

Г. С. ОСИПОВ

**МЕТОДЫ
ИСКУССТВЕННОГО
ИНТЕЛЛЕКТА**



Осипов Г.С.

Методы искусственного интеллекта



МОСКВА
ФИЗМАТЛИТ®

УДК 519,816
ББК 32,813
О 74



*Издание осуществлено при поддержке
Российского фонда фундаментальных
исследований по проекту 10-07-07014*

Осипов Г. С. **Методы искусственного интеллекта.** — М.: ФИЗМАТЛИТ, 2011. — 296 с. — ISBN 978-5-9221-1323-6.

Монография содержит изложение основных методов искусственного интеллекта: методов представления знаний, методов моделирования рассуждений, методов моделирования поведения, методов обучения и приобретения знаний интеллектуальными системами. Весь материал излагается с единых позиций. В качестве основных средств используются системы правил в их общем виде и семантические сети; особое внимание уделено неоднородным семантическим сетям, а также методам планирования и моделирования целенаправленного поведения. Системы правил использованы и для описания этих методов. Описаны методы автоматизации приобретения знаний, для чего применен аппарат неоднородных семантических сетей.

Для специалистов, аспирантов и студентов старших курсов университетов, изучающих информатику и информационные технологии.

ОГЛАВЛЕНИЕ

| | |
|---|-----------|
| Предисловие автора | 7 |
| Введение | 9 |
| Глава 1. Методы представления знаний | 23 |
| Введение | 23 |
| 1.1. Формальные языки и формальные системы | 23 |
| 1.1.1. Язык исчисления предикатов первого порядка (23). | |
| 1.1.2. Элементы исчисления предикатов первого порядка (25). | |
| 1.1.3. Формальные системы (27). 1.1.4. Алгебраические системы (28). 1.1.5. Интерпретация (28). 1.1.6. Выполнимость и истинность (29). | |
| 1.2. Системы, основанные на правилах, или продукционные системы . . | 31 |
| 1.2.1. Правила для представления знаний (32). 1.2.2. Рабочая память (32). 1.2.3. Стратегии управления (33). 1.2.4. Разрешение конфликтного множества правил (35). 1.2.5. Пример (39). | |
| 1.3. Семантические сети для представления знаний | 43 |
| 1.3.1. Простые и расширенные семантические сети (43). | |
| 1.3.2. Универсум Эрбрана и семантические сети (45). 1.3.3. Неоднородные семантические сети (48). | |
| 1.4. Совместность | 54 |
| 1.4.1. Вектор совместности событий (55). 1.4.2. Матрицы совместности элементов (55). | |
| 1.5. Представление знаний в системах фреймов | 57 |
| 1.5.1. Фреймы (57). 1.5.2. Системы фреймов (58). 1.5.3. Основная вычислительная задача в системе фреймов (59). | |
| 1.6. Элементы дескриптивной логики | 59 |
| 1.6.1. Основные понятия (61). 1.6.2. База знаний в дескриптивной логике (61). 1.6.3. Рассуждения в дескриптивной логике (62). 1.6.4. Семейство языков дескриптивных логик (65). 1.6.5. Отображение дескриптивной логики в логику первого порядка (66). 1.6.6. Дескриптивная логика с конкретным доменом (67). 1.6.7. Правила вывода (69). | |

| | |
|--|-----|
| Глава 2. Методы автоматизации рассуждений | 72 |
| Введение | 72 |
| 2.1. Автоматизация дедуктивных рассуждений. Поиск доказательств теорем методом резолюций | 75 |
| 2.1.1. Скулемовская стандартная форма (76). 2.1.2. Метод резолюций для исчисления высказываний (80). 2.1.3. Метод резолюций для исчисления предикатов первого порядка (82). | |
| Примеры | 84 |
| 2.2. Индуктивные рассуждения | 86 |
| 2.2.1. Понятие квазиаксиоматической теории (86). 2.2.2. Немонотонные рассуждения (87). 2.2.3. ДСМ — метод индуктивного вывода (89). | |
| 2.3. Аргументационные рассуждения. | 93 |
| 2.4. Рассуждения на основе прецедентов | 97 |
| 2.4.1. Метрики на множестве прецедентов (97). 2.4.2. Согласование прецедентов (102). 2.4.3. Предпочтения и глобальная релевантность (103). 2.4.4. Адаптация прецедентов (104). | |
| Глава 3. Методы интеллектуального планирования | 107 |
| Введение | 107 |
| 3.1. Хронология методов интеллектуального планирования | 109 |
| 3.2. Планирование как поиск доказательства теорем | 110 |
| 3.3. Планирование в пространстве состояний | 111 |
| 3.3.1. Постановка задачи STRIPS-планирования (111). 3.3.2. Алгоритм STRIPS (113). 3.3.3. Неполнота алгоритма STRIPS (113). 3.3.4. Вычислительная сложность задачи STRIPS-планирования (116). 3.3.5. Языковые средства описания доменов планирования (118). | |
| 3.4. Поиск в пространстве планов. | 118 |
| 3.4.1. Основная идея (118). 3.4.2. Алгоритм SNLP (121). 3.4.3. Принцип малой связности (122). | |
| 3.5. Планирование как задача удовлетворения ограничений | 122 |
| 3.5.1. Постановка задачи удовлетворения ограничений (122). 3.5.2. Синтез планов на основе техники прямого распространения ограничений (123). 3.5.3. Алгоритм GraphPlan (127). | |
| 3.6. Планирование на основе прецедентов | 130 |
| 3.6.1. Общая схема метода планирования на основе прецедентов (130). 3.6.2. Методы адаптации прецедентов (131). 3.6.3. Некоторые системы планирования, основанного на прецедентах (133). | |
| Глава 4. Интеллектуальные динамические системы | 150 |
| Введение | 150 |
| 4.1. Уточнение постановки задачи. Правила | 152 |
| 4.2. Стратегии применения правил. Состояния и траектория системы . . | 153 |
| 4.3. Управляемые динамические системы, основанные на правилах . . . | 155 |

