

Модели представления задач

ЛОГИЧЕСКИЕ МОДЕЛИ

Постановка и решение любой задачи всегда связаны с ее "погружением" в подходящую предметную область. Так, решая задачу составления расписания обработки деталей на металлорежущих станках, мы вовлекаем в предметную область такие объекты, как конкретные станки, детали, интервалы времени, и общие понятия "станок", "деталь", "тип станка" и т. п. Все предметы и события, которые составляют основу общего понимания необходимой для решения задачи информации, называются *предметной областью*. Мысленно предметная область представляется состоящей из реальных или абстрактных объектов, называемых *сущностями*.

Сущности предметной области находятся в определенных *отношениях* друг к другу (ассоциациях), которые также можно рассматривать как сущности и включать в предметную область. Между сущностями наблюдаются различные отношения подобия. Совокупность подобных сущностей составляет *класс сущностей*, являющийся новой сущностью предметной области.

ЛОГИЧЕСКИЕ МОДЕЛИ

Отношения между сущностями выражаются с помощью суждений. *Суждение* -это мысленно возможная ситуация, которая может иметь место для предъявляемых сущностей или не иметь места. В языке (формальном или естественном) суждениям отвечают *предложения*. Суждения и предложения также можно рассматривать как сущности и включать в предметную область.

Языки, предназначенные для описания предметных областей, называются *языками представления знаний*.

Универсальным языком представления знаний является естественный язык. Однако использование естественного языка в системах машинного представления знаний наталкивается на большие трудности ввиду присущих ему нерегулярностей, двусмысленностей, пресуппозиций и т. п. Но главное препятствие заключается в отсутствии формальной семантики естественного языка, которая имела бы достаточно эффективную операционную поддержку.

ЛОГИЧЕСКИЕ МОДЕЛИ

Для представления математического знания в математической логике давно пользуются логическими формализмами - главным образом *исчислением предикатов*, которое имеет ясную формальную семантику и операционную поддержку в том смысле, что для него разработаны механизмы вывода. Поэтому исчисление предикатов было первым логическим языком, который применили для формального описания предметных областей, связанных с решением прикладных задач.

Описания предметных областей, выполненные в логических языках, называются **(формальными) логическими моделями**.

СЕТЕВЫЕ МОДЕЛИ

Введем ряд определений.

Под сущностью будем понимать объект произвольной природы. Этот объект может существовать в реальном мире. В этом случае он будет называться ***П-сущностью***.

В базе знаний ему соответствует некоторое описание, полнота которого определяется той информацией, которую имеет о П-сущности ИС. Такое представление в базе знаний называется ***М-сущностью***.

Отметим, что могут существовать М-сущности, для которых в окружающем ИС мире нет соответствующих П-сущностей. Такие М-сущности представляют собой **абстрактные объекты**, полученные в результате операций типа обобщения внутри базы знаний.

СЕТЕВЫЕ МОДЕЛИ

Разделение на два типа сущностей позволяет использовать в сетевых моделях идеи, впервые сформулированные в теории *семиотических моделей* и основанном на них *ситуационном управлении*.

Под *семиотическими моделями проблемных областей* будет пониматься комплекс процедур, позволяющих отображать в базе знаний П-сущности и их связи, фиксируемые в проблемной области инженером по знаниям, в совокупность связанных между собой М-сущностей. Способ интерпретации взаимосвязанных П-сущностей будет называться *денотативной семантикой*, а способ интерпретации взаимосвязанных М-сущностей - *коннотативной семантикой*.

СЕТЕВЫЕ МОДЕЛИ

П-сущность по отношению к соответствующей ей в базе знаний М-сущности называется *денотатом* или *референтом* этой М-сущности, а М-сущность по отношению к исходной П-сущности – ее *десигнатом*, *именем*, *меткой*, *идентификатором* и т. п.

