

ИННОВАЦИОННЫЙ ЕВРАЗИЙСКИЙ УНИВЕРСИТЕТ
МАГИТРАТУРА

Кафедра «Автоматизированные системы обработки информации и
управления»

Магистерская диссертация

ПРИМЕНЕНИЕ ЕСТЕСТВЕННО-ЯЗЫКОВЫХ ИНТЕРФЕЙСОВ В
ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ
6N0607 «Вычислительная техника и программное обеспечение»

Исполнитель _____ Л.Н.Изотова
(подпись, дата)

Научный руководитель

Профессор _____ Р.А.Шагиева
(подпись, дата)

Допущена к защите:
Зав.кафедрой «АСОИиУ»
профессор _____

В..В.Наумов
(подпись, дата)

Павлодар, 2009

РЕФЕРАТ

Магистерская диссертация на тему «Применение естественно-языковых интерфейсов в образовательном процессе» содержит 87 листов, 29 иллюстраций, 10 таблиц, имеет ссылки на 29 источников.

Ключевые слова: естественно-языковой интерфейс, естественно-языковая система, предметная область, пирамида знаний, семантическое пространство, извлечение знаний, лингвистический аспект извлечения знаний, концепты, метапонятия, семантические связи, психосемантика, субъективное пространство предметной области, многомерное шкалирование, кластерный анализ, логикодискриптивный подход, семантическая база.

Цель работы: выявление латентных (скрытых) структур знаний по дисциплине «Программирование на алгоритмических языках», проверка гипотезы о возможности их применения для контроля знаний студентов.

Актуальность темы обусловлена все возрастающей ролью открытых образовательных систем, реализованных как дистанционная форма обучения. Средства промежуточного и итогового контроля знаний в этих условиях должны быть еще более надежными и точными.

Объект исследования: семантические пространства предметной области «Программирование на алгоритмических языках» экспертов (преподавателей дисциплины) и испытуемых (студентов).

Методы исследования: на этапе выявления семантических пространств применялись такие методы инженерии знаний, как составление оглавления, составление концептов предметной области, метод карточек, метод замкнутых кривых; на этапе обработки семантических пространств: статистические многомерные методы: многомерное шкалирование, кластерный анализ.

Результаты исследования: экспериментальные данные о семантических пространствах предметной области «Программирование на алгоритмических языках» экспертов и испытуемых, критерии сравнения семантических пространств, методы оценки семантических пространств с целью контроля знаний. Собраны данные о возможной методике проведения таких исследований в других предметных областях.

Достоверность полученной модели проверена с помощью методов многомерного шкалирования и кластерного анализа. Эта проверка показала, что выделенные экспериментально концепты и семантические связи предметной области объединяются в метапонятия в соответствии с логической структурой языков программирования, что доказывает адекватность полученной и реальной моделей.

Достоверность полученных результатов подтверждена сравнением традиционных оценок испытуемых по дисциплине «Программирование на алгоритмических языках» и оценок их семантических пространств в этой предметной области. В целом, традиционные и полученные оценки совпали на 80%.

Практическая значимость работы состоит в том, что полученные результаты позволяют продолжить разработку специального программного средства, в рамках которого осуществляется оценка знаний студентов на основе их ответов, данных в свободной форме на естественном языке.

СОДЕРЖАНИЕ

Введение	5
1. Естественно-языковые интерфейсы	7
1.1 Структура ЕЯ-систем	8
1.2 Классификация ЕЯ-систем	16
1.3 Методы реализации компонента понимания высказываний ЕЯ-систем	19
2. Построение пирамиды знаний предметной области	23
2.1 Процесс формирования структуры предметной области в процессе обучения	23
2.2 Извлечение знаний о структуре предметной области	25
2.3 Психосемантика и инженерия знаний	28
2.4 Многомерные методы обработки данных. Многомерное шкалирование	38
2.5 Многомерные методы обработки данных. Кластерный анализ	48
2.6 Основные статистические понятия организации эксперимента	51
3. Проведение эксперимента и обработка полученных данных	55
3.1 План эксперимента	55
3.2 Ход эксперимента	60
3.3 Результаты эксперимента	78
3.4 Выводы и рекомендации	83
Заключение	88
Список использованных источников	90
Приложение А	

ВВЕДЕНИЕ

В конце 60-х годов в исследованиях по искусственному интеллекту сформировалось самостоятельное направление, получившее название «обработка естественного языка» (Natural Language Processing). Задачей данного направления является исследование методов и разработка систем, обеспечивающих реализацию процесса общения с компьютерными системами на естественном языке (систем ЕЯ - общения или ЕЯ-систем).

Следует отметить, что проблематика коммуникативного взаимодействия, и в частности ЕЯ-общения, находится в центре внимания многих наук, например, лингвистики, психологии, логики и философии. Однако все они исследуют лишь отдельные аспекты процесса общения. В отличие от них искусственный интеллект, как прикладная дисциплина, вынужден моделировать в рамках ЕЯ-систем все или, по крайней мере, основные аспекты ЕЯ-общения, правда, может быть, не на столь глубоком уровне.

Проблема взаимодействия человека с компьютером существует с момента появления вычислительной техники. На начальном этапе непосредственное взаимодействие с ЭВМ осуществляли только программисты, а специалисты других областей – потребители результатов, полученных на компьютере, выступали в роли косвенных конечных пользователей, т. е. общались с компьютером через программистов. По мере расширения сферы использования компьютера и увеличения масштабов их применения конечные пользователи стали вовлекаться в процесс непосредственного взаимодействия с компьютером, что привело к появлению массовой категории пользователей – прямых конечных пользователей, работающих в диалоговом режиме. К пользователям этой категории относятся специалисты в различных проблемных областях, которые решают свои профессиональные задачи, непосредственно используя компьютер, т. е. Прикладные программы и пакеты (прикладные системы), входящие в программное обеспечение компьютера. Как правило, эти пользователи не обладают знаниями в области компьютерной обработки данных и не умеют программировать.

Сложность создания средств общения, предназначенных для конечных пользователей, обусловлена в значительной степени отсутствием единой теории языкового общения, охватывающей все аспекты взаимодействия коммуникантов. Поэтому при разработке средств общения конечных пользователей на процесс взаимодействия часто налагаются различные «спонтанные» ограничения, последствия которых не до конца осознаются разработчиками. Эти ограничения приводят к тому, что многие человеко-машинные системы, на разработку которых тратятся огромные средства, не удовлетворяют требованиям конечных пользователей.

Естественно-языковые системы используются для поиска в текстах, распознавания речи, голосового управления и обработки данных. Их доля на рынке составляет около 14%. В данном направлении выделяются следующие категории информационных продуктов:

- средства, обеспечивающие естественно-языковой интерфейс к базам данных;
- средства естественно-языкового поиска в текстах и содержательного сканирования текстов (Natural Language text retrieval and Contents Scanning Systems);
- масштабируемые средства для распознавания речи (Large-Vocabulary Talkwriter);
- средства голосового ввода, управления и сбора данных (Voice Input and Control Products and Data Collection Systems);
- компоненты речевой обработки (Voice-Recognition Programming Tools).

Программные продукты первой категории преобразуют естественно-языковые запросы пользователя в SQL-запросы к базам данных. Средства естественно-языкового поиска в текстах осуществляют по запросам пользователей поиск, фильтрацию и сканирование текстовой информации. В отличие от продуктов предыдущей группы, где поиск осуществляется в базах данных, имеющих четкую и заранее известную структуру, средства данной категории осуществляют поиск в неструктурированных текстах, оформленных в соответствии с правилами грамматики того или иного естественного языка. Средства для распознавания речи распознают голосовую информацию и преобразуют ее в последовательность символов. Они ориентированы на работу со словарями объемом от 30000 до 70000 слов. В отличие от этого, средства голосового ввода ориентированы на работу со словарем около 1000 слов и существенно ограничены в возможностях распознавания. Программные средства этого типа предназначены для ввода голосовых команд, управляющих работой некоторого продукта, например, программы сбора данных в тех приложениях, в которых у исполнителей заняты руки. Компоненты речевой обработки предназначены для программистов, которых хотят добавить возможности по распознаванию речи в разрабатываемые ими приложения [1].

В настоящее время разработано множество моделей языкового общения и лингвистических моделей представления языка на всех его уровнях. Создано множество прикладных программ, в которых представлены все основные функциональные компоненты ЕЯ - интерфейса.

Однако помимо функционально полных систем ведется интенсивная разработка систем, которые можно назвать фрагментарными. Цель их создания - исследование или реализация новых методов решения достаточно узких задач (например таких, как анализ, интерпретация, определение целей пользователя, контроль знаний и т. п.).

Благодаря модульности структуры ЕЯ-систем фрагментарные системы могут успешно использоваться в качестве отдельных функциональных блоков, встраиваемых (хотя бы на логическом уровне) в различные функционально полные системы. Например, в автоматизированных системах обучения, интенсивно развивающихся в условиях открытого образования, может присутствовать функциональный блок обработки ответов пользователя (обучаемого) на естественном языке с целью контроля его знаний [2].

1 ЕСТЕСТВЕННО-ЯЗЫКОВЫЕ ИНТЕРФЕЙСЫ

Естественно-языковые интерфейсы (ЕЯ-интерфейсы) – это комплекс средств, позволяющий неподготовленному пользователю общаться с компьютером на естественном языке. Такие программы также называют естественно-языковыми системами (ЕЯ-системы). В теории такое общение ничем не ограничено, на практике же не существует ни одной программы, полностью воспроизводящей человеческое общение. Большинство программ, работающих с естественным языком, имеет массу ограничений. Одно из самых главных – применение конкретного интерфейса в конкретной предметной области.

Прежде чем приступить к рассмотрению архитектуры ЕЯ-систем, остановимся на понятиях, которые широко используются при их разработке и описании [3].

Общение – коммуникативное взаимодействие, диалог – процесс достижения его участниками определенных согласованных целей путем обмена связанными высказываниями, выраженными в языке о некотором реальном или гипотетическом мире (проблемной области). Применительно к диалогу между пользователем и ЭВМ под общением понимают процесс обмена взаимосвязанными высказываниями, направленный на достижение целей пользователя.

В общем случае процесс общения не может быть сведен к обмену изолированными парами высказываний вопрос-ответ. Высказывания участников общения образуют связный текст – дискурс, имеющий, как правило, достаточно сложную структуру. Связность дискурса обеспечивается как лингвистическими (родовидовыми, модальными, стилистическими согласованиями и т.п.), так и экстралингвистическими (ситуативными) средствами, т.е. с помощью временных, причинно-следственных и других связей, существующих в проблемной области. Следует отметить, что разговорный ЕЯ более компактен, чем литературный, «письменный» язык, так как при общении широко используются разнообразие умолчания, восстанавливаемые (раскрываемые) участниками исходя из текущих целей диалога.

