



Сибирский государственный университет
науки и технологий имени академика М.Ф. Решетнева

Н. В. Лутошкина

МОДЕЛИ ЗНАНИЙ И ОНТОЛОГИИ

Министерство науки и высшего образования Российской Федерации
Сибирский государственный университет науки и технологий
имени академика М. Ф. Решетнева

Н. В. Лутошкина

МОДЕЛИ ЗНАНИЙ И ОНТОЛОГИИ

*Утверждено редакционно-издательским советом университета
в качестве учебного пособия для студентов бакалавриата
по направлениям подготовки
09.03.01 «Информатика и вычислительная техника»,
09.03.02 «Информационные системы и технологии»
всех форм обучения*

Красноярск 2021

УДК 004.82(075.8)

ББК 87.1я7

Л86

Рецензенты:

кандидат физико-математических наук, доцент А. А. ШЛЕПКИН
(Сибирский федеральный университет);
кандидат физико-математических наук, доцент Л. В. ВОПИЛОВА
(Сибирский государственный университет науки и технологий
имени академика М. Ф. Решетнева)

Лутошкина, Н. В.

Л86 Модели знаний и онтологии : учеб. пособие / Н. В. Лутошкина ;
СибГУ им. М. Ф. Решетнева. – Красноярск, 2021. – 80 с.

Приведены основные понятия и определения, рассмотрены вопросы представления и моделирования знаний в информационных системах; описаны процессы проектирования и построения онтологий, инструментарий моделирования и создания программного продукта. Предназначено для студентов направлений подготовки 09.03.01 «Информатика и вычислительная техника» и 09.03.02 «Информационные системы и технологии».

УДК 004.82(075.8)
ББК 87.1я7

ОГЛАВЛЕНИЕ

Введение	4
1. Модели знаний	5
1.1. Представление знаний	5
1.2. Формализация знаний	6
1.3. Лексикографическое (словарное) описание	9
1.4. Логико-интуитивное описание	10
1.5. Формально-языковое описание	11
1.6. Свойства знаний	11
1.7. Классификация знаний	14
1.8. Модели представления знаний	17
Контрольные вопросы и задания	33
2. Онтологии	33
2.1. Онтология. Основные понятия	34
2.2. Цели создания онтологий	36
2.3. Структура онтологии	37
2.4. Классификация онтологий	39
2.5. Жизненный цикл онтологии	43
2.6. Существующие онтологические ресурсы	44
2.7. Модели онтологий	47
2.8. Методы построения онтологий	49
2.9. Инструментарий построения онтологий	51
2.10. Языки описания онтологий	52
2.11. Стандарт онтологического исследования IDEF5	56
2.12. Программные инструменты для построения и работы с онтологиями	59
2.13. Построение онтологий верхнего уровня	60
2.14. Определение структур словарной статьи и соответствующего ей фрейма	61
2.15. Построение семантической сети	63
2.16. Построение иерархической системы терминов	65
2.17. Построение онтологий предметных областей	66
Контрольные вопросы и задания	72
Заключение	74
Библиографический список	75
Приложения	77
<i>Приложение 1</i>	77
<i>Приложение 2</i>	78

ВВЕДЕНИЕ

Информационные технологии становятся все более важным фактором научно-технического прогресса. Кибернетизация производства и науки демонстрирует радикально новые возможности возникающей принципиально новой техники и технологии. К двум фундаментальным компонентам развития цивилизации – материи и энергии добавилась третья – информация, характеризующая степень разнообразия первой и второй компонент. Движущей силой развития общества стало производство не материального, а информационного продукта, в результате чего изменился весь уклад жизни, изменилась система ценностей. Главную роль стали играть интеллект и знания.

Искусственный интеллект – одна из новейших наук, появившихся во второй половине XX века на базе вычислительной техники, математической логики, программирования, психологии, лингвистики, нейрофизиологии и других отраслей знания. Сейчас искусственный интеллект рассматривают как прикладную область исследований, связанных с имитацией отдельных функций интеллекта человека.

Представление знаний в системах искусственного интеллекта – это не только фундаментальное понятие, но и решающий аспект их разработки. Выбор модели представления знаний оказывает огромное влияние на любую часть системы искусственного интеллекта и предопределяет их возможности (свойства и характеристики).

Еще одним инструментом построения баз знаний являются онтологии. Онтологии используются для формальной спецификации понятий и отношений, которые характеризуют определенную область знаний. Преимуществом онтологий в качестве способа представления знаний является их формальная структура, которая упрощает их компьютерную обработку.

Дисциплина «Модели знаний и онтологии» преподается в 8 семестре. Главы учебного пособия предназначены структурировать знания, полученные студентами, изучавшими такие дисциплины, как дискретная математика, математическая логика, теория алгоритмов, основы программирования, структуры и представление данных, геоинформационные системы.

«Модели знаний» – раздел, описывающий основные положения и понятия представления знаний; модели и методы моделирования знаний.

«Онтологии» – раздел, знакомящий с основными понятиями компьютерных онтологий, методами моделирования и основными моделями, языком описания онтологий, инструментарием построения и визуализации, а также с имеющимися онтологическими ресурсами.

Объем дисциплины «Модели знаний и онтологии» составляет 108 часов. В учебном пособии приведен перечень ключевых слов для дисциплины (прил. 1).

1. МОДЕЛИ ЗНАНИЙ

1.1. Представление знаний

Под представлением знаний подразумевают соглашение о том, как описывать реальную предметную область (понятия и отношения). Иногда такое соглашение называют нотацией. Каждая модель знаний определяет форму представления знаний и является формализмом, призванным отобразить объекты, связи между ними и отношения, иерархию понятий предметной области и изменение отношений между объектами.

В рамках направления «представление знаний» решаются задачи, связанные с формализацией представлением знаний в памяти ИС. Для этого разрабатываются специальные модели представления знаний и языки для описания знаний, выделяются различные типы знаний. Основными вопросами при представлении знаний являются определение состава знаний, т. е. определение того «что представлять» и «как представлять» знания.

При представлении знаний решаются следующие задачи:

- определение состава представляемых знаний;
- организация знаний;
- определение проблемной среды;
- разработка архитектуры информационных систем;
- определение потребностей и целей пользователей;
- выбор языка общения.

Для функционирования информационных систем требуются следующие знания:

- знания о процессе решения задачи;
- знания о языке общения и способах организации диалога, используемые лингвистическим процессором;
- знания о способах представления модификации знаний, используемые компонентом приобретения знаний;
- поддерживающие структурные и управляющие знания, используемые объяснительным компонентом;
- знания о методах взаимодействия с внешним окружением;
- знания о модели внешнего мира.

Состав знаний зависит от требований пользователя и проявляется в следующем:

- какие задачи и с какими данными хочет решать пользователь;
- каковы предпочтительные способы и методы решения;
- при каких ограничениях на количество результатов и способы их получения должна быть решена задача;

- каковы требования к языку общения и организации диалога;
- какова степень конкретности знаний о проблемной области, доступная пользователю;
- каковы цели пользователей.

1.2. Формализация знаний

Интеллектуальная деятельность человека связана с поиском решений (действий, закономерностей), в новых, нестандартных ситуациях. Любая интеллектуальная деятельность опирается на знания о предметной области, в которой ставятся и решаются задачи.

Данные – это факты, сведения и идеи, представленные в формализованном виде, позволяющем передавать или обрабатывать их

Под термином знания подразумевается не только та информация, которая поступает в мозг через органы чувств. Такого типа знания чрезвычайно важны, но недостаточны для интеллектуальной деятельности. Дело в том, что объекты окружающей нас среды обладают свойством не только воздействовать на органы чувств, но и находиться друг с другом в определенных отношениях. Ясно, что для того, чтобы осуществлять в окружающей среде интеллектуальную деятельность (или хотя бы просто существовать), необходимо иметь в системе знаний модель этого мира. В этой информационной модели окружающей среды реальные объекты, их свойства и отношения между ними не только отображаются и запоминаются, но и могут мысленно целенаправленно преобразовываться. При этом существенно то, что формирование модели внешней среды происходит в процессе обучения на опыте и адаптации к разнообразным обстоятельствам.

Знания – это совокупность сведений, образующих целостное представление, соответствующее определенному уровню осведомленности о некотором вопросе, предмете, проблеме, явлении. Знания описывают основные закономерности предметной области, позволяющие человеку решать конкретные производственные, научные и другие задачи. Знания являются основным понятием в информационных системах.

Под термином знания в искусственном интеллекте понимают совокупность специализированных (ориентированных на решение многих задач из ограниченной предметной области) фактов, правил их обработки, условий применения правил к конкретным фактам, методов получения новых фактов и способов организации процесса логического вывода.

Проблема выделения знаний, прежде всего, относится к областям преобладания эмпирического знания, где накопление фактов опережает развитие теории. Знания важны там, где определения размыты, понятия меняются, ситуация зависит от множества контекстов, где велика неопределенность и нечеткость информации (контекст – относительно закончен-

ная в смысловом отношении часть текста высказывания). Таким образом, знания – это специальная форма представления смысловой информации, позволяющая хранить, воспроизводить и понимать эту информацию.

Знания обычно представляют в форме фактов, характерных для окружающего мира, и правил манипулирования фактами. Причем под фактом понимают элементарное высказывание с некоторой оценкой. Любую осмысленную часть факта считают данными, т. е. факты – это совокупности данных.

Данные сами по себе не несут смысловой нагрузки. Например, число 16.40 не имеет в себе смысла до тех пор, пока мы не узнаем, что это время отправления поезда или цена товара, то есть, данные нуждаются в интерпретации. В отличие от данных знания несут в себе определенную смысловую нагрузку, представляя собой нечто большее, чем просто последовательность символов. Этот смысл позволяет путем символической обработки получать новую информацию.

Знания людей представляют собой проверенный практикой результат познания действительности, являющийся верным ее отражением в их мышлении в виде «картины мира». Мышление как познавательная способность человека является средством формирования «картины мира» а сама «картины мира» – ее результатом. Это проявляется в таких мыслительных процессах, как предвосхищение, понимание и др. Однако знания, используемые в этих процессах, рассматриваются как прошлый опыт. В результате мышления формируются также и новые знания, но, будучи полученными, они тут же автоматически переходят в разряд прошлого опыта. Таким образом, основной формой существования знаний является прошлый опыт, который хранится в памяти человека.

В настоящее время уделяется большое внимание исследованиям памяти: процессов запоминания, хранения, воспроизведения, забывания информации человеком, а также выявлению когнитивных структур, в виде которых человек хранит информацию об окружающем его мире. Результаты этих исследований находят отражение в формализации знаний человека в интеллектуальных системах.

Каждая из интеллектуальных систем соотносится с определенной частью реального мира – сферой деятельности человека, выделенной и описанной в соответствии с некоторыми целями и называемой *предметной областью*. Описание предметной области представляет собой совокупность сведений: а) обо всех предметах – объектах, процессах и явлениях, выделенных с точки зрения рассматриваемой деятельности; б) об отношениях между выделенными предметами и/ или их частями; в) обо всех проявившихся и возможных взаимодействиях между предметами, их частями и отношениями, возникших в результате осуществления деятельности человека.

Сведения о предметах представляются в знаковой форме. В зависимости от типа знаковой системы сведения могут быть представлены в виде

естественно – или искусственно – языковых, графических, речевых, жестомимических образов, а также в другой форме, например при использовании индексных знаковых систем, в виде показаний приборов.

Источником знаний является субъект. Любые технологии, процедуры, операции с знаниями могут быть реализованы только им самим или совместно с ним как некоторая процедура их «опредмечивания», обретения ими натуралистического предмета – носителя образа (знака). Назовем эту процедуру *автоформализацией знаний*. Название процедуры представления знаний выглядит весьма обоснованно: *автоформализация знаний – это придание владельцем знаний (их автором) им некоторой формы*.

Введем некоторые определения с целью систематизации последующих рассуждений о представлении знаний. *Представление знаний* (то есть автоформализация знаний) – любые действия со знаниями, осуществляемые субъектом, направленные на формирование внешнего образа, в том числе и первое представление.

Интерпретация знаний – последующие представления знания, формирование последующих внешних образов (метаобразов, моделей, метазнаний, метамоделей и т. д.).

Методы (способы) представления знаний – действия, система действий (совокупность процедур, операций и приемов) по представлению знаний в форме внешних образов.

Формализация знаний, организация знаний, анализ знаний, извлечение знаний и т. д. – именованные производные информационные технологии представления знаний.

Формализация знаний – это представление знаний субъекта о предметной области с использованием знаковых систем. В зависимости от типа знаковой системы данные могут быть представлены в виде естественно-или искусственно-языковых, графических, речевых, жестомимических образов, а также в другой форме.

Дифференцированными методами формализации знаний являются: естественно-языковое описание, лексикографическое описание, тезаурусное описание, формально-языковое.

Естественно-языковое описание. Формализация знаний на основе естественно-языковой знаковой системы приводит к формированию естественно-языкового описания, текста. Текст является наиболее полным представлением знаний субъекта. Он отражает не только сами образы, но и процесс их порождения, т.е. содержит и декларативную и процедуральную компоненты знания.

Инструментарием естественно-языкового представления знаний является сам естественный язык, который можно рассматривать как универсальный код представления знаний. Любое знаковое представление знаний развернуто в пространственно-временном континууме. Текст отражает течение мысли и в связи с этим его организация последовательна и

непрерывна. Знание в тексте распределено по всему его протяжению от начала – первого образа, и до конца – последнего. Развиваясь во времени, как в процессе порождения, так и в процессе последующего восприятия, текст характеризуется некоторой величиной накопленного знания и динамикой его накопления.

Формализация на основе естественного языка представляет собой информационную технологию, являющуюся основной. Приемы, процедуры и операции этой технологии являются составными частями других информационных технологий.

Инструментарий формализации включает теоретические и практические методы различных наук. Современные средства поддержки технологии – автоматизированные системы анализа и обработки текстовых данных (текстовые редакторы и процессоры).

1.3. Лексикографическое (словарное) описание

Для формализации знаний о реальном мире может оказаться достаточно использовать такие конструкции естественного языка, которые позволяют несвязно, непоследовательно и прерывно описывать объекты и процессы в виде системы слов и их определений. В этом случае описание слова частично, не многосторонне представляет знание субъекта. Такой способ представления знаний называется лексикографическим, а его результат – лексикографическим (словарным) описанием, словарем.

Лексикографическая формализация знаний порождает следующие типы текстовых структур: словники, словоуказатели, словари. В общем случае объектом лексикографического описания может быть знание (образ), не связанное только с какой-либо одной языковой единицей, например словом. В этих случаях лексикографическое описание будет представлять словарь языковых единиц.

Если в качестве языковой единицы в лексикографическом описании будет рассматриваться текст, то такого типа словари принято относить к методам организации знаний.

Лексикографическая информационная технология включает следующий набор основных процедур: отбор источников, анализ текстов, составление словников и словоуказателей, анализ словоупотреблений, составление словарных статей, собрание словаря.

Инструментарий лексикографической технологии представляют методы теоретической и практической лексикологии и лексикографии, научных дисциплин языкознания, а также современные автоматизированные системы анализа и построения словарей, а также реализованные в виде информационно-программных изделий электронные словари и энциклопедии.

1.4. Логико-интуитивное описание

Для формализации знаний может оказаться достаточным использование таких средств естественного языка, которые только обозначают внутренние образы субъекта (предметы реального мира и их понимание), именуют их в форме слов-понятий. В этом случае знание субъекта представляется только в декларативной форме. Система понятий, образующаяся при этом, содержит в себе сами слова-понятия и слова-понятия отношений между ними, называемые *парадигматическими*. Таким образом, знание субъекта оказывается представленным в форме пар слов-понятий, связанных между собой поименованными словами-отношениями. Каждая такая пара является логической конструкцией, называемой *суждением* или *утверждением*.

Парадигматические отношения представляют понимание субъектом внутреннего образа, т. е. конечную цепочку преобразований внутреннего знака (включая концептные, денотативные и коннотативные компоненты), и фиксируют его в виде внешнего знака (понятия). Парадигматические отношения, таким образом, являются концептными (смысловыми), денотативными (призначными) и коннотативными (чувственно-оценочными) свойствами знаний. В качестве примера можно выделить следующие виды парадигматических отношений смысловых свойств знаний: эквивалентность смыслов слов (полная или частичная *синонимия*), полярная противопоставленность смыслов слов (*антонимия*), включение смысла одного слова в смысл другого (отношения «род–вид»), связь смыслов типа «причина–следствие», связь смыслов типа «часть–целое», функциональное *сходство* и др.

Формализацию знаний путем именованного предметов и отношений между ними словами-понятиями естественного языка называют *тезаурусным описанием*, а результаты такого описания – *тезаурусами*.

Особенностью тезаурусного способа представления знаний является интуитивный характер выявленных отношений, нечеткость и произвольность. Тезаурусное представление знаний фрагментирует знание, структурирует его в пространственно-временном континууме так, что всё знание оказывается разделенным на отдельные группы понятий (парадигматические классы, классы эквивалентности), связанных между собой определенными отношениями.

В связи с тем, что тезаурус представляет собой систему связанных между собой понятий, являющихся образами предметной области реального мира, его можно назвать *понятийной картиной мира*. Если считать, что в самом общем случае любое слово естественного языка может быть соотнесено (связано отношением) с любым другим словом, то тезаурус можно назвать *языковой картиной мира*.

Информационная технология тезаурусного описания в общем случае включает процедуры формирования списков слов-понятий (слов-отношений)

и группировки их в парадигматические классы. Более подробно она будет рассмотрена ниже.

Инструментарий тезаурусной технологии представлен теоретическими и практическими методами многих наук, в числе основных: языкознание и информатика. Современными средствами технологии являются информационно-программные изделия анализа и организации естественно-языковой информации.

1.5. Формально-языковое описание

Для формализации знаний могут быть использованы искусственные знаковые системы – формальные языки. При этом вводятся абстрактные языковые конструкции, которым ставятся в соответствие объекты и процессы реального мира, тем самым осуществляется описание знаний в виде формальных моделей. Произвольный формальный язык, используемый для описания предметной области, базируется на конкретном наборе первичных, абстрактных конструкций, которые определяют или описывают знания и изменяют их. Инструментальными средствами поддержки формально-языкового описания знаний являются различные информационно-программные изделия.

1.6. Свойства знаний

Знания обладают рядом существенных свойств:

– единицей обрабатываемой информации является факт, а не запись данных;

– знания обладают *внутренней интерпретированностью* – понимание смысла, семантики информационных единиц обеспечивается внутри модели, а не только в сознании исследователя; Данные в ЭВМ могут интерпретироваться только соответствующей программой. Знания отличаются тем, что в них присутствует возможность содержательной интерпретации. Каждая информационная единица должна иметь уникальное имя, по которому ИС находит ее, а также отвечает на запросы, в которых это имя упомянуто;

– знания обладают *активностью* – появление новых фактов в системе приводит к инициации процедур, обрабатывающих уже известные системе факты, т. е. данные управляют программой;

– знания обладают *структурируемостью*. Структурированность знаний проявляется при использовании отношений типа «элемент–класс» и «часть–целое», что позволяет выделять реализовать возможность вложения одних знаний в другие; Информационные единицы должны обладать

гибкой структурой. Разнообразные формы хранения данных не обеспечивают возможности компактного описания всех связей между различными типами данных. При переходе к знаниям между отдельными единицами знаний можно установить такие отношения как «элемент–множество», «тип–подтип», «ситуация–подситуация», «часть–целое», отражающие характер их взаимосвязей. Каждая информационная единица может быть включена в состав другой, и из каждой информационной единицы можно выделить некоторые составляющие ее информационные единицы;

– знания обладают *связностью* – между фактами и правилами имеется возможность установления функциональных, казуальных, структурных, семантических отношений, отношений эквивалентности (подобия, тождества), противоположности и др. В информационной базе между информационными единицами должна быть предусмотрена возможность установления связей различного типа. Прежде всего, эти связи могут характеризовать отношения между информационными единицами. Семантика отношений может носить декларативный или процедурный характер. Например, две или более информационные единицы могут быть связаны отношением «одновременно», две информационные единицы – отношением «причина-следствие». Эти отношения характеризуют декларативные знания. Если между информационными единицами установлено отношение «аргумент-функция», то оно характеризует процедурное знание. Различают отношения структуризации, функциональные отношения, казуальные отношения и семантические. С помощью первых задаются иерархии информационных единиц, вторые несут процедурную информацию, третьи задают причинно-следственные связи, четвертые соответствуют всем остальным отношениям;

– знания обладают *семантической метрикой* – отношение близости понятий, силы ассоциативной связи между ними. Ее наличие позволяет выделять информацию, относящуюся к типовым и степень неопределенности информации о предметной области в процессе перехода к очередному этапу решения интеллектуальной задачи. Это ситуации, например, «выход из строя элемента системы», «блокировка пакетов» и т. п.;

– знания обладают *конвертируемостью представления*. Конвертируемость представления – свойство изменять форму представления, уровень детализации и степень неопределенности информации о предметной области в процессе перехода к очередному этапу решения интеллектуальной задачи. Это свойство, в отличие от предыдущих, формально отражает динамичность знаний и связано с обучением системы.

