

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

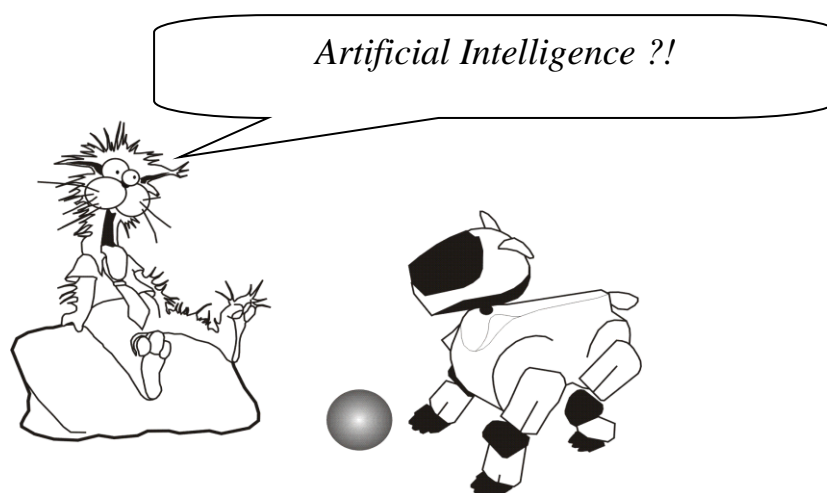


Институт компьютерных технологий и информационной безопасности
Кафедра математического обеспечения и применения ЭВМ

С.И. РОДЗИН

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

Учебное пособие



ТАГАНРОГ 2018-19

Рецензенты:

*Профессор Ростовского государственного университета путей сообщения,
доктор технических наук, профессор Ковалев С.М.*

*Зав. кафедрой САПР ИКТИБ ЮФУ,
доктор технических наук, профессор Курейчик В.В.*

Родзин С.И. ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: Учебное пособие. – Таганрог ИКТИБ ЮФУ, 2018. – 148 с.

ISBN 978-5-8327-0335-0

В пособии нашли отражение модели представления знаний и механизмы вывода на знаниях. В последнее время разработка систем, основанных на знаниях, стала экономически выгодной, хотя довольно трудной, а потому интересной. Тенденции в области применения компьютеров характеризуются возрастанием значения методов искусственного интеллекта в программном обеспечении. Происходит непрерывный процесс «интеллектуализации компьютеров». Сейчас корень *compute* в слове компьютер является анахронизмом. Современный компьютер – это информационная машина, которая способна обрабатывать знания. Системы искусственного интеллекта с пользой применяются в реальной жизни, от медицинской диагностики до управления космическими аппаратами. К этому можно относиться как угодно, но это невозможно отрицать. Именно поэтому курс «Системы искусственного интеллекта» присутствует в учебных планах подготовки бакалавров и аспирантов по информатике.

Кроме лекций, пособие включает вопросы для собеседования, контрольные вопросы, список литературы, глоссарий и персоналии. Пособие основано на образовательном контенте, используемом автором в образовательных программах Института компьютерных технологий и информационной безопасности Южного федерального университета.

Для студентов направлений «Математическое обеспечение и администрирование информационных систем», «Программная инженерия», «Информатика и вычислительная техника», изучающих дисциплину «Системы искусственного интеллекта». Пособие может быть полезно магистрантам и аспирантам направлений «Программная инженерия», «Информатика и вычислительная техника», занимающимся проблематикой искусственного интеллекта, теоретической информатики, математического моделирования, интеллектуализации САПР.

ISBN 978-5-8327-0335-0

© С.И. Родзин, 2018

© ИКТИБ ЮФУ, 2018

Содержание

Информация об авторе курса	4
Предисловие	6
Цель пособия.....	6
Принятые обозначения	6
Как пользоваться пособием.....	7
Введение	9
Общие сведения о дисциплине	9
Методические указания к самостоятельной работе.....	10
Лекция « Общие вопросы искусственного интеллекта»	12
Лекция «Постулаты искусственного интеллекта. Нейроинформатика»	27
Лекция «Интеллектуальные задачи и проблем. Данные-информация-знания»	38
Лекция «Свойства знаний. Понятия и образы»	50
Лекция «Классификация понятий. Модели знаний»	57
Лекция «Достоверный логический вывод. Метод резолюций»	70
Лекция «Эвристический поиск»	82
Лекция «Правдоподобный вывод»	92
Заключение	104
Вопросы для собеседования	116
Контрольные вопросы	117
Учебно-методическое обеспечение курса	126
Глоссарий	129
Персоналии	144

Информация об авторе курса



Родзин Сергей Иванович, профессор кафедры МОП ЭВМ ИКТИБ ЮФУ, член Российской ассоциации искусственного интеллекта, Европейского союза исследователей в области искусственного интеллекта (ECCAI), международной ассоциации «Fault-Tolerant Computing Systems». Награждён знаком отличия «Почётный работник высшего профессионального образования Российской Федерации», юбилейной медалью «За заслуги перед Южным федеральным

университетом».

Автор свыше 270 трудов, среди которых 20 монографий, 25 учебных пособий, свыше 150 статей в отечественных и зарубежных научных журналах. Некоторые публикации:

1. Родзин С.И. Умное диспетчирование и роевой потоковый алгоритм // Известия РАН. Теория и системы управления. 2014. №1. С.106-112. <http://elibrary.ru/item.asp?id=20991767>

2. Родзин С.И. Организация параллельных эволюционных вычислений // Вестник компьютерных и информационных технологий. 2010. №9. С.7-13. <http://elibrary.ru/item.asp?id=15405203>

3. Родзин С.И. Метрика и алгоритмы меметики // Вестник РГУПС. 2013. №4. С.59-67. <http://elibrary.ru/download/72028481.pdf>

Родзин С.И., Титаренко И.Н. Конвергенция нано-, био-, инфо-, когнитив-ных технологий и электронная культура // Открытое образование. 2014. №3. С. 10-17. <http://elibrary.ru/item.asp?id=21610423>

4. Rodzin S.I. Smart Dispatching and Metaheuristic Swarm Flow Algorithm // Jour. Comp.&Systems Sciences International. 2014. vol.53. No.1. P.109–115. <http://www.maik.rssi.ru/cgi-perl/search.pl?type=abstract&name=compsys&number=1&year=2&page=123>

5. Rodzin S., Rodzina L. Theory of Bioinspired Search for Optimal Solutions and its Application for the Processing of Problem-Oriented Knowledge // Proc. of the 8th IEEE Int. Conf. Application of Information and Communication Technologies (AICT'2014), Astana, Kazakhstan, 2014. P.142-146. <http://aict.info/2014/?page=callfor>

6. Родзин С.И. Биомеметика - интеграция методов машинного обучения и эволюционных вычислений // Известия Южного федерального университета. Технические науки. 2014. №7. С.26-34. <http://elibrary.ru/item.asp?id=21782578>

7. Родзин С.И. Обобщенная модель базового цикла эволюционных вычислений // Известия Южного федерального университета. Технические науки. 2012. №7. С.49-55. <http://izv-tn.tti.sfedu.ru/?p=4600>

8. Родзин С.И., Родзина О.Н. Метод биогеографии для решения трансвычислительных задач комбинаторной оптимизации // Информационные технологии в науке, образовании и управлении: труды межд. конф. IT + S&E'15. М.: ИНИТ, 2015. С. 204-213. <http://elibrary.ru/item.asp?id=23518153>

9. Rodzin S., Rodzina L. Mobile Learning Systems and Ontology // Proc. of the 4th Computer Science On-line Conf. (CSOC'15), 2015, Vol 3: Software Engineering in Intelligent Systems. P.45-54. http://link.springer.com/chapter/10.1007%2F978-3-319-18473-9_5#close

10. Курейчик В.В., Курейчик В.М., Родзин С.И. Теория эволюционных вычислений. Монография. – М.: Физматлит, 2012. 260 с. http://www.rfbr.ru/rffi/ru/books/o_1780977

Научно-педагогический стаж – 35 лет.

Руководит грантом Российского фонда фундаментальных исследований, участвует в выполнении гранта Российского научного фонда, подготовкой аспирантов. Эксперт ведущих научных фондов.


Читает курсы лекций: «Теория принятия решений», «Искусственный интеллект», «Принятие инженерных решений», «Системы искусственного интеллекта».

Контактная информация: °e-mail - srodzin@sfedu.ru.


Предисловие

Цель пособия

Пособие предназначено для студентов и аспирантов, изучающих дисциплину «Системы искусственного интеллекта» (СИИ).

 ВНИМАНИЕ!	<p>Цель пособия – дать представление о фундаментальных фактах и принципах искусственного интеллекта (ИИ), познакомить с задачами ИИ, современными моделями представления и методами достоверного вывода знаний, компонентами интеллектуальных систем, помочь овладеть способами формализации интеллектуальных задач, необходимыми для разработки математического и программного обеспечения СИИ.</p>
---	--

Мы ожидаем, что слушатели данного курса, достигнут следующих результатов:






 ПРИМЕР	<ol style="list-style-type: none"> 1. Знание основных принципов наиболее известных способов представления знаний: правил продукции, семантических сетей, фреймов формул исчисления высказываний и предикатов. 2. Умение выбрать подходящий способ представления знаний для конкретной задачи в простых случаях; умение оценить адекватность использования для конкретной задачи того или иного метода поиска решения. 3. Свободное владение основными понятиями из области искусственного интеллекта и представления знаний; умение изучить по литературе другие методы искусственного интеллекта; умение оценить конкретную прикладную систему искусственного интеллекта и воспользоваться ею.
--	--

Принятые обозначения

В пособии используются следующие шрифты:

Шрифт	Описание
Жирный шрифт	Выделяет в основном тексте важную информацию.
<i>Курсив</i>	Используется для выделения отдельных значимых слов, фраз, пояснений, а также для указания информационных источников, в которых дано более детальное описание текущих вопросов.

В тексте пособия используются следующие смысловые акценты и пиктограммы:

Пиктограмма	Описание
 ВНИМАНИЕ	Принципиально важный момент. Не следует обходить его своим вниманием.
 СОВЕТ	Рекомендации по повышению эффективности работы.
 СПРАВКА	Информация к сведению.
 ПРИМЕР	Поясняющий пример.
 ВОПРОС	Проблемный вопрос.

Как пользоваться пособием

Основная задача пособия - помощь при самостоятельной работе слушателей. Даже самый полный учебник не в состоянии вместить в себя весь объем информации, которая может понадобиться при освоении дисциплины, всегда требуется дополнительная литература. С появлением Интернета и бурным развитием тематических сайтов и порталов различного назначения стало возможным найти практически любую информацию, подключившись к сети и сделав несколько запросов к поисковым машинам. Но с подобной системой поиска информации пока имеются определенные сложности. В данном случае преимуществом пособия является то, что значительная часть необходимого для освоения дисциплины материала собрана в одном месте и слушателям не приходится тратить время на поиск этого материала по различным источникам. Кроме того, слушатель может провести самопроверку усвоенного материала, используя тестовые задания для проверки знаний. Таким образом, пособие может использоваться как в контексте лекций, так и в качестве материалов для самостоятельной работы слушателей.

Несмотря на некоторые преимущества, которые вносит в учебный процесс использование пособия, следует учитывать, что оно является вспомогательным инструментом, дополняет, а не полностью заменяет другие виды образовательного контента. Варианты использования данного пособия могут быть различными, они

зависят от целей слушателя, а также опыта работы с образовательными ресурсами. Если Вы только приступили к изучению пособия, то рекомендуем изучать представленный материал последовательно. Если Вы продолжаете осваивать курс, то, при необходимости, просмотреть глоссарий и внимательно изучить соответствующий модуль или тему лекции.

При работе с пособием необходимо учитывать рабочую программу изучаемой дисциплины. Поэтому календарно-тематический план и графики освоения дисциплины являются различными для различных слушателей и приводятся в рабочих программах этих дисциплин.

Для оценки степени достижения и реализации целей и задач дисциплины СИИ используются методы, основанные на систематическом контроле и мониторинге знаний, которыми должны обладать слушатели по окончании изучения дисциплины.

Допущение о том, что слушатель аттестуется по дисциплине только потому, что он просто выполнил в течение семестра всю учебную нагрузку по программе, не является доказательством достижения данным студентом задач дисциплины.

Система контроля и мониторинга представлены в рабочей программе дисциплины и фонде оценочных средств, вместе с критериями, определяющими и численно устанавливающими уровень и качество самостоятельной работы слушателей по дисциплине.

Система оценки достигаемых каждым слушателем результатов по дисциплине является накопительной, в итоге аттестации определяется число баллов как количественная оценка степени достижимости задач дисциплины.

Введение


Общие сведения о дисциплине

Дисциплина «Системы искусственного интеллекта» входит в учебные планы подготовки бакалавров направлений «Математическое обеспечение и администрирование информационных систем», «Программная инженерия», магистров и аспирантов направления «Информатика и вычислительная техника». Поэтому дисциплина является важной составляющей профессиональной подготовки слушателей по указанным направлениям.


Программа СИИ соответствует требованиям образовательных стандартов, а также рекомендации авторитетных международных и российских организаций по преподаванию дисциплин в области искусственного интеллекта для специальностей, связанных с информационно-компьютерными технологиями.

Трудоемкость изучения дисциплины СИИ в академических часах и зачетных единицах определяется согласно учебным планам.

СИИ является междисциплинарным курсом, потому что ИИ – это междисциплинарное направление в информатике и современном естествознании. Воплощением СИИ являются компьютерные системы, способные решать интеллектуальные задачи или действовать, используя принципы и механизмы живой природы.


 СПРАВКА	<p>Изучение СИИ опирается на междисциплинарные связи и использование знаний, умений и навыков, полученных при освоении таких предметов, как «Философия», «Алгебра», «Математическая логика и теория алгоритмов», «Дискретная математика», «Теория вероятностей», «Структуры и алгоритмы обработки данных», «Базы данных», «Функционально-логическое программирование», «Объектно-ориентированное программирование», «Технология разработки программного обеспечения».</p>
--	---

Вклад дисциплины в достижение ожидаемых результатов в профессиональной части указанных образовательных программ заключается в том, что в результате изучения дисциплины студент должен быть готов продемонстрировать:


 ВНИМАНИЕ!	<ul style="list-style-type: none"> • способность применять знания математики и компьютерной техники, а также умение идентифицировать и формулировать инженерные задачи; • способность применять методы, связанные с моделированием инженерных задач, а также анализировать и объяснять полученные результаты; • знание теоретических основ искусственного интеллекта для
--	---

	освоения современных интеллектуальных пакетов прикладных программ и решения задач по администрированию и управлению знаниями.
--	---

СИИ представляют собой экспериментальную науку, базирующуюся на фундаменте компьютерных наук и являющуюся вершиной развития информационных технологий. Создавая модели человеческих рассуждений, моделируя структуру мозга человека, отбирая лучшие из моделей, а также сравнивая поведение интеллектуальных систем путём их имитации, исследователь модифицирует их, пытаясь добиться лучших результатов. В этом смысле дисциплина, изучающая процессы извлечения, представления и обработки знаний, принципы рассуждений, не имеет пределов, что также способствует широкому удовлетворению потребностей личности в изучении инженерных наук об искусственных системах.

 СПРАВКА	На рынке возрастает спрос на кадры, умеющие работать и управлять знаниями в автоматизированных системах, владеющие эффективными методами обработки, формализации и структурирования знаний. Это задача для инженеров-специалистов по знаниям.
--	--

Основанием к формированию содержания рабочей программы дисциплины ИС является приближение инженерной подготовки к ведущим мировым тенденциям и потребностям на рынке труда в области интеллектуальных информационно-компьютерных технологий:

 СПРАВКА	<ul style="list-style-type: none"> • открытие технопарков; • стремительное развитие рынка оффшорного и заказного программирования; • развитие российских национальных проектов и информатизация госструктур; • потребности частного бизнеса.
--	--

Методы искусственного интеллекта сейчас очень быстро развиваются и меняются, и невозможно дать заранее готовые рецепты на все случаи применения ИИ. Вместо этого в пособии дается общее представление об основных принципах искусственного интеллекта, определяются наиболее распространенные методы представления знаний в компьютере.

Методические указания к самостоятельной работе

Основой для самостоятельной работы является ее образовательный контент. Наряду с рабочей программой и данным учебным пособием, образовательный контент также включает основную и дополнительную литературу, методические

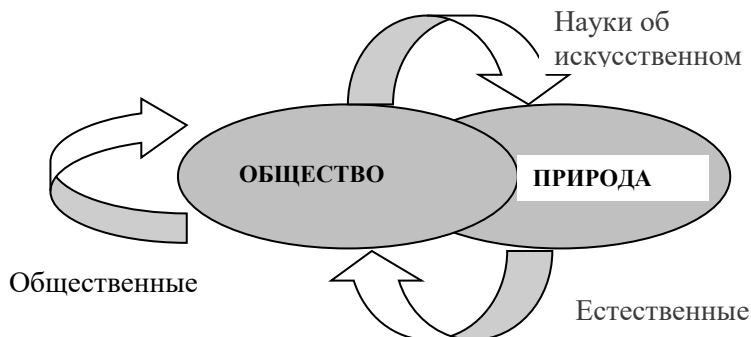
разработки к выполнению лабораторных работ, практических заданий, вопросы для собеседования и контрольные вопросы.

Самостоятельная работа с пособием предусматривает критический подход к идеям в области ИИ, способам их доказательства и экспериментальным данным, выработку умения видеть проблемы как теоретического, так и практического характера, выявлять взаимосвязь между учебными модулями.

Пособие охватывает примерно половину лекционной части курса. На его основе можно осуществить эффективный процесс обучения, отличающийся минимально необходимым временем аудиторных занятий при максимально возможной мотивации самостоятельной работы.

Лекция « Общие вопросы искусственного интеллекта »

Классификация наук (Г.Саймон)



Отличия наук об искусственном от естественных наук состоят в следующем:

- 1). В естественных науках вопрос «что изучать?» задает природа, в науках об искусственном – исследователь;
- 2). Главный инструмент познания в естественных науках – анализ, в искусственных – синтез и компьютерное моделирование;
- 3). Модели естественных объектов являются дескриптивными (описательными), в отличие от нормативных (что должен обеспечивать объект) моделей наук об искусственном;
- 4). Смена парадигм, переход к технологиям в науках об искусственном происходит гораздо быстрее.

Искусственный интеллект (ИИ) занимает центральное место среди наук об искусственном. К ним также относятся такие науки как виртуалистика (наука об искусственных средах), когнитивная наука, объединяющая теорию познания, когнитивную психологию, нейрофизиологию и когнитивную лингвистику. Саймон подчеркивал связь наук об искусственном с инженерными науками:

- искусственные объекты в инженерии конструируются людьми. В науках об искусственном - не только людьми, например, с помощью ЭВМ;
- объекты в инженерии и в науках об искусственном можно характеризовать их функциями, целями и степенью приспособления к требованиям среды;
- при проектировании объектов в инженерии и в науках об искусственном используются категории, что должен обеспечивать конструируемый объект, т.е. применяются в основном нормативные модели;
- искусственные объекты могут внешне походить на естественные, но заметно отличаться от последних в одном или нескольких аспектах.

Из истории ИИ

Мифы (античность):

- Афродита услышала мольбы кипрского царя Пигмалиона и оживила **Галатею** из статуи.
- Талос - мифический великан, первый робот в истории.
- Пифия передает посетителям слова Дельфийского оракула - первой в истории экспертной системы.

Аристотель (384-322 до н.э.). Сформулировал законы логического мышления, разработал систему силлогизмов и алгоритм, предназначенный для проведения правильных рассуждений. Алгоритм Аристотеля был реализован через 2300 лет Ньюэллом и Саймоном в программе *GPS* (General Problem Solver – универсальный решатель задач).

Мифы (Ренессанс):

- Логическая машина Раймунда Луллия (в виде подвижных концентрических кругов, построенных по троичной логике. Круги он разделил поперечными линиями на сектора, которые обозначали общие понятия. Вращая круги можно получать множество новых комбинаций, в которых Луллий видел новые реальные истины. Умер в 1315 г);
- Легенда о Големе, глиняном истукане, в котором при соответствующих условиях зарождается искра разума;
- Легенда о Франкенштейне.

Автоматизация вычислений началась примерно в 1500 году. Л.да Винчи (1452—1519) спроектировал, но не построил механический калькулятор. Недавняя реконструкция показала, что его проект был работоспособным. Первая известная вычислительная машина была создана примерно в 1623 году немецким ученым В.Шиккардом. Более известна машина, построенная в 1642 году Б.Паскалем.

Г.Лейбниц (1646-1716) создал механическое устройство, предназначенное для выполнения операций над понятиями, а не над числами.

Р.Декарт (1596-1650) впервые опубликовал результаты обсуждения различий между разумом и материей, а также возникающих при этом проблем. Интеллект, по Декарту, не подчиняется физическим законам.

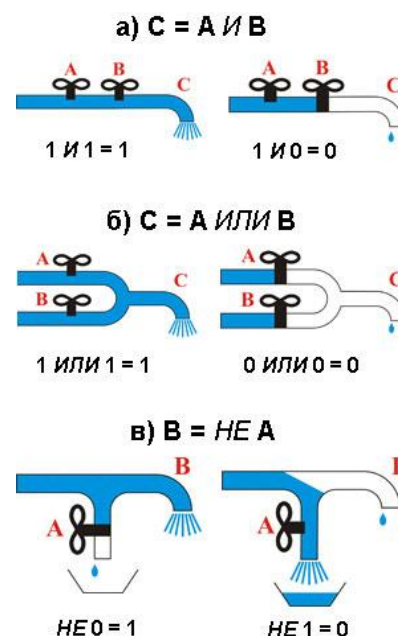
Когда появилась идея ИИ? В XVII в. Б. Спиноза, Р. Декарт, Г.В. Лейбниц и др. говорили именно об ИИ, а не о механических куклах. Основоположники ИИ были оптимистами, верили в реализуемость идеи.

Пессимистами в этом вопросе были Ф. Бэкон, Дж. Локк и др. Известна полемика между Локком и Лейбницем (примерно 300 лет назад):

- по Локку разум – это «чистая доска», на которой ощущения от органов чувств, отпечатываются в виде “идей”. Локк игнорировал язык как средство программирования деятельности разума, по нему идеи у каждого разума свои, отсюда трудности в понимании. Язык по Локку не несет в себе “структуры знания”, это лишь набор знаков. Эти представления сейчас лежат в основе работ по распознаванию образов;

- Лейбниц рассматривал разум как «базу знаний». Наш язык по Лейбницу отражает структуру знаний о мире. Это предвосхищает современные идеи создания языков программирования.

Большинство мыслителей VIII-IX веков были уверены, что формальная логика может объяснить наше мышление. Например, законы булевой логики с простыми операциями «И», «ИЛИ» и «НЕТ» до сих пор используются многими «продвинутыми» системами контекстного поиска в Интернете.



Может ли машина быть умнее своего создателя?

Для этого:

- 1). Машина должна быть свободной, т.е. обладать правами на существование, на распоряжение своим временем, памятью, информацией. Свободная машина способна сама себя включать и выключать.
- 2). Машина должна быть генетически связанной (по памяти) создателем или предшествующими машинами.
- 3). Машина должна иметь программу понимания и программу самообучения.

Может ли знание храниться вне мозга?

Оптимисты утверждают, что знание может храниться вне мозга. Их доводы таковы: познание как процесс поддается формализации; интеллект можно измерить (IQ, объем памяти, и др.); к знанию применимы информационные меры (бит, байт и др.).

Пессимисты считают, что ИИ не способен хранить знание, т.к. он – всего лишь имитация мышления. Их доводы таковы: человеческий интеллект уникален, творчество не поддается формализации, мир цел и неделим на информационные дискреты, образность мышления человека богаче машинной логики.

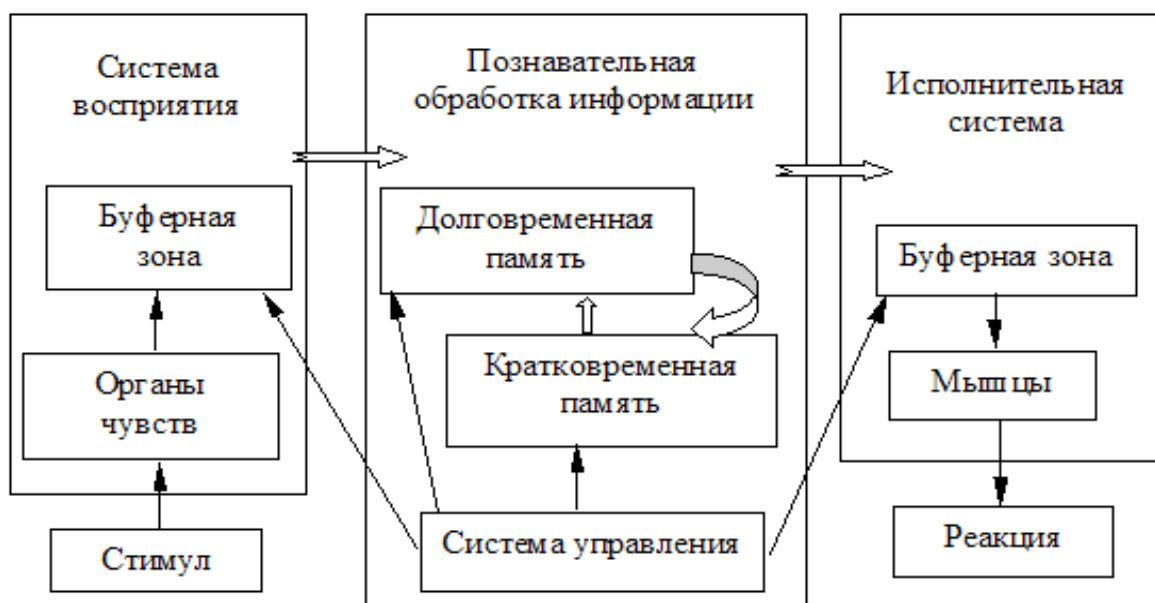
Кто прав в этом споре, покажет время. Рассмотрим, каковы особенности процесса мышления и обработки информации у человека.

Процесс мышления чрезвычайно сложен. Одна ячейка глаза способна выполнять за 10 мс обработку, эквивалентную решению системы из 500 нелинейных дифуравнений.

Глаз человека насчитывает не менее 10 миллионов ячеек, компьютеру Cray-1 необходимо затратить около 100 лет, чтобы воспроизвести процессы, происходящие каждую секунду в нас в глазу.

Отношения между сознанием и мозгом напоминают модель отношений между software и hardware.

Система обработки информации у человека



- Данные из внешнего мира воспринимаются с помощью органов чувств и помещаются в буфер кратковременной памяти для анализа.

- В долговременной памяти хранятся символные образы и смысловые связи между ними, которые используются для объяснения новой информации, поступающей из кратковременной памяти.

- Большие объемы данных постоянно записываются в кратковременную память, и мы непрерывно анализируем и фильтруем получаемую информацию для того, чтобы определить степень ее важности и то, как она соотносится с образами, хранящимися в долговременной памяти.

Доступ к информации

- Доступ к информации в долговременной памяти осуществляется достаточно эффективно. Любой элемент данных может быть извлечен в течение не более 70 мс и затем преобразован

- Перемещение данных из кратковременной памяти в долговременную память занимает ~ 15–20 минут.

- Если человек в получил мозговую травму, то долговременная память может восстановиться. Однако вся информация за 15–20 минут до катастрофы, будет потеряна и никогда не восстановится.

Способ хранения символьных образов

- Человек хранит не числовые данные, а образы или символы.
- Способ хранения образов в долговременной памяти схож с хранением числовой информации в БД сетевого типа.
- В памяти существует система указателей, позволяющая нам быстро извлечь нужный символ и все данные, которые с ним связаны.

Чанки

- Символьные образы в мозге человека объединены в **чанки – наборы фактов и связей между ними, запомненные и извлекаемые как единое целое.**

- В каждый момент времени человек может обрабатывать и интерпретировать не более 4-7 чанков.

Чтобы убедиться в этом, прочтите следующую строку и попытайтесь написать на листе бумаги то, что вы прочли:

НЛОИМЙКРТС ЗРЗЗЙНК ВНБНМ ГВССМЗЛ

Теперь проверьте написанный Вами текст и подсчитайте, сколько символов вам удалось воспроизвести правильно. Повторите эксперимент со следующим предложением:

ЭКСПЕРТНЫЕ СИСТЕМЫ ПОЧТИ РАЗУМНЫ

Оба предложения состоят из одинакового количества букв и слов. Почему же так отличаются результаты?

Объем кратковременной памяти составляет от 5 - 7 чанков. Если люди не повторяют (мысленно или вслух) поступившую в кратковременной память информацию, то она быстро забывается. Забывание происходит оттого, что новые чанки вытесняют старые, либо информация угасает со временем.

Эксперты обладают способностью объединять в чанки большие объемы данных и устанавливать связи между ними, чтобы быстро распознавать новые ситуации по мере поступления информации о них.

Средний специалист в конкретной предметной области помнит от 50 000 до 100 000 чанков.

Самая влиятельная личность в ИИ

Скромная мемориальная доска, относительно недавно установленная на стене одной из лондонских гостиниц, гласит: «*Здесь родился Алан Тьюринг (1912—1954), взломщик кодов [Code-breaker] и пионер информатики [computer science]*».

Был безразличен к борьбе за приоритет в научных открытиях. Только сейчас (но отнюдь не при жизни!) Тьюринг признан одним из основателей информатики и теории ИИ, его считают первым теоретиком современного программирования и, наконец, первым в мире «хакером» (во время второй мировой войны внес существенный вклад в победу союзных войск над германским флотом, расшифровав «Энигму»).

Его роль в ИИ не исчерпывается изобретением «машины Тьюринга», теста Тьюринга. Он доказал, что в математике существуют истинные выражения, которые невозможно вычислить компьютером, каким бы мощным он ни был. Тьюринг вполне может быть причислен к плеяде величайших математических и философских умов, составляющих гордость человечества.

Тест Тьюринга - эмпирический тест, идея которого была предложена А.Тьюрингом в статье «Вычислительные машины и разум» (*Computing Machinery and Intelligence*), опубликованной в 1950 году в журнале «*Mind*». Тьюринг задался целью определить, может ли машина мыслить.

Стандартный тест Тьюринга: «Человек взаимодействует с одним компьютером и одним человеком. На основании ответов на вопросы он должен определить, с кем он разговаривает: с человеком или компьютерной программой. Задача компьютерной программы - ввести человека в заблуждение, заставив сделать неверный выбор».

Участники теста не видят друг друга. Если судья не может сказать, кто из собеседников является человеком, то считается, что машина прошла тест. Чтобы протестировать именно интеллект машины, а не её возможность распознавать устную речь, беседа ведется в режиме «только текст», например, с помощью клавиатуры и экрана. Переписка должна идти через контролируемые промежутки времени (чтобы судья не сделал заключение, исходя из скорости ответов). В 1966 г. Джо Вайзенбаум написал программу «Элиза» из примерно 200 строк кода. Она, в основном, лишь повторяла фразы собеседника в форме вопросов - но этого оказалось достаточно, чтобы поразить воображение тысяч людей. Пока ни одна из существующих программ не приблизилась к прохождению теста. Сегодня

существует премия Лёбнера, которая выдается по результатам ежегодного конкурса программам, которые смогли пройти тест Тьюринга. Существуют золотая (зрительная и аудио), серебряная (аудио) и бронзовая (текст) награды. Первые две еще не вручались, бронзовые же медали вручались программам, которые смогли лучше всего симитировать человека во время переписки. Известный футуролог Реймонд Курцвейл, занимающий должность технического директора Google, заявил, что компьютеры смогут с лёгкостью проходить тест Тьюринга к 2029 году.

Пример работы программы «Элиза»

Пациентка - Все мужчины одинаковы.

Врач - В каком смысле?

Пациентка - Они всегда пристают к тебе с чем-нибудь.

Врач - Вы имеете в виду что-нибудь конкретное?

Пациентка - Ну да, это мой парень заставил меня прийти сюда.

Врач - Ваш парень заставил вас прийти сюда?

Пациентка – Да, он говорит, что я почти все время подавлена.

Врач - Прискорбно слышать, что вы все время подавлены.

Пациентка - Это правда. Я несчастна.

Врач - Думаете Ваш приход сюда поможет вам не быть несчастной?

Пациентка - Да. Мне нужна чья-нибудь помощь.

Врач - Что значит для Вас "получить помощь"?

Пациентка - Может быть, я научусь в семье ладить с моей мамой.

Врач - Расскажите поподробнее о вашей семье.....

Структура программы «Элиза» была чрезвычайно проста и включала в себя всего несколько десятков эвристических правил, предназначенных для распознавания грамматических конструкций (путем простого сравнения с образцом) и преобразования поступающих вопросов в «естественные» ответы. В результате эта программа могла вести вполне правдоподобный диалог, зачастую, при этом, отвечая вопросом на вопрос. Дж.Вейценбаум (1923–2008), шокированный поведением своего детища, пришел к выводу, что «основная идеология развития искусственного интеллекта - безнравственна». За свои взгляды на ответственность ученых-программистов перед человечеством снискал в среде специалистов по информатике репутацию диссидента и еретика: «Существуют задачи, выполнение которых не следует поручать вычислительным машинам, независимо от того, можно ли добиться, чтобы вычислительные машины их решали».

Достоинства и недостатки теста Тьюринга. Сильной стороной теста Тьюринга является то, что можно разговаривать о чем угодно. Однако проверяется

только способность машины походить на человека, а не разумность машины вообще. Тест не проверяет, например, способность решать сложные задачи или выдвигать оригинальные идеи. Тест требует, чтобы машина обманывала: она должна притворяться не слишком умной, чтобы пройти тест. Если же машина способна быстро решить некую вычислительную задачу, непосильную для человека, она провалит тест. Тест непрактичен, т.к. он не может быть по-настоящему полезным при разработке разумных машин.

Вариации теста Тьюринга:

Тест **CAPTCHA** — это разновидность обратного теста Тьюринга. Перед тем как разрешить выполнение некоторого действия на сайте, пользователю выдается искаженное изображение с набором цифр и букв и предложение ввести этот набор в специальное поле.

Тест Минского. Система должна прочесть простую детскую книгу, понять сюжет и объяснить его «своими словами» либо задать логичные вопросы.

Китайская комната.

В статье, опубликованной Сёрлем в 1990г., критикуется возможность моделирования человеческого понимания естественного языка и описывается следующий эксперимент.

Возьмём язык, которого Вы не понимаете, например, китайский. Вас поместили в комнату, в которой расставлены корзинки, полные китайских иероглифов. Вам дали учебник на родном языке. В нем есть правила, например: «Возьмите иероглиф из корзины номер один и поместите его рядом с таким-то иероглифом из корзины номер два». Китайцы за дверью передают в комнату наборы символов. Вы в ответ, манипулируя символами по правилам, передаете обратно другие наборы символов. Книга правил — это «компьютерная программа», люди, написавшие её, — «программисты», а Вы играете роль «компьютера». Корзинки с символами — это «база данных»; наборы символов, передаваемых в комнату, это «вопросы», а наборы, выходящие из комнаты, это «ответы». Книга правил такая, что Ваши «ответы» на «вопросы» не отличаются от ответов человека, владеющего китайским.

Вы можете выдержать тест Тьюринга, не понимая ни слова по-китайски! Но Вы никогда не научитесь этому языку в рассматриваемой системе, т.к. нет способа, с помощью которого Вы могли бы узнать смысл хотя бы одного символа. Подобно компьютеру, Вы манипулируете символами, но не можете придать им какого бы то ни было смысла. Вывод Сёрля: *«Компьютеры выполняют чисто автоматические операции, поэтому нет оснований считать их разумными и понимающими».*

Вывод Сёрля требует уточнения того, что такое понимание? Тест Тьюринга выдержал не человек, а система – комната+книга правил+человек. Сёрль «хитрит», обращая внимание только на человека, выполняющего чисто механическую работу. Система является разумной и понимает китайский язык. Сёрль фактически показал обратное своему утверждению - пример эффективного использования искусственных систем

Аргументы «против» искусственного интеллекта.

Машина никогда не сможет мыслить, поскольку сознание - продукт общественного развития.

У изолированной машины сознание не возникнет. На это и человек не способен. Сознание человека зачастую определяется тем, какую информацию вкладывали в мозг в период становления его личности. Машины могут быть обучены схожим образом. На этот путь указывал еще А.Тьюринг. Можно создать сообщество взаимообучающихся интеллектуальных систем.

Мозг - это продукт эволюции за миллиарды лет. Создать мозг, учитывая его необычайную сложность и надежность, нереально.

Эволюция жизни и человека на Земле по Дарвину происходила путем случайных мутаций и естественного отбора миллиарды лет. Человек ставит перед собой четкую цель (создание ИИ), предвидит результат своих действий. Закон причинности не нарушается, зато достигается экономия времени и средств на результаты, которые случайными пробами получились лишь в далеком будущем. Поэтому для создания ИИ нужны не миллиарды лет, а гораздо меньше времени.

Человек состоит из элементов периодической системы. Для изготовления компьютера применяется кремний. Узлы и детали мыслящей машины могут быть более надежны, чем элементы мозга. Надежность мозга обусловлена не столько надежностью его нейронов, сколько их избыточностью, взаимосвязанностью. Установление этих связей - трудная задача. Не следует преуменьшать технических трудностей, стоящих на пути создания мыслящей машины

Р.Пенроуз считает, что реализация ИИ на базе нейронов бесперспективна.

Случайное хаотическое движение молекул газа в сосуде приводит в целом к точной и предсказуемой зависимости между температурой, давлением и объемом (закон Бойля). Нейроны мозга и синаптические связи между ними являются недетерминированной системой. Это означает, по мнению Пенроуза, что мы никогда не сможем понять работу мозга и разума путем анализа отдельных нейронов, реакции которых непредсказуемы. Сознание, разум, память возникают как свойства мозга в целом, а не как свойства его отдельных элементов.

Такое рассуждение не применимо к мозгу. Исследования показали, что многие интеллектуальные функции выполняются отдельными частями мозга. Аналогии Пенроуза с хаотическими частицами не совсем уместны для мозга.

Аргументы «за» искусственный интеллект

1. Создание нового разума биологическим путем для человека дело вполне привычное. Дети большую часть знаний приобретают путем обучения. Знания не заложены в них заранее. Это утверждение не доказано, но по внешним признакам все выглядит именно так.

2. Игра в шахматы, шашки, распознавание зрительных и звуковых образов, синтез инженерных решений, на практике оказалось не таким уж сложным делом (теперь работа ведется не на уровне возможности или невозможности реализации перечисленного, а о нахождении наиболее оптимального алгоритма). Есть надежда, что и полное моделирование мышления человека и самовоспроизведения ИС окажется не таким уж и сложным делом.

3. ИИ не противоречит Библии. Даже люди далекие от религии, знают слова священного писания: "И создал Бог человека по образу и подобию своему ...". Возможно, мы сможем создать кого-то по образу и подобию человека.

Лауреаты премии Тьюринга в области ИИ:

Джон Маккарти (1971) – за лекцию «Современное состояние исследований по ИИ», которая освещает состояние дел в этой области, и существенные успехи его работ по ИИ.

Аллен Ньюэлл/ Герберт Саймон (1975) – за фундаментальный вклад в ИИ, психологию механизмов человеческого восприятия и обработку списков.

Фейгенбаум/Редди (1994) – за первопроходческие разработки, создание крупномасштабных СИИ и демонстрацию практической важности и потенциальной коммерческой выгоды от технологий использующих ИИ.

Дуглас Энгельбарт (1997) – за вдохновляющее предвидение будущего развития интерактивных вычислений и изобретение гипертекстовых технологий

Лесли Вэлиант (2010) – за новаторские подходы в разработке СИИ, самообучающихся алгоритмов, интеллектуального поиска и программ распознавания образов, письма и речи.

Джуда Перл (2011) – за фундаментальный вклад в ИИ посредством разработки исчисления для проведения вероятностных и причинно-следственных рассуждений.

Крупные центры исследований ИИ

МТИ, Стенфорд, Белл лаб., Беркли, Токийский, Марсельский, Эдинбургский Uni; СО РАН, РАИИ, МЭИ, МГТУ, НИИ СИИ, ЮФУ.

УЧЕБНОЕ ПОСОБИЕ «ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ»
Основные направления исследований в области ИИ

1. *Программы решения отдельных интеллектуальных задач* (компьютерное доказательство теорем, семантический анализ и обработка ЕЯ-информации, распознавание образов, игровые программы)
2. *Работа со знаниями* (Модели знаний, представление, извлечение, обработка, управление знаниями, вывод на знаниях)
3. *Интеллектуальное программирование*
4. *Интеллектуальные автоматизированные системы* (АСУ, САПР, АСНИ, роботы, экспертные системы)

Происхождение и понимание термина «интеллект»

Понятие интеллекта происходит от латинских понятий *intellego* (замечать, понимать, подразумевать) и *intellectus* (понимание, познание). Со временем интеллект стали отождествлять со всеми возможностями человеческого сознания – умом, рассудком, разумом – вообще, с мыслительными способностями человека.

Признаки естественного интеллекта:

- Способность обучаться (например, благодаря любознательности);
- Способность творить, создавать новую информацию);
- Способность адаптироваться к среде;
- Способность классифицировать явления, ситуации, объекты;
- Способность к анализу (дедукции) и синтезу (индукции) как методам познания;
- Чувство юмора, рефлексия и др.

Происхождение термина «искусственный интеллект».