Если участники достигли цели, поставленные в начале общения, то говорят, что общение завершилось успехом (глобальным успехом), в противном случае – неудачей (глобальной неудачей). В процессе общения могут возникать различные локальные неудачи, вызванные, например, неправильностью высказываний участников, непониманием друг друга из-за различных представлений о теме диалога или о проблемной области и языке общения и т.п. Большая часть локальных неудач не приводит к глобальной неудаче.

Под неправильностью высказываний пользователя понимается следующее: пользователь может применять как «правильные» предложения, т.е. такие, которые будут однозначно поняты и верно обработаны системой, так и «неправильные». Неправильности могут быть вызваны, во-первых, тем, что пользователь обычно не в состоянии учесть все ограничения системы общения в части ее возможностей и знаний, во-вторых, использованием умолчаний, характерных для естественного общения и допускающих неоднозначное толкование высказываний, и, в-третьих, отклонений предложений от грамматической нормы.

Шаг диалога трактуется как пара действие – реакция, где высказывание активного (т.е. владеющего инициативой) участника соответствует действию, а пассивного – реакции. Действие и реакция, образующие шаг диалога, могут в общем случае не соответствовать соседним (во временной последовательности) высказываниям участников. Соответствие нарушается при перехватах инициативы.

Перехват инициативы возникает в тех случаях, когда пассивный участник вместо цели (подцели), предложенной активным участником, выбирает иные цели (подцели), в частности подцели, предусматривающие преодоление локальных неудач. Например, вместо ответа на вопрос (что соответствовало бы стандартной реакции) второй участник может задать встречный вопрос (т.е. совершить действие и тем самым взять на себя активную роль) и лишь после получения ответа на него ответить на первоначально заданный вопрос (и тем самым вернуть инициативу). Таким образом, перехват инициативы как бы разрывает первоначально инициированный шаг диалога и открывает поддиалог – происходит смена цели (темы) диалога, в котором инициативой владеет ранее пассивный участник.

1.1 Структура ЕЯ-систем

Традиционные средства общения не позволяют обеспечить взаимодействие конечных пользователей с ЭВМ [18]. Чтобы быть полноправным участником общения, ЕЯ-система должна выполнять некоторые обязательные функции. К этим функциям относятся, по крайней мере, следующие:

- ведение диалога – определение его структуры и той роли, которую система и пользователь выполняют на текущем шаге диалога;
- понимание – преобразование поступающих от пользователя высказываний на ЕЯ в высказывания на языке внутреннего представления;
- обработка высказываний – формирование или определение заданий на решение задач или подзадач на данном шаге диалога; генерация – формирование выходных высказываний на ЕЯ.

Приведенные функции имеют обобщенный характер. Поэтому необходимо подчеркнуть, что при реализации конкретных ЕЯ-систем суть этих функций может в значительной степени варьироваться.

В соответствии с выделенными функциями обобщенная схема ЕЯ-системы может быть представлена в виде трех компонентов: диалоговый; компонент понимания высказываний; компонент генерации высказываний (рисунок 1).

1.1.1 Диалоговый компонент

К основным задачам диалогового компонента относятся:

- ведение диалога;
- формирование или обработка перехватов инициативы.

Ведение диалога состоит в том, чтобы обеспечивать целесообразные (т.е. способствующие достижению конечных целей пользователя) действия системы на текущем шаге диалога. В связи с тем, что возможности существующих ЕЯ-систем не позволяют им самостоятельно формировать целесообразное поведе-

ние, в систему обычно вводится информация, определяющая общую и тематическую структуры диалога. По структуре и текущему состоянию диалога диалоговый компонент формирует (если инициатива принадлежит системе) или определяет (если инициатива принадлежит пользователю) задание, выполняемое системой на текущем шаге (например, генерация вопроса, понимание ответа и его обработка, генерация утверждения и т.п.).

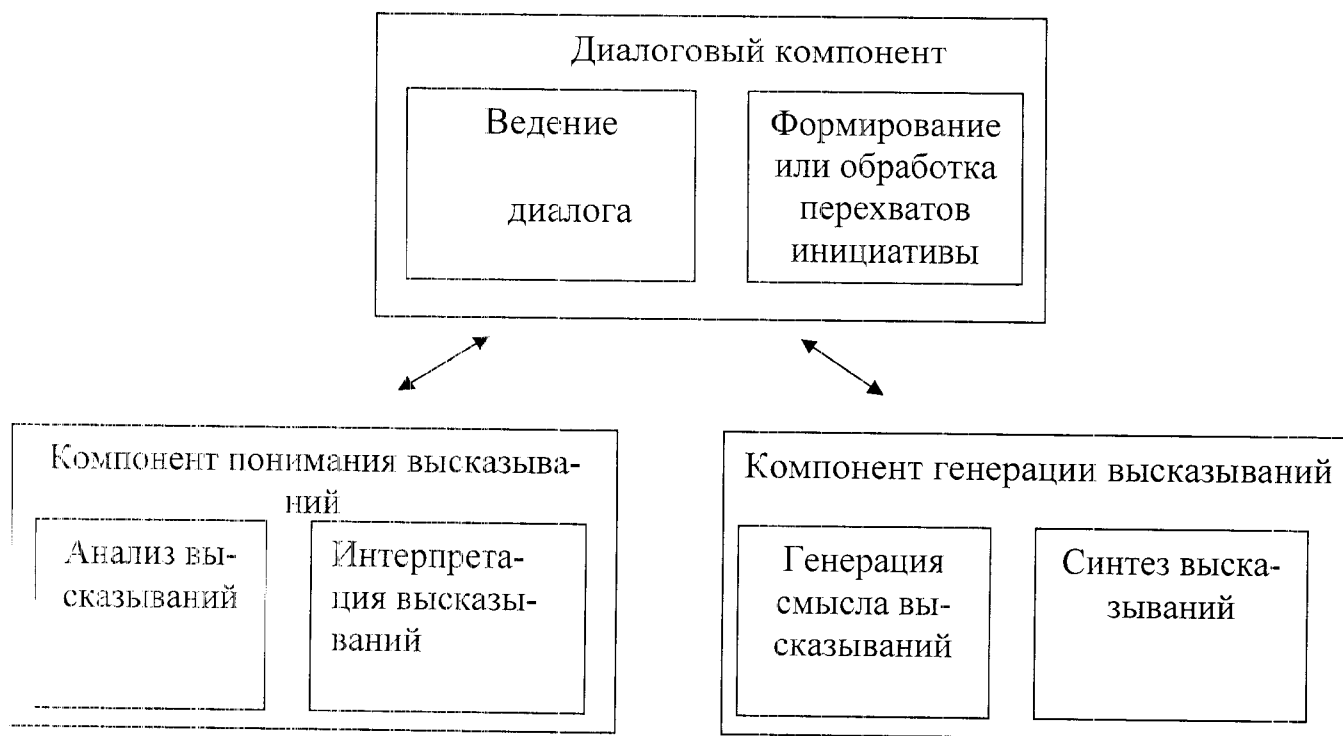


Рисунок 1 – Структура ЕЯ-интерфейса

Ведение диалога выполняется в одном из двух режимов:

- диалог ведет пользователь;
- диалог ведет система.

В первом случае инициатива в основном (за исключением сообщений об ошибках) находится у пользователя, а система только реагирует на его требования, определяя по виду требования тип задания. Разбиение задачи на подзадачи и распределения ролей осуществляет пользователь, т.е. для системы весь диалог сводится к выработке реакций на текущие высказывания пользователя. Можно сказать, что в этом варианте работы функции диалогового компонента практически вырождены. Во втором случае инициатива в основном принадлежит системе. Система ведет диалог в соответствии с имеющимися у нее представлениями о структуре диалога (т.е. о разбиении задач на подзадачи и о том, кто из участников, когда и какую подзадачу решает) и о способе обмена высказываниями.

Если роли участников неизменны, однозначны и предопределены заранее, то структуру диалога называют жесткой. В простейшем случае такая структура диалога сводится к двум взаимосвязанным высказываниям участников (вопрос - ответ) с указанием участника, владеющего инициативой. Развитием жесткой структуры является альтернативная структура, которая задает множество возможных (но заранее предписанных) направлений течения диалога. Выбор одно-

го из возможных направлений осуществляет пассивный участник. Если роли участников общения распределяются в ходе общения, то структуру диалога называют гибкой. Гибкие структуры подразделяются по степени свободы выбора момента перехвата (предопределенные моменты, произвольные моменты) и способам перехвата инициативы (предопределенный способ перехвата, произвольный способ).

Формирование или обработка перехватов инициативы. Вторая задача диалогового компонента вызвана тем, что реакция одного участника могут не соответствовать ожиданиям другого. В зависимости от того, кто осуществляет перехват инициативы, система либо формирует перехват, либо обрабатывает его. Формирование происходит в тех случаях, когда система определяет, что текущая ситуация не соответствует ситуации, предусмотренной структурой диалога. Если же перехват инициативы осуществляет пользователь, то задача системы – обработать его, т.е. распознать наличие перехвата инициативы, определить новую тему (цель), на которую перешел пользователь, и перейти на структуру диалога, соответствующую новой теме.

1.1.2 Компонент понимания высказываний

Компонент предназначен для выделения смысла входного высказывания и выражения этого смысла на внутреннем языке системы. Под смыслом высказывания обычно понимается вся та семантико-прагматическая информация, которую пользователь хотел передать системе. Выявление смысла высказывания в общем случае требует его рассмотрения в контексте всего диалога.

Традиционно задача понимания высказываний подразделяется на этапы:

- анализ;
- интерпретацию.

На этапе анализа высказываний выделяются описания сущностей, упомянутых во входном высказывании, выявляются свойства этих сущностей и отношения между ними. Анализ обычно выполняется отдельным блоком – анализатором, служащим ядром компонента понимания. Анализаторы, разрабатываемые для ЕЯ-систем, различаются по ряду параметров. Основные из них следующие:

- тип анализируемых предложений: повествовательные, вопросительные, отрицательные, полные, неполные, простые, сложные, распространенные;
- выделяемые описания сущностей: понятия, отношения;
- глубина проникновения в смысл: множество ключевых слов, имя события и описания участников события, пространственно-временное или причинное представление ситуации;
- используемые для анализа средства: морфологический, синтаксический, семантический, прагматический анализ.