Десигнат - это простейший элемент в сетевой модели. Он входит в класс терминальных объектов сетевой модели. **Терминальным объектом** называется М-сущность, которая не может быть разложена на более простые сущности. Остальные М-сущности называются **производными объектами** или производными М-сущностями.

СЕТЕВЫЕ МОДЕЛИ

Перечень терминальных объектов, которые могут образовывать классы или типы, задается при проектировании ИС. Ими могут быть целые вещественные числа, идентификаторы, строки, списки и т. п. Семантика терминальных объектов определяется набором допустимых процедур, оперирующих с ними, например: арифметические действия над числами, сравнение между собой строк или идентификаторов, операции ввода-вывода, включающие необходимые трансформации представлений, и т. д.

ПРОДУКЦИОННЫЕ МОДЕЛИ

Продукции наряду с фреймами являются наиболее популярными средствами представления знаний в ИС. Продукции, с одной стороны, близки к логическим моделям, что позволяет организовывать на них эффективные процедуры вывода, а с другой стороны, более наглядно отражают знания, чем классические логические модели. В них отсутствуют жесткие ограничения, характерные для логических исчислений, что дает возможность изменять интерпретацию элементов продукции.

В общем виде под *продукцией* понимается выражение следующего вида:

(i); Q; P; A=>B; N.

ПРОДУКЦИОННЫЕ МОДЕЛИ

Здесь ***i*** - *имя продукции*, с помощью которого данная продукция выделяется из всего множества продукций. В качестве имени может выступать некоторая лексема, отражающая суть данной продукции (например, "покупка книги" или "набор кода замка"), или порядковый номер продукции в их множестве, хранящемся в памяти системы.

Элемент ***Q*** характеризует *сферу применения продукции*. Такие сферы легко выделяются в когнитивных структурах человека. Наши знания как бы "разложены по полочкам". На одной "полочке" хранятся знания о том, как надо готовить пищу, на другой - как добраться до работы и т. п. Разделение знаний на отдельные сферы позволяет экономить время на поиск нужных знаний. Такое же разделение на сферы в базе знаний ИС целесообразно и при использовании для представления знаний продукционных моделей.

ПРОДУКЦИОННЫЕ МОДЕЛИ

Основным элементом продукции является ее *ядро*: $A \Rightarrow B$. Интерпретация ядра продукции может быть различной и зависит от того, что стоит слева и справа от знака секвенции \Rightarrow . Обычное прочтение ядра продукции выглядит так: **ЕСЛИ A , ТО B** , более сложные конструкции ядра допускают в правой части альтернативный выбор, например, **ЕСЛИ A , ТО B_1 , ИНАЧЕ B_2** . Секвенция может истолковываться в обычном логическом смысле как знак логического следования B из истинного A (если A не является истинным выражением, то о B ничего сказать нельзя). Возможны и другие интерпретации ядра продукции, например A описывает некоторое условие, необходимое для того, чтобы можно было совершить действие B .

ПРОДУКЦИОННЫЕ МОДЕЛИ

Элемент P есть *условие применимости ядра продукции*. Обычно P представляет собой логическое выражение (как правило, предикат). Когда P принимает значение "истина", ядро продукции активизируется. Если P ложно, то ядро продукции не может быть использовано. Например, если в продукции "НАЛИЧИЕ ДЕНЕГ; ЕСЛИ ХОЧЕШЬ КУПИТЬ ВЕЩЬ X , ТО ЗАПЛАТИ В КАССУ ЕЕ СТОИМОСТЬ И ОТДАЙ ЧЕК ПРОДАВЦУ" условие применимости ядра продукции ложно, т. е. денег нет, то применить ядро продукции невозможно.

Элемент N описывает *постусловия продукции*. Они актуализируются только в том случае, если ядро продукции реализовалось. Постусловия продукции описывают действия и процедуры, которые необходимо выполнить после реализации V . Например, после покупки некоторой вещи в магазине необходимо в описи товаров, имеющихся в этом магазине, уменьшить количество вещей такого типа на единицу. Выполнение N может происходить не сразу после реализации ядра продукции.