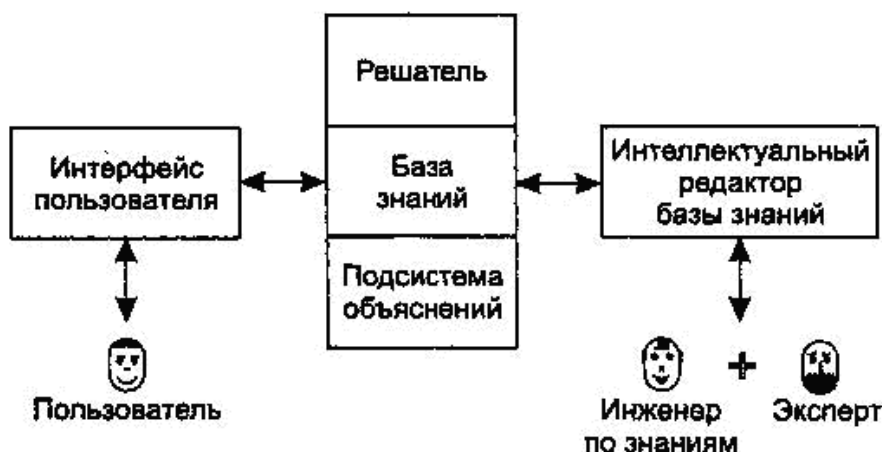
Термин Artificial intelligence (AI) был предложен в 1956 г. профессором Дж.МакКарти: наука и технология создания интеллектуальных машин и компьютерных программ.

ИИ и интеллект человека - это разные понятия. Большинство работ в ИИ касаются изучения проблем, которые требуется решать человечеству в промышленном и технологическом смысле. Поэтому ИИ-исследователи вольны использовать методы, которые не наблюдаются у людей, если это необходимо для решения конкретных проблем.

Понимание термина ИИ в России. Следует признать, что английское словосочетание Artificial Intelligence перевели на русский язык неудачно. Во-первых, термин «Искусственный интеллект» имеет слегка фантастическую антропоморфную окраску. Во-вторых, слово Intelligence означает «умение

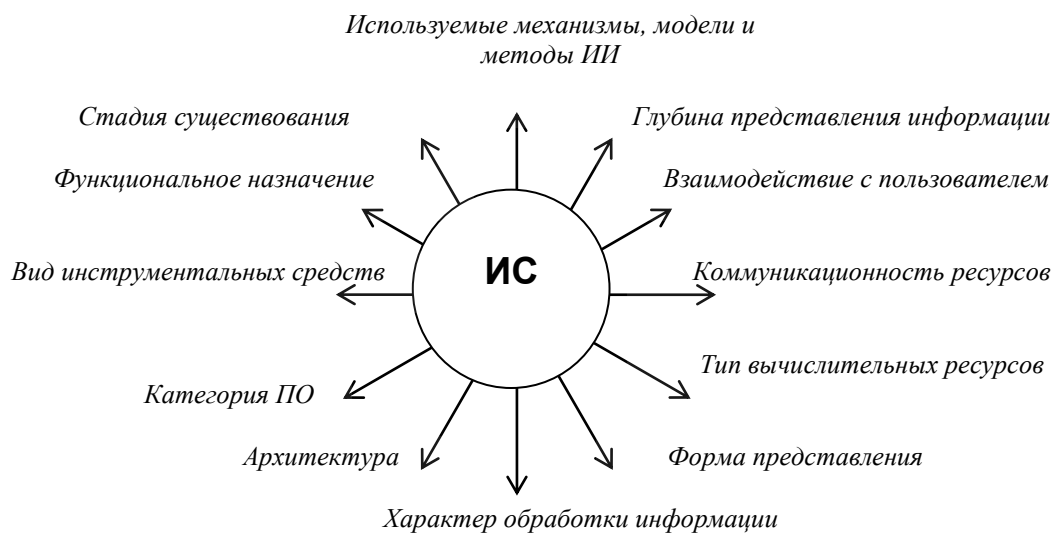
рассуждать разумно», а вовсе не «интеллект», для которого есть английский аналог intellect.

*Интеллектуальная система (ИС, **Intelligent System**)* — это техническая или программная система, способная решать творческие задачи в конкретной предметной области, знания о которой хранятся в главном блоке такой системы - базе знаний.



Кроме БЗ в структуру ИС обычно включают решатель и интеллектуальный интерфейс.

Многомерная классификация ИС



ИИ является междисциплинарной наукой. Это обстоятельство подчёркивает отсутствие единой классификации СИИ.

Например, классификация ИС по функциональному назначению

- экспертные системы,
- компьютерные системы с обработкой естественного языка,
- компьютерные системы интеллектуального анализа данных,

- обучающие системы,
- системы распознавания, анализа и синтеза изображений, речи,
- системы поддержки принятия решений,
- робототехника (интеллектуальные роботы),
- многоагентные интеллектуальные системы,
- интеллектуальные САПР, интеллектуальные системы реинжиниринга предприятий (CASE),
- машинное творчество.



Интеллектуальной информационной системой называют автоматизированную информационную систему, основанную на знаниях, или комплекс программных, лингвистических и логико-математических средств для реализации основной задачи – осуществления поддержки деятельности человека и поиска информации в режиме продвинутого диалога на естественном языке.










Информационно-вычислительными системами с интеллектуальной поддержкой для решения сложных задач называют те системы, в которых логическая обработка информации превалирует над вычислительной.

Таким образом, любая информационная система, решающая интеллектуальную задачу или использующая методы искусственного интеллекта, относится к интеллектуальным. Для интеллектуальных информационных систем характерны следующие признаки:

- развитый интерфейс, возможность формулирования произвольного запроса на языке, приближенном к естественному;
- умение решать сложные плохо формализуемые задачи, для которых характерны неопределенность и динамичность исходных данных и знаний;
- способность к самообучению;
- адаптивность – способность к развитию системы в соответствии с изменениями проблемной области.

Иерархическая классификация подразумевает следующий перечень интеллектуальных систем, основанных на знаниях:


 <p>ПРИМЕР</p>	<ul style="list-style-type: none"> • экспертные системы (динамические, статические), их оболочки, средства разработки и основные подсистемы;
 <p>ПРИМЕР</p>	<ul style="list-style-type: none"> • информационные системы (гипертекстовые, мультимедийные, геоинформационные, виртуальной реальности, электронные библиотеки и др.);

 ПРИМЕР	<ul style="list-style-type: none"> • компьютерные системы обработки текста на естественном языке (рубрикация текстов, аннотирование и реферирование, машинный перевод, интеллектуальный семантический анализ текстов, обучение, электронные словари и др.);
 ПРИМЕР	<ul style="list-style-type: none"> • системы интеллектуального анализа данных;
 ПРИМЕР	<ul style="list-style-type: none"> • системы когнитивной графики и машинного обучения;
 ПРИМЕР	<ul style="list-style-type: none"> • системы распознавания образов (OCR-системы, синтез изображений, анализ и синтез речи);
 ПРИМЕР	<ul style="list-style-type: none"> • системы поддержки принятия решений (многокритериальные, управления технологическими процессами в реальном времени, планирования, имитационного моделирования, ситуационного управления и др.);
 ПРИМЕР	<ul style="list-style-type: none"> • интеллектуальные роботы (техническое зрение, мобильные роботы и др.);
 ПРИМЕР	<ul style="list-style-type: none"> • многоагентные интеллектуальные системы;
 ПРИМЕР	<ul style="list-style-type: none"> • интеллектуальные САПР, АСУ, CASE;
 ПРИМЕР	<ul style="list-style-type: none"> • системы машинного творчества (литература, живопись, графика, игры, доказательство теорем).


Междисциплинарность ИИ диктует различные подходы к его изучению:


- как совокупность элементарных процессов переработки информации (информационный подход);
- как продукт обучения (образовательный подход);
- как особая форма сознания (феноменологический подход);


- как особая форма человеческой деятельности (деятельностный подход);
- как система познавательных процессов (структурный подход);
- как результат социализации (социокультурный подход);
- как следствие адаптации к внешнему миру (эволюционный подход).


 СПРАВКА	<p><i>Л. Заде</i> даже ввёл понятие коэффициента машинного интеллекта (MIQ), который в отличие от IQ изменяется во времени. Это понятие является машинно-зависимым и опирается на такую характеристику, как распознавание образов, которая для определения IQ человека считается тривиальной.</p>
---	---

В рамках ИИ сегодня обсуждается множество вопросов:

 Вопрос	<p>Существует или нет универсальный алгоритм обработки всех данных, поступивших в мозг от органов чувств ?</p>
--	--

 Вопрос	<p>Правильно ли мы понимаем принцип работы мозга? Ведь мозг не отвечает выходным сигналом на каждую информацию. Мозг прогнозирует, люди действуют, результаты действий мозг запоминает и строит новые прогнозы, чтобы правильно предсказать и понимать, что будет дальше, в отличие от компьютера, который главным образом считывает данные, мало рассуждает и прогнозирует.</p>
---	--

 Вопрос	<p><i>Тьюринг</i> (машина Тьюринга), <i>Пост</i> (продукционные системы), <i>Марков</i> (марковские цепи), <i>Чёрч</i> (рекурсивные функции) предложили мощные модели для формализации понятий и описания вычислений. Может, для решения интеллектуальных задач не нужны мощные вычислители, а важнее построить модели рассуждений?</p>
--	---

 СПРАВКА	<p>В книге «Космическая Одиссея 2001 года» Артур Кларк рассматривает интересные морально-этические, социальные и философские аспекты этих и других вопросов</p>
---	---

Разумно говорить о «сильном» и «слабом» ИИ. Создание «сильного» ИИ является мегапроблемой. Она заключается в разработке компьютера, способного думать, как люди. Создание «слабого» ИИ заключается в разработке широкого диапазона новых информационных технологий, добавление которых в существующие системы придаст им «разумные» свойства.

Лекция «Постулаты искусственного интеллекта. Нейроинформатика»

Постулат нейроинформатики

Единственный объект, способный мыслить — это человеческий мозг. Поэтому любое «мыслящее» устройство должно каким-либо образом воспроизводить структуру человеческого мозга

Постулат кибернетики черного ящика (постулат логико-символьных систем)

Для искусственного интеллекта не важно, как устроено «мыслящее» устройство. Главное, чтобы на заданные воздействия оно реагировало бы так же, как и мозг человека. По сути, это постулат А. Тьюринга, который считал, что все проблемы, решаемые людьми, могут быть сведены к набору алгоритмов.

Научные школы в ИИ

В соответствии с постулатами в ИИ существуют две научные школы: конвенциональный (традиционный) ИИ и вычислительный интеллект.

1. *Конвенциональный* (традиционный) ИИ. Методы конвенционального ИИ основаны на машинном самообучении, формальной логике и статистическом анализе и включают:

- Экспертные системы (программы основаны на базе знаний и механизмах вывода по определенным правилам, обрабатывают информацию и выдают заключение на её основе);
- Рассуждения на основе аналогий;
- Байесовские сети (статистический метод анализа данных);
- Поведенческие модели (включают программы-агенты, поведение которых зависит от изменений внешней среды).

2. *Вычислительный ИИ*. Методы вычислительного ИИ основаны на поэтапной разработке и обучении систем на несимвольных эмпирических данных и «мягких» вычислениях и включают:

- Нейросети;
- Нечеткие системы (рассуждения в условиях НЕ-факторов);
- Эволюционные вычисления (эволюционные алгоритмы, разновидности Монте-Карло, гармоничный поиск, роевые алгоритмы, меметика и т.д.);
- Гибридные системы (например, исходные экспертные правила генерируются нейросетью, а заключения - байесовской сетью).

Современное состояние:

- Нейросетевой подход очень привлекателен. Трудности связаны с необходимостью переобучение сети, локальной оптимальностью решений, выбором адекватной архитектуры сети и, главное, нельзя выяснить, как было получено определенное решение задачи. Нейросетевой подход пока не применим даже для моделирования мозга муравьев и пчел;

- Разработка гибридных моделей и алгоритмов вывода знаний в условиях неопределенности в реальном времени, онтологий знаний, многоагентных систем, роботов, игровых программ.

Наиболее значимые достижения в области теории:

- Формирование нейроматематики, создание нейроалгоритмов и нейрокомпьютеров на их основе, разработку «мягких» вычислительных моделей мозга, связанных с восприятием, языком и знаниями, разработка эволюционных и квантовых алгоритмов обучения нейросетей, распознавание речи и роботы, соревнующиеся по футболу;

- Тест Тьюринга, модели и языки для представления знаний, разработка ЭС, автоматическое доказательство теорем, Deep Blue победил чемпиона мира по шахматам, Интернет-проект на сайте 20q.net по мотивам игры «20 вопросов».

Сферы практического применения:

- Системы распознавания; прогнозирование в бизнесе, экономике, медицине; управление в космосе, робототехнике, металлургии, машиностроении, нефте- и газовой промышленности; решение трудных физ.-мат. задач моделирования в реальном времени;

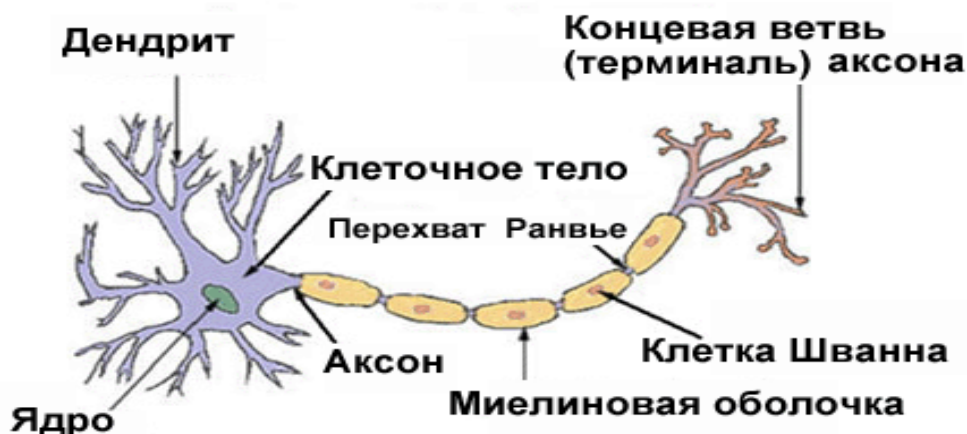
- Экспертные системы в медицине, геологии, бизнесе, экономике, промышленности; системы управления знаниями и принятия решений, обучающие системы, системы интеллектуального анализа данных, семантический анализ и обработка информации на естественном языке; визуализация и интеллектуализация интерфейса.

Нейроинформатика

Параметры мозга как системы обработки информации:

Количество нейронов мозга составляет около 10^{21} .

Структура биологического нейрона:



Объем коры: толщина 2-3 мм, площадь около 2200 см².

Потребляемая мощность: 1 – 25 Вт.

Нейрон имеет в среднем 10000 связей.

100 триллионов синапсов.

Огромные объемы информации мозг обрабатывает за доли секунды при том, что мозг – тихход (время реакции нейрона несколько миллисекунд).

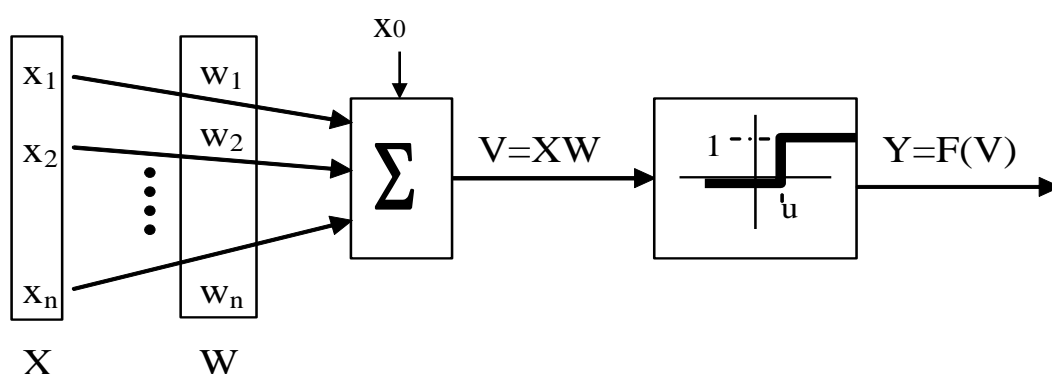
Объем передаваемой мозгом информации равен:

10^{12} (волокон) \times 10^3 (импульсов) = 10^{15} .

В лучших ЭВМ: 10^3 (каналов) \times 10^9 (импульсов в сек) = 10^{12} , т.е. 1000 лучших ЭВМ равнозначны мозгу по производительности.

Нейроморфный чип IBM True North: 5,4 млрд. транзисторов (28nm), расход энергии 72 mW (примерно 400×10^9 синаптических операций/с*W), новый язык программирования Corelet.

Искусственный нейрон МакКаллока - Питтса



На вход нейрона поступает некоторое множество сигналов x_j , $j=1, 2, \dots, n$, каждый из которых является выходом другого нейрона. Нейрон вычисляет взвешенную сумму $V = \sum x_j w_j$ входных сигналов x_j и формирует на выходе сигнал величины **1**, если эта сумма превышает определенный порог θ , и **0** - в противном случае.

Нейрон описывается математической моделью в виде уравнения

$$Y = F(V) = F\left(\sum_j x_j w_j > \theta\right), \quad j = 1, 2, \dots, n,$$

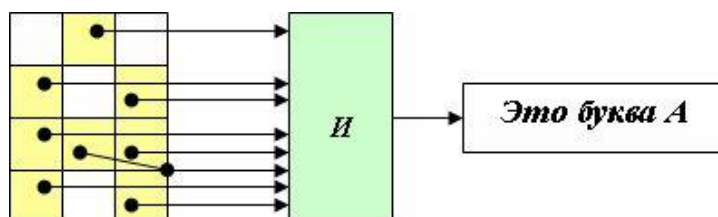
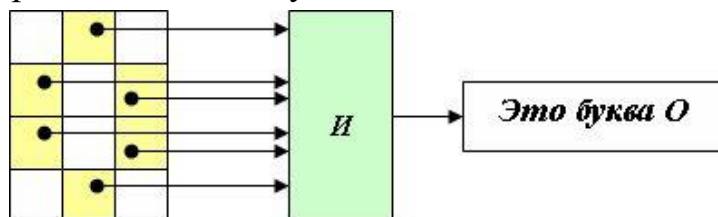
где Y – выходной сигнал нейрона, F – функция активации выхода нейрона, w_j – «вес» входа x_j , θ – пороговое значение.

Добавив постоянный единичный вход $x_0=1$ и положив $w_0 = -\theta$, получим

$$Y = F\left(\sum_j x_j w_j + x_0\right), \quad j=0, 1, 2, \dots, n.$$

Нейронная сеть для распознавания символов

Пусть перед нами экран, разбитый на двенадцать клеток (4x3). При фокусировании изображения клетка либо засвечивается, либо нет. "Засветка" определяет единичное значение величины ее возбуждения, "не засветка" — нулевое. Так, буквы О и А определяют засветку клеток, как это показано на рисунках:



Что надо сделать, чтобы некий конструируемый нами прибор мог сказать, какая это буква?

Очевидно, надо все сигналы возбуждения клеток экрана, засвечиваемые буквой О, подать на конъюнктор, реализующий схему И. Единичный сигнал на выходе конъюнктора, как показано на рис.2.1, сформируется тогда и только тогда, когда засветятся все клетки экрана, на которое ложится изображение буквы О. Наличие единичного сигнала на выходе конъюнктора и определит ответ: "Это буква О".

То же необходимо сделать и для буквы А.

Пометим каждую клетку экрана ее координатами. Тогда на языке математической логики сделанное нами можно записать в виде логических высказываний — предикатов:

$$(1,2) \wedge (2,1) \wedge (2,3) \wedge (3,1) \wedge (3,3) \wedge (4,2) \rightarrow O$$

$$(1,2) \& (2,1) \& (2,3) \& (3,1) \& (3,2) \& (3,3) \& (4,1) \& (4,3) \rightarrow A$$

Буквы не будут "мешать" друг другу, так как засветка соответствующих им клеток экрана частично не совпадает, и единичное значение конъюнкции определится только для одной из них.

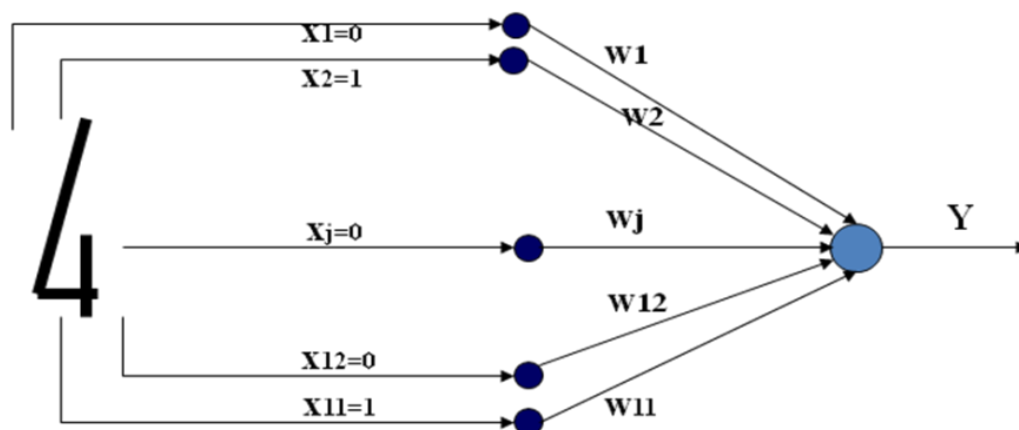
А если на экран подать букву **К**? Тогда ни один из двух конъюнкторов не выработает единичное значение, так как не произойдет полное совпадение засветки соответствующих им клеток экрана. Чтобы "научить" систему букве **К**, необходимо ввести еще один конъюнктор и проделать те же построения, что и выше. Таким образом, мы можем говорить, что построили систему распознавания двух "правильно" заданных букв.

Но что делать, если буквы на экране пишутся дрожащей рукой? Тогда мы должны разрешить альтернативную засветку каких-то соседних клеток экрана и учитывать это с помощью операции дизъюнкции, ИЛИ. Например, можно построить предикат для распознавания буквы **О**, допустив, например, возможность засветки клеток (1,1), (1,3), (4,1), (4,3). Аналогично, для буквы **А** можно допустить засветку клеток (1,1) и (1,3) и построить соответствующий предикат.

Объединив оба предиката, можно получить систему распознавания двух букв **О** и **А**.

Персептрон Розенблатта и правило Хебба

Идея нейрона Мак-Каллоха–Питса была материализована Розенблаттом вначале в виде компьютерной программы, а затем в виде электронного устройства-персептрона, моделирующего глаз, который обучался распознаванию символов. Например персептрон на 12 фотоэлементах для различения четных и нечетных цифр имеет вид:



На матрицу накладывается карточка с изображением цифры 4. Фотоэлемент срабатывает (сигнал $x_j=1$), если на него попадает фрагмент цифры, иначе – $x_j=0$. Цель обучения: $y=1$, если на карточке четная цифра, $y=0$ – в противном случае. Она достигается путем корректировки весов w_j по правилам **Хебба**:

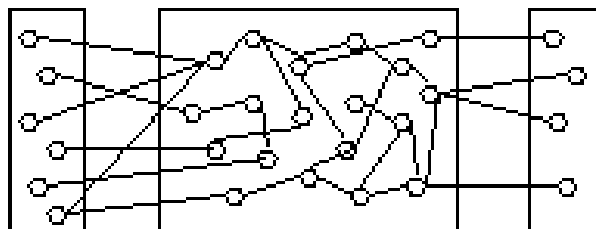
- если выход неправильный и равен 0, то увеличить веса для $x_j=1$;
- если выход неправильный и равен 1, то уменьшить веса для $x_j=1$.

Теорема (о сходимости персептрона).

Если существует множество значений весов, обеспечивающих распознавание образов, то в конечном итоге алгоритм его обучения приведет к этому множеству.

От простого персептрона к многослойным нейросетям

Понятно, что пока никто не знает, как на самом деле функционирует мозг, как он строит связи внутри себя, как происходит процесс обучения нейронной сети мозга? Есть лишь общее представление об этом:

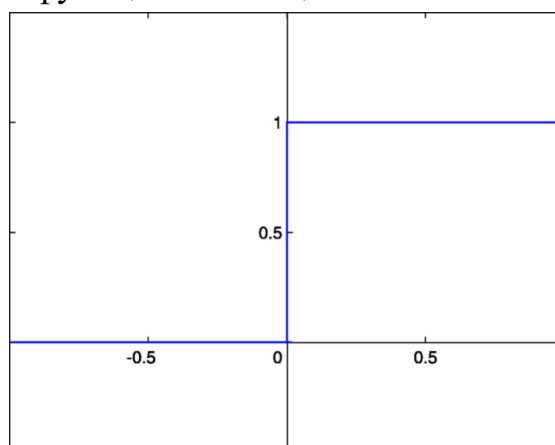


Рецепторы Интернейроны Эффекторы

Нейронов случайное число и соединены они случайно. Требуется построить алгоритм соединения, после которого модель будет действовать целесообразно. Персептрон является лишь простейшей попыткой моделирования мозга

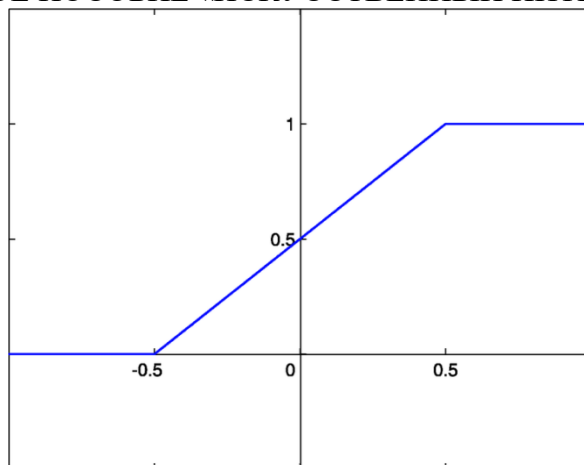
Круг решаемых персептроном задач значительно расширяется, если выходные сигналы будут не только 0 или 1, а могут принимать непрерывные значения. Использование различных функций активации позволяет вносить нелинейность в работу нейрона и нейросети. На практике и в теоретических исследованиях используются различные функции активации: пороговая (функция Хевисайда), линейная, логистическая, гиперболический тангенс, радиально-базисная, экспонента, тригонометрический синус, модульная, квадратичная.

Например, пороговая функция активации описывается формулой вида:



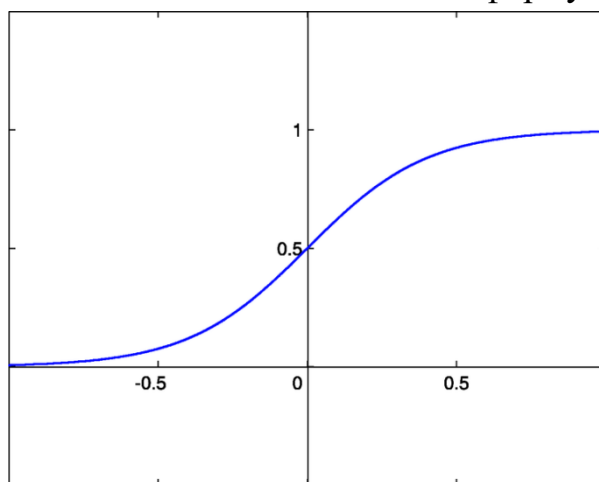
$$Y(x) = \begin{cases} 1, & \text{если } x \geq \Theta, \\ 0, & \text{иначе.} \end{cases}$$

Линейная функция активации описывается формулой вида:



$$Y(x) = \begin{cases} 0, & \text{если } x \leq -0,5 \\ 1, & \text{если } x \geq 0,5 \\ x, & \text{иначе.} \end{cases}$$

Логистическая функция активации описывается формулой вида:



$$Y(x) = \frac{1}{1 + \exp(-ax)},$$

где a – параметр функции, определяющий её крутизну.

Когда a стремится к бесконечности, функция вырождается в пороговую. При $a = 0$ сигмоида вырождается в постоянную функцию со значением 0,5.

Недостатками пороговой и линейной активационных функций является их недифференцируемость на всей оси абсцисс. Как следствие, нейроны с такими функциями нельзя использовать в сетях, обучающихся по алгоритму обратного распространения ошибки и другим алгоритмам, требующим дифференцируемости активационной функции. Использование сигмоидальных функций, напротив, позволило перейти от бинарных выходов нейрона к аналоговым. Нейроны с сигмоидальными функциями, чаще всего используются в скрытых (внутренних)

слоях многослойных нейросетей. Достоинством логистической функции является простота её производной:

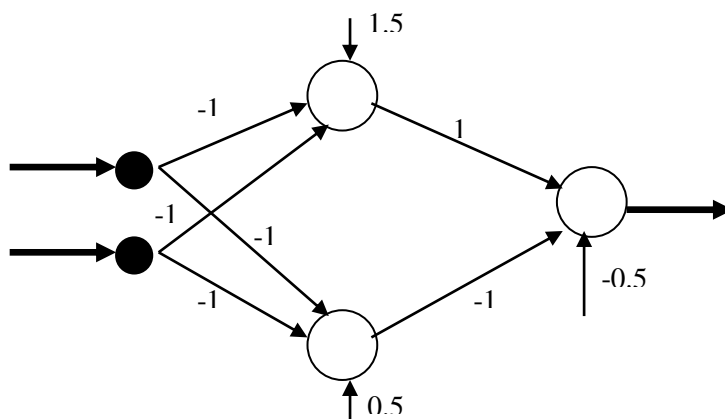
$$\frac{dY(x)}{dx} = tY(x)(1 - Y(x)).$$

Это привело к новым подходам к обучению персептронов: *минимизировать среднеквадратичную ошибку*

$$\varepsilon = \frac{1}{2} \sum_i (Y_{i \text{ эталон}} - Y_{i \text{ факт}})^2.$$

Однако Минский и Пайперт доказали, что однослойные персептроны не могут решать многие простые задачи, например, реализовать логическую функцию «Исключающее ИЛИ» (XOR). Так появились многослойные нейросети, в которых между слоями входных и выходных нейронов располагались нейроны скрытого слоя.

Проиллюстрируем это на примере. Рассмотрим модель многослойной нейросети следующего вида:



Модель содержит два входных элемента, два скрытых элемента и один выходной элемент. Все связи идут только в направлении от входного слоя к выходному. Скрытые элементы не получают данных от внешней среды непосредственно и не посылают данные непосредственно во внешнюю среду.

Функция XOR отображает пару двоичных входных сигналов:

Входы		Выход
x_1	x_2	XOR
1	1	0
1	0	1
0	1	1
0	0	0

В нашем случае значение комбинированных входов вычисляется по формуле

$$V_j = \sum_{i=0}^n x_i w_{ij},$$

а выходное значение нейрона получается как результат применения пороговой функции:

$$Y(V_j) = \begin{cases} 1, & \text{если } V_j \geq 0, \\ 0, & \text{если } V_j < 0. \end{cases}$$

В качестве вводимых данных рассмотрим вначале пару $x_1=1$, $x_2=1$. Для первого скрытого элемента со смещением 1.5 получаем

$$V=(x_0 \times 1.5)+(x_1 \times (-1))+(x_2 \times (-1)) = (1 \times 1.5)+(1 \times (-1))+(1 \times (-1)) = -0.5,$$

Поэтому выходным значением элемента будет 0. Для второго скрытого элемента со смещением 0.5 получаем

$$V=(x_0 \times 0.5)+(x_1 \times (-1))+(x_2 \times (-1)) = (1 \times 0.5)+(1 \times (-1))+(1 \times (-1)) = -1.5,$$

Поэтому выходным значением будет 0. Если процедуру повторить для трех оставшихся пар входов, то можно убедиться в том, что вывод представленной выше сети соответствует функции XOR.

Основным алгоритмом обучения многослойных нейросетей является *алгоритм обратного распространения ошибки (back propagation)*.

Его идея состоит в следующем. Мы знаем, что должно быть на выходе нейросети (по обучающей выборке) и постепенно вычисляем вход от слоя к слою по цепочке определенных формул. Реализуется как бы *метод градиента*. Задача стоит в следующем: найти все w_{ij} , то есть настроить веса всех связей так, чтобы нейросеть выдавала нужный выходной сигнал на соответствующий входной. Для настройки (обучения) нейросети на задачу необходимо реализовать множество итераций. Цель уменьшить ошибку ε до нуля. В результате находятся все лучшие значения w_{ij} . Обучение происходит экспоненциально. Если ошибка E не приходит к нулю, то означает, что сложности сети мало для обучения данному примеру (примерам), количество слоев или нейронов в слоях надо увеличить.

Общие признаки нейросетевой технологии следующие. Система начинает обнаруживать закономерности во входной информации. Система не знает, как она обучается – ей безразличен предмет рассуждений. Система легко доучивается и переучивается. Нейросеть можно "недокормить" примерами, но можно и «перекормить» (переобучить).

Оптимизация обучения нейросети

Таким образом, существует две парадигмы обучения нейронных сетей – с учителем и без учителя. В первом случае, на входной вектор имеется готовый ответ, во втором случае нейронная сеть самообучается. У каждого вида обучения есть своя

ниша задач и по большому счету они не пересекаются. На данный момент придумано и запатентовано большое количество архитектур нейронных сетей и методов их обучения. Но основными (исходными) являются – для обучения с учителем это «алгоритм обратного распространения ошибки», а для обучения без учителя это алгоритмы Хебба и Кохонена. В последнее время, сформировалась еще одна парадигма – обучение с подкреплением. В этой парадигме есть и обучение с учителем и обучение без учителя.

Другие архитектуры нейросетей

Строгой теории выбора оптимального числа скрытых слоев для многослойных нейросетей пока нет. Их возможности математически не доказаны. Эффективными оказались гибридные алгоритмы их обучения на основе эволюционных алгоритмов.

Кроме того, нейрофизиологи показали, что между биологическим нейронами мозга имеются не только прямые, но и обратные связи, нейронные структуры способны самообучаться. Так появились новые архитектуры: сети Хопфилда, Кохонена

Нейропакеты, нейровычислители, нейрочипы

Сегодня в области проектирования нейросетей разработаны разнообразные нейропакеты (программные имитаторы), нейровычислители и нейрочипы.

Наиболее известными нейропакетами являются *NeuNet Toolbox MatLab*, *Neuro Windows*, *Neuro Office*, *NNet+* (универсальные пакеты), ряд специализированных нейропакетов для обработки изображений (*ImageLib*, *Wisard NVision*), распознавания образов, текстов и речи (*NNetSheet-C*, *ICPAC*, *Propagator*, *Synaptics*, *NIntelligence Voc*, *AQIRE*), управления динамическими объектами, обработки сигналов, финансового анализа, а также средства для проектирования гибридных нейросетевых систем (*G2*, *GURU*, *Nexpert Object*, *GeneHunter*, *Evolver*, *Genesis*, *Fuzzi Calc*, *Judgement Maker*, *Fuzzi Logic Toolbox Matlab*).

Современный уровень теории ИНС, нейроматематики, схемотехники и микроэлектроники позволяет создавать на аппаратном уровне нейровычислители и нейрочипы. Они реализуются на заказных кристаллах, микроконтроллерах, программируемых логических интегральных схемах (ПЛИС), транспьютерах, цифровых сигнальных процессорах (DSP) или нейрочипах.

Нейрочип – это специализированная СБИС, ориентированная на реализацию нейросетевых алгоритмов (начало разработок – середина 90-х годов). По принципам построения, назначению и характеристикам они сильно отличаются друг от друга (аналоговые, цифровые, гибридные).

Недавно в США был создан прототип *супернейрокомпьютера* – системы обработки изображений с производительностью 80 pflops ($80 \cdot 10^{15}$ операций с

УЧЕБНОЕ ПОСОБИЕ «ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ»
плавающей точкой в секунду) при объеме, равном мозгу, и потребляемой мощности 20 Вт.

Преимущества нейросетевых систем

Подытожим привлекательные черты распределенной обработки информации в нейросетевых системах:

- Параллелизм обработки информации - глобальность связей между нейронами. До обучения эти связи произвольны и обычно малы. Обучение на примерах "проявляет" конкретную структуру сети под конкретную задачу;
- Единый и эффективный принцип обучения нейросетей - минимизация эмпирической ошибки методом ее обратного распространения по сети. Извне задается лишь цель обучения - то есть способ определения ошибки по выходам сети. Далее сеть постепенно модифицирует свою конфигурацию, минимизируя эту ошибку, то есть все лучше справляясь с возложенной на нее задачей;
- Надежность функционирования. Избыточность связей приводит к тому, что значения каждого веса по отдельности не играют решающей роли. Вывод из строя ограниченного числа нейронов или обрыв некоторых связей не сказываются критическим образом на качестве работы всей сети;
- Способность решать неформализованные задачи - следует из способности нейросетей самостоятельно вырабатывать весьма сложные алгоритмы обработки данных, формализовать которые самостоятельно зачастую не могут даже лучшие эксперты в данной предметной области. Отсюда - относительная дешевизна нейросетевых разработок.

Почему же нейрокомпьютинг при таком наборе преимуществ стал активно применяться на практике лишь в 90-е годы? Очевидно, что появление нейросетевых систем должно быть востребовано экономикой. Именно экономическая потребность в суперкалькуляторах вызвала к жизни последовательные ЭВМ, способные решать любые формализованные задачи обработки как численной, так и символьной информации. Это, в свою очередь, подняло на должную высоту проблему человеко-машинного интерфейса. Отсюда и возросший интерес к проблемам искусственного интеллекта и нейросетей. Времена меняются. Вместо разрозненных персональных компьютеров появилась Сеть с ее неисчерпаемыми информационными ресурсами.

Лекция «Интеллектуальные задачи и проблем. Данные-информация-знания»

Что нужно для создания интеллектуальной системы?

1. Создать материальную систему поддержки ИИ (современные компьютеры не вполне подходят для этого).
2. На этой базе создать активную информационную структуру, потенциально способную к обучению и саморазвитию, к решению интеллектуальных задач.
3. Сформировать структуру и функции ИС во взаимодействии с экспертами, пользователями и окружающей средой. При этом задача экспертов и инженеров по знаниям - формализовать знания в предметной области, запрограммировать и вложить формализованные знания в систему, решить задачи обработки и вывода на знаниях (дедуктивного, индуктивного, абдуктивного и др.).

Если задачу относят к интеллектуальным, то любое ее решение свидетельствует о некотором интеллекте. Если решение добывается программным путем, то эта программа содержит элементы интеллекта. Программа, умеющая решать задачи в предметной области, которую традиция относит к интеллектуальным, называется программой с элементами искусственного интеллекта.

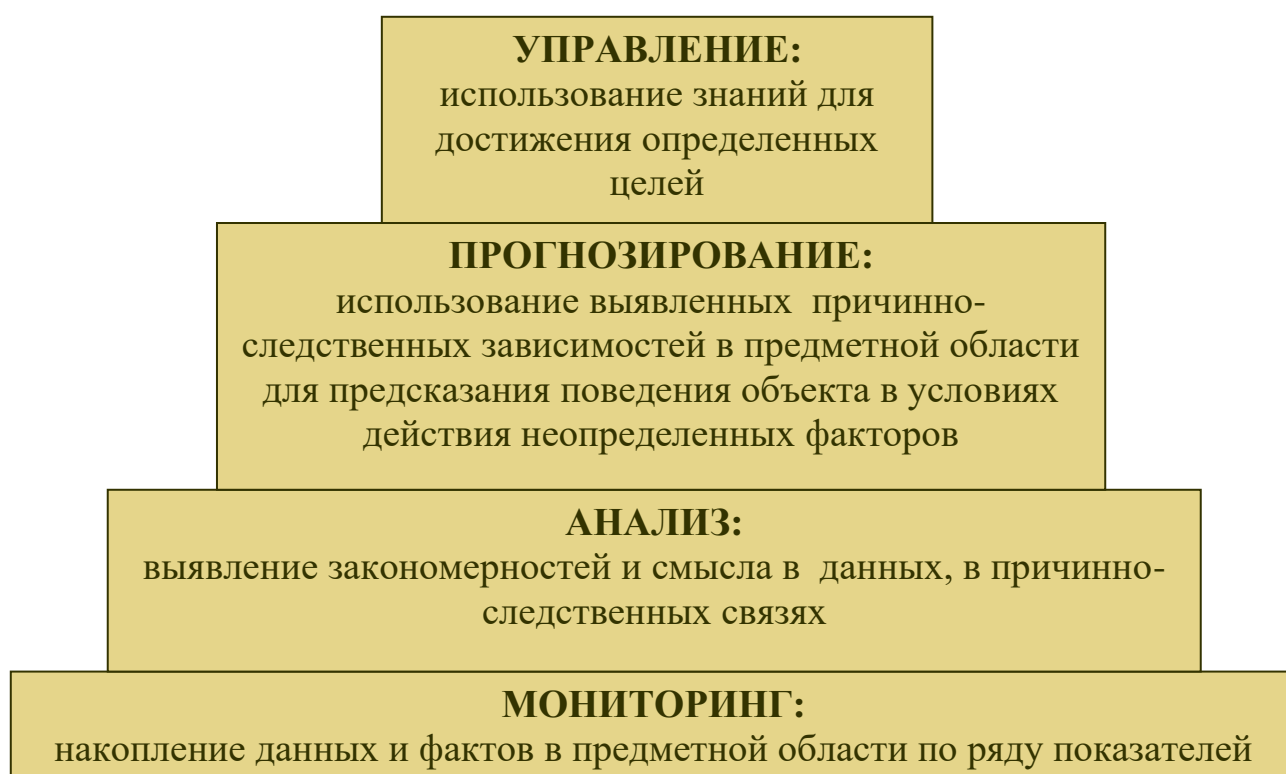
По мере прогресса меняются традиции отнесения предметных областей к интеллектуальной сфере, и меняется оценка степени «интеллектуальности» многих задач. Например, расчет конструкции Эйфелевой башни в конце XIX века потребовал огромного труда, недюжинного интеллекта и изобретательности, а сейчас подобный расчет может быть выполнен рядовым инженером-конструктором с помощью готового пакета программ быстро, надежно и без особых умственных усилий. Такой пакет, вне всякого сомнения, демонстрирует интеллектуальное поведение, однако мы не считаем его системой ИИ, поскольку все расчеты в конструкторском пакете выполняются по хорошо известным, заранее заданным однозначным алгоритмам.