Формально под фактом подразумевается запись

$$F = \langle N, V, C, L, M \rangle.$$

Здесь N – имя или идентификатор факта; V – значение факта, определяемое на численной шкале с метрикой, логической, нечеткой или

лингвистической шкале; S – степень уверенности; L – множество связей факта с другими знаниями; M – множество допустимых функций, преобразований, операций, способов вычисления значения факта, имеющих смысл в рассматриваемой предметной области.

Другими словами, факт – это запись данных, наделенная семантикой. Правилами в общем случае являются знания вида

«Если X есть A , то Y есть B , иначе есть C ».

Условия применимости правил, а также знания относительно способов использования фактов и правил относятся к метазнаниям (знаниям о знаниях), которые необходимы для управления выводом, пополнения знаний и т. п. Часто такие правила и способы являются эвристическими. Система семантической обработки информации, основанная на концепции баз знаний, включает:

- базу знаний, состоящую из база фактов и правил как декларативной части, а также базы процедур и функций, как процедурной части описания объекта;

- механизм логического вывода – высокоуровневый интерпретатор, обеспечивающий обработку фактов на основе правил и процедур формирования решений для задач пользователя;

- интерфейс с пользователем на языке, близком к естественному;

- базу целей, содержащую механизм целеполагания в рамках исследуемой предметной области и целевые установки самой системы.

Для того чтобы данные превратились в знания, они должны быть определенным образом структурированы.

Знания представляют собой иерархические структуры. Общие знания, касающиеся целых подобластей данной предметной области, включают в себя более узкие, касающиеся каких-то отдельных признаков или специальных вопросов из предметной области.

Между элементами и объектами знаний существуют *функциональные* и *казуальные* (причинностные) отношения. *Функциональные* отношения несут процедурную информацию, позволяющую определять или вычислять одни объекты через другие. *Казуальные* отношения задают причинно-следственные связи.

Семантика (*смысловое значение, содержание*) отношений между объектами может носить декларативный (*данные*) или процедурный (*программы*) характер.

Перечисленные особенности информационных единиц определяют ту грань, за которой данные превращаются в знания, а базы данных перерастают в базы знаний. Совокупность средств, обеспечивающих работу с знаниями, образует систему управления базой знаний.

1.7. Классификация знаний

Знания делятся на формализованные и неформализованные. *Формализованные знания* выражаются в виде законов, формул, алгоритмов, моделей и т. п. Такие знания описываются в книгах и руководствах и отражают точные и универсальные знания в виде строгих суждений.

Неформализованные знания (вербальные, словесные) субъективны и приближительны. Они являются результатом обобщения многолетнего опыта работы и интуиции специалиста и представляют собой некоторое множество эмпирических приемов и правил логического вывода. Это ключевые понятия для искусственного интеллекта.

Поверхностные знания – это в основном приближительные знания, эвристики и некоторые закономерности, устанавливаемые опытным путем. Такие знания в силу их приближительности называют также экспертными.

Глубинные – отражают наиболее общие принципы, в соответствии с которыми развиваются все процессы в предметной области и свойства этих процессов. К глубинным относятся знания, основанные на теориях, абстракциях и аналогиях, в которых отражается понимание структуры предметной области. Для получения глубинных знаний необходимо понять внутренние механизмы, действующие в предметной области, и, прежде всего, основные закономерности, которые обуславливают принятие правильных решений. Глубинные знания используются прежде всего при решении неординарных ситуаций.

Процедурные знания – это знания, которые могут быть представлены процедурой или процессом.

В компьютерной программе эти знания хранятся как код, а не как данные. Программные алгоритмы являются формой процедурных знаний, так как они содержат информацию о том, как решить конкретную задачу.

Декларативные – это знания, которые хранятся как данные. В декларативном представлении легко добавлять или изменять знания, поскольку они независимы от программы.

Статистические – тип знаний, которые не изменяются в процессе решения задачи.

Динамические – могут приобретаться с течением времени.

Жесткие знания – позволяют получать *однозначные* четкие рекомендации при заданных начальных условиях.

Мягкие знания – допускают множественные, расплывчатые решения и различные варианты рекомендаций.

Кроме указанных понятий используется понятие метазнания (*знания о знаниях*). Оно используется для обозначения знаний о способах использования знаний и свойств знаний.

В общем виде знания в ЭВМ представляются некоторой знаковой (семиотической) системой. С понятием «знак» непосредственно связаны

понятия денотат и концепт. *Денотат* – это объект, обозначаемый данным знаком. *Концепт* – свойства денотата.

Экстенционал знака определяет класс всех его допустимых денотатов.

Интенционал знака определяет содержание связанных с ним понятий.

Интенциональные знания описывают абстрактные объекты, события, отношения. Например, поставщик, потребитель, транспорт.

Экстенциональные знания представляют собой данные, характеризующие конкретные объекты, их состояние, значения параметров в определенных отрезки времени. Экстенционалом поставщика может быть завод, потребителя – предприятие, транспорта – автомобиль.

В *семиотической* (знаковой) системе выделяют три аспекта: синтаксический, семантический, прагматический.

Синтаксис описывает внутреннее устройство знаковой системы, т. е. правила построения и преобразования сложных знаковых выражений.

Семантика определяет отношения между знаками и их концептами, т. е. задает смысл или обозначения конкретных знаков.

Прагматика определяет знак с точки зрения конкретной сферы применения, либо субъекта, использующего данную знаковую систему.

Для хранения данных и знаний используются базы данных и базы знаний.

База данных – это совокупность связанных данных, хранящихся с минимальной избыточностью и используемых различными приложениями посредством системы управления базами данных.

База знаний – это совокупность описывающих предметную область правил и фактов, позволяющих с помощью механизма вывода решать вопросы, ответ на которые в явном виде в базе отсутствует. БЗ как программное средство обеспечивает поиск, хранение, преобразование и запись в память ПК сложно структурированных информационных единиц – знаний.

С учетом архитектуры информационных систем знания целесообразно делить на интерпретируемые и не интерпретируемые.

К первому типу относятся те знания, которые способен интерпретировать решатель. Все остальные знания относятся ко второму типу.

Не интерпретируемые знания подразделяются:

– на *вспомогательные*, хранящие информацию о лексике и грамматике языка общения, информацию о структуре диалога;

– *поддерживающие знания*.

Вспомогательные знания обрабатываются естественно-языковой компонентой. *Поддерживающие* знания используются при создании системы и при выполнении объяснений. Поддерживающие знания выполняют роль описаний (обоснований) как интерпретируемых знаний, так и действий системы.

Поддерживающие знания подразделяются на *технологические* и *семантические*.

Технологические содержат сведения о времени создания описываемых ими знаний, об авторе знаний и т. п.

Семантические содержат смысловое описание этих знаний. Они содержат информацию о причинах ввода знаний, о назначении знаний, описывают способ использования знаний и получаемый эффект.

Интерпретируемые знания можно разделить на *предметные*, *управляющие*, *знания о представлении*.

Предметные знания содержат данные о предметной области и способах преобразования этих данных при решении поставленных задач. В предметных знаниях выделяют описатели и собственно предметные знания. Предметные знания разбиваются на факты и исполняемые утверждения. Факты определяют возможные значения сущностей и характеристик предметной области. Исполняемые утверждения содержат информацию о том, как можно изменять описание предметной области в ходе решения задачи, т. е. это знания, задающие процедуры обработки. Описатели содержат определенную информацию о предметных знаниях, такую, как коэффициент определенности правил и данных, меры важности и сложности.

Управляющие знания можно разделить на фокусирующие и решающие.

Фокусирующие описывают, какие знания следует использовать в той или иной ситуации.

Решающие знания содержат информацию, используемую для выбора способа интерпретации знаний, подходящего к текущей ситуации. Эти знания используются для выбора стратегий или эвристик.

Качественные и количественные показатели информационных систем могут быть значительно улучшены за счет использования *метазнаний*, т. е. знаний о знаниях. *Метазнания* не представляют некоторую единую сущность, они могут применяться для достижения различных целей.

Метазнания применяются:

- в виде стратегических метаправил они используются для выбора необходимых правил;
- используются для обоснования целесообразности применения правил;
- используются для обнаружения синтаксических и семантических ошибок в предметных правилах;
- позволяют системе адаптироваться к окружению путем перестройки предметных правил и функций;
- позволяют явно указать возможности и ограничения системы, т. е. определить, что система знает, а чего не знает.

1.8. Модели представления знаний

Системы искусственного интеллекта являются системами, базирующимися на знаниях. Основные свойства таких систем:

- раздельное хранение знаний, представленных в символьной форме и компонентов обработки этих знаний;
- системы могут делать выводы и принимать решения на основании сохраненной информации, которая представляется в системах не явно, однако органически связана с этими системами;
- системы обладают способностью к пояснениям, т. е. по требованию пользователя должны быть в состоянии вывести понятную и ясную для него цепочку рассуждений;
- способность к обучению – способность выводить знания на основании информации, полученной от пользователя.

Под моделью знаний понимается *способ* описания знаний в базе знаний.

В общем случае модели представления знаний могут быть условно разделены на декларативные и процедурные.

Декларативная модель основывается на предположении, что проблема представления некоторой предметной области решается независимо от того, как эти знания потом будут использоваться. Поэтому модель знаний состоит как бы из двух частей: структур, описывающих знания, и механизма вывода, оперирующего этими структурами, независимо от содержательного наполнения этих структур. При этом синтаксические и семантические аспекты разделены. (*Синтаксис – внутреннее устройство знаковой системы – правила построения и преобразования знаковых выражений. Семантика задает смысл или обозначение конкретных знаков*). Описания выполняемых процедур не содержатся в явном виде. Предметная область представляется в виде описания ее состояния, а вывод решения описывается в основном на процедурах поиска в пространстве состояний.

Процедурная модель основывается на небольших программах (процедурах), которые определяют, как поступать в конкретных ситуациях. В этой модели семантика заложена непосредственно в описание элементов базы знаний. Общие правила и правила вывода представлены в виде специальных целенаправленных процедур.

Требования к модели представления знаний:

- однородность представления;
- простота понимания;
- упрощение механизма управления выводом.

Наиболее распространенными являются четыре модели представления знаний в интеллектуальных системах и их комбинации:

- логическая или логика предикатов;

- продукционная;
- фреймовая;
- семантические сети.

Перечень реальных моделей, применяемых в системах искусственного интеллекта, широк и трудно классифицируем. Большинство моделей в реальных системах являются гибридными, да еще и со значительной долей эвристик.

1.8.1. Логическая модель представления знаний

В основе моделей такого типа лежит формальная система, задаваемая четверкой вида

$$M = \langle T, P, A, B \rangle.$$

Множество T – это множество базовых элементов. Например, это могут быть слова из какого то ограниченного словаря или детали конструктора. Для множества T существует некоторый способ определения принадлежности или непринадлежности произвольного элемента к этому множеству. Процедура такой проверки может быть любой, но за конечное число шагов она должна давать ответ на вопрос, является ли x элементом множества T или нет.

Множество P – это множество синтаксических правил. С их помощью из элементов множества T образуют синтаксически правильные совокупности (СПС). Например, из слов словаря строятся синтаксически правильные фразы, из деталей конструктора собираются новые механизмы. Существует процедура, с помощью которой за конечное число шагов можно определить, является ли совокупность X синтаксически правильной.

Во множестве синтаксически правильных совокупностей выделяется некоторое подмножество A . Элементы множества A называются аксиомами. Аксиомы составляют ядро предметной области. Существует процедура, с помощью которой для любой СПС можно получить ответ на вопрос является ли она аксиомой.

Множество B – это множество правил вывода. Применяя их к элементам множества A , можно получать новые СПС, к которым снова можно применять правила из множества B . Так формируется множество выводимых в данной формальной системе совокупностей. Если имеется процедура, с помощью которой можно определить для любой СПС, является ли она выводимой, то такая формальная система называется разрешимой. Правила вывода являются самой сложной частью логической модели.

В логической модели знаний можно считать, что множества A образует все единицы знаний, которые введены в базу знаний извне, а с помощью правил вывода из них выводятся новые производные знания.

Формальная система представляет собой генератор порождения новых знаний, образующий множество выводимых в данной системе знаний. Это свойство логических моделей позволяет хранить в базе лишь те знания, которые образуют множество A , а все остальные знания из них получать по правилам вывода.

На практике логическая модель чаще всего используется в системе логики предикатов и выведения заключений с помощью силлогизмов. *Предикатом или логической функцией* называется функция от любого числа аргументов, принимающих истинное значение И(1) и Л(0). Аргументы принимают значения из произвольного конечного или бесконечного множества M , называемого предметной областью. Предикат от n аргументов является n -мерным предикатом.

Силлогизм – это заключение. С помощью силлогизмов из двух суждений выводится третье. Например:

- все металлы электропроводны;
- медь – металл.

Заключение – медь электропроводна.

Отличительными чертами логических моделей является единственность толкования и возможность реализации системы точных определений и выводов.

Формальные логические модели, основанные на исчислении предикатов, используются как один из методов представления знаний в искусственном интеллекте, не используемых, как правило, в промышленных информационных системах.

В любом языке для создания языковых форм должно быть определено следующее; множество знаков, которые можно в нем использовать; полное определение слов через знаковые последовательности; грамматические правила образования предложений из слов (семантические правила).

Каждое слово из предложения соответствует объектам и действиям того реального мира, который описывается этим языком. Таким образом, в языке присутствуют слова, которые описывают сущности, и слова, которые описывают атрибуты сущностей и действия над ними.

Для формулировки знаний о некоторой предметной области средствами логики предикатов в проблемной области выделяют два основных типа констант:

- дискретные объекты, которые называют сущностями;
- отношения между сущностями.

Именам отношений соответствуют термы определений, а сущностям – аргументы. Символы, используемые для обозначения высказываний, называются атомами. Интерпретацией формулы называется приписывание каждому атому, входящему в формулу, истинностного значения. Таблицей истинности формулы называется таблица, содержащая истинностные

значения формулы при всех возможных комбинациях истинностных значений входящих в нее атомов.

Логический вывод (силлогизм) в классической логике строится двумя путями: дедукцией и индукцией. В современной логике вывод строится также абдукцией. Индукция и абдукция называются вероятностными силлогизмами.

Дедукция – вывод, позволяющий получать заключение из большой или малой посылки. Большая посылка называется дедуктивным правилом, малая – фактическим заявлением (декларацией). Пример: Большая посылка: рыба – живое существо, умеющее плавать. Малая посылка: карась – рыба. Вывод: карась – живое существо, умеющее плавать.

Большая и малая посылки называются данными о знаниях или просто знаниями и их объем должен быть достаточно большим. Упорядоченное множество таких знаний и есть база знаний.

Дедуктивное правило позволяет детализировать и формализовать причинно-следственные связи явлений (процессов) и предметов, то есть это знания, необходимые для выполнения алгоритма принятия решений. Основная часть знаний кодируется не только арифметическими, но и логическими символами, т. е. требуется специальный язык для ЭВМ.

Индукция – получение большой посылки из заключения ее малой посылки. Можно ли считать правомерным определение «рыба – это живое существо, умеющее плавать» в качестве большой посылки, исходя из того, что известны многие виды рыб? Вывод неправоможен, так как могут быть неизвестные виды рыб, перемещающиеся по суше. По мере накопления наблюдений повышается достоверность большой посылки.

Абдукция – получение малой посылки из заключения и большой посылки. На основании факта, что некоторые из живых существ умеют плавать, делается заключение, что это живое существо – рыба. Однако этот вывод нельзя считать безусловно достоверным. Человек в обыденной жизни имеет постоянное дело с нечеткими, вероятностными оценками.

Все операции исчисления высказывания переносятся в исчисления предикатов и используются для связывания предикатов и формул, т. е. позволяют получить из простых высказываний сложные.

Основные операции алгебры логики:

1. \neg – отрицание. Высказывание истинно, если высказывание A – ложно и ложно, если A – истинно.

2. $(A \wedge B)$ – конъюнкция или логическое умножение (A и B). Высказывание истинно только том случае, если истины A и B .

3. $(A \vee B)$ – дизъюнкция или логическое сложение ($A + B$). Высказывание истинно только в том случае, если истинно хотя бы одно из слагаемых.

4. $(A \rightarrow B)$ импликация или следование B из A . Читается также «если A , то B ». Высказывание ложно только в том случае, если A истинно и B ложно.

5. $(A \equiv B)$ – эквивалентность «А тогда и только тогда, когда В». Истинно тогда и только тогда, когда А и В имеют одно и то же истинностное значение.

Таким образом, основным операторам, образующим функции, соответствуют символы логических связок: \neg (не, отрицание), \square (или, дизъюнкция), \square (и конъюнкция), \rightarrow (импликация), \sim (эквивалентность).

В логике предикатов для компактной записи высказываний типа «для любого x истинно $F(x)$ » и «существует такое x , для которого истинно $F(x)$ » дополнительно вводятся две операции – квантор общности и квантор существования. Вышеприведенные высказывания можно записать в следующем виде:

$\forall xF(x)$ – высказывание истинно когда $F(x)$ истинно для всех $x \in M$ и ложно в противоположном случае;

$\exists xF(x)$ – высказывание истинно, когда существует элемент $x \in M$, для которого $F(x)$ истинно и ложно – в противоположном случае.

В логике предикатов обычно используются 6 типов символов:

- предметные переменные x, y, z, u, v или те же буквы с индексами;
- константы a, b, c, d, e или те же буквы с индексами;
- функциональные символы f, q, r, s, t или те же буквы с индексами;
- предикатные символы p, q, r, s, t или те же буквы с индексами;
- символы алгебры логики: $\exists, \forall, \neg, \wedge, \vee, \rightarrow, \sim$;
- вспомогательные символы – круглые скобки и т. п.

Вывод, дающий заключение из двух посылок, можно представить следующим образом:

Предпосылка 1. Все металлы электропроводны;

Предпосылка 2. Медь – металл.

Заключение. Медь электропроводна.

Если вместо имен субъектов подставить переменные x, y , то можно получить следующую логическую формулу:

Предпосылка 1. $\forall x$ металл (x) электропроводен (x)

Предпосылка 2. Металл (медь)

Заключение электропроводна (медь).

Логика предикатов рассматривает вопросы, можно ли, представляя предложения в логических формулах с помощью выводов получить из нескольких логических формул некоторую конечную логическую формулу.

Таким образом, в логике предикатов основным объектом исследования является формула. При ее определении используется понятие «терм» (некоторая сущность), объединяющая названия элементарных функций, к которым применима предикатная буква. Терм определяется следующим образом:

- всякая предметная переменная или константа являются термами;
- если f – n -местная функция из n предметов и t_1, \dots, t_n – термы, то $f(t_1, \dots, t_n)$ есть терм.

Выражение $p(t_1 \dots t_m)$, где p предикатный символ m аргументов и $t_1 \dots t_m$ – термы называется атомарной функцией или атомом.

Атом и всякие выражения из них есть правильно построенные формулы (ППФ).

Пример:

Предпосылка 1 (логическая формула) $\forall x(p(x) \rightarrow q(x))$

Предпосылка 2 (атом) $p(a)$

Заключение (атом) $q(a)$

Здесь p – металл; q – электропроводность; a – медь.