| | | |
|---|---|------------|
| 4.3.1. Возмущения (156). | 4.3.2. Управление как способ компенсации возмущений (156). | |
| 4.4. Особенности баз знаний динамических систем, основанных на правилах | | 157 |
| 4.4.1. Синтез обратной связи по траектории (158). | 4.4.2. Стратегия синтеза обратной связи по состояниям (161). | |
| 4.5. Элементы теории управляемости интеллектуальных динамических систем | | 163 |
| 4.6. Примеры интеллектуальных динамических систем. | | 165 |
| 4.6.1. Описание агентов (165). | 4.6.2. Функции и формулы (169). | |
| 4.6.3. Замыкание состояний. Аксиомы (172). | 4.6.4. Крестообразный перекресток равнозначных однополосных дорог (174). | |
| 4.6.5. Движение по попутной проезжей части (177). | 4.6.6. Стратегия применения правил (179). | |
| 4.6.7. Модели корабля, станции и управления. Общее описание (181). | 4.6.8. Параметры модели орбитальной станции (183). | |
| 4.6.9. Параметры процесса стыковки (184). | 4.6.10. Правила замыкания (184). | |
| 4.6.11. Правила переходов (185). | 4.6.12. Управление. Подцели и зоны управления (186). | |
| 4.6.13. Правила выбора цели (186). | 4.6.14. Правила управления (187). | |
| 4.6.15. Результаты модельного эксперимента (189). | | |
| Глава 5. Приобретение знаний и машинное обучение | | 192 |
| Введение | | 192 |
| 5.1. Источники знаний для интеллектуальных систем | | 193 |
| 5.2. Прямые методы приобретения знаний | | 194 |
| 5.2.1. Имена (194). | 5.2.2. Признаки (195). | |
| 5.2.3. Виды семантических связей (196). | 5.2.4. Типы семантических связей (200). | |
| 5.2.5. Метод интервью (204). | | |
| 5.3. Приобретение знаний из примеров | | 214 |
| 5.3.1. Задачи машинного обучения (214). | 5.3.2. Поиск (218). | |
| 5.3.3. Индуктивный алгоритм построения деревьев решений (TDIDT) (223). | 5.3.4. Последовательное покрытие: AQ-обучение (228). | |
| 5.3.5. Оценка обучающих алгоритмов (232). | 5.3.6. Машинное обучение в языке исчисления предикатов первого порядка (234). | |
| 5.4. Искусственные нейронные сети и их обучение | | 244 |
| 5.4.1. Область применения искусственных нейронных сетей (244). | 5.4.2. Достоинства и недостатки искусственных нейронных сетей (246). | |
| 5.4.3. Персептрон (247). | 5.4.4. Процедура обратного распространения (247). | |
| 5.4.5. Сети встречного распространения (249). | 5.4.6. Сети с обратными связями (251). | |
| Глава 6. Приобретение знаний и анализ текстов | | 255 |
| Введение | | 255 |
| 6.1. Коммуникативная грамматика русского языка. | | 266 |
| 6.1.1. Минимальные синтаксические единицы — синтаксемы (266). | 6.1.2. Категориальная семантика лексических единиц (267). | |

| | |
|---|-----|
| 6.2. Реляционно-ситуационный анализ текста | 272 |
| 6.2.1. Морфологический и синтаксический анализ (272). | |
| 6.2.2. Установление значений синтаксем (274). | |
| 6.3. Установление значений синтаксем в безглагольных предложениях | 279 |
| 6.3.1. Выбор объектов, признаков и свойств (279). 6.3.2. Обнару- жение правил установления значений синтаксем (283). | |
| 6.4. Установление отношений на множестве синтаксем | 284 |
| Список литературы | 288 |

Предисловие автора

Настоящая книга, как следует из ее названия, посвящена методам искусственного интеллекта. Мне приходилось не раз заявлять, что искусственный интеллект является экспериментальной наукой. Тем не менее, книга посвящена изложению основ теоретических методов искусственного интеллекта.

Побудительные мотивы написать книгу такого свойства возникли у меня давно. Разумеется, в искусственном интеллекте возникли и развивались подходы, вобравшие в качестве своих органических частей методы математической логики, теории алгоритмов, комбинаторики, распознавания образов, когнитивной психологии и лингвистической семантики. Понятно, что по этой причине существует соблазн назвать книгу о методах искусственного интеллекта (как случается иногда) книгу, содержащую изложение всех имеющих то или иное отношение к делу методов других дисциплин. Но тогда возникает вопрос: в чем заключается содержание, собственно, методов искусственного интеллекта?

Цель написания книги состояла в ответе на этот последний вопрос. Сегодня искусственный интеллект является достаточно разветвленной наукой и продолжает развиваться. За пятьдесят лет своего существования как науки он накопил значительный арсенал средств и собственных подходов и, поэтому, книга содержит изложение именно основ методов искусственного интеллекта, к коим относятся методы представления знаний, методы моделирования рассуждений, методы моделирования поведения, методы приобретения знаний и, конечно же, не только они. На протяжении всей книги материал излагался, по возможности, с единых позиций при сохранении приемлемого уровня строгости. При этом всюду, где это оказалось возможным, привлекались математические средства, именно в качестве средств изложения. На отбор материала оказали существенное влияние как мое понимание того, что сегодня следует, безусловно, включить в арсенал методов искусственного интеллекта, так и мои интересы. Поэтому, как перечень, так и содержание указанных разделов не претендует на полноту и «равномерность» представления материала, хотя все необходимое для понимания современной проблематики искусственного интеллекта в книге имеется.

При изложении способов представления знаний в качестве основных средств использовались системы правил в их общем виде

и семантические сети; особенное внимание уделено неоднородным семантическим сетям. Представление знаний в виде систем правил использовалось и при описании методов планирования и моделирования поведения.

Проблематике моделирования рассуждений было уделено не очень много внимания, так как на русском языке ей посвящена обширная литература. Напротив, большое внимание в книге уделено методам планирования целенаправленного поведения, так как в монографической литературе на русском языке эти вопросы освещены, повидимому, впервые. Существенное внимание уделено интеллектуальным динамическим системам и основанным на них методам моделирования целенаправленного поведения. В последних двух главах описаны методы автоматизации приобретения знаний, при этом, по крайней мере в пятой главе, активно применяется аппарат неоднородных семантических сетей.

Здесь надо сказать, что в основе части книги лежат изданные в 2009 г. мои «Лекции по искусственному интеллекту», правда существенно переработанные и расширенные. Разумеется, книга содержит и материал, никаким образом в «Лекциях» не представленный.

Замечу, что книга совершенно не касается архитектур интеллектуальных систем, их инструментальных средств и прикладных интеллектуальных систем. Даже в тех случаях, когда нельзя было обойтись без кратких сведений о тех или иных системах, речь шла, главным образом, об используемых в них идеях и методах. Архитектуры интеллектуальных систем и их инструментальных средств — обширная и важная область деятельности и она заслуживает отдельной книги, которую следовало бы тогда назвать, например «Интеллектуальные системы» или «Системы искусственного интеллекта».

В заключение хотелось бы выразить признательность Вадиму Николаевичу Вагину, который не только прочел рукопись и указал ряд недостатков, но и дал ценные советы по включению дополнительного материала, что, надеюсь, способствовало улучшению книги.

Книга предназначена как специалистам, так и аспирантам и студентам старших курсов университетов.

Автор

Введение

Научная дисциплина под названием «искусственный интеллект» входит в комплекс *компьютерных наук*, а создаваемые на основе ее результатов технологии относятся к *информационным технологиям*.

Основной целью исследований в искусственном интеллекте является получение методов, моделей и программных средств, позволяющих искусственным устройствам реализовать целенаправленное поведение и разумные рассуждения.

На этом пути возникает ряд задач, важнейшая особенность которых состоит в том, что, в большинстве случаев, до получения результата решения задачи не известен алгоритм ее решения. Например, точно неизвестно, как происходит узнавание изображения, понимание текста, поиск доказательства теоремы, построение плана действий. Алгоритмы решения таких задач являются, обычно, одним из результатов их решения. В частности, алгоритм доказательства теоремы можно извлечь из ее доказательства.