Классификация интеллектуальных задач

- Анализ данных - процесс выявления закономерностей и смысла в данных, в причинно-следственных связях путем многовариантного анализа данных.
- Диагностика - процесс соотнесения объекта с некоторым классом объектов и/или обнаружения неисправности в системе, заболевания живых организмов, природных аномалий и т.д.
- Мониторинг - наблюдение и накопление по ряду показателей данных и фактов в предметной области, сигнализация о выходе параметров за допустимые пределы.

- Обучение - особым образом организованное общение между теми, кто обладает знаниями и определённым опытом, и теми, кто их приобретает, усваивает.
- Планирование – поиск планов действий, относящихся к объектам, способным выполнять некоторые функции.
- Поддержка принятия решения – совокупность процедур, обеспечивающая ЛПР необходимой информацией для оптимального выбора решений среди альтернатив.
- Прогнозирование – предсказание событий или явлений на основании анализа имеющихся данных.
- Проектирование – подготовка спецификаций на создание объектов с заранее определёнными свойствами.
- Управление – процессы использование знаний для достижения определенных целей системы в соответствии с заданными спецификациями.

«Пирамида» интеллектуальных задач и проблем имеет вид, представленный на рисунке, и отражает возрастание степени сложности задач и проблем (от основания пирамиды к вершине):



Экспертные системы являются одним из примеров систем, основанных на знаниях, способных решать указанные выше интеллектуальные задачи. Это компьютерная программа, способная частично заменить специалиста-эксперта в разрешении проблемной ситуации. ЭС разрабатываются с 1970-х г., а в 1980-х получили коммерческое подкрепление. Экспертные системы – один из видов СИИ,

которые получили широкое распространение и нашли практическое применение. Существуют экспертные системы по военному делу, геологии, инженерному делу, информатике, космической технике, математике, медицине, метеорологии, промышленности, сельскому хозяйству, управлению, физике, химии, электронике, юриспруденции и т.д.

Предтечи ЭС были предложены в России в 1832г. С.Н.Корсаковым. Он создал механические устройства (назвал их «интеллектуальными машинами»), позволявшие находить решения по заданным условиям. Например, определять наиболее подходящие лекарства по наблюдаемым у пациента симптомам заболевания.

Особенности ЭС известны. Они ориентированы на решение задач на основе дедуктивных суждений в узкой предметной области, способны объяснять свои рекомендации и выводы, способны приобретать новые знания от экспертов.

Применение ЭС позволяет, например, при проектировании интегральных микросхем повысить производительность труда, при диагностике ускорить поиск неисправностей; повысить производительность труда программистов; при профессиональной подготовке сократить время на индивидуальную работу с обучаемым персоналом и пр.

Широко известные экспертные системы:

CLIPS – весьма популярная ЭС

OpenCyc – динамическая ЭС с глобальной онтологической БЗ и поддержкой независимых контекстов

WolframAlpha – поисковая система, интеллектуальный «вычислительный движок знаний»

MYCIN –система для диагностики и наблюдения за состоянием больного при менингите и бактериальных инфекциях.

PROSPECTOR – ЭС для выдачи геологам информации о наличии залежей ископаемых; о выборе мест для бурения.

HASP/SIAP –определяет местоположение и типы судов в океане по данным акустических систем слежения.

Акинатор — интернет-игра. Игрок должен загадать любого персонажа, а Акинатор должен его отгадать, задавая вопросы. База знаний автоматически пополняется, поэтому программа может отгадать практически любого известного персонажа.

В общем, различают 6 основных типов интеллектуальных информационных систем (тип определяется целью, ресурсами, характером использования и предметной областью):

Диалоговая система обработки запросов (Transaction Processing System);

Система информационного обеспечения (Information Provision System);

Система поддержки принятия решений (Decision Support System);

Программируемая система принятия решения (Programmed Decision System) для автоматического отбора решений;

Экспертные системы (Expert System);

Проблемно-ориентированные интеллектуальные системы, основанные на знаниях (Knowledge Based System).

Данные – информация – знания

Понятия "знание", "информация", "данные" часто отождествляются. В ИИ их необходимо различать. Взаимосвязь между понятиями «данные», «информация» и «знания» непростая. Для того чтобы уверенно оперировать понятиями "информация", "данные", "знание", необходимо не только понимать суть этих понятий, но и прочувствовать отличия между ними.

Например, понятие Data Mining переводится на русский язык при помощи трех различных понятий: добыча данных, извлечение информации, раскопка знаний! Чтобы разобраться с этими понятиями рассмотрим несколько простых примеров:

1. Студент, который сдает экзамен по СИИ, нуждается в данных.
2. Студент, который сдает экзамен по СИИ, нуждается в информации.
3. Студент, который сдает экзамен по СИИ, нуждается в знаниях.

В первом примере возникает мысль, что студенту нужны данные, например, для вычислений при решении задачи в экзаменационном билете.

Во втором примере, очевидно, речь идет об информации в виде конспекта лекций или учебного пособия по СИИ.

Наиболее логичным выглядит третий вариант: студент при наличии соответствующих данных и информации, а также их использовании студент может рассчитывать в определенных случаях, что полученные данные и информация перейдут в знания.

Даже большие объемы данных и информации не гарантируют получение знаний. Много зависит от качества процедур обработки информации и данных. Информацию в виде текста на иностранном языке нельзя превратить в знание при отсутствии словаря или переводчика.

Данные

В упрощённом представлении данными можно считать полученные факты (сведения, сообщения, сигналы), характеризующие объект. Данные не тождественны информации. Станут ли данные информацией, зависит от того, известен ли метод преобразования данных в известные понятия. Информация – это

не статичный объект. Она динамически меняется и существует только в момент взаимодействия данных и методов их преобразования, т.е. в момент протекания информационного процесса. Все остальное время она содержится в виде данных. Данные объективны, а методы субъективны, в их основе лежат алгоритмы, придуманные людьми (субъектами)

Информация

Прошло более 60 лет с тех пор, как Н. Винер в "Кибернетике" (1948 г.) ввел известное до него понятие информации в общенаучный обиход, а оттуда оно распространилось на все уровни – от повседневности до философии. Примерно в то же время сформировалась и теория информации (К.Э. Шеннон). Определений понятия информации достаточно много (известны более ста определений информации), но столь желаемого однозначного определения, устраивающего всех, так и не найдено.

Информация – это обработанные данные независимо от формы их представления. Информация, в отличие от данных, имеет смысл. Однако трудно найти понятие, более общее для всех наук и более загадочное, чем информация. Существует несколько подходов к измерению информации. Рассмотрим наиболее популярные из них.

Мера Хартли. Пусть имеется система S . Она может находиться в N равновероятных состояниях. Если каждое состояние системы закодировать, например, двоичными кодами определённой длины d , то эту длину необходимо выбрать так, чтобы число всех различных комбинаций было бы не меньше, чем N . Наименьшее число, при котором это возможно или мера разнообразия множества состояний системы задаётся формулой Р. Хартли:

$$I = k \log_a N$$

где k – коэффициент пропорциональности (масштабирования, в зависимости от выбранной единицы измерения), a – основание рассматриваемой системы.

Пример. Чтобы узнать положение точки в системе из двух клеток, т.е. получить некоторую информацию, необходимо задать 1 вопрос ("Левая или правая клетка?"). Узнав положение точки, мы увеличиваем суммарную информацию о системе на 1 бит ($I = \log_2 2$). Если система имеет n различных состояний, то максимальное количество информации равно $I = \log_2 n$.

Если измерение ведётся в экспоненциальной (натуральной) системе, то

$$k = 1, I = \ln N \text{ (нат);}$$

если измерение ведётся в двоичной системе, то

$$k = 1/\ln 2, I = \log_2 N \text{ (бит);}$$

если измерение ведётся в десятичной системе, то

$$k = 1/\ln 10, \quad I = \lg N \text{ (дит).}$$

Утверждение Хартли: если во множестве $X = \{x_1, x_2, \dots, x_n\}$ выделить произвольный элемент $x_i \in X$, то для того, чтобы найти его, необходимо получить не менее $\log_a n$ (единиц) информации.

По Хартли, чтобы мера информации имела практическую ценность - она должна быть такова, чтобы информация была пропорциональна числу выборов.

Пример. Имеются 50 монет, из которых одна фальшивая. Определим сколько взвешиваний нужно произвести, чтобы определить ее. Если положить на весы равное количество монет, то получим 3 возможности: а) левая чашка ниже; б) правая чашка ниже, в) чашки в равновесии. Таким образом, каждое взвешивание дает количество информации $I = \log_2 3$. Следовательно, для определения фальшивой монеты нужно сделать не менее k взвешиваний, где k удовлетворяет условию $\log_2 3^k \geq \log_2 50$. Отсюда, $k \geq 4$ (надо сделать не менее 4 взвешиваний).

Формула Хартли отвлечена от семантических свойств рассматриваемой системы. Это положительная сторона формулы. Но формула не учитывает разную вероятность N состояний системы.

Уменьшение (увеличение) I может свидетельствовать об уменьшении (увеличении) разнообразия состояний N системы.

Мера К. Шеннона. Формула Шеннона дает оценку информации независимо от ее смысла:

$$I = - \sum_{i=1}^N p_i \log_2 p_i,$$

где N - число состояний системы; p_i - вероятность (или относительная частота) перехода системы в i -ое состояние, причем

$$\sum_{i=1}^N p_i = 1.$$

Если все состояния равновероятны (т.е. $p_i = 1/N$), то $I = \log_2 N$, т.е. мера Шеннона совпадает с мерой Хартли.

Формулы Хартли и Шеннона подтверждаются данными нейропсихологии.

Пример. Время t реакции испытуемого на выбор предмета из имеющихся N предметов линейно зависит от $\log_2 N$: $t = 200 + 180 \log_2 N$ (мс). По аналогичному закону изменяется и время передачи информации в живом организме.

Пример. Один из опытов по определению психофизиологических реакций человека состоял в том, что перед испытуемым большое количество раз зажигалась одна из n лампочек, которую он должен указать. Оказалось, что среднее время, необходимое для правильного ответа испытуемого, пропорционально не числу n

лампочек, а именно величине I определяемой по формуле Шеннона, где p_i - вероятность зажечь лампочку номер i .

Легко видеть, что в общем случае:

$$I = -\sum_{i=1}^N p_i \log_2 p_i \leq \log_2 N.$$

Если выбор i -го варианта predetermined заранее (выбора, собственно говоря, нет, $p_i=1$), то $I=0$.

Сообщение о наступлении события с меньшей вероятностью несёт в себе больше информации, чем сообщение о наступлении события с большей вероятностью. Сообщение о наступлении достоверно наступающего события несёт в себе нулевую информацию (и это вполне ясно, - событие всё равно произойдёт когда-либо).

В термодинамике известна формула Больцмана

$$H = -k \sum_{i=1}^N p_i \ln p_i,$$

где k - постоянная Больцмана ($k=1.38 \times 10^{-16}$ эрг/град), которая известна как энтропия или мера хаоса, беспорядка в системе. Сравнивая выражения для I и H видим, что I можно понимать как информационную энтропию или энтропию из-за нехватки информации о системе.

Нулевой энтропии соответствует максимальная информация.

Увеличение (уменьшение) меры Шеннона свидетельствует об уменьшении (увеличении) энтропии (организованности, порядка) системы. При этом энтропия может являться мерой дезорганизации систем от полного хаоса ($H=H^{max}$) и полной информационной неопределённости ($I=I^{min}$) до полного порядка ($H=H^{min}$) и полной информационной определённости ($I=I^{max}$) в системе.

Однако информацию вряд ли можно отождествлять с энтропией: процесс роста энтропии принципиально необратим и относится к окружающей среде, а информация – к объекту или системе.

Понятие информации имеет не только количественный аспект, приводящий к мере, т.е. к числу, но и качественный аспект, не связанный с измерением. Информация – не мера, не число. Измеряемое в теории информации количество информации еще не есть информация, как количество людей не есть человечество, а мощность множества – не множество. Количественные меры объектов – их частные свойства, но не смыслы.

Большие количества материи и энергии управляются энергетически слабыми сигналами, несущими информацию в форме команд и данных. Такую информацию принято считать *функциональной*. Информация, хранимая (переносимая)

информационным полем, представляет собой устойчивое свойство мира – его атрибут, и соответственно может быть названа *атрибутивной* информацией.

Сложность и многообразие информации в живой природе поражает воображение. Казалось бы, что принцип естественного отбора Ч.Дарвина в эволюции, дополненный случайностью мутаций, должен приводить именно к такому многообразию. Однако эпохальное открытие Э.Геккеля (1866), свидетельствует об ином. Зародыш высших живых существ, включая человека, повторяет в своём развитии все этапы эволюции живого в их хронологической последовательности!

Не означает ли это, что существует единая генеральная линия развития? Тогда почему сложность систем неуклонно увеличивается согласно закону роста информации?

Кроме того, существует огромный разрыв между ограниченной информационной ёмкостью гамет (половых клеток) организма и гигантским объёмом информации, необходимым для превращения зародыша во взрослую особь. Откуда берется эта информация?

Возникают также другие вопросы: как достигается фантастическая точность и устойчивость передачи сложной генетической информации от родителей к детям на протяжении тысячелетий? Где находится единая система контроля и штаб управления?...

Чем еще характеризуют информацию? - Скоростью приёма (чем больше шум, тем медленнее на выходе могут воспроизвестись изменения на входе). Избыточный текст скучно читать, а «плотный» текст требует внимания. Другими характеристиками информации являются репрезентативность, полнота, доступность, своевременность, точность, достоверность, безопасность, ценность.

Например, под ценностью информации понимается прагматическое отношение между системой, информацией и целью системы. Ценность информации нужно измерять, но по какой шкале: абсолютной или относительной? Практика свидетельствует: ни одна информация за всю историю природы, жизни и разума не обрела статуса абсолютно ценной. Наоборот, очередной виток развития ставил новые цели и изменял критерии ценности, а то, что казалось прежде абсолютно ценным на все времена, приобретало статус относительной, мифотворческой ценности или вообще низвергалось на свалку истории и науки. Можно ввести абсолютную шкалу ценности: ценнее считать ту информацию, которая генерирует новую информацию с большей вероятностью. Но как оценить эту вероятность? В частности, ценность научных трудов сейчас стало модно оценивать по т.н. "индексу цитирования" (чем чаще цитируют труд, тем он ценнее, тем выше индекс). Но

известны выдающиеся философы и ученые, которые не публиковались или "писали в стол" и только после смерти стали известны миру. Таков Сократ, который по свидетельству учеников (Платона и др.) утверждал, что "письмена мертвы". Таков Г.Кавендиш, занимавшийся наукой вне официальной науки; в его бумагах, найденных после смерти, были обнаружены научные открытия и законы, которые известны науке в другом авторстве (закон Кулона и др.). Таковы основатели многих религиозных учений и школ. Их учения, как правило, передавались изустно, как и "крылатые фразы" известных людей. Не следует фетишизировать индекс цитирования как количественный показатель ценности научной информации. Накануне экзаменационной сессии ценность информации о причудах преподавателя гораздо выше, чем в начале семестра, а о своем преподавателе несравнимо выше, чем о чужом. Изменение цели изменяет и ценность информации, но не наоборот, т.к. ценность информации аксиологически вторична по отношению к цели, преследуемой потребителем информации.

Система в каждый момент своего существования ставит перед собой цели, реализовать которые она может только через информацию, обладающую для этого необходимыми свойствами и являющуюся импульсом целенаправленной деятельности системы. Информация ценна лишь постольку, поскольку она способствует достижению цели. Но в теории информации отсутствуют понятия цели информационного процесса и его безопасности, из которых явно следовало бы, что информация всегда селективируется от дезинформации и шума. Соответственно, метрика ценности, связанная с количеством информации, не обеспечивает такой селекции. В этом-то и проблема данной метрики, часто фатальной для системы – механизм селекции начинает работать, не дожидаясь решения о ценности полученной информации. Например, так случается с общественно-политическими системами в периоды бурь и потрясений. Стоит ли потом удивляться или кусать локти, что сделанный выбор оказался ложным? В целом проблема сводится к оценке априорной ценности информации. Обобщенной, значимой априорной меры ценности информации нет.

Кодирование информации – обязательный подпроцесс любого информационного процесса. На каждом уровне иерархии систем возникает проблема выбора информационного кода. В современных компьютерах используется квазиоптимальный двоичный код, уступающий троичному коду по быстродействию (теоретически оптимален код значности $e \approx 2,718281828459...$). Оптимальный троичный код был реализован всего в двух компьютерах за всю историю вычислительной техники (СССР, Япония). Но троичный кодер оказался настолько ненадежным (в распознавании трех состояний), а троичная арифметика

настолько не развитой, что от идеи троичного компьютера пришлось отказаться до лучших времен.).

Знания

Знания – это обработанная информация, обеспечивающая увеличение вероятности достижения цели, «ноу-хау», технология. Знание - это информация, но не всякая информация - знание. Между информацией и знаниями имеется разрыв. Человек должен творчески перерабатывать информацию, чтобы получить новые знания. Приведем несколько любопытных определений понятия «знание».

«В моем текущем Знании нет ничего абсолютного, и оно ничтожно мало. Я знаю, что ничего не знаю. Чем больше Знание, тем больше и Незнание» (Сократ).

«Знание есть особая форма Незнания. Незнание □ неизменно, объективно, бесконечно, всеобще, изначально. Знание – эклектично, субъективно, ограничено, локально и временно. Только для философии и искусственного интеллекта предметом изучения являются сами Знания» (А.С.Нариньяни).

«Знание – обоснованное истинное убеждение» (В.Н. Вагин).

При изучении некоторого объекта понятия появляются в следующем порядке: данные-информация-знания.

Факты окружающего мира представляют собой данные. Затем при использовании данных в процессе решения конкретных задач появляется информация. Результаты решения задач, проверенная информация, обобщенная в виде законов или теорий представляет собой знания.

В настоящее время делаются попытки создания онтологий, т.е. баз знаний специального типа, формально описываемых тройкой множеств $\langle X, R, F \rangle$, где X – множество понятий в предметной области; R – множество отношений между понятиями; F – множество функций для интерпретации понятий. Онтология в ИИ - это попытка детальной формализации некоторой области знаний с помощью концептуальной схемы. Схема состоит из структуры данных, содержащей все релевантные классы объектов, их связи и правила (теоремы, ограничения), принятые в этой области. Словари, тезаурусы и каталоги являются частными случаями онтологий.

Язык описания онтологий — формальный язык, используемый для кодирования онтологии. Существует несколько подобных языков:

- OWL - ontology web language, язык для семантических утверждений, разработанный как расширение RDF и RDFS;
- KIF (Knowledge Interchange Format или формат обмена знаниями) - основанный на S-выражениях для логики;

- CysL — онтологический язык, использующийся в проекте Сус, основан на исчислении предикатов с некоторыми расширениями более высокого порядка;
- DAML+OIL (FIPA).

Наиболее известными реализованными проектами онтологий являются Сус, WordNet, SUMO.

Сус – проект (запущен в разработку в 1985 г.) по созданию энциклопедической онтологической базы знаний, позволяющей программам решать сложные задачи из области ИИ на основе логического вывода и привлечения здравого смысла. Содержит 200000 термов и в среднем около 10 правил для каждого термина. База знаний разбита на микротеоории для решения частных задач. Типичным примером знаний в базе являются «Всякое дерево является растением» и «Растения смертны». Если спросить «умирают ли деревья?», машина логического вывода может дать правильный ответ. БЗ содержит около 2-х миллионов занесённых туда людьми утверждений, правил и общеупотребительных идей. Они формулируются на языке CysL, который основан на исчислении предикатов и имеет схожий с ЛИСПом синтаксис. Англоязычные пользователи шутят что они «велосипедисты» (от англ. cyclist - велосипедист). Большая часть сегодняшней работы в проекте Сус все ещё связана с инженерией знаний - описание фактов и знаний вручную и реализация логического вывода на основе этих знаний. Однако ведётся работа над тем, чтобы дать системе Сус возможность самостоятельно общаться с пользователями на естественном языке, и над ускорением процесса пополнения базы с помощью машинного обучения.

Например, одна из версий OpenCys_1.0 была выпущена в июле 2006 г. Включает миллионы утверждений. База знаний содержит 47000 концепций и 306000 фактов и её можно просмотреть на сайте OpenCys. База знаний выпущена под лицензией Apache. Интерпретатор CysL (программа, которая позволяет вам просматривать и изменять базу данных и делать выводы) выпущена бесплатно, но только в двоичном виде, без исходных текстов. Она работает как под GNU/Linux так и под Microsoft Windows.

Версия ResearchCys_1.0, предназначенная для исследовательского сообщества. включает значительно больше семантических знаний, написанных на Java для редактирования знаний и создания запросов к базе. Одна из указанных целей - создать полностью свободный и неограниченный семантический словарь для использования в Semantic Web.

WordNet – проект лингвистической направленности, в котором база знаний разбита на классы эквивалентности по синонимии (синсеты), включает 150 000 слов

и 115 000 синсетов, а также содержит лингвистические отношения между словами, позволяющие делать заключения о соотношении смыслов используемых слов.

SUMO – онтологическая система, содержащая онтологию для почти 20 предметных областей. Написана на языке SUO-KIF (расширение логики предикатов 1-го порядка), включает 20 000 термов и 60 000 аксиом, работает для очень разных по структуре языков (английский, китайский, хинди), отображается на базу знаний проекта WordNet.

Отношения между знанием и языком, как и между классами кур и яиц, нетривиальны. Мы напрямую не разговариваем с компьютером на машинном языке, а пользуемся переводчиками-трансляторами языков программирования в двоичный код. В свою очередь, за языками программирования – языки алгоритмов, за которыми языки задач и, наконец, языки понимания решаемых проблем. А переходы между языками обеспечиваются соответствующими "переводчиками" – трансляторами. Музыкант тоже пользуется языком объяснения – нотным станом, но лишь как путеводителем к взаимопониманию с инструментом, испытывая при этом гадамеровские муки нехватки языка.

Природа языка понимания – информационная, и все языковые проблемы и коллизии – чисто информационного свойства.

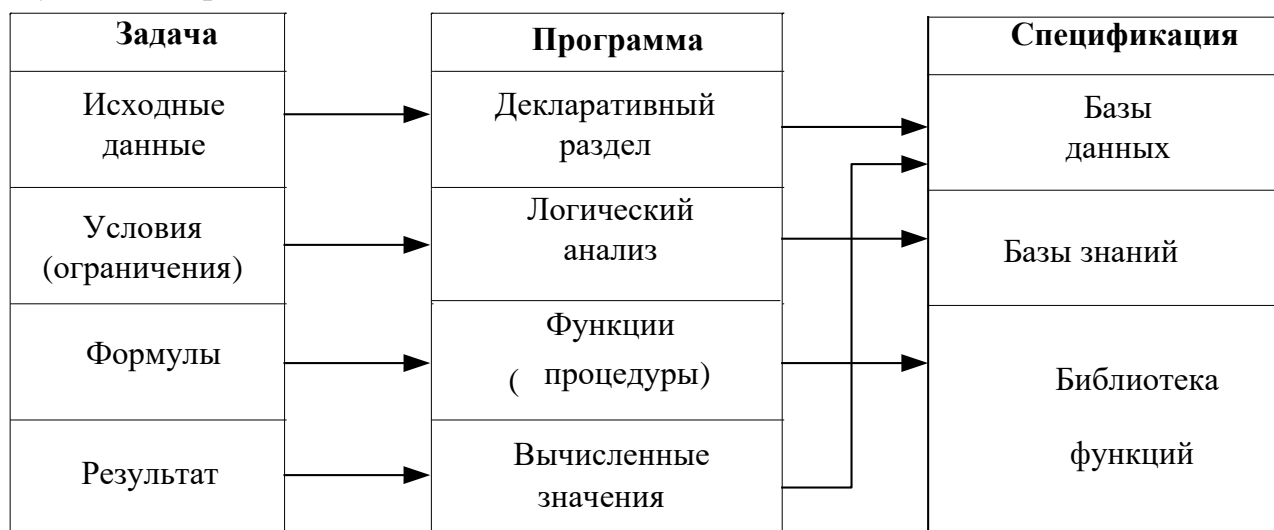
Анализ, построение, развитие и использование моделей предметной области — самая важная и перспективная задача информатики в настоящее время, причем не только в ИИ, но и в других информационных технологиях. Достаточно упомянуть, например, концепцию систем управления бизнес-процессами (*Business Process Management Systems - BPMS*) в области информационно-управляющих систем, концепцию семантической паутины (*Semantic Web*) в Интернете или концепцию проблемно-ориентированных языков (*Domain Specific Languages*) в прикладном программировании. Все эти перспективные концепции апеллируют к понятию онтология.

Лекция «Свойства знаний. Понятия и образы»

Классификация знаний

- по степени общности различают *абстрактные и конкретные* знания (абстрактные применимы в разных предметных областях, конкретные – в одной);
- по способу представления бывают *фактические и концептуальные* знания (фактические задаются совокупностью фактов, концептуальные – обобщающим понятием);
- по форме представления знания делятся на *образные и знаковые* (образная форма реализует отношение целое-часть, знаковая – отношение часть-целое);
- по изменчивости во времени различают *статические и динамические* знания;
- по способу существования знания классифицируют на *хранимые и воспроизводимые*;
- по языку описания бывают *содержательные* (на языке предметной области) и *формальные* (на математическом языке) знания;
- по степени истинности различают *теоретические и эвристические* знания;
- по квалификации источника знания делят на *обыденные и экспертные*;
- по степени соответствия реальности знания бывают *достоверными и правдоподобными*;
- по способу размещения в пространстве различают *локальные и распределённые* знания;
- по способу программирования знания классифицируют на *процедурные и декларативные*.

В информатике разделение данных и знаний объясняется специализацией фрагментов программ за годы эволюции программных систем. Специализация отдельных фрагментов программ в самостоятельные системы происходила следующим образом:



Исходные данные решаемых задач и результаты решения послужили основой для разработки *баз данных*. Условия и правила трансформировались в *базы знаний*, а функции и процедуры - в библиотеки и приложения.

Свойства знаний

1) *Внутренняя интерпретируемость*. Она означает использование *имен* для поиска знаний. Данные в памяти лишены имен и могут идентифицироваться только программой, извлекающей их из памяти. Что скрывается за тем или иным двоичным кодом машинного слова, системе неизвестно. При переходе к знаниям каждая информационная единица должна иметь уникальное имя, по которому ИС находит ее, а также отвечает на запросы, в которых это имя упомянуто. Если, например, в память ЭВМ нужно записать сведения о сотрудниках учреждения, то без внутренней интерпретации в память была бы занесена совокупность из четырех машинных слов, соответствующих строкам этой таблицы:

Фамилия	Год рождения	Специальность	Стаж работы
Попов	1965	Экономист	15
Сидоров	1971	Радиотехник	10
Иванов	1952	Юрист	30
Петров	1977	Программист	5

Информация о том, как закодированы сведения у системы, отсутствует. Ее знает лишь программист. Система не может ответить на вопрос "Что известно о Попове?" или "Есть ли среди специалистов юрист?" При переходе к знаниям в память ЭВМ вводится информация о том, например, в каких разрядах хранятся те или иные сведения. По ним можно осуществлять поиск нужной информации.

2) *Вложимость*. Возможность включения некоторых знаний в состав других (*рекурсия*) и, наоборот, выделение из знаний отдельных информационных единиц. Иными словами, каждая информационная единица может быть включена в состав любой другой, и из каждой информационной единицы можно выделить некоторые составляющие ее информационные единицы.

3) *Структурированность* (связность). Должна существовать возможность установления между отдельными информационными единицами отношений типа «часть – целое», «род – вид», причина-следствие, аргумент-функция и др. Семантика отношений может носить декларативный или процедурный характер.

Отношения "одновременно", "причина - следствие" или "быть рядом" характеризуют декларативные знания. Если между двумя информационными единицами установлено отношение "аргумент - функция", то оно характеризует процедурное знание, связанное с вычислением определенных функций.

Перечисленные три свойства знаний позволяют ввести общую модель представления знаний - семантическую сеть с вершинами, в которых находятся информационные единицы с именами. Дуги – семантические связи между ними.

4) *Семантическая метрика*. Используется для объединения понятий по степени их смысловой близости или *релевантности*. Отношение релевантности позволяет находить знания, близкие к уже найденным. На множестве информационных единиц в некоторых случаях полезно задавать отношение, характеризующее ситуационную близость информационных единиц, т.е. силу ассоциативной связи между информационными единицами.

5) *Активность*. Все вычислительные процессы инициируются командами, а данные используются этими командами лишь в случае необходимости. Иначе говоря, данные пассивны, а команды активны. Активность знаний означает необходимость действий, возможность модификации знаний или связей между понятиями. Действия в БЗ инициализируются не командами, а ее состоянием, например, появлением фактов или установлением связей между событиями в системе понятий предметной области. БЗ (обязательный элемент ИС) отличается от БД возможностью логического вывода.

С момента появления ЭВМ и разделения используемых в ней информационных единиц на данные и команды создалась ситуация, при которой данные пассивны, а команды активны. Все процессы в ЭВМ инициируются командами, а данные используются этими командами лишь в случае необходимости.

Для ИС эта ситуация не приемлема. Выполнение программ в ИС должно инициироваться текущим состоянием информационной базы. Появление в базе фактов, событий или установление связей между информационными единицами и понятиями предметной области может стать источником активности системы.

6) *Корректность*. Корректность включает проверку БЗ на *полноту* (отношение выводимости в БЗ с помощью определенных операций и средств) и *непротиворечивость* (синтаксическое и семантическое совпадение значений одних и тех же знаний и данных, в разных частях БЗ).

Понятия и образы

Знания делят на фрагменты, которые называются *предметными областями*. Знания характеризуются с помощью понятий и образов.

Понятие (лат. *concept*) – это класс сущностей (*denotat*), объединяемых на основе общности признаков.

Сущность – это объект произвольной природы, принадлежащий реальному или виртуальному миру.

Класс сущностей – это совокупность предметов, их свойств, состояний, процессов, событий, оценок событий, их модификаторов, квантификаторов, модальностей. Например, в области программирования предметом может быть программа, одним из её свойств – надёжность, состоянием – неисправность, процессом – тестирование, событием – запуск программы, оценкой события – удовлетворительное (или нет) качество работы, модификатором – быстродействие программы, квантификатором – частота использования, модальностью – возможность использования.

Множество существенных признаков, характеризующих понятие, называется его *содержанием (интенсионалом)*. Содержание понятия A задаётся через свойства принадлежащих ему сущностей: $A = \{A_1, A_2, \dots, A_k\}$, где A_i – свойство понятия или его существенный признак.

Например, понятие **Стул** = {Приспособление для фиксации сидячей позы, Плоская поверхность, Ножки, Спинка}.

Содержание понятия должно отвечать требованиям полноты и избыточности. Содержание понятия зависит от границ предметной области, поэтому для минимизации содержания понятия желательно явно указать предметную область.

Объём понятия (экстенционал) – это сущности, объединяемые в понятие. Сущности, входящие в объём, задаются обычно перечислением. Объём понятия, в отличие от содержания, может изменяться путём задания ограничений (например, число стульев в комнате измеряется единицами, хотя существуют миллионы стульев).

Основными формами представления понятийных знаний являются знаковая и образная формы. Носителем знаковой формы представления знаний является естественный язык (ЕЯ). Для представления понятия в ЕЯ используется слово, называемое именем понятия. Оно семантически отражает сущность какого-то объекта или явления, а синтаксически — конкретный субъект или объект высказывания. Перспективным считается семиотический подход к моделированию знаний. На рисунке показана схема, известная в семиотике как **треугольник Фреге**:



Объекты реального мира (*денотаты*) отражаются в сознании человека (ментальном мире). В результате возникают *представления* о денотатах. Для различения денотатов человек использует *имена* и формирует *понятия*. Связи в треугольнике Фреге соответствуют механизмам мышления. Связь 1 позволяет по имени активизировать в памяти сведения о свойствах понятия. Эта же связь в обратном направлении позволяет по описанию свойств понятия определить его имя (загадки: «Не лает, не кусает, а в дом не пускает?»). Связь 2 позволяет по представлению узнать свойства понятия. Связь 3 соединяет представление о денотате с его именем. Треугольник Фреге позволяет отображать синонимию и омонимию понятий.

Структура треугольника Фреге, называется *знаком* или *семой*.

Знак – минимальный носитель языковой информации. Совокупность знаков образует знаковую (семиотическую) систему, или язык.

Символ в отличие от знака не имеет семантики и прагматики, а только синтаксис. Проблемы семантики, смысла языковых выражений обсуждаются в лингвистической и логико-математической литературе. Ясность в эту область логико-лингвистической проблематики будет вносить практика, в качестве которой выступают работы по машинному переводу.

Имена в предметной области часто называют *терминами*. К терминам предъявляются некоторые требования: однозначность, стилистическая нейтральность, соответствие нормам языка, обладание словообразующей функцией, краткость и пр. Связь между именем и понятием реализуется обычно путём определения понятия.

Определение раскрывает содержание (иногда объём) понятия. Существуют различные способы определения понятий: *остенсивные* (указывают на конкретный предмет, явление, действие), *номинальные* (одно слово заменяет несколько слов), *интенциональные* (раскрываются существенные признаки понятия), *экстенциональные* (перечисляются все объекты, входящие в понятие). К

определениям, как и к терминам, также предъявляются определённые требования: избыточность; соразмерность объёма определяемого понятия и тех понятий, которые включаются в него; отсутствие «порочного круга»; ясность; лингвистическая правильность и краткость.

Люди оперируют осмысленными понятиями в отличие от компьютеров, которые оперируют лишь содержимым ячеек памяти. Мысли выстраиваются из совокупности понятий и образов, связанных между собой. В данном контексте вопрос А. Тьюринга о том, могут ли машины мыслить, попросту неуместен. Машины пока вообще не имеют дела с мыслями.

В процессе мышления одновременно присутствует как понятийная, так и образная логика. Для понятийных знаний разработан достаточно мощный язык математической логики. Язык для оперирования образами в процессе решения интеллектуальных задач пока не создан. Разработки такого языка ведутся в когнитологии на стыке эпистемологии, герменевтики и психолингвистики.

Основные положения когнитологии заключаются в следующем. Мысль идеальна, а формы её существования различны: модель, понятие, образ. Модель – это «сильная» форма мысли. Она отражает отдельные стороны или свойства реальных объектов. Человек постоянно сравнивает модели с объективной реальностью. Понятие представляет собой словесно выраженную мысль о существенных признаках сходства или различия предметов и является более «слабой» формой мысли, нежели модель, отражающая только одно отношение («включает», «содержит», «характеризуется»). Неявно, через распознавание понятие связано не только с моделью, но и с образом. Образ – это трудно формализуемое субъективное представление о предметах реального или виртуального мира. Образы или формируются в нашей памяти, или находятся вне нашего сознания, т.е. представлены на различных языках (естественных, математических, языке музыкальной нотации, химических формул, языке жестов, мимики и т.д.) и в различных формах (текстовой, графической, зрительной, звуковой, вкусовой, осязательной и т.д.). Можно говорить о правополушарных (имплицитных или контекстных) и левополушарных (ментальных) образах в нашей памяти.

В реальном процессе усвоения знаний присутствует единая образная и понятийная логика. В образном мышлении не используются словесные знания в виде умозаключений, а основными операциями являются абстрагирование, конкретизация, классификация и моделирование. Моделируется не фактическая сущность, а представление о ней, т.е. её образ. Самым мощным средством моделирования мыслей является естественный язык. Языки у людей разные, а мышление, в принципе, одно. Математические языки и логики являются наиболее

формальными средствами моделирования. К ним относятся также модели представления знаний (фреймы, продукционные правила, семантические сети и др.).

Спам не случайно стал объектом лингвистического анализа. Разработчики антиспамерского софта вынуждены сейчас искать способы отождествления искажённого и "нормального" текстов. Причина состоит в том, что намеренные искажения текста, позволяющие эффективно обойти почтовые фильтры, практически не влияют на его восприятие за счёт образного характера мышления человека. Намеренное искажение может существовать как на уровне слова, так и на уровне текста в целом.

Лекция «Классификация понятий. Модели знаний»

Виды классификации понятий

Существуют различные виды классификации понятийных знаний: родо-видовая, фасетная, иерархическая, партитивная и др.

Связь «род-вид» соответствует отношению «общее-частное», в котором одно понятие является исходным (родовым), а расширяющий родовое понятие видовой признак называется видовым отличием. В силу родства некоторых понятий для оценки их семантической близости обычно используют ранг r (степень родства). Для родового понятия ранг $r = 0$, а для образования любого видового понятия применяется рекурсивная формула вида $A_{r+1, B} = A_r + B$, где A_r – исходное понятие, B – видовое отличие. Данная формула отражает наследование признаков. Объём видового понятия $A_{r+1, B}$ определяется по формуле $A_{r+1, B} = A_r \& B$.

Например



В этой схеме понятие «контроль знаний по СИИ» является минимальным по содержанию, а «промежуточный рейтинг-контроль знаний» – максимальным.

Объём видового понятия *уже*, чем объём родового понятия, в то время как его содержание *шире*. В объём видового понятия обычно включаются элементы и их классы. Связь «элемент-класс» является частным случаем связи «род-вид». В английской терминологии отношение «элемент-класс» обозначают «*is a*». Например, Petrov *is a* student. Если Белов, Орлов и Петров учатся в ТТИ ЮФУ, то их можно *обобщить* в класс «Студенты». *Конкретизацией* класса «Студенты» является подкласс «Успевающие студенты», например Орлов и Петров. Обобщение индуктивно в отличие от конкретизации, которая дедуктивна.

Элементы, входящие в объём понятия, наследуют различные видовые отличия по признаку *основания деления*. Понятие может многократно делиться. Деление состоит в том, что входящие в некоторый объём понятия распределяются по классам (фасетам, таксонам). Доказано, что каждое понятие, порожаемое на основе других понятий, имеет непустой объём.

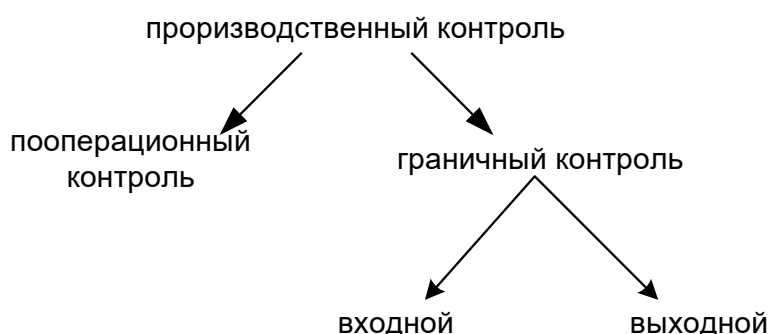
В зависимости от способа деления различают *фасетную* (параллельную) и *иерархическую* (последовательную) классификации понятий.

Пример. Пусть фасетная классификация понятия «техническое диагностирование» включает 3 фасета по 2 видовых отличия:

Основание деления	Видовые отличия (в скобках) и понятия
1. Глубина диагностирования	(Равна 1) Контроль (исправен\неисправен) (Больше 1) Поиск неисправности
2. Режим диагностирования	(Тестовый) Тестовое диагностирование (Рабочий) Рабочее диагностирование
3. Анализируемые признаки	(Функция объекта) Функциональное диагностирование (Параметр объекта) Параметрическое диагностирование

Параметром иерархической классификации является ранг понятия, а параметром фасета служит число содержащихся в фасете видовых понятий. Отличительной особенностью иерархической классификации является преемственность между основаниями деления соседних уровней, когда каждое последующее деление детализирует предыдущее. Наибольшее распространение получила дихотомическая иерархическая классификация (два основания деления на каждом уровне). Доказано, что любая недихотомическая классификация может быть сведена к дихотомической.