Интерпретация. Формула имеет определенный смысл, т. е. обозначает определенное высказывание, если существует какая-либо интерпретация. Интерпретировать формулу – значит связать с ней определенное множество M , т. е. конкретизировать предметную область, называемую также областью интерпретации и указать соответствие относящееся:

- каждой предметной константе в формуле конкретный элемент из M ;
- каждой n -местной функциональной букве в формуле конкретную n -местную функцию на M ;

- каждой n -местной предикатной букве в формуле конкретное отношение между n элементами из M .

Иными словами – интерпретирование формул исчисления предикатов – это конкретизация предметной области M и соответствия между символами, входящими в формулы с одной стороны, и элементами, функциями и отношениями на M с другой.

Пример: элементарная формула $G(f(a,b), g(a,b))$

Интерпретация: M – множество действительных чисел, a и b – числа 2, 3 соответственно;

f – функция сложения $f(a,b) = (a+b)$; g – функция умножения $g(a,b) = (a*b)$; G – отношение «не меньше».

Формула обозначает высказывание «сумма $2+3$ не меньше произведения $2*3$ ». Результат – ложь или 0. Если изменить интерпретацию, $v = 1$ или $v = 2$, то результат – истина.

Другая формула при той же интерпретации:

$G(f(g(x,x), g(x,y)), g(a,g(x,y)))$

Формула обозначает высказывание $x^2 + y^2 > 2xy$

Высказывание верно при любых x и y из M и всегда истинно.

Для описания высказываний на языке предикатов в заданной предметной области часто для обозначения предикатных букв и констант используются слова или сокращения, которые являются названиями определяемых ими свойств, отношений, объектов.

$(\exists x)$ [Дельфин (x) умный (x)]

Дельфин наделен умственными способностями.

$(\forall x)$ [Слон(x) \rightarrow цвет (x , серый)]

Все слоны имеют серую окраску.

Область применения – для решения сравнительно простых задач.

Достоинства логических моделей – ясная семантика, возможность использования развитых механизмов логического вывода.

Одним из наиболее удобных средств поиска логического вывода является метод резолюций. Он заключается в том, что вместо заданной формулы рассматривается ее отрицание и доказывается противоречивость этой формулы.

1.8.2. Продукционные модели знаний

Продукционные модели знаний занимают особое положение, так как они являются наиболее декларативным способом представления знаний. Продукционная модель представления знаний – это набор правил вида

ЕСЛИ <условие>, ТО <действие>,

где <условие> – образец для поиска в базе данных (утверждение о состоянии базы данных), <действие> – действие, выполняемое при успешном исходе поиска в базе (процедуры, которые могут изменять состояние базы данных). Действия могут быть промежуточными, выступающие далее как условия и целевыми, завершающими работу системы.

В продукционных моделях используются некоторые элементы логических моделей, что позволяет организовывать на них эффективные процедуры вывода, а с другой стороны, более наглядно отражают знания, чем классические логические модели. Правила вывода в этих моделях называются продукциями. Системы продукций – это набор правил, используемый как базы знаний, поэтому его называют еще базой правил. Продукции соответствуют навыкам в долгосрочной памяти человека. Подобно навыкам в долгосрочной памяти эти продукции не изменяются при работе системы. Они вызываются «по образцу» для решения данной проблемы. Рабочая память продукционной системы соответствует краткосрочной памяти, или текущей области внимания человека. Содержание рабочей области после решения задачи не сохраняется.

Примеры фактов и правил.

Факт 1. Зажженная плита – горячая.

Правило 1. Если положить руку на зажженную плиту, то можно обжечься.

В общем виде под продукцией понимается выражение следующего вида:

$$(I);Q;P;A=>B;N,$$

Здесь I – имя продукции, с помощью которого эта продукция выделяется из всего множества продукций. В качестве имени может выступать

какое-либо словосочетание или порядковый номер продукции в их множестве, хранящемся в памяти системы.

Q – характеризует сферу применения продукции. Примерами сфер применения могут быть «роботизированный участок», «приготовление пищи» и другие. Разделение знаний на отдельные сферы позволяет экономить время на поиск нужных знаний.

Основным элементом продукции является ее ядро: $A \Rightarrow B$. Интерпретация ядра продукции может быть различной, но чаще всего звучит так: ЕСЛИ А, ТО В, более сложные конструкции ядра допускают в правой части альтернативный выбор, например: ЕСЛИ А, ТО В1, ИНАЧЕ В2.

Элемент Р – условие применимости. Обычно Р представляет собой логическое выражение. Когда Р принимает значение «истина» ядро активизируется. Если Р «ложно», то ядро не может быть использовано.

Элемент N описывает постуловия продукции. Это действия, которые выполняются только в том случае, если ядро продукции реализовалось.

Если в памяти системы хранится некоторый набор продукций, то они образуют систему продукций. В системе продукций должны быть заданы специальные процедуры управления продукциями, с помощью которых происходит активизация продукций и выбор для выполнения той или иной продукции из числа активизированных. Продукционные системы имеют давнюю историю. Их ограничение – в применении к крупномасштабным задачам, так как большое число правил приводит к замедлению скорости вывода. Кроме того, продукционная модель не имеет механизма вывода из тупиковых состояний в процессе поиска. Она продолжает работать, пока не будут исчерпаны все допустимые продукции.

Преимущества продукционной модели представления знаний:

- простота и гибкость выделения знаний;
- отделение знаний от программы поиска;
- модульность продукционных правил (правила не могут вызывать другие правила);
- простота пополнения и модификации;
- возможность эвристического управления поиском;
- возможность трассировки «цепочки рассуждений»;
- простота механизма вывода;
- независимость от выбора языка программирования;
- для ПК это простой и точный механизм использования знаний с высокой однородностью, описанных по одному синтаксису;
- продукционные правила являются правдоподобной моделью решения задачи человеком.

1.8.3. Семантические сети

В бытовом понимании семантика означает смысл слова, действия, художественного произведения и т. п. Семантическая сеть – это граф, вершинам которого сопоставляются понятия (объекты, процессы, явления), дуги графа – это отношения между вершинами. Объект – структура, содержащая конкретную информацию.

Метки вершин имеют ссылочный характер и представляют собой некоторые имена. В роли имен могут выступать, например, слова естественного языка. Метки дуг обозначают элементы множества отношений.

В семантических сетях используются три основных типа объектов: понятия, события, свойства.

Понятие – это сведения об абстрактных или физических объектах предметной области. Общие понятия интерпретируются как множество параметров или констант.

События – это действия, которые могут внести изменения в предметную область. Результатом события является некоторое новое состояние предметной области. Можно задать желаемое (целевое) состояние предметной области и поставить задачу отыскания в семантической сети последовательности событий, приводящей к новому состоянию.

Свойства – используются для уточнения понятий или событий. Для понятий это особенности или характеристики (цвет, размер, качество). Для свойств – продолжительность, время, место.

Объекты предметной области, отображаемые в семантической сети, можно разделить на три группы: обобщенные, индивидуальные (конкретные) и агрегатные.

Обобщенный объект соответствует некоторой собирательной абстракции реально существующего объекта, процесса или явления предметной области. Например, «изделие», «предприятие», «сотрудник» и т. п. Обобщенные объекты фактически представляют определенные классы предметной области.

Индивидуальный объект – это каким-то образом выделенный единичный представитель (экземпляр) класса. Например, «сотрудник Петров» И. Н.».

Агрегатным называется составной объект, образованный из других объектов, которые рассматриваются как его составные части. Например, изделие состоит из совокупности деталей, предприятие состоит из совокупности отделов, служб, цехов.

Введенная классификация является относительной. В зависимости от решаемой задачи один и тот же объект может рассматриваться как обобщенный или индивидуальный, как агрегатный или неагрегатный.

Возможные отношения в семантических сетях (не полный список):

Агент – это то, что (тот, кто) вызывает действие. Агент часто является подлежащим в предложении. Например, «Иванов ударил мяч».

Объект – это то, на что (на кого) направлено действие. В предложении объект часто выполняет роль прямого дополнения. Например, «Робот взял пирамиду».

Инструмент – это средство, которое используется агентом для выполнения действия. Например, «Иванов открыл дверь с помощью ключа».

Соагент – служит как подчиненный партнер главному агенту. Например, «Иванов сдал экзамен с помощью Петрова».

Пункт отправления и пункт назначения – это отправная и конечная позиция при перемещении агента или объекта. Например, «Робот переместился от одного станка к другому».

Траектория – это перемещение от пункта отправления к пункту назначения. Например, «Они прошли через дверь по ступенькам на лестницу».

Средство доставки – то в чем или на чем происходит перемещение. Например, «Иванов всегда едет домой на машине».

Местоположение – то место, где произошло (происходит, будет происходить) действие. Например, «Он работал за столом».

Потребитель – то лицо, для которого выполняется действие. Например, «Иванов собрал шпаргалки для Кати».

Сырье – это, как правило, материал из которого что-то сделано или состоит. Обычно сырье вводится предлогом «из». Например, «Иванов собрал робот из интегральных схем».

Время – указывает на момент совершения действия. Например, «Он закончил работу поздно вечером».

Под *фактом* понимают конкретизацию определенного отношения между объектами. В графической интерпретации факт – это подграф семантической сети, имеющий звездообразную структуру. Корень подграфа – вершина предикатного типа, помеченная уникальной меткой, включающей имя соответствующего отношения. Из вершины факта выходят ребра, помеченные именами атрибутов данного факта, ведущие в вершины базового множества, которые являются значениями этих атрибутов. Иногда удобна табличная форма записи фактов:

Метка факта	Отношение «меньше»
F1	1 → 2
F2	1 1
F3	0 → 2

Отношения могут быть самого различного вида, но чаще всего применяются следующие основные связи Р: «род-вид», «является представителем», «является частью». Наличие связи типа «род-вид» между обобщенными объектами А и В означает, что понятие А более общее, чем понятие В. Любой объект, отображаемый понятием В, отображается и понятием А, но не наоборот. Например, понятие «предприятие» – это родовое понятие для объекта «цех».

Связь «является представителем» существует обычно между обобщенным и индивидуальным объектом, когда индивидуальный объект выступает в роли представителя некоторого класса. Так, индивидуальный объект «студент Иванов» является представителем обобщенного объекта «студент».

Иерархические отношения между объектами системы имеют двусторонний характер: «сверху-вниз» и «снизу-вверх». Отношение «снизу-вверх» – это отношение типа «это-есть» (англ. – is-a). Например, скорость – *это-есть* кинематическая величина, кинематическая величина – *это-есть* механическая величина, механическая величина – *это-есть* физическая величина. Отношения «сверху-вниз» – это отношения типа «одним-из» (англ. kind-of).

Например, «одним-из» видов физических величин есть механические величины, «одним-из» видов механических величин есть кинематические величины, «одним-из» видов кинематических величин есть скорость.

В ряде случаев между связями «род-вид» и «является представителем» не делают различий, отмечая, что эти связи задают отношение «общее-частное», используя для формализации таких связей отношения АКО и is-a.

Не менее важно отношение «является частью» (англ. Part-of). Данное отношение связывает агрегатный объект с его составными частями. Оно позволяет отражать в базе знаний структуру объектов предметной области. Иногда данное отношение обозначают меткой «имеет». Такие отношения являются отношениями по горизонтали.

Такие отношения окружают нас повсеместно. В целое могут быть объединены самые разнообразные части. Известно, что многие объекты состоят из частей и могут быть расчленены на части. Но при этом существует общее свойство, заключающееся в том, что целое обладает признаками, которыми не обладает ни одна из его частей. Из частей состоит работающий двигатель, движущийся автомобиль и т. п. Вышесказанное в полной мере относится и к знаниям.

Пример семантической сети (рис. 1): Поставщик осуществил поставку изделий по заказу клиента до 1 июня 2008 г. в количестве 1000 шт.

Каким же образом осуществляется вывод на семантических сетях? Мощным средством вывода в семантических сетях является сопоставление с образцом. В данном случае происходит сопоставление отдельных фрагментов сети. При этом запрос к базе знаний представляется в виде автономного подграфа, который строится по тем же правилам, что и семантическая сеть. Поиск ответа на запрос реализуется сопоставлением подграфа с фрагментами семантической сети. Для этого осуществляется наложение подграфа запроса на соответствующий фрагмент сети. Успешным будет то наложение, в результате которого фрагмент сети оказывается идентичным подграфу запроса. При этом допускается использование в запросе переменных. Переменная запроса сопоставляется с константой фрагмента сети.

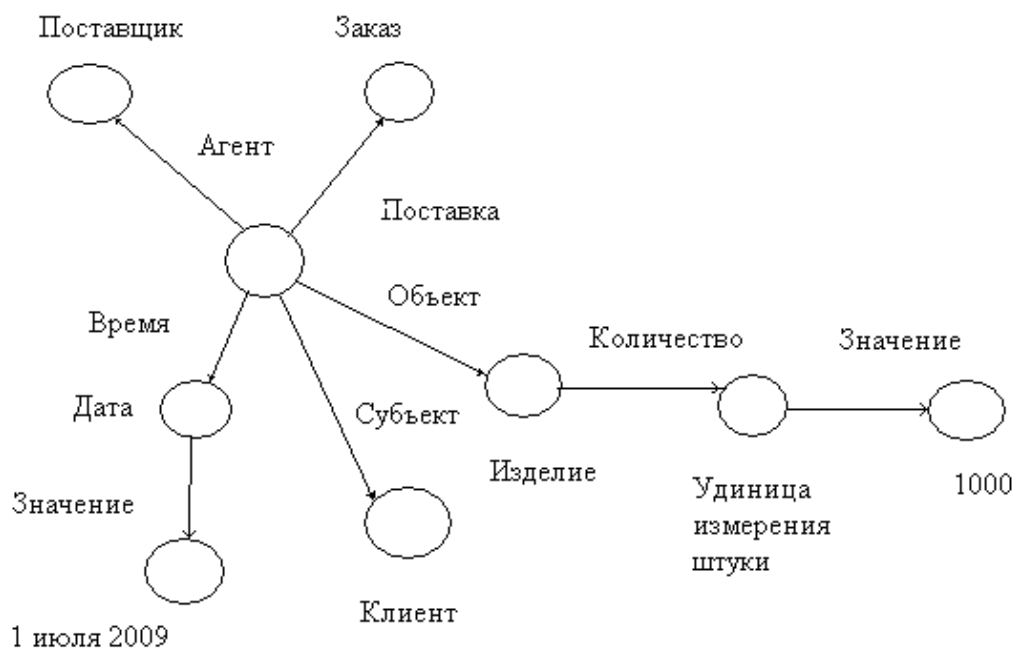


Рис. 1. Пример семантической сети

1.8.4. Классификация семантических сетей

По структуре

Сети простого типа – сети, у которых вершины не имеют собственной внутренней структуры.

Однородные сети – сети, имеющие одинаковые отношения между вершинами.

Сети иерархического типа – таким сетям свойственны структуры с вершинами разного ранга, имеющими разный уровень или подчиненность от низших к высшим.

По характеру отношений, приписываемых дугам сети:

Функциональные сети. Дуги отражают тот факт, что вершина, из которой выходит дуга, играет по отношению к вершине, куда идет дуга, роль аргумента. Описания, соответствующие вершине – функции, задают процедуру нахождения результата.

Сценарии – однородные сети, в которых в качестве единственного отношения выступает отношение нестрогого порядка (например, отношение «не раньше, чем»), которое допускает одновременность. Чаще всего эти отношения определяют все возможные последовательности событий.

Семантические сети делятся на интенциональные и экстенциональные.

Интенциональная сеть содержит интенциональные знания и описывает общую структуру модернизируемой предметной области на основе абстрактных объектов и отношений, т. е. обобщенных представителей некоторых классов объектов и отношений. Например, такие объекты как производст-

венный участок, груз, деталь могут являться обобщенными понятиями множества значений, от которых образуется множество имен конкретных производственных участков: (токарный, прессовый и т. п.), множество имен грузов (заготовка, кассета), множество классов деталей (болт, вал, гайка и т. п.).

Экстенциональная семантическая сеть описывает экстенциональные знания о модернизируемых объектах, являясь как бы «фотографией» его текущего состояния.

Семантические отношения делят на следующие четыре класса.

Лингвистические отношения – наиболее употребительные – падежные, к которым относятся, в частности, следующие: агент – отношение между событием и тем кто (что) его вызывает; объект – отношение между событием и тем, над чем производится действие; условие – отношение, указывающее логическую зависимость между событиями; инструмент – объект, с помощью которого совершается событие; место – место совершения события.

Другой тип лингвистических отношений – это характеристика глаголов и атрибутивных отношений. К характеристике глаголов относятся наклонение, время, род, число, залог. Атрибутивные отношения – цвет, форма, размер. Например: фраза «Большие красные шары» с использованием атрибутивных отношений может быть представлена структурой, изображенной на рис. 2.

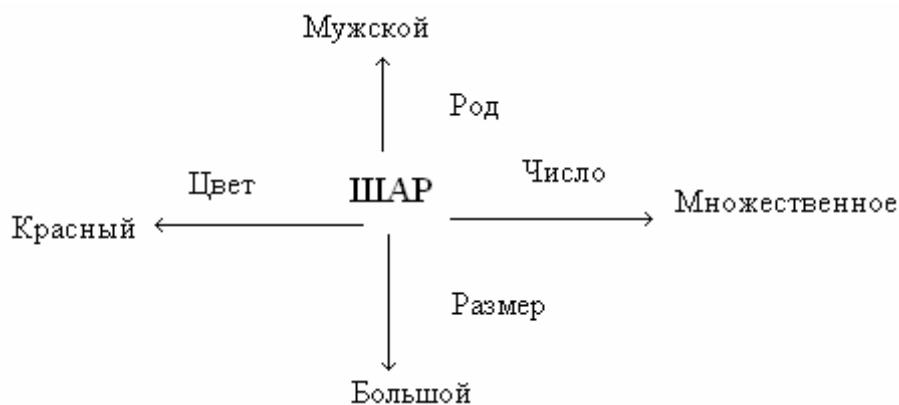


Рис. 2. Фраза «большие красные шары»

Логические отношения представляют собой операции алгебры логики.

Теоретико-множественные – это подмножество, супермножество, элемент множества, отношение части и целого и др.

Этот класс отношений используется для построения иерархических соподчиненных структур, для представления обобщенной информации (рис. 3).

Квантифицированные отношения – это логические кванторы общности и существования. Логические кванторы применяются для представления знаний декларативного типа. Например: «каждый станок требует

профилактического ремонта», «Существует робот А, который может обслуживать станки группы В».



Рис. 3. Иерархия классификации роботов

Важным понятием в семантических сетях является десигнат – уникальное внутрисистемное имя, которое ставится в соответствие некоторому объекту в предметной области, если о нем в данный момент времени нет полной информации. Например, «Станок С1 имеет накопитель». Это предложение имеет неопределенность относительно характеристик накопителя. По мере поступления информации будут уточняться емкость накопителя и другие данные.

(F1: имеет агент Станок объектD1)

(D1: имя накопитель)

(D1: габарит _____)

(D1: емкость _____)

F1 – метка факта, D1 – метка десигната.

В момент первого напоминания вводится информация, которая используется в дальнейшем. В семантических сетях информация может представляться в фреймовом виде.

При построении интеллектуальных банков данных на основе семантических сетей основным принципом их организации является разделение экстенциональных и интенциональных знаний, при этом экстенциональные семантические сети являются основой базы данных, а интенциональные сети – основой базы знаний (рис. 4).

Однородные семантические сети.

Частным случаем семантических сетей являются *сценарии*, или однородные семантические сети. *Сценарий* – это структурированное представление, описывающее стереотипную последовательность событий в частном контексте. Сценарии используются в системах понимания естественного языка для организации базы знаний в терминах ситуаций, которые система должна понимать. Объекты в этих сетях связаны единственным отношением строгого или нестрогого порядка с различной семантикой (например, «причина-следствие», «цель-подцель», «часть-целое»). Каждая последовательность действий в сценах обладает свойством казуальных

цепочек: всякое предшествующее действие создает условия для последующего действия.



Рис. 4. Формирование интеллектуальных банков на основе сетей

Сценарий включает следующие компоненты:

Начальные условия, которые должны быть истинными при вызове сценария.