Человек решает задачи такого рода, используя, в числе прочего, свои знания и компетентность. В то же время, у искусственных устройств (например, компьютеров), привлекаемых для решения таких задач, обычно отсутствует достаточный уровень начальной компетентности. Это означает, что одной из основных задач искусственного интеллекта является разработка механизмов переноса компетентности, т.е. обучения искусственных устройств (или, как принято говорить, приобретения знаний). При этом считается, что результаты решения задач и сам ход решения должны быть транспарентны — «прозрачны» для человека и допускать объяснение.

Таким образом, важной характеристикой представления результатов и хода решения задач искусственного интеллекта является их, в значительной степени, *вербальный характер*.

Создавая те или иные начальные компьютерные представления и модели, исследователь или разработчик сравнивает их поведение между собой и с примерами решения тех же задач специалистом в соответствующей области, модифицирует их на основе этого сравнения, пытаясь добиться лучшего соответствия результатов. Таким образом, искусственный интеллект представляет собой экспериментальную науку.

Для того чтобы модификация программ улучшала результаты их работы, надо иметь разумные исходные представления и модели. Такие

представления и модели доставляют психологические исследования, в частности, исследования в области когнитивных наук.

Первые исследования, относящиеся к искусственному интеллекту, были предприняты почти сразу же после появления первых вычислительных машин.

В 1954 г. американский исследователь А. Ньюэлл (A. Newel) решил написать программу для игры в шахматы. Этой идеей он поделился с аналитиками корпорации «РЭНД» (RAND Corporation (J. Show) и Г. Саймоном (H. Simon), которые предложили Ньюэллу свою помощь. В качестве теоретической основы такой программы было решено использовать метод, предложенный в 1950 г. Клодом Шенноном (K. Shannon), основателем теории информации. Точная формализация этого метода была выполнена Аланом Тьюрингом (Alan Turing). Он же промоделировал его вручную.

К работе была привлечена группа голландских психологов под руководством А. Де Гроота (A. de Groot), изучавших стили игры выдающихся шахматистов. Через два года совместной работы этот коллектив создал язык программирования ИПЛ1 — по-видимому первый символьный язык обработки списков. Вскоре была написана и первая программа, которую можно отнести к достижениям в области искусственного интеллекта. Эта была программа «Логик-Теоретик» (1956 г.), предназначенная для автоматического поиска доказательств теорем в исчислении высказываний.

Собственно же программа для игры в шахматы, **NSS**, была завершена в 1957 г. В основе ее работы лежали так называемые *эвристики* (правила выбора в отсутствии теоретических оснований) и описания целей. Управляющий алгоритм пытался уменьшить различия между оценками текущей ситуации и оценками цели или одной из подцелей.

В 1960 г. та же группа на основе принципов, использованных в **NSS**, написала программу, которую ее создатели назвали **GPS** (General Problem Solver) — универсальный решатель задач. **GPS** могла справиться с рядом головоломок, вычислять неопределенные интегралы, решать некоторые другие задачи. Эти результаты привлекли внимание специалистов в области вычислений. Появились программы автоматического поиска доказательств теорем из планиметрии и решения алгебраических задач.

Джона Маккарти (J. McCarty) из Стэнфорда заинтересовали математические основы этих результатов и вообще символьных вычислений. В результате в 1963 г. он разработал язык ЛИСП (LISP, от List Processing), основу которого составило использование единого спискового представления для программ и данных, применение выражений для определения функций, скобочный синтаксис.

В это же время в СССР, в основном, в Московском государственном университете и Академии наук СССР был выполнен ряд пионерских исследований, возглавленных Вениамином Пушкиным и Дмитрием Поспеловым [1], целью которых было выяснение того, как же, в действительности, человек решает переборные задачи.

В качестве полигона для этих исследований были выбраны различные математические игры, в частности, игра «15» и игра «5», а в качестве инструментального метода исследования — регистрация движения глаз или гностическая динамика. Основными методами регистрации движения глаз были электроокулограмма и использование присоски, помещаемой на роговицу.

Цель каждой такой игры заключается в переходе от некоторой исходной ситуации к конечной. Переходы осуществляются путем последовательного перемещения фишек по горизонталям и вертикалям на свободное поле.

Возьмем, например, игру «5», исходная и конечная ситуации в которой выглядят, соответственно, следующим образом:

| | | |
|---|---|---|
| 2 | 3 | 5 |
| | 1 | 4 |

и

| | | |
|---|---|---|
| 1 | 2 | 3 |
| | 4 | 5 |

Оптимальным образом задача решается за шесть ходов, которые соответствуют перемещениям фишек 1, 4, 5, 3, 2, 1. Решение было бы намного сложнее, если бы на первом ходу двигалась бы, например, фишка 2, или на втором ходу — фишка 3. Понятно, что задача может быть представлена в виде дерева (или лабиринта), корнем которого является исходная ситуация, а перемещение каждой фишки приводит в новую вершину. Все ситуации являются при таком подходе вершинами графа или точками на дереве игры и именно они являются теми элементами, из которых строится «модель мира». Два элемента связывает ход — преобразование одной ситуации в другую.

Такая модель игры приводит, вообще говоря, к полному перебору или «лабиринту» вариантов и составляет основу так называемой лабиринтной гипотезы решения переборных задач.

С другой стороны, анализ экспериментальных данных позволил вычленить два вида изменений параметров гностической динамики в процессе обучения решению задачи. А именно, изменения ряда пара-

метров уже при решении второй или третьей из множества однотипных задач у одной из групп испытуемых характеризуется появлением точки излома.

К числу этих параметров относятся время решения задачи, количество осмотров условий, количество осмотров цели, общее количество осмотров, плотность осмотра и отношение числа осмотров условий к числу осмотров цели. У другой же группы испытуемых таких изменений не происходит.

Так, например, отношение числа осмотров условий задачи к числу осмотров цели у первой группы испытуемых претерпевает излом после решения второй задачи и продолжает уменьшаться при решении числа последующих задач. У второй группы испытуемых уменьшения этого отношения не происходит. То же относится и к времени решения задач.

Анализ и других экспериментальных данных подтвердил существование некоторых общих тенденций в динамике обучения решению задач.

Есть все основания полагать, что основным фактором, влияющим на временные характеристики этого процесса у первой группы испытуемых, является момент понимания эквивалентности задач или транспозиции (переноса) отношений, сформированных в ходе решения первых задач.

Изучение всей совокупности данных позволяет связать формирование подобной системы отношений со временем решения второй и последующих задач — именно тогда формируется то общее, что связывает первую и вторую задачи. Осознание общности и, следовательно, «открытие» эквивалентности происходит при столкновении с третьей задачей.

