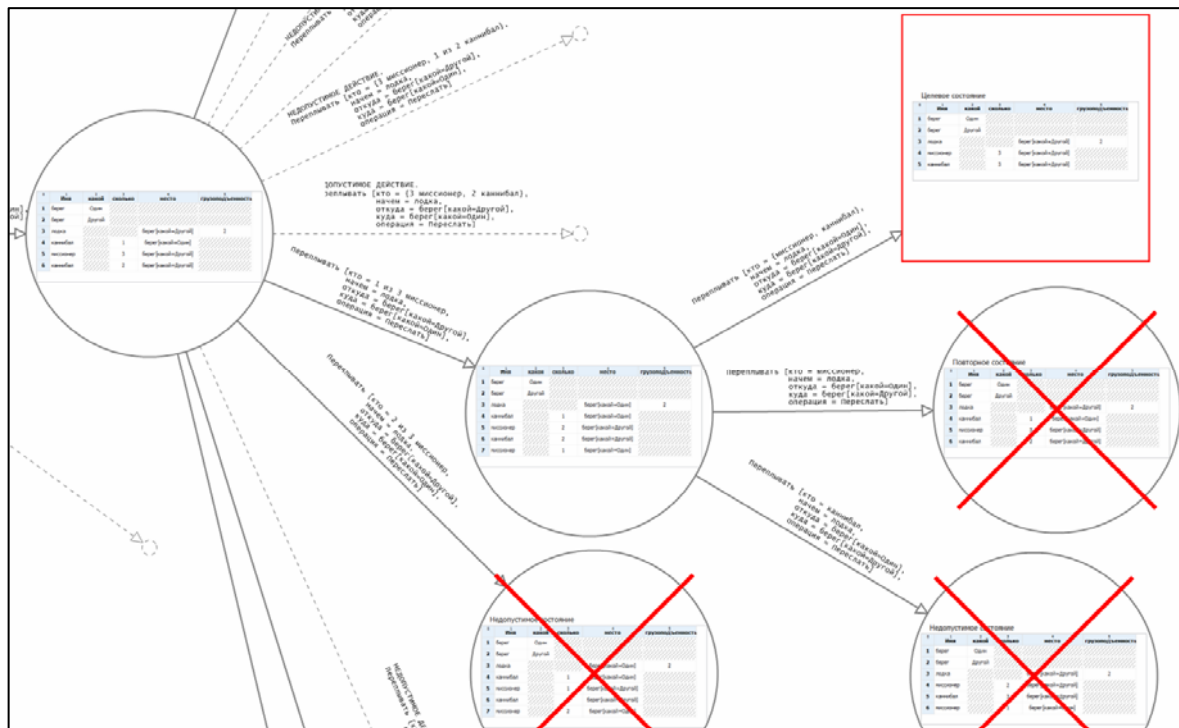


Рис. 4. Фрагмент графа состояний задачи

процессуально ориентированная онтология рассматривает процесс как квантификатор процессуальных отношений – систему зависимостей, где каждое отношение на своем месте, в своей роли, но каждое из них оказывает влияние на другие.

Специализированная плекс-грамматика, порождая грамматически правильные синтактико-семантические конструкции, позволяет на ранних этапах, еще до поиска в базе знаний, устранить часть противоречий, возникающих из-за многозначности слов и словосочетаний.

Лучевой граф окрестности (ЛГО) заданного радиуса как инструмент сопоставления графов сюжетов, в отличие от популярного лучевого поиска, который задает направление, где следует искать, оценивает окрестность всю сразу и целиком, и, в отличие от волнового метода, рассматривает ее в пределах необходимости, что позволяет автоматически настраивать стратегию поиска. Предикат ЛГО вычисляет значение истинности (возможности) существования семантической конструкции текста или его фрагмента и одновременно показывает отличия от найденного аналога.