Результаты или факты, которые являются истинными, когда сценарий завершается.

Предположения, которые поддерживают контекст сценария. Множество предположений описывают принятые по умолчанию условия реализации сценария.

Роли являются действиями, которые совершают отдельные участники сценария.

Сцены. Сценарий состоит из последовательности сцен, каждая из которых представляет временные аспекты сценария.

Сценарии рассматриваются как средство представления проблемно-зависимых казуальных знаний и задаются в виде фреймоподобных списочных структур.

сценарий: название

роли: список участников сценария.

цель: ключевое событие сценария, посылки и следствия ключевого события.

Сцена 1: наименование сцены
(ход сцены)

Сцена 2: наименование сцены
(ход сцены)

Сцена 3: наименование сцены
(ход сцены)

.....

Сцена N: наименование сцены
(ход сцены)

Особенности представления знаний в семантических сетях состоят в следующем:

– в семантической сети могут быть представлены такие виды объектов как понятия, события, специализированные методы решения. Увеличение номенклатуры объектов снижает однородность сети и приводит к необходимости увеличения набора механизмов и методов вывода.

– многомерность семантических сетей позволяет представить в них сложные семантические отношения, связывающие отдельные понятия, понятия и события в предложении, а также предложения в текстах.

– на каждой стадии формирования решения можно четко разделить полное знание системы (полная семантическая сеть) и текущее знание – возбуждаемый участок сети, в котором производятся некоторые операции (процесс вывода, понимания и т. п.).

Известны многие варианты семантических сетей, общие характеристики которых сводятся к следующему:

– описание объектов предметной области происходит на уровне естественного языка;

– все знания, включая вновь поступающие факты, а также специализированные методы решения, накапливаются в относительно однородной структуре памяти;

– определяется ряд более или менее унифицированных семантических отношений между объектами, которым ставятся в соответствие унифицированные методы вывода;

– запросы вместе с методами вывода определяют участки знания (семантической сети), имеющие отношение к поставленной задаче, фиксируя тем самым понимание запроса и вытекающую из него цепь выводов, соответствующих решению задачи; выводы в семантических сетях отличаются значительной полнотой.

Таким образом:

– семантические сети описывают отношения между объектами, которые задаются узлами сети;

– узлы обозначаются окружностями и имеют имена;

– отношения между узлами указываются связывающими их линиями;

– семантические сети можно использовать для создания структур или объектов;

– семантические сети можно использовать для создания правил баз знаний.

Совокупность модели представления знаний и связанных с ней процедур образуют систему представления знаний. База знаний и база данных рассматриваются как разные уровни информации, хранящейся в интеллектуальном банке информации. Системы управления базами знаний являются развитием систем управления базами данных.

Контрольные вопросы и задания

1. Модели представления знаний и их типы.
2. Декларативные и процедурные модели представления знаний (отличительные особенности).
3. Две формы информации: фактуальные знания и операционные знания.
4. Описания знаний об информации. Применение информационных моделей.
5. Классификация информационных моделей.
6. Элементарная единица структурного знания. Объектная модель. Триплет – предикат (Объект, Значение).
7. Представление знаний о предметной области в виде предикатных формул.
8. Исчисление предикатов первого порядка, основные аксиомы и правила логического вывода исчисления предикатов.
9. Метод резолюции и использование резолюционного вывода в исчислении предикатов.
10. Формальный метод представления знаний.
11. Модели знаний.
12. Классификация знаний.
13. Характеристики знаний и отличия знаний от данных.
14. Логическая модель (предикаты первого порядка).
15. Продукционная модель (эвристические правила вывода).
16. Семантическая сеть (знания в виде бинарных типизированных отношений между объектами).
17. Понятие фрейма, его структура, классификация фреймов. Типы значений слотов.
18. Виды отношений между фреймами. Наследование атрибутов во фреймовых системах.
19. Фреймовая модель (семантическая сеть).
20. Основная структурная единица фрейма.
21. Тип данных слота.
22. Модель данных.
23. Реляционная модель данных.
24. Сетевая модель данных.
25. Иерархическая модель данных.
26. Многомерные модели данных.
27. Объектно-реляционные модели данных.
28. Типы объектов и отношений в семантических сетях.
29. Понятие неточных знаний.
30. Графовые и гиперграфовые модели. И-ИЛИ графы.
31. Деревья. Поиск в глубину и в ширину. Поиск с возвратом. Поиск на основе стоимости дуг, эвристический поиск.

2. ОНТОЛОГИИ

2.1. Онтология. Основные понятия

Развитие наукоемких областей человеческой деятельности в современном обществе сопровождается возрастанием роли компьютерных технологий. С увеличением потока информации, появилась необходимость поиска новых способов ее хранения, представления, формализации и систематизации, а также автоматической обработки. Увеличение объема профессиональных знаний и высокая динамика развития информационных технологий порождает большое количество сетевых образовательных средств и продуктов. Таким образом, растет интерес к всеобъемлющим базам знаний, которые возможно использовать для различных практических целей. Огромный интерес вызывают системы, способные без участия человека извлечь какие-либо сведения из текста. Как результат, на фоне вновь возникающих потребностей развиваются новые технологии, призванные решить заявленные проблемы. Наряду с World Wide Web появляется его расширение, SemanticWeb, в котором гипертекстовые страницы снабжаются дополнительной разметкой, несущей сведения о семантике включаемых в страницы элементов. Неотъемлемым компонентом SemanticWeb является понятие онтологии, описывающее смысл семантической разметки.

Онтологии используются для формальной спецификации понятий и отношений, которые характеризуют определенную область знаний. Преимуществом онтологий в качестве способа представления знаний является их формальная структура, которая упрощает их компьютерную обработку.

Термин «онтология» используется в нескольких областях знания и имеет два различных значения:

- «Философская дисциплина, которая изучает наиболее общие характеристики бытия и сущностей»
- «Онтология – артефакт, структура, описывающая значения элементов некоторой системы»

Термин «онтология» имеет долгую историю в философии, где он еще со времен Аристотеля обозначал науку о бытии. Предметом онтологии выступало само по себе сущее, в рамках этого раздела философии выделялись базовые категории и общие свойства, типы сущностей. Как упоминают С. Ниренбург и В. Раскин, в качестве синонима термина «онтология» в его философском понимании часто используется термин «метафизика».

Понятие онтологии в инженерной области знания рождает большое количество дискуссий, в которых каждый автор стремится предложить свое определение. Отчасти это можно объяснить новизной области исследования, с другой стороны, разнообразием практических задач, решаемых с использованием онтологий.

В современных информационных технологиях наиболее часто упоминается и используется определение онтологии, сформулированное Н. Грубером: «Онтология – это спецификация концептуализации». Эта дефиниция является своеобразным обобщением, формальной интерпретацией многих других определений. Центральным в нем является понятие «концептуализация», которое было введено в работе. Основная сложность заключается в том, что термин «концептуализация» имеет разнообразные контексты употребления, поэтому данный термин вызывает разногласия.

Как поясняет Н. Гуарино, этот термин в общих чертах интуитивно понятен, и его четкая формулировка не дается при обсуждении понятия «онтология». Стоит пояснить, что под «концептуализацией» понимается строгое описание системы понятий, объектов и других сущностей и отношений, связывающих их друг с другом. Можно сказать, что концептуализация – это абстрактное, упрощенное видение мира, который мы хотим представить для каких-то целей. Таким образом, концептуализация расчленяет какую-либо область знаний, существующую в целостном виде, выделяет из этой области отдельные объекты, а затем формулирует отношения, свойственные для данной области. Основная часть формально представленного знания базируется на концептуализации, каждая база знаний или система, основанная на знаниях, явно или неявно связывается с какой – то концептуализацией.

Однако определение, связанное с понятием «концептуализации», является далеко не единственным. В литературе можно также найти определение онтологии как «теории того, какие сущности могут существовать в уме хорошо осведомленного (knowledgeable) агента». Данное определение выявляет другой подход к этому понятию. Эта формулировка позволяет включать в онтологию набор понятий, но не дает возможность задавать их структуру. Этот пробел в определении особенно значим, так как в рамках искусственного интеллекта в качестве синонима онтологии часто используется понятие «терминология», а в ней структура безусловно содержится.

Объединением выше упомянутых дефиниций становится определение из работы, вводящее «онтологию», как «спецификация концептуализации на уровне эксплицитных знаний, зависящее от предметной области или задачи, для которой она предназначена». Таким образом, онтология зависит от определенной точки зрения, однако, как раз степень этой зависимости является определяющим фактором для возможности ее многократного использования. А ведь именно в возможности многократного использования онтологий видится их значимость и ценность.

Н. Гуарино упоминает еще один подход к определению «онтологии»: «Онтология – это конкретный артефакт, созданный для выражения значений, подразумеваемых у совместно используемой лексики». Здесь упоминается, что онтология предоставляет средства для передачи подразумеваемого значения.

Н. Гуарино предлагает также определять онтологию, как «логическую теорию, которая ограничивает подразумеваемые модели логического языка». Таким образом, под онтологией понимается нечто большее, чем просто детализированный набор понятий и отношений. В онтологию включаются и ограничения, накладываемые на отношения в рамках данной области. Это некоторый набор аксиом, который строится на базе понятий и отношений между ними. Таким образом, например, в рамках искусственного интеллекта можно описать онтологию программы, определив множество объектов и связав их с описаниями, а также введя формальные аксиомы, которые ограничивают интерпретацию и совместное употребление этих терминов.

Формально онтологию можно назвать формулировкой логической теории, некоего исчисления со своими правилами. Эта теория позволяет систематизировать категории действительности и/ или выражаемые в языке значения. Следовательно, такое определение онтологии можно считать более широким взглядом на данное понятие, нежели предыдущие.

В качестве рабочего определения, наиболее приспособленного для целей компьютерной обработки информации, можно взять дефиницию, «Онтология – это базы знаний специального типа, которые могут «читаться» и пониматься, отчуждаться от их разработчика и/ или физически разделяться их пользователями».

2.2. Цели создания онтологий

Совместное использование людьми или программными агентами общего понимания структуры информации является одной из наиболее общих целей разработки онтологий.

Обеспечение возможности повторного использования знаний предметной области стало одной из движущих сил недавнего всплеска в изучении онтологий. Например, для моделей многих различных предметных областей необходимо сформулировать понятие времени. Это представление включает понятие временных интервалов, моментов времени, относительных мер времени и т. д. Если одна группа ученых детально разработает такую онтологию, то другие могут просто повторно использовать ее в своих предметных областях. Кроме того, если нам нужно создать большую онтологию, мы можем интегрировать несколько существующих онтологий, описывающих части большой предметной области.

Создание явных допущений в предметной области, лежащих в основе реализации, дает возможность легко изменить эти допущения при изменении наших знаний о предметной области. Жесткое кодирование предположений о мире на языках программирования приводит к тому, что приложение быстро устаревает, что значительно сокращает его время жизни.

Отделение знаний предметной области от оперативных знаний – это еще один вариант общего применения онтологий. Мы можем описать задачу конфигурирования продукта из его компонентов в соответствии с требуемой спецификацией и внедрить программу, которая делает эту конфигурацию независимой от продукта и самих компонентов. После этого мы можем разработать онтологию компонентов и характеристик ЭВМ и применить этот алгоритм для конфигурирования нестандартных ЭВМ. Анализ знаний в предметной области возможен, когда имеется декларативная спецификация терминов. Формальный анализ терминов чрезвычайно ценен как при попытке повторного использования существующих онтологий, так и при их расширении.

2.3. Структура онтологии

В общем виде структура онтологии представляет собой набор элементов четырех категорий:

- понятия;
- отношения;
- аксиомы;
- отдельные экземпляры;

Понятия рассматриваются как концептуализации класса всех представителей некой сущности или явления (например, Животное, Чувство). *Классы* (или понятия) являются общими категориями, которые могут быть упорядочены иерархически. Каждый класс описывает группу индивидуальных сущностей, которые объединены на основании наличия общих свойств.

Отношения связывают понятия. Самым распространенным типом отношений, используемым во всех онтологиях, является отношение категоризации, то есть отнесение к определенной категории. Этот тип отношений имеет ряд других названий, встречающийся в различных исследованиях:

- таксономическое отношение;
- отношение IS-A;
- класс – подкласс;
- лингвистика: гипоним – гипероним;
- родовидовое отношение;
- отношение a-kind-of.

Аксиомы задают условия соотнесения категорий и отношений, они выражают очевидные утверждения, связывающие понятия и отношения. Под аксиомой можно понимать утверждение, вводимое в онтологию в готовом виде, из которого могут быть выведены другие утверждения.

Они позволяют выразить ту информацию, которая не может быть отражена в онтологии посредством построения иерархии понятий и установки различных отношений между понятиями. В качестве примера аксиомы можно привести следующее высказывание: «Если X смертен, то X когда-нибудь умрет». Аксиомы позволяют в дальнейшем осуществлять умозаключения в рамках онтологии. Они могут снабжать исследователей информацией о правилах, позволяющих автоматически добавлять информацию. Аксиомы могут также представлять собой ограничения, накладываемые на какие-либо отношения, делающие возможным выведение умозаключений. Приведем несколько примеров таких ограничений. Понятийные ограничения указывают на то, какой тип понятий может выражать данное отношение (например, свойство Цвет может выражаться только понятиями категории Цвет). Примером числовых ограничений является утверждение того, что для Человека количество биологических родителей равно 2. Количество и степень детализации аксиом обычно зависят от типа онтологии.

Наряду с указанными элементами онтологии в нее также входят так называемые «экземпляры». В литературе они могут выступать также под названиями:

- конкретные экземпляры;
- инстанции;
- индивидуальные экземпляры.

Экземпляры – это отдельные представители класса сущностей или явлений, то есть конкретные элементы какой-либо категории (например, экземпляром класса Человек будет королева Виктория).

Составляющие онтологии подчиняются своеобразной иерархии. На нижнем уровне этой иерархической лестницы находятся экземпляры, конкретные индивиды, выше идут понятия, то есть категории. На уровень выше располагаются отношения между этими понятиями, а обобщающей и связующей является ступень правил или аксиом.

Термину «онтология» удовлетворяет широкий спектр структур, представляющих знания о той или иной предметной области». Так, к онтологиям можно отнести ряд структур, отличающихся разной степенью формализованности:

- 1) глоссарий;
- 2) простая таксономия;
- 3) тезаурус (таксономия с терминами);
- 4) понятийная структура с произвольным набором отношений;
- 5) полностью аксиоматизированная теория.

Однако в этих структурах не всегда представлены все составляющие онтологии, которые описывались в данном разделе.

2.4. Классификация онтологий

Онтологии сильно различаются по ряду параметров, и исследователи выделяют различные основания для их классификации. Так Э. Хови говорит, что онтологии различаются в зависимости от набора элементов, содержащихся в них, а также типов вводимых отношений. Он выделяет так называемые «*терминологические онтологии*» и «*настоящие онтологии*». Под первыми Э. Хови понимает онтологии, включающие сущности, явления, свойства, связи предметной области и объединяющие их структурные отношения. «Настоящие» же онтологии включают в себя также дефиниционные отношения и отношения дополнительной информации. Наряду с этим в них входят аксиомы, определяющие взаимозависимости между отношениями и понятиями.

Э. Хови выстраивает подробную классификацию различных характеристик онтологий. Он упоминает, что основными параметрами могут быть: форма (то, как формируется онтология), содержание, а также средства использования онтологий.

Разбиение онтологий по количеству и качеству понятий, включаемых в них:

1. *Онтологии верхней зоны* обычно насчитывают примерно 100–500 концептов. В них включены наиболее абстрактные категории, обладающие свойством универсальности. Они являются базовым разбиением наблюдаемой действительности на категории. Обычно они строятся теоретиками, философами. Зачастую концепты даже не лексикализуются. Составление аксиом в данном типе онтологий с высоким уровнем обобщения достаточно сложно и требует некоторого воображения. Преимуществом таких онтологий является возможность их использования во многих областях и даже во многих языках. Для данного рода онтологий характерен ограниченный набор обобщенных отношений, которые можно отнести к базовым (таких как родовидовые отношения, отношения часть-целое и ассоциативные отношения). В таких онтологиях типичными на верхнем уровне разбиения являются такие понятия, как:

- сущность;
- явление;
- объект;
- процесс;
- роль.

Однако этот типичный набор может быть представлен в усеченном виде, например, в онтологии MicroKosmos, разработанной С. Ниренбургом и В. Раскиным на верхнем уровне появляются лишь три категории понятий: «объект», «процесс» и «роль». При этом авторы претендуют на универсальность этого разбиения для онтологической семантики в целом и, таким образом, для всех онтологий верхнего уровня.

2. *Онтологии средней зоны, здесь элементов обычно уже больше (500–100 000 концептов). Они представляют мир в целом и в общем случае это неаксиоматизированная область.* Сложность заключается в том, что для данного вида онтологий требуется выводить слишком большое количество аксиом. Обычно выходом является использование методов автоматизированного вывода аксиом из уже существующих онтологий. Построением онтологий этого уровня чаще всего занимаются когнитологи и лингвисты.

3. *Онтологии нижней зоны (предметной области) наиболее обширные, обычно они насчитывают около 200–2 000 концептов.* Они описывают конкретные предметные области с их спецификой. При этом круг решаемых задач и вопросов, на которые онтология отвечает, ограничен выбранной областью. Для данного типа онтологий характерно наличие отношений, специфичных для конкретной области. Это высоко аксиоматизированная зона, то есть для нее возможно построение большого количества аксиом и правил. В большинстве случаев этот тип онтологий строится экспертами области знания или при их содействии. В связи с большой спецификой каждой отдельной предметной онтологии ее повторное использование зачастую возможно только в рамках предметной области.

Наряду с описанным делением все онтологии могут быть разделены на глубинные и поверхностные.

Поверхностные онтологии (легковесные) строятся на поверхностной семантике, они определяют понятия через значения слов. Однако здесь возникает проблема, какое количество смыслов выделять для каждого слова. *Глубинные онтологии используют глубинную семантику.*

Б. Й. Вилинга и А. Т. Шрайбер выделяют два измерения для оценки онтологий: «объем и тип структуры концептуализации и предмет концептуализации». Однако, как указывает Н. Гуарино, критерий для выделения в рамках первого измерения информационных, терминологических онтологий и онтологий, моделирующих знания, не отличается четкостью. Проще разделить онтологии по степени детальности, используемой для характеристики концептуализации:

Очень детализированная онтология подробно специализирует подразумеваемую концептуализацию, но платой за это оказывается более громоздкий язык, который может быть сложно применять на практике.

Простая онтология может развиваться с допущением каких-то скрытых условий, которые подразумеваются создателями, и ее могут использовать те, кто уже договорился о лежащей в основе концептуализации (и осознает эти допущения). Можно различать «отсылочные (также называемые оффлайн) онтологии» (reference ontologies) и «осуществляемые (совместно используемые, онлайн) онтологии» (implemented (shareable) ontologies). Несложная структура, описывающая, например, лексикон, может помещаться онлайн, в то время как замысловатые теории,

определяющие значение терминов из лексикона, могут находиться оффлайн.

Основные виды онтологий и их характеристики приведены в табл. 1, которая может использоваться как классификационная.

Таблица 1

Классификационная таблица онтологий

Виды онтологий	Характеристики				
	Содержание	Источник знаний	Структура представленных знаний	Формальная структура на языке знаний в электронном виде	Декларативный компонент системы
Онтологии верхнего уровня	Терминосистемы определенной области знаний	Терминологический словарь	Иерархия понятий в виде семантической сети	Семантическая сеть фреймов	Совокупность словарных статей
Онтологии предметных областей	Категорийно-понятийный аппарат предметной области	Учебники, учебные пособия и монографии	Иерархия понятий в виде семантической сети	То же	То же
Онтологии задач	Методы решения конкретных задач или деятельности	Аналитик, спецификации структур и методов обработки данных	База продукционных правил	Представление методов решения задач как множеств возможных ситуаций	База продукционных правил
Онтологии – приложения	Правила вывода	Онтология предметных областей и онтологии задач	Семантические сети и база продукционных правил	Система доступа в виде системы вывода, основанной на знаниях	

С точки зрения предмета концептуализации исследователи выделяют:

- 1) *прикладные онтологии,*
- 2) *онтологии области знания,*
- 3) *общие (родовые) онтологии и репрезентационные онтологии* (речь идет об онтологиях метауровня, включающих в себя репрезентационные первоэлементы).