Сопоставление экспериментальных данных свидетельствует также о том, что соотнесение различных ситуаций связано между собой посредством такого когнитивного компонента, как анализ цели. Иначе говоря, анализ исходной ситуации управляется анализом цели и процессом соотнесения исходной и конечной ситуаций. Таким образом, моделирование исходной ситуации является управляемым компонентом, а установленные в конечной ситуации отношения являются регулятором этого моделирующего процесса. Сама же модель исходной ситуации рассматривается с точки зрения ситуации конечной.

Эту модель можно также изобразить в виде графа, но вершинами этого графа будут не ситуации, как при использовании «лабиринта» вариантов, а элементы ситуаций. Ребрами, соединяющими вершины, будут не переходы из одной ситуации в другую, а те отношения, которые были выявлены на множестве этих элементов с помощью гностической динамики. Эти соображения составили основу модельной

гипотезы решения переборных задач и привели к появлению в 1964 г. языка (и метода) ситуационного управления [2].

К исследованиям в области искусственного интеллекта стали проявлять интерес и логики. В том же 1964 г. была опубликована работа ленинградского логика Сергея Маслова «Обратный метод установления выводимости в классическом исчислении предикатов», в которой впервые предлагался метод автоматического поиска доказательства теорем в исчислении предикатов.

На год позже (в 1965 г.) в США появляется работа Дж. А. Робинсона (J. A. Robinson), посвященная иному методу автоматического поиска доказательства теорем в исчислении предикатов первого порядка. Этот метод был назван методом резолюций и послужил отправной точкой для создания нового языка программирования со встроенной процедурой логического вывода — языка **Пролог (Prolog)** в 1971 г.

В 1966 г. в СССР Валентином Турчиным был разработан язык рекурсивных функций Рефал, предназначенный для описания языков и разных видов их обработки. Хотя он и был задуман как алгоритмический метаязык, но для пользователя это был, подобно ЛИСПу и **Прологу**, язык обработки символьной информации.

В конце 60-х годов появились первые игровые программы, системы для элементарного анализа текста и решения некоторых математических задач (планиметрии, взятия неопределенных интегралов). В возникавших при этом сложных переборных проблемах количество перебираемых вариантов резко снижалось применением всевозможных эвристик и «здравого смысла». Такой подход стали называть *эвристическим программированием*. Дальнейшее бурное развитие эвристического программирования шло по пути усложнения алгоритмов и улучшения эвристик. Однако вскоре стало ясно, что существует некоторый предел, за которым никакие улучшения эвристик и усложнения алгоритма не повысят качества работы системы и, главное, не расширят ее возможностей. Программа, которая играет в шахматы, никогда не будет играть в шашки или карточные игры.

Постепенно исследователи стали понимать, что всем ранее созданным программам недостает самого важного — знаний в соответствующей области. Специалисты, решая задачи, достигают высоких результатов, благодаря своим знаниям и опыту; если программы будут обращаться к знаниям и применять их, то они тоже достигнут высокого качества работы.

Это понимание, возникшее в начале 70-х годов, по существу, означало качественный скачок в работах по искусственному интеллекту.

Основополагающие соображения на этот счет высказал в 1977 г. на 5-й Объединенной конференции по искусственному интеллекту американский ученый Э. Фейгенбаум (E. Feigenbaum).

Уже к середине 70-х годов появляются первые прикладные интеллектуальные системы, использующие различные способы представления знаний для решения задач — *экспертные системы*. Одной из первых была экспертная система **DENDRAL**, разработанная в Станфордском университете и предназначенная для порождения формул химических соединений на основе спектрального анализа. В настоящее время **DENDRAL** [3] поставляется покупателям вместе со спектрометром. Система **MYCIN** предназначена для диагностики и лечения инфекционных заболеваний крови. Система **PROSPECTOR** прогнозирует залежи полезных ископаемых. Имеются сведения о том, что с ее помощью были открыты залежи молибдена, ценность которых превосходила 100 миллионов долларов. Система оценки качества воды, реализованная на основе российской технологии **SIMER + MIR** [4] обнаружила причины превышения предельно допустимых концентраций загрязняющих веществ в Москве-реке в районе Серебрянного Бора. Система **CASNET** предназначена для диагностики и выбора стратегии лечения глаукомы и т. д.

В настоящее время разработка и реализация экспертных систем превратилась в инженерную дисциплину. Научные же исследования сосредоточены в ряде сформировавшихся и формирующихся направлений; некоторые из них перечислены ниже.

1. Представление знаний. Представление знаний (knowledge representation) — одно из наиболее сформировавшихся направлений искусственного интеллекта. Традиционно к нему относилась разработка формальных языков и программных средств для отображения и описания так называемых *когнитивных структур*¹⁾. Сегодня к представлению знаний причисляют также исследования по *дескриптивной логике, логикам пространства и времени, онтологиям*.

Пространственные логики позволяют описывать конфигурацию пространственных областей, объектов в пространстве; изучаются также семейства пространственных отношений. В последнее время эта область, из-за тесной связи с прикладными задачами, становится доминирующей в исследованиях по представлению знаний. Например, для задач роботики важно уметь по изображению некоторой сцены восстановить ее вербальное (формальное) описание, для того чтобы далее это описание использовать, например, для планирования действий робота.

Объектами дескриптивной логики являются так называемые *концепты* (базовые структуры для описания объектов в интеллектуальных системах) и, связанные в единое целое, множества концептов (аг-

¹⁾ Структур человеческого сознания, отражающих представление личности о действительности

регированные объекты). Дескриптивная логика вырабатывает методы работы с такими сложными концептами, методы рассуждений об их свойствах и выводимости на них. Дескриптивная логика может быть использована, кроме того, для построения объяснительной компоненты базы знаний.

Онтологические исследования посвящены способам концептуализации знаний и методологическим соображениям о разработке инструментальных средств для анализа знаний.

2. Автоматизация рассуждений. В основе автоматизации рассуждений лежат различные способы представления знаний.

Автоматизация рассуждений, помимо автоматизации дедуктивных рассуждений, включает: автоматизацию индуктивных рассуждений, автоматизацию рассуждений на основе прецедентов (case-based reasoning, **CBR**), на основе аргументации, на основе ограничений, автоматизацию рассуждений с неопределенностью, рассуждения о действиях и изменениях, автоматизацию немонотонных рассуждений и др. Остановимся кратко на некоторых из них.

Рассуждения на основе прецедентов. Здесь главные проблемы — «фокусировка поиска» на использовании прошлого опыта, оценка сходства прецедентов, поиск алгоритмов адаптации прецедентов и технологии визуализации.

Пусть задано множество прецедентов как множество пар **⟨СЛУЧАЙ, РЕШЕНИЕ⟩**, множество зависимостей между различными атрибутами **СЛУЧАЕВ** и **РЕШЕНИЙ**, а также целевая проблема **ЦЕЛЬ**. Для возникающей новой ситуации («нового случая») требуется найти пару **⟨НОВЫЙ СЛУЧАЙ, ИСКОМОЕ РЕШЕНИЕ⟩**, которая решает целевую проблему.