ПРОДУКЦИОННЫЕ МОДЕЛИ

Если в памяти системы хранится некоторый набор продукций, то они образуют *систему продукций*. В системе продукций должны быть заданы специальные процедуры управления продукциями, с помощью которых происходит актуализация продукций и выбор для выполнения той или иной продукции из числа актуализированных.

В ряде ИС используются комбинации сетевых и продукционных моделей представления знаний. В таких моделях декларативные знания описываются в сетевом компоненте модели, а процедурные знания - в продукционном. В этом случае говорят о работе продукционной системы над семантической сетью.

СЦЕНАРИИ

Особую роль в системах представления знаний играют стереотипные знания, описывающие известные стандартные ситуации реального мира. Такие знания позволяют восстанавливать информацию, пропущенную в описании ситуации, предсказывать появление новых фактов, которых можно ожидать в данной ситуации, устанавливать смысл происхождения ситуации с точки зрения более общего ситуативного контекста.

Для описания стереотипного знания используются различные модели. Среди них наиболее распространенными являются сценарии. *Сценарием* называется формализованное описание стандартной последовательности взаимосвязанных фактов, определяющих типичную ситуацию предметной области. Это могут быть последовательности действий или процедур, описывающие способы достижения целей действующих лиц сценария (например, обед в ресторане, командировка, полет самолета, поступление в вуз). В ИС сценарии используются в процедурах понимания естественно-языковых текстов, планирования поведения, обучения, принятия решений, управления изменениями среды и др.

ИНТЕЛЛЕКТУАЛЬНЫЙ ИНТЕРФЕЙС

Предположим, что на вход ИС поступает текст. Будем говорить, то ИС понимает текст, если она дает ответы, правильные с точки зрения человека, на любые вопросы, относящиеся к тому, о чем говорится в тексте. Под "человеком" понимается конкретный человек-эксперт, которому поручено оценить способности системы к пониманию. Это вносит долю субъективизма, ибо разные люди могут по-разному понимать одни и те же тексты.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

В существующих ИС можно выделить пять основных уровней понимания и два уровня метапонимания.

Первый уровень характеризуется схемой, показывающей, что любые ответы на вопросы система формирует только на основе прямого содержания, введенного из текста. Если, например, в систему введен текст: "В восемь утра, после завтрака, Петя ушел в школу. В два часа он вернулся домой. После обеда он ушел гулять", то на первом уровне понимания система обязана уметь отвечать правильно на вопросы типа: "Когда Петя ушел в школу?" или "Что сделал Петя после обеда?". В лингвистическом процессоре происходит морфологический, синтаксический и семантический анализ текста и вопросов, относящихся к нему. На выходе лингвистического процессора получается внутреннее представление текста и вопросов, с которыми может работать блок вывода. Используя специальные процедуры, этот блок формирует ответы. Другими словами, уже понимание на первом уровне требует от ИС определенных средств представления данных и вывода на этих данных.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Второй уровень: На втором уровне добавляются средства логического вывода, основанные на информации, содержащейся в тексте. Это разнообразные логики текста (временная, пространственная, каузальная и т. п., которые способны породить информацию, явно отсутствующую в тексте. Для нашего примера на втором уровне возможно формирование правильных ответов на вопросы типа: "Что было раньше: уход Пети в школу или его обед?" или "Гулял Петя после возвращения из школы?" Только построив временную структуру текста, ИС сможет ответить на подобные вопросы.