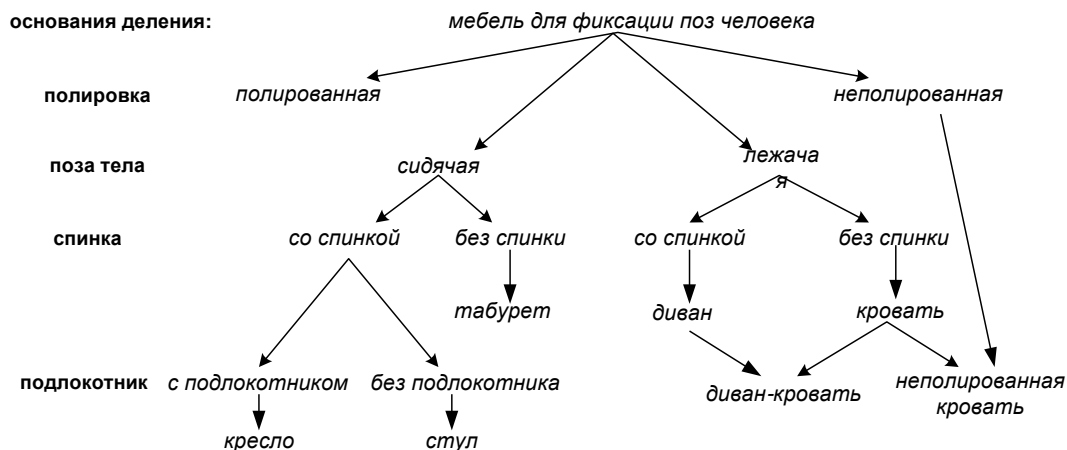
Пример дихотомической классификации:



Реальные классификации понятий, как правило, являются *межвидовыми* или *смешанными*. В первом случае основания деления порождающих понятий различны, во втором – основания деления общие.

Например, классификации понятия «Мебель»:

УЧЕБНОЕ ПОСОБИЕ «ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ»



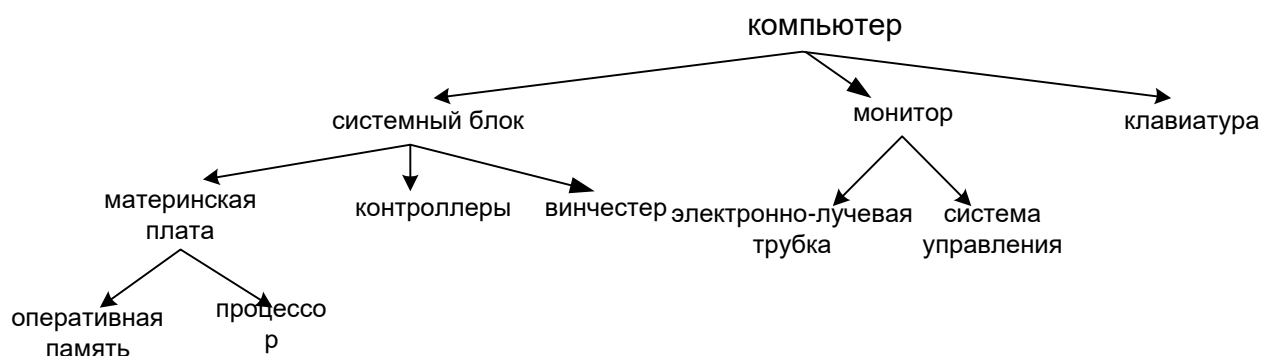
Здесь четыре основания деления, смешанное понятие (диван-кровать), межвидовое понятие (неполированная кровать)

Правила деления понятий:

1. Деление должно производиться только по одному основанию.
2. Элементы деления находиться в отношении подчинения.
3. Элементы одного уровня должны быть одной общности.
4. Деление должно быть соразмерным, т.е. сложение объёмов элементов деления должно дать в точности объём делимого понятия.

Например, в утверждении «Планеты Солнечной системы делятся на имеющие всего один спутник и на имеющие больше одного естественного спутника» нарушено 4-е правило. В солнечной системе существуют планеты, не имеющие вообще естественных спутников.

Партитивная связь понятий – это связь между целым и его частями, обозначается как «*part of*» (Monitor is *part of* a PC). Партитивная связь характеризует разделение целого понятия на отдельные части, например



В чём различие между партитивным и родовидовым понятием? Различие заключается в следующем: понятия-части не являются однородными; свойства понятия-целого не распространяются на свойства понятия-части.

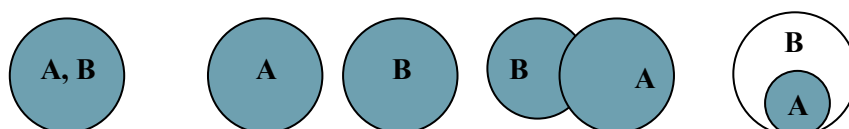
Собирательная связь объединяет понятия таких объектов, которые существуют самостоятельно, но могут быть объединены в целое (агрегат), причём свойства понятия «агрегат» могут быть эмерджентными (вновь появившимися), т.е. не

присущими отдельным его частям. Собирабельная связь может быть применена как к однородным объектам (человек–народ), так и к неоднородным объектам (человеко-машинная система).

Классификация понятий и отношений позволяет разрабатывать компьютерные модели и технологии понятийного мышления. Каждой сущности в реальном мире соответствует понятие, представление о ней и имя (треугольник Фреге). Попытки представления системы понятий в предметной области привели к возникновению БЗ специального типа в виде *онтологий*: $\langle X, R, F \rangle$, где X – множество понятий, R – множество отношений на X , F – множество функций.

Ранее отмечалось, что понятия определяются через *содержание* и *объём*. Причём, чем *шире* набор свойств, составляющих содержание понятия, тем *уже* класс объектов с этими свойствами. Наоборот, чем *уже* содержание понятия, тем *шире* его объём.

Понятия A и B , выраженные через их содержание, могут находиться в отношениях *равнозначности*, *несравнимости*, *толерантности*, *подчиненности* и др:



Как оценить сходство/различие понятий?

Возникает вопрос: как оценить сходство или различие двух понятий? Обычно, для оценки различия между понятиями A, B вводится некоторая *метрика* для оценки «расстояния» между ними:

$$d(A, B) = (\sum_i r_i * p_i(A, B)) / \sum_i r_i,$$

где $p_i(A, B)$ – указатель *различия* i -х свойств понятий A, B ; r_i – «вес» i -го свойства, равный $n, n-1, \dots, 2, 1$, причём $p_i(A, B) = 1$, если $A_i \neq B_i$, $p_i(A, B) = 0$, если $A_i = B_i$.

Пример 1. В предметной области «Мебель» пять понятий (*стол, табурет, стул, кресло, кровать*) представлены четырьмя признаками (*поза тела, спинка, подлокотник, цвет*). Введем метрику для признаков. Для признака «поза тела» зададим значения в нелинейной шкале, соответственно 0 для сидячей позы тела, 0,8 – для сидячей за столом и 1 – для лежащей позы. Для признаков «спинка» и «подлокотник» примем булеву шкалу $[0, 1]$. Для признака «цвет» примем шкалу из 7 отрезков длиной 0,16 на интервале $[0, 1]$. Наиболее значимым признаком будем считать признак «поза тела» ($r_1 = 4$), наименее значимым – «цвет» ($r_4 = 1$). Получим следующую таблицу:

Признак	Поза тела ($r_1 = 4$)	Спинка ($r_2 = 3$)	Подлокотник ($r_3 = 2$)	Цвет ($r_4 = 1$)
Понятие				
Стол	Сидячая (за столом)	0	0	Жёлт.
Табурет	Сидячая	0	0	Жёлт.
Стул	Сидячая	1	0	Жёлт.
Кресло	Сидячая	1	1	Жёлт.
Кровать	Лежачая	0	0	Оранжев.

Для оценки сходства/различия между парами понятий воспользуемся метрикой, представив результаты расчётов в таблице:

Признак	Поза	Спинка	Подлокотник	Цвет	Различие (d)	Сходство ($1-d$)
Пара понятий						
Стол - табурет	0,8	0	0	0	0,32	0,68
Табурет - стул	0	1	0	0	0,3	0,7
Стул - кресло	0	0	1	0	0,2	0,8
Стол - кровать	0,2	0	0	0,166	0,0966	0,9034

Таким образом, при выбранных признаках и шкалах для четырёх указанных в таблице пар понятий минимальное семантическое различие (максимальное сходство) будет между *столом* и *кроватью*, наибольшее различие – между *столом* и *табуретом*. На результат также влияет «вес» признака.

Пример 2. Зачастую при распознавании объектов признаки трудно выразить численно, поэтому важно просто их наличие или отсутствие у объекта. Пусть даны три объекта: *вишня*, *яблоко* и *банан*. Они характеризуются признаками: *цвет* (*жёлтый* и/или *красный*), *семечки*, *косточки*. Если некоторый признак j присутствует в i -м объекте, то переменная $X_{ij} = 1$, иначе $X_{ij} = 0$. Составим таблицу для трёх указанных объектов, зафиксировав в ней значения признаков:

Признак	Жёлтый	Красный	Семечки	Косточки
Объект				
Вишня	$X_{11} = 0$	$X_{12} = 1$	$X_{13} = 0$	$X_{14} = 1$
Яблоко	$X_{21} = 1$	$X_{22} = 1$	$X_{23} = 1$	$X_{24} = 0$
Банан	$X_{31} = 1$	$X_{32} = 0$	$X_{33} = 0$	$X_{34} = 0$

Оценим меру сходства/различия между некоторыми парами понятий, представленных в таблице. При сравнении пары объектов (i, k) по одному из признаков j возможны четыре варианта (X_{ij}, X_{kj}) : $(1,1)$, $(1,0)$, $(0,1)$, $(0,0)$. При сравнении пары объектов (i, k) по всем признакам j необходимо вычислить:

– число случаев, когда оба объекта обладают одними и теми же признаками $a = \sum_j X_{ij} * X_{kj}$;

– число случаев, когда оба объекта не обладают одними и теми же признаками $b = \sum_j (1-X_{ij}) * (1-X_{kj})$;

– число случаев, когда i -й объект не обладает признаками k -го объекта $h = \sum_j (1-X_{ij}) * X_{kj}$;

– число случаев, когда i -й объект обладает признаками, отсутствующими у k -го объекта $g = \sum_j X_{ij} * (1-X_{kj})$.

Операции в формулах для a , b , h и g представляют собой обычные арифметические операции над булевыми переменными. Из анализа формул для a , b , h , g следует, что чем больше сходства между парой объектов, тем большее влияние имеет переменная a . На практике используются различные функции сходства S :

– Рассела и Рао $S_1 = a / (a + b + h + g) = a / n$;

– Жокара и Нидмена $S_2 = a / (n - b)$;

– Дайса $S_3 = a / (2a + h + g)$;

– Сокаля и Снифа $S_4 = a / (a + 2(h + g))$;

– Кульжинского $S_5 = a / (h + g)$;

– Юла $S_6 = (ab - gh) / (ab + gh)$ и др.

Разнообразие формул означает, что среди них нет универсальной для различных задач. Кроме того, необходимо учитывать следующий эффект: два объекта, обладающих большим числом признаков, представляются более похожими, нежели два объекта, обладающих меньшим числом признаков. Поэтому следует вносить поправку в параметр a . Заметим также, что в случае идентичных объектов ($b = n - a$, $g = h = 0$) функция S_5 стремится к бесконечности, $S_2 = S_4 = S_6 = 1$, а $S_3 = 0,5$. Функция S_5 наиболее чувствительна к сходству двух объектов.

Найдем сходство между вишней (X_1) и яблоком (X_2), используя функции S_2 и S_5 . Для пары (вишня, яблоко) подсчитываем по исходной таблице значения: $a = 1$, $b = 0$, $g = 1$, $h = 2$. Тогда $S_2(X_1, X_2) = 0,25$, $S_5(X_1, X_2) = 0,333$.

Теперь сравним пару яблоко (X_2) и банан (X_3): $a = 1$, $h = 0$, $b = 1$, $g = 2$. Для этой пары значения соответствующих функций будут равны:

$$S_2(X_2, X_3) = 0,333, S_5(X_2, X_3) = 0,5.$$

Сравнение показывает, что при данном наборе признаков яблоко и банан имеют больше сходства, нежели яблоко и вишня.

Модели представления знаний

Наиболее общей моделью представления знаний является *алгебраическая система Мальцева*:

$\langle A, C, F, P \rangle$,

где A – множество переменных; C – множество констант; F – множество функций; P – множество предикатов (отношений).

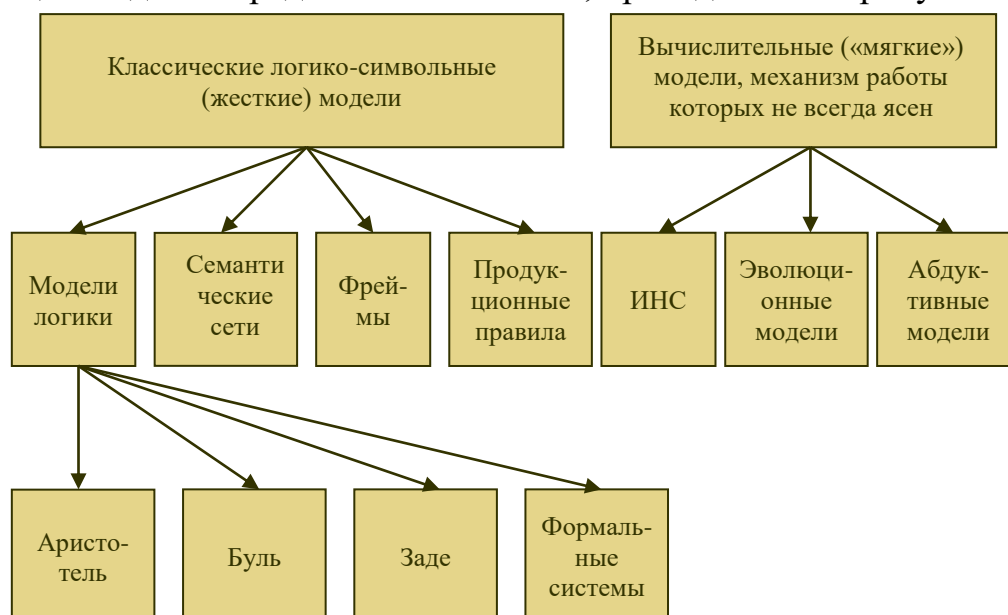
Множество $\Sigma = \langle F, P \rangle$ называется *сигнатурой* или *языком системы*.

Алгебраическая система состоит из двух частей: *алгебры* $\langle A, C, F \rangle$ и *реляционной системы* $\langle A, C, P \rangle$. Алгебре в ИИ соответствует *процедурная модель*, реляционной системе – *декларативная модель*.

База знаний – основа любой интеллектуальной системы, где знания описаны на некотором языке представления знаний.

Совокупность знаний принято называть предметной областью. В любой предметной области есть свои понятия и связи между ними, своя терминология, свои законы, процессы и события. Каждая предметная область имеет свои методы решения задач.

Базы знаний базируются на моделях представления знаний, подобно базам данных, которые основаны на моделях представления данных (иерархической, сетевой, реляционной, постреляционной и т.д.). При представлении знаний в памяти интеллектуальной системы традиционные языки являются неэффективными. Для этого используются специальные языки представления знаний, основанные на символьном представлении данных. Наиболее часто используется на практике классификация моделей представления знаний, приведенная на рисунке:



Сюда входят как классические (символьные) модели, имитирующие мышление и структуру памяти человека, так и модели нейросетей, эволюционные модели, модели немонотонной логики.

К моделям знаний выдвигаются целый ряд требований:

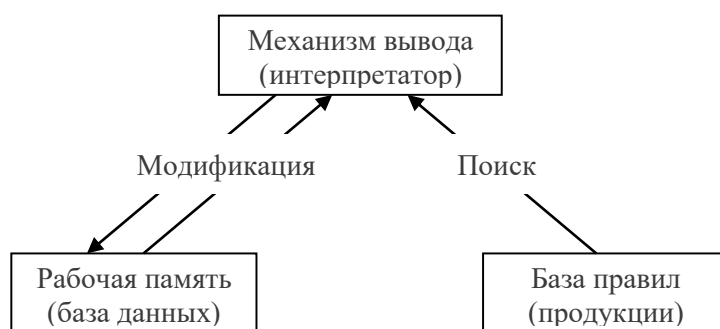
- универсальность;
- наглядность представления знаний;
- реализация в модели свойства активности знаний;
- открытость;
- возможность отражения структурных отношений объектов предметной области;
- возможность оперирования нечеткими знаниями;
- использование многоуровневых представлений (данные, модели, метамодели, метаметамодели и т.д.).

Существующие модели представления знаний не удовлетворяют полностью этим требованиям, чем и объясняется их многообразие и активное развитие данного направления.

К декларативным моделям относятся семантические сети, продукционные правила, фреймы.

Продукционная модель знаний

Архитектура продукционных систем имеет следующий вид:



В рабочей памяти хранятся исходные данные о решаемой задаче и результаты работы системы. База правил включает правила типа

«ЕСЛИ (условия), ТО (действия)».

Механизм вывода использует правила в соответствии с содержимым рабочей памяти. Если текущее содержимое базы данных удовлетворяет условию некоторого правила, то может выполняться действие, указанное в этом правиле. Но возможность применить правило еще не означает необходимость его применять. Будет ли применимое правило действительно применено, определяет механизм вывода. Содержимое рабочей памяти может соответствовать подмножеству правил,

которые образуют конфликтное множество. В соответствии с некоторой стратегией управления выводом для применения выбирается одно из правил конфликтного множества. После применения правила база данных меняет свое состояние. Между правилами нет прямой связи по управлению. Иными словами, правила не вызывают друг друга. Между правилами нет и прямой связи по данным. Правила ничего не сообщают друг другу, все данные находятся в базе данных. Правила являются функционально и информационно прочными модулями.

Преимущества продукционных правил: простота создания и понимания отдельных правил; простота пополнения, модификации и аннулирования; простота механизма логического вывода, наличие программных средств (языки Prolog, LISP, CLIPS, ЭС).

Недостатки ПС: неясность взаимных отношений правил; сложность оценки целостного образа знаний; невысокая эффективность обработки; отсутствие гибкости в логическом выводе; сложность проверки правил на непротиворечивость, особенно если правил больше 1000, неоднозначность выбора правил.

Таким образом, если объектом является небольшая задача, выявляются только сильные стороны системы продукций. В случаях увеличения объема знаний, необходимости решения сложных задач, выполнения гибких выводов или повышения скорости вывода требуется структурирование базы данных.

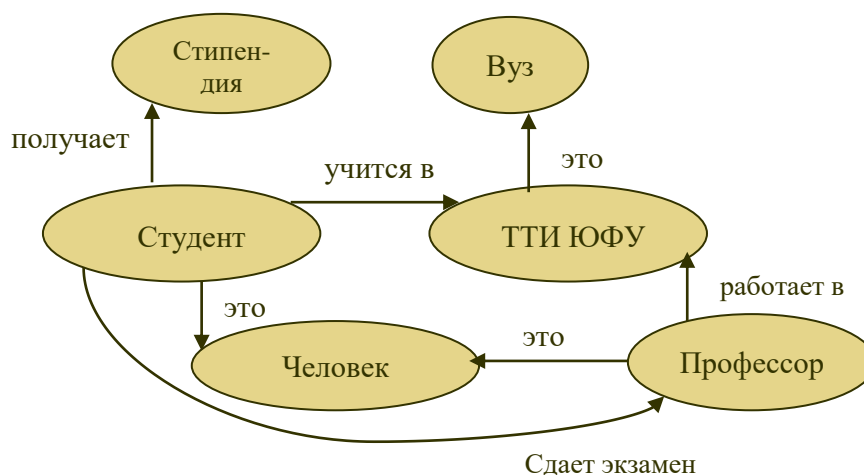
Кроме того, в конкретной ситуации и факты верны не абсолютно, и применение того или иного правила оправдано лишь частично. В системах продукций для этого используют термины «коэффициент достоверности», «обоснованность», «индекс уверенности» и т.д. В последние годы разработчики ищут более формальный базис для работы с неоднородными продукциями. Известны попытки использовать нечеткие логики, дескриптивные логики и другие теории. Однако эти попытки полностью дезавуируются тем обстоятельством, что оценки достоверности исходных фактов и правил в конкретизированной предметной области сами не являются достоверными - их выставляют эксперты также свойственно ошибаться. Какой бы не была продуманной и сбалансированной реализация системы с элементами ИИ и на какой бы теоретический базис она не опиралась, но, если исходный материал не вполне достоверен, то найденное решение также будет носить характер возможного, но не абсолютного!

Резюмируя отметим, что системы продукций — это исторически первый и до сих пор часто применяемый способ представления знаний; продукционные правила эффективно программно реализуемы и являются основой многих практических программных систем; использование систем продукций может принести пользу только при полном понимании всех сопутствующих им ограничений и недостатков.

Семантическая сеть

Одной из структурных моделей долговременной памяти является предложенная Куиллианом модель понимания смысла слов, получившая название TLC- модели (Teachable Language Comprehender: доступный механизм понимания языка). В данной модели для описания структуры долговременной памяти была использована сетевая структура как способ представления семантических отношений между концептами (словами). Данная модель имитирует естественное понимание и использования языка человеком. Поэтому основной ее идеей было описание значений класса, к которому принадлежит объект, его прототипа и установление связи со словами, отображающими свойства объекта.

Идея семантической сети (СС) – любое знание можно представить совокупностью понятий и отношений между ними. СС – это ориентированный граф, вершины которого понятия, а ребра – отношения между понятиями:



Сеть наглядна, имеет аналогию с организацией долговременной памяти человека, но вывод знаний по ней сложен.

Важность модели семантической сети с точки зрения многочисленных приложений определяется следующими моментами. В отличие от традиционных методов семантической обработки с анализом структуры предложения были предложены новые парадигмы в качестве модели представления структуры долговременной памяти, в которой придается значение объему языковой активности. Был предложен способ описания структуры отношений между фактами и понятиями с помощью средства, называемого семантической сетью, отличающейся несложным представлением понятий, а также способ семантической обработки в мире понятий на основе смысловой связи (смыслового обмена) между прототипами.

Была создана реальная система TLC, осуществлено моделирование человеческой памяти и разработана технологическая сторона концепции понимания смысла.

Достоинства семантической сети: описание объектов и событий производится на уровне очень близком к естественному языку; обеспечивается возможность соединения различных фрагментов сети; отношения между понятиями и событиями образуют небольшое, хорошо организованное множество; для каждой операции над данными или знаниями можно выделить некоторый участок сети, который охватывает необходимые в данном запросе характеристики; обеспечивается наглядность системы знаний, представленной графически; близость структуры сети, представляющей знания, семантической структуре фраз на естественном языке; соответствие сети современным представлениям об организации долговременной памяти человека.

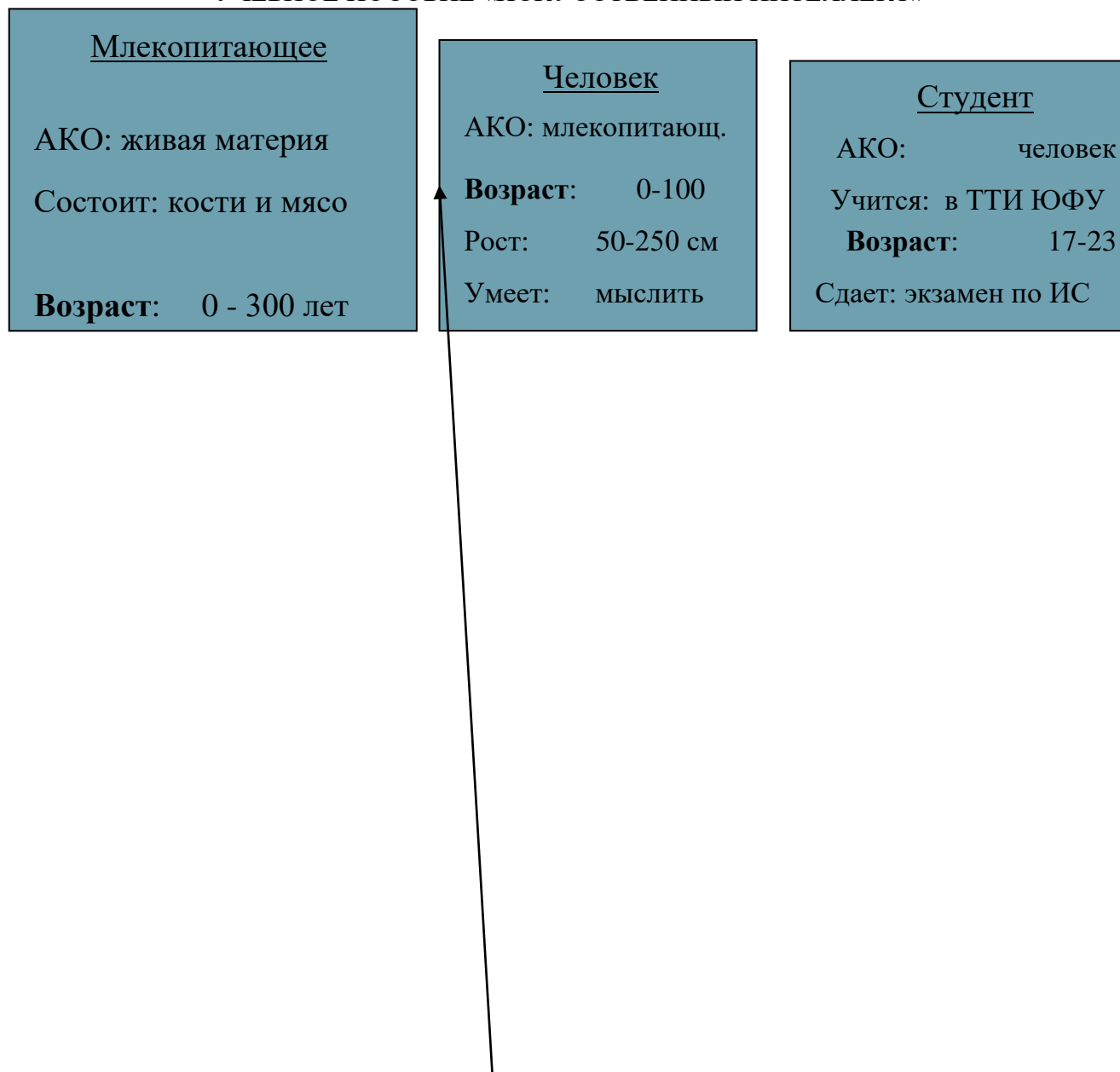
Недостатки семантической сети: сетевая модель не дает ясного представления о структуре предметной области, поэтому формирование и модификация такой модели затруднительны; сетевые модели представляют собой пассивные структуры, для обработки которых необходим специальный аппарат формального вывода и планирования.

Семантические сети нашли применение в основном в системах обработки естественного языка, частично в вопросно-ответных системах, а также в системах искусственного зрения. В последних семантические сети используются для хранения знаний о структуре, форме и свойствах физических объектов. В области обработки естественного языка с помощью семантических сетей представляют семантические знания, знания о мире, эпизодические знания (т.е. знания о пространственно-временных событиях и состояниях).

Фреймовая модель

Фрейм – это структура данных с именем, служащая для представления стереотипных образов, в которых сконцентрированы знания об объекте. Фрейм состоит из слотов и значений атрибутов. Процедуры, подключаемые к фрейму для выполнения запроса называют демонами.

Фреймы имеют иерархическую структуру. Их свойства наследуются сверху вниз через АКО-связи (A Kind Of):



Преимущества фреймов: наглядность, соответствие модели памяти человека, нет ограничений по размерам, поддерживается программно. Достоинство фрейма – представления во многом основываются на включении в него предположений и ожиданий. Это достигается за счет присвоения по умолчанию слотам фрейма стандартных ситуаций. В процессе поиска решений эти значения могут быть заменены более достоверными. Некоторые переменные выделены таким образом, что об их значениях система должна спросить пользователя. Часть переменных определяется посредством встроенных процедур, называемых внутренними. По мере присвоения переменным определенных значений осуществляется вызов других процедур. Этот тип представления комбинирует декларативные и процедурные знания.

Фреймовые модели обеспечивают требования структурированности и связанности. Это достигается за счет свойств наследования и вложенности, которыми обладают фреймы, т.е. в качестве слотов может выступать система имен слотов более низкого уровня, а также слоты могут быть использованы как вызовы каких-либо процедур для выполнения.

Недостаток: трудность обмена большими объемами данных.

Для многих предметных областей фреймовые модели являются основным способом формализации знаний.

К процедурным моделям знаний относятся методы и алгоритмы. Популярными также являются функциональные (поведенческие) модели и функционально-временные модели.

Функциональные модели задаются следующим образом: $\Phi = \langle A, C, F \rangle$, где $A = X \cup Y$, причём X – входы, Y – выходы, F – некоторая функция преобразования множества X в Y . Модель описывает примитивный интеллект на уровне условного рефлекса (комбинационный автомат в схемотехнике, блок условий в программе).

Функционально-временные модели имеют следующий вид: $\Phi В = \langle A, C, F \rangle$, где $A = X \cup Y \cup Q$, где Q – состояния объекта; $F = F_{\text{вых}} \cup F_{\text{пер}}$ – функции выходов и переходов. Эта модель обладает памятью (например, автоматы Мили и Мура).

Известны также гибридные декларативно-процедурные модели, к которым относится, например, *сеть Петри* $\langle T, P, I, O \rangle$, где T – множество вершин переходов; P – множество вершин позиций; I – множество функций входов; O – множество функций выходов. С их помощью моделируются многие трудные проблемы параллельного программирования и систем реального времени, а также задачи сетевого планирования и управления, синтеза химических соединений и т.п.

Представление знаний – весьма сложный и творческий процесс. Слушателям, желающим продвинуться в изучении СИИ для решения практических задач, следует ознакомиться с моделями представления знаний. Наиболее ценными в ближайшие годы будут гибридные модели знаний. Однако их глубокое понимание и эффективное применение невозможно без изучения, сегодня кажущихся уже довольно примитивными, нейросетевых, а также декларативных и процедурных моделей.

Лекция «Достоверный логический вывод. Метод резолюций»

Вывод знаний как задача поиска по дереву решений

Логический вывод, как метод доказательства путём рассуждений, является одним из ключевых вопросов ИИ. Например, в экспертных системах вывод знаний – это новое заключение, полученное по определённым правилам из фактов, имеющихся в базе знаний.

Различают *правдоподобный* и *достоверный* механизмы вывода. К правдоподобию выводу относят *абдуктивный* (от частного к частному), *индуктивный* (от частного к общему), *нечёткий* (основанный на логике Заде), *нейросетевой* и некоторые другие методы вывода. Единственным достоверным методом вывода является *дедуктивный логический вывод* (рассуждения от общего к частному на основе аксиом).

Вывод знаний удобно интерпретировать как задачу поиска в пространстве решений.

Если поиск является целенаправленным, а не случайным, то используется модель графа в виде И/ИЛИ-дерева. Задаётся множество начальных вершин графа, с которых поиск может начинаться, и множество конечных (целевых) вершин, при достижении которых поиск прекращается. И/ИЛИ-граф обладает следующими свойствами: при движении по входным дугам к некоторой вершине реализуется либо конъюнкция, либо дизъюнкция.

Методы поиска по дереву различаются по способу обхода путей на графе: поиск *в глубину* или *в ширину* (относительно порядка обхода вершин), *прямой* (от корня к висячей вершине) или *обратный* поиск в глубину; поиск *без возврата* или *с возвратом*; *безусловный* или *условный* поиск (следующий ход зависит от предыдущего); *полный* или *сокращённый* перебор ходов по дереву; поиск без прогнозной оценки (метод проб и ошибок) или поиск с прогнозной оценкой (метод ветвей и границ, симплекс-метод и др.).

Для реальных задач с неполиномиальной оценкой сложности (*NP-задачи*) дерево поиска может стать вычислительно необозримым, экспоненциально растущим при линейном увеличении размерности задачи (комбинаторный взрыв). Сокращение размерности достигается либо разбиением задачи на фрагменты, либо использованием методов эвристического программирования.

Классификация задач поиска по дереву представлена в следующей таблице:

Число шагов	Число вариантов одного шага	Тип задачи поиска
1	1	Задача о назначении

Число шагов	Число вариантов одного шага	Тип задачи поиска
1	Больше 1	Одноходовая альтернативная задача
Больше 1	Больше 1, известны целевые состояния	Задача распознавания
Больше 1	Больше 1, неизвестны целевые состояния	Задачи оптимизации, игровые задачи
Больше 1	Больше 1, одно целевое состояние	Игры-головоломки, задачи верификации, изоморфизма

С позиции теории информации каждый ход при поиске решения должен приводить к уменьшению энтропии в системе от максимального значения до нуля. Согласно Шеннону, энтропия (неопределённость) – это вероятность $p(m_i)$ получения определённого сообщения из множества $M=\{m_1, \dots, m_n\}$. Объём информации, содержащейся в этом сообщении, будет равен

$$I(m_i) = -\log p(m_i).$$

Здесь объём информации в сообщении и вероятность получения этого сообщения связаны обратной монотонной зависимостью. Энтропия множества сообщений M является суммой вида

$$U(M) = - \sum_i p(m_i) \log p(m_i), i=1, \dots, n.$$

Чем более неожиданным является сообщение, тем оно информативнее. Отсюда следует эвристика выбора при построении вершин дерева поиска: выбирается то состояние (правила, действие), которое сулит наибольший прирост информации.

При достоверном логическом выводе используется такое понятие из математической логики как общезначимость (формула является *общезначимой*, если она истинна при любых значениях входящих в неё переменных). Пусть, например, необходимо проверить общезначимость формулы

$$F = ((A+B) \rightarrow C) \rightarrow (A \rightarrow (B \rightarrow C)),$$

где A, B, C – булевы переменные (1 или 0); операция импликация (\rightarrow) является ложной тогда и только тогда, когда посылка истинна, а заключение ложно.

Проверим общезначимость формулы несколькими методами.

В алгоритме Квайна для проверки используется семантическое дерево из 2^k висячих вершин (k – число переменных). Надо пройти все маршруты от корня до этих вершин.

Например, пусть $A = 1$. Тогда $F = ((1+B) \rightarrow C) \rightarrow (1 \rightarrow (B \rightarrow C)) = C \rightarrow (B \rightarrow C)$.

Если $B = 1$, то $F = C \rightarrow (1 \rightarrow C) = C \rightarrow C = 1$. Если $B = 0$, то $F = C \rightarrow (0 \rightarrow C) = C \rightarrow 1 = 1$.

Пусть теперь $A = 0$. Тогда $F = ((0+B) \rightarrow C) \rightarrow (0 \rightarrow (B \rightarrow C)) = (B \rightarrow C) \rightarrow 1$. Здесь, если $B = 1$, то $F = 1$; если $B = 0$, то $F = 1$. Следовательно, $F \equiv 1$.

Алгоритм редукции методом от противного использует свойства операции импликации. Предположим, что формула F ложна на некотором наборе A, B, C . Тогда $f = ((A+B) \rightarrow C) = 1$, $g = (A \rightarrow (B \rightarrow C)) = 0$, откуда следует, что $A = 1$, $B = 1$, $C = 0$, однако $f = ((1+1) \rightarrow 0) = 0$. Получили противоречие. Следовательно, F является общезначимой формулой.

Наконец, если переменных в проверяемой на общезначимость формуле немного, то доказательство можно провести с использованием булевых таблиц истинности.

Ньюэлл и Саймон обосновали две гипотезы, на которых базируется большинство исследований в области ИИ. В соответствии с этими гипотезами человеческое мышление представляет собой систему, оперирующую с символами, отображающими действительность.

Гипотеза 1. Необходимым и достаточным условием для осуществления интеллектуальных действий в символьных системах является универсальность формальных манипуляций над конкретными символами (достаточно прочесть строку символов, разделить её на компоненты и переупорядочить, добавив или удалив какие-то символы без учёта семантики символов).

Гипотеза 2. Символьные системы решают задачи при помощи поиска, т.е. они генерируют потенциальные решения и модифицируют их до тех пор, пока решения не будут удовлетворять заданным условиям поиска. Поиск - это движение по дереву решений от одних узлов этого пространства к другим.

Для описания символьных систем используются языки логики или исчисления. *Исчисление* – это знаковая система, состоящая из алфавита, аксиом и процедур вывода истинных (синтаксически правильных) формул из аксиом. Если символам языка приписать конкретные значения, то это формальный язык. Различают исчисление высказываний и исчисление предикатов.

В *исчислении высказываний* вопрос об истинности или ложности высказываний решается с помощью формул, которым подчиняются логические операции конъюнкции, дизъюнкции, отрицания, импликации и др. В естественном языке высказыванием может быть повествовательное предложение, о котором можно сказать, истинно оно или ложно.

В *исчислении предикатов* наряду с формулами исчисления высказываний, используются формулы, в которые могут входить отношения (предикаты), связывающие между собой группы элементов исчисления и *кванторы общности и существования*. Логика предикатов в отличие от логики высказываний позволяет

количественно охарактеризовать связи между понятиями, их свойства и отношения. Язык логики предикатов — один из формальных языков, наиболее приближенных к человеческому языку. Язык искусственного интеллекта Пролог основан на логике предикатов 1-го порядка (под знаком квантора не могут находиться символы предикатов).

В основе языка Пролог лежит вывод по методу резолюций, который разработали Робинсон и Маслов. Рассмотрим вначале метод резолюций в исчислении высказываний.

В исчислении высказываний из простых высказываний можно составлять сложные высказывания посредством применения логических операций, например:

Высказывание	Название операции	Обозначение
X и Y	Конъюнкция	$X \& Y$
X или Y	Дизъюнкция	$X \vee Y$
не X	Отрицание	$\neg X$
если X, то Y	Импликация	$X \rightarrow Y$

Истинность высказываний, получающихся в результате этих операций, определяется по следующей таблице:

X	Y	$X \& Y$	$X \vee Y$	$\neg X$	$X \rightarrow Y$
0	0	0	0	1	1
0	1	0	1	1	1
1	0	0	1	0	0
1	1	1	1	0	1

Формула является символьным представлением высказывания. Формула без операций называется *атомарной*.

Чтобы судить об истинности формулы, необходимо связать её с содержанием, т.е. выполнить *интерпретацию* формулы. Нельзя просто сказать, истинна некоторая формула или ложна. Но если мы объявим высказыванием предположение, например, что «Земля вращается вокруг Солнца», то можно утверждать, что это высказывание истинно.

Интерпретацию будем обозначать буквой **j**, за которой в скобках указывается обозначение интерпретируемой формулы. Например, то, что формула **X** в интерпретации **j** истинна, можно записать в виде $j(X) = 1$.

Основная цель логического вывода состоит в доказательстве того, является ли данное утверждение следствием других.

Формула **G** называется *логическим следствием* формул **F₁, F₂, ..., F_k**, если при любой интерпретации **j** из $j(F_1) = j(F_2) = \dots = j(F_k) = 1$ следует, что $j(G) = 1$.

Понятие логического следствия тесно связано с понятием выполнимости.

Множество формул $\{F_1, F_2, \dots, F_k\}$ называется *выполнимым*, если существует интерпретация **j** такая, что $j(F_1) = j(F_2) = \dots = j(F_k) = 1$.

Проверить выполнимость множества формул $\{F_1, F_2, \dots, F_k\}$ можно, построив для них таблицы истинности. Если найдется хотя бы одна строка, в которой в столбцах формул **F₁, F₂, ..., F_k** стоят единицы, то это множество формул выполнимо. Если такой строки нет, то множество формул невыполнимо.

Идея *метода резолюций* основана на следующей теореме.

Теорема. Формула **G** является логическим следствием формул **F₁, F₂, ..., F_k** тогда и только тогда, когда множество формул $L = \{F_1, F_2, \dots, F_k, \neg G\}$ невыполнимо.

Доказательство теоремы сводится к тому, что некоторая формула **G** (её называют *гипотезой* теоремы), является логическим следствием множества формул **F₁, ..., F_k** (*допущения* теоремы). Текст теоремы может быть сформулирован следующим образом: если формулы **F₁, ..., F_k** истинны, то истинна и формула **G**.

Метод резолюций

Задача логического вывода сводится к задаче проверки выполнимости множества формул $\{F_1, F_2, \dots, F_k, \neg G\}$. Чтобы установить невыполнимость такого множества формул, применяется определённое правило. В этом правиле используются следующие понятия:

- *литерал* – атомарная формула или её отрицание;
- *дизъюнкт* – дизъюнкция одного или нескольких литералов;
- *пустой дизъюнкт* – специальный дизъюнкт, не содержащий литералов. Для его обозначения используется специальный символ **■**. Считается, что пустой дизъюнкт ложен при любой интерпретации;
- *противоположные литералы* – литералы **X** и $\neg X$.

Правило резолюции. Из дизъюнктов $(X \vee F)$ и $(\neg X \vee G)$ выводим дизъюнкт $(F \vee G)$. Другими словами, дизъюнкт $(F \vee G)$ является логическим следствием дизъюнктов $(X \vee F)$ и $(\neg X \vee G)$. Правило резолюции — очень мощное правило! Многие правила, например, *modus ponens* являются частными проявлениями правила резолюции.

Резолюция – это приём, используемый при достоверном логическом выводе. Этот приём заключается в нахождении двух дизъюнктов, один из которых содержит литеру, а другой – её отрицание. На основании их сравнения формируется новый дизъюнкт, называемый *резольвентой*. Именно порождение новых дизъюнктов,

являясь основой метода резолюций, широко применяется в интеллектуальных системах.

Уточним понятие резолютивного вывода. Пусть S – множество дизъюнктов. *Выводом* из S называется последовательность дизъюнктов D_1, D_2, \dots, D_n такая, что каждый дизъюнкт этой последовательности принадлежит S или следует из предыдущих по правилу резолюции. Дизъюнкт D выводим из S , если существует вывод из S , последним дизъюнктом которого является D .

Применение метода резолюций основано на следующей теореме.

Теорема (о полноте метода резолюций). Множество дизъюнктов логики высказываний S невыполнимо тогда и только тогда, когда из S выводим пустой дизъюнкт \blacksquare .