Онтологии могут быть также разделены на одноязычные и многоязычные. Уже существует ряд онтологий, ориентированных на представление знаний на нескольких языках, например, EuroWordNet, MikroKosmos

и некоторые другие. Сложность создания таких онтологий обычно заключается в том, что возможно наличие различий в понятийных системах разных языков.

Различаются онтологии по количеству экземпляров:

- *безэкземплярные*;
- *экземплярные*.

Как понятно из названия данного типа онтологий, безэкземплярные онтологии отличаются отсутствием конкретных экземпляров. На нижних уровнях иерархии таких онтологий находятся не конкретные экземпляры, а понятия. Эта особенность онтологии накладывает некоторый отпечаток и на вводимые в данной онтологии отношения.

Таким образом, существует множество подразделений онтологий, но эти классификации не всегда бывают достаточно четкими и последовательными.

Основными составляющими онтологии являются:

Классы или понятия используются в широком смысле. Понятием может быть любая сущность, о которой может быть дана какая – либо информация. Классы – это абстрактные группы, коллекции или наборы объектов. Они могут включать в себя экземпляры, другие классы, либо же сочетания и того, и другого. Классы в онтологиях обычно организованы в таксономию – иерархическую классификацию понятий по отношению включения.

Компоненты, из которых состоят онтологии, зависят от парадигмы представления. Но практически все модели онтологий, в той или иной степени, содержат концепты (понятия, классы, сущности, категории), свойства концептов (слоты, атрибуты, роли), отношения между концептами (связи, зависимости, функции) и дополнительные ограничения (определяются аксиомами, в некоторых парадигмах фасетами).

Экземпляр – этот термин используется для представления элементов в предметной области, т. е. элемента данного концепта. Экземпляр может являться классом онтологии с полностью определёнными свойствами. Онтология вместе с множеством отдельных экземпляров составляет базу знаний.

Отношения представляют тип взаимодействия между понятиями предметной области. Формально n-арные отношения определяются как подмножество произведения n множеств: $R \subseteq C_1 \times C_2 \times \dots \times C_n$. Пример бинарного отношения – отношение ЧАСТЬ-ЦЕЛОЕ. Отношения тоже могут быть организованы в таксономию по включению; например, отношения *быть_отцом_для* и *быть_матерью_для* на множестве людей содержатся в отношении *быть_родителем_для*, которое в свою очередь содержится в отношении *быть_предком_для*.

Функции – это специальный случай отношений, в которых n-й элемент отношения однозначно определяется n-1 предшествующими элемен-

тами. Формально функции определяются следующим образом: $F: C_1 \times C_2 \times \dots \times C_{n-1} \rightarrow C_n$.

Аксиомы используются, чтобы записать высказывания, которые всегда истинны. Они могут быть включены в онтологию для разных целей, например, для определения комплексных ограничений на значения атрибутов, аргументы отношений, для проверки корректности информации, описанной в онтологии, или для вывода новой информации.

В качестве примера того, что в рамках онтологий понимается под аксиомами, можно привести следующее положение и его формальную запись на языке исчисления предикатов первого порядка:

Работник, являющийся руководителем проекта, работает в проекте.

К множеству аксиом относятся ограничения, накладываемые как на классы, так и на отношения.

2.5. Жизненный цикл онтологии

Понятие «жизненный цикл», вызывает множество вопросов – как и любое другое важное понятие, в конечном итоге по-разному определяющееся разными заинтересованными сторонами в силу особенностей их деятельности, и поэтому (как и любое другое понятие, в отличие от «просто термина») позволяющее наладить коммуникацию между носителями онтологий разных предметных областей.

ISO 15288:2008 (определения):

Жизненный цикл – эволюция системы, продукции, услуги, проекта или иного рукотворного объекта от замысла до прекращения использования.

Описание жизненного цикла – возможно организованный в стадии набор практик и мероприятий, связываемый с жизненным циклом, также служащий общим первоисточником для коммуникации и понимания.

ISO TR 24748 (подраздел 3.2.1) пишет об этом так:

Каждая система, вне зависимости от ее вида и масштаба, проходит жизненный цикл согласно некоторому описанию от своего изначального замысла до окончательного прекращения использования. Продвижение системы по частям этого описания, в какой бы то ни было последовательности и каким бы то ни было образом, называется жизненным циклом системы. Описание жизненного цикла, таким образом, – это концептуальная сегментация определения потребности в системе, ее реализации в виде продукции или услуги и ее использования, эволюции и вывода из эксплуатации. Описание жизненного цикла обычно сегментировано по стадиям, способствующим планированию, разворачиванию, эксплуатации и поддержке целевой системы. Такие сегменты дают упорядоченное продвижение системы через установленные пересмотры выделения ресурсов, что

снижает риски и обеспечивает удовлетворительное продвижение. Основной причиной применения описаний жизненного цикла является потребность в принятии решений по определенным критериям до продвижения системы на следующую стадию.

Специалисты по управлению знаниями сходятся в следующем понимании этапов жизненного цикла онтологии (рис. 5):

- 1) построение,
- 2) заполнение,
- 3) проверка,
- 4) использование,
- 5) поддержка
- 6) развитие.

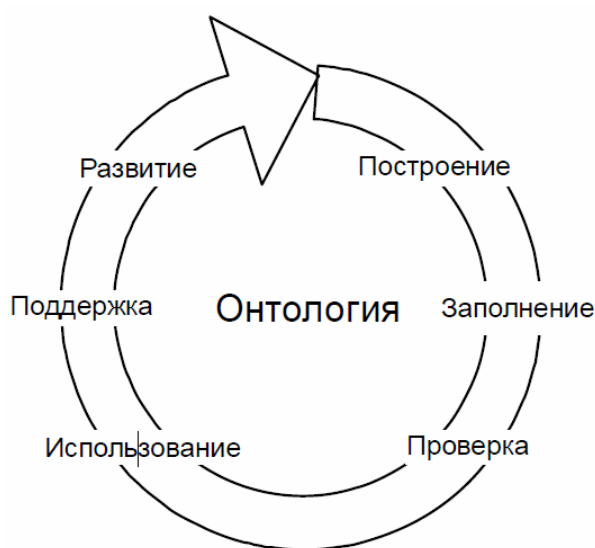


Рис. 5. Жизненный цикл онтологии

2.6. Существующие онтологические ресурсы

Как упоминалось ранее, на данном этапе существует ряд разнообразных онтологических ресурсов. Эти онтологии, противопоставляемые по направленности, отличаются друг от друга по форме и содержанию. Опишем некоторые из онтологий, соотнося их с ранее описанными классами.

Онтологии верхней зоны.

1. Онтологическая структура, созданная под руководством Э. Хови, которая включает 400 записей и описывает наиболее общее разбиение понятий по категориям.

2. Penman Upper Model, состоит всего лишь из 300 элементов. В ней нет лексических единиц и аксиом. Ее записи представляют понятийно-грамматические классы, выступающие как связующее звено между языком

и реальным миром. Между классами верхней модели и синтаксическими типами установлены связи. Эта модель, протестированная для грамматик разных языков и машинного перевода, может быть использована при построении и координации онтологий областей знаний.

3. Онтология DOLCE. Она была создана под руководством Н. Гуарино в Италии, в институте LADSEB. Это поверхностная модель, обладающая очень высокой степенью абстрактности. В ней содержатся примерно 500 понятий и нет лексических элементов. Описание производится с помощью языков XML или RDF, оно очень четкое и последовательное. Однако в данной онтологии малая степень иерархичности и все выглядит как верхние уровни. При конвертации этой онтологии в другие форматы могут возникать проблемы, так как пропадает часть функциональности, теряются некоторые аксиомы и ограничения.

4. Онтология, разработанная Дж. Совой (Sowa'sKROntology), также относится к категории структур верхней зоны. Эта онтология базируется на структурах, разработанных философами. Главной целью, преследуемой ее автором, являлось создание основы, в которую могут быть включены все остальные онтологии или модели их верхних зон. Структурная решетка этой онтологии представляет собой 12 понятий, являющихся всеми комбинациями трех отличительных признаков:

- первичность/ вторичность/ третий порядок,
- материальный/ абстрактный,
- длительный/ мгновенный (continuant/ occurrent) (объект/ процесс).

Составление таких решеток помогает избежать проблемы очередности использования дифференциальных признаков при делении понятий.

В онтологии Дж. Совы используются также некоторые стандартные ситуативные роли:

- детерминирующий участник (участник, определяющий направление процесса, то есть инициатор или цель);
- неотъемлемый участник (присутствует на протяжении всего процесса, но активно не контролирует происходящее);
- источник (должен присутствовать в начале процесса, но не обязан принимать участие во всем процессе);
- продукт (может появляться в конце процесса, но не обязан принимать участие во всем процессе).

Онтологии средней зоны

1. К онтологиям средней зоны можно отнести Mikrokosmos, созданную в Университете Нью-Мексико. Она содержит около 6 000 записей. Онтология Принстонского университета WordNet на несколько порядков больше, она состоит из 110 000 записей, в которых в большинстве случаев отражена конкретная, нетерминологическая лексика. В ее рамках заданы

различные элементы с их значениями, объединенные в группы (синсеты) по сходству значений.

2. CYC (<http://www.openyc.org/>) является одной из самых крупных и разработанных онтологий. Она направлена на использование в рамках искусственного интеллекта, в ней содержится большое количество (более миллиона) аксиом. Версия ResearchCYC, преобразованная в формат RDF, не включает вторичных понятийных выражений, и таким образом, в ней пропала часть родовых понятий высшего уровня. По мнению Э. Хови, проблемой использования этой онтологии является запутанность ее структуры, она очень сложна для понимания.

3. Suggested Upper Model Ontology (SUMO (<http://www.ontologyportal.org/>)). Она и включенные в нее онтологии предметных областей представляют собой самую большую общедоступную онтологию, существующую на сегодняшний день. SUMO состоит из 20 000 элементов и 60 000 аксиом. В нее включаются сами разработки SUMO, онтология среднего уровня (MIL0) и набор отраслевых онтологий (в сфере коммуникаций, транспорта, географии и многих других). Помимо этого она содержит большое количество аксиом, причем все ее элементы формально определены и это описание не зависит от конкретной системы осуществления выводов.

4. Онтология SENSUS также разрабатывалась как структура, в которую может быть включена дополнительная информация. Она содержит 70 000 записей и является расширением и реорганизацией WordNet; на верхнем уровне добавлены записи из Penman Upper Model и ветви WordNet переорганизованы.

5. Omega представляет собой среду для различных онтологий и ресурсов. Она содержит примерно 120 600 понятий и слов, в нее включены WordNet, Mikrokosmos, Penman Upper Model, а также дополнительные элементы. Это не одноязычная структура, в ней есть как английская и испанская лексика. Кроме того, в Omega содержится множество различных ситуативных ролей (около 13 000), взятых из ресурсов Framenet, WordNet, LCS и PropBank. Эта среда обладает обширной базой отдельных экземпляров: 1,1 миллион персоналий, 765 000 фактов, 5,7 миллионов местоположений. Omega включает огромное количество (более 28 миллионов) утверждений о понятиях, отношениях и отдельных экземплярах. Она представляет собой ресурс с обширными возможностями построения, изменения и пополнения онтологий. Этот ресурс является библиотекой различных онтологий, позволяющей использовать объединять онтологии разных уровней для использования в какой-то прикладной задаче. Это далеко не единственный ресурс, предоставляющий доступ к многочисленным отдельным онтологиям. Также существуют ресурсы, объединяющие онтологии, описанные с помощью какого-то языка представления, например, библиотека онтологий DAML (<http://www.daml.org/ontologies/>). Создатели онторедкторов (программ, упрощающих создание онтологий) предлагают

своим пользователям самостоятельно формировать библиотеки онтологий, добавляя результаты своей работы прямо в специально созданный ресурс. Зачастую онтологии, содержащиеся в таких библиотеках, не отличаются общностью тематики. Однако такие ресурсы дают возможность не строить онтологии «с нуля», а использовать уже существующие наработки, находя все онтологии на одной ресурсе. К таким библиотекам можно отнести библиотеку, сформированную пользователями онторедатора Protege (ресурсы представлены на сайте http://protegewiki.stanford.edu/index.php/Protege_Ontology_Library), или Ontolingua (<http://www.ksl.stanford.edu/software/ontolingua/>), систему, разработанную в Стэнфордском университете, которая обеспечивает распределенную совместную среду для просмотра, создания, редактирования, модификации и использования онтологий.

К библиотекам онтологий относится и Onto Select (<http://olp.dfki.de/ontoselect/>). Этот ресурс «отслеживает» появляющиеся онтологии в Сети, аннотирует их, предоставляя пользователям возможность в дальнейшем эффективно искать онтологии по интересующей тематике и в дальнейшем использовать их в своих целях. Также этот ресурс предоставляет возможность добавлять свою онтологию в библиотеку. На данном ресурсе можно найти ссылки на онтологии, описанные с помощью разных языков представления, так это и OWL, DAML, RDFS и другие. В Onto Select представлены онтологии, созданные для разных языков, однако преимущественно все ресурсы разработаны для английского языка.

2.7. Модели онтологий

Формальные модели.

Упрощенная модель онтологии.

Модель онтологии (без словаря и без определения типа отношений) приведена в трудах Т. А. Гавриловой, Ф. В. Хорошевского. Под онтологией понимается упорядоченная тройка вида

$$O = \langle C, R, F \rangle,$$

где C – конечное множество концептов (понятий, терминов) предметной области, которую представляет онтология O ; R – конечное множество отношений между концептами заданной предметной области; F – конечное множество функций интерпретации (аксиоматизации), заданных на концептах и/или отношениях онтологий O .

Естественным ограничением, налагаемым на множество C , является его конечность и непустота. Иначе обстоит дело с множествами R и F . При этом R и F тоже должны быть конечными множествами.

Пусть $R = \emptyset$ и $F = \emptyset$. Тогда онтология O трансформируется в простой словарь: $O = V = \langle C, \{\}, \{\} \rangle$.

Такая вырожденная онтология может быть полезна для спецификации, пополнения и сопровождения словарей. Однако онтологии-словари имеют ограниченное применение, поскольку не вводят эксплицитного смысла терминов. В отдельных случаях, когда используемые термины принадлежат очень узкому (например, техническому) словарю и их значение устоялось в пределах некоторого профессионального сообщества, такие онтологии также могут применяться на практике.

Иная ситуация возникает при использовании терминов естественного языка или когда общаются программные агенты. В этом случае необходимо характеризовать смысл элементов словаря с помощью подходящей аксиоматизации с целью исключения неоднозначной интерпретации всеми пользователями онтологии.

Другой вариант соответствует случаю, когда $R = 0$, но $F \neq 0$. В этом случае каждому элементу множества терминов C может быть поставлена в соответствие функция интерпретации $f \in F$. Формально это может быть записано следующим образом:

Пусть $C = C1 \cup C2$, причем $C1 \cap C2 \neq 0$, где $C1$ – множество интерпретируемых терминов; $C2$ – множество интерпретирующих терминов.

Тогда $\exists(x \in C1, y^1, y^2, \dots, y^k \in C2)$, такие, что $x = f(y^1, y^2, \dots, y^k)$, где $f \in F$.

Пустота пересечения $C1$ и $C2$ исключает циклические интерпретации, а введение в рассмотрение функции k аргументов призвано обеспечить более полную интерпретацию. Вид отображения f из F определяет выразительную мощь и практическую полезность этого вида онтологии. Например, если предположить, что функция интерпретации задается оператором присваивания значений ($C1 := C2$, где $C1$ – имя интерпретации $C2$), то онтология в данном случае трансформируется в пассивный словарь V^P :

$$O = V^P = \langle C1 \cup C2, \{\}, \{:=\} \rangle.$$

Такой словарь пассивен, так как все определения терминов и $C1$ берутся из уже существующего, фиксированного и явно прописанного множества $C2$. Практическая ценность его выше, чем просто словаря V , но явно недостаточна, например, для представления знаний в обработке информации в Интернет в силу мультипредметного характера этой среды. Ценность словаря возрастет, если часть интерпретирующих терминов из множества $C2$ будет задаваться процедурно, а не декларативно.

То есть их смысл будет «вычисляться» при каждой интерпретации. Для создания онтологии, которой по силам решение разнообразных задач обработки информации, можно отказаться от предположения $R = 0$.

Вначале определен специальный подкласс онтологий – простая таксономия:

$$O = T^0 = \langle C, \{is-a\} \rangle.$$

Под таксономической структурой понимается иерархическая система понятий, связанных между собой отношением *is-a* (быть элементом класса).

Отношение *is-a* имеет априори фиксированную семантику и позволяет строить структуру понятий онтологии типа дерево.

Продолжая цепь рассуждений, можно обобщить частные случаи онтологий таким образом, чтобы обеспечить:

- представление множества концептов C в виде сетевой структуры;
- использование расширенного множества отношений R , включающего не только таксономические, но и отношения, отражающие специфику конкретной предметной области, плюс средства расширения множества R ;
- использование декларативных и процедурных интерпретаций и отношений, включая возможность построения новых интерпретаций.

Полная модель.

Добавим в рассмотрение множество типов отношения и словарь онтологии, содержащий набор знаков для бинарных отношений и набор знаков (лексических единиц) для понятий (концептов) онтологии.

В общем виде формальная модель онтологии может быть описана следующим кортежем:

$$O = \{L, C, F, G, H, R, A\},$$

где $L = L^C \cup L^R$ – словарь онтологии, содержащий набор лексических единиц (знаков) для понятий L^C и набор знаков для отношений L^R ; C – набор понятий онтологии, причем для каждого понятия $c \in C$ в онтологии существует по крайней мере одно утверждение; F и G – функции ссылок такие, что $F: F^{LC} \rightarrow 2^C$ и $G: F^{LR} \rightarrow 2^R$. То есть F и G связывают наборы лексических единиц $\{L_j\} \subset L_c$ наборами понятий и отношений, на которые они соответственно ссылаются в данной онтологии. При этом одна лексическая единица может ссылаться на несколько понятий или отношений и одно понятие или отношение может ссылаться на несколько лексических единиц. Инверсиями функций ссылок являются F^{-1} и G^{-1} ; H – фиксирует таксономический характер отношений (связей), при котором понятия онтологии связаны нереклексивными, ациклическими, транзитивными отношениями $H \subset C \times C$.

Выражение $H(C_1, C_2)$ означает, что понятие C_1 является подпонятием C_2 ; R – обозначает бинарный характер отношений между понятиями онтологии, фиксирующие пары области применения (*domain*)/ области значений (*range*), то есть пары (D, R) с $D, R \in C$; A – набор аксиом онтологии.

2.8. Методы построения онтологий

Существует общий укрупненный алгоритм построения:

- 1) перечислить категории, обозначающие сущности или явления в моделируемой области;
- 2) связать эти категории определенными отношениями;

3) соотнести категориям набор конкретных экземпляров.

Но это лишь общий, упрощенный алгоритм, а в реальности этот процесс противоречив и рождает много дискуссий.

Так, ряд решений должен быть принят уже на начальных этапах создания онтологий. Нужно определить, создавать новый элемент и должен ли этот элемент быть включен в структуру. Следует понять, где по отношению к другим объектам должен располагаться вновь создаваемый, должен ли он быть видом какого-либо класса или же сам представляет собой родовой термин. Помощь в этом может оказать формулировка особых, уникальных свойств термина, то есть его отличительных характеристик. При этом не следует смешивать свойства понятия и его отличительные признаки.

Поскольку при формировании онтологий привлекаются специалисты различных областей, и для каждой области существуют свои методы:

– *абстракция и комбинирование свойств* – обычно используют философы, например, по Аристотелю следует выделять понятия как атомарные понятия, строящиеся из набора дифференциальных признаков;

– *интуитивные отличия* – метод когнитивистов, которые склонны полагаться на интуицию. Например, Э. Рош полагает, что функциональным предназначением классов является предоставление максимума информации при минимуме когнитивных усилий. Считая, что люди склонны формировать классы на основе прототипов, она предлагает вводить такие классы в дополнение к родовидовым иерархическим системам или вместо них;

– *метод дескриптивных логик* – специалисты в компьютерной области оперируют логическими теориями и используют структуры, построенные на умозаключениях;

– *описание межъязыковых соответствий* – метод, используемый обычно в лингвистике.