Интерпретация заключается в отображении входного высказывания на знания системы. Основными задачами данного этапа являются:

- буквальная интерпретация высказывания в контексте диалога;
- интерпретация высказывания на намерения говорящего.

1.1.3 Компонент генерации высказываний

Этот компонент решает в соответствии с результатами, полученными остальными компонентами системы, две основные задачи:

- генерация смысла, т.е. определение типа и смысла выходного высказывания система во внутреннем представлении;
- синтез высказывания, т.е. преобразование смысла в высказывание на естественном языке.

Первая из указанных задач является сложной и мало изученной. Тип высказывания зависит от состояния системы и результатов, полученных предыдущими компонентами. Так, если система должно генерировать ответ на вопрос, то необходимо определить по состоянию системы, будет ли ответ прямой или косвенный. Прямой ответ (т.е. по существу заданного вопроса) может быть дан только в том случае, если обработка вопроса завершилась успешно. Если в процессе обработки вопроса возникают какие-то затруднения, то более уместным может быть косвенный ответ, раскрывающий суть возникших затруднений и объясняющий невозможность прямого ответа.

Вторая задача компонента генерации высказываний состоит в синтезе естественно-языкового выражения, соответствующего внутреннему представлению выходного высказывания. Данная задача подразделяется на этапы семантического, синтаксического и морфологического синтеза. Сложность задачи синтеза определяется требованиями естественности и выразительности мощности выходных высказываний. Под естественностью обычно понимается степень близости к естественному языку, т.е. наличие таких свойств, как синонимия и омоимия слов и словосочетаний, свободный порядок слов и т.д. Под выразительной мощностью понимается возможность выразить разнообразные понятия, отношения, кванторы, процедуры и т.д. Естественность и выразительная мощность выходных высказываний в существующих системах могут быть различными. Например, высказывания могут фактически не синтезироваться, а выбираться из заранее заготовленного списка либо имеется шаблон ответа, в который подставляются некоторые слова, представляющие собой значения искомых атрибутов; при этом может использоваться морфологический синтез. Большая естественность достигается если выходное высказывание формируется из семантического представления события (или понятия) с применением морфологии, синтаксиса (для определения порядка и формы слов) и семантики (для выбора лексем и обеспечения семантической сочетаемости слов в синтезируемом высказывании).

1.1.4 Знания, используемые в ЕЯ-системе

Для понимания принципов построения ЕЯ-систем важен также вопрос об используемых в системе знаниях, поскольку именно знания, представленные в различных формах, являются той базой, на которой осуществляется решение рассмотренных выше задач. На верхнем уровне классификации знаний ЕЯ-систем выделяются собственно знания и способ представления знаний (рисунок 2).

Собственно знания классифицируются по областям и по видам знаний. Наиболее существенными с точки зрения процесса ЕЯ-общения являются сле-

дующие области знаний: проблемная область, область языка, область системы, область пользователя, область диалога (дискурса). Разнообразие областей определяет множество возможных интерпретаций входных высказываний.

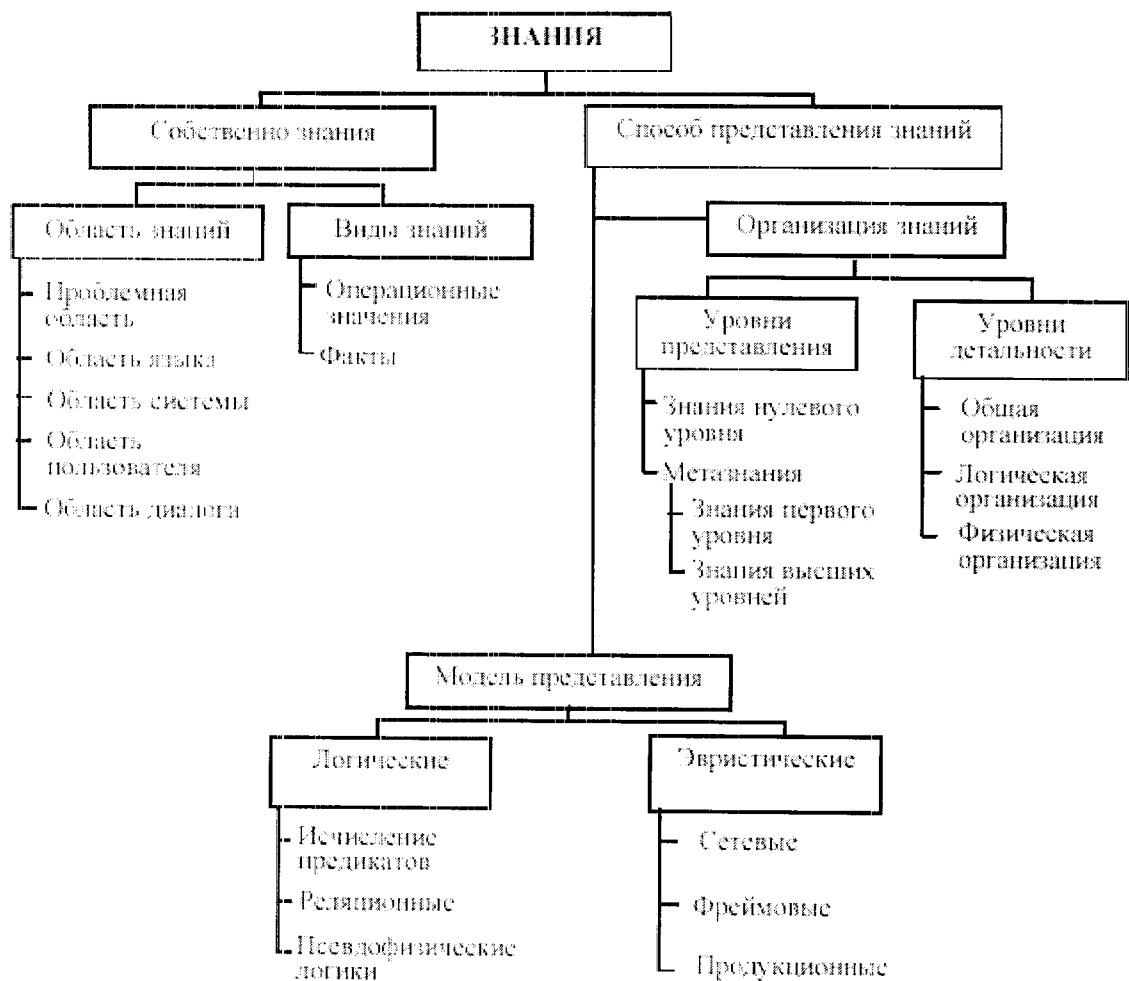


Рисунок 2 – Знания ЕЯ-систем

К основным видам знаний относятся факты (фактические знания) и операционные знания. Факты представляют собой возможные знания о сущностях, составляющих некоторую область знаний. Операционные знания составляет информация о способах изменения фактических знаний. Иначе говоря, эти знания задают процедуры преобразования.

Способ представления знаний включает два аспекта: способ организации знаний и модель представлений. По уровням представления выделяют знания нулевого уровня (конкретные и абстрактные знания) и знания более высоких уровней (метазнания). Первый уровень составляют знания о том, как в системе представлены знания нулевого уровня. Число уровней представления может быть продолжено. Разделение знаний по уровням представления обеспечивает возможность гибкой настройки и адаптации ЕЯ-систем.

Организация знаний по уровням детальности позволяет рассматривать знания с различной степенью подробности. Количество уровней детальности зависит от специфики решаемых задач, количества знаний и способа их представления. Обычно выделяется не менее трех уровней, отражающих общую органи-

зацию знаний, логическую и физическую организацию частных структур знаний. Введение нескольких уровней детальности обеспечивает дополнительную гибкость системы, так как изолирует изменения одного уровня от других.

Модели представления знаний обычно подразделяются на логические и эвристические, на декларативные и процедурные.

Декларативная модель основывается на предположении, что проблема представления некоей предметной области решается независимо от того, как эти знания потом будут использоваться. Поэтому модель как бы состоит из двух частей: статических описательных структур знаний и механизма вывода, оперирующего этими структурами и практически независимого от их содержательного наполнения. При этом в какой-то степени оказываются раздельными синтаксические и семантические аспекты знания, что является определенным достоинством указанных форм представления из-за возможности достижения их определенной универсальности.

В декларативных моделях не содержатся в явном виде описания выполняемых процедур. Эти модели представляют собой обычно множество утверждений. Предметная область представляется в виде синтаксического описания ее состояния (по возможности полного). Вывод решений основывается в основном на процедурах поиска в пространстве состояний.

В процедурном представлении знания содержатся в процедурах – небольших программах, которые определяют, как выполнять специфичные действия (как поступать в специфичных ситуациях). При этом можно не описывать все возможные состояния среды или объекта для реализации вывода. Достаточно хранить некоторые начальные состояния и процедуры, генерирующие необходимые описания ситуаций и действий.

Семантика непосредственно заложена в описание элементов базы знаний, за счет чего повышается эффективность поиска решений. Статическая база знаний мала по сравнению с процедурной частью. Она содержит так называемые "утверждения", которые приемлемы в данный момент, но могут быть изменены или удалены в любой момент. Общие знания и правила вывода представлены в виде специальных целенаправленных процедур, активизирующихся по мере надобности. Процедуры могут активизировать друг друга, их выполнение может прерываться, а затем возобновляться. Возможно использование процедур - "демонов", активизирующихся при выполнении операций введения, изменения или удаления данных.

Средством повышения эффективности генерации вывода в процедурных моделях является добавление в систему знаний о применении, т.е. знаний о том, каким образом использовать накопленные знания для решения конкретной задачи. Эти знания, как правило, тоже представляются в процедурной форме.

Главное преимущество процедурных моделей представления знаний заключается в большей эффективности механизмов вывода за счет введения дополнительных знаний о применении, что, однако, снижает их общность. Другое важное преимущество заключено в выразительной силе. Эти системы способны смоделировать практически любую модель представления знаний. Выразительная сила процедурных систем проявляется в расширенной системе выводов, реализуемых в них.

Большинство расширенных форм выводов может быть охарактеризовано понятием "предположение об отсутствии" и сводится к схеме: "Если А (предварительное условие) - истинно и нет доказательств против В, то предложить В". Подобные правила вывода оказываются полезными в основном в двух случаях:

1. Неполнота знаний. Если в системе представления отдельные факты не представлены или невыводимы, правила вывода позволяют гипотетически признавать их верными при условии, что в системе нет или в ней невыводимы доказательства противного.

2. Вывод в условиях ограниченности ресурсов. Из-за ограниченности ресурсов процессы вывода не могут завершиться, а должны быть оставлены для получения результатов. В этом случае правила определяют дальнейшие действия системы.