Процедура доказательства по методу резолюций на самом деле является процедурой опровержения, то есть вместо доказательства общезначимости формулы доказывается, что ее отрицание противоречиво, в значительной степени, такой подход выбран, исходя из соображений «технического удобства». Метод резолюций – это метод автоматического доказательства теорем, основанный на опровержении множества посылок F и отрицания целевой теоремы G с использованием правила резолюции.

Алгоритм резолютивного вывода состоит в доказательстве того, что формула G является логическим следствием множества формул F_1, \dots, F_k и включает следующую последовательность шагов.

1. Составляется множество формул $\{F_1, \dots, F_k, \neg G\}$.
2. Каждая из этих формул приводится к конъюнктивной нормальной форме (КНФ). В КНФ зачёркиваются знаки конъюнкции. Получается множество дизъюнктов S .
3. Осуществляется поиск вывода пустого дизъюнкта \blacksquare из S . Если пустой дизъюнкт выводим из S , то формула G является логическим следствием формул F_1, \dots, F_k . Если из S нельзя вывести \blacksquare , то G не является логическим следствием формул F_1, \dots, F_k .

Рассмотрим несколько примеров вывода по методу резолюций.

Пример 1. Пусть даны два утверждения:

- 1) «Яблоко красное и ароматное»,
- 2) «Если яблоко красное, то яблоко вкусное».

Докажем утверждение, что «Яблоко вкусное».

Для этого введем множество формул, описывающих простые высказывания, соответствующие приведённым выше утверждениям:

X_1 – «Яблоко красное»,

X_2 – «Яблоко ароматное»,

X_3 – «Яблоко вкусное».

Исходные утверждения запишем в виде следующих формул:

$X_1 \& X_2$ — «Яблоко красное и ароматное»,

$X_1 \rightarrow X_3$ — «Если яблоко красное, то яблоко вкусное».

Утверждение, которое надо доказать, выражается формулой X_3 .

Докажем, что X_3 является логическим следствием формул $(X_1 \& X_2)$ и $(X_1 \rightarrow X_3)$.

Для этого составляем множество формул с отрицанием доказываемого высказывания:

$\{(X_1 \& X_2), (X_1 \rightarrow X_3), \neg X_3\}$.

Приводим все формулы к КНФ:

$\{(X_1 \& X_2), (\neg X_1 \vee X_3), \neg X_3\}$.

Зачёркивая конъюнкции, получаем следующее множество дизъюнктов:

$S = \{X_1, X_2, (\neg X_1 \vee X_3), \neg X_3\}$.

Ищем вывод пустого дизъюнкта:

$X_1, (\neg X_1 \vee X_3), X_3, \neg X_3, \blacksquare$.

Получили пустой дизъюнкт. Следовательно, утверждение о том, что яблоко вкусное, верно.

Пример 2. Пусть даны два утверждения:

1) $X_1 \rightarrow X_2$,

2) $X_2 \rightarrow X_3$.

Докажем, что утверждение $X_1 \rightarrow X_3$ является логическим следствием из этих утверждений. Для этого составляем множество формул с отрицанием доказываемого высказывания:

$\{(X_1 \rightarrow X_2), (X_2 \rightarrow X_3), \neg(X_1 \rightarrow X_3)\}$.

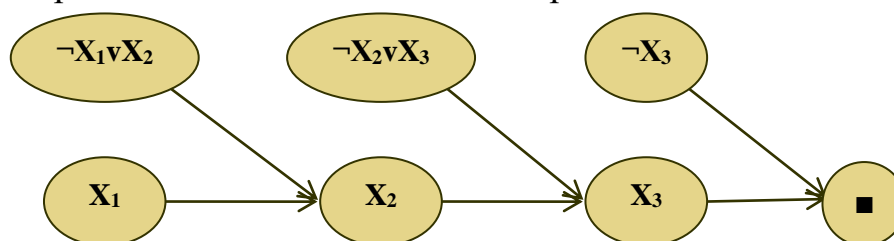
Устраняем импликации и приводим все формулы к КНФ:

$\{(\neg X_1 \vee X_2), (\neg X_2 \vee X_3), \neg(\neg X_1 \vee X_3)\} = \{(\neg X_1 \vee X_2), (\neg X_2 \vee X_3), (X_1 \& \neg X_3)\}$.

Зачёркивая конъюнкции, получим множество из четырёх дизъюнктов:

$S = \{(\neg X_1 \vee X_2), (\neg X_2 \vee X_3), X_1, \neg X_3\}$.

Представим резолютивный вывод в виде дерева поиска:



Получен пустой дизъюнкт. Следовательно, утверждение $X_1 \rightarrow X_3$ является логическим следствием утверждений $X_1 \rightarrow X_2$ и $X_2 \rightarrow X_3$.

Отметим также, что известное правило логического вывода «*modus ponens*» (если X – истина и из $X \rightarrow G$, то G – истина) получается по правилу резолюции: из дизъюнктов (X) и $(\neg X \vee G)$ выводим дизъюнкт G . Кроме того, метод резолюций является обобщением метода доказательства от противного.

Метод резолюций является *частично корректной процедурой*. Потому что возможен следующий исход: процесс не заканчивается, правило резолюции применяется, множество предложений пополняется, среди них нет пустых, и есть резольвируемые. Исход нельзя назвать заикливанием, поскольку нет никакого способа определить, почему метод резолюции продуцирует новые резольвенты, но при этом не получается пустого предложения. Означает ли это, что теорема верна, но мы просто не дождались завершения? Не известно. Означает ли это, что в результате так никогда и не получится пустой формулы? И этого мы тоже не знаем. Это логически неразрешимая задача. В принципе нельзя узнать, как долго нужно ждать для того, чтобы получить ответ на этот вопрос.

Метод резолюций применим также к логике предикатов 1-го порядка. Для этого требуется внести дополнения к версии метода для логики высказываний. В частности, в исчислении предикатов используется понятие *переменной*. Поэтому правило резолюций необходимо дополнить возможностью делать *подстановку* переменной.

В методе резолюций для логики предикатов используются ещё два правила *склейки*.

В остальном всё сводится к проверке невыполнимости множества дизъюнктов: множество дизъюнктов S логики первого порядка невыполнимо тогда и только тогда, когда из S выводим пустой дизъюнкт \blacksquare .

Например, пусть даны два утверждения:

- 1) «Существуют студенты, которые любят всех преподавателей»,
- 2) «Ни один из студентов не любит невежд».

Докажем утверждение «Ни один преподаватель не является невеждой».

Введем множество формул, описывающих следующие простые утверждения:

$C(x)$ – x есть студент;

$P(y)$ – y есть преподаватель;

$H(y)$ – y есть невежда;

$L(x, y)$ – x любит y .

Тогда два исходных утверждения можно записать в виде формул:

- 1) $x(C(x) \& y(P(y) \rightarrow L(x, y)))$;
- 2) $x(C(x) \rightarrow y(H(y) \rightarrow \neg L(x, y)))$.

Утверждение, которое надо доказать, выражается следующей формулой:

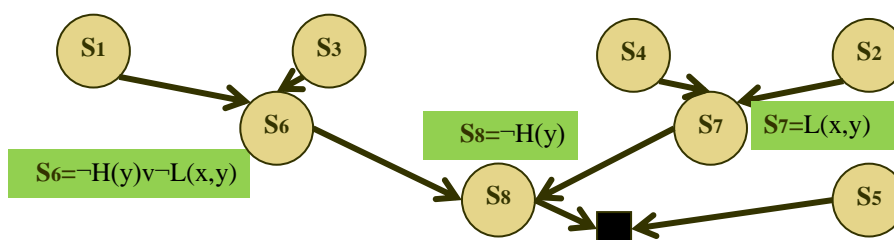
$$y(P(y) \rightarrow \neg H(y)).$$

Применяя специальный алгоритм, приводим утверждения к набору дизъюнктов $S = \{S_1, S_2, S_3, S_4, S_5\}$:

$$S_1 = C(x); S_2 = \neg P(y) \vee L(x, y); S_3 = \neg C(x) \vee \neg H(y) \vee \neg L(x, y);$$

$$S_4 = P(y); S_5 = H(y).$$

Дерево вывода для данного примера, построенное согласно методу резолюций для логики предикатов 1-го порядка, имеет вид, представленный на рисунке:



Одним из направлений в ИИ является разработка инструментального программного обеспечения для решения интеллектуальных задач: языков программирования, ориентированных на особенности задач ИИ; языков представления знаний; оболочек экспертных систем и других инструментальных средств.

Например, язык Пролог (*Prolog*) – это декларативный язык программирования для задач искусственного интеллекта, обработки естественных языков и др. В настоящее время имеется обширное семейство языков, построенных на базе языка Пролог, например: *Prolog++*, *Turbo Prolog* и др. В Прологе реализован частный случай линейной резолюции (на дереве вывода одна центральная ветвь и множество боковых ветвей).

Основной конструкцией Пролога являются *дизъюнкты Хорна* в обратной импликативной записи:

$$Q \leftarrow P_1 \& P_2 \& \dots \& P_n,$$

где Q – заголовок дизъюнкта (факты); $P_1 \& P_2 \& \dots \& P_n$ – тело (множество целей, которые надо доказать).

Пролог-программа интерпретирует метод поиска по дереву решений, причём интерпретатор произвольно выбирает дизъюнкты до получения пустого множества целей. Используется «поиск в глубину» (основной в Прологе), «поиск в ширину» и др.

На Прологе легко решаются разного рода головоломки, например, ханойская башня, задача о 8 ферзях, о волке, козе и капусте.



ПРИМЕР

Найти последовательность, которую должен выполнить человек для переезда в лодке через реку волка, козы и капусты. В лодке может

<p>поместиться человек, а с ним или волк, или коза, или капуста. Если оставить волка с козой без человека, то волк её съест. Если оставить козу с капустой, то она её съест. При человеке никто никого не съест.</p>
--

Решением этой задачи является алгоритм пересечений реки в виде графа возможных состояний, на котором необходимо найти путь, удовлетворяющий исходным условиям. Граф состояний строится при условии, что лодка может быть использована для перевоза человека и волка, человека и козы, человека и капусты и только человека.

Задача Стимроллер

Бенчмаркой для оценки эффективности алгоритмов вывода является *задача Стимроллер*. Эта задача была сформулирована Шубертом в 1978 г. в качестве теста для систем достоверного логического вывода и доказательства теорем.

Задача состоит в определении логической корректности рассуждения:

1. Существуют волки, лисы, птицы, гусеницы и улитки.
2. Волки, лисы, птицы, гусеницы и улитки – животные.
3. Существуют злаки.
4. Злаки – растения.
5. Каждое животное либо ест все растения, либо ест всех животных, которые меньше него и едят некоторые растения.
6. Гусеницы и улитки меньше птиц.
7. Птицы меньше лис.
8. Лисы меньше волков.
9. Волки не едят лис и злаки.
10. Птицы едят гусениц.
11. Птицы не едят улиток.
12. Гусеницы и улитки едят некоторые растения.
13. Следовательно, существует животное, которое ест некоторых питающихся злаками животных.

Чтобы решить задачу на Прологе надо записать ее условия на языке исчисления предикатов 1-го порядка. Для этого каждое их понятий представляется предикатом: $A(X)$: X – животное; $B(X)$: X – птица; $C(X)$: X – гусеница; $F(X)$: X – лиса; $G(X)$: X – злак; $P(X)$: X – растение; $S(X)$: X – улитка; $W(X)$: X – волк; $E(X, Y)$: X ест Y ; $M(X, Y)$: X меньше Y .

С их помощью описывается множество посылок, например:

1. $W(w)$ (Существуют волки)
2. $\neg F(X) \vee A(X)$ (Лисы - животные)

3. $\neg G(X) \vee P(X)$ (Злаки - растения)
4. $\neg A(X) \vee \neg A(Z) \vee P(Y) \vee \neg P(V) \vee E(X, Y) \vee \neg M(Z, X) \vee \neg E(Z, V) \vee E(X, Z)$ (Каждое животное либо ест все растения, либо ест всех животных, которые меньше него и едят некоторые растения)
5. $\neg C(X) \vee \neg B(Y) \vee M(X, Y)$ (Гусеницы меньше птиц)
6. $\neg W(X) \vee \neg G(Y) \vee \neg E(X, Y)$ (Волки не едят злаки)
7. $\neg C(X) \vee E(X, h(X))$ (Гусеницы едят некоторые растения) и т.д.

Здесь где X, Y, Z, V – переменные; w, f, b, c, s, g – константы; h и i – сколемовские функции. Эти функции служат для удаления кванторов существования (\exists или A^{-1}).

Заключение задачи можно записать следующим образом:

$E^{-1} X E^{-1} Y (A(X)) \& A(Y) \& (E(X, Y) \& E^{-1} Z (G(Z) \& E(Y, Z)))$ или его отрицание

$\neg A(X) \vee \neg A(Y) \vee \neg G(Z) \vee \neg E(Y, Z)$ (Не существует животного, которое ест некоторых, питающихся злаками, животных).

Задача характеризуется экспоненциальным ростом пространства поиска. Анализ хода решения задачи «Стирроллер» на Прологе показывает, что «ход рассуждений» программы мало похож на человеческий.

Мы разбиваем задачу на подзадачи и стремимся делать общие, а не частные выводы. Например, «Волки не едят растения». Пролог-программа пытается получить частные факты. Например, «Лиса f ест птицу b » и пробует найти доказательство путем подстановки частных фактов в исходные посылки.

Для эффективного решения задачи в процессе логического вывода необходимо сокращать множество дизъюнктов и находить наилучшие связи для резольвент.

Ядро всякой экспертной системы можно рассматривать как систему автоматического поиска вывода в некоторой формальной теории, то есть как доказательство теорем. Доказательство теорем безусловно предполагает затрату интеллектуальных усилий, требующих не только дедуктивных навыков, но и в значительной степени интуитивных догадок (например, какие леммы необходимы для той или иной теоремы). К настоящему времени разработано несколько программ с элементами ИИ, способных работать в этом направлении, то есть программ автоматического доказательства теорем (*Automatic Theorem Proving*).

В основе работы этих программ положены:

- обогащенный язык исчисления предикатов первого порядка,
- алгоритм Тарского перечисления правильно построенных формул, который позволяет генерировать гипотезы исчерпывающим образом,
- метод резолюций.

Возникает подкупающая идея. Записать аксиомы предметной области на языке исчисления предикатов. Запустить алгоритм типа алгоритма Тарского, который будет генерировать новые гипотезы одну за другой. Каждую гипотезу будем пытаться автоматически доказывать методом резолюций, и если получится, то компьютер сам будет добывать все новые и новые знания. Процесс можно запустить на 24 часа в сутки, 7 дней в неделю, 365 дней в году, и не на одном компьютере, а на всех незанятых в данный момент.

Однако такой прямолинейный подход оказывается безнадежно неэффективным. Сложные, интересные теоремы при лобовом подходе автоматически доказать не удастся, потому что пространство перебора слишком велико. Тем не менее, автоматическое доказательство теорем - это фундаментальная тема ИИ.

Лекция «Эвристический поиск»

Стратегии поиска

Ньюэлл и Саймон обосновали две гипотезы, на которых базируется большинство исследований в области ИИ. В соответствии с этими гипотезами человеческое мышление представляет собой систему, оперирующую с символами, отображающими действительность.

Гипотеза 1. Необходимым и достаточным условием для осуществления интеллектуальных действий в символьных системах является универсальность формальных манипуляций над конкретными символами (достаточно прочесть строку символов, разделить её на компоненты и переупорядочить, добавив или удалив какие-то символы без учёта семантики символов).

Гипотеза 2. Символьные системы решают задачи при помощи поиска, т.е. они генерируют потенциальные решения и модифицируют их до тех пор, пока решения не будут удовлетворять заданным условиям поиска. Поиск - это движение по дереву решений от одних узлов этого пространства к другим.

Для реальных NP-задач дерево поиска может стать вычислительно необозримым, экспоненциально растущим при линейном увеличении размерности задачи (комбинаторный взрыв). Сокращение размерности достигается либо разбиением задачи на фрагменты, либо использованием методов эвристического программирования.

О каких признаках интеллекта уместно говорить применительно к интеллектуальным системам, реализующим поиск по дереву? ИС должна уметь в процессе поиска в наборе фактов распознать существенные и сделать выводы не только с использованием дедукции, но и с помощью аналогии, индукции, абдукции и т.д.

Можно ли, например, считать шахматную программу интеллектуальной системой? Если шахматная программа при повторной игре делает одну и ту же ошибку - то нельзя. Обучаемость, адаптивность, накопление опыта и знаний - важнейшие свойства интеллекта. Если шахматная программа реализована на компьютере с бесконечно-высоким быстродействием и обыгрывает человека за счет просчета всех возможных вариантов игры по жестким алгоритмам - то такую программу мы также не назовем интеллектуальной. Но если шахматная программа осуществляет выбор и принятие решений в условиях неопределенности на основе эффективных методов принятия решений и эвристического поиска, корректируя свою игру от партии к партии в лучшую сторону, то такую программу можно считать достаточно интеллектуальной.

Информативность дерева поиска

С позиции теории информации каждый ход при поиске решения должен приводить к уменьшению энтропии в системе от максимального значения до нуля. Согласно К. Шеннону, объем информации в сообщении связан с вероятностью получения этого сообщения обратной монотонной зависимостью. Тогда энтропия множества сообщений **M** является суммой вида:

$$U(M) = - \sum_i p(m_i) \log p(m_i), i = 1, \dots, n.$$

Например, в БЗ ИС имеется обучающая выборка, состоящая из 14 объектов, из них 9 относятся к классу I, а 5 – к классу II. Следовательно, энтропия множества сообщений будет равна:

$$- (9/14) * \log_2 (9/14) - (5/14) * \log_2 (5/14) = 0,94 \text{ бит.}$$

Чем более неожиданным является сообщение, тем оно информативнее. Отсюда следует следующая эвристика выбора при построении вершин дерева поиска: выбирается то состояние (правила, действие), которое сулит наибольший прирост информации. Это позволяет устранить наиболее узкое место в создании ЭС – найти эффективные решения задачи автоматизации приобретения и вывода в базе общезначимых знаний.

Поиск по дереву для задачи коллаборативной фильтрации

В качестве примера поиска и обучения по дереву решений возьмем задачу из области коллаборативной фильтрации. Коллаборативная (совместная) фильтрация должна давать автоматические прогнозы (фильтрацию) об интересах пользователя по собранной информации о его вкусах. В век информационного взрыва такие технологии могут оказаться чрезвычайно полезными, поскольку количество объектов даже в одной категории (фильмы, музыка, книги, новости, веб-сайты) стало настолько большим, что отдельный человек не способен просмотреть их все, чтобы выбрать подходящие.

Принцип действия систем коллаборативной фильтрации довольно прост. Вначале создается база объектов-примеров (прецедентов) и задается шкала оценок (5-балльная, 10-балльная, 100-балльная). Затем множество пользователей проставляет оценки данным объектам по своему усмотрению. На основе оценок объектов и соответствия оценок конкретного пользователя оценкам других клиент получает список рекомендуемых к ознакомлению объектов с приблизительной оценкой каждого из них.

Предположим, что необходимо дать рекомендацию по продуктам культурного потребления, например кинофильмам, на базе анализа индивидуальных предпочтений пользователя. Параметров предпочтений может быть много, ограничимся четырьмя:

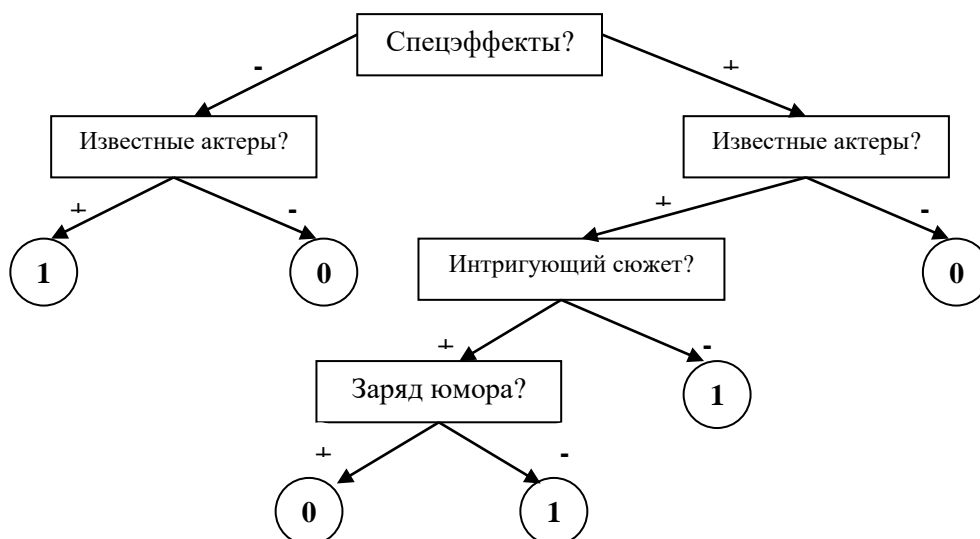
- наличие спецэффектов;
- в фильме играют известные актеры;
- интригующий сюжет;
- заряд юмора.

Ниже в таблице представлена статистика, полученная, например, путем анкетирования.

Фильм	Параметры	Спец- эффекты	Известные актеры	Сюжетная интрига	Заряд юмора	Рекомен- довать фильм?
«Обитаемый остров»		+	+	+	+	0
«Унесенные ветром»		+	+	+	-	1
«Криминальное чтиво»		+	+	-	-	1
«Игры разума»		-	+	-	-	1
«Девятая рота»		-	-	-	-	0
«Красотка»		-	+	-	+	1
«Неуловимые мстители»		+	-	+	+	0
«Стиляги»		-	-	+	-	??

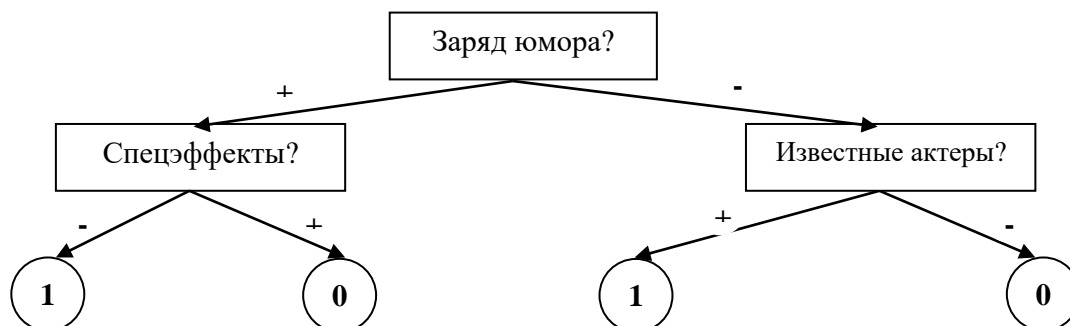
Рекомендовать ли фильм «Стиляги» с параметрами, представленными в последней строке таблицы? Попробуем ответить на этот вопрос с помощью деревьев решений.

В узлах дерева, не являющихся листьями, расположим параметры, по которым различаются фильмы. В узлах-листьях – значения целевой функции. Вначале построим дерево, используя параметры в том порядке, в котором они указаны в таблице. Полученное дерево представлено на рисунке:



Двигаясь от корня, определим, в какой из листьев попадает фильм «Стиляги». Судя по дереву, фильм рекомендовать не надо.

Однако глубина построенного дерева равна 4, что не идеально. Попробуем поместить в корень дерева вопрос о том, как в фильме с юмором. В случае если фильм не несет заряда юмора, то следующим параметром выберем наличие в фильме спецэффектов, а в случае если фильм юмористический, то будем смотреть на то, снимались ли в фильме известные актеры. Получим следующее дерево:



Глубина оптимального дерева решений равна 2, хотя дерево с такой глубиной, возможно, является не единственным. Как автоматически получить оптимальное дерево решений?

Чтобы построить оптимальное дерево, необходимо на каждом шаге выбирать параметры, наилучшим образом характеризующие целевую функцию. Это требование можно формализовать через понятие энтропии.

Пусть имеется множество A из n элементов. Из них m обладают некоторым свойством S . Тогда энтропия множества A по отношению к свойству S равна:

$$H(A, S) = -\frac{m}{n} \log_2 \frac{m}{n} - \frac{n-m}{n} \log_2 \frac{n-m}{n}.$$

Из этой формулы следует, что по мере возрастания пропорции m/n от 0 до 1/2 энтропия тоже возрастает, а после 1/2 – симметрично убывает. Если свойство S не бинарное, а может принимать s различных значений, каждое из которых реализуется в m_i случаях, то формула обобщается следующим образом:

$$H(A, S) = - \sum_{i=1}^s \frac{m_i}{n} \log_2 \frac{m_i}{n}.$$

В теории деревьев принятия решений принято следующее определение. Пусть множество A элементов, некоторые из которых обладают свойством S , уже классифицировано с помощью параметра Q , имеющего q возможных значений. Тогда прирост информации равен:

$$Gain(A, Q) = H(A, S) - \sum_{i=1}^q \frac{|A_i|}{|A|} H(A_i, S),$$

где A_i – множество элементов A , на которых параметр Q имеет значение i . На каждом шаге алгоритм должен выбирать тот параметр, для которого прирост информации максимален.

Определим, например, оптимальный параметр для рассмотренного примера. Вначале вычислим исходную энтропию:

$$H(A, \text{фильм}) = -\frac{4}{7} \log_2 \frac{4}{7} - \frac{3}{7} \log_2 \frac{3}{7} \approx 0,9852.$$

Теперь вычислим приросты информации для различных параметров:

$$\begin{aligned} Gain(A, \text{эффекты}) &= H(A, \text{фильм}) - \frac{4}{7} H(A_{\text{есть}}, \text{фильм}) - \frac{3}{7} H(A_{\text{нет}}, \text{фильм}) \approx \\ &\approx 0,9852 - \frac{4}{7} \left(-\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} \right) - \frac{3}{7} \left(-\frac{2}{3} \log_2 \frac{2}{3} - \frac{1}{3} \log_2 \frac{1}{3} \right) \approx 0,0202. \end{aligned}$$

Аналогично вычисляем прирост информации для других параметров:

$$\begin{aligned} Gain(A, \text{актеры}) &= H(A, \text{фильм}) - \frac{5}{7} H(A_{\text{есть}}, \text{фильм}) - \frac{2}{7} H(A_{\text{нет}}, \text{фильм}) \approx 0,47, \\ Gain(A, \text{сюжет}) &= H(A, \text{фильм}) - \frac{3}{7} H(A_{\text{есть}}, \text{фильм}) - \frac{4}{7} H(A_{\text{нет}}, \text{фильм}) \approx 0,13, \\ Gain(A, \text{юмор}) &= H(A, \text{фильм}) - \frac{3}{7} H(A_{\text{есть}}, \text{фильм}) - \frac{4}{7} H(A_{\text{нет}}, \text{фильм}) \approx 0,13. \end{aligned}$$

Максимальный прирост дает информацию о том, заняты ли в фильме известные актеры. Это логично: в обучающей выборке нет ни одного примера, когда фильм рекомендуется без участия известных актеров, т.е. на дереве решений одна из веток сразу оборвется! На дереве только в одном случае из 5 результат будет отличаться от других. Это означает, что при дальнейшем использовании алгоритма глубина дерева будет равна 3, т.к. одного параметра будет недостаточно для выделения одного случая из 5.

Как оптимизировать дерево в такой ситуации? Одним из наиболее эффективных алгоритмов оптимизации является алгоритм ID3.

Рекурсивный алгоритм ID3 (A, S, Q) (Дж. Квинлан).

1. Создать корень дерева.
2. Если свойство S (целевая булева функция) выполняется на всех элементах множества A , то поставить в корень метку **1** и выйти.
3. Если свойство S не выполняется ни на одном элементе A , то поставить в корень метку **0** и выйти.
4. Если $Q=0$ и если S выполняется на большей части A , то поставить в корень метку **1** и выйти;
Если $Q=0$ и если S не выполняется на половине или на большей части A , то поставить в корень метку **0** и выйти.
5. Выбрать Q из множества Ω , для которого прирост информации $Gain(A, Q)$ максимален.
6. Поставить в корень метку Q .
7. Для каждого значения q параметра Q :
 - а) добавить нового потомка корня и пометить соответствующее исходящее ребро меткой q ;
 - б) если в A нет ситуаций, для которых Q принимает значение q , то пометить этого потомка в зависимости от того, на какой части A выполняется S (аналогично п.4);
 - в) иначе запустить алгоритм **ID3**($A, S, \Omega \setminus \{Q\}$) и добавить его результат как поддерево с корнем в этом потомке.

Когда дерево принятия решений строится посредством алгоритма **ID3** или аналогичных ему, то оно удовлетворяет всем имеющимся данным. Однако на практике часто получаются слишком детализированные деревья, а число допускаемых ошибок прогнозирования растет. Это явление называется *оверфиттингом*. Предположим, что юмористический фильм рекомендуется в 90% случаев. Но среди исходных данных есть юмористический фильм с известными актерами, который не рекомендуется к просмотру. Результат работы алгоритма будет следующий: не рекомендовать к просмотру фильмы с аналогичными параметрами. Между тем рекомендацию для юмористических фильмов в 90% случаев никто не отменял, однако дерево решений перестанет справляться со своими задачами.

К преимуществам алгоритма ID3 относятся простота и интерпретируемость, линейная трудоемкость относительно длины обучающей выборки, простота реализации и возможность усовершенствования. Недостатки алгоритма: жадность, высокая чувствительность к составу выборки, склонность к переобучению.

Основная причина этих недостатков - неоптимальность жадной стратегии наращивания дерева. Для устранения указанных недостатков на практике применяют различные эвристические приемы: редукцию, элементы глобальной оптимизации, «заглядывание вперёд» и др.

Эвристические алгоритмы

Для реальных NP-задач дерево поиска может стать вычислительно необозримым и экспоненциально растущим при линейном увеличении размерности задачи (комбинаторный взрыв). Например, для игры «8» дерево поиска составляет $9!$, для шашек – 10^{40} , для шахмат – 10^{120} .

Сокращение размерности достигается либо разбиением задачи на подзадачи, либо использованием алгоритмов эвристического программирования: эволюционные алгоритмы, «жадные» алгоритмы, алгоритмы поиска в лабиринте, первый лучший (алгоритм A^*) и т.д.

Эвристика определяет набор правил или *оценочную функцию* выбора ветвей дерева, которые с наибольшей вероятностью ведут к приемлемому решению, однако её применение не гарантирует получения оптимального решения.

Сравнительный анализ конкурирующих эвристических алгоритмов оптимизации с точки зрения их результативности показал следующее. Пусть A_1 и A_2 – два конкурирующих эвристических алгоритма; F – множество целевых функций (задач) оптимизации; H – гистограмма их значений F ; m – количество полученных решений. Обозначим через $p(H|F, m, A)$ вероятность получения с помощью A -алгоритма m различных значений F , имеющих вид гистограммы H . Вольперт и Макрид сформулировали и доказали следующую теорему.

NFL-теорема (No-Free-Lunch, Вольперт-Макрид). Для любой пары эвристических алгоритмов поисковой оптимизации (A_1 и A_2) справедливо следующее соотношение:

$$\sum_F p(H | F, m, A_1) = \sum_F p(H | F, m, A_2).$$

Это означает, что все эвристические алгоритмы оптимизации в среднем одинаково результативны при их сравнении по всем F . Если по алгоритму A_1 получаются результаты лучшие, чем по алгоритму A_2 , то существуют оптимизационные задачи с другой целевой функцией, для которых лучшие результаты даёт алгоритм A_2 . При этом время решения задачи конкурирующими алгоритмами не учитывается.

В качестве примера рассмотрим эвристический алгоритм «первый лучший» для головоломки «Восьмерка». Головоломка заключается в том, чтобы перемещая фишки по полю (3x3) перейти от исходной позиции к целевой за минимального числа перестановок. Эвристические процедуры поиска на графе стремятся к тому,

чтобы минимизировать некоторую комбинацию стоимости пути к цели и стоимости поиска. Оценочная функция может быть разной:

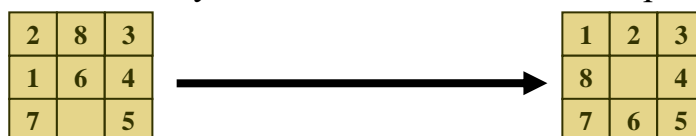
$$f_1 = \sum_{j=1}^8 d_j, \text{ где } d_j - \text{число ходов } j\text{-й фишки от текущей до целевой позиции};$$

$$f_2 = \sum_{j=1}^8 c_j, \text{ указывает на число фишек, стоящих не на месте.}$$

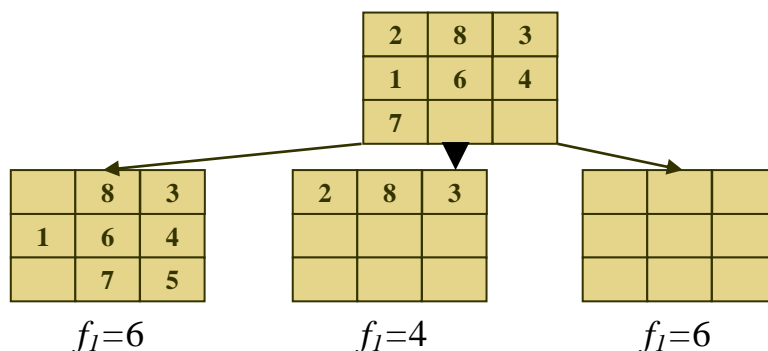
Выбор хода осуществляется по минимальному значению оценочной функции. Пусть применяется стратегия поиска в глубину, а оценочной функцией является f_1 . Тогда алгоритм «первый лучший» состоит в следующем:

1. Вычисляется функция f_1 для начальной расстановки фишек (корень дерева).
2. В зависимости от местоположения пустой клетки рассматривается несколько вариантов хода и для каждой позиции вычисляется f_1 (строим ветви дерева).
3. Выбираем ход (ветвь дерева) с *минимальным* значением f_1 (поиск в глубину дерева решений). Процесс повторяется (пп.1-2 алгоритма) до тех пор, пока f_1 не станет равным 0 (достигли целевой позиции).

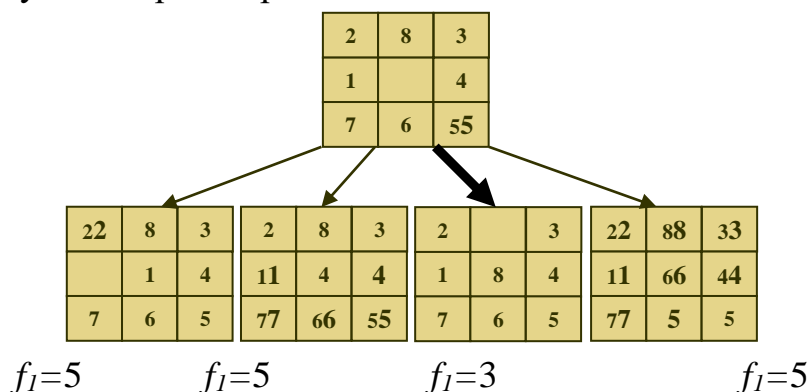
Применим алгоритм для следующих исходной и целевой расстановки фишек:



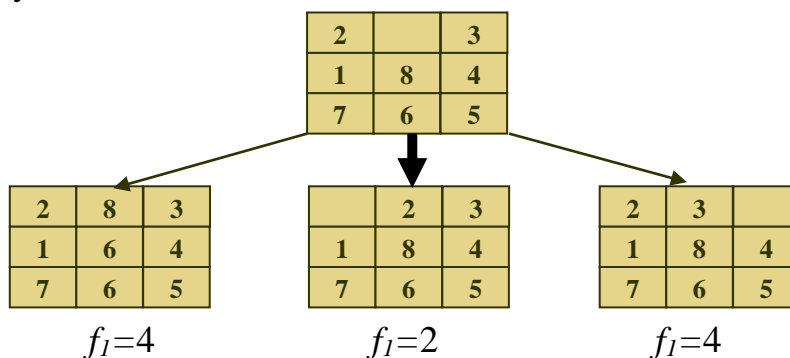
Положение пустой клетки в исходной расстановке предусматривает 3 варианта хода:



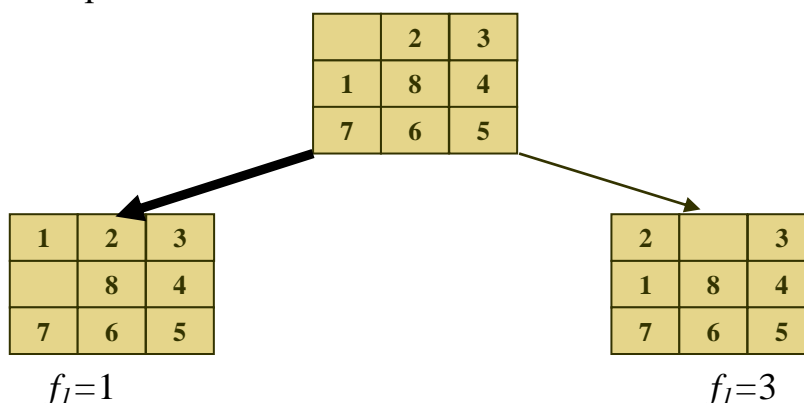
Выбираем ход (ветвь дерева) с минимальным значением ($f_1=4$) и строим дерево решений в глубину. Повторяем процесс.



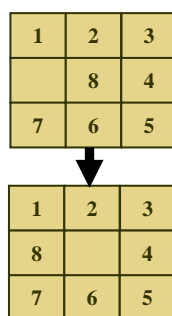
Выбираем ход с минимальным значением $f_l=3$ и продолжаем строить дерево решений в глубину.



Выбираем ход с минимальным значением ($f_l=2$) и строим дерево решений в глубину. Повторяем процесс.



Выбираем ход с минимальным значением ($f_l=1$), строим дерево решений в глубину и повторяем процесс.



Положение пустой клетки предусматривает 3 варианта хода. Однако первый ход приводит к результату $f_l = 0$. Поиск завершается, целевое состояние достигнуто.

Сокращение перебора в задаче можно оценить количественно с помощью критерия целенаправленности:

$$P = L / T,$$

где L – длина пути от корня дерева до целевого состояния, T – число вершин дерева, построенных в процессе перебора (исключая корневую). Целенаправленность перебора в данной задаче равна $P=5/13=0,385$.

Другими наиболее известными эвристическими алгоритмами поиска являются алгоритм минимакса и альфа-бета процедура.

Алгоритм минимакса предложили О. Моргенштерн и Дж. фон Нейман (1945 г.). Он нашёл широкое применение для решения задач теории антагонистических игр с нулевой суммой. Идея алгоритма состоит в предположении, что противники используют одинаковую оценочную функцию. Выбор хода со стороны одного игрока определяется максимальным значением оценочной функции для текущей позиции. В то время как противник стремится сделать ход, который минимизирует оценочную функцию. Поэтому этот метод и получил название минимакса.

Альфа-бета процедура теоретически эквивалентна алгоритму минимакса. С её помощью всегда получается такой же результат, как по алгоритму минимакса, но заметно быстрее, так как целые части дерева решений исключаются без проведения анализа. В основе этой процедуры лежит идея Дж. Маккарти об использовании двух переменных, обозначенных α и β (1961 г.): сравниваются наилучшие оценки, полученные для полностью изученных ветвей дерева, с наилучшими предполагаемыми оценками для оставшихся. Можно показать, что при определенных условиях некоторые вычисления являются лишними.

Использование алгоритмов эвристического поиска для поиска по И/ИЛИ-дереву выигрышной стратегии в более сложных задачах и играх (например, в шахматах или в Го) нереален. Хотя и здесь имеются определенные достижения. В частности, игра Го – древнее шахмат и требует немыслимого перебора ходов. Однако программа *MoGo* с гандикапом в 7 камней победила игрока 9 дана. «Минимаксное» дерево поиска, подходящее для шахматных программ, – в Го работает плохо. Компьютерщики воспользовались алгоритмом *Monte Carlo Tree Search*: программа, стартуя из имеющейся позиции, сначала перебирает случайно несколько миллионов игр, которые могут быть сыграны из этого положения. Каждую игру машина проигрывает по правилам до конца, не заботясь о разумности или выгоды ходов. Получив запись ходов для каждой игры, программа составляет статистику – какие ходы ведут к выигрышу. Далее выбирается выигрышный ход и проверяется ещё миллион игр из нового положения.

Лекция «Правдоподобный вывод»

Индуктивный вывод

Знания о предметных областях являются неполными, неточными и лишь правдоподобными. Это приводит к эффекту *немонотонности* процессов их поиска и вывода, снижению достоверности. Поэтому наряду с классическими методами достоверного дедуктивного вывода применяются методы правдоподобного вывода и эвристического поиска.

Индуктивный вывод - это вывод от частного к общему. Он позволяет на основании обобщения частных примеров выдвинуть гипотезу о существовании общей закономерности. В интеллектуальных системах, использующих индуктивный вывод, работает механизм, позволяющий при формировании гипотезы приписывать ей оценку правдоподобия (например, вероятность того, что данная гипотеза является истинной).

Для проверки индуктивного вывода необходимо большое число частных случаев, примеров, опытов, подтверждающих данный вывод. Для опровержения индуктивного вывода достаточно одного единственного контр примера.

Пример индуктивного вывода. Известно, что

(1) Эти студенты являются студентами группы КТбо4-2,

(2) Эти студенты юны.