Алгоритмы для таких задач основаны обычно на сравнении прецедентов с новым случаем (в какой-либо метрике), с использованием зависимостей между атрибутами случаев и атрибутами решения. Такие зависимости могут задаваться человеком при построении базы случаев, или обнаруживаться в базе случаев автоматически. При поиске решения для целевой проблемы выполняется адаптация уже имеющихся в базе прецедентов решения. Для этой адаптации и используются означенные зависимости.

Важной проблемой рассуждений на основе прецедентов (**CBR**) является проблема выбора подходящего прецедента. Естественно искать подходящий прецедент в той области пространства поиска, где находятся решения сходных проблем. Но как определить, какие именно решения надо считать сходными?

Одна из гипотез, называемая гипотезой компактности, состоит в том, что сходство проблем налагает ограничения на сходство соот-

ветствующих решений в форме слабой связи между ними. Это обстоятельство и используется для ограничения области поиска решений.

Методы **CBR** уже применяются в множестве прикладных задач — в медицине, управлении проектами, для анализа и реорганизации среды, для разработки товаров массового спроса с учетом предпочтений разных групп потребителей, и т. д. Следует ожидать приложений методов **CBR** для задач интеллектуального поиска информации, электронной коммерции (предложение товаров, создание виртуальных торговых агентств), планирования поведения в динамических средах, компоновки, конструирования, синтеза программ.

Автоматизация рассуждений на основе ограничений. Наиболее интересны здесь задачи моделирования рассуждений, основанных на процедурных динамических ограничениях. Они мотивированы сложными актуальными задачами — например, планированием в реальной обстановке. Решение ищется в области значений, удовлетворяющих заданные ограничения.

Под задачей удовлетворения ограничений понимается четверка множеств: множество переменных, множество соответствующих областей переменных, множество ограничений на переменные и множество отношений над областями. Решением проблемы удовлетворения ограничений называется набор значений переменных, удовлетворяющих ограничениям на переменные, такой, что при этом области, которым принадлежат эти значения, удовлетворяют отношениям над областями.

Задача удовлетворения динамических ограничений есть последовательность задач удовлетворения ограничений, в которой каждая последующая задача есть ограничение предыдущей. Эти задачи по смыслу близки задачам динамического программирования. Они связаны также с интервальной алгеброй.

Немонотонные рассуждения. К немонотонным рассуждениям относятся исследования по логике умолчаний (default logic), по логике «пересматриваемых» (defeasible) рассуждений, логике программ, теоретико-аргументационной характеристике логик с отменами, характеристике логик с отношениями предпочтения, построению эквивалентных множеств формул для логик с очерчиванием (circumscription) и некоторые другие. Такого рода модели возникают при реализации индуктивных рассуждений, например, по примерам; связаны они также с задачами машинного обучения и некоторыми иными задачами. В частности, в задачах моделирования рассуждений на основе индукции источником первоначальных гипотез служат примеры. Если некоторая гипотеза H возникла на основе N положительных примеров (например, экспериментального характера), то никто не может дать гарантии, что в базе данных или в поле зрения алгоритма не окажется

$N + 1$ -й пример, опровергающий гипотезу (или меняющий степень ее истинности). Это означает, что ревизии должны быть подвержены и все следствия гипотезы H .

Рассуждения о действиях и изменениях. Большая часть работ в этой области посвящена применениям ситуационного исчисления — формализма, предложенного Джоном Маккарти в 1968 г. для описания действий, рассуждений о них и эффектов действий. Для преобразования плана поведения робота в исполняемую программу, достигающую с некоторой вероятностью фиксированной цели, вводится специальное логическое исчисление, основанное на ситуационной логике. Для этой логики предложены варианты реализации на языке **pGOLOG** — версии языка **GOLOG**, содержащей средства для введения вероятностей.

Активно исследуются логики действий, применение модальных логик для рассуждений о знаниях и действиях.

Рассуждения с неопределенностью. В основе таких рассуждений находится использование байесовского формализма в системах правил и сетевых моделях. Байесовские сети — это статистический метод обнаружения закономерностей в данных. Для этого используется первичная информация, содержащаяся либо в сетевых структурах, либо в базах данных. Под сетевыми структурами понимается в этом случае множество вершин и отношений на них, задаваемое с помощью ребер. Содержательно, ребра интерпретируются как причинные связи. Всякое множество вершин Z , представляющее все пути между некоторыми двумя иными вершинами X и Y , соответствует условной зависимости между этими двумя последними вершинами.

Далее задается некоторое распределение вероятностей на множестве переменных, соответствующих вершинам этого графа и полученная, но минимизированная (в некотором смысле) сеть называется байесовской сетью.

На такой сети можно использовать так называемый байесовский вывод, т.е. для вычисления вероятностей следствий событий можно использовать (с некоторой натяжкой) формулы теории вероятностей. Иногда рассматриваются так называемые гибридные байесовские сети, с вершинами которых связаны как дискретные, так и непрерывные переменные. Байесовские сети часто применяются для моделирования технических систем.

3. Приобретение знаний, машинное обучение и автоматическое порождение гипотез. Работы в области приобретения знаний интеллектуальными системами были и остаются важнейшим направлением теории и практики искусственного интеллекта. Целью этих работ является создание методологий, технологий и программных средств

переноса знаний и компетентности в базу знаний системы. При этом в качестве источников знаний выступают эксперты (т. е. высококвалифицированные специалисты предметных областей), тексты и данные, например, хранимые в базах данных.

В соответствии с этим, развиваются различные методы приобретения знаний.

Машинному обучению в мире уделяется большое внимание. Существует множество методов машинного обучения, среди самых распространенных — методы конструктивной индукции, методы построения деревьев решений, методы ближайших соседей и др. Исходной информацией, например, для построения дерева решений является множество примеров. С каждой вершиной дерева ассоциируется наиболее (на текущем шаге) частотный класс примеров. На следующем шаге этот принцип рекурсивно применяется к текущей вершине, т. е. множество примеров, связанных с текущей вершиной, также разбивается на подклассы. Алгоритм завершает свою работу либо при удовлетворении некоторого критерия, либо при исчерпании подклассов (если они заданы).

Активно исследуются методы *обучения причинам действий*. Иногда говорят о так называемой *теории действий*, имея в виду ситуационное исчисление в духе Джона МакКарти. В этой теории причины действий и сами действия описываются в виде клаузальных хорновских структур (импликация, левая часть которой есть конъюнкция атомарных формул, а правая состоит из одной атомарной формулы).

Далее, методы индуктивного логического программирования модифицируются таким образом, чтобы быть применимыми к поиску таких структур. Когда такие структуры найдены, их можно использовать в языках логического программирования для рассуждений о действиях и их причинах.

Многие работы этого направления посвящены «нейронной парадигме». Нейросетевой подход используется в огромном количестве задач — для кластеризации информации из Интернета, автоматической генерации локальных каталогов, представления образов (в рекурсивных нейронных сетях). Среди активно изучаемых в последнее время тем — неоднородные нейронные модели с отношениями сходства. (Heterogeneous Neural Networks with similarity relation).