Схема ИС, с помощью которой может быть реализован второй уровень понимания, имеет еще одну базу знаний. В ней хранятся закономерности, относящиеся к временной структуре событий, возможной их пространственной организации, каузальной зависимости и т. п., а логический блок обладает всеми необходимыми средствами для работы с псевдофизическими логиками.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Третий уровень: К средствам второго уровня добавляются правила пополнения текста знаниями системы о среде. Эти знания в ИС, как правило, носят логический характер и фиксируются в виде сценариев или процедур иного типа. На третьем уровне понимания ИС должна дать правильные ответы на вопросы типа: "Где был Петя в десять утра?" или "Откуда Петя вернулся в два часа дня?" Для этого надо знать, что означает процесс "пребывание в школе" и, в частности, что этот процесс является непрерывным и что субъект, участвующий в нем, все время находится "в школе".

Схема ИС, в которой реализуется понимание третьего уровня, внешне не отличается от схемы второго уровня. Однако в логическом блоке должны быть предусмотрены средства не только для чисто дедуктивного вывода, но и для вывода по сценариям.

Три перечисленных уровня понимания реализованы во всех практически работающих ИС. Первый уровень и частично второй входят в разнообразные системы общения на естественном языке.

Следующие два уровня понимания реализованы в существующих ИС лишь частично.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Четвертый уровень: Вместо текста в ней используется расширенный текст, который порождается лишь при наличии двух каналов получения информации. По одному в систему передается текст, по другому-дополнительная информация, отсутствующая в тексте. При человеческой коммуникации роль второго канала, как правило, играет зрение. Более одного канала коммуникации имеют интеллектуальные роботы, обладающие зрением.

Зрительный канал коммуникации позволяет фиксировать состояние среды "здесь и сейчас" и вводить в текст наблюдаемую информацию. Система становится способной к пониманию текстов, в которые введены слова, прямо связанные с той ситуацией, в которой порождается текст. На более низких уровнях понимания нельзя понять, например, текст: "Посмотрите, что сделал Петя! Он не должен был брать это!" При наличии зрительного канала процесс понимания становится возможным.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

При наличии четвертого уровня понимания ИС способна отвечать на вопросы типа: "Почему Петя не должен был брать это?" или "Что сделал Петя?" Если вопрос, поступивший в систему, соответствует третьему уровню, то система выдает нужный ответ. Если для ответа необходимо привлечь дополнительную ("экзегетическую") информацию, то внутреннее представление текста и вопроса передается в блок, который осуществляет соотнесение текста с той реальной ситуацией его порождения, которая доступна ИС по зрительному или какому-нибудь иному каналу фиксации ситуации внешнего мира.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Пятый уровень: Для ответа на этом уровне ИС кроме текста использует информацию о конкретном субъекте, являющемся источником текста, и хранящуюся в памяти системы общую информацию, относящуюся к коммуникации (знания об организации общения, о целях участников общения, о нормах участия в общении). Теория, соответствующая пятому уровню - это так называемая теория речевых актов.

Было обращено внимание на то, что любая фраза не только обозначает некое явление действительности, но и объединяет в себе три действия: *локуцию, иллокуцию и перлокуцию*. Локуция - это говорение как таковое, т. е. те действия, которые говорящий произвел, чтобы высказать свою мысль. Иллокуция - это действие при помощи говорения: вопрос, побуждение (приказ или просьба) и утверждение. Перлокуция - это действие, которым говорящий пытается осуществить некоторое воздействие на слушающего: "льстить", "удивлять", "уговаривать" и т. д. *Речевой акт* можно определить как минимальную осмысленную (или целесообразную) единицу речевого поведения. Каждый речевой акт состоит из локутивного, иллокутивного и перлокутивного акта.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Для четвертого и пятого уровней понимания интересны результаты по невербальным (несловесным) компонентам общения и психологическим принципам, лежащим в основе общения. Кроме того, в правила пополнения текста входят правила вывода, опирающиеся на знания о данном конкретном субъекте общения, если такие знания у системы есть. Например, система может доверять данному субъекту, считая, что порождаемый им текст истинен. Но может не доверять ему и понимать текст, корректируя его в соответствии со своими знаниями о субъекте, породившем текст. Знания такого типа должны опираться на психологические теории общения, которые пока развиты недостаточно.