Следовательно, *все студенты из группы КТбо4-2 юны.*

Задача. Некий человек устроил себе тайный погребок: пробил в стене квадратное отверстие, разделил его на 9 отделений и в каждое отделение поместил бутылки дорогого анжуйского вина, до которого он был большой охотник, - так, как показано на рисунке: в угловые отделения он поместил по 6 бутылок, а в средние – по 9, центральное же отделение оставил пустым (для закуски):

6	9	6
9		9
6	9	6

Этот человек регулярно проверял сохранность своего погребка, однако, ленился пересчитывать все бутылки и ограничивался тем, что считал их количество по сторонам квадрата: на каждой стороне была ровно 21 бутылка.

У него был слуга, тоже любитель анжуйского вина. Заметив, что хозяин следит только за тем, чтобы на каждой стороне было по 21 бутылке, слуга украл 4 бутылки, а остальные расставил так, что на каждой стороне осталось по 21 бутылке. Через

некоторое время слуга украл еще 4 бутылки и опять хозяин ничего не заметил. Спрашивается: как слуга расставил бутылки после очередной кражи и сколько всего бутылок он смог украсть незаметно для хозяина?

Примером научной индукции может служить следующее рассуждение о знаниях студентов КТб04-2, КТб04-12 по СИИ. Так, выбрав 5 студентов, которые регулярно пропускают лекции по СИИ, можно отметить, что не у одного из них каких либо знаний по СИИ не выявлено. Если на этой основе сделать обобщение, что все студенты групп КТб04-2, КТб04-12 не обладают никакими знаниями по СИИ, то очевидно, что такая популярная индукция даст маловероятное заключение. Иное дело, если выбор того же количества студентов будет сделан с учетом регулярного присутствия на лекциях и наличия умного лица (в очках). Тогда можно предположить, что все студенты групп КТб04-2, КТб04-12 обладают феноменальными знаниями по СИИ. Однако более достоверным будет заключение, если выбирать студентов случайно (очередное доказательство связи между успеваемостью студентов и зависанием на социальных сервисах получили в Университете Огайо. Согласно опросу среди студентов этого вуза, у тех, кто регулярно пользуется сервисом *Facebook*, средний балл колеблется в пределах 3-3,5, в то время как прочие учатся в среднем на 3,5-4 балла).

Часто приходится иметь дело с массовыми события (рождаемость людей, распространение заболеваний, динамика преступлений, всевозможные прогнозы погоды, выборов, продажи товаров и т.д.). Необходимо установить устойчивое распределение основных признаков явления. Здесь возникает вопрос: до каких пределов можно уменьшить численность объектов с минимальным влиянием на результат обработки данных? Поэтому применяется выборка – один из видов неполной индукции. Статистика – это неполный вывод по индукции, в котором количественная информация о частоте признака в выборке переносится на все множество.

Тогда схема индуктивного вывода имеет примерно следующий вид: Если $n\%$ выборки имеет данный признак, вероятно и $n\%$ всей совокупности будет иметь этот признак.

Еще пример вывода по индукции. Студенты Миша, Никита, Петр и Олег участвовали в соревнованиях. На вопрос, какие ими были заняты места, были получены следующие ответы:

1. Олег был второй, Петр – третий.
2. Олег был первым, Никита – вторым.
3. Миша был вторым, Петр – четвертым.

В каждом из ответов, одна часть верная, другая – нет. Какие места заняли студенты в соревнованиях?

Обозначим: Миша – М, Никита – Н, Петр – П, Олег – О. Введем индексы, которые будут соответствовать возможным, занятым ими местами. Составим логические уравнения для каждого ответа, причем, зная, что одна часть ответа истина, то дизъюнкция для каждого ответа будет истинной, поэтому в правой части уравнения поставим 1.

$O_2 + P_3 = 1$ – по первому ответу,

$O_1 + H_2 = 1$ – по второму ответу,

$M_2 + P_4 = 1$ – по третьему ответу.

Получили систему уравнений. Ответом на задачу будет конъюнкция этих уравнений. Можно перемножить сразу все уравнения, но лучше перемножим первые два, а после упрощения, умножим на третье.

$(O_2 + P_3)(O_1 + H_2) = 1$; $O_2 O_1 + O_2 H_2 + P_3 O_1 + P_3 H_2 = 1$, очевидно, что $O_2 O_1 = 0$ и $O_2 H_2 = 0$, поэтому осталось уравнение: $P_3 O_1 + P_3 H_2 = 1$, которое умножим на третье уравнение: $(P_3 O_1 + P_3 H_2)(M_2 + P_4) = 1$ – после упрощения получим: $P_3 O_1 M_2 = 1$.
Ответ. $O_1 M_2 P_3 H_4$ – занятые места.

Вывод по аналогии

Вывод, основанный на перенесении рассуждения из исследованной области на другую область, похожую на исследованную. Если имеется вывод $A \rightarrow B$, и область, в которой определено А, гомоморфна области, где определена С, а область, где определено В, гомоморфна области, где определено D, то вывод $A \rightarrow B$ порождает вывод $C \rightarrow D$. Вывод по аналогии - это частный случай правдоподобного вывода.

Вывод по аналогии дает возможность делать открытия новых свойств неизученных объектов на основании их аналогии с ранее изученными. Так, обратив внимание на аналогию между принципом действия нервной системы и цифровых вычислительных устройств, Норберт Винер начал свои исследования в области конструирования логических машин. Аналогично был открыт новый химический элемент – гелий, когда стало известно, что химический состав Солнца и Земли сходен по признакам. На Солнце был обнаружен еще не известный на Земле элемент (гелий). По аналогии было сделано предположение, что такой элемент есть и на Земле.

Умение проводить аналогии между различными, но в чем-то схожими явлениями, предметами и другими объектами помогает студентам осваивать науки, использовать на практике правила и формулы. Так, одно и то же дифференциальное уравнение описывает разные законы физики, биологии, социологии и др. Например, с помощью уравнения показательного роста $y' = ky$, имеющего решение $y = C \exp(kx)$

можно описать распад радиоактивного вещества, рост народонаселения. Аналогия – это умение переносить знания с одного предмета на другой в аналогичных заданиях.

Д. Пойа сформулировал два принципа вывода по аналогии:

- Предположение становится более правдоподобным, когда оказывается истинным аналогичное рассуждение;
- Предположение становится несколько более правдоподобным, когда становится более правдоподобным аналогичное предположение, а также схемы правдоподобных выводов:
- X аналогично Y . Если Y истинно, то Y более правдоподобно
- X аналогично Y . Если X более правдоподобно, то Y несколько более правдоподобно.

Уточнения этих схем связаны со степенью правдоподобия.

Вероятностный вывод

Это вывод, при котором каждое выражение, используемое в нем, имеет оценку правдоподобия в виде вероятности того, что оно является истинным. Например, из пруда неводом выловили 100 окуней, пометили их несмываемой краской и выпустили обратно. Через день снова поймали 100 окуней. Из них 5 штук оказались мечеными. Сколько окуней в пруду? – Ответ: каждый двадцатый лещ в озере – меченый, а всего в этом пруду 100 меченых лещей. Следовательно, можно предположить, что в пруду примерно 2000 окуней.

При вероятностном выводе применяются специальные процедуры для вычисления вероятности истинного значения результирующего выражения по вероятностям посылок, используемых при выводе, например байесовский вывод. Вывод базируется на теореме Байеса. Теорема Байеса подправляет вероятность гипотезы, данную новым свидетельством, следующим образом:

$$P(H|E) = \frac{P(E|H) P(H)}{P(E)},$$

где H представляет конкретную гипотезу, которая может быть, а может и не быть некоторой нулевой гипотезой. $P(H)$ называется априорной вероятностью H , которая была выведена прежде, чем новое свидетельство E стало доступным. $P(E|H)$ называется условной вероятностью наблюдения свидетельства E , если гипотеза H оказывается верной; её также называют функцией правдоподобия, когда она рассматривается как функция H для фиксированного E . $P(E)$ называется маргинальной вероятностью E : априорная вероятность наблюдения нового свидетельства E согласно всем возможным гипотезам; может быть вычислена по формуле полной вероятности:

$$P(E) = \sum P(E|H_i)P(H_i)$$

как сумма произведений всех вероятностей любого полного набора взаимно исключающих гипотез и соответствующих условных вероятностей. $P(H|E)$ называется апостериорной вероятностью H для данного E .

Пример 1 (из какой вазы конфета). Есть две полных вазы конфет. В 1-й вазе 10 шоколадных и 30 простых конфет, в то время как во 2-й вазе по 20 конфет каждого сорта. Ваша подруга выбирает вазу наугад, и затем выбирает конфету наугад. Конфета оказалась простой. Нет никакой причины полагать, что Ваша подруга рассматривает одну вазу иначе, чем другую, аналогично и для конфет. Насколько вероятно, что подруга выбрала конфету из 1-й вазы?

Интуитивно, решение кажется ясным. Точный ответ дается теоремой Байеса. Пусть H_1 - выбор вазы 1, а H_2 - выбор вазы 2. Предполагается, что вазы идентичны и $P(H_1) = P(H_2) = 0,5$.

Событие E - наблюдение простой конфеты. Из содержания ваз, мы знаем что $P(E|H_1) = 30/40 = 0,75$; $P(E|H_2) = 20/40 = 0,5$.

Формула Бейеса тогда даёт

$$p(H_1|E) = \frac{0,75 \times 0,5}{0,75 \times 0,5 + 0,5 \times 0,5} = 0,6.$$

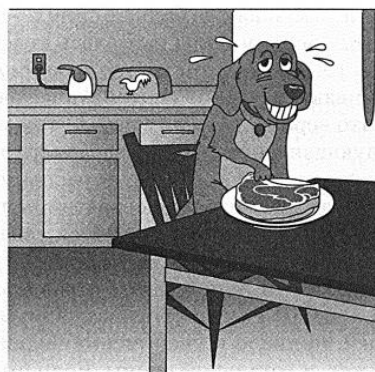
До того, как мы наблюдали конфету, вероятность, которую мы назначили для подруги, выбиравшую 1-ю вазу, была априорной вероятностью $P(H_1) = 0,5$. После наблюдения конфеты нужно пересмотреть вероятность $P(H_1|E)$, которая теперь равна 0.6.

Пример 2.

Проделка Фидо

Предположим, что вы оставили вашу собаку по кличке Фидо охранять ваш дом от грабителей, которые могут ворваться и украсть 10-фунтовый кусок мяса, размораживающийся на столе. Когда вы вернулись, все замки были в порядке, так что вы уверены, что никаких грабителей здесь не было. Однако мясо пропало. Само собой разумеется, что главный подозреваемый — Фидо.

На основе прошлого опыта, двух визитов к собачье-психиатру и хитрого взгляда вы оцениваете вероятность того, что это сделал Фидо, как 0,95. Однако, прежде чем обвинить Фидо, вы решаете получить еще одну улику. Вы готовите его обычный обед и предлагаете ему. К вашему удивлению, он съедает его до последней крошки. Едва ли этого можно ожидать от вора, который только что съел 10 фунтов мяса. Вы оцениваете вероятность того, что Фидо может сделать это, если он действительно съел мясо лишь в 0,02. Хотя обычно у него хороший аппетит и он съедает свой обед с вероятностью 0,99. Как вы должны пересмотреть ваши первоначальные подозрения, учитывая охотно съеденный обед? Очевидно, может оказаться полезной теорема Байеса. Учитывая только что съеденный обед, вероятность того, что Фидо виновен, можно выразить следующим образом:



$$P(\text{Виновен} | E) = \frac{P(E|\text{Виновен}) \times P(\text{Виновен})}{P(E|\text{Виновен}) \times P(\text{Виновен}) + P(E|\text{Невиновен}) \times P(\text{Невиновен})}$$

Мы знаем, что

$$\begin{aligned} P(\text{Виновен}) &= 0,95; \\ P(\text{Невиновен}) &= 0,05; \\ P(E|\text{Виновен}) &= 0,02; \\ P(E|\text{Невиновен}) &= 0,99. \end{aligned}$$

Поэтому

$$\begin{aligned} P(\text{Виновен} | E) &= \frac{(0,02)(0,95)}{(0,02)(0,95) + (0,99)(0,05)} = \\ &= \frac{0,0190}{0,0190 + 0,0495} = \\ &= \frac{0,0190}{0,0685} = \\ &= 0,28. \end{aligned}$$

До эксперимента с обедом обстоятельства складывались не в пользу Фидо. Однако при помощи теоремы Байеса мы смогли учесть результаты эксперимента с обедом и заключить, что Фидо, скорее всего, невиновен. Всякий любитель собак может на этом примере увидеть полезность теоремы Байеса.

Нечеткий вывод

Нечеткие вычисления оперируют с гранулированной информацией (*Information Granulation*), вычисления производятся со словами, а не с числами. Гранулирование информации — есть процесс объединения схожих точек или объектов в одну группу. Тогда нечеткость подобных групп есть прямое следствие нечеткости понятия сходства. Простыми примерами таких групп являются понятия «средний возраст», «деловая часть города» и др. Отметим, что стремление объединять схожие по свойствам объекты в одну группу характерно для человеческих рассуждений. Понятие гранулированной информации находят свое отражение в теории нечетких вычислений в терминах нечеткой и лингвистической переменной

Нечетким является вывод, при котором используются значения функций принадлежности. Правило вывода определяет значение этой функции для результата по значениям функций принадлежности посылок. Нечеткий вывод отличается от вероятностного подхода к представлению неопределенности знаний и базируется на

теории нечетких множеств Л. Заде. Теория имеет свою аксиоматику и набор базовых операторов, действующих несколько иначе, чем аналогичные булевы операторы.

В частности, нечетким множеством называется множество, определенное на произвольном непустом множестве X как множество пар вида:

$$\tilde{A} = \{\mu_{\tilde{A}}(x) / x\}, \text{ где } x \in X, \mu_{\tilde{A}}(x) \in [0, 1].$$

Здесь $\mu_{\tilde{A}}(x)$ - степень принадлежности элемента x нечеткому множеству \tilde{A} .

Проиллюстрируем понятие нечеткого множества на примере. Формализуем нечеткое понятие "горячий чай". В качестве множества X будет выступать шкала температуры в градусах Цельсия, которая будет изменяться от 0 до 100 градусов. Нечеткое множество для понятия "горячий чай" может выглядеть следующим образом:

$$\tilde{A} = \{0/0; 0/10; 0/20; 0,15/30; 0,30/40; 0,60/50; 0,80/60; 0,90/70; 1/80; 1/90; 1/100\}.$$

Так, чай с температурой 60 С принадлежит к множеству "Горячий" со степенью принадлежности 0,80. Для одного человека чай при температуре 60 С может оказаться горячим, для другого – не слишком горячим. Именно в этом и проявляется нечеткость задания соответствующего множества. Задание функции принадлежности нечеткому множеству осуществляется субъективно, экспертами, аксиоматически или эмпирически.

Понятия функции принадлежности и вероятности часто путают, ведь они имеют одинаковый диапазон изменения от 0 до 1, и оба описывают меру неопределенности. Однако эти меры совершенно различны: функция принадлежности является описанием сложного состояния; дополнительные данные не изменяют её значения, а вероятности зависят от частоты события, поэтому последующие выборки могут изменить вероятность.

Продemonстрируем их отличия на простом примере. Имеются два стакана воды и одна попытка утолить жажду. О воде в первом стакане известно, что степень её принадлежности к питьевой воде равна 0,9. Вода во втором стакане с вероятностью 0,9 является питьевой. Какой стакан выбрать для питья? - Вода в первом стакане похожа на питьевую, но не вполне пригодна для питья (возможна, она грязная). Вода во втором стакане с вероятностью 10% может оказаться непитьевой (например, ядом). Поскольку попытка утолить жажду у нас одна, а вода в первом стакане по большей части является питьевой, то предпочтительно выпить воду из первого стакана.

Операции конъюнкции и дизъюнкции в нечёткой логике также не имеют никакой связи ни с теорией вероятности, ни с булевой логикой. Степень принадлежности конъюнкции двух нечетких множеств \tilde{A} и \tilde{B} вычисляется как:

$$\mu_{\tilde{A} \& \tilde{B}}(x) = \min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)).$$

Степень принадлежности дизъюнкции двух нечетких множеств \tilde{A} и \tilde{B} вычисляется как:

$$\mu_{\tilde{A} \vee \tilde{B}}(x) = \max(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)).$$

Отрицанием нечеткого множества A называется множество \bar{A} с функцией принадлежности:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x),$$

для каждого $x \in X$.

Существуют различные алгоритмы нечеткого вывода: максиминная композиция, Мамдани, Сугено, Ларсена, Цукamoto. Они различаются главным образом видом используемых правил, логических операций и разновидностью метода дефазификации. Рассмотрим, например, правила максиминной композиции.

Пусть U и V - два множества, состоящих из переменных u и v соответственно. Пусть A и F - нечеткие подмножества множеств U и $U \times V$. Тогда композиционное правило нечеткого вывода утверждает, что из нечетких множеств A и F следует нечеткое множество $B = A \bullet F$:

$$\mu_B(v) = \max_{u \in U} \{ \min[\mu_A(u), \mu_F(u, v)] \}$$

Пусть $U = V = \{1, 2, 3, 4\}$, $A_{\text{МАЛЫЙ}} = \{(1/1), (0,6/2), (0,2/3), (0/4)\}$,

$F_{\text{ПРИМЕРНО РАВНЫ}} =$

	1	2	3	4
1	1	0,5	0	0
2	0,5	1	0,5	0
3	0	0,5	1	0,5
4	0	0	0,5	1

Тогда получим как результат композиционного нечёткого вывода:

$$B = [1 \ 0,6 \ 0,2 \ 0] \bullet \begin{bmatrix} 1 & 0,5 & 0 & 0 \\ 0,5 & 1 & 0,5 & 0 \\ 0 & 0,5 & 1 & 0,5 \\ 0 & 0 & 0,5 & 1 \end{bmatrix} = [1 \ 0,6 \ 0,5 \ 0,2],$$

что можно интерпретировать следующим образом:

$B = \text{БОЛЕЕ ИЛИ МЕНЕЕ МАЛЫЙ}$, если терм БОЛЕЕ ИЛИ МЕНЕЕ определяется как оператор увеличения нечёткости. Словами этот правдоподобный нечёткий вывод можно записать в виде:

u – МАЛЫЙ предпосылка

u, v – ПРИМЕРНО РАВНЫ предпосылка

v – БОЛЕЕ ИЛИ МЕНЕЕ МАЛЫЙ результат нечёткого вывода.

Существуют и другие модели нечеткого вывода.

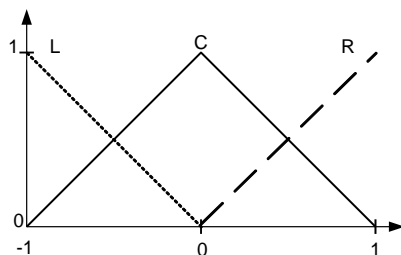
Одна из них представляется в виде совокупности нечетких продукционных правил следующего вида:

Если A есть R , то B есть L ;

Если A есть C , то B есть C ;

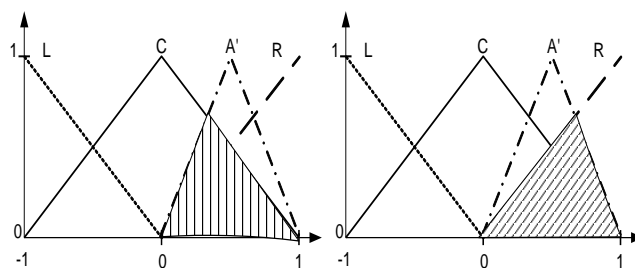
Если A есть L , то B есть R .

Здесь R , C , L – функции принадлежности нечетких множеств A и B , означающие соответственно: R – быть справа; C – быть в центре; L – быть слева. Зададим функции принадлежности R , C , L графически в треугольной форме:



Пусть наблюдается какая-то информация A , тогда *прямой нечеткий вывод* заключается в последовательном пересечении функций принадлежности R , C и L с множеством A .

Согласно нечеткой логике операция пересечения трактуется как минимум. Полученные результаты объединяются (операция объединения трактуется как максимум), и на основании A можно получить выводы о B : – фигура, получаемая объединением заштрихованных областей, представленных на рисунках:



Другими способами нечеткого вывода являются вывод по Мамдани и Суджено. В системах с выводом по Мамдани база знаний состоит из правил вида «Если X_1 =низкий и X_2 =средний, то Y =высокий». В системах с выводом по Суджено применяются правила вида «Если X_1 =низкий и X_2 =средний, то $Y=a_0+a_1X_1+a_2X_2$ », где a_0 – константа, a_1 , a_2 – весовые коэффициенты влияния X_1 и X_2 на Y . Отличие вывода по Мамдани от Суджено – в определении значений Y в правилах базы знаний. Значения Y по Мамдани задаются нечеткими термами, а по Суджено – как линейная комбинация входных переменных.

Приведем пример *обратного нечеткого вывода*. Пусть дано множество неисправностей автомобиля $X=\{x_1, x_2\}$, где x_1 – неисправность аккумулятора, x_2 – неисправность, связанная с отработкой машинного масла. Имеется множество симптомов $Y=\{y_1, y_2, y_3\}$, где y_1 – затруднения при запуске двигателя, y_2 – ухудшение цвета выхлопных газов, y_3 – недостаток мощности двигателя. Свяжем симптомы с неисправностями. Например, путем опроса экспертов выясним степени принадлежности каждого симптома y неисправности x , построив матрицу нечетких отношений:

$$\begin{aligned} &0.9/(x_1y_1), 0.1/(x_1y_2), 0.2/(x_1y_3); \\ &0.6/(x_2y_1), 0.5/(x_2y_2), 0.5/(x_2y_3). \end{aligned}$$

Пусть наблюдается симптом y_1 (не запускается двигатель). Так как $0.9/(x_1y_1) > 0.6/(x_2y_1)$, то состояние автомобиля оценивается следующим образом:

$$S=0.9/y_1+0.1/y_2+0.2/y_3.$$

Требуется определить неисправность из множества $A=\mu_1/x_1+\mu_2/x_2$, где μ_1 и μ_2 – шансы возникновения неисправностей x_1 и x_2 соответственно. Для этого необходимо решить уравнение вида

$$S=A \times R \text{ или}$$

$$\begin{bmatrix} 0.9 & 0.1 & 0.2 \end{bmatrix} = \begin{bmatrix} \mu_1 & \mu_2 \end{bmatrix} \times \begin{bmatrix} 0.9 & 0.1 & 0.2 \\ 0.6 & 0.5 & 0.5 \end{bmatrix}$$

Из этого матричного уравнения следует, что

$$0.9 = 0.9 \times \mu_1 + 0.6 \times \mu_2,$$

$$0.1 = 0.1 \times \mu_1 + 0.5 \times \mu_2,$$

$$0.2 = 0.2 \times \mu_1 + 0.5 \times \mu_2.$$

С учетом особенностей выполнения операторов умножения и сложения в нечеткой логике, из первого уравнения этой системы следует $\mu_1 \geq 0.9$; из второго уравнения – $\mu_2 \leq 0.1$; анализируя третье уравнение, получаем $\mu_2 \leq 0.2$.

Таким образом, $\mu_1 > \mu_2$.

Вывод: желательно заменить аккумулятор.

Абдуктивный вывод

Абдукция в анатомии означает отведение конечности в сторону, например, отведение мизинца от остальных пальцев, абдукция в бытовом значении означает похищение людей неизвестными существами, абдукция в ИИ – вид правдоподобного вывода от частного к частному.

Идея абдукции принадлежит Ч.Пирсу и состоит в следующем.

Пусть имеется некоторая группа фактов и гипотез **Th**, представляющих знания о предметной области. Если с их помощью можно объяснить имеющий место факт **f**, то они принимаются.

Абдуктивный вывод имеет следующую схему:

Пусть **h** – объяснение факта **f** в **Th**. Предположим, что **Th&f** – выполнимая формула, но **f** – не является следствием **Th**, т.е. знания о предметной области **Th** не объясняют факта **f**.

Необходимо вывести дополнительные факты **h**, объясняющие **f** относительно **Th**, т.е. **Th, h** \models **f**, причем к фактам **h** предъявляются требования *непротиворечивости и минимальности* (абдуктивное объяснение **h** для **f** считается непротиворечивым, если **Th&h** – выполнимая формула; минимальным – если каждое объяснение, являющееся логическим следствием **f**, эквивалентно **h**).

Пример (В.Н.Вагин). Пусть *Варвар* – имя слона. Тогда:

$Th : \forall x(\text{Слон}(x) \Rightarrow \text{Серый}(x))$

$f : \text{Серый}(\text{Варвар})$

$h : 1.\text{Слон}(\text{Варвар}),$

$2.\text{Слон}(\text{Цезарь}) \& \text{Несерый}(\text{Цезарь}),$

$3.\text{Слон}(\text{Варвар}) \& \text{Женщина}(\text{Старая леди})$

где 1 – минимальное и непротиворечивое абдуктивное объяснение факта **f** в **Th**; 2 – противоречивое объяснение; 3 – избыточное объяснение.

Абдукция – это форма правдоподобного вывода, механизм формирования гипотезы, объясняющий наблюдаемые факты на основе существующих теоретических положений.

В настоящее время разработано несколько методов абдуктивного вывода. Одной из проблем, возникающих в процессе нахождения гипотез-объяснений в процессе абдуктивного вывода, является экспоненциальная зависимость числа возможных гипотез от размерности задачи. Вероятность многих из логически возможных гипотез объяснений настолько мала, что их не стоит рассматривать. Таким образом, нужно учитывать вероятности найденных гипотез объяснений и не создавать маловероятные гипотезы. Другая проблема заключается в том, что абдуктивный вывод имеет немонотонный гипотетический характер, допускает много интерпретаций. Кроме того, современные языки логического программирования не позволяют в полной мере формализовать абдуктивный вывод.

Приведем примеры абдуктивного вывода:

- Всем известно, что трава во дворе дома будет мокрой после дождя или после работы дождевальной установки. Проснувшись утром, мы обнаружили, что трава мокрая. Соответственно, мы предлагаем гипотезу объяснения: ночью шел дождь либо работала дождевальная установка.

- Иоганн Кеплер пришел к заключению, что орбиты планет представляют собой эллипсы, на основании наблюдения того, что значения долготы орбиты Марса не соответствуют предполагаемой круговой орбите. Естественно, прежде чем предположить, что орбиты планет описываются эллиптически, он проанализировал

несколько других гипотез, не прошедших последующих проверок. Более того, Кеплер был вынужден сделать ряд достаточно сильных допущений, прежде чем открыл свой знаменитый закон «заметания площадей». Будучи сторонником гелиоцентрической гипотезы, он предположил, что именно Солнце является причиной наблюдаемого движения планет, а потом, опираясь на наблюдения Марса, обобщил свои представления на другие планеты, предположив «одинаковость» состояний всех планет.

- Известно, что (1) Все студенты из группы КТбо4-2 юны, (2) Эти студенты юны. Следовательно, эти студенты являются студентами группы КТбо4-2.

- Все люди смертны, Сократ смертен, следовательно, Сократ человек (может показаться, что здесь все нормально, но если вдуматься, то становится ясно, что вывод неправильный: из того, что Сократ смертен, вовсе не следует, что Сократ человек, ведь смертны и кошки, и собаки, и бабочки, и, может быть, деревья...).

- Все планеты круглые. Маша круглая. Значит, Маша - планета (полный бред, на самом деле, Маша - официант в пивном баре).

Вывод по абдукции успешно применяется для решения задач диагностики, понимания естественного языка, распознавания, накопления знаний, составления расписаний и, несомненно, является очень важной составляющей интеллектуальных систем.

Заключение

Тест Тьюринга сыграл определенную роль в развитии искусственного интеллекта, в том числе и критика самого теста. Здесь можно провести аналогию с авиацией. Хорошими летательными аппаратами, по логике теста Тьюринга, должны считаться такие, которые неотличимы от птиц до такой степени, что даже птицы принимают их за своих. Развитие авиации началось тогда, когда конструкторы перестали копировать птиц, а занялись аэродинамикой, материаловедением и теорией прочности. Робототехника стала индустрией после того, как перестала копировать анатомию человека.

За полвека развития искусственного интеллекта было предложено и проверено на практике достаточно много различных подходов к представлению и обработке знаний. Среди них естественным образом выделяются: продукционные правила, семантические сети, фреймы; логические исчисления, основанные на математической логике; другие (например, нейронные и эволюционные системы). Именно эти три группы методов положены в основу структуры книги. Предлагаемые в последнее время подходы к представлению знаний, явным образом тяготеют к одной из этих групп. Между системами продукций и логическим выводом много общего. Другие методы представления знаний, такие как фреймы, семантические сети, нейронные сети и им подобные, по своей природе эклектичны и, в конечном счете, опираются на базовый математический аппарат. Таким образом, знакомство с идеями представления и обработки знаний в рамках выделенных подходов является вполне достаточным для того, чтобы быть в курсе основных тенденций искусственного интеллекта

Несмотря на разнообразие проблем, связанных с выводом в ИИ, во всех областях разработки СИИ наблюдаются общие черты:

- использование компьютеров для доказательства теорем, решения головоломок, распознавания образов, обучения рассуждениям и др.;
- внимание к проблемам, не поддающимся алгоритмическим решениям, и слабо формализованным задачам;
- попытки решить задачи семантического поиска;
- попытки получить «хорошее» решение, когда точный ответ трудоемок или невозможен;
- использование специальных знаний (онтологические базы знаний, ЭС).

Необходимо понимать, что достоверный вывод в СИИ моделирует только функции, связанные с логической обработкой информации.

Эвристические алгоритмы увеличивают вероятность получения работоспособного, но не всегда оптимального решения творческой задачи, возникшей, например, из-за пробелов в теории, неполноты или недостоверности исходных данных. С помощью эвристических алгоритмов можно находить решения даже в очень сложных, непредвиденных ситуациях, однако по качеству решения они уступают точным алгоритмическим подходам.

Субъекты искусственного интеллекта получили право на жизнь после того, как прекратились попытки построить системы ИИ, думающие и действующие подобно людям, а начали строить системы, действующие и думающие рационально, т.е. достигающие наилучшего результата. Последние достижения в области ИИ можно представить следующими проектами.

Программа *Remote Agent* (Автономное планирование и составление расписаний), используется для комплексного управления работой космических аппаратов, удаленных далеко за пределы околоземной орбиты, в т.ч. диагностики и устранения неисправностей по мере их возникновения.

Программа *Deep Blue* компании IBM стала первой программой, которой удалось победить чемпиона мира в шахматном матче.

Система компьютерного зрения *Alvinn* была обучена вождению автомобиля, придерживаясь полосы движения. На протяжении 2850 миль система обеспечивала управление автомобилем в течение 98% времени.

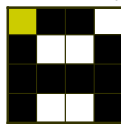
Медицинские диагностические программы сумели достигнуть уровня опытного врача в нескольких областях медицины.

Система *DART* (*Dynamic Analysis and Re-planning*) обеспечивает автоматизированное планирование поставок и составление графика перевозок, охватывая одновременно до 50000 автомобилей, людей и грузы. Разработчики этой системы заявили, что одно это применение окупит их 30-летние инвестиции в искусственный интеллект.

Нейросетевой вывод

Вывод в нейросетевой базе знаний существенно отличается от выводов в символично-логических ИС. Механизм нейросетевого вывода скрыт от пользователя. Кроме того, нейросетевая БЗ использует знания экспертов из предыдущего опыта относительно конкретной проблемы и строит не общую нейросеть, а сеть для решения именно этой проблемы. Поэтому при появлении новых знаний нейросеть должна быть дообучена и сохранена. Таким образом, нейросетевой вывод представляет собой итерационный процесс, при котором накапливается и сохраняется опыт эксперта.

Рассмотрим пример нейросетевого вывода. Пусть необходимо построить персептрон, способный распознавать образы: «буква А» и «не буква А». Пусть растровое представление буквы А имеет следующий вид:



Суммирующая функция нейросети имеет вид

$$F(X, W) = \sum x_{ij} \times w_{ij},$$

где x_{ij} – входные значения (0 или 1). Весовые коэффициенты w_{ij} вначале положим равными 1. Пороговое значение функции активации пусть равно 10. Обучая нейросеть, получим следующие значения весовых коэффициентов: $w_{11} = 0$, $w_{12} = 1$, $w_{13} = 1$, $w_{14} = 0$, $w_{21} = 1, \dots, w_{44} = 1$.

Пусть решающим правилом будет $\sum x_{ij} \times w_{ij} \geq 10$.

Перейдем в режим распознавания. Предъявим, например, букву **R** (отличается от буквы **A** в точке (1,1)). Поскольку решающее правило выполняется, то буква **R** будет распознаваться как буква **A**! Если же в решающем правиле установить строгое равенство, то персептрон станет очень чувствительным к форме написания буквы. Поэтому эксперту необходимо усложнять правила распознавания, например, определять минимум точек для распознавания буквы или определять особенности её геометрии и т.п. В итоге нейросетевой вывод с помощью персептрона как нелинейного преобразователя становится малопонятным.

Эволюционный вывод

ИИ - междисциплинарная наука. Она привлекает интересные идеи, теории из других областей знаний, адаптируя их для своих задач. Так было с нейронными сетями, с моделированием рассуждений, с компьютерной лингвистикой и т.д. Теория Ч. Дарвина (1859 г.) стала отправной точкой для еще одного направления исследований – эволюционного моделирования.

Основной тезис эволюционного моделирования – заменить процесс моделирования сложного объекта моделированием его эволюции. Дарвин сформулировал основной закон развития органического мира, охарактеризовав его взаимодействием трех следующих факторов:

- наследственность (потомки сохраняют свойства родителей);
- изменчивость (потомки почти всегда не идентичны);
- естественный отбор.

Понятие «эволюционное моделирование» сформировалось в работах Л.Фогеля, А.Оуэнса, М.Уолша. В 1966 г. вышла их совместная книга «ИИ и эволюционное моделирование». История эволюционных вычислений началась с разработки ряда различных независимых моделей. Основными из них были генетические алгоритмы (ГА) и классификационные системы Д.Холланда, ставшие классикой после выхода в

свет книги «Адаптация в естественных и искусственных системах» (*Adaptation in Natural and Artificial Systems*, 1975). В 70-х годах в рамках теории случайного поиска Л.А.Растригиным был предложен ряд алгоритмов, использующих идеи бионического поведения особей. Развитие этих идей нашло отражение в работах И.Л.Букатовой по эволюционному моделированию, М.Л.Цетлина по оптимальному поведению стохастических автоматов. В начале 90- годов Д.Коза разработал метод генетического программирования.

Алгоритмы эволюционного моделирования позволяют не только успешно решать трудные оптимизационные задачи, но с их помощью удалось заново открыть десятки запатентованных схмотехнических решений: усилители, фильтры, контролеры и т.д.

Эволюционный вывод – это процедура поиска решений на основе эволюционных вычислений. Эволюционные вычисления – это математические преобразования, трансформирующие согласно принятой модели входной поток информации в выходной, основаны на правилах имитации природных механизмов эволюционного синтеза, а также на статистическом подходе к исследованию ситуаций и итерационном приближении к искомому решению.

Достоинства эволюционных вычислений:

- широкая область применения;
- возможность проблемно-ориентированного кодирования решений, подбора начальной популяции, комбинирования эволюционных вычислений с другими эвристическими алгоритмами, продолжения процесса эволюции до тех пор, пока имеются необходимые ресурсы;
- пригодность для поиска в сложном пространстве решений большой размерности;
- отсутствие ограничений на вид целевой функции;
- ясность схемы и базовых принципов эволюционных вычислений;
- интегрируемость эволюционных вычислений с другими неклассическими парадигмами искусственного интеллекта, такими как искусственные нейросети и нечеткая логика.

Недостатки эволюционных вычислений:

- эвристический характер эволюционных вычислений не гарантирует оптимальности полученного решения;
- относительно высокая вычислительная трудоемкость, которая может быть преодолена за счет распараллеливания на уровне организации эволюционных вычислений и на уровне их непосредственной реализации в вычислительной системе;

- относительно невысокая эффективность на заключительных фазах моделирования эволюции (операторы поиска в эволюционных алгоритмах не ориентированы на быстрое попадание в локальный оптимум);
- нерешенность вопросов самоадаптации.

Эволюционный вывод может находить эффективные решения в самых разных областях, когда даже неизвестно, как искать правильное решение. Несмотря на кажущуюся хаотичность, процедура эволюционного вывода является очень устойчивой. Главное, чтобы у нас был хоть какой-то критерий оценки качества получаемых решений, т.е. признак, на основании которого производится сравнительная оценка возможных решений и выбор наилучшего. Этого достаточно, чтобы алгоритм эволюционного вывода заработал. Для реальных практических задач такой критерий обычно имеется.

Существует множество разновидностей эволюционного вывода, реализованных в виде генетических алгоритмов, генетического программирования, эволюционных стратегий, эволюционного программирования, роевого интеллекта, меметики и др. В общем случае алгоритмы эволюционного вывода включают следующие элементы:

$$\mathcal{EA} = (P_0, N, L, F, GO),$$

где P_0 – исходная популяция решений; N – размер популяции; L – длина хромосомы, кодирующей решение; F – целевая функция, определяющая «пригодность» решения; GO – эволюционные операторы (селекция, кроссинговер, мутация, транслокация и др.).

Алгоритм эволюционного вывода включает в себя следующие элементы:

– **Инициализация модели.** Она предусматривает генерацию начальной популяции $P(t=0)$ из μ индивидов (хромосом), представляющих собой закодированные решения исходной задачи;

– **Оценка решений.** Для каждого индивида \bar{x}_k устанавливается функция качества $\Phi(\bar{x}_k)$, которую называют фитнес-функцией или функцией пригодности, приспособленности, целевой функцией;

– **Генерация потомков.** Задача состоит в получении λ потомков из μ родителей. Для этого необходимо выбрать родителей, провести их рекомбинацию и репликацию (копирование и передача фенотипа родителей потомкам), а затем мутацию потомков;

– **Оценка решений-потомков.** Производится оценка каждого потомка и ограничивается размер промежуточной популяции;

– **Селекция;**

– **Проверка условий остановки алгоритма:** максимальное число шагов эволюции t_{max} , отсутствие прогресса в смысле заметного улучшения значений

целевой функции, малая разница между лучшим и худшим значением функции для текущей популяции и т.п.

Рассмотрим пример эволюционного вывода по алгоритму генетического программирования.

Постановка задачи. Выбор математического выражения, наилучшим образом описывающего вычислительную модель для принятия решений. Решить задачу символьной регрессии означает получить математическое выражение (регрессию), которое можно анализировать, упрощать, использовать для моделирования, оптимизации и т.п.

Исходные данные. Экспериментальные данные или экспертные знания об объекте, отражающие неизвестную зависимость определенного свойства объекта Y от другого свойства или параметра X со случайной погрешностью, распределенной, как правило, по нормальному закону.

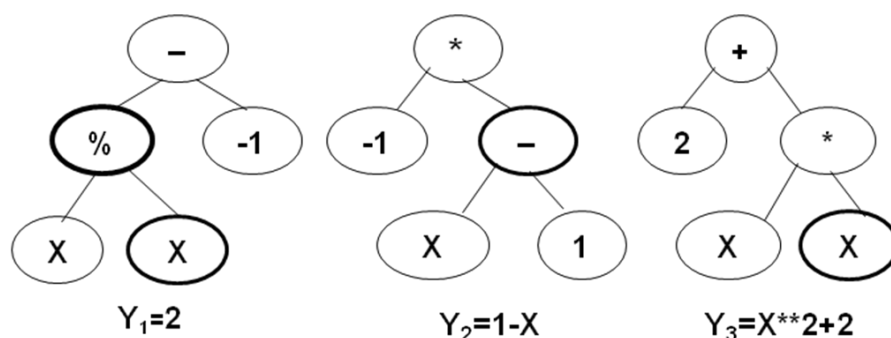
Найти: Регрессионную функцию, которая отображает зависимость Y от X с минимальной погрешностью, определив численные значения ее коэффициентов, т.е. идентифицировать объект. Например, найти функцию $Y = f(X) = ax + b$

Пусть регрессионной функцией является квадратный полином вида

$Y = X^2 - X + 2$, где X принимает значения в диапазоне $[-1, 1]$.

Необходимо с помощью ГП синтезировать компьютерную программу, реализующую данную регрессию.

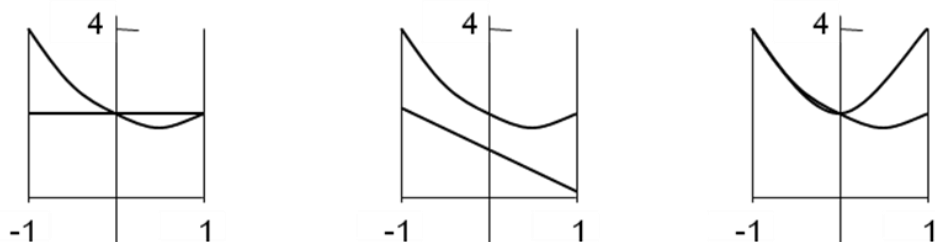
Для программного вычисления регрессии будем использовать множество математических операций: сложение (+), вычитание (−), умножение (*) и деление (%); - а также множество переменных и множество числовых констант из некоторого диапазона, например, от −2 до +2. Пусть случайно сгенерированы три программы:



Оценку программы, автоматически генерируемой с помощью генетических операторов, будем осуществлять по критерию близости к целевому значению функции получаемых программой результатов. Чем меньше *среднеквадратичная ошибка*, тем лучше программа.