Это отношение сходства определяется на множестве входов и множестве состояний сети, а мерой сходства является скалярное произведение векторов либо эвклидово расстояние (где один вектор — вектор входов, а другой является распределением весов нейронов, описывающих текущую ситуацию).

Работы по *автоматическому порождению гипотез* преследуют, главным образом, формализацию правдоподобных рассуждений, поиск

зависимостей причинно-следственного типа между некоторыми сущностями или, в широком смысле, создание познавательных процедур получения нового знания на основе анализа эмпирических данных. В качестве примеров можно привести: порождение гипотез о свойствах химических соединений (прогноз биологических активностей), о возможных причинах дефектов (диагностика) и т. п.

4. Интеллектуальный анализ данных и обработка образной информации. Это сравнительно новое направление, основу которого составляют две процедуры: обнаружение закономерностей в исходной информации и использование обнаруженных закономерностей для предсказания (прогнозирования). Сюда относят задачи выбора информативных данных из большой их совокупности, выбора информативных характеристик некоторого объекта из более широкого множества его характеристик, задачи построения модели, позволяющие вычислять значения выбранных информативных характеристик по значениям других характеристик, и т. п.

Значительную часть этого направления составляют исследования по различным аспектам распознавания изображений, в частности, с помощью нейросетей (включая *псевдооптические нейросети*). Изучаются методы распознавания последовательностей видеообразов на основе декларативного подхода и извлечения семантически значимой информации. К этому же направлению принадлежат исследования по графической технологии программирования в Интернете.

5. Многоагентные системы, динамические интеллектуальные системы и планирование. Это направление, изучающее интеллектуальные программные агенты и их коллективы.

Основные задачи в этой области таковы: реализация переговоров интеллектуальных агентов и разработка языков для этой цели, координация поведения агентов, разработка архитектуры языка программирования агентов.

Следует подчеркнуть, что со времени появления агентских технологий интерес к ним переместился из сферы академических исследований в сферу коммерческих и промышленных приложений, а идеи и методы агентских технологий весьма быстро мигрировали из искусственного интеллекта в практику разработки программного обеспечения и другие вычислительные дисциплины.

С агентскими технологиями тесно связаны задачи *планирования поведения*, или *ИИ-планирования*, — способность интеллектуальной системы синтезировать последовательность действий для достижения желаемого целевого состояния. Работы по созданию эффективных методов такого синтеза востребованы и активно ведутся уже около 30 лет.

Планирование является основой автоматического управления автономным целенаправленным поведением программно-технических систем.

Среди методов ИИ-планирования сегодня выделяют *классическое* планирование, т. е. планирование в условиях статической среды, *динамическое* планирование, т. е. планирование в условиях изменения среды и, главное, с учетом такого изменения, *иерархическое* планирование, т. е. когда действия абстрактного плана высокого уровня конкретизируются более детальными планами нижнего уровня, *частично-упорядоченное* (или *монотонное*) планирование, когда план строится на основе частично упорядоченного множества подпланов. При этом общий план (элементами которого являются подпланы), обязан быть монотонным, а каждый из подпланов может быть немонотонным. Добавлю, что монотонность — это такое свойство плана, когда каждое его действие уменьшает различия между текущим состоянием и целью поведения. Например, если план движения робота к цели таков, что каждый его шаг приближает к цели, то план монотонен, но если он наткнулся при этом на препятствие и требуется его обойти, то монотонность плана нарушится. Однако если план обхода препятствия выделить в отдельный подплан и рассматривать таковой как *элемент* исходного плана, то монотонность последнего восстановится.

Активно ведутся работы и в области распознавания планов, построения планировщиков и расширения их возможностей, эвристического планирования с ресурсными ограничениями, управления планированием посредством временной логики, планирования с использованием графов.

Рассматриваются подходы к планированию, при которых построение текущих планов выполняется непрерывно для каждого состояния системы в реальном времени. Для этого предусмотрен непрерывный мониторинг объекта управления.

Задачи планирования относятся в наше время к весьма важным и перспективным направлениям в искусственном интеллекте.

Интеллектуальные динамические системы — результат интеграции интеллектуальных систем с динамическими системами. В общем случае это двухуровневые динамические модели, где один из уровней отвечает за стратегию поведения системы (или, как иногда говорят, носит делиберативный характер), а другой уровень отвечает за решение частных задач, например, реализацию того или иного режима на основе применения математической модели.

Для описания временных зависимостей при построении интеллектуальных динамических систем исследуются соответствующие логики.

Работы в области *систем поддержки принятия решений* посвящены моделированию сложных технологических и технических систем,

поиску решений в условиях чрезвычайных ситуаций, задачам проектирования систем управления техническими объектами, использованию вероятностных подходов и сценариев при принятии решений, ряду других проблем.

6. Обработка естественного языка, пользовательский интерфейс и модели пользователя. Это направление связано с разработкой систем поддержки речевого общения, с решением проблем понимания естественного языка в интеллектуальных системах, с повышением точности поиска, с задачами сегментации текстов по тематическим топикам, с задачами управления диалогом, с задачами анализа естественного языка, с использованием различных эвристик. Сюда же включаются проблемы дискурса (иногда под дискурсом понимают совокупность речевых актов вместе с их результатами).

По-прежнему актуальны задачи обучения контекстному анализу текста, задачи приобретения знаний интеллектуальными системами и извлечения информации из текстов.

Важнейшей задачей в процессе извлечения информации, как, впрочем, и в процессе приобретения знаний, является минимизация роли эксперта — участника процесса.

Важность этого направления нельзя недооценивать. Причина тому — возрастание потоков текстовой информации, существующий социальный заказ на поиск релевантной информации в Интернете, на анализ текстовой информации, на извлечение данных из текстов.

Предметом исследований в этом направлении является также динамическое моделирование пользователя, в частности, в системах электронной коммерции, адаптивный интерфейс, мониторинг и анализ пользовательского поведения в Интернете.

7. Когнитивное моделирование. Когнитивное моделирование исследует сущность когнитивных функций в широком смысле этого слова посредством их эмпирического изучения и последующего моделирования.

В отличие от подходов, направленных на развитие методов решения трудных задач с априорно отсутствующими алгоритмами решения (таких, как представление знаний, моделирование рассуждений и поведения и ряда других), когнитивное моделирование является *процессно-ориентированным*.

Это означает, что в основе когнитивного моделирования лежат наблюдения над интеллектуальными и аффективными *процессами*, построение и экспериментальное исследование моделей этих процессов с последующими оценками релевантности как примененных моделей, так и методов моделирования.

Работы в области когнитивного моделирования включают исследование церебральных механизмов распространения и обработки информации в процессе реализации когнитивных функций, механизмов категоризации, именования (действий, объектов), механизмов целобразования и поведения, возникновения и развития когнитивных конфликтов.