Общим же методом для всех является использование классификаций понятий (концептов).

При классификации концептов исследователи сталкиваются еще с одной принципиальной проблемой, так называемой «проблемой примитивности». Э. Хови указывает на то, что почти вся семантика и представление знаний основываются на композиционной гипотезе: можно определить ограниченный набор единичных сущностей («атомов»), а все остальные («молекулы») представлять как комбинацию (композицию) атомов. При этом возникает вопрос: как много таких «атомов» необходимо?

В современных исследованиях можно найти два различных подхода: экономный и неэкономный.

Экономный подход призывает создавать малое количество элементарных концептов, семантически простых (семантических примитивов), с помощью которых можно объяснить значение более сложных понятий. При таком положении вещей легко обнаружить связанность понятий, просто

определять и осуществлять умозаключения, однако достаточно сложно составлять сложные значения. Семантические примитивы – это лексические единицы, выражающие элементарные, базовые значения. Количество таких единиц не превышает 100, и их значения универсальны для всех языков.

Неэкономный подход позволяет создавать любое количество индивидуальных сущностей – столько, сколько захочется создателю онтологии. Это количество может варьироваться от 10 до 100 000 и более. Такая точка зрения свойственна, например, создателям WordNet. Здесь затруднительно определять связанность понятий, сложно работать с умозаключениями. Данный подход сопровождается, по существу, отказом от композиционной гипотезы, но это влечет за собой и преимущество: отсутствие необходимости составлять сложные значения.

Каким бы образом не решались проблемы построения, результатом должна быть онтология отвечающая общим требованиям:

- общая структура онтологии должна быть понятной;
- должна существовать возможность ее многократного использования;
- онтология должна быть ясной и легко передавать подразумеваемый смысл;
- объективной;

Последовательной – в ней должны содержаться утверждения, которые не противоречат друг другу, иерархии понятий, связывающим их отношениям, экземплярам.

– возможность расширения: наличие возможности введения новых элементов без пересмотра остальных элементов;

– минимальная степень специализации онтологии: нежелательность полного подчинения онтологии конкретной задаче, что может осложнить ее последующее использование в других задачах.

2.9. Инструментарий построения онтологий

Как упоминалось ранее, на данном этапе развития технологий появилась идея расширения существующей ныне World Wide Web за счет добавления к уже существующим документам семантической информации. Это расширение Интернет ресурсов получило название SemanticWeb. Создатель WWW Тим Бернерс-Ли определяет SemanticWeb как «расширение существующей WWW, в которой информации придается четко определенное значение, позволяющее людям и компьютерам сотрудничать». Майкл Ушолд поясняет, что «точного определения SW нет, но ее основная черта – наличие семантики, доступной для обработки машинами (программными агентами Сети) при взаимодействии друг с другом». SemanticWeb дает возможность компьютерам обрабатывать гипертекстовые страницы, предоставляющие информацию людям, и автоматически получать данные.

2.10. Языки описания онтологий

Любой элемент в SemanticWeb может получать идентификатор, называемый «Uniform Resource Identifier» или URI. Используется единообразная система идентификаторов, и каждый идентифицированный элемент рассматривается как ресурс, получивший определенное имя. Если что-то имеет URI, можно сказать, что оно находится в сети. Именно URI является основой и связующим звеном SemanticWeb, и они не подлежат перемещению. URI децентрализованы, они напрямую не контролируются никакой организацией.

Простой идентификации ресурса недостаточно, чтобы понять, как он может использоваться. Таким образом, URI бесполезны, если не уточнить, что они значат. Для этого и нужны онтологии и так называемые схемы, они являются неотъемлемой частью SemanticWeb и становятся основой для семантической разметки. В рамках SemanticWeb онтологии занимают центральное положение, что можно увидеть на рис. 7, они задают отношения между понятиями и определяют логические правила для рассуждений о них. Таким образом, компьютер может понимать смысл данных, обращаясь к онтологиям за требуемой информацией.

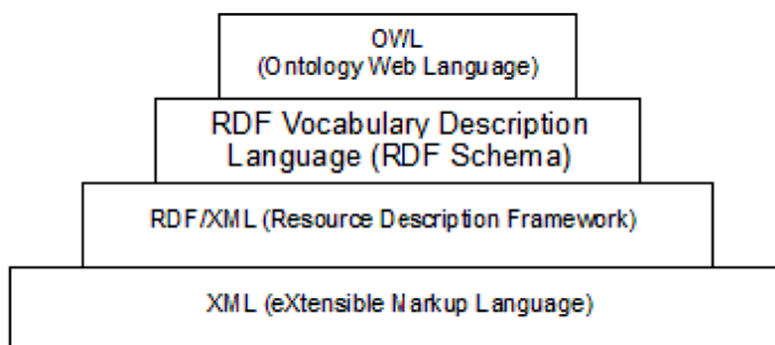


Рис. 7. Логическая зависимость языков описания онтологий

Основные этапы развития языков представления онтологий в SemanticWeb освещены в цикле лекций Б. В. Доброва и соавторов. Изначально для описания SemanticWeb использовался Extensible Markup Language (XML). Этот язык является простым инструментом создания документов с универсальными средствами описания. Он позволяет каждому описывать свою структуру, формат документа, в том числе различные виды разметки. Введенная с помощью XML «интеллектуальная» разметка поддается компьютерной обработке и может интерпретироваться особым образом.

Однако недостаточно просто описать элементы SemanticWeb, надо, чтобы компьютер мог понимать и анализировать составленные нами высказывания. Больше возможностей для обработки высказывания предоставляет формализм Resource Description Framework (RDF). Выражение

RDF имеет структуру простого предложения, с единственным отличием – все слова являются URI. Каждое высказывание RDF представляет собой тройку (S, P, O), где S – субъект, P – предикат и O – объект. Выражения RDF позволяют утверждать какую-либо информацию об элементах SemanticWeb и делать выводы. Этот язык удобен также для работы с базами данных.

Однако стоит помнить, что базы данных постоянно меняются, а слишком сложное описание может затруднить их быстрое обновление.

Затем стал развиваться язык RDFSchema (RDFS) – язык описания словарей RDF-терминов. RDFSchema определяет классы, свойства и другие ресурсы. Таким образом, RDFS стал семантическим расширением RDF. RDF и RDFS позволяют работать с метаданными, обеспечивать компьютер семантической информацией и обрабатывать эту информацию автоматически. Однако, используя RDF, «кто угодно (т. е. любой пользователь RDF) может сказать что угодно (т. е. фиксировать произвольное утверждение) о чем угодно (т. е. о любом ресурсе Сети)». Кроме того, «RDF не запрещает делать бессмысленных утверждений или утверждений не согласующихся с другими. Следовательно, нет никакой гарантии целостности и непротиворечивости RDF-описаний», поэтому при использовании информации пользователи должны сами проверять эту целостность.

Как уже упоминалось ранее, ядром SemanticWeb должны были стать онтологии, однако, как отмечается в отсутствие стандартизированных, единых языков онтологий затрудняло их использование внутри взаимосвязанных систем в одних и тех же областях знания. Было сложно устанавливать связи между онтологиями и соединять их. Онтологии, описывающие сходные специализированные знания, существенно отличались в синтаксисе и семантике в зависимости от используемого онтологического языка. Это препятствовало повсеместному использованию онтологий.

Создатели SemanticWeb попытались найти решение проблемы совместимости описаний, создав в 2004 году World Wide Web Consortium (W3C) для обсуждения возникших проблем. Ими был предложен универсальный стандарт для сетевого обмена онтологической информацией – Web Ontology Language (OWL). Данный язык помогает преодолеть проблему взаимодействия систем и становится основой для многих сетевых приложений. С его помощью эксперты предметной области и разработчики приложений могут создавать, модифицировать и соединять различные онтологии.

Стоит упомянуть, что сейчас существуют различные онтологические языки, однако OWL является их своеобразным универсальным объединением. Так, одним из первых таких языков являлся Simple HTML Ontology Extension (SHOE), разработанный в Университете Мериленда. Этот язык был дополнен специальными тэгами для включения семантической информации в обычный код HTML. Вскоре за ним в 2000 году в Амстердам-

ском университете был создан язык *Ontology Interchange Level (OIL)*. Здесь уже была добавлена формальная семантика, основанная на логическом описании. Этот язык базировался на таких упомянутых ранее стандартах W3C, как XML и RDF. Одновременно с ним в США появилась программа *DARPA Agent Markup Language (DAML)*. Она выпустила *DAML-ONT*, язык спецификации онтологий для реализации более сложных определений RDF классов, чем реализуемые с помощью RDFS. Позднее проект DAML соединил усилия с OIL и выпустил *DAML+OIL*. В *DAML+OIL* была усовершенствована языковая семантика, она стала яснее и успешнее взаимодействует с инструментами построения.

Web Ontology Language (OWL). Данный язык строится на основании RDF, который сам по себе строится на синтаксисе XML. RDF и OWL дают возможность создавать классы, свойства и отдельные экземпляры. Таким образом, данный язык реализует структуру онтологий. Можно привести пример использования этих трех компонентов, предложенный в работе: определим класс сущностей "People" и некоторые свойства "People", такие как "name", "birthday" и "friend".

Средствами синтаксиса RDF класс и его свойств будут описаны так:

```
<Class ID="Person"/>
<Property ID="name"/>
<Property ID="birthday"/>
<Property ID="friend"/>
```

Когда определены классы, их можно использовать при описании отдельных экземпляров. Информация «Joe Blog, born January 1, 1950, is friends with John Doe» будет представляться следующим образом:

```
<Person ID="Joe">
  <name>Joe Blog</name>
  <birthday>January 1, 1950</birthday><friend
resource="#John"/></Person>
  <Person ID="John">
  <name>John Doe</name></Person>
```

Такое описание может быть подвергнуто большей степени обобщения и использоваться как общее описание класса с указанием ограничений на свойства и отношения.

Язык OWL расширяет функциональность RDF, сохраняя при этом совместимость с другими программами и являясь открытой и свободной для распространения системой. Существует несколько вариантов данного языка: OWL Lite, OWL DL, и OWL Full. Каждая разновидность характеризуется своим соотношением детальности описания и объема, сложности прикладной области:

– OWL Lite предназначен для пользователей, которым необходима лишь классификационная иерархия и некоторые простые условия согласованности сущностей;

– OWL DL (Description Logic) рассчитан на пользователей, которым необходима максимальная степень выразительных возможностей языка без потери вычислительной полноты (с гарантией получения всех возможных умозаключений, получаемых формально-логическим путем) и разрешимости (все вычисления выполняются за конечное время). Уровень OWL DL ориентирован на существующие сегодня системы описания знаний и системы логического программирования;

– OWL Full рассчитан на тех пользователей, которым необходимы максимальные выразительные возможности языка и вся свобода выбора синтаксических средств, предоставляемая в RDF, но не обязательны вычислительная полнота и разрешимость. Онтология, записанная на OWL Full, позволяет расширять значения терминов, взятых из заданных словарей;

– OWL обладает рядом преимуществ перед RDF. К дополнительным возможностям, появившимся в этом языке, можно отнести возможность создавать локальные ограничения области распространения. Ранее у каждого свойства, отношения мог задаваться домен, но он оставался неизменным во всех областях применения. Однако порой то, как отношение применяется, зависит от конкретного класса, с которым мы работаем. Можно привести пример: рассмотрим свойство «ест» для класса «Человек». Возможно, для уточнения мы захотим добавить ограничение, указав, что значение для «ест» берется из класса «Еда». Это утверждение верно для общего класса «Человек», однако оно может не распространяться на подклассы, например, подкласс «Вегетарианец». Значение «ест» для этого более узкого класса не «Еда», а ее подмножество (все, кроме мяса). Однако такая детализация была невозможна на уровне RDF, в OWL же мы можем указать локальное ограничение свойства для конкретного класса. Это уточняет описание и расширяет возможности при создании онтологии.

Кроме того, в OWL вводятся основные функции над множествами, такие как объединение, пересечение, дополнение и непересекаемость. Ценность этой возможности можно проиллюстрировать на примере: пусть в нашем распоряжении есть два подкласса «Человек», называемых «Живой Человек» и «Мертвый Человек». При использовании описания RDF не существует никаких ограничений на определение объекта, который одновременно является экземпляром класса «Живой Человек» и класса «Мертвый Человек», так как ничто не определяет эти классы взаимоисключающими. Однако эта проблема успешно решается на уровне OWL.

Еще одним преимуществом OWL является введение понятия мощности. OWL позволяет накладывать ограничения на свойство, требуя чтобы оно использовалось для любого экземпляра минимальное количество раз (минимальная мощность), максимальное количество раз (максимальная мощность) или определенное количество раз (мощность). Это означает, что можно, например, потребовать, чтобы человек-билингв говорил на двух языках.

На данном этапе OWL применяется в ряде областей, она демонстрирует, что использование онтологический умозаключений сопоставимо по результатам со стандартными системами информационного поиска. Данная система включает в себя примерно 100 миллионов утверждений. Система поддерживает сложное формирование логического вывода и работу с запросами, используя подмножество OWL. Следует упомянуть, что OWL используется также в системах, не относящихся к анализу естественного языка.

Нужно отметить, что используемые ныне языки представления онтологий также рассчитаны и на моделирование нечетких знаний. Например, такие расширения RDF и OWL, как Fuzzy RDF и Fuzzy OWL, позволяют совмещать стандартные модели предметных областей и нечеткую логику, применяемую на уровне задания аксиом. Известно также расширение OWL, где используются вероятностные модели знаний, например, Bayes OWL.

Подводя итог, можно сказать, что OWL обеспечивает достаточно богатую семантику для описания онлайн-онтологий, которые важны для современных проектов анализа естественного языка и совместимы с web-стандартами и архитектурой. Сейчас существуют планы дальнейшего развития OWL для усовершенствования и возможно расширения описаний.

2.11. Стандарт онтологического исследования IDEF5

Исторически, понятие онтологии появилось в одной из ветвей философии, называемой метафизикой, которая изучает устройство реального мира. Основной характерной чертой онтологического анализа является, в частности, разделение реального мира на составляющие и классы объектов (at its joints) и определение их онтологий, или же совокупности фундаментальных свойств, которые определяют их изменения и поведение. Таким образом, естественная наука представляет собой типичный пример онтологического исследования. Например, атомная физика классифицирует и изучает свойства наиболее фундаментальных объектов реального мира, таких как элементарные частицы, а биология, в свою очередь, описывает характерные свойства живых организмов, населяющих планету.

Однако фундаментальные и естественные науки не обладают достаточным инструментарием для того, чтобы полностью охватить область, представляющую интерес для онтологического исследования. Например, существует большое количество сложных формаций или систем, созданных и поддерживаемых человеком, таких как производственные фабрики, военные базы, коммерческие предприятия и т. д. Эти формации представляют собой совокупность взаимосвязанных между собой объектов и

процессов, в которых эти объекты тем или иным образом участвуют. Онтологическое исследование подобных сложных систем позволяет накопить ценную информацию об их работе, результаты анализа которой будут иметь решающее мнение при проведении процесса реорганизации существующих и построении новых систем.

Методология IDEF5 обеспечивает наглядное представление данных, полученных в результате обработки онтологических запросов в простой естественной графической форме.

Основные принципы онтологического анализа

Онтологический анализ обычно начинается с составления словаря терминов, который используется при обсуждении и исследовании характеристик объектов и процессов, составляющих рассматриваемую систему, а также создания системы точных определений этих терминов. Кроме того, документируются основные логические взаимосвязи между соответствующими введенным терминам понятиями. В дальнейшем мы не будем делать различия между понятиями и терминами. Результатом этого анализа является онтология системы, или же совокупность словаря терминов, точных их определений взаимосвязей между ними.

Таким образом, онтология включает в себя совокупность терминов и правила, согласно которым эти термины могут быть скомбинированы для построения достоверных утверждений о состоянии рассматриваемой системы в некоторый момент времени. Кроме того, на основе этих утверждений, могут быть сделаны соответствующие выводы, позволяющие вносить изменения в систему, для повышения эффективности её функционирования.

В любой системе существует две основные категории предметов восприятия, такие как сами объекты, составляющие систему (физические и интеллектуальные) и взаимосвязи между этими объектами, характеризующие состояние системы. В терминах онтологии, понятие взаимосвязи, однозначно описывает или, другими словами, является точным дескриптором зависимости между объектами системы в реальном мире, а термины – являются, соответственно, точными дескрипторами самих реальных объектов.

При построении онтологии, в первую очередь происходит создание списка или базы данных дескрипторов и с помощью них, если их набор достаточен, создается модель системы. Таким образом, на начальном этапе должны быть выполнены следующие задачи:

1. Создание и документирования словаря терминов
2. Описание правил и ограничений, согласно которым на базе введенной терминологии формируются достоверные утверждения, описывающие состояние системы.

3. Построение модели, которая на основе существующих утверждений, позволяет формировать необходимые дополнительные утверждения.

Что мы имеем в виду под необходимыми дополнительными утверждениями? Дело в том, что при рассмотрении каждой системы существует огромное количество утверждений, достоверно отображающих ее состояние в различных разрезах, а построенная онтологическим способом модель должна выбирать из них наиболее полезные для эффективного рассмотрения в том или ином контексте. Дополнительно, эта модель помогает описывать поведение объектов и соответствующее изменение взаимосвязей между ними, или, другими словами, поведение системы. Таким образом, онтология представляет собой некий словарь данных, включающий в себя и терминологию и модель поведения системы.

Концепции IDEF5

Процесс построения онтологии, согласно методологии IDEF5 состоит из пяти основных действий:

1. Изучение и систематизирование начальных условий. Это действие устанавливает основные цели и контексты проекта разработки онтологии, а также распределяет роли между членами проекта.

2. Сбор и накапливание данных. На этом этапе происходит сбор и накапливание необходимых начальных данных для построения онтологии.

3. Анализ данных. Эта стадия заключается в анализе и группировке собранных данных и предназначена для облегчения построения терминологии.

4. Начальное развитие онтологии. На этом этапе формируется предварительная онтология, на основе отобранных данных.

5. Уточнение и утверждение онтологии – Заключительная стадия процесса.

Язык описания онтологий в IDEF5

Для поддержания процесса построения онтологий в IDEF5 существуют специальные онтологические языки: схематический язык (Schematic Language-SL) и язык доработок и уточнений (Elaboration Language-EL). SL является наглядным графическим языком, специально предназначенным для изложения компетентными специалистами в рассматриваемой области системы основных данных в форме онтологической информации. Этот несложный язык позволяет естественным образом представлять основную информацию в начальном развитии онтологии и дополнять существующие онтологии новыми данными. EL представляет собой структурированный текстовый язык, который позволяет детально характеризовать элементы онтологии.

Язык SL позволяет строить разнообразные типы диаграмм и схем в IDEF5. Основная цель всех этих диаграмм – наглядно и визуально представлять основную онтологическую информацию.

Несмотря на кажущееся сходство, семантика и обозначения схематического языка SL существенно отличается от семантики и обозначений других графических языков. Дело в том, что часть элементов графической схемы SL может быть изменен или вовсе не приниматься во внимание языком EL. Причина этого состоит в том, что основной целью применения SL является создание лишь вспомогательной структурированной конструкции онтологии, и графические элементы SL не несут достаточной информации для полного представления и анализа системы, тем самым они не предназначены для сохранения при конечном этапе проекта. Тщательный анализ, обеспечение полноты представления структуры данных, полученных в результате онтологического исследования, являются задачей применения языка EL.

Использование стандартизованных языков позволяет точно описывать содержание, лежащее в основе HTML страниц, а также устанавливать скрытую информацию. Так, описанию могут подвергаться изображения и видеоматериалы, содержащиеся на сайтах или же какие-либо недоступные при простом чтении страницы данные.

2.12. Программные инструменты для построения и работы с онтологиями

К настоящему времени получили известность средства создания онтологий, такие, как Ontoligua, Onto Edit, Oil Ed, Protege, Web-Deso.