Тогда ошибка регрессии для каждой из трех хромосом совпадает с областями между кривыми на интервале $[-1, 1]$:

$$Y = x^2 - x + 2$$

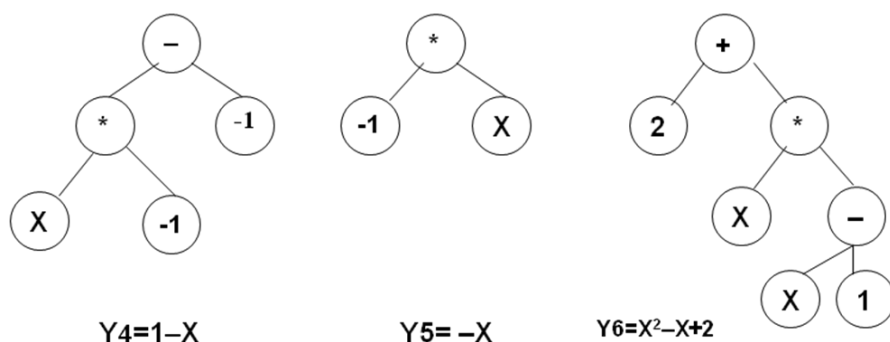


Абсолютная ошибка для $Y_1=2$ равна 1.0; для $Y_2=1-x$ равна 1.33; для $Y_3=x^2+2$ равна 1.0

Для формирования новой популяции хромосом используем операторы мутации и кроссинговера. Пусть *мутация* случайно заменяет в дереве, соответствующем Y_1 , вершину «%» на вершину «*», а правую вершину «X» на вершину «-1». Получим регрессию $Y_4=1-x$.

Кроссинговер между Y_2 и Y_3 дает два потомка: $Y_5=-x$ и $Y_6=x^2-x+2$.

После выполнения операторов мутации и кроссинговера новые хромосомы для Y_4 , Y_5 и Y_6 будут иметь следующий вид:



Дерево справа эквивалентно исходной регрессии, т.е. является алгебраически правильным решением задачи автоматического синтеза программы, реализующей вычисление функции квадратного полинома вида $Y=X^2-x+2$ в диапазоне значений $X [-1,+1]$.

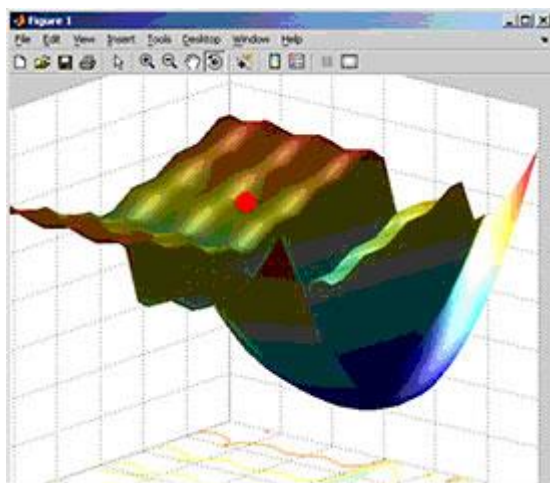
Оптимизация с использованием MATLAB и Genetic Algorithm and Direct Search Toolbox

Инженеры и научные работники постоянно связаны с проблемой поиска наиболее оптимального решения поставленной задачи, им приходится проводить анализ с целью выбора компромиссного решения, рассматривать множество вариантов. Сочетание Optimization Toolbox и недавно внедренного в MATLAB Genetic Algorithm and Direct Search Toolbox расширяет возможности для решения разнообразных оптимизационных задач.

Традиционные методы в Optimization Toolbox разработаны для решения задач оптимизации с непрерывными и дифференцируемыми целевыми функциями. Для негладких функций такие методы, как генетический алгоритм являются вполне оправданной альтернативой и реализованы в Genetic Algorithm and Direct Search Toolbox.

Возьмем некую задачу только с двумя переменными. Взятая нами целевая функция является кусочно-непрывной функцией, то есть имеются участки гладких функций с встроенными между ними разрывами одним примечательным участком, где функция является всюду недифференцируемой. Функция является гладкой в области точки минимума. Наша задача - найти минимум данной функции.

На рисунке представлен график выбранной нами целевой функции. Красное пятно относится к расположению особой области:



Минимизация целевой функции с использованием генетического алгоритма

В ГА не используются никакие виды производных с целью определения траектории наискорейшего спуска, что представляет собой определенные преимущества для задач с недифференцируемой функцией.

В Genetic Algorithm Tool определены следующие опции:

Функция пригодности - та целевая функция, которую необходимо оптимизировать.

Количество переменных в принимаемой оптимизации.

Графическая функция пользователя `gplotbestfun.m`, которая отображает на графике наилучшее значение функции на каждом этапе генерации в зависимости от числа итераций (или генераций). Эта функция доступна для загрузки.

Графический интервал - число генераций между последовательными обращениями к графическим функциям.

Область исходного поколения - область индивидуальных объектов в исходном поколении.

Основными настраиваемыми параметрами в GATool являются:

- популяция (вкладка Population);
- масштабирование (вкладка Fitness Scaling);
- оператор отбора (вкладка Selection);
- оператор репродукции (вкладка Reproduction);
- оператор мутации (вкладка Mutation);

- оператор скрещивание (вкладка Crossover);
- перенесение особей между популяциями (вкладка Migration);
- специальные параметры алгоритма (вкладка Algorithm settings);
- задание гибридной функции (вкладка Hybrid function);
- задание критерия остановки алгоритма (вкладка Stopping criteria);
- вывод различной дополнительной информации по ходу работы генетического алгоритма (вкладка Plot Functions);
- вывод результатов работы алгоритма в виде новой функции (вкладка Output function);
- задание набора информации для вывода в командное окно (вкладка Display to command window);
- способ вычисления значений оптимизированной и ограничивающей функций (вкладка User function evaluation).

Например, во вкладке критерия остановки (*Stopping criteria*) указываются ситуации, при которых алгоритм совершает остановку. При этом, настраиваемыми являются следующие параметры:

- максимальное число поколений, после превышения которого произойдет остановка;
- лимит времени на работу алгоритма;
- количество мало отличающихся поколений, по прошествии которых алгоритм остановится;
- минимальные значения изменений оптимизируемой функции, при которых алгоритм продолжит работу.

Эволюционный вывод находит широкое применение в различных областях.

В 1994 году Э.Кин из университета Саутхемптона использовал алгоритм генетического программирования в дизайне космических кораблей. Взяв за основу модель опоры космической станции, спроектированной в NASA, Кин перевел ее параметры в бинарные строчки-"хромосомы", сделал приличное количество копий и запустил кибер-секс на 11 компьютерах. "Хромосомы" могли обмениваться "генами" и производить "потомство", после чего компьютер выбирал лучших - и они продолжали развиваться. После смены 15 поколений, включавших 4.500 вариантов дизайна, получилась модель, превосходящая по тестам тот вариант, что разработали люди.

В NASA тоже сделали выводы: аналогичный алгоритм генетического программирования был использован при разработке антенны для серии микроспутников, которые были запущены в конце 2004 г. для исследования магнитосферы Земли. Помимо успехов на тестах, разработчиков в обоих случаях поразили органичные формы выращенных конструкций: космическая опора

оказалась похожей на берцовую кость, а антенна для спутников NASA - на рог оленя.

Джон Коза из Стэнфорда развил технику генетического программирования. Результатом эволюции здесь становятся не отдельные числовые параметры, а целые программы, которые являются виртуальными аналогами реальных устройств. Выращивая в таком виртуальном бульоне усилители, фильтры, контроллеры и другие электронные схемы, компания *Genetic Programming* смогла повторить 15 человеческих изобретений. Шесть из них были запатентованы после 2000 г., т.е. представляют собой самые передовые достижения. Один из контроллеров, выведенных в *GP*, даже превосходит аналогичную человеческую разработку – и, как полагает сам Коза, в данном случае можно говорить о первом патенте на изобретение, сделанное машиной.

Глава компании *Imagination Engines* С. Тайлер утверждает, что его программа *Creativity Machine* уже давным-давно изобрела за него кучу вещей, начиная от зубной щетки *Oral-B CrossAction* и кончая красивыми названиями типа *Synaptrix*. Трудно сказать, насколько все эти машинные творения самостоятельны - однако среди клиентов Тайлера числятся такие компании, как *Gillette*, *Boeing* и *General Electric*.

Плоды электронной эволюции можно найти в самых разных сферах: от двигателя самолета *Boeing 777* до новых антибиотиков. По мнению Д.Андрэ, одного из соратников Козы, "в ближайшие 5 лет генетическое программирование создаст больше продуктов, чем продается и используется в настоящее время". Очевидно, что при этом наибольший приоритет получают те, кто контролирует "вычислительный океан". Пока это доступно немногим: компания *Genetic Programming* смогла сделать свои изобретения благодаря самодельному суперкомпьютеру из тысячи 350-мегагерцовых "Пентиумов". Однако Дж.Коза считает, что 50-гигагерцовые компьютеры станут обычным делом, так что стоящие патентов изобретения можно будет выращивать "в домашних условиях".

До сих пор речь шла о виртуальных моделях, материализация которых невозможна без участия людей. Однако с каждым годом цифровая эволюция становится все более независимой. В 2002 г. в британском центре *Magna* открылся павильон *Live Robots*, где бились за выживание 12 роботов двух видов - "гелиофаги" и "хищники". Первые сами добывали энергию через солнечные батареи. Вторые не могли пользоваться светом, поэтому охотились на "гелиофагов" и заряжались от них. Главным "чувством" роботов являлся "инфразвуковой запах". Те, кто выжил, загружали свои "гены" в роботов следующего поколения. Один из роботов в ходе этого эксперимента поумнел настолько, что убежал и был пойман на парковке, где на него наехал автомобилист.

Еще дальше пошли в *Brandeis University*, где была создана программа *Golem*, которая сама конструировала роботов. В программу были заложено описание деталей (трубки-кости, моторчики-мышцы и искусственные нейроны), а также механизм мутаций и функция пригодности для "отсеивания" неудачников - тех, кто не научился двигаться. После того, как *Golem* выводил работоспособную виртуальную модель (600 поколений за несколько дней), ему позволяли создать реальную версию победителя с помощью трехмерного принтера. Результатом эксперимента стали три ползающих робота и много интересных наблюдений. В частности, симметрия некоторых моделей никак не была прописана в условиях, однако появилась в ходе эволюции как полезная черта, позволяющая двигаться прямо.

Очевидно, что такие разумные системы, работающие с минимальным участием людей и без использования какой-либо человеческой логики, принесут множество сюрпризов. Как это случилось в Университете Сассекса, где ученые использовали эволюционный метод для выращивания электронной схемы осциллятора. Получившийся в итоге набор транзисторов выдавал желаемый результат - регулярно повторяющийся сигнал. Однако при ближайшем рассмотрении выращенная схема оказалась не осциллятором, а радиоприемником: устройство не производило собственных колебаний, зато ловило сигнал работающего рядом компьютера и выдавало его за "свой". Легко представить, что на следующем витке цифровая эволюция выведет устройство, которое воспользуется не соседним компьютером, а человеком - например, в качестве батарейки, как в "Матрице".

Тем не менее, "матричное" будущее, вероятно, не грозит нам по крайней мере в ближайшие 20 лет: автономные системы выращивания роботов требуют развития целого ряда технологий, которые пока находятся в зародышевом состоянии. С другой стороны, как полагает Дж. Коза, генетическое программирование уже привело к рождению машинного интеллекта, способного тягаться с человеческим. Чтобы это проверить, достаточно использовать в качестве теста не умение вести диалог (тест Тьюринга), а способность делать изобретения, достойные патентов: ведь для их выдачи Патентное бюро не требует физического присутствия изобретателя!

Не исключено, что самый умный искусственный разум родится в процессе борьбы с спамерами, рассылающими рекламу по электронной почте. Один из таких зародышей - фильтр компании *Cloudmark*: система ежедневно анализирует 130 миллионов сообщений, отслеживая 98% мусорных писем на основе "нездоровых" комбинаций почтовых "генов" (300 специфических характеристик письма). В случае ошибочного пропуска мусорного письма программа запускает в своем "зоопарке" процесс мутации до тех пор, пока не будет выведена такая комбинация

"генов", которая встречается в данном письме. В следующий раз при появлении подобных "тварей" машина отловит и их.

С развитием беспроводных технологий спруты машинных разумов грозят достать нас повсюду. Несколько лет назад в *Hewlett-Packard* создали компьютерного ди-джея *HPDJ*, который пишет музыку, отслеживая настроение людей. Каждому посетителю клуба выдается браслет-датчик, фиксирующий пульс человека и его местонахождение в зале. Эти данные *HPDJ* использует для выращивания новых мелодий. Вначале машина вносит в музыку случайные мутации - а затем отслеживает реакции и выбирает те изменения, которые пришлись людям по вкусу.

В Гарварде Г. Золтман развивает новую рыночную технологию под названием "нейромаркетинг". Идея состоит в сканировании мозгов людей во время демонстрации рекламы, что позволяет отслеживать воздействие рекламных образов на человека, а затем конструировать образы, воздействующие наиболее сильно. Клиенты Золтмана - *Proctor&Gamble* и *Coca Cola*.

Интересный вопрос: что является продуктом эволюции в этих экспериментах? Ведь "твари", которые подвергаются мутациям и тестам на выживание - это не люди и не машины, а письма, мелодии и логотипы.

Ранее, в середине 70-х подобные размышления привели оксфордского профессора Ричарда Докинза к идее о том, что основными единицами биологической эволюции являются не виды, а гены. Это они стараются сохранить себя в наибольшем числе копий, а все живые существа - лишь их носители, своеобразные "дискеты для генов". Однако теперь у эгоистичных генов появились серьезные конкуренты - **мемы**: "Надо ли нам отправляться в далекие миры в поисках репликаторов иного типа и, следовательно, иных типов эволюции? Мне кажется, что репликатор нового типа недавно возник именно на нашей планете. Он пока еще находится в детском возрасте, барахтается в своем первичном бульоне, но эволюционирует с такой скоростью, что оставляет старый добрый ген далеко позади" (Р. Докинз). Мемы - аналоги генов в человеческой культуре [Родзин С.И. Метрика и алгоритмы меметики // Вестник РГУПС. 2013. №4. С.59-67. <http://elibrary.ru/download/72028481.pdf>]. Именно эти сущности, вырастающие из навязчивых идей, мелодий, картинок и прочих психических вирусов, становятся основным предметом эволюции на нынешнем ее витке. Эволюция вначале создала человеческий мозг, потом к нему добавились книги, радио, телевидение, компьютер и, наконец, Интернет с живущими в нем программами, которые, возможно, смогут самостоятельно размножаться, оставляя человеку только самую грязную работу - менять сгоревшие кулеры и платить на электричество...

Вопросы для собеседования

1. Какие междисциплинарные подходы используются в исследованиях ИИ?
2. Какие области исследований в ИИ?
3. В чём состоит различие представлений о «слабом» и «сильном» ИИ?
4. Каковы особенности естественного интеллекта?
5. Каковы параметры мозга как системы обработки информации?
6. Сформулируйте основные постулаты кибернетики «чёрного ящика» и нейроинформатики.
7. Сформулируйте гипотезы Ньюэлла – Саймона.
8. В чём заключается тест Тьюринга?
9. Что нужно для создания интеллектуальной системы?
10. Как выглядит «пирамида» интеллектуальных задач и проблем?
11. Объясните различия в понятиях «данные», «информация», «знания»?
12. Чем характеризуется информация?
13. Что такое онтология?
14. Как классифицируются знания?
15. Как в программировании происходила эволюция отдельных фрагментов программ в самостоятельные системы?
16. Охарактеризуйте свойства знаний.
17. Что такое понятие, сущность, класс сущностей?
18. Что такое интенционал и экстенционал понятия?
19. Что собой представляет имя понятия?
20. Опишите треугольник Фреге.
21. Что такое знак?
22. Перечислите основные способы определения понятий.
23. Что собой представляет родовидовая классификация понятий?
24. Что собой представляет фасетная классификация понятий?
25. Что собой представляет иерархическая классификация понятий?
26. В чём различие между партитивным и родовидовым понятием?
27. Перечислите основные виды отношений между понятиями.
28. Как вводится метрика «расстояний» между понятиями?
29. Как определить меру сходства/различия между парой понятий?
30. Что такое алгебраическая система Мальцева?
31. Опишите общую классификацию моделей представления знаний.
32. Приведите пример семантической сети.
33. Опишите модель продукционных правил.

34. Что собой представляет фреймовая модель?
35. Приведите примеры декларативно-процедурных моделей.
36. Как различаются методы поиска решений по способу обхода путей на графе?
37. Как классифицируются задачи поиска по дереву решений?
38. Как, согласно К.Шеннону, определяется объём информации, содержащейся в сообщении?
39. Какая формула является общезначимой?
40. Как проверяется общезначимость формулы?
41. Какой метод лежит в основе языка Пролог?
42. Что представляет собой каноническая система?
43. Какими свойствами обладают канонические системы?
44. В чём заключаются прямой и обратный логические выводы?
45. Что такое формальная система?
46. Что понимается под эвристическим поиском?
47. В чём состоит теорема Вольперта – Макрида?
48. В чём состоит идея алгоритма «первый лучший»?
49. В чём состоит идея алгоритма ID3?
50. В чём состоит идея абдукции?
51. Сформулируйте теорему Байеса.
52. Что такое нечёткое правило?
53. Что такое нечёткая функция принадлежности?
54. Как выглядит типовая модель нечёткого вывода?
55. Как выглядит базовая схема эволюционного алгоритма вывода?
56. Приведите пример нейросетевого вывода.
57. Приведите общую постановку задачи вывода знаний.

Контрольные вопросы

1. Искусственный интеллект – это одно из основных направлений информатики, цель которого – разработка программно-аппаратных средств, позволяющих: а) решать трудные вычислительные задачи, б) имитировать мышление человека, в) доказывать теоремы, г) создавать разумные системы, д) понимать естественный язык, е) управлять компьютерными системами.

2. Признаками интеллектуальных систем является способность: а) к самообучению, б) рассуждениям, в) передаче информации, г) целеполаганию, д) адаптации к ситуации, е) хранению информации.

3. Задачи, которые изучаются в рамках инженерии знаний как направления в ИИ: а) подготовка инженеров по знаниям, б) управление информацией, в) извлечение знаний экспертов, г) представление знаний, д) распознавание образов.

4. Известные Вам междисциплинарные подходы к изучению ИИ: а) генетический, б) воспитательный в) образовательный, г) информационный, д) макроэкономический.

5. Основными теоретическими проблемами и вопросами ИИ являются: а) создание мощных вычислителей, б) планирование и прогнозирование, в) автоформализация знаний, г) разработка компьютерных методов и алгоритмов, д) понимание естественного языка.

6. Единственный объект, способный мыслить, – это мозг человека. Поэтому любое «мыслящее» устройство должно воспроизводить его структуру. Это основной постулат ... (слово).

7. Имитация некоторых процессов мышления, как одна из основных целей ИИ, предполагает изучение и моделирование: а) процедур ввода/вывода информации, б) механизмов вывода знаний, в) механизмов протекания мышления.

8. Тест «Если человек в ходе диалога не заподозрит в собеседнике компьютер, то машина является думающей» для ответа на вопрос, может ли машина мыслить, предложил: а) Буль, б) Пospelов, в) Розенблатт, г) Тьюринг, д) Чёрч.

9. Программы ИИ отличаются от всех остальных: а) наличием особой структуры; б) языком, на котором они написаны, в) способностью к обучению, г) быстроедействию.

10. Кто разработал первый нейрокомпьютер: а) У. Маккалок, б) М. Минский, в) Ф. Розенблатт, д) нет правильного ответа.

11. Что из нижеперечисленного относится к персептрону: а) однослойная нейронная сеть, б) нейронная сеть прямого распространения, в) многослойная нейронная сеть, г) нейронная сеть с обратными связями, д) создан Ф. Розенблаттом, е) создан У. Маккалоком и В. Питтом.

12. Какую нейронную сеть обучают с помощью алгоритма обратного распространения ошибки: а) однослойную нейронную сеть, б) многослойную нейронную сеть прямого распространения, в) многослойную нейронную сеть с обратными связями, г) нет правильного ответа.

13. Охарактеризуйте понятие «данные»: а) всегда пассивны, б) всегда активны, в) иногда активны, иногда пассивны.

14. Информацию подразделяют на: а) процедурную, б) активную, в) декларативную, г) пассивную.

15. Декларативная информация о веществе: а) в данных, б) процедурах, в) функциях.

16. Процедурная информация о веществе: а) в данных, б) программах, в) логической модели.

17. Вероятность некоторого сообщения X равна $p(X)$. Если это сообщение будет получено, то его информационный объём равен ... бит (*формула*).

18. Знания – это: а) обработанные данные, б) закономерности, полученные теоретически или опытным путём, в) факты, объекты, их свойства.

19. Охарактеризуйте понятие «знания»: а) всегда пассивны, б) всегда активны, в) могут быть как пассивными, так и активными.

20. Особенности знаний являются: а) связность, б) структурированность, в) корректность, г) непротиворечивость, д) внутренняя интерпретируемость.

21. При изучении некоторого объекта понятия появляются в следующем порядке: а) знания-информация-данные, б) данные-информация-знания, в) информация-данные-знания, г) знания-данные-информация.

22. Словарь, основанный на лексике естественного языка и отражающий семантические отношения между понятиями, называется ... (*слово*).

23. Компоненты продукционной системы: а) библиотека, б) база правил, в) интерпретатор, г) процессор, д) рабочая память, е) оперативная память.

24. Для выполнения продукционного правила необходимо: а) выполнение хотя бы одного из его условий, б) выполнение хотя бы одного из его заключений, в) выполнение всех его условий, г) выполнение всех его заключений.

25. Условное утверждение в левой части продукционного правила, которое должно выполняться в рабочей памяти для того, чтобы были выполнены соответствующие действия в правой части правила, называется ... (*слово*).

26. Заключение или действие в правой части продукционного правила, которое должно быть совершено над базой данных в случае выполнения соответствующих условий в левой части правила, называется ... (*слово*).

27. Основные проблемы при обслуживании системы продукций: а) корректность, б) непротиворечивость, в) эффективность вывода.

28. Максимальный размер базы знаний в продукционной модели не превышает: а) 10 записей, б) 100 записей, в) 1000 записей, г) 65 534 записей.

29. Правильная последовательность работы интерпретатора продукционных правил (например, в языке CLIPS): а) разрешить конфликт, б) применить выбранное правило, в) сопоставить условные части правил и элементы рабочей памяти.

30. Соответствие между используемыми в продукционных системах механизмами разрешения конфликтов (1) стратегия сложности, 2) стратегия

глубины, 3) МЕА-стратегия) и принципами их разрешения (а) новизна, б) разнообразие, в) специфика).

31.Особенностями знаний являются: а) связность, б) структурированность, в) корректность, г) непротиворечивость, д) внутренняя интерпретируемость.

32.Требования, предъявляемые к терминам: а) краткость, б) общезначимость, в) однозначность, г) полнота, д) стилистическая нейтральность.

33.Дано следующее определение: «Сутки – промежуток времени, состоящий из 24 часов». Укажите известный Вам способ формулировки данного определения: а) интенциональный, б) номинальный, в) остенсивный, г) экстенциональный.

34.Требования, предъявляемые к определениям: а) неизбыточность, б) общезначимость, в) отсутствие «порочного круга», г) полнота, д) соразмерность.

35.Понятия «лётчик» и «космонавт» находятся в отношении: а) равнозначность, б) подчинение, в) другое.

36.Понятия «треугольник» и «прямоугольный треугольник» находятся в отношении: а) равнозначность, б) подчинение, в) толерантность.

37.Интенционал понятия – это: а) внешнее содержание понятия, б) описание понятия через его свойства, в) внутреннее содержание понятия.

38.Понятие «персональный компьютер», заданное через интенционал, – это ... (задайте в форме множества).

39.Соответствие между видом знаний и признаком их классификации:

ВИД ЗНАНИЙ

ПРИЗНАК КЛАССИФИКАЦИИ ЗНАНИЙ

1) Образные и знаковые.

А) Степень соответствия реальности.

2) Формальные и содержательные.

Б) Способ программирования.

3) Достоверные и правдоподобные.

В) Язык описания.

4) Процедурные и декларативные.

Г) Степень общности.

Д) Форма представления.

40.Треугольник Фреге – это отношение, связывающее: а) имя, б) фамилию, в) отчество, г) понятие, д) идею, е) представление о понятии.

41.Равнозначные понятия: а) круг – окружность, б) эллипс – круг, в) булева алгебра – алгебра логики, г) квадрат – равносторонний прямоугольник.

42.Экстенционал понятия – это: а) расширение данного понятия, б) набор близких по смыслу понятий, в) класс конкретных сущностей, объединяемых в понятие через перечисление.

43.Способами оценки различия/сходства понятий являются: а) метод «от противного», б) функции сходства/различия, в) метрика расстояния, г) алгоритм Квайна.

44.Словарь, основанный на лексике естественного языка и отражающий семантические отношения между понятиями, называется ... (*слово*).

45.Основной компонент интеллектуальной системы, множество правил, эвристик и процедур, организованных различными моделями представления знаний, называются ... (*словосочетание*).

46.Что понимается под представлением знаний: а) кодирование информации на каком-либо формальном языке, б) знания, представленные в программе на языке C++, в) знания, представленные в учебниках по математике, г) моделирование знаний специалистов-экспертов.

47.Алгебра включает множество: а) вершин, б) констант, в) рёбер, г) переменных, д) предикатов, е) функций.

48.Реляционная система включает множество: а) вершин, б) констант, в) рёбер, г) переменных, д) предикатов, е) функций.

49.Модели представления знаний: а) база данных, б) продукционные правила, в) семантическая сеть, г) фреймы.

50.Языки представления знаний: а) бинарные отношения, б) логика, в) продукционные правила, г) семантические сети, д) фреймы.

51.Процедурные модели: а) автоматы Мили или Мура, б) фреймы, в) продукционные системы, г) комбинационные схемы без памяти.

52.Компоненты продукционной системы: а) библиотека, б) база правил, в) интерпретатор, г) процессор, д) рабочая память, е) оперативная память.

53.Ядро продукционного правила вывода является недетерминированным, если при выполнимости левой (условной) части продукции: а) правая часть может выполняться или не выполняться, б) правая часть должна обязательно выполняться.

54.Необходимые условия выполнения продукционного правила: а) выполнение хотя бы одного из его условий, б) выполнение хотя бы одного из его заключений, в) выполнение всех его условий, г) выполнение всех его заключений.

55.Условное утверждение в левой части продукционного правила, которое должно выполняться в рабочей памяти для того, чтобы были выполнены соответствующие действия в правой части правила, называется ... (*слово*).

56.Заключение или действие в правой части продукционного правила, которое должно быть совершено над базой данных в случае выполнения соответствующих условий в левой части правила, называется ... (*слово*).

57.Основные проблемы при обслуживании системы продукций: а) обеспечение корректности, б) поддержание непротиворечивости, в) обеспечение эффективности вывода.

58. Максимальный размер базы знаний в продукционной модели не превышает:

а) 10 записей, б) 100 записей, в) 1000 записей, г) 65534 записей.

59. Правильная последовательность работы интерпретатора продукционных правил (например, в языке CLIPS): а) разрешить конфликт, б) применить выбранное правило, в) сопоставить условные части правил и элементы рабочей памяти.

60. Соответствие между используемыми в продукционных системах механизмами разрешения конфликтов: 1) стратегия сложности, 2) стратегия глубины, 3) МЕА-стратегия и принципами их разрешения: а) новизна, б) разнообразие, в) специфика).

61. Фрейм – структура для представления знаний: а) об объектах без чёткой структуры, б) о стереотипных ситуациях, в) о вызываемых объектах.

62. Фрейм, как модель для представления знаний, была предложена: а) Аристотелем, б) Булем, в) Минским, г) Фреге.

63. Фреймы могут объединяться в сети: а) да, б) нет, в) могут, но это бессмысленно.

64. Для фреймов характерно свойство наследования по АКО-связям: а) да, б) нет, в) зависит от контекста.

65. Какой из основных типов отношений семантической сети, представленных ниже, может быть назван как АКО (А - Kind - Of): а) это, б) элемент класса, в) имеет частью, г) принадлежит, д) функциональная связь.

66. Инкапсуляция это: а) бессмысленное слово, б) механизм наследования, в) способ объединения кода и данных в объекте.

67. Основные свойства объектов: а) изоморфизм, б) инкапсуляция, в) полиморфизм, г) наследование.

68. Ограничения в количестве фреймов, используемых при моделировании: а) 256 штук, б) 1000 штук, в) число ограничено объёмом ОЗУ, г) не ограничено.

69. Представление знаний в виде фреймов значительно более эффективно, чем при помощи: а) нечёткой логики, б) продукционных правил, в) семантических сетей, г) эффективность зависит от задачи.

70. Представление знаний в виде фреймов эффективно при решении следующих задач: а) автоматический перевод, б) анализ пространственных сцен, в) проектирование микросхем, г) распознавание текста.

71. Семантическая сеть – это: а) орграф, в котором вершины являются понятиями, а ребра – отношениями, б) иерархическая классификационная структура, в) несколько семантически связанных предложений в тексте.

72. Семантическая сеть, как способ представления знаний, представляет собой ориентированный граф, в котором вершинами являются: а) процедуры, б) понятия, в) слоты, г) действия, д) сигналы, е) отношения, ж) объекты.

73. Рёбрами семантической сети обычно выступают: а) действия б) понятия, б) абстрактные или конкретные объекты, в) отношения.

74. Семантические сети чаще всего применяются в предметных областях: а) с хорошо установленной таксономией, б) с бесконечным числом отношений, в) с неопределённой таксономией.

75. Типичными задачами для применения семантической сети являются: а) анализ естественного языка, б) вычислительные расчёты, в) медицинская диагностика.

76. Основной недостаток семантической сети: а) сложность вывода, б) сложность построения, в) высокие требования к ресурсам памяти.

77. Проверить общезначимость булевой формулы $F = (((x \& y) \rightarrow z) \& (x \rightarrow y)) \rightarrow (x \rightarrow z)$.

78. Проверить тождественную истинность булевой формулы $F = ((x \& y) \rightarrow z) \rightarrow (x \rightarrow (y \rightarrow z))$.

79. Машина вывода - это: а) компьютер, на котором происходит вывод информации, б) совокупность программно-аппаратных средств ИИ, в) программа управления перебором правил.

80. С точки зрения формальной логики содержание мысли не оказывает никакого влияния на правильность рассуждений: а) неверно, б) верно, в) иногда верно, иногда неверно.

81. Всегда истинным является: а) категорическое умозаключение; б) логическое умозаключение; в) дедуктивное умозаключение, г) индуктивное умозаключение.

82. Дедуктивное умозаключение: а) всегда является истинным, если не были нарушены правила вывода, б) не всегда истинно, в) всегда ложно.

83. Если известно, что импликация $P \rightarrow Q$ истинна и заключение Q ложно, то отсюда можно вывести: а) P , б) $\neg P$, в) другое.

84. Предложение исчисления высказываний, записываемое в виде выражения « Q , если $P_1 \& P_2 \& \dots \& P_k$, – истина», которое является основной конструкцией языка Пролог, есть: а) резольвента, б) закон де Моргана, в) дизъюнкт Хорна, г) консеквент.

85. Формула, истинная при всех значениях переменных: а) выводима, б) выполнима, в) общезначима, г) противоречива.

86. Формула, истинная при некоторых значениях переменных: а) выводима, б) выполнима, в) общезначима, г) противоречива.

87. Формула, ложная при всех значениях переменных: а) выводима, б) выполнима, в) общезначима, г) противоречива.

88. Формальная система обладает полнотой, если в ней всегда существует такая истинная формула, что ни она сама, ни её отрицание не являются выводимыми в этой системе: а) верно, б) неверно, в) иногда верно, иногда неверно.

89. В формальной системе формула, утверждающая непротиворечивость этой системы, не является выводимой в ней: а) верно, б) неверно, в) иногда верно, иногда неверно.

90. Если $p(H|F, m, A)$ – вероятность получения с помощью A -алгоритма m различных решений, имеющих вид гистограммы H , то для любой пары эвристических алгоритмов A_1 и A_2 на множестве различных целевых функций F справедливо следующее соотношение: ... (формула).

91. Индуктивное умозаключение: а) всегда истинно; б) всегда ложно, в) может быть как истинным, так и ошибочным.

92. Индуктивное умозаключение опирается: а) на логический закон, б) факты, в) ни на что не опирается.

93. Обобщение – переход от частного к общему – характерно для: а) абдукции, б) дедукции, в) индукции.

94. Тип вывода по цепочке рассуждений «Поскольку это животное ест бананы, висит обычно на хвосте и громко кричит, то, возможно, это обезьяна»: а) прямая цепочка, б) обратная цепочка, в) силлогизм.

95. Тип вывода по цепочке рассуждений «Если это крокодил, то у него должен быть хвост и сам он должен быть зелёный»: а) прямая цепочка, б) обратная цепочка, в) силлогизм.

96. Продукционная модель вывода и язык CLIPS подходят для решения следующих задач: а) оценка материального ущерба, б) расчёт себестоимости продукции, в) решение дифференциальных уравнений, г) диагноз в медицине, д) прогноз курса валют.

97. Известны: $p(H) = 0,4$ – априорная вероятность истинности гипотезы H ; $p(E|H) = 0,7$ – вероятность факта E при условии, что H верна; $p(E|\neg H) = 0,2$ – вероятность факта E при условии, что H не верна. Вычислить по формуле Байеса апостериорную вероятность $p(H|E)$ истинности гипотезы H при условии, что получен факт E ... (формула).

98. Теория нечёткой логики разработана: а) Аристотелем, б) Булем, в) Заде, г) Тьюрингом.

99. Известные Вам НЕ-факторы знаний: а) незамкнутость, б) нелинейность, в) неопределённость, г) неполнота, д) неточность, е) неустойчивость, ж) нечёткость.

100. Примерами нечётких понятий являются: а) тинейджер, б) молодой человек, в) высокий человек, г) неизвестный солдат.

101. Процедура преобразования базовой переменной в нечёткую лингвистическую переменную, характеризующуюся функцией принадлежности, называется ... (слово).

102. Основные формы фазирования функции принадлежности нечётких множеств: а) круг, б) колокол, в) трапеция, г) треугольник, д) квадрат.

103. В нечёткой логике степень истинности конъюнкции нескольких высказываний определяется: а) наиболее правдоподобным, б) наименее правдоподобным, в) средним значением.

104. Значения выходной переменной при нечётком выводе по Суджено задаются: а) нечёткими термами, б) как линейная комбинация входов, в) как нелинейная комбинация входов.

105. В основе языка Пролог лежит математическая модель, основанная на методе: а) эволюций, б) революций, в) резолюций, г) инволюций.

106. Проверить общезначимость формулы $F = ((A+B) \rightarrow ((\neg A \& B) + (A \& \neg B)))$.

107. Перед розыгрышем футбольного кубка между командами «Спартак», «Динамо», «Ростов», «Зенит» были высказаны следующие прогнозы: победит «Ростов»; победит «Динамо»; «Зенит» не победит; «Ростов» не победит. Известно, что подтвердился только один прогноз. Кубок выиграла команда ... (слово).

108. Высказывания «бесплатный проезд» и «безбилетный проезд» выражают: а) разные понятия, б) одни и те же понятия.

109. Кто считается «отцом» генетических алгоритмов: а) Д. Голдберг, б) Д. Холланд, в) К. Де Йонг, г) нет правильного ответа.

110. Какие понятия относятся к генетическим алгоритмам: а) особь, б) фенотип, в) ген, г) ДНК, д) нейрон, е) функция активации.

111. Какие бывают операторы генетического алгоритма: а) кроссинговер, б) скрещивание, в) транслитерация, г) транслокация, д) мутация, е) конверсия.

112. Какой оператор применен к особи (0001000 \rightarrow 0000000): а) инверсии, б) кроссинговер, в) скрещивания, г) нет правильного ответа.

Учебно-методическое обеспечение курса

Основная литература

1. Башмаков А.И., Башмаков И.А. Интеллектуальные информационные технологии: Уч. пособие. М.: Изд-во МГТУ им. Баумана, 2005.
2. Вагин В.Н. и др. Достоверный и правдоподобный вывод в интеллектуальных системах: Уч. пособие. М.: Физматлит, 2008.
3. Люгер Дж. Ф. Искусственный интеллект: стратегия и методы решения сложных проблем: Уч. пособие. М.: Вильямс, 2005.
4. Родзин С.И., Родзина О.Н. Модели представления знаний: Уч. пособие. Таганрог: Изд-во ЮФУ, 2014. – 148 с.
http://sfedu.ru/pls/rsu/umr_main.umr_show?p_startpage=2&p_umr_name=&p_umr_autho r=&p_umrc_id=&p_umrr_id=&p_per_id=-3001497
5. Системы искусственного интеллекта. Практический курс: учебное пособие / под ред. И.Ф. Астаховой. - М.: БИНОМ. Лаборатория знаний, 2008.

Дополнительная литература

6. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. СПб.: Питер, 2000. 384 с.
7. Джексон П. Введение в экспертные системы. М.: Издат. дом «Вильямс», 2001. 624 с.
8. Джонс М.Т. Программирование искусственного интеллекта в приложениях: Пер. с англ. М.: ДМК Пресс, 2006. 312 с.
9. Комарцова Л.Г., Максимов А.В. Нейрокомпьютеры: Учеб. пособие. Изд-во МГТУ им. Н.Э. Баумана, 2004. 400 с.
10. Ландэ Д.В. Поиск знаний в Internet. Профессиональная работа. М.: Издательский дом «Вильямс», 2005.
11. Родзин С.И. и др. Системы искусственного интеллекта. Лабораторный практикум: Уч. пособие. Таганрог: Изд-во ТТИ ЮФУ, 2007. – 128 с.
12. Родзин С.И. и др. Программирование алгоритмов распознавания изображений в интерактивной среде MatLab. Лабораторный практикум, курсовое проектирование: Уч. пособие. Таганрог: Изд-во ТТИ ЮФУ, 2010.
13. Смолин Д.В. Введение в искусственный интеллект. Конспект лекций. М.: Физматлит, 2004. 208 с.
14. Частиков А.П. и др. Разработка экспертных систем. Среда CLIPS. СПб.: БХВ-Петербург, 2003. 608 с.
15. Ярушкина Н.Г. Основы теории нечётких и гибридных систем: Учеб. пособие. М.: Финансы и статистика, 2004. 320 с.

Классические работы, рекомендуемые к ознакомлению

16. Искусственный интеллект. Под ред. Д.А.Поспелова. – М.: Радио и связь, 1990.
17. Нильсон, Н. Принципы искусственного интеллекта. – М.: Радио и связь, 1985.

Периодические издания и Интернет-ресурсы

18. Журнал «Искусственный интеллект и принятие решений» <http://www.aidt.ru/> . В журнале публикуются оригинальные статьи, обзорные работы, которые содержат теоретические результаты, методы, описания прикладных систем, технологий, инструментальных средств, опыта их практического применения в следующих направлениях: автоматизация рассуждений, интеллектуальный анализ данных и машинное обучение, представление знаний и инженерия знаний, обработка естественного языка и интеллектуальный поиск информации, моделирование поведения и планирование, моделирование образного мышления, когнитивное моделирование, мягкие вычисления, оптимальный выбор, многокритериальный анализ решений, коллективные решения, интеллектуальные системы и технологии, системы поддержки принятия решений, интеллектуальные роботы.

19. Applied Artificial Intelligence
<http://www.tandfonline.com/action/journalInformation?journalCode=uaai20&>

20. Autonomous Agents and Multi-Agent Systems
<http://www.springer.com/computer/ai/journal/10458>

21. Computational Intelligence (journal)
<http://eu.wiley.com/WileyCDA/WileyTitle/productCd-COIN.html>

22. Data Mining and Knowledge Discovery
<http://www.springer.com/computer/database+management+&+information+retrieval/journal/10618>

23. Electronic Letters on Computer Vision and Image Analysis
<http://elcvia.cvc.uab.es/>

24. Evolutionary Computation (journal) <http://www.mitpressjournals.org/loi/evco>

25. Fuzzy Sets and Systems <http://www.sciencedirect.com/science/journal/01650114>

26. IEEE Intelligent Systems
<http://www.computer.org/portal/web/computingnow/intelligentsystems>

27. IEEE Transactions on Evolutionary Computation <http://cis.ieee.org/ieee-transactions-on-evolutionary-computation.html>

28. IEEE Transactions on Pattern Analysis and Machine Intelligence
<http://www.computer.org/portal/web/tpami>

29. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems
<http://www.worldscientific.com/worldscinet/ijufks>

30. Journal of Artificial Intelligence Research <http://www.jair.org/>

31. Journal of Experimental and Theoretical Artificial Intelligence
<http://www.tandfonline.com/action/journalInformation?journalCode=teta20>

32. Machine Learning <http://www.springer.com/computer/ai/journal/10994>

33. Neurocomputing <http://www.journals.elsevier.com/neurocomputing>

34. Pattern Recognition Letters <http://www.journals.elsevier.com/pattern-recognition-letters/>

35. Andras Kornai. Mathematical Linguistics. Springer. 2010. The book introduces the mathematical foundations of linguistics to computer scientists, engineers, and mathematicians. The book presents linguistics as a cumulative body of knowledge from the ground up: no prior knowledge of linguistics is assumed.