Системы представления, содержащие подобные правила, оказываются не-монотонными, т.е. добавление новых утверждений может запретить генерацию вывода, который первоначально мог быть получен. Добавление новых фактов может привести к возникновению противоречий. В некоторых системах кроме самих утверждений содержатся также записи причин, по которым были приняты эти утверждения. При добавлении новых фактов осуществляется проверка того, сохраняются ли справедливость утверждений и соответствие причинам.

Можно выделить ряд общих для всех систем представления знаний (СПЗ) черт [4,5]. Все СПЗ имеют дело с двумя мирами - представляемым и представляющим. Вместе они образуют систему для представления. Существует также ряд общих для всех СПЗ проблем. К ним можно отнести, в частности, проблемы: приобретения новых знаний и их взаимодействие с уже существующими, организации ассоциативных связей, неоднозначности и выбора семантических примитивов, явности знаний и доступности, выбора соотношения декларативной и процедурной составляющих представления, что влияет на экономичность системы, полноту, легкость кодировки и понимания.

Рассмотрим различные формы моделей представления знаний.

Продукционные модели представляют собой набор правил в виде "условие - действие", где условия являются утверждениями о содержимом БД (фактов), а действия есть некоторые процедуры, которые могут модифицировать содержимое БД. Продукционные модели из-за модульного представления знаний, легкого расширения и модификации нашли широкое применение в ЭС и ЕЯ-системах.

Другая важная схема представления знаний – семантические сети, представляющие собой направленный граф, в котором вершинам ставятся в соответствие конкретные объекты, а дугам, их связывающим, - семантические отношения между этими объектами. Семантические сети могут использоваться как для декларативных, так и для процедурных знаний.

Перспективной формой представления знаний являются фреймы, которые быстро завоевали популярность у разработчиков систем ИИ благодаря своей универсальности и гибкости.

Принципиальным методом для логического представления знаний является использование логики предикатов первого порядка (исчисление предикатов). При таком подходе знания о некоторой предметной области могут рассматри-

ваться как совокупность логических формул. Изменения в модели представления знаний происходят в результате добавления или удаления логических формул.

В редуccionных моделях осуществляется декомпозиция исходной задачи на ряд подзадач, решая которые последовательно определяют решение поставленной задачи.

Логические представления легки для понимания и располагают правилами вывода, необходимыми для операций над ними. Однако в логических моделях представление знаний отношения между элементами знаний выражаются ограниченным набором средств используемой формальной системы, что не позволяет в полной мере отразить специфику предметной области. Недостатком логического представления является также тенденция потреблять большие объемы памяти ЭВМ.

Ряд понятий человеческих знаний оказывается трудно, а иногда и невозможно описать количественно, используя детерминированные или стохастические методы. Трудности возникают при создании моделей не полностью определенных, неточных, нечетких знаний. Это связано с тем, что человеческому мышлению присуща лингвистическая неопределенность; знания и понятия, которыми оперирует человек, часто имеют качественную природу, они ситуативны, бывают неполными. Для формализации знаний такого типа используется аппарат теории нечетких множеств, создание которого связано с именем известного американского ученого Л. Заде.

Неточность, неопределенность или неполнота, заключенные в смысловых значениях или выводах, присущи естественным языкам с их сложной структурой и многообразием понятий. Различают несколько типов неопределенности в прикладных системах ИИ. Первый связан с ненадежностью исходной информации — неточность измерений, неопределенность понятий и терминов, неуверенностью экспертов в своих заключениях.

Второй — обусловлен нечеткостью языка представления правил, например в экспертных системах. Неопределенность возникает также, когда вывод базируется на неполной информации, т.е. нечетких посылах. Еще один тип неопределенности может появляться при агрегации правил, исходящих от разных источников знаний или от разных экспертов. Эти правила могут быть противоречивыми или избыточными.

Различие между декларативным и процедурным представлением можно выразить как различие между «знать что» и «знать как». Каждое представление имеет свои достоинства и недостатки.

В заключение необходимо отметить, что деление моделей представления знаний на декларативные и процедурные весьма условно, так как стремление наиболее полно использовать достоинства обоих видов представления знаний привело к разработке смешанных представлений, т.е. декларативных представлений с присоединенными процедурами (например, фреймовые модели и модели, использующие расширенные семантические сети).

Оптимальное решение задачи выбора: первый прототип реализуется на специализированных средствах, и в случае достаточной эффективности этих средств на них могут быть написаны действующий прототип, и даже промыш-

ленная система. Однако в большинстве случаев прототип следует "переписать" на традиционных средствах.

1.2 Классификация ЕЯ систем

В разделе 1.1 были определены и в общем виде рассмотрены основные функциональные компоненты ЕЯ-систем: ведение диалога, понимание высказываний и генерация высказываний. В зависимости от назначения прикладных систем, в состав которых входят конкретные реализации ЕЯ-систем, задачи, решаемые отдельными функциональными компонентами (как по постановке, так и по методам решения), могут в значительной степени варьироваться. Исходя из этого, а также с учетом истории развития ЕЯ-систем, различают следующие основные классы систем общения:

- интеллектуальные вопрос - ответные системы;
- системы общения с базами данных;
- диалоговые системы решения задач;
- системы обработки связных текстов;
- системы машинного перевода.

Исторически ЕЯ-системы происходят от информационно-поисковых систем, с одной стороны, и систем машинного перевода - с другой. Поэтому на начальном этапе ЕЯ-системы представляли собой макеты информационно-поисковых систем, демонстрирующие принципиальную возможность ввода данных (фактов) и обработки запросов на естественном языке. Такие системы часто назывались интеллектуальными вопрос-ответными системами. Название можно, по-видимому, объяснить стремлением их разработчиков подчеркнуть, что в отличие от обычных информационно-поисковых систем и систем машинного перевода того времени в данных системах широко используются концепции, выработанные в исследованиях по искусственному интеллекту.

Основное внимание при разработке интеллектуальных вопрос-ответных систем уделялось не столько возможностям их практического использования в реальных задачах, сколько развитию моделей и методов, позволяющих осуществлять перевод ЕЯ-высказываний, относящихся к узким и заранее фиксированным проблемным областям, в формальное представление, а также обратный перевод. Накопленный опыт разработки интеллектуальных вопрос-ответных систем позволил, с одной стороны, углубить понимание процесса ЕЯ-общения и, следовательно, поставить новые проблемы (в том числе и специфичные для общения в различных классах проблемных областей), требующие дальнейшей проработки, а с другой – оценить перспективы практического применения ЕЯ-систем.

Первые предпосылки для практического использования ЕЯ-систем создало появление баз данных (БД). В связи с этим возникла проблема обеспечения доступа к информации, хранящейся в БД, широкому классу неподготовленных конечных пользователей, к которым относят специалистов в той или иной предметной области, как правило, не обладающих знаниями о логической структуре БД, о системе представления информации в БД и не умеющих пользоваться формализованными языками запросов. Для решения этой проблемы

стали создаваться системы общения с базами данных, основная задача которых (в простейшем случае) заключается в выполнении перевода запросов неподготовленных конечных пользователей с ЕЯ на формализованные языки запросов к БД.

Диалоговые системы решения задачи в отличие от систем общения с БД берут на себя не только функции ЕЯ-доступа к БД, но и функции интеллектуального монитора, обеспечивающего решение заранее определенных классов задач (например, планирование путешествий, боевых операций, составление контрактов и т. п.). В этом случае разбиение задач на подзадачи и распределение ролей между участниками, т. е. определение, кто из участников (пользователь или система) решает ту или иную подзадачу, осуществляется не пользователем (как в случае применения систем общения с БД), а диалоговой системой.

Решение подзадач, «порученных» системе, может осуществляться как на основе использования собственных знаний и механизмов вывода, так и в результате обращения к прикладным программам и пакетам, не входящим в состав ЕЯ-системы. Основным направлением практического использования ЕЯ-систем данного класса является реализация ЕЯ-общения с экспертными системами.

Возникновение последнего класса ЕЯ-систем — систем обработки связных текстов, обусловлено возрастанием объема хранимой в ЭВМ текстовой информации (газетные статьи, сообщения о различных событиях, патенты, авторские свидетельства и т. п.) и необходимостью извлечения из нее разнообразных сведений (например, о структуре некоторых объектов, о действующих лицах некоторых событий, о мотивах их поступков и т. д.).

Каждый из классов ЕЯ-систем обладает специфическими особенностями, которые хорошо заметны при рассмотрении характера задач, решаемых основными функциональными компонентами этих систем (рис. 3).

Приведенная выше классификация ЕЯ-систем охватывает лишь функционально полные системы, т. е. такие, в которых представлены все основные функциональные компоненты.

Однако помимо функционально полных систем ведется интенсивная разработка систем, которые можно назвать фрагментарными. Цель их создания — исследование или реализация новых методов решения достаточно узких задач (например таких, как анализ, интерпретация, определение целей пользователя и т. п.).

Наибольший интерес в рамках данной работы имеют системы обработки связных текстов. Системы данного класса моделируют процесс понимания законченных описаний определенных фрагментов действительности (историй, рассказов, эпизодов и т. п.), выраженных в виде текста на естественном языке, т. е. последовательности связанных друг с другом предложений. Понимание текста трактуется как извлечение из него всей существенной с точки зрения системы информации и присоединение ее к собственной базе знаний. После этого система может отвечать на вопросы относительно фактов, событий, явлений и прочих сущностей, которые явно или косвенно описаны во введенных текстах.