8. Нечеткие модели и мягкие вычисления. Это направление представлено нечеткими схемами вывода, «вывода по аналогии», взглядом на теорию нечетких мер с вероятностных позиций, нечеткими представлениями, аналитическими моделями для описания геометрических объектов, алгоритмами эволюционного моделирования с динамическими параметрами, такими, как время жизни и размер популяции, методами решения оптимизационных задач с использованием технологий генетического поиска, гомеостатических и синергетических принципов и элементов самоорганизации.

9. Разработка инструментальных средств. Это обширная сфера деятельности, ставящая перед собой задачи:

- а) создания программных средств приобретения знаний для автоматизированного переноса знаний и компетентности в базы знаний. При этом в качестве источников могут выступать не только «прямые» их носители — эксперты различных областей, но и текстовые материалы — от учебников до протоколов, а также, разумеется, базы данных (имплицитные источники знаний). *Вербализация*, т. е. перевод таких источников в эксплицитную форму, составляет содержание методов обнаружения знаний в данных, в том числе различных методов обучения по примерам (включая предобработку больших массивов данных для дальнейшего анализа);
- б) реализации программных средств поддержки баз знаний;
- в) реализации программных средств автоматизации рассуждений;
- г) реализации программных средств поддержки проектирования интеллектуальных систем. Набор таких средств обычно содержит редактор текстов, редактор понятий, редактор концептуальных моделей, библиотеку моделей, систему приобретения знаний от экспертов, средства обучения по примерам и ряд других модулей.

Глава 1

МЕТОДЫ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Введение

В этой главе рассматриваются основные методы представления знаний в интеллектуальных системах: системы правил, семантические сети и системы фреймов. Достаточно подробно описаны системы правил, неоднородные семантические сети и отношения в них. В последнем параграфе главы описаны основы дескриптивной логики, возникшей относительно недавно и играющей роль логической основы некоторых методов представления знаний. При изложении основных способов представления знаний активно применяется язык исчисления предикатов первого порядка, атомарные формулы которого использованы для описания составных частей конструкций, используемых для представления знаний.

Поэтому прежде чем переходить собственно к методам представления знаний, уделим некоторое внимание исчислению предикатов первого порядка и одной из его интерпретаций.

1.1. Формальные языки и формальные системы

Существуют работы, где язык исчисления предикатов рассматривается как язык представления знаний, однако это не главное его назначение и мы будем использовать его, главным образом, в качестве средства описания элементов конструкций других языков, более ориентированных на представление знаний.

Опишем вначале основные конструкции языка исчисления предикатов первого порядка и их интерпретацию в духе работ [5, 6].

1.1.1. Язык исчисления предикатов первого порядка. Основные конструкции языка L — языка исчисления предикатов первого порядка — называются *формулами*. Введем вначале *алфавит* языка L . Алфавит включает:

1. Счетное множество букв x, y, \dots, z , которое будем называть множеством символов для обозначения переменных языка.

2. Счетное множество букв a, b, \dots, c , которое будем называть множеством символов для обозначения констант языка.
3. Счетное множество прописных букв P, Q, \dots для обозначения предикатных символов языка.
4. Счетное множество строчных букв f, g, \dots для обозначения функциональных символов.
5. Символы для логических связок \rightarrow (влечет), \neg (не).
6. Символ для квантора \forall (для любого).
7. $(,)$ — скобки.

Предикатные буквы P, Q, \dots и функциональные буквы f, g, \dots могут быть n -местными или, как еще говорят, n -арными. Иначе говоря, с каждым предикатным или функциональным символом будем связывать некоторое натуральное число, равное числу его аргументов.

Определим теперь понятие формулы или правильно построенного выражения языка исчисления предикатов первого порядка.

Формулы языка определяются индуктивным образом. Начнем с определения *терма* языка:

1. Символ для обозначения переменной есть терм.
2. Символ для обозначения константы есть терм.
3. Если $t_1, t_2, \dots, t_m, \dots, t_n$ — термы, а f и g — функциональные символы арности m и n соответственно, то $f(t_1, t_2, \dots, t_m)$ и $g(t_1, t_2, \dots, t_n)$ также термы.
4. Если $t_1, t_2, \dots, t_m, \dots, t_n$ — термы, а P и Q — предикатные символы арности m и n соответственно, то $P(t_1, t_2, \dots, t_m)$ и $Q(t_1, t_2, \dots, t_n)$ — атомарные формулы.
5. Атомарная формула есть формула.
6. Если A, B — формулы, то $A \rightarrow B, \neg A, \neg B$ — формулы.
7. Если A — формула, то $\forall x A$ — формула.
8. Всякое слово в алфавите языка является формулой тогда и только тогда, когда это можно показать с помощью конечного числа применений пп. 1–7.

Таким образом, мы завершили одно из возможных определений языка исчисления предикатов первого порядка. Существуют и другие определения, однако язык, определенный нами, является полным, т. е. в нем выразимо все то, что выразимо в языках (исчисления предикатов первого порядка), определенных любым иным способом.

Можно, например, определить логические связки \wedge, \vee (читается *и* и *или*), выразив их через связки \rightarrow и \neg :

1. $A \wedge B = \neg(A \rightarrow \neg B)$,
2. $A \vee B = \neg A \rightarrow B$.

Квантор существования \exists (существует) также выражается через квантор всеобщности и отрицание:

$$\exists xA(x) = \neg\forall x\neg A(x).$$

Разумеется, \wedge , \vee и \exists с тем же успехом можно было бы включить в язык в качестве трех дополнительных символов. Есть, однако, некоторые преимущества в том, чтобы сохранить список символов как можно более коротким. Например, индуктивные определения и доказательства по индукции оказываются в этом случае короче.

В дальнейшем нам придется использовать понятия *свободного* и *связанного* вхождения переменной в формулу. Вхождение переменной x в формулу A называется *связанным*, если эта переменная следует за кванторами существования или общности, предшествующими формуле A и находится в области их действия. В противном случае вхождение переменной называется *свободным*. Если в формуле A отсутствуют свободно входящие в нее переменные (т. е. либо все переменные связаны, либо отсутствуют), то формула называется *замкнутой формулой* или *предложением*. Атомарную замкнутую формулу будем называть *фактом*. В том случае, если язык состоит только лишь из предложений, то он называется пропозициональным языком, а буквы A, B, \dots , входящие в формулы этого языка — пропозициональными переменными.

1.1.2. Элементы исчисления предикатов первого порядка. Рассмотрим вкратце основные понятия исчисления предикатов первого порядка.

Введем вначале *аксиомы* исчисления предикатов:

1. $A \rightarrow (B \rightarrow A)$.
2. $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$.
3. $(\neg A \rightarrow \neg B) \rightarrow (B \rightarrow A)$.

Однако исчисление предикатов первого порядка не исчерпывается приведенными выше тремя аксиомами и правилами вывода. Смысл кванторов устанавливается еще двумя аксиомами и одним правилом вывода.