Например, на вход системы поступает текст: "Нина обещала скоро прийти". Если о Нине у системы нет никакой информации, она может обратиться к базе знаний и использовать для оценки временного указателя "скоро некоторую нормативную информацию. Из этой информации можно узнать, что с большой долей уверенности "скоро" не превышает полчаса. Но у системы может иметься специальная информация о той Нине, о которой идет речь во входном тексте. В этом случае система, получив нужную информацию из базы знаний, может подготовиться, например, к тому, что Нина скорее всего придет не ранее чем через час.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Первый метауровень: На этом уровне происходит изменение содержимого базы знаний. Она пополняется фактами, известными системе и содержащимися в тех текстах, которые в систему введены. Разные ИС отличаются друг от друга характером правил порождения фактов из знаний. Например, в системах, предназначенных для экспертизы в области фармакологии, эти правила опираются на методы индуктивного вывода и распознавания образов. Правила могут быть основаны на принципах вероятностей, размытых выводов и т. п. Но во всех случаях база знаний оказывается априорно неполной и в таких ИС возникают трудности с поиском ответов на запросы. В частности, в базах знаний становится необходимым немонотонный вывод.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Второй метауровень: На этом уровне происходит порождение метафорического знания. Правила порождения знаний метафорического уровня, используемые для этих целей, представляют собой специальные процедуры, опирающиеся на вывод по аналогии и ассоциации. Известные в настоящее время схемы вывода по аналогии используют, как правило, диаграмму Лейбница, которая отражает лишь частный случай рассуждений по аналогии. Еще более бедны схемы ассоциативных рассуждений.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Если рассматривать уровни и метауровни понимания с точки зрения архитектуры ИС, то можно наблюдать последовательное наращивание новых блоков и усложнение реализуемых ими процедур. На первом уровне достаточно лингвистического процессора с базой знаний, относящихся только к самому тексту. На втором уровне в этом процессоре возникает процедура логического вывода. На третьем уровне необходима база знаний. Появление нового канала информации, который работает независимо от исходного, характеризует четвертый уровень. Кроме процедур, связанных с работой этого канала, появляются процедуры, увязывающие между собой результаты работы двух каналов, интегрирующие информацию, получаемую по каждому из них. На пятом уровне развитие получают разнообразные способы вывода на знаниях и данных. На этом уровне становятся важными модели индивидуального и группового поведения. На метауровнях возникают новые процедуры для манипулирования знаниями, которых не было на более низких уровнях понимания. И этот процесс носит открытый характер. Понимание в полном объеме - некоторая, по-видимому, недостижимая мечта. Но понимание на уровне "бытового понимания" людей в ИС вполне достижимо.

КЛАССИФИКАЦИЯ УРОВНЕЙ ПОНИМАНИЯ

Существуют и другие интерпретации феномена понимания. Можно, например, оценивать уровень понимания по способности системы к объяснению полученного результата. Здесь возможен не только уровень *объяснения*, когда система объясняет, что она сделала, например, на основе введенного в нее текста, но и уровень *обоснования (аргументации)*, когда система обосновывает свой результат, показывая, что он не противоречит той системе знаний и данных, которыми она располагает. В отличие от объяснения обоснование всегда связано с суммой фактов и знаний, которые определяются текущим моментом существования системы. И вводимый для понимания текст в одних состояниях может быть воспринят системой как истинный, а в других-как ложный. Кроме объяснения и обоснования возможна еще одна функция, связанная с пониманием текстов,- *оправдание*. Оправдать нечто означает, что выводимые утверждения не противоречат той системе норм и ценностей, которые заложены в ИС. Существующие ИС типа экспертных систем, как правило, способны давать объяснения и лишь частично обоснования. В полном объеме процедуры обоснования и оправдания еще не реализованы.