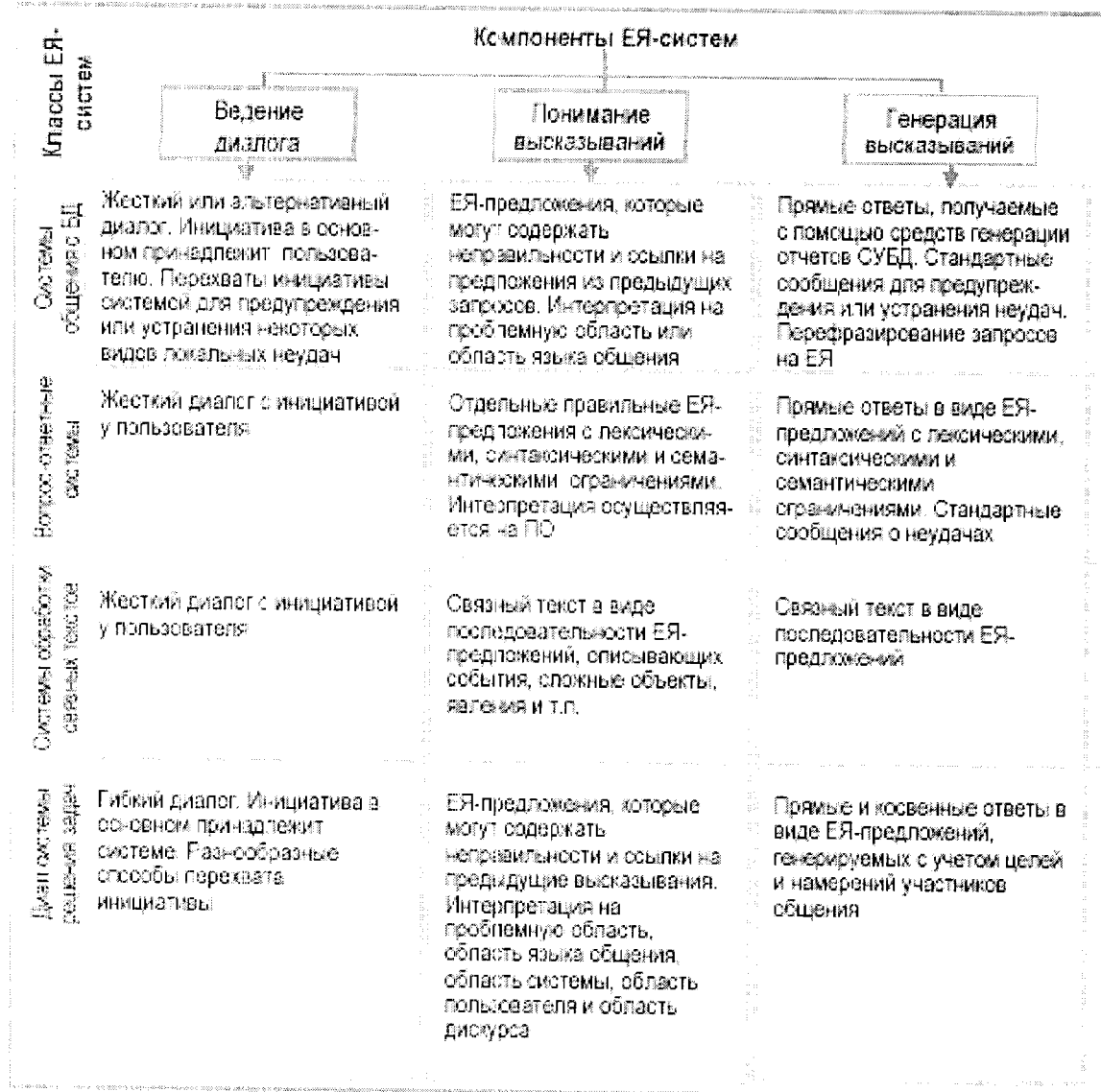


Рисунок 3 – Сравнительная характеристика классов ЕЯ-систем

Очевидно, что в практическом плане модели и методы, развиваемые в системах обработки связных текстов, могут быть полезны при создании интеллектуальных систем автоматического индексирования и реферирования.

Задача понимания связных текстов превосходит по сложности задачи, решаемые ЕЯ-системами ранее рассмотренных классов. Наиболее сложными для понимания являются тексты, описывающие взаимоотношения и поступки активных действующих лиц. Для их понимания система должна обладать громадным объемом знаний о мире, иметь совершенные механизмы вывода. Поэтому в настоящее время системы обработки связных текстов находятся на стадии разработки экспериментальных образцов, которые используются для исследования и оценки методов решения этой крайне сложной и многогранной задачи.

В заключение можно сказать, что благодаря модульности структуры ЕЯ-систем и, как правило, универсальному (т. е. не зависящему от специфики прикладных областей) характеру языка внутреннего представления, фрагментарные системы могут успешно использоваться в качестве отдельных функциональных блоков, встраиваемых (хотя бы на логическом уровне) в различные функционально полные ЕЯ-системы.

1.3 Методы реализации компонента понимания высказываний ЕЯ-систем

В рамках данной работы из всех функциональных компонентов системы наибольший интерес представляет организация работы компонента понимания высказывания. Рассмотрим ее более подробно.

Понимание высказываний включает анализ и интерпретацию.

В методах анализа обычно выделяют анализ слов, предложений и текстов. Анализ слов сводится к морфологическому анализу и обнаружению и исправлению орфографических ошибок. Цель морфологического анализа состоит в получении основ (под основой понимается словоформа с отсеченным окончанием) со значениями грамматических категорий (часть речи, род, число, падеж) для каждой из словоформ высказывания, поступившего на вход системы.

Анализ предложений обычно сводится к синтаксическому и семантическому анализу, выполняемому отдельным функциональным блоком анализатором. Наиболее распространенные методы анализа предложений, так же как и методы морфологического анализа предназначены для обработки только «правильных», т.е. не содержащих отклонений от грамматических норм, предложений. Обычно при описании анализаторов основное внимание уделяется распределению функций между синтаксическим и семантическим анализом и порядку их выполнения. Однако с точки зрения современных требований к ЕЯ-системам более важным является вопрос о том, насколько существующие анализаторы могут быть приспособлены к обработке «неграмматичностей», т.е. характерных для диалога между людьми высказываний с отклонениями от грамматической нормы (лексические и грамматические ошибки, пропуски, повторы, шумы, эллипсис, идиомы и т.п.). По этому параметру различают следующие типы анализаторов.

1.3.1 Традиционные анализаторы

Наиболее распространенным способом анализа ЕЯ-предложений является разбор сверху вниз, слева направо, основанный на некоторой фиксированной грамматике. Хрупкость таких анализаторов обусловлена тем, что их алгоритм осуществляет поиск сверху вниз среди разборов, допускаемых грамматикой, того разбора, который соответствует обрабатываемому предложению. Если некоторый частный разбор при сопоставлении ему очередного слова противоречит используемой грамматике, то для анализатора это сигнал того, что на более раннем этапе поиска сделан ошибочный выбор. Таким образом, неудача на некотором шаге разбора является сигналом для выбора очередного из возможных разборов, т.е. принципиальные затруднения возникают при обработке предложений, содержащих отклонения от грамматики.

Один из возможных подходов к преодолению хрупкости традиционных анализаторов состоит в одновременном применении нескольких подграмматик. Каждая из подграмматик предназначена для анализа частных конструкций какого-либо одного вида. Применение подграмматик осуществляется независимо, поэтому неудача одной подграмматики не влияет на возможности других.

1.3.2 Концептуальные анализаторы

Анализаторы данного типа используют методы разбора, направляемые значениями базовых событий, обнаруженных в анализируемых предложениях. Наиболее известными разновидностями данного подхода являются анализаторы, основанные на модели концептуальной зависимости. Концептуальные анализаторы не разрабатывались специально для анализа неграмматических предложений. Однако заложенные в них идеи в принципе позволяют этим алгоритмам работать в условиях пропусков и повторов слов. Такие системы по сути обладают иммунитетом к ошибкам, так как они игнорируют непонятные им слова, а понятные приспособливают (даже если в них есть ошибки) к базовым событиям обрабатываемого предложения.

1.3.3 Анализаторы, использующие сопоставление по образцам

Анализаторы данного класса основаны на том, что в простейшем случае анализ сводится к сопоставлению предложения с некоторым множеством образцов, представляющих собой последовательности из одного или нескольких слов. Многие методы анализа, основанные на сопоставлении по образцам, содержат в образце не только константы, но и переменные. При этом предполагается, что переменные образца могут сопоставляться с любой строкой символов. Возможности методов анализа, основанных на сопоставлении по образцам, можно увеличить, используя частичное сопоставление предложения с несколькими образцами. Но при таком подходе возникают некоторые ограничения. Если предложение сопоставлять с несколькими образцами, каждый из которых подходит к отдельному фрагменту предложения, то возникает задача определения границ фрагментов. Кроме того, после сопоставления одного предложения с несколькими образцами возникает задача получения общей интерпретации всех фрагментов данного предложения.

1.3.4 Анализаторы, использующие разнообразные методы

Недавние исследования показали, что использование в одном анализаторе нескольких специфических методов позволяет обеспечить гибкость процесса анализа, необходимую для обработки неграмматических конструкций.

Методы анализа связного текста (дискурса)

Связность дискурса достигается как лингвистическими средствами, имеющими языковое выражение, так и экстралингвистическими (ситуационными) средствами – «умолчаниями», не имеющими языкового выражения и основанными на общности знаний участников общения о цели общения и проблемной области. На этапе анализа связного текста, как правило, решается задача выявления связей между предложениями, выражаемых лингвистическими средствами, а на этапе интерпретации – ситуационными.

К основным лингвистическим связям предложений относятся ссылки и эллипсис. В проблеме установления ссылок могут быть выявлены две задачи: поиск в предыдущих предложениях (контексте) сущности (референта), обозначаемой данной ссылкой; определение соответствия между референтом и ссылкой. Простейшим методом решения первой задачи является поиск референта в заданном количестве предыдущих предложений. Однако отсутствие критерия

для определения количества просматриваемых предыдущих предложений приводит на практике как к увеличению времени поиска, так и к ошибкам в установлении ссылок. Решением задачи является тривиальным для простейших видов ссылок (т.е. в случае тождества референта и ссылки) и весьма трудным для случаев несовпадения референта и ссылки.

Задача обработки эллиптических конструкций решается на этапе анализа также в ограниченной постановке. Под эллипсисом понимается сжатая форма высказывания, смысл которой определяется либо предыдущими высказываниями (текстовый эллипсис), либо ситуацией, имеющей место в проблемной области (ситуативный эллипсис). С формально-грамматической точки зрения высказывания, содержащие эллипсис, выглядят как неполные (содержащие пропуски слов) предложения. На этапе анализа может быть обработан (восстановлен) только текстовый эллипсис. Сущность методов восстановления текстового эллипсиса состоит в подстановке фрагментов предыдущих высказываний в текущее высказывание, содержащее эллипсис. При этом пропущенные слова в текущем высказывании как бы восстанавливаются из текста предыдущего высказывания. Восстановление ситуационного эллипсиса осуществляется на этапе интерпретации.

Интерпретация высказываний пользователя

На этапе интерпретации решаются две основные задачи: буквальная интерпретация высказываний в контексте диалога; интерпретация на цели участников общения. Методов решения этих задач в общей постановке не существует. Однако применительно к простым проблемным областям их решение существенно упрощается.