36. Artificial Intelligence A Modern Approach - Stuart J. Russell , Peter Norvig.pdf

37. Artificial Intelligence and Soft Computing Behavioral and Cognitive Modeling of

the Human Brain - Konar Amit.pdf

38. Data Mining with Computational Intelligence - Lipo Wang , Xiuju Fu.pdf

Глоссарий

Абдукция – правдоподобный вывод от частного к частному.

Адаптация – способность живого организма или технической системы изменять своё состояние и поведение (параметры, структуру, алгоритм функционирования) в зависимости от изменения условий внешней среды путём накапливания и использования информации о ней.

Адаптивный резонанс – свойство ART-нейросетей (С. Гроссберг) обучаться запоминанию новых образов, не теряя при этом ранее накопленную информацию.

Аксон – выходное окончание биологического нейрона, простая длинная ветвь нейрона, которая передаёт выходной сигнал. В искусственных нейросетях аксон моделируется как выход системы.

Антецедент – условное утверждение в левой части продукционного правила, которое должно выполняться в базе данных (рабочей памяти) для того, чтобы были выполнены соответствующие действия в правой части правила.

Аппроксимация функции – определение по экспериментальным данным функции, наилучшим образом приближающейся к неизвестной зависимости и удовлетворяющей определённым критериям.

Ассоциативные правила – алгоритмы поиска закономерностей между связанными событиями. Алгоритм Apriori – эффективный масштабируемый алгоритм, позволяющий решить задачу в базе данных больших размеров (гига- и терабайты) за приемлемое время.

База данных – 1) в интеллектуальных системах это набор фактов, утверждений и заключений, используемых при сопоставлении с правилами; 2) при управлении информацией это совокупность связанных файлов, таблиц, отношений и т.п., объединённых в систему хранения данных с оперативным доступом к информации из приложений под управление СУБД.

База знаний – основной компонент интеллектуальной системы, содержащий экспертные знания об определённой предметной области. Собрание правил, эвристик и процедур, организованных различными моделями представления знаний.

Вывод – получение новых информационных единиц из ранее известных. Частным случаем является логический вывод.

Гибридная интеллектуальная система – система, которая сочетает, по крайней мере, две интеллектуальные технологии. Например, «мягкие вычисления» являются гибридной технологией, объединяющей нейросети, нечёткую логику, эволюционные алгоритмы, интеллектуальный анализ данных и т.д.

Гипотеза Ньюэлла. Необходимое и достаточное условие интеллектуальности системы – универсальность формальных манипуляций над конкретными символами. Иными словами, достаточно прочесть строку символов, разделить её на компоненты и переупорядочить, добавив или удалив какие-то символы без учёта смысла (семантики) логических и математических символов.

Данные – элементарные символьные, цифровые, звуковые или образные описания предметов, объектов, событий, действий или транзакций, которые записаны, классифицированы и сохранены, но не организованы для передачи в качестве информации или знаний.

Дерево решений – иерархическая древовидная структура, представляющая собой совокупность решений. Деревья решений разбивают данные на группы на основе значений переменных, в результате чего возникает иерархия правил «ЕСЛИ (антецедент) ТО (консеквент)», которые классифицируют данные. Используются, например, при интеллектуальном анализе текстов.

Дизъюнкт Хорна – предложение исчисления высказываний, записываемое в виде выражения «Q, если $P_1 \& P_2 \& \dots \& P_k$ – истина»; он имеет вид $\neg P_1 \vee \neg P_2 \vee \dots \vee \neg P_k \vee Q$ и является основной конструкцией языка Пролог.

Зависимость – в искусственном интеллекте связь между антецедентами и консеквентами, создаваемая в результате применения правила вывода. Способ, которым решение выводится из исходных данных и предшествующих решений.

Закон Парето – первые 20% усилий дают 80% желаемого результата. Необходимо только найти требуемые ресурсы и реализовать их. 80% работы выполняет 20% людей. Следствие закона Парето: если система имеет 99% необходимых возможностей и её создали за 10 человеко-лет, то на практике для доведения её до уровня 100% потребуется ещё не менее 10 человеко-лет.

Знак – информационная единица, структура которой представляет собой треугольник Фреге.

Знания – 1) в узком смысле это обработанная информация, хранимая в *базе знаний* и отражающая убеждения специалистов (экспертов) в определённой предметной области, а также образующая целостное описание некоторой проблемы с доступной степенью детализации; 2) в широком смысле концепция знаний объединяет черты процедурной и декларативной информации и трактуется как обоснованное истинное убеждение, проверенный практикой результат познания действительности. Отличительные характеристики знаний: внутренняя интерпретируемость, структурированность, связность, семантическое пространство с метрикой, активность. Метазнания – знания о знаниях.

ИИ-программирование – разработка инструментального программного обеспечения для решения задач ИИ: языков программирования, ориентированных на особенности задач ИИ, интеллектуальных вспомогательных средств, языков *представления знаний* и манипулирования ими, *пустые экспертные системы* и оболочки и другие инструментальные средства.

Имитация интеллектуального поведения – воспроизведение процедур формирования целенаправленного поведения человека и животных во внешнем мире в зависимости от возникающих в нём ситуаций. Для имитации разрабатываются специальные модели и методы планирования деятельности. Она также широко используется в интеллектуальных роботах.

Индукция – метод перехода от частных наблюдений к общей закономерности, которой удовлетворяют все частные наблюдения.

Инженер по знаниям – специально подготовленный системный аналитик по инженерии знаний, который «извлекает» экспертные знания, встраивает знания в интеллектуальную систему, устанавливает методы рассуждений, выбирает необходимый программный инструментарий, проектирует и тестирует систему.

Интеллектуальная информационная технология – технология обработки информации и решения задач с помощью ЭВМ, опирающаяся на достижения ИИ. Основной идеей, используемой в технологии, является автоматизация процедуры построения программы, интересующей пользователя, на основе введённого им в систему описания постановки задачи на привычном для него профессиональном языке. Для реализации данной технологии необходимо, чтобы ЭВМ обладала интеллектуальным интерфейсом, базой знаний и механизмом вывода, т.е. была бы интеллектуальной системой. Другой её чертой является распределённый способ решения задачи, когда пользователи, занятые решением общей задачи, общаются между собой через сеть ЭВМ, электронную почту и общую базу данных и знаний.

Интеллектуальная система – система, построенная на основе искусственного интеллекта, способная целеустремлённо, в зависимости от состояния информационных входов, изменять не только параметры функционирования, но также способ своего поведения с учётом как текущего, так и предыдущих состояний системы. Единой классификации интеллектуальных систем нет.

Основными признаками многомерной классификации являются: используемый механизм вывода (нейросети, деревья решений, эволюционные алгоритмы, нечёткая логика, «мягкие вычисления» и т.п.); глубина представления информации (базы данных и знаний, электронные библиотеки, тезаурус, онтология и др.); архитектура (открытая, закрытая); категория программного обеспечения (системное программное обеспечение, инструментальное средство, прикладная система); стадия

существования (демонстрационный прототип, промышленный прототип, коммерческая система и др.).

Иерархическая классификация подразумевает несколько иной перечень интеллектуальных систем, основанных на знаниях: экспертные системы (динамические, статические), их оболочки, средства разработки и основные подсистемы; информационные системы (гипертекстового поиска, мультимедиа, геоинформационные, электронные библиотеки, виртуальной реальности и др.); компьютерные системы с обработкой естественного языка (рубрикация текстов, аннотирование и реферирование, машинный перевод, интеллектуальный анализ текстов, интеллектуальные системы обучения, электронные словари и корректоры текста, семантические сети и др.); системы интеллектуального анализа данных; системы машинного обучения; системы распознавания образов (OCR-системы, синтез изображений, анализ и синтез речи); системы поддержки принятия решений (многокритериальные, управления технологическими процессами в реальном времени, планирования, имитационного моделирования, ситуационного управления и др.); интеллектуальные роботы (техническое зрение, мобильные роботы и др.); многоагентные интеллектуальные системы; интеллектуальные САПР, АСУ, CASE; системы машинного творчества (литература, живопись, графика, игры, доказательство теорем).

Информатика (ср. нем. *Informatik*, фр. *Informatique*, англ. *Computer Science* – компьютерные науки – в США, англ. *Computing Science* – вычислительная наука – в Великобритании) – наука о способах получения, накоплении, хранении, преобразовании, передаче и использовании информации. Информатика не более наука о компьютерах, чем астрономия – наука о телескопах. Согласно тезису Чёрча – Тьюринга, все известные типы вычислительных машин качественно эквивалентны в своих возможностях: любое действие, выполнимое на одной вычислительной машине, также выполнимо и на другой. Тезис иногда преподносят как фундаментальный принцип информатики, обращая особое внимание на машину Тьюринга и машину фон-неймановской архитектуры, поскольку они имеют явное сходство с большинством из ныне действующих компьютеров. В рамках современной информатики учёные изучают также и другие типы машин, не только практически осуществимые (параллельные и квантовые компьютеры), но и сугубо абстрактные математические модели (к примеру, РАМ-машина, которая имеет бесконечное число регистров). Термин «информатика» был впервые введён в Германии К. Штейнбухом в 1957 г. Отдельной наукой информатика была признана лишь в 1970-х годах XX века. С момента своего признания отдельной наукой информатика разработала собственные методы и терминологию. Высшей наградой

за заслуги в области информатики является премия Тьюринга. Днем рождения информатики в нашей стране считается 4 декабря 1948 г., когда Государственный комитет Совета министров СССР зарегистрировал за номером 10475 изобретение И. С. Бруком и Б. И. Рамеевым цифровой электронной вычислительной машины.

Информационная технология – совокупность методов, процессов и программно-технических средств, объединённых в технологическую цепочку, обеспечивающую сбор, хранение, обработку, вывод и распространение информации.

Искусственный интеллект (Artificial Intelligence). Интеллект – это способность к обучению, рассуждениям, рефлексии, целеполаганию, познавательная активность, адаптация к ситуации, формирование обобщений и синтез познавательных процедур. ИИ – это одно из направлений информатики, цель которого – разработка программно-аппаратных средств, позволяющих выполнять действия, требующие человеческого интеллекта, имитировать некоторые виды интеллектуальной деятельности, ставить и решать интеллектуальные задачи, общаясь с компьютером на ограниченном подмножестве естественного языка. Теоретической базой ИИ является работа со знаниями: модели знаний; методы, средства и системы представления знаний и баз знаний; методы извлечения и обработки знаний (поиск, вывод, верификация, систематизация знаний, аргументация и объяснение на основе знаний). Прикладные разработки ведутся по следующим основным направлениям: решение отдельных интеллектуальных задач (компьютерное доказательство теорем, поиск знаний в Интернете, семантический анализ и обработка информации на естественном языке, машинный перевод и реферирование, синтез речи, распознавание образов, моделирование поведения бионических систем, игры и т.д.); интеллектуальное программирование (языки ИИ, языки представления знаний, языки семантической разметки, языки многоагентного взаимодействия и т.п.), создание инструментальных средств для автоматического синтеза программ, интеллектуального интерфейса; проектирование и разработка интеллектуальных систем (экспертные системы, интеллектуальные АСУ, САПР, системы управления знаниями и поддержки принятия решений, интеллектуальные обучающие системы, системы интеллектуального анализа данных и текстов, нейропакеты и интеллектуальные роботы и др.).

Информационный объект включает структуру, контент и контекст. Структура объекта характеризуется набором атрибутов и информационных связей. Контент описывает информационное содержание объекта с помощью понятий и отношений между ними. Контекст в отличие от контента характеризует информационный объект как единое целое и явно не зависит от его содержания.

Когнитология – междисциплинарная теория (искусственный интеллект, психология, лингвистика, философия, нейрофизиология, образование), с помощью которой изучаются процессы восприятия, познания, понимания, мышления, обучения и моделируется работа естественных и искусственных интеллектуальных систем.

Коннекционизм – раздел искусственного интеллекта, связанный с созданием, исследованием и развитием простейших моделей мозга человека, в которых все его функции определяются связями между нейронами.

Консеквент – действие в правой части продукционного правила, которое должно быть совершено над базой данных в случае выполнения соответствующих условий в левой части правила.

Коэффициент уверенности – оценка, предоставляемая экспертной системой, которая указывает на вероятность того, насколько заключение системы является правильным. Также степень уверенности эксперта в истинности заключения, если исходная предпосылка истинна. Используется в теории уверенности (Буханан, Шортлифф) для оценки уровня доверия к гипотезе.

Лингвистическая переменная. Её значениями являются нечёткие подмножества слов или предложений на естественном или искусственном языке. Задаётся набором $\{X, T(X), U, G, M\}$, где X – имя переменной; $T(X)$ – термножество лингвистических значений переменной; U – универсальное множество; G – синтаксическое правило для порождения значений переменной X ; M – семантическое правило, которое ставит в соответствие каждому значению переменной её смысл.

Логика высказываний – формальная логическая система рассуждений, в которой заключения выводятся из законов и утверждений в соответствии со строгими правилами логического вывода.

Логика предикатов – логическая система рассуждений, используемая в интеллектуальных системах для обозначения отношений между данными. Логика предикатов является расширением логики высказываний, так как основным объектом в ней является переменное высказывание (предикат), истинность или ложность которого зависит от значений его переменных.

Логический вывод – 1) формальный метод доказательства путём логических рассуждений; 2) в *экспертных системах* новое заключение, полученное по определённым правилам вывода из фактов, имеющих в *базе знаний*. Обычно различают абдуктивный, дедуктивный и индуктивный выводы.

Абдуктивный вывод – процесс формирования наилучшей (минимальной, избыточной) объясняющей гипотезы при заданной теории и факте, предложенном

для объяснения. Например, пусть наблюдается факт F , не следующий из наших знаний о предметной области; если бы гипотеза H была истиной, то F мог бы иметь место; следовательно, можно предположить, что H – истина.

Дедуктивный вывод, являясь единственно достоверным, основан на законах логики и представляет собой процесс рассуждений от общего к частному на основе аксиом. Метод резолюций (Робинсон) реализует дедуктивный вывод в языке Пролог.

Индуктивный вывод – процесс перехода в рассуждениях от единичных фактов к общим выводам. Например, $X\%$ наблюдаемых явлений F есть G ; поэтому приблизительно $X\%$ всех F есть G . Чтобы узнать степень достоверности заключения в некоторой гипотезе G при доводе F , согласно индуктивному выводу, вычисляют условную вероятность $p(G/F)$, т.е. оценивают правдоподобность G при данном доводе F . Индукция служит для оценки гипотез и степени их согласованности с реальностью. Абдукция, дедукция и индукция соответствуют этапам научного исследования: генерация гипотез (инсайт), вывод заключений из гипотез, оценка гипотетических заключений.

Нечёткий логический вывод – процесс рассуждений, основанный на нечёткой логике Л.Заде.

Машинное обучение – адаптивный механизм, который даёт возможность компьютерам обучаться на основе опыта решения задач с использованием примеров и аналогий. Наиболее популярными подходами к машинному обучению являются нейросети и эволюционные алгоритмы.

Модель – объект (реальный или виртуальный), отличный от исходного, но способный заменить его в рамках решаемых задач.

Модель мира – способ отображения в памяти интеллектуальной системы знаний о внешней среде.

Модель обучения – модель, лежащая в основе процесса обучения человека или технического устройства. Различают дескриптивную и нормативную модели. Дескриптивная модель извлекается из описания процесса деятельности, которой человек или система должны обучаться. Это извлечение может происходить разными способами. Наиболее известный из них основан на процедуре обучения на примерах. Нормативная модель задаётся заранее. Часто обучение нормативного типа называют обучением с учителем.

Модель поведения – модель (техническая или программная), воспроизводящая некоторые виды поведения объектов при определённых условиях внешней среды (преодоление препятствий, реакция на внешние воздействия, выбор решений и пр.).

Модель представления знаний – это структуры предметных знаний в интеллектуальной системе с целью облегчения поиска решения задачи. Различают декларативные и процедурные модели знаний. Декларативные модели знаний, явно не содержащие описания выполняемых процедур, обычно подразделяются на продукционные, сетевые (семантические сети) и фреймы. В процедурных моделях семантика непосредственно заложена в описание базы знаний.

Модель «стимул-реакция» – модель поведения, опирающаяся на принцип «чёрного ящика». В ней рассматривается конечное множество стимулов, которые могут восприниматься субъектом или подаваться на выход искусственной системы, и правила соотнесения с этими стимулами выходных реакций субъекта или системы. Внутренние процессы, связывающие стимулы и реакции, не анализируются и не учитываются. Модель находит применение в интеллектуальных системах на уровне воспроизведения простейших поведенческих реакций на раздражения, поступающие из внешней среды.

Неопределённость – свойство интерпретации выражений, когда им приписываются оценки правдоподобия, отличные от оценок абсолютной истины и лжи. Работа с такими выражениями требует специальных средств пересчёта оценок правдоподобия. При логическом выводе, когда имеется неопределённость, используются либо многозначные логики, либо правдоподобные рассуждения.

Неполнота – свойство описания предметной области, заключающееся в том, что это описание не может быть преобразовано в формальную систему. При работе с неполной информацией используются правдоподобные рассуждения.

Неразрешимость алгоритмическая – ситуация, при которой для множества однотипных задач нельзя найти общего алгоритма их решения, хотя для подмножеств этого множества можно построить специфические алгоритмы поиска решений. Существование таких алгоритмически неразрешимых проблем строго доказано.

НЕ-факторы знаний – нечёткость, неопределённость, неточность, неполнота и т.п. в отличие, например, от НЕ-факторов управления (неустойчивость, нелинейность, незамкнутость и т.п.). Эти факторы могут включать неизвестные данные, неточный язык, неявное смысловое содержание и трудности сочетания взглядов разных экспертов.

Нейрон биологический – нервная клетка мозга, состоящая из тела (сомы), дерева входных дендритов и выходного аксона. На соме и дендритах располагаются окончания (синапсы) других нервных клеток.

Нейрон искусственный – элементарный процессор, используемый в узлах нейросети. Описывается математической моделью в виде уравнения

$y = f(g) = f(\sum_i a_i x_i + a_0)$, где y – выходной сигнал нейрона; $f(g)$ – функция активации выхода нейрона; a_i – вес i -го входа; x_i – i -й входной сигнал; a_0 – начальное состояние (возбуждение) нейрона; $i = 1, 2, \dots, n$ – номер входа нейрона; n – число входов. Функция активации может быть пороговой, сигмоидальной, степенной и др.

Нейронная сеть – динамическая система, представляющая совокупность связанных между собой искусственных нейронов, способная генерировать выходную информацию в ответ на входное воздействие.

Нечёткая логика – способы и системы нечёткого вывода и рассуждений в условиях НЕ-факторов (неопределённости и неполноты информации). Теория нечёткой логики, разработанная Л.Заде, в отличие от булевой логики является многозначной, базируется на теории нечётких множеств, в ней используются нечёткие правила.

Нечёткое правило – условное высказывание вида «Если X есть A , то Y есть B », где A и B – нечёткие множества. Правило указывает на связь между нечёткими множествами и служит для построения знаний.

Обработка естественного языка – совокупность процессов анализа текстов на естественном языке, их понимания и синтеза. В наиболее развитых системах обработки естественного-языковых сообщений происходят морфологический, синтаксический и семантический анализы текста, в результате чего выявляется глубинная структура текста, которая переводится во внутреннее представление, используемое в базе знаний интеллектуальной системы. Соотнесение этой структуры с теми знаниями, которые хранятся в системе, позволяет понять смысл исходного текста. При синтезе текстов сначала формируется семантическая структура текста, которая затем наполняется лингвистическими единицами с учётом синтаксиса и морфологии выбранного естественного языка. С обработкой языка связано решение задач машинного перевода, автоматического реферирования, общения с пользователем на ограниченном профессиональном естественном языке и т. п.

Обработка знаний – описание смыслового содержания задач в форме, которая гарантирует их правильную обработку формальными методами.

Обратное распространение ошибки (Back Propagation) – наиболее известный итеративный градиентный алгоритм обучения многослойных нейросетей без обратных связей. Имеет строго доказанную сходимость для дифференциальных уравнений (для бесконечно малых шагов в пространстве весов).

Онтология – база, знания в которой могут читаться, пониматься, отчуждаться от разработчика и/или физически разделяться пользователями. Формальной моделью онтологии является тройка вида $O = \langle X, R, F \rangle$, где X – множество понятий

предметной области, которую представляет онтология; R – множество отношений между понятиями; F – множество функций интерпретации, заданных на понятиях или отношениях онтологии.

Переобучение сети (*Over training, Overfitting*). Если в результате обучения нейронная сеть хорошо распознает примеры из обучающего множества, но не приобретает свойство обобщения, то есть не распознает или плохо распознает любые другие примеры, кроме обучающих, то говорят, что сеть переобучена. Переобучение – это результат чрезмерной подгонки сети к обучающим примерам.

Персептрон – физическая модель зрительного восприятия (Ф. Розенблатт), состоящая из искусственных нейронов. Доказано, что если обучающую последовательность из векторов X и Y предъявить персептрону достаточное число раз, то он в конце концов разделит её (если это в принципе возможно) на два класса.

Поиск – движение в структурированном пространстве от одних узлов этого пространства к другим. Если поиск является целенаправленным, то задаётся множество начальных узлов, с которых поиск может начинаться, и множество конечных узлов, при достижении которых поиск прекращается. Движение по структуре поискового пространства определяется стратегией поиска.

Поиск по дереву решений – совокупность различных методов организованного процесса поиска решений (в глубину, в ширину, с возвратом и т.д.) в пространстве состояний для эффективного нахождения приемлемого решения.

Поиск эвристический – метод поиска на основе эвристических правил, т.е. правил, упрощающих или ограничивающих пространство поиска решения. Эвристики могут быть как количественными, так и качественными, они лежат в основе эвристического программирования как одного из направлений искусственного интеллекта.

Правдоподобный вывод – вывод, при котором каждый шаг сопровождается вычислением оценки достоверности полученного утверждения. Частными случаями правдоподобного вывода являются, например, абдуктивный и индуктивный выводы.

Правила обучения нейросети – коррекция по ошибке, машина Больцмана, дельта-правила Хебба, соревнование.

Правило *modus ponens* – если P – истина и из P следует Q (что равносильно $\neg P \vee Q$), то Q тоже истина, т.е. если имеются две пары P и $\neg P \vee Q$, то Q (резольвента) выводится путём удаления литер P и $\neg P$.

Прогнозирование – определение параметров системы на будущий момент времени по известным значениям параметров в предыдущие моменты времени.

Продукционная (каноническая) система Поста – система, включающая алфавит, из символов которого формируются строки; множество строк, которые

рассматриваются как аксиомы, а также множество грамматических правил манипулирования строками символов (правила порождений). Система должна обладать разрешимостью, непротиворечивостью и полнотой. В продукционной системе знания обычно представляются совокупностью правил вывода вида «ЕСЛИ (условия) ТО (действия)». Различают продукционные системы с прямым и обратным логическими выводами. Прямой вывод означает, что рассуждения строят, отталкиваясь от выполненных условий (антецедент), к заключениям (консеквент), вытекающим из этих условий. Обратный вывод означает, что рассуждения строят, отталкиваясь от заданной цели, к условиям, при которых возможно её достижение. Обычно продукционные системы включают базу правил, состоящую из набора правил вывода (продукций), рабочей памяти (база данных, которая определяет текущее состояние задачи и содержит множество фактов, описание цели и промежуточные результаты) и интерпретатора правил, который осуществляет логический вывод на основании фактов и решает, когда надлежит применить каждое из правил. База правил и база данных образуют базу знаний, а интерпретатор соответствует механизму логического вывода. Одним из языков продукционного программирования является CLIPS, ориентированный на разработку экспертных систем и поддерживающий объектно-ориентированную и процедурную парадигмы программирования.

Распознавание образов – отнесение входного набора данных, представляющего распознаваемый объект, к одному из заранее известных классов.

Резолюция – приём, используемый при достоверном логическом выводе, заключающийся в нахождении двух дизъюнктов, один из которых содержит литеру, а другой – её отрицание. На основании этого сравнения формируется новый дизъюнкт, называемый резольвентой. Порождение новых дизъюнктов является основой метода резолюций, широко применяемого в интеллектуальных системах.

Семантика – 1) один из аспектов семиотики, рассматривающий содержание знаков независимо от того, кто служит адресатом знака; 2) значение отдельных единиц знака; 3) изучение отдельных единиц языка в языковедении, где элементарный объект рассматривается как единое понятие (например, треугольник Фреге) и включает последовательность звуков или знаков, объект действительности и отражение этого объекта в сознании человека.

Семантическая сеть – декларативная модель представления знаний в виде графа, вершины которого соответствуют понятиям или объектам, а дуги – отношениям между объектами.

Семиотика прикладная – направление в искусственном интеллекте, основанное на переходе от формальных систем к семиотическим (знаковым), на

создании семиотических моделей, а также установлении соответствия между базовыми структурами семиотики (треугольник Фреге, фрейм).

Силлогизм – форма логических рассуждений, состоящая из двух посылок и связывающего их заключения.

Система MatLab – система, включающая в себя высокопроизводительный язык, среду программирования, графическую подсистему, библиотеку функций, программный интерфейс и множество полезных пакетов прикладных программ для решения аналитических и инженерных задач. MatLab позволяет пользователю программировать в удобной среде различные задачи искусственного интеллекта, создавать алгоритмы и модели вычислений, анализировать, исследовать и визуализировать символьные и графические данные, разрабатывать различные приложения, включая создание графического интерфейса.

Системы нечёткого вывода Мамдани и Суджено. В системах с выводом по Мамдани база знаний состоит из правил вида «Если x_1 =низкий и x_2 =средний, то y =высокий». В системах с выводом по Суджено база знаний состоит из правил вида «Если x_1 =низкий и x_2 =средний, то $y=a_0+a_1x_1+a_2x_2$ », где a_0 – константа; a_1 , a_2 – весовые коэффициенты, определяющие степень влияния входных переменных на значение выходной переменной. Основное отличие между выводами по Мамдани и по Суджено заключается в способе определения значений выходной переменной в правилах, образующих базу знаний. Значения выходной переменной при выводе по Мамдани задаются нечёткими термами, в системах вывода по Суджено – как линейная комбинация входных переменных.

Системы управления знаниями – совокупность программных и аппаратных средств для организации в базах знаний процедур, связанных с поиском знаний, пополнением базы знаний, её корректировкой и т. п.

Слабоформализованная задача имеет алгоритм решения, качество или достижимость решения которого из-за проблемы комбинаторного взрыва («проклятие размерности») трудно оценить (моделирование в САПР, подготовка производства и т.п.).

Спам – непрошенное рекламное сообщение, сетевой мусор, рассылаемый по электронной почте. Рассылка спама считается нарушением этикета и правил применения компьютерных сетей.

Сценарий – семантическая сеть, в которой в качестве отношений используются причинно-следственные отношения или отношения типа "действие-результат", "действие-цель", "орудие-действие" и т. п.

Тезаурус – это система понятий, семантических отношений и связей между ними в виде множества терминов, определений, ключевых слов и т.п.

Теория Демпстера – Шеффера – метод рассуждений при работе с НЕ-факторами знаний. В отличие от байесовской вероятности используются функции доверия и меры правдоподобия.

Тест Тьюринга – эмпирический тест, идея которого была предложена А. Тьюрингом в статье «Вычислительные машины и разум», опубликованной в 1950 г. в философском журнале «Mind». Тьюринг задался целью определить, может ли машина мыслить. Стандартная интерпретация этого теста звучит следующим образом: «Человек взаимодействует с одним компьютером и одним человеком. На основании ответов на вопросы он должен определить, с кем он разговаривает: с человеком или компьютерной программой. Задача компьютерной программы – ввести человека в заблуждение, заставив сделать неверный выбор».

Все участники теста не видят друг друга. Если судья не может сказать определенно, кто из собеседников является человеком, то считается, что машина прошла тест. Чтобы протестировать именно интеллект машины, а не её возможность распознавать устную речь, беседа ведётся в режиме «только текст», например, с помощью клавиатуры и экрана (компьютера-посредника). Переписка должна производиться через контролируемые промежутки времени, чтобы судья не мог делать заключения исходя из скорости ответов. Во времена Тьюринга компьютеры реагировали медленнее человека. Сейчас это правило необходимо, потому что они реагируют гораздо быстрее, чем человек. По состоянию на 2009 г. ни одна из существующих компьютерных систем не приблизилась к прохождению теста.

Технологическая сингулярность – гипотетический взрывоподобный рост скорости научно-технического прогресса, предположительно следующий из создания искусственного интеллекта и самовоспроизводящихся машин, интеграции человека с вычислительными машинами либо значительного увеличения возможностей человеческого мозга за счёт биотехнологий. По некоторым прогнозам, технологическая сингулярность может наступить уже около 2030 г. Сторонники теории технологической сингулярности считают, что если возникнет принципиально отличный от человеческого разум (постчеловек), дальнейшую судьбу цивилизации невозможно предсказать, опираясь на человеческое (социальное) поведение.

Треугольник Фреге – базовая структура в прикладной семиотике, представляющая знак как триединство понятия объекта, его имени и знаний о нём.

Фаззификация – процедура преобразования базовой переменной в нечёткую лингвистическую переменную, характеризующуюся функцией принадлежности.

Факт – декларативные знания, утверждающие истинность или ложность информации в искусственном интеллекте.

Формализованная задача имеет чёткий алгоритм решения (решение алгебраических уравнений, сортировка данных и т.п.).

Формальная система – *модель*, лежащая в основе многих математических теорий, состоящая из множества базовых элементов, синтаксических правил, аксиом и правил вывода.

Формула общезначимая – формула логики, которая истинна во всех интерпретациях.

Формула противоречивая – формула логики, ложная во всех интерпретациях.

Фрейм – декларативная модель представления и формализации знаний в виде структуры данных для представления стереотипных ситуаций. Фрейм идентифицируется уникальным именем и включает в себя множество слотов, в которых описывается информация о свойствах и характеристиках фрейма. Отражение в иерархии фреймов родовидовых отношений обеспечивает возможность реализации операции наследования.

Функция принадлежности – математическая функция, которая определяет нечёткое множество. Функции принадлежности используют субъективные данные, решают логическую задачу отнесения объекта к одному из классов и показывают не только факт принадлежности, но и меру принадлежности объекта определённому классу. После задания нечётких множеств для характеристики взаимосвязей входных и выходных величин исследуемого процесса применяются те или иные простые формы фазирования функции принадлежности, как линейные (треугольники или трапеции), так и нелинейные (колокол). Детально функции принадлежности изучаются в рамках теории Демпстера–Шеффера.

Эволюционные алгоритмы – группа методов эвристического поиска решений, моделирующих процессы природной эволюции. Представляется объектом вида $\mathcal{EA}(P_0, N, L, F, Sl, Cr, M)$, где P_0 – исходная популяция решений; N – размер популяции; L – длина битовой строки (хромосомы), кодирующей решение; F – фитнес-функция, определяющая «пригодность» решения; Sl – оператор селективного отбора; Cr – оператор кроссинговера (скрещивания), определяющий возможные новые решения; M – оператор мутации. Группа ЭА включает генетические алгоритмы, генетическое программирование, эволюционные стратегии, эволюционное программирование.

Эвристика – приём решения задачи, основанный не на строгих математических моделях и алгоритмах, а на соображениях "здравого смысла". Если этот приём удаётся запрограммировать, то такие программы называются эвристическими. Эвристики часто используются при программировании игр, имитации творческих процессов, в экспертных системах.

Экспертная система – интеллектуальная система, основанная на знаниях и использующая логику эксперта для эффективного решения задач в узкой предметной области. Такие системы представляют знания символически, исследуют и объясняют свои рассуждения. В динамических экспертных системах, в отличие от статических, информация изменяется в режиме реального времени решения задачи.

Эпистемология – наука о знании в рамках философии. Философы, занимающиеся данной проблематикой, решают вопросы, схожие с теми, которые решаются инженерами ИИ о том, как лучше представлять и использовать знания и информацию.

Эффективность – 1) степень достижения цели, 2) отношение входа к выходу, 3) приемлемое использование ресурса.

Язык представления знаний – способ описания моделей представления знаний. На сегодняшний день известны языки для модели знаний в виде фреймов (LISP, FRL, KRL и др.), а также ряд продукционных языков.

Ящик чёрный – введённое У.Р. Эшби наименование объекта исследования, внутреннее устройство которого неизвестно или не принимается во внимание. Модель «чёрного ящика» строится на основе имитации его поведения, т.е. реакции на воздействия, поступающие на вход извне, и характеризует связи между реакциями и вызвавшими их воздействиями. Модели чёрного ящика теоретически обосновываются в экспериментальной психологии (бихевиоризм) и обычно называются моделями "стимул - реакция".

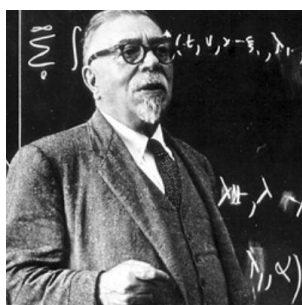
Персоналии

Вагин Вадим Николаевич (род. 1940) – доктор технических наук, профессор,



академик РАЕН. Автор фундаментальных работ в области математической логики, методов дедуктивного, индуктивного и абдуктивного выводов, нетрадиционных логик, интеллектуальных систем.

Винер Норберт (1894 – 1964) – выдающийся математик и философ,



основоположник кибернетики и теории искусственного интеллекта. Книга Винера «Кибернетика, или управление и связь в животном и машине» увидела свет в 1948 г. Был удостоен золотой медали Учёного, высшей награды для человека науки в Америке. На торжественном собрании, посвящённом этому событию, тогдашний президент США

Л.Джонсон произнёс: «Ваш вклад в науку на удивление универсален, Ваш взгляд всегда был абсолютно **оригинальным**, Вы потрясающее воплощение симбиоза чистого математика и прикладного учёного». При этих словах Винер достал носовой платок и прочувствованно высморкался.

Заде Лотфи (род. 1921 г.) – основатель теории нечётких множеств и нечёткой



логики, профессор Калифорнийского университета (Беркли). В 1965 г. опубликовал работу, в которой изложил математический аппарат теории нечётких множеств. В 1973 г. предложил теорию нечёткой логики, позднее – теорию «мягких вычислений».

Курейчик Виктор Михайлович (род. 1945) – доктор технических наук,



профессор, академик РАЕН, член Американского математического общества, автор фундаментальных работ в области генетических алгоритмов, автоматизации проектирования микроэлектронных устройств, научный руководитель школы «Эволюционное моделирование,

генетические алгоритмы и интеллектуальные САПР».

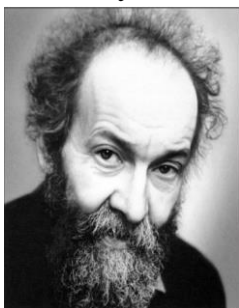
Лебедев Сергей Алексеевич (1902 – 1974) – академик, основоположник



вычислительной техники в СССР. Под его руководством была разработана первая в Европе малая электронно-счётная машина (МЭСМ), созданы 15 типов ЭВМ, начиная с ламповых (БЭСМ-1, БЭСМ-2, М-20) и заканчивая суперкомпьютером «Эльбрус».

Резко выступал против начавшегося в 1970-е годы копирования американской системы IBM 360, которая носила название ЕС ЭВМ. РАН учредила премию имени Лебедева за выдающиеся работы в области разработок вычислительных систем.

Ляпунов Алексей Андреевич (1911 – 1973) – выдающийся математик, один из



основоположников кибернетики, член-корреспондент АН СССР. Специалист в области теории функций вещественного переменного и математических вопросов кибернетики. Основные труды относятся к теории множеств, теоретическим вопросам программирования, математической лингвистике, математической биологии. В 1996 г. Ляпунову была присуждена медаль «Пионер

компьютерной техники».

Маккарти Джон (род. 1927) – американский учёный, специалист по теории



ЭВМ, математической логике, языкам программирования ЭВМ, искусственному интеллекту. Автор одного из языков программирования (Лисп), используемого при решении ряда сложных задач на ЭВМ, член ряда научных ассоциаций, занимающихся теорией ЭВМ, методами вычислений.

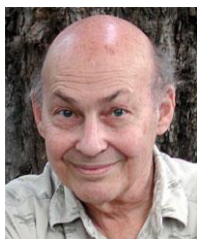
Мелихов Аскольд Николаевич (1939 – 1996) – основатель кафедры МОП ЭВМ



Таганрогского радиотехнического института, доктор технических наук, профессор, академик РАЕН. Автор фундаментальных работ в области теории автоматов, теории графов и нечётких множеств («Ориентированные графы и конечные автоматы», «Применение графов для проектирования дискретных устройств»,

«Ситуационные советующие системы с нечеткой логикой» и др.).

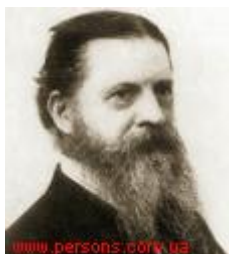
Минский Марвин (род. 1927) – американский учёный в области искусственного интеллекта, основатель Лаборатории искусственного интеллекта в Массачусетском технологическом институте, автор многочисленных публикаций по искусственному интеллекту и философии, в том числе книги «Перцептроны» (с С. Пайпертом), ставшей фундаментальной работой для последующих разработок в области искусственных нейронных сетей. Лауреат премии Тьюринга (1969).



Нейман Джон (1903 – 1957) – выдающийся математик, сделавший важный вклад в квантовую логику, функциональный анализ, теорию множеств, информатику, экономику и другие отрасли науки. Наиболее известен как основоположник современной архитектуры компьютеров, а также как участник Манхэттенского проекта и как создатель теории игр. С 1933 г. работал в Принстонском университете.



Пирс Чарлз Сандерс (1837 – 1914) – один из основоположников семиотики, в которой ввёл понятия экстенционала (объём понятия) и интенционала (содержание понятия). Для языковых конструкций ввёл понятия субъекта, предиката и связки, при помощи которых характеризовал предложения и высказывания.



Поспелов Дмитрий Александрович (род. 1932) – доктор технических наук, профессор, разработал подход к принятию решений, опирающийся на семиотические модели. Автор фундаментальных работ в области ситуационного управления большими системами. Создал теорию псевдофизических логик, моделирующих рассуждения "здорового смысла" о времени, пространстве, каузальных цепочках и т.п., что позволило в интеллектуальных системах поддержки принятия решений строить рассуждения о закономерностях физического мира и действиях в нём. Заведующий отделом ВЦ РАН, заведующий Международной лабораторией ЮНЕСКО по искусственному интеллекту, президент Российской ассоциации искусственного интеллекта.



Пост Эмиль Леон (1897 – 1954) – математик и логик, один из основателей



многозначной логики. Автор фундаментальных трудов по математической логике: алгебра Поста, классы Поста функций алгебры логики; предложил независимо от А. Тьюринга абстрактную вычислительную машину – машину Поста. Ему принадлежит одно из наиболее употребительных определений понятия непротиворечивости и полноты формальных систем; он

описал системы многозначной логики; одновременно с А.А.Марковым доказал алгоритмическую неразрешимость ряда проблем математической логики и алгебры.

Саймон Герберт (род. 1916) – американский ученый, лауреат Нобелевской



премии (1978) за новаторские исследования процесса поддержки принятия решений. В работах по исследованию процессов принятия сложных решений он использовал методы других наук, в частности искусственного интеллекта, оказавшие существенное влияние на развитие кибернетики.

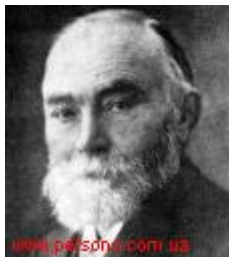
Тьюринг Алан (1912 – 1954). Его роль в истории информатики отнюдь не



исчерпывается одним лишь изобретением "машины Тьюринга". Он вполне может быть причислен к плеяде величайших математических и философских умов, составляющих гордость человечества, таких как Р.Декарт, Г.В. Лейбниц, Б.Рассел, Д.Гильберт, А.Витгенштейн и др. Был безразличен к борьбе за приоритет в научных открытиях. Мемориальная доска,

установленная на стене одной из лондонских гостиниц, гласит: «Здесь родился Алан Тьюринг (1912 – 1954), взломщик кодов [Code-breaker] и пионер информатики [computer science]». Только сейчас (но отнюдь не при жизни!) Тьюринг признан одним из основателей информатики и теории искусственного интеллекта, его считают первым теоретиком современного программирования и, наконец, первым в мире «хакером» (внёс во время второй мировой войны существенный вклад в победу союзных войск над германским флотом, расшифровав код «Энигмы»).

Фреге Готлиб (1848 – 1925) – немецкий философ и логик, один из основателей



логической семантики. Его труды стали новым этапом в развитии математической логики. Фреге осуществил аксиоматическое построение логики высказываний предикатов и положил начало теории математического доказательства. В своём фундаментальном труде «Основные законы арифметики»

он построил систему формализованной арифметики на основе разработанного им расширенного исчисления предикатов. Идеи Фреге во многом определили развитие логики; он впервые ввёл символы для обозначения кванторов, раскрыл отличие между значением и смыслом языковых выражений.

Шеннон Клод (1916 – 2001) – американский математик и электротехник, один



из создателей математической теории информации; в значительной мере предопределил развитие общей теории дискретных автоматов, которая является важной составляющей кибернетики. Работал в знаменитой Лаборатории Белла, преподавал в Массачусетском технологическом институте.

Автор фундаментальных работ «A Mathematical Theory of Communication» и «Communication Theory of Secrecy Systems», в которых он сформулировал математические основы теории информации и криптографии.