Среда разработки Ontoligua предназначена для коллективного использования системы базовых знаний при построении собственных онтологий. Она предоставляет разработчику библиотеку модулей, на основе которой осуществляется расширение онтологий. Среда разработки Onto Edit предназначена для проектирования, приспособления и импорта/экспорта моделей знаний в форматах RDF, DAML+OIL, Flogic для/из прикладных систем. Редактор онтологий Oil Ed в большей степени предназначен для проверки разработанных онтологий на согласованность. Система Protege является библиотекой, предоставляющей доступ другим приложениям для просмотра баз знаний и позволяющей редактировать и наращивать базы знаний. Система Web-Deso предназначена для создания онтологий некоторой предметной области. Онтологии предметных областей соединяются в одну результирующую онтологию предметной области и помещаются в библиотеку вместе с источниками знаний. Такая же операция выполняется для онтологий задач. Сформированные в библиотеке онтологии интегрируются в онтологию-приложение, которая тоже хранится в библиотеке и предназначена для обеспечения многократного доступа к представленным знаниям.

2.13. Построение онтологий верхнего уровня

Данные онтологии должны интегрироваться с уже созданными онтологиями или создаваемыми в перспективе. Это требование определяет то, что понятия и отношения, закладываемые в эти онтологии, носят общеизвестный характер и извлекаются из устоявшихся источников. В связи с этим онтологию верхнего уровня предлагается строить в виде тезауруса, описывающего терминологию предметных областей как терминосистему в виде словаря с концептуальным входом и фиксированными семантическими связями между его единицами с возможностью их редактирования в процессе функционирования.

Под терминосистемой будем понимать систематизированную совокупность терминов. При построении тезаурусов будем использовать тематическую и иерархическую классификации – членение дисциплины науки на разделы или направления, а внутри раздела единицы тезауруса могут быть связаны между собой иерархическими и неиерархическими отношениями.

В тезаурусе определим два вида словарных статей – «понятие» и «действие»/ «операция». Предполагается, что знания для создания терминосистемы будут извлекаться из терминологических словарей. Структура терминосистемы должна определять связи терминов, переходы внутри общей совокупности терминов; описывать семантику, синтактику и прагматику отдельных терминов; включать описание набора семантических предикатов, регулярно связывающих термины в научных текстах.

Для построения компьютерной терминосистемы будем использовать аппарат теории семиотического моделирования, предоставляющий математический базис для построения систем такого типа. Рассмотрим квадрат Д. А. Поспелова, показанный на рис. 8.

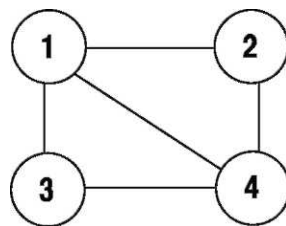


Рис. 8. Квадрат Поспелова

В этом квадрате первая вершина определяет синтаксис, или способ кодирования знака, вторая – семантику, или понятие о знаке, третья соответствует прагматике – тем процедурам, которые связаны с этим знаком, четвертая – множеству знаков, или фрагменту некоторой структуры на множестве знаков (она играет роль денотата метазнака). Фрагмент структуры на множестве знаков обладает собственным именем, выделяющим

его среди остальных. Это имя представлено в вершине 1, понятие о фрагменте дано в вершине 2, а связанные с ним действия – в вершине 3. Стороны квадрата и его диагональ соответствуют различным процедурам, связывающим компоненты знака. Метазнак образует вершины 1, 2 и 3 квадрата.

В соответствии с данной схемой представления знака в семиотической системе словарные статьи тезауруса можно представить в виде фреймов, а множество фреймов, описывающих термины тезауруса, должно образовывать семантическую сеть иерархического типа.

Дуги такой сети соответствуют различным связям между ними, при этом иерархические связи определяются отношениями структуризации, а не иерархические – отношениями иных типов. В связи с тем, что построение тезауруса в виде семантической сети фреймов будет выполняться на основе анализа терминологического словаря или нескольких словарей, следует:

- 1) определить структуры словарной статьи и соответствующего ей фрейма;
- 2) определить отношения, которые необходимы для построения семантической сети, и способы их выявления;
- 3) построить иерархическую систему терминов в виде семантической сети понятий.

Так как нам требуется создать тезаурус для общего доступа, он должен иметь два представления: на естественном языке и формальном. Сначала необходимо создать тезаурус на естественном языке, а затем, проанализировав его и придав ему окончательную форму, на его основе можно построить семантическую сеть фреймов.

2.14. Определение структур словарной статьи и соответствующего ей фрейма

В качестве примера рассмотрим структуру словарной статьи «Понятие». В каждой словарной статье описывается одно понятие/термин. Термин – это знак специальной семиотической системы, обладающий номинативно-дефинитивной функцией или по второму определению: термин – это устоявшееся понятие. Предполагаем, что в терминологических словарях описаны только устоявшиеся понятия рассматриваемой области знаний. Включим в словарную статью следующие элементы: денотат понятия, дефиниции понятия, свойства понятия, синонимы, оппозиции, список терминов, с которыми данное понятие имеет отношения. Тогда структуру словарной статьи «Понятие» можно представить в виде восьмерки:

$$T = \langle A, B, C, D, E, F, G, H \rangle.$$

Рассмотрим элементы структуры.

$A = \langle A:, A2 \rangle$ – денотат понятия, где $A:$ – уникальное имя понятия или имя словарной статьи; $A2$ – знак понятия, соответствующий идентификатору фрейма.

$B = \{Bb B2, B3, \dots\}$ – множество дефиниций понятия. Для одного понятия может быть несколько дефиниций. Очевидно, что из общего набора утверждений для дефиниции отбирается только часть. Поскольку термин – это узел в сложной структуре теоретической системы, то различные дефиниции можно рассматривать как «разные пути или способы вхождения нашей мысли фактически в одну и ту же структуру». Обычно дефиницию определяют как словесно выраженный интенционал, достаточный для задания экстенционала, поэтому будем различать содержательную и формальную дефиниции. Множество содержательных дефиниций будет храниться в тезаурусе на естественном языке. Будем считать, что формальная дефиниция определяет понятие через другие понятия, а именно – через собственные свойства и связи с другими понятиями. Формальная дефиниция должна быть представлена в обоих видах тезауруса. Это согласуется с мнением Г. П. Мельникова, который выделяет следующие виды дефиниций: субстанциальные, структурные, тезаурусные и системные. Автор отмечает, что системная дефиниция – это такое сочетание субстанциональной (содержательной) и структурной (формальной) форм, при котором из структурной информации вытекает представление о наиболее вероятной субстанциальной информации, а из субстанциальной – наиболее вероятные структурные взаимодействия элементов поля терминосистемы. Таким образом, эти два аспекта обеспечивают представление о ее целостности и функциональной оправданности.

$C = \langle Cj, C2, C3, C4 \rangle$ – множество типов описываемого концептуального объекта, в котором выделены четыре типа согласно Дальбергу:

сущность Cj материальные и нематериальные объекты, способы их рассмотрения;

свойства $C2$: количественные, качественные, релятивные (отношения);

действия $C3$: операции, процессы, состояния;

величины (dimensions) $C4$: время, положение, пространство.

$D = \langle D:, D2 \rangle$ – пара множеств свойств понятия, где $D:$ – множество качественных свойств; $D2$ – множество количественных свойств.

$E = \{E:, E2, E3, \dots\}$ – множество понятий, описывающих методы/функции, свойственные данному понятию, и отражающих прагматику, связанную с данным понятием.

$F = \{F:, F2, F3, \dots\}$ – множество синонимов понятия, или, другими словами, множество понятий, имеющих квантитативные отношения (отношение тождества) с данным понятием.

$G = \{Gb G2, G3, \dots\}$ – множество коррелятов, или, другими словами, множество понятий, имеющих отношение оппозиции к данному понятию.

$H = \langle H1, H2 \rangle$ – пара множеств понятий, имеющих качественные отношения с данным, где:

$H1$ – множество понятий, составляющих отношение обобщения с данным $\langle H11, H12 \rangle$, $H11$ – родовое понятие, $H12$ – множество видовых понятий;

$H2$ – множество понятий, составляющих отношение агрегации с данным $\langle H21, H22 \rangle$, $H21$ – понятие, являющееся «целым» по отношению к описываемому, $H22$ – множество понятий, являющихся частью описываемого.

Протоструктура фрейма имеет три уровня иерархии. В ней по аналогии с регулярными выражениями символ «*» означает итерацию, круглыми скобками выделены вложенные уровни иерархии.

2.15. Построение семантической сети

Рассмотрим виды отношений для построения семантической сети.

Виды отношений:

1 Концептуальные отношения (по Дальбергу):

– *количественные* (совпадают с логическими отношениями тождества, включения, пересечения, дизъюнкции);

– *качественные* (в большинстве онтологические и включают в себя отношения иерархии, части – целого, оппозитивные и функциональные).

Отношения иерархии (обобщение). Эти отношения принадлежат сфере «быть» и делятся на три подвида:

род \leftrightarrow вид;

признак \leftrightarrow значение признака;

инвариант \leftrightarrow вариант.

Родовидовое или внутри категориальное отношение род \leftrightarrow вид определяет принадлежность к классу, строится на основе комплексного использования параметра смысла и объема номинации и подразделяется на структурные, функциональные, семантические.

Основные диагностирующие конструкции предложения для поиска родового понятия В и видового понятия А:

А относится к В;

родом А является В;

А принадлежит классу/ семейству В.

Основные диагностирующие конструкции предложения для поиска видовых понятий В и родового понятия А:

А имеет следующие виды: В₁, В₂, ...;

к видам А относятся В₁, В₂, ...

Отношение признак \leftrightarrow значение или отношение типизация-конкретизация. Здесь признак или тип – это категория, а значение – это имя конкретной категориальной формы.

Основные диагностирующие конструкции предложения для поиска А – понятия-категории, В – имени категории:

А – В;

А имеет имя В;

А именуется В;

А называется В.

Отношение инвариант \leftrightarrow вариант. В этом отношении инвариант – это неизменная структурная единица языка науки, а вариант – разновидность структурной единицы языка науки.

Основные диагностирующие конструкции предложения для поиска понятия инварианта А и варианта В:

к вариантам А относятся В1, В2, ...;

инвариантом В является А.

Отношения агрегации. Они принадлежат сфере «иметь». Лингвисты выделяют четыре подвида:

- целое \leftrightarrow часть (компонент);
- объект \leftrightarrow пространство реализации объекта (локализации или позиции);
- объект \leftrightarrow свойства/признак;
- уровень \leftrightarrow единица уровня.

В отношении целое \leftrightarrow часть (компонент) целое – это то, что перестает существовать, если отнять любую часть; часть – то, что без целого не существует. Для их поиска будем использовать диагностирующие конструкции, из которых состоит библиотека образцов:

– целое В – часть А:

А входит в состав В; А составляет часть В; А является частью; А – элементом В;

– часть В – компонент А: А состоит из В-ов;

А включает в себя В как часть;

А включает в свой состав В как часть;

А имеет своими частями/ элементами В1, В2, Вп.

В отношении объект \leftrightarrow пространство реализации объекта пространство реализации – это то пространство, где объект А проявляет свои свойства и функции.

Диагностирующие выражения:

место А – В;

А происходит в В;

А находится в В, А входит в В.

В отношении объект \leftrightarrow свойства свойство – это то, что присуще предметам, что отличает их от других; свойства делятся на существенные и несущественные. В некоторых случаях вместо термина «свойство» удобно использовать термин «признак». Признак – все то, в чем предметы, явления сходны или отличны (показатель, сторона предмета), при этом существуют отличия свойства/ признака от части. К ним относятся:

логические: свойство не должно иметь субстратного сходства с объектом и свойство не может существовать само по себе, но может проявляться в другой сущности;

языковые признаки: свойство имеет номинацию, в которой его «признаковость» выражена морфологическим способом: аффикс «-ость» определяет абстрактное свойство, кроме того, свойство не является главной частью речи.

Этот тип отношений позволяет строить иерархии, при этом каждый уровень будет соответствовать одному метаязыку, кроме первого, так как на первом уровне определяется отношение между термином, относящимся к плану содержания, и терминами, относящимися к плану выражения.

Отношение термин одного метаязыка \leftrightarrow термин второго метаязыка позволяет установить соответствие между двумя знаковыми системами как в одной предметной области, имеющей разные уровни представления знаний, так и между смежными предметными областями.

2.16. Построение иерархической системы терминов

После анализа терминологического словаря будут заполнены не все элементы словарных статей тезауруса, а только та часть, которая имеется в терминологическом словаре в описании терминов. Для дальнейшей систематизации терминов, т. е. построения иерархической системы, необходимо выполнить анализ заполненных словарных статей. Для этого будем использовать следующие типы операций обобщения: по именам, признакам, характеристикам и структуре.

Обобщение по именам. Эта операция связана с установлением отношения «элемент/ класс» между некоторой группой знаков, т. е. установление определенного сходства сущностей в процессе сравнения или сопоставления их признаков (свойств), в результате которого может порождаться новое, более общее понятие, которое объединяет целый класс подобных понятий, что соответствует абстракции типизации. Это новое понятие может находиться в другой словарной статье, и тогда необходимо установить связи между этими фрагментами тезауруса. Если такое понятие отсутствует в тезаурусе, то его необходимо ввести. Добавление понятия можно осуществлять в интерактивном режиме или использовать другой терминологический словарь. Таким образом, данная операция является основой способа поиска пары противоположных абстракций типизация–конкретизация.

Обобщение по признакам. Эта операция устанавливает отношение «вид/ род» между некоторой группой знаков на ниже/ вышележащем уровне по отношению к уровню, где располагается анализируемый термин. При выявлении родового понятия анализируются понятия, расположенные на одном уровне с рассматриваемым термином, и осуществляется объеди-

нение понятий на основе их сходства или выявленного подобия части признаков. При этом порождается новое (или существующее в тезаурусе) понятие, которое и является обобщением исходных понятий. Как и в предыдущем случае, прописываются связи между фрагментами. При поиске видовых понятий текущего понятия осуществляется анализ понятий нижележащего уровня. Как правило, используется нисходящий подход при поиске родовидовых отношений. Операция обобщение по признакам в такой интерпретации является основой способа поиска пары противоположных абстракций обобщение–специализация.

Эта операция используется также для поиска отношения «инвариант<–>вариант». В этом случае необходимо полное сходство признаков двух рассматриваемых понятий. Если это условие выполняется, то в тезаурусе прописывается их взаимосвязь.

Обобщение по характеристикам. Эту операцию будем использовать для установления связей между понятиями, находящимися в отношении «часть–целое». Это отношение невозможно определить посредством каких-либо операций, поэтому будем полагать, что они будут прописаны при анализе терминологического словаря. Здесь необходимо только прописать ссылки между ними.

Обобщение по структуре. Обобщение такого типа становится возможным лишь на уровне семиотического моделирования, когда используется язык, ориентированный на знаковую сущность моделей окружающего мира. Эта операция может быть использована для поиска семантических отношений в тезаурусе.

Все операции формализуются в продукционные правила. Применение продукций позволит окончательно построить иерархию понятий в виде семантической сети. На этом этапе должно быть сформировано фреймовое представление словарных статей

2.17. Построение онтологий предметных областей

Онтологию предметных областей обычно строят в виде тезауруса, описывающего категорийно-понятийный аппарат предметной области. Структуры тезаурусов идентичны, однако в данном случае роль источника знаний играют учебники, учебные пособия, монографии. Для построения категорийно-понятийного аппарата предметной области необходимо решить следующие задачи:

- выделение понятий научного текста и их классификация на терминологию и профессионализмы;
- построение взаимосвязей между понятиями;
- анализ тезауруса и выявление общенаучной, межнаучной и частнонаучной терминологий.

2.17.1. Построение взаимосвязей между понятиями

Формирование иерархической системы признаков можно выполнить способом, аналогичным описанному способу в разделе «Построение иерархической системы терминов».

Будем считать, что терминосистема содержит понятия общенаучного характера для данной области знаний, а тезаурус, содержащий категориально-понятийный аппарат, должен содержать частнонаучную систему понятий. Межнаучная система понятий выявляется в том случае, когда имеется несколько тезаурусов, описывающих смежные области знаний.

При анализе созданного категориально-понятийного аппарата должны анализироваться два созданных тезауруса. Для этого должна использоваться операция «сходства/ различия», которая заключается в проверке денотата, дефиниций и свойств понятия на соответствие. Применение данной операции может привести к следующим результатам:

- полное совпадение;
- частичное совпадение;
- полное различие.

Полное совпадение означает, что понятие носит общенаучный характер и в данной словарной статье необходимо оставить только ссылку на тезаурус, содержащий терминосистему, и все элементы, кроме денотата, следует обнулить.

Частичное совпадение означает, что данное понятие тоже носит общенаучный характер, но имеет конкретное назначение для описываемой предметной области. В этом случае необходимо в денотат понятия добавить ссылку на тезаурус, содержащий терминосистему.

Полное различие означает, что понятие носит научно-частный характер и принадлежит только описываемой предметной области.

2.17.2. Программная реализация

Фрагменты описания терминов, соответствующие пунктам алгоритма создания онтологии приведены в табл. 2, 3, 4.

OWL – конструкции описывают каждую строку таблиц. Приведем примеры строки таблицы (табл. 2) и конструкцию описания на языке OWL.

Таблица 2

Фрагмент словаря концептов

Концепт i	Предикат	Концепт j
Лекция	Состоит из	Тема
Раздел	Часть	Темы
Лабораторная работа	Состоит из	Задания

Пример реализации утверждения «Тема – часть – Лекция»:

```

owl:Classrdf:ID="Лекция">
<rdfs:labelrdf:datatype="http://www.w3.org/2001/XMLSchema#string">
Лекция</rdfs:label>
<rdfs:subClassOf>
<owl:Classrdf:about="#Тема"/>
<rdfs:commentrdf:datatype="http://www.w3.org/2001
XMLSchema#string"
</rdfs:subClassOf>
</owl:Class>

```

Таблица 3

Фрагмент таблицы описания бинарных отношений

Имя отношения	Пункт
Исходный концепт	Раздел
Порядок следования	(1,n)

Пример конструкции для описания бинарных отношений:

```

<owl:Data type Propertyrdf:about="#часть-целое">
<rdfs:domain>
<owl:Class>
<owl:unionOfrdf:parseType="Collection">
<owl:Classrdf:about="#Термин_дисциплины"/>
</owl:unionOf>
</owl:Class>

```

Таблица 4

Фрагмент таблицы описания атрибутов экземпляра

Имя атрибута экземпляра	Порядок следования
Тип значения	Целый
Комментарий	Строка

Пример конструкции для описания экземпляра «комментарий»:

```

<owl:Classrdf:about="#Правовое_обеспечение">
<rdfs:commentrdf:datatype="http://www.w3.org/2001/XMLSchema#string">
Совокупность законов, инструкций, лицензий и правил, обеспечи-
вающих проведение интерактивного обучения и легитимность документов,
подтверждающих квалификацию обучаемого</rdfs:comment>
<rdfs:labelrdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Правовое_обеспечение</rdfs:label>
</owl:Class>

```

2.17.3. Визуализация онтологии

Методы визуализации онтологий могут быть сгруппированы различными способами: по способу представления, размерности изображения, способа взаимодействия с пользователем. По типам, методы могут быть разбиты на следующие группы:

- 1) иерархический список;
- 2) узлы-связи и деревья;
- 3) масштабируемые;
- 4) заполнение пространства;
- 5) фокусирование и искажение;
- 6) трёхмерные ландшафты.

Методы, попавшие в одну из категорий, могут содержать в себе черты другой. Дополнительно методы в каждой группе разбиты на двух- и трехмерные. Двухмерные методы используют поверхность экрана как плоскость без каких-либо элементов глубины. Трёхмерные методы вводят еще одно измерение для того, чтобы приблизиться к представлению реального мира или улучшить взаимодействие с пользователем. Также некоторые трехмерные методы позволяют пользователю передвигать и вращать объекты в виртуальном мире и/или перемещаться в нём.

Иерархический список

Многие инструменты визуализации онтологий, в том числе Protege, OntoEdit, Каон и некоторые другие, предлагают в качестве основного способа визуализации представления в стиле Windows Explorer. При этом таксономия онтологии представляется как дерево.