К простым проблемным областям относятся различные области, над которыми решаются простые задачи информационного обслуживания. Такие задачи оперируют ограниченным множеством сущностей. Структура простых задач информационного обслуживания соответствует слотам фрейма, а весь вид обслуживания представляется некоторым фреймом. Работа ЕЯ-системы при обслуживании состоит в заполнении (означивании) слотов и, возможно, в выполнении некоторых манипуляций над фреймами с заполненными слотами.

Для решения обеих задач интерпретации в рамках фрейм-представлений используется единый механизм означивания фреймов. При этом структура целей участников общения определяется структурой фрейма, описывающего данную задачу, а подцели состоят в заполнении слотов, т.е. в идентификации сущностей через описания, представленные в результатах анализа входных высказываний.

В общем случае процесс идентификации некоторой сущности может иметь три исхода: однозначный – данному описанию сопоставляется единственная сущность; многозначный – описанию сопоставляется более чем одна сущность; неудовлетворительный – описанию не сопоставляется ни одна сущность. Последние два исхода рассматриваются как неудачи буквальной интерпретации и служат сигналами о необходимости установления подцелей более глубокого уровня, предусматривающих устранение неудач. При этом в диалоговый компонент кроме сообщения о неудаче и типе неудачи передаются исходные данные, позволяющие сформировать (с помощью компонента генерации высказы-

ваний) действие системы по перехвату инициативы и открытии уточняющего поддиалога, преследующего новую подцель.

В случае многозначной интерпретации вместе с сообщением о неудаче передаются возможные варианты сопоставления, что позволяет системе генерировать альтернативный вопрос. При неудовлетворительной интерпретации от компонента понимания требуется указать причины, по которым рассматриваемое описание является неудовлетворительным. В большинстве случаев причины неудовлетворительной интерпретации сводятся к неудовлетворительным (неполным или неоднозначным) результатам анализа входных высказываний и к несоответствию пресуппозиций (основные знания, позволяющие понимать весь текст в целом и отдельные высказывания в частности) пользователя знанием системы.

При решении задач интерпретации важную роль играет имеющееся в системе представление общей точки зрения на то, о чем идет речь в текущий момент. Эту точку зрения часто называют фокусом (или фокусом внимания). Разделяемый участниками фокус позволяет им повысить компактность диалога за счет того, что сущности, находящиеся в фокусе, могут либо вообще не упоминаться в высказывании (эллипсис), либо упоминаться в виде кратких описаний (ссылок).

Еще раз подчеркнем, что изложенные выше методы базируются на фрейм-представлениях и широко применяются в ЕЯ-системах, ориентированных на решение простых задач информационного обслуживания. Что же касается методов интерпретации, используемых в более сложных проблемных областях (например, понимание связных текстов, описывающих разворачивающиеся по времени события со многими действующими лицами), то они находятся в стадии становления и пока не поддаются обобщенному описанию, так как сильно зависят как от условий конкретных задач, решаемых ЕЯ-системой, так и от специфики применяемых средств представления знаний.

2 ПОСТРОЕНИЕ ПИРАМИДЫ ЗНАНИЙ ПРЕДМЕТНОЙ ОБЛАСТИ

Как показано в главе 1, одной из основных трудностей при создании ЕЯ-систем является формирование структуры предметной области. Если говорить о создании конкретного ЕЯ интерфейса в конкретной предметной области, то ее выявление можно организовать как процесс извлечения экспертных знаний инженером-когнитологом с последующей концептуализацией и структуризацией выявленных знаний. Используя термины инженерии знаний, для полного описания предметной области мы должны построить ее пирамиду знаний. Подробное объяснение этого понятия, указание ее структуры приведены в разделе 2 данной главы. Возможные методы ее построения рассмотрены в разделах 3 и 4.

С другой стороны, чтобы построенная пирамида (модель) была адекватной и релевантной поставленной задаче исследования, мы должны рассмотреть процесс ее поэтапного формирования в рамках взаимной деятельности обучающего и обучаемого. Попытка этого описана в разделе 2.1.

2.1 Процесс формирования структуры предметной области в процессе обучения

Современные дидакты и педагоги различают две цели, преследуемые процессом обучения. Содержание этой цели в одном случае понимается как совокупность знаний, умений и навыков, которые должны быть переданы обучающим и усвоены обучаемыми, в другом - как комплекс умений и навыков по решению проблем и задач, которые должны быть сформированы у обучающихся под руководством обучающихся. Обратим особое внимание на неслучайность использования нами в одном случае термина "обучаемые", который традиционно применяется в рамках репродуктивного обучения, в другом – понятия "обучающиеся", преимущественно использующегося сторонниками продуктивного обучения.

В условиях репродуктивного обучения возможны как минимум три варианта симбиоза деятельности преподавания и учения. Первый – когда обучающий сообщает информацию о знаниях, умениях и навыках, а обучаемые принимают ее и запоминают. Второй – когда обучающий совершает познавательную деятельность на глазах у обучаемых с применением знаний, умений и навыков, которые необходимо передать, а обучаемые повторяют ее (доказательство теоремы). Третий – когда обучающий рассказывает о том, какой должна быть познавательная деятельность и особенно ее результаты, но сам при этом ее не совершает, а обучаемые слушают его (ситуация, когда преподаватель немецкого языка обучает тому, как правильно пользоваться глагольными формами, говоря при этом на русском языке).

В первом и третьем вариантах обучаемые могут слушать обучающего, но не слышать его. Но даже услышав то, что им сообщается, они могут принять это или не принять, запомнить или не запомнить, усвоить или не усвоить. Все это определяется актуальным состоянием потребностно-мотивационной сферы обучаемых. Следовательно, от обучающего требуется, по крайней мере, учет этого состояния, не говоря уже об управлении им.

ективного содержания обучения. На это обстоятельство обратил внимание еще Дж.Брунер, который считал, что: а) новый опыт индивиды воспринимают избирательно, в зависимости от своих потребностей и целей; б) усваивают его в соответствии с отложившимся в когнитивной структуре прежним индивидуальным и социальным опытом.

Из всего этого следует, что в системе "преподавание - учение" главным является деятельность обучающего, поскольку именно от него зависит определение конкретного содержания того или иного учебного занятия, будь то лекция, семинар, практическое или лабораторное занятие, а также подбор соответствующих возможностям обучающихся методов и средств обучения. Это означает, что фактически деятельность обучающего начинается еще задолго до аудиторного занятия, в период подготовки к нему. И главным здесь является, как известно, создание и систематизация определенного фонда знаний, который должен быть передан обучающимся. Подготовленный обучающим фонд знаний выступает в процессе обучения с психологической стороны как предмет познавательной деятельности обучающихся, или иначе как действительный мотив. Следовательно, само содержание познания в процессе обучения определяется структурой фонда знаний, а также тем, насколько эта структура может обеспечить побуждение и управление познавательной деятельности вначале самого обучающего, а затем обучающихся.

На основе сделанных выше обобщений можно заключить, что на любом учебном занятии, в том числе и на лекции, осуществляется не столько процесс передачи знаний от обучающего к обучающимся, сколько последовательная и целенаправленная, а иногда и спонтанная организация познавательной деятельности. Которая, будучи вначале монологической по форме, постепенно преобразуется по мере включения обучающихся в процесс познания в диалогическую по форме и познавательную по содержанию. Фонд знаний переходит при этом от обучающего к обучающимся. Собственно этот фонд знаний, присвоенный обучающимися, и выступает в качестве результатов обучения, выявляемых посредством традиционных и нетрадиционных диагностических процедур.

Здесь необходимо сделать некоторые уточнения относительно психологического содержания того, что мы обозначаем словом знания. Знания – это система понятий, закрепленная посредством слов. Всякое понятие имеет две стороны: объективную – значение, и субъективную – смысл. Следовательно, передача знаний со своей внутренней, психологической стороны предстает как единство двух взаимосвязанных процессов: передачи значений и передачи смыслов. Известно, что последние чрезвычайно сложно дифференцировать в процессе прямого внешнего наблюдения за ходом учебного занятия. Но, тем не менее, как показали исследования, необходимо не только различать их в сложном процессе передачи знаний, но и специально планировать с целью управления усвоением знаний в процессе обучения.

Важно иметь в виду, что необходимость дифференциации значений и смыслов диктуется различной психологической структурой процессов передачи и усвоения знаний. Причем, при обучении преподаватель, как правило, делает больший акцент на его объективную сторону, то есть на передачу системы значений, в то время как усвоение знаний в большей степени обеспечивается assi-

милацией обучающимися системы смыслов. Кроме того, усвоение системы значений приводит к формированию так называемых пассивных знаний, а усвоение системы смыслов способствует формированию знаний активных, определяющих готовность субъекта к практическим действиям, лежащих в основе успешной адаптации к процессу обучения.

В этой связи необходимо обратить внимание на то, что сама обучающая деятельность преподавателя также состоит из двух компонентов. С одной стороны, это передача обучающимся значений знаний, а с другой – формирование у них субъективного отношения к этим знаниям путем передачи их смыслов. Если первая сторона обучения сводится к формированию практических и умственных действий с учебным материалом, то вторая его сторона заключается в различных пояснениях и уточнениях процесса выполнения этих действий.

Особенно важно понимать то, что прежде чем произойдет передача смысла через систему значений, должен осуществиться процесс означивания смысла, своеобразная трансформация смысла в значение, при которой возможна частичная потеря смысла, являющаяся следствием актуализации недостаточно адекватного значения.

Многочисленный обмен значениями в процессе общения, сопровождаемый обменом смыслами, приводит к формированию новых смыслов (метапонятий), обобщенных по своему содержанию, и далее – обмену их посредством новых значений, что в свою очередь порождает новые смыслы. Этот процесс может длиться достаточно долго. Переход его в "покоящееся свойство", новый осознанный образ, аккумулярованный в значении, знаменует собой рождение достаточно осознанного семантического пространства предметной области. Ее структура отражает степень восприятия и ассимиляции значений и смыслов предметной области, т.е. уровень владения терминологическим и функциональным ее аппаратом [5].

2.2 Извлечение знаний о структуре предметной области

Процесс извлечения знаний противоположен по цели процессу их усвоения, но совсем не совпадает по методологии. Инженер-когнитолог, извлекающий знания эксперта, должен обладать определенными навыками и несомненным лингвистическим чутьем, т.к. технология получения и структуризации знаний происходит в непосредственном общении когнитолога и эксперта. Способ и эффективность общения определяется лингвистическим аспектом.

Лингвистический аспект касается исследований языковых проблем, так как язык — это основное средство общения в процессе извлечения знаний.

В инженерии знаний можно выделить три слоя лингвистических проблем: «общий код», понятийная структура, словарь пользователя [6].