4. **Аксиома генерализации:** $(\forall x)((A \rightarrow B) \rightarrow (A \rightarrow (\forall x)B))$, где x не является свободной переменной в A ;
5. **Аксиома спецификации:** $\forall tA(t) \rightarrow A(x)$, где t — терм, а x не содержится в t в качестве свободной переменной.

Правила вывода:

1. **Правило отделения:** если выводимо A и выводимо $A \rightarrow B$, то выводимо B .

2. **Правило подстановки:** в любую аксиому на место любой пропозициональной переменной можно подставить любое предложение, предварительно переименовав пропозициональные переменные подставляемого предложения так, чтобы они не совпадали с пропозициональными переменными аксиомы.

Если в аксиомах 1–3 все переменные являются пропозициональными, то такое исчисление называется *пропозициональным исчислением* или *исчислением высказываний*.

Рассмотрим пример вывода в исчислении высказываний. Возьмем, например, три закона логики, сформулированные Аристотелем и называемые постулатами Аристотеля. В языке исчисления высказываний они записываются следующим образом.

Пусть P — пропозициональная переменная исчисления высказываний:

Постулат 1. $P \rightarrow P$.

Постулат 2. $P \vee \neg P$.

Постулат 3. $\neg(P \wedge \neg P)$.

Первый из постулатов Аристотеля — это так называемый закон тождества; второй — закон исключенного третьего и третий — закон противоречия.

Докажем один из постулатов, например $P \rightarrow P$.

Используем аксиому 1 и правило подстановки (вместо B подставим $P \rightarrow P$: получим $A \rightarrow ((P \rightarrow P) \rightarrow A)$).

Из аксиомы 2: $(A \rightarrow ((P \rightarrow P) \rightarrow C)) \rightarrow ((A \rightarrow (P \rightarrow P)) \rightarrow (A \rightarrow C))$.

Вместо A, C подставим P :

$$\underbrace{(P \rightarrow ((P \rightarrow P) \rightarrow P))}_X \rightarrow \underbrace{((P \rightarrow (P \rightarrow P)) \rightarrow (P \rightarrow P))}_Y.$$

Применим правило отделения: та часть последней формулы, которая обозначена через X , является аксиомой, т. е. выводима, тогда, в силу правила отделения, выводима формула, обозначенная через Y .

Теперь применим правило отделения к Y :

$$\underbrace{(P \rightarrow (P \rightarrow P))}_{X'} \rightarrow \underbrace{(P \rightarrow P)}_{Y'}$$

и, рассуждая таким же образом, получим, что Y' выводима. Таким образом, закон тождества Аристотеля является *теоремой* исчисления высказываний. Действуя таким же образом, можно доказать, что второй и третий постулаты Аристотеля также являются *теоремами* исчисления высказываний.

3. **Правило обобщения:** если выводимо A , то выводимо $\forall xA$, где x — свободная переменная в A .

Аксиомы 1–5 исчисления предикатов первого порядка (или математической логики первого порядка) называются *логическими* аксиомами, они описывают логические законы, справедливые всегда, независимо от предметной области. Если же к аксиомам 1–5 добавить еще и аксиомы, описывающие некоторую предметную область, например, арифметику или теорию групп, то получим *формальную* теорию — формальную арифметику или формальную теорию групп, соответственно. При этом, разумеется, в алфавит языка следует ввести специальные функциональные символы, такие как сложение в арифметике или умножение в теории групп.

Словосочетание «первый порядок» относится исключительно к тому обстоятельству, что кванторы \forall и \exists действуют на некотором множестве U . Логика второго порядка разрешает одному из кванторов действовать на подмножествах множества U и на функциях из степеней U в U . Логика третьего порядка может использовать кванторы по множествам функций и т. д. Уже из этих разъяснений видно, что в логиках более высоких порядков (как говорят, более сильных логиках) используются и некоторые нелогические понятия, такие как множество.

Некоторым обобщением понятия исчисления является понятие формальной системы.

1.1.3. Формальные системы. Будем полагать, что если заданы некоторый алфавит, множество формул, множества аксиом и правил вывода, то тем самым задана некоторая *формальная система*. Иначе говоря, формальная система F представляет собой совокупность следующих объектов:

$$F = \langle T, P, A, \Pi \rangle,$$

T — счетное множество символов и операций над ними; P — множество правил грамматики, применение которых к символам из T позволяет строить правильно построенные формулы; A — множество аксиом; Π — множество правил вывода.

Если среди аксиом имеются нелогические аксиомы (аксиомы, описывающие некоторую предметную область), то формальная система называется *формальной теорией*.

Выводом (или доказательством) в формальной системе называется конечная последовательность правильно построенных формул A_1, A_2, \dots, A_n , таких что каждая из формул последовательности либо является аксиомой, либо получена из предыдущих формул последовательности с использованием аксиом и правил вывода. Формула A_n

в этом случае называется выводимой формулой (или теоремой) формальной системы F .

1.1.4. Алгебраические системы. Введем теперь некоторые алгебраические понятия, которые окажутся полезными в дальнейшем.

Определение 1.1.4. n -арным отношением R на M называется подмножество M^n ; записывается так: $R \subseteq M^n$.

Пусть задано непустое множество — универсум M .

Упорядоченную тройку $S = \langle M, G, R \rangle$, где универсум M называется основным множеством, G — множество n -местных функций из M^n в M , а R — семейство n -арных отношений на M , будем называть алгебраической системой [7].

Если в определении алгебраической системы S положить $R = \emptyset$, то такая алгебраическая система называется алгеброй, а если $G = \emptyset$, то тогда алгебраическая система называется моделью.

Иногда удобнее использовать понятие *многосортовой* или *многоосновной* алгебраической системы, полагая существование нескольких основных множеств M_1, M_2, \dots, M_n вместо одного множества M . Тогда функции из G и отношения из R становятся многосортными, т.е. с каждым местом в списке аргументов функции или отношения связывается некоторый сорт i , задаваемый, например, индексом одного из множеств из M_i ($i = 1, 2, \dots, n$), а сами функции и отношения определяются на декартовых произведениях множеств из числа основных множеств.

1.1.5. Интерпретация. Пусть заданы формальная система F с языком L и универсум M . Зададим отображение I , которое всякому константному символу a языка L системы F ставит в соответствие некоторый элемент $m \in M$, всякому n -местному предикатному символу P — n -местное отношение $R \subseteq M^n$, а всякому t -местному функциональному символу f — t -местную функцию $G: M^n \rightarrow M$. Тогда I называется *интерпретирующим отображением*, а алгебраическая система $S = \langle M, G, R \rangle$ — *моделью* или *интерпретацией формальной системы* F . Элементы множества M , функции и отношения, соответствующие константным, функциональным и предикатным символам языка L в смысле отображения I , называют иногда *интерпретациями* константных, функциональных и предикатных символов языка L соответственно.

При определенных обстоятельствах можно говорить и о моделях языка. Моделью языка L называется пара $\langle M, I \rangle$, где M и I — уже использовавшиеся только что универсум и интерпретирующее отображение.

Заметим, что при данном универсуме M для символов языка L существует много различных интерпретаций.