Узлы-связи и деревья

Методы этой категории отображают онтологию как набор соединяющихся узлов, представляющих онтологию. Пользователю обычно предоставляется возможность разворачивать и сворачивать узлы и поддеревья для управления детализацией информации.

Двумерные

OntoViz плагин визуализации для Protege, использует библиотеку GraphViz для создания простого представления онтологии в виде двумерного графа. Есть возможность для каждого класса отображать имя, свойства, наследование и роли. Случаи могут быть показаны разными цветами.

IzaVi – визуальная среда для создания и редактирования rdf-онтологий в форме направленных графов. Графы визуализируются с помощью эллипсов, прямоугольников и рёбер между ними. Узлами являются классы, индивиды и значения свойств, свойства представляются как рёбра графа.

SpaceTree – визуализатор деревьев, выполненный на основе диаграмм узел-связь с использованием сворачивания узлов, которые не могут быть отображены. В данной реализации свёрнутые узлы представляются

специальными иконками, затенение которых пропорционально количеству свёрнутых узлов, а ширина – глубине скрываемого поддерева.

OntoTrack – инструмент «в одном представлении» для просмотра и редактирования онтологий с иерархическим расположением. Данный метод напоминает визуализацию SpaceTree, поскольку представляет, отход от иерархий с треугольниками ширины, длины и штриховки, которая приближает глубину, ветви и число подклассов.

GoSurfer – инструмент визуализации для поиска данных, связанный с определенными генами Go, как входными данными. Используется принцип визуализации дерева сверху вниз и инструменты для того, чтобы сравнить гены относительно их соответствующих сроков в онтологии Go, то есть, сравнение путей онтологии.

Трёхмерные

Onto Sphere. Предлагает визуализацию типа узел-связь, и использует три различных представления онтологии для представления пользователю обзора или деталей, в зависимости от его потребностей. Корневая сцена представляет сферу с расположенными на её поверхности классами верхнего уровня, представляемых как малые сферы. Она визуализирует не таксономию, а ролевые отношения между классами. Цвет и размер используются для выделения поддеревьев и их размера. Сцена дерева отображается по левому щелчку на классе и показывает класс и его поддерево.

Масштабируемые

Grokker – система для отображения карт знаний. Она предлагает графическое представление для информации типа результатов web-поиска или файлового поиска. Механизм кластеризации показывает документы как наборы вложенных диаграмм Вена. Пользователь может перемещаться по иерархии простыми кликами на круге. Когда круг выделен, он увеличивается, делая своё содержание видимым. Круги заполнены цветом, подсказывающим, что они находятся на нижних уровнях иерархии. На нижних уровнях иерархии пользователь может выбирать документы для просмотра в большом окне.

Jambalaya – плагин визуализации для Protege, использующий SHriMP (Simple Hierarcical Multi Perspective). SHriMP использует вложенный просмотр графа и концепцию вложенных перемещаемых представлений.

Crop Circles

Визуализация, представляющая дерево иерархии классов в виде набора окружностей. Узлам выделяется соответствующее место для того, чтобы обеспечить размещение всех поддеревьев. Единственный потомок помещается как концентрический круг в своего родителя, несколько потомков помещаются в родителя от больших к меньшим. Пользователь может щелчком по окружности выделить её и просмотреть список непосредственных потомков соответствующего узла.

Заполнение пространства

TreeMaps. Использует 2-мерный подход к заполнению пространства для представления иерархий, с помощью прямоугольной области с прямоугольным разбиением. Размер и цвет используются для представления данных. Пользователь может более детально просмотреть интересующую его область двойным щелчком, при этом она увеличится до размера всего окна.

Information Slices. Использует один или несколько дисков для компактного двумерного представления иерархий. Каждый диск представляет многие уровни иерархии, обычно на каждом диске размещается 5–10 уровней иерархии.

Sequoia View визуализирует деревья подобно TreeMap.

Происходит объединение Treemaps (заштрихованная область) с Squarified Treemaps, который использует прямоугольники с меньшим форматом изображения.

Фокусирование и искажение.

Двумерные

TGViz Tab. Встраивает визуализацию Touchgraph в Protege. Touchgraph – это среда с открытым исходным кодом для создания и просмотра графов. Она размещает семантически схожие узлы рядом. Это визуализация позволяет пользователю выполнять навигацию постепенно делая видимыми разные части графа. Также возможно сворачивать и разворачивать узлы. Кроме того пользователь имеет полный контроль над цветом и видимостью отдельных типов связей, может менять степень увеличения или делать граф гиперболическим.

Moire Graphs – визуализация объединения топологии графа. Центр поддержек и контекст с рядом методов взаимодействия для исследования графа, имея визуальное содержание, например, изображения, документы и т. д.). Эта техника использует радиальные графы. В этих графах сосредоточенный узел появляется в центре, в то время как узлы, связанные с этим, помещены вокруг него. Каждый следующий уровень узлов отдален от центрального, соответствуя внешнему концентрическому кругу.

Трёхмерные

3DHyperbolus Tree. Было создано для визуализации web-сайтов, но использовалось как файловый браузер. Он представляет дерево в гиперболическом пространстве для достижения большей плотности отображаемых данных. Узлы дерева размещаются на поверхности сферы. Информационные ландшафты Filesystem navigator. Создавался как трёхмерный файловый обозреватель для UNIXсистем. Высота узлов на плоскости представляет количество содержащихся в них файлов. При взгляде сверху узлы образуют двумерное дерево, представляющее иерархию файловой системы. Выбор узла подсвечивает его, а двойной щелчок открывает его для детального просмотра.

Harmony Information Landscape. Был разработан для гипертекста и упорядочивает узлы, представляемые как трёхмерные объекты, непосредственно на плоскости. Объекты при этом отличаются цветом и размером в зависимости от содержимого. Кроме того, так как документы гипертекстовые, между ними также отображаются связи. В случае с онтологией аналогично могут визуализироваться роли.

Таким образом, в настоящее время существует множество методов и инструментов визуализации онтологий. В основном это методы в области графов, визуализации иерархии, а также в двумерном и трехмерном пространстве. Выбор какого-то конкретного метода визуализации для данной предметной области является специфическим и должен иметь свои особенности.

Не существует одного определенного метода, который соответствовал бы приложению. Следовательно, жизнеспособное решение состоит в том, чтобы предоставить пользователю несколько визуализаций, таким образом, чтобы он в состоянии мог выбрать тот, который является соответствующим для его текущих потребностей. Некоторые инструменты управления онтологией уже обеспечивают комбинации методов визуализации. Например, Protégé, который включает несколько плагинов визуализации.

В Приложение 2 приведена онтология декларативных знаний раздела «Теория алгоритмов» дисциплины «Математическая логика и теория алгоритмов». Стрелки синего цвета обозначают связи между классами, красного – связь между экземпляром и классом, к которому этот экземпляр принадлежит. Онтология выполнена в редакторе Protégé и визуализирована при помощи плагина Jambalaya в режиме просмотра графа.

Контрольные вопросы и задания

1. Что означает термин «онтология»?
2. Какие формы онтологии наиболее типичны?
3. Для чего используются аксиомы в онтологии?
4. Каковы этапы создания онтологий?
5. Сформулируйте определение онтологии на формальном уровне
6. Сформулируйте определение таксономии
7. Структура реальности, рассматриваемая независимо от словаря предметной области и конкретной ситуации называется (концептуализация).
8. Какие уровни онтологий выделяют при классификации по цели создания?
9. Какие концепты определяют общую онтологию?
10. Что такое синсеты?

11. Дайте определения лингвистических и формальных онтологий, укажите различия.
12. Что представляют собой названия свойств классов?
13. Что происходит с типами в системе WordNet, которые не находятся в родо-видовой иерархии?
14. Какие существуют способы передачи/ получения метаданных в сети ?
15. Какие формы высказывания существуют?
16. Основная задача, решаемая при проектировании архитектуры метаданных Сети?
17. Какие языки запросов к RDF-хранилищам на сегодняшний день являются наиболее популярными?
18. Сформулируйте определение понятий: омонимы, синонимы, антонимы, паронимы, меронимия, холонимия.
19. Как называются элементарные структурные единицы WordNet?
20. Какие существуют отношения между сенсетам глаголов?
21. Какие существуют проблемы использования онтологии в информационном поиске?
22. Основное назначение ассоциативных отношений между дескрипторами.
23. Понятие тезауруса. Область применения и ограничения.
24. Онтология в информатике.
25. Семантическая сеть.
26. Типы отношений в семантических сетях.
27. Организация знаний с помощью онтологии.
28. IDEF5 – стандарт онтологического исследования. Основные положения.
29. Языки разработки онтологий.
30. Сравнительные характеристики инструментальных средств разработки онтологий.

ЗАКЛЮЧЕНИЕ

Данное учебное пособие ориентировано на студентов, обучающихся в области информационных технологий и автоматизированных систем. В первую очередь это относится к студентам бакалавриата по направлениям подготовки 09.03.01 «Информатика и вычислительная техника», 09.03.02 «Информационные системы и технологии» всех форм обучения

Разделы учебного пособия включают изложение представления знаний, моделирования знаний, современных концепций и технологий в организации баз данных и баз знаний, которые потом могут быть востребованы в профессиональной деятельности.

Примеры понятий и определений, которые приводятся в учебном пособии, дают картину реального развития современных информационных технологий, а также послужат базой для дальнейшего самообразования, самосовершенствования в своей профессиональной сфере.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Вирт, Н. Алгоритмы, структуры данных, программы / Н. Вирт. – М. : Мир, 1985. – 432 с.
2. Гаврилова, Т. А. Базы знаний интеллектуальных систем : учебник / Т. А. Гаврилова, В. Ф. Хорошевский. – СПб. : Питер, 2000. – 384 с.
3. Галеев, И. Х. Сетевое обучение проектированию онтологий на примере онтологии компьютерных вирусов [Электронный ресурс] / И. Х. Галеев, Л. Н. Абуталипов, А. И. Филяев О. В. Колосов. – Режим доступа: http://ifets.ieee.org/russian/depositary/ONTOLOGY_publ_r.rtf. – Загл. с экрана.
4. Грибова, В. В. Управление программными средствами в интеллектуальных системах / В. В. Грибова, А. С. Клещев, Е. А. Шалфеева // Известия РАН. Теория и системы управления. – 2010. – № 6. – С. 122–137.
5. Грибова, В. В. Развитие онтологического подхода для автоматизации разработки пользовательских интерфейсов с динамическими данными / В. В. Грибова, Н. Н. Черкезишвили // Информационные технологии. – 2010. – № 10. – С. 54–58.
6. Добров, Б. В. Формирование базы терминологических словосочетаний по текстам предметной области / Б. В. Добров, Н. В. Лукашевич // Электронные библиотеки: Перспективные методы и технологии, электронные коллекции : труды Пятой всерос. науч. конф. – М. : Науч.-исслед. вычислительный центр МГУ им. М. В. Ломоносова, 2003.
7. Доррер, Г. А. Технологии моделирования и разработки учебных электронных изданий / Г. А. Доррер, Г. М. Рудакова ; отв. ред. В. С. Соколов. – Новосибирск : Изд-во СО РАН, 2006. – 272 с.
8. Загорулько, Ю. А. Искусственный интеллект. Инженерия знаний : учебное пособие для вузов [Электронный ресурс] / Ю. А. Загорулько, Г. Б. Загорулько. – М. : Юрайт, 2018. – 93 с. – (Серия : Университеты России). – ISBN 978-5-534-07198-6. – Режим доступа: www.biblio-online.ru/book/172BD6D4-D6E7-4D94-8390-054975CB16C5.
9. Клещев, А. С. Классификация свойств онтологий. Онтологии и их классификации / А. С. Клещев, Е. А. Шалфеева // Научно-техническая информация. – Сер. 2. – 2005. – № 9. – С. 16–22.
10. Клещев, А. С. Необогатенные системы логических соотношений / А. С. Клещев, И. Л. Артемьева // Научно-техническая информация. Сер. 2. – 2000. – Ч. 1, № 7; Ч. 2, № 8.
11. Кучеренко Е. И. Некоторые аспекты анализа развития нечетких онтологий / Е. И. Кучеренко, Д. А. Павлов // Искусственный интеллект. Донецк, 2005. – С. 162–169.
12. Ломов, П. А. Преобразование owl-онтологий для визуализации и использования в качестве основы пользовательского интерфейса /

П. А. Ломов, М. Г. Шишаев, В. В. Диковицкий // *Онтология проектирования*. – 2012. – № 3. – С. 49–61.

13. Московиченко, А. Л. *Дерево целей инженерной деятельности* / А. Л. Московиченко // *Кибернетика и вуз*. – 1987. – Вып. 13. – С. 123–129.

14. Нефедов, В. Н. *Курс дискретной математики* / В. Н. Нефедов, В. Ж. Осипова. – М. : МАИ, 1992. – 367 с.

15. Никитина, С. Е. *Семантический анализ языка науки* / Е. Никитина. – М. : Наука, 1987.

16. Новиков, Ф. А. *Символический искусственный интеллект: математические основы представления знаний : учеб. пособие для акад. бакалавриата* / Ф. А. Новиков. – М. : Юрайт, 2018. – 278 с. – (Серия: Бакалавр. Академический курс. Модуль.). – ISBN 978-5-534-00734-3. – Режим доступа: www.biblio-online.ru/book/01E78622-B773-43C9-A583-91B73B00F44D.

17. Овдей, О. М. *Обзор инструментов инженерии онтологий* / О. М. Овдей, Г. Ю. Проскудина // *Журнал ЭБ*. – 2004. – Вып. 4. – Режим доступа: <http://www.elbib.ru/index.phtml?page=elbib/rus/journal/2004/part4/op>.

18. Поспелов, Д. А. *Прикладная семиотика и искусственный интеллект* / Д. А. Поспелов // *Программные продукты и системы*. – 1996. – № 3.

19. *Разработка онтологий 101: руководство по созданию Вашей первой онтологии [Электронный ресурс]*. – Режим доступа: http://protege.stanford.edu/publications/ontology_development/ontology101.html.

20. Смирнов, А. В. *Онтологии в системах искусственного интеллекта: способы построения и организации* / А. В. Смирнов, М. П. Пашкин, Н. Г. Шилов, Т. В. Левашова // *Новости искусственного интеллекта*. – 2002. – № 1.

21. Соловьев, В. Д. *Онтологии и тезаурусы : учеб. пособие* / В. Д. Соловьев, Б. В. Добров, В. В. Иванов, Н. В. Лукашевич. – Казань ; М., 2006. – 173 с.

22. Сумароков, Л. Н. *В целях оптимизации обучения* / Л. Н. Сумароков, А. Г. Романенко, Э. В. Мухин // *Вестник высшей школы*. – 1968. – № 2.

23. Филиппов, А. А. *Применение нечеткой базы знаний проблемной области в задаче поиска архитектурно подобных программных проектов* / А. А. Филиппов, В. С. Мошкин, Г. Ю. Гуськов, Н. Г. Ярушкина // *Нечеткие системы и мягкие вычисления*. – 2017. – 12:2. – С. 107–120.

24. Юсавичене, П. *Теория и практика модульного обучения* / П. Юсавичене. – Каунас : Швиеса, 1989. – 272 с.

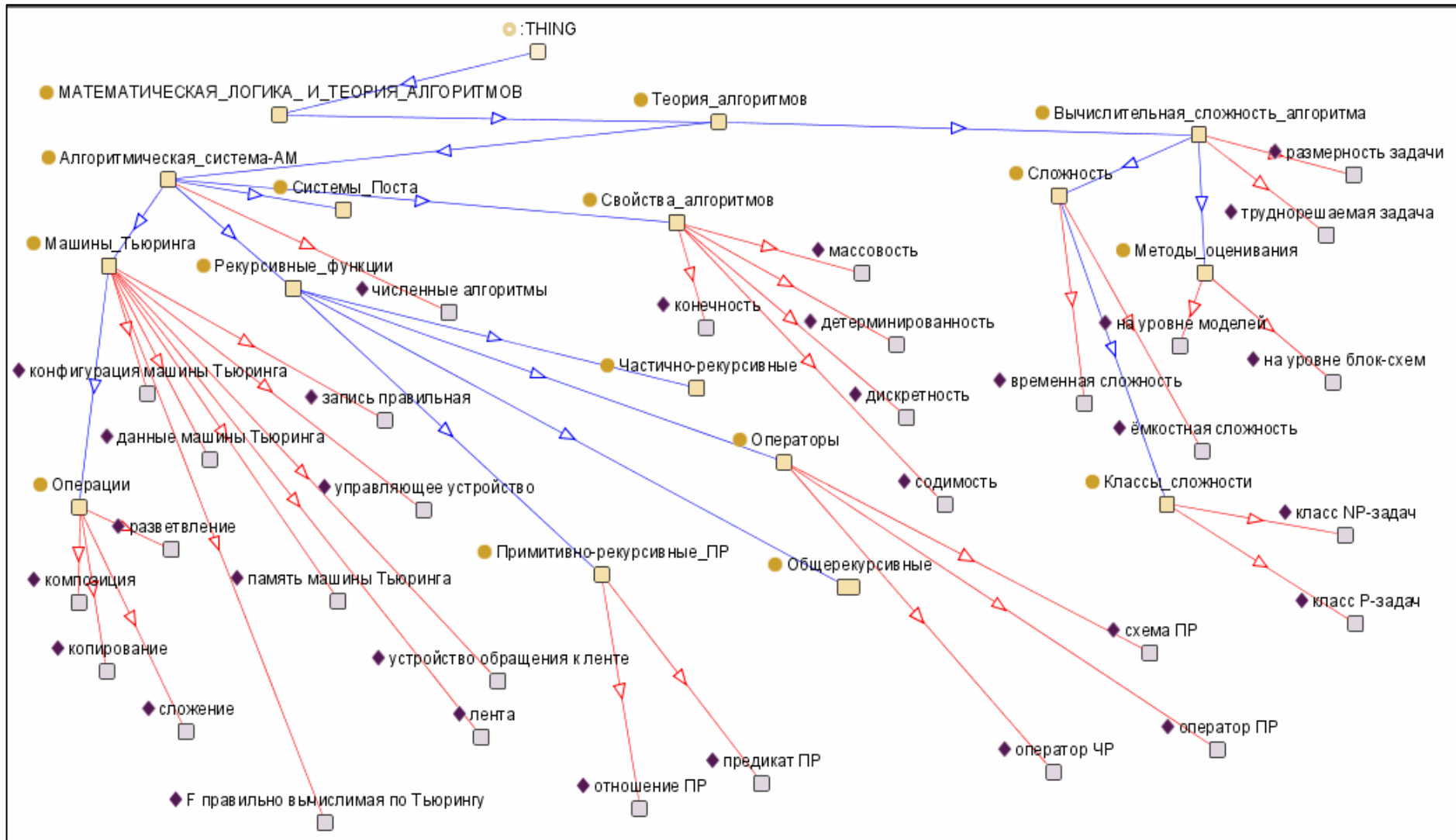
ПРИЛОЖЕНИЕ

Приложение 1

Перечень ключевых слов

Визуализация онтологии
Жизненный цикл онтологии
Иерархическая система терминов
Инструментарий построения онтологий
Классификация знаний
Классификация онтологий
Классификация семантических сетей
Лексикографическое (словарное) описание
Логико-интуитивное описание
Логическая модель представления знаний
Методы построения онтологий
Модели знаний
Модели онтологий
Модели представления знаний
Онтологические ресурсы
Онтология верхней зоны
Онтология нижней зоны
Онтология предметной области
Онтология средней зоны
Представление знаний.
Программная реализация онтологии
Программные инструменты для построения и работы с онтологиями
Продукционные модели знаний
Свойства знаний
Семантическая сеть
Семантические сети
Стандарт онтологического исследования IDEF5
Структура онтологии
Формализация знаний
Формально-языковое описание
Фрейм
Цели создания онтологий
Языки описания онтологий

Онтология «Теория алгоритмов»



Учебно-теоретическое издание

Лутошкина Наталья Васильевна

МОДЕЛИ ЗНАНИЙ И ОНТОЛОГИИ

Учебное пособие