«Общий код» решает проблему языковых ножиц между профессиональной терминологией эксперта и обыденной литературной речью аналитика и включает следующие компоненты:

- общенаучная терминология;
- специальные понятия из профессиональной литературы;
- элементы бытового языка;

- неологизмы, сформированные за время совместной работы;
- профессиональный жаргон и др.

Различие языков аналитика и эксперта обуславливает тот «языковой барьер» или «языковые ножницы» в их общении.

Эти два языка являются отражением «внутренней речи» эксперта и аналитика, поскольку большинство психологов и лингвистов считают, что язык – это основное средство мышления наряду с другими знаковыми системами «внутреннего пользования» (универсальный семантический код - УСК, языки «смысла», концептуальные языки и др. [Информатика]).

Язык аналитика состоит из трех компонентов:

- общенаучной терминологии из его «теоретического багажа»;
- терминов предметной области, которые он почерпнул из специальной литературы в период подготовки;
- бытового разговорного языка, которым пользуется аналитик.

Язык эксперта включает в себя:

- общенаучную терминологию;
- специальную терминологию, принятую в предметной области;
- бытовой язык;
- неологизмы, созданные экспертом во время работы, то есть его профессиональный жаргон.

Если считать, что бытовой и общенаучный языки у двух участников общения примерно созпадают (хотя реально объем второго компонента у эксперта существенно больше), то некоторый общий язык, или код, который необходимо выработать партнерам для успешного взаимодействия, будет складываться из следующих потоков: во-первых, бытовой язык и общенаучные термины как аналитика, так и эксперта, во-вторых, специальная терминология эксперта и специальные термины из литературы инженера по знаниям.

В дальнейшем этот общий код преобразуется в некоторую понятийную (семантическую) сеть, которая является прообразом поля знаний предметной области.

Выработка общего кода начинается с выписывания аналитиком всех терминов, употребляемых экспертом, и уточнения их смысла. Фактически это составление словаря предметной области. Затем следует группирование терминов и выбор синонимов (слов, означающих одно и то же). Разработка общего кода заканчивается составлением словаря терминов предметной области с предварительной группировкой их по смыслу, то есть по понятийной близости (это уже первый шаг структурирования знаний).

На этом этапе аналитик должен с большим вниманием отнестись ко всем специальным терминам, пытаясь максимально вникнуть в суть решаемых проблем и терминологию. Освоение аналитиком языка предметной области – первый рубеж на подступах к созданию адекватной базы знаний.

Таким образом, лингвистический аспект предполагает изучение и управление процессом разработки специального промежуточного языка, необходимого для взаимодействия инженера по знаниям и эксперта.

Понятийная структура.

Проблемы формирования понятийной структуры представляют следующий слой лингвистического аспекта извлечения знаний. Особенности формирования понятийной структуры обусловлены постулатом когнитивной психологии о взаимосвязи понятий в памяти человека и наличии семантической сети, объединяющей отдельные термины во фрагменты, фрагменты в сценарии и т.д. [7]. Построение иерархической сети понятий, так называемой «пирамиды знаний», – важнейшее звено в проектировании интеллектуальных систем.

Большинство специалистов по ИИ и когнитивной психологии считают, что основная особенность естественного интеллекта и памяти в частности – это связанность всех понятий в некоторую сеть. Поэтому для разработки базы знаний нужен не словарь, а «энциклопедия» [8], в который все термины объяснены в словарных статьях со ссылками на другие термины.

Таким образом, лингвистическая работа инженера по знаниям на данном слое проблем заключается в построении таких фрагментов с помощью «сшивания» терминов. Фактически эта работа является подготовкой к этапу концептуализации, где это «шитье» (по Шенку – КОП, концептуальная организация памяти [9]) приобретает некоторый законченный вид.

При тщательной работе аналитика и эксперта в понятийных структурах начинает просматриваться иерархия понятий. Такие структуры имеют важное гносеологическое и дидактическое значение и по сути своей являются онтологиями. Следует заметить, что эта иерархическая организация хорошо согласуется с теорией универсального предметного кода (УПК) [9, 10], согласно которой при мышлении используются не языковые конструкции, а их коды в форме некоторых абстракций, что в общем согласуется с результатами когнитивной психологии [7, 9].

Иерархия абстракций – это глобальная схема, которая может быть положена в основу концептуального анализа структуры знаний любой предметной области. Лингвистический эквивалент иерархии – иерархия понятий, которую необходимо построить в понятийной структуре, формируемой инженером по знаниям.

Следует подчеркнуть, что работа по составлению словаря и понятийной структуры требует лингвистического «чутья», легкости манипулирования терминами и богатого словарного запаса аналитика, так как зачастую он вынужден сам разрабатывать словарь признаков. Чем богаче и выразительнее получается код, тем более полная база знаний.

Аналитик вынужден все время помнить о трудности передачи образов и представлений в вербальной форме. Полезными тут оказываются свойства многозначности слов естественного языка. Часто инженеру по знаниям приходится подсказывать слова и выражения эксперту, и такие новые лексические конструкции оказываются полезными.

Способность к словесной интерпретации зависит и от пола аналитика. Установлено, что традиционно женщины придают большую значимость невербальным компонентам общения, а в вербальных имеют более обширный алфавит признаков. И вообще, существуют половые различия восприятия не только в бытовой сфере, что очевидно, но и в профессиональной. Следовательно, у

эксперта-мужчины и у эксперта-женщины могут существенно отличаться алфавиты для вербализации признаков воспринимаемых объектов.

Словарь пользователя.

Лингвистические результаты, относящиеся к слоям общего кода и понятийной структуры, направлены на создание адекватной базы знаний. Однако часто профессиональный уровень конечного пользователя не позволяет ему применить специальный язык предметной области в полном объеме.

Неожиданным для начинающих разработчиков являются проблемы формирования отдельного словаря для создания дружественного интерфейса с пользователем ЭС, исследуемые в этом слое. Необходимы специальные приемы, увеличивающие прозрачность и доступность системы. Для разработки пользовательского интерфейса требуется дополнительная доработка словаря общего кода с поправкой на доступность и «прозрачность» системы.

3.2 Психосемантика и инженерия знаний

Большинство систем приобретения знаний облегчают сложный и трудоемкий процесс формирования баз знаний и реализуют прямой диалог с экспертом. Однако выявляемые таким образом структуры знаний часто отражают лишь поверхностную составляющую знаний эксперта, не затрагивая их глубинной структуры. Этим же недостатком обладает большинство методов непосредственного извлечения знаний [6].

3.2.1 Психосемантика и ее применение в процессе извлечения знаний.

Для извлечения глубинных пластов экспертного знаний можно воспользоваться методами психосемантики – науки, возникшей на стыке когнитивной психологии, психолингвистики, психологии восприятия и исследований индивидуального сознания. Психосемантика исследует структуры сознания через моделирование индивидуальной системы знаний и выявление тех категориальных знаний, которые могут не осознаваться (латентные, имплицитные, или скрытые).

Психосемантика сразу была оценена специалистами в области искусственного интеллекта как перспективный инструмент, позволяющий реконструировать семантическое пространство памяти как психологическую модель глубинной структуры знаний эксперта. Такие методы обращаются непосредственно к испытуемому. В их практической реализации перед испытуемым ставится задача оценить «сходство знаний» с помощью некоторой градуированной шкалы, например от 0 до 9 или от +2 до –2. В этом случае исследователь получает в руки численно представленные стандартизованные данные, легко поддающиеся статистической обработке.

В психологии семантические пространства выступают как модель категориальной структуры индивидуального сознания. При концептуальном анализе знаний структуру семантического пространства эксперта можно считать основой для формирования поля знаний. При этом отдельные параметры семантического пространства соответствуют различным компонентам поля (размерность пространства соотносится со сложностью поля, выделенные понятийные структуры с метапонятиями, содержательные связи между понятиями-стимулами – это суть отношения и т.д.).

В основе построения семантических пространств определенных предметных областей, как правило, лежит статистическая процедура, позволяющая группировать ряд отдельных признаков описания в более емкие категории-факторы. Это делается с помощью различных форм субъективного шкалирования, таких как факторный анализ, многомерное шкалирование или кластерный анализ, что позволяет группировать ряд отдельных признаков в более емкие категории-факторы. Говоря на языке поля знаний, это – построение концептов более высокого уровня абстракции. При геометрической интерпретации семантического пространства значения отдельного признака отображаются как точка или вектор с заданными координатами внутри n -мерного пространства, координатами которого выступают выделенные факторы. Построение семантического пространства, таким образом, включает переход к описанию предметной области на более высоком уровне абстракции, то есть переход языка, содержащего большой алфавит признаков описания, к более емкому языку концептуализации, содержащему меньшее число концептов и выступающему своеобразным метаязыком по отношению к первому.

В зависимости от опыта и профессиональной компетентности испытуемых размерность пространства и расположение в ней первичных концептов может существенно варьироваться. Так, почти все эксперименты показывают, что с увеличением уровня профессионализма эксперта размерность пространства уменьшается. Эта особенность семантических пространств может быть использована на стадии контроля в процессах обучения, при тестировании экспериментов и пользователей.

При анализе индивидуальных семантических пространств выявляются вопросы, которые не усвоены и не уложились в систему. Контроль структуры знаний проводится на основе сопоставления семантических пространств хороших специалистов и новичков (студентов, молодых специалистов). Степень согласованности семантических пространств (их размерность, признаки и конфигурации понятий) будет определять уровень знаний испытуемого.

Однако здесь необходимо учесть, что семантические пространства двух квалифицированных специалистов могут быть разными, так как содержат индивидуальные различия восприятия, так как содержат индивидуальные различия восприятия, отражающие опыт и характер деятельности человека. Поэтому не всегда можно формально сравнить семантические пространства эксперта и новичка, следует предварительно изучить семантические пространства нескольких специалистов, а затем производить сравнение.

В работе [10] описан подобный эксперимент. Были получены когнитивные структуры знаний опытного летчика-истребителя и пилота-новичка, с использованием двух методов: многомерного шкалирования MDS и сетевого шкалирования со взвешенными связями Pathfinder. Оба алгоритма основаны на использовании оценок психологической близости. Опытный пилот и пилот-новичок оценивали все возможные парные сочетания 30 связанных с полетом понятий, приписывая числа от 0 до 9 каждой паре, где 0 означал самую слабую степень связи между понятиями, а 9 – самую сильную.

Эти оценки затем обрабатывались с применением обоих алгоритмов шкалирования.