## Литература

1. Интеллектуальная поисковая система Exactus. <http://www.exactus.ru>.
2. Осипов Г. С. Приобретение знаний интеллектуальными системами: Основы теории и технологии. — М.: Наука. Физматлит, 1997. — 112 с.
3. Осипов Г.С., Смирнов И.В., Тихомиров И.А. Реляционно-ситуационный метод поиска и анализа текстов и его приложения // Искусственный интеллект и принятие решений. № 2.-2008. - С. 3-10.
4. Золотова Г. А., Онипенко Н. К., Сидорова М. Ю. Коммуникативная грамматика русского языка. Институт русского языка РАН им. В. В.Виноградова, М. 2004. – 544с.
5. Sowa J.F. Conceptual Structures: Information Processing in Mind and Machine, Addison-Wesley, Boston, 1984.
6. Chein M., Mugnier M.-L. Graph-Based Knowledge Representation: Computational Foundations of Conceptual Graphs, Springer, London, 2008.
7. Morton S.K. Conceptual Graphs and fuzziness in artificial intelligence, PhD Thesis, University of Bristol, 1987.
8. Wuwongse V., Manzano M. Fuzzy conceptual graphs, in Conceptual Graphs for Knowledge Representation, LNAI Vol. 699, eds G.W. Mineau, B. Moulin and J.F. Sowa (Spring-Verlag, Berlin, 1993) pp. 430-449.
9. Jeovons P.G. On the algebraic structure of combinatorial problems. Theoretical Computer Science (TCS), 200:185-204, 1998.
10. Cao, Tru H.A. Formalism for Representing and Reasoning with Linguistic Information. International Journal of

- Uncertainty, Fuzziness & Knowledge-Based Systems; Jun2002, Vol. 10 Issue 3, p281, 27p.
11. Рассел Стюарт, Норвиг Питер. Искусственный интеллект: современный подход, 2-е изд.: пер. с англ. – М.:Издательский дом «Вильямс», 2006. – 1408 с.: ил.
  12. Lee J., Han W.-S., Kasperovics R., Lee J.-H. An in-depth comparison of subgraph isomorphism algorithms in graph databases. Proceedings of the VLDB Endowment. Vol. 6, Is. 2 (2012) 133-144.
  13. Кучуганов В.Н., Вербализация реальности и виртуальности. Ассоциативная семантика // Искусственный интеллект и принятие решений, 2011. – №1. – С. 95-106.
  14. Feder J. Plex languages, Information Sci., 1971. - №3, P. 255-241.
  15. Fu K. S. Syntactic Methods in pattern recognition, Academic Press. New York and London, 1974. Русский перевод: Фу К. Структурные методы в распознавании образов. – М.: Мир, 1977.
  16. Кучуганов В.Н., Элементы теории ассоциативной семантики. – Управление большими системами. Вып. 40. – М.: ИПУ РАН, 2012, с.30-48.
  17. Fillmore Ch.J. The case for case, Universals in Linguistic Theory, ed. by E.Bach, R.T. Harms. New York, 1968.
  18. Kuchuganov A.V. Recursions in Image Analysis Problems. – Pattern Recognition and Image Analysis, 2009, Vol. 19, No. 3, pp. 501-507, копирайт «Pleiades Publishing, Ltd.».
  19. Программы для морфоанализа. URL: <http://macrocosm.narod.ru/madown.html> (дата обращения: 18.05.2012).
  20. Bruck Piter. The computer age and its potencial for management. Harper & Row. p. 62, 1965.
  21. Mark de Mey. The cognitive paradigm: an integrated understanding of scientific development. University of Chicago Press, 1992.
  22. Лихтарников Л.М. Занимательные логические задачи. (Для учащихся начальной школы)/ Оформление С.Григорьева - Издательство: СПб: Лань, МИК, 1996. - 125 с.

**Кучуганов Валерий Никонорович.** Профессор, заведующий кафедрой «Автоматизированные системы обработки информации и управления» Ижевского государственного технического университета им. М.Т.Калашникова. Окончил Ижевский механический институт в 1964 году. Доктор технических наук. Автор 115 статей и четырех монографий. Область научных интересов: когнитивное моделирование, интеллектуальные информационные технологии.