В соответствии с алгоритмом многомерного шкалирования каждый концепт, выражающий некоторое понятие, помещается в k -мерное пространство таким образом, что расстояние между точками отражает психологическую близость соответствующих концептов. Алгоритм Pathfinder строит семантическую сеть, дуги в которой могут быть ориентированными (несимметричное отношение), либо неориентированными (симметричное отношение). Оба метода обеспечивают сжатия больших объемов данных (в форме попарных оценок) к меньшему набору параметров; однако нацелены они на выявление разных свойств исследуемых структур. Если в алгоритме Pathfinder центром внимания являются локальные отношения между концептами, то алгоритм MDS обеспечивает более широкое понимание свойств метризуемого пространства концептов.

Результирующие когнитивные структуры оказались близкими для летчиков-истребителей с одинаковым уровнем опыта, но были различными для разных групп испытуемых. Авторы экспериментов обнаружили, что по когнитивной структуре, характерной для летчика-истребителя, можно установить, новичок он или опытный пилот. Наконец, проведенный ими же анализ когнитивных структур выявил наличие концептов и отношений, общих для представлений опытного специалиста и новичка, и, кроме того, ряд концептов и отношений, которые появились только в одном из представлений. Прямым развитием рассмотренной работы стала экспертная система управления воздушным боем ACES.

Аналогичное исследование механизмов накопления опыта было проведено в области программирования на ЭВМ. С помощью методов шкалирования было показано, что один из аспектов программистского опыта включает в себя организацию знаний соответственно замыслу программы, или семантике, а не в соответствии с синтаксисом.

Сетевое представление абстрактных понятий программирования на основе оценок связности концептов у программистов на основе анализа структуры семантического пространства можно разбить на три группы: новички, неопытные специалисты среднего уровня, опытные специалисты. Кроме того, исследовалась эволюция когнитивной структуры программиста по мере его продвижения от новичка до опытного программиста.

Интерпретация выявленных отношений (связей) между понятиями требует дополнительных усилий от коллектива разработчиков ЭС. Так, например, означивание связей на рисунке 4 потребовало дополнительного эксперимента, участникам которого было предложена пара понятий и поставлена задача дать словесное описание связи между понятиями пары. Результаты работы представлены в таблице 1. Таким образом, ассоциативная связь может быть превращена в семантическую.

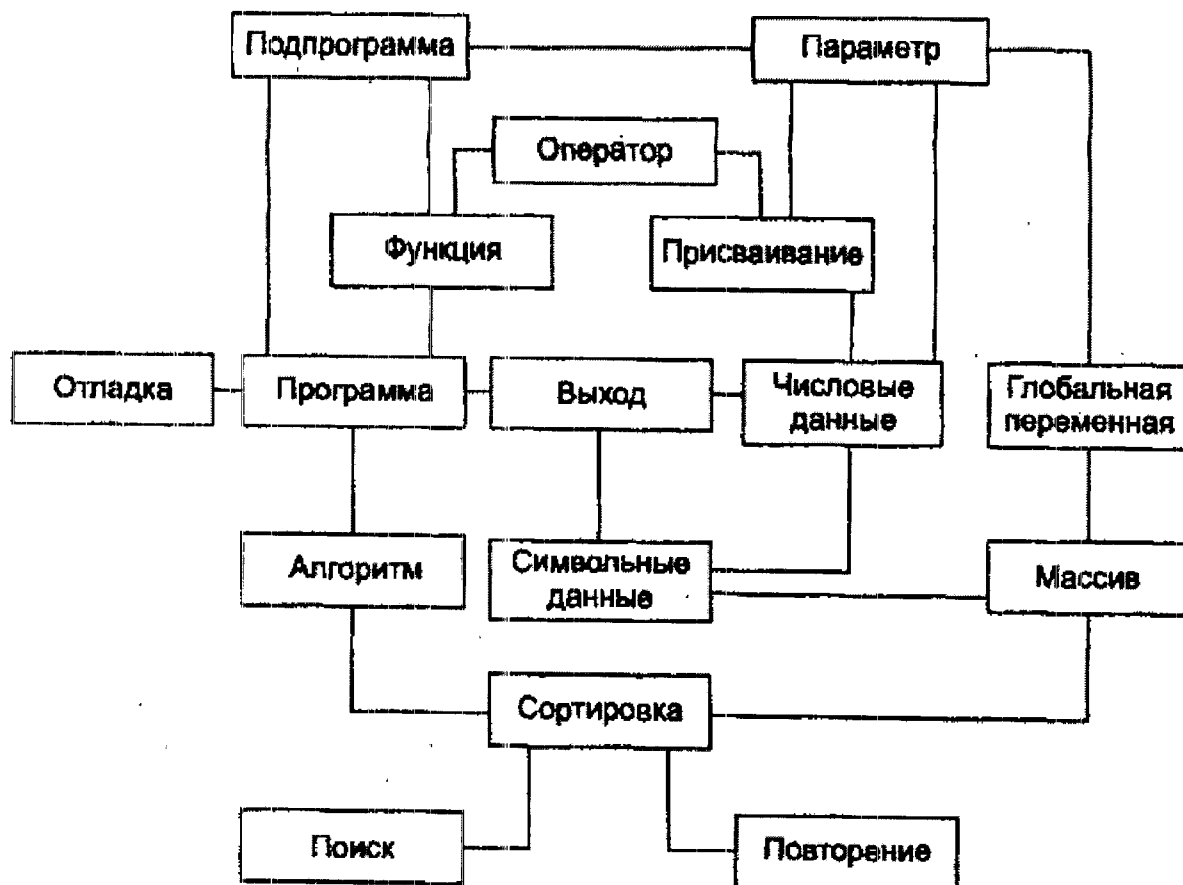


Рисунок 4 - Ассоциативная сеть эксперта.

Таблица 1. Связи между парами понятий на ассоциативной сети эксперта

Пара понятий	Значение связи
Подпрограмма – программа	Есть часть
Символьные данные – выход	Есть тип
Параметр – программа	Используется
Программа – выход	Производит
Сортировка – поиск	Включает в себя
Численные данные – параметр	Может быть
Функция -- оператор	Есть
Функция -- программа	Есть часть
Отладка – программа	Подвергается
Выход – численные данные	Состоит из
Массив – символьные данные	Может состоять из
Функция -- подпрограмма	Есть
Присваивание значений – параметр	Используется для
Массив – глобальная переменная	Может быть
Массив – численные данные	Может состоять из
Повторение – сортировка	Есть часть

Программа – алгоритм	Есть выполнение
Сортировка – алгоритм	Есть
Сортировка – численные данные	Делается на
Присваивание – оператор	Есть

Все упомянутые выше методы можно отнести к методам психологического шкалирования. Их основу составляют алгоритмы преобразования сложных структур данных в более понятную форму, которая предполагается психологически содержательной.

Формальная методология психосемантического шкалирования позволяет частично автоматизировать процесс структурирования знаний и получать «когнитивный разрез» его представлений о предметной области. Методология шкалирования позволяет выявить структуры знаний косвенным путем при получении ответов от экспертов на довольно простые вопросы (например, «насколько близки понятия X_1 и X_2 » вместо «скажите, какова связь между X_1 и X_2 , как они влияют друг на друга»).

Почти все эксперименты позволили выявить одну закономерность. Размерность семантического пространства с повышением профессионального уровня уменьшается. Этот вывод согласуется с известными положениями когнитивной психологии о том, что процесс познания сопровождается обобщением.

Построение семантического пространства обычно включает три последовательных шага:

1. выбор и применение соответствующего метода оценки семантического сходства. Этот шаг включает в себя эксперимент с испытуемыми, которым предлагается оценить общность предъявляемых стимульных признаков на некоторой шкале.

2. построение структуры семантического пространства на основе математического анализа полученной матрицы сходства. При этом происходит уменьшение числа исследуемых понятий за счет обобщения и получения генерализированных осей.

3. идентификация и интерпретация выделенных структур, кластеров или групп объектов, осей и т.д. На этом шаге необходимо найти смысловые эквиваленты, языковые «ярлыки» для выделенных структур. Здесь большое значение приобретает лингвистическое чутье и профессионализм специалиста, проводящего исследование, и экспертов-испытуемых. Часто к интерпретации привлекают группу экспертов.

Традиционно МШ применяют для решения трех типов задач:

1. Поиск и интерпретация латентных (то есть скрытых, непосредственно не наблюдаемых) переменных, объясняющих заданную структуру попарных расстояний (связей, близостей).

2. Верификация геометрической конфигурации системы анализируемых объектов в координатном пространстве латентных переменных.

3. Сжатие исходного массива данных с минимальными потерями в их информативности.

3.2.2 Простейшие методы структурирования

В качестве простейшего прагматического подхода к формированию поля знаний начинающему инженеру по знаниям можно предложить следующий алгоритм.

1. Определение входных $\{X\}$ и выходных $\{Y\}$ данных. Этот шаг совершенно необходим, так как он определяет направление движения в поле знаний – от входа к выходу. Кроме того, структура входных и выходных данных существенно влияет на форму и содержание поля знаний. На этом шаге определение может быть достаточно размытым, в дальнейшем оно будет уточняться.

2. Составление словаря терминов и наборов ключевых слов N . На этом шаге проводится текстуальный анализ всех протоколов сеансов извлечения знаний и выписываются все значимые слова, обозначающие понятия, явления, процессы, предметы, действия, признаки и т.п. При этом следует попытаться разобраться в значении терминов. Важен осмысленный словарь.

3. Выявление объектов и понятий $\{A\}$. Производится просеивание словаря N и выбор значимых для принятия решений и их признаков. В идеале на этом шаге образуется полный семантический набор терминов из какой-либо области знаний.

4. Выявление связей между понятиями. Все в мире связано. Но определить, как направлены связи, что ближе, а что дальше, необходимо на этом этапе. Таким образом, строится сеть ассоциаций, где связи только намечены, но пока не поименованы. Например, понятия «день», «ночь», «утро», «вечер» явно как-то связаны, связаны также и понятия «красный флаг» и «красный галстук», но характер связи тут существенно отличен.

5. Выявление метапонятий и детализация понятий. Связи, полученные на предыдущем шаге, позволяют инженеру по знаниям структурировать понятия и как выявлять понятия более высокого уровня обобщения (метапонятия), так и детализировать на более низком уровне.

6. Построение пирамиды знаний. Под пирамидой знаний мы понимаем иерархическую лестницу понятий, подъем по которой означает углубление понимания и повышения уровня абстракции (обобщенности) понятий. Количество уровней в пирамиде зависит от особенностей предметной области, профессионализма экспертов и инженеров по знаниям.

7. Определение отношений $\{RA\}$. Отношения между понятиями выявляются как внутри каждого из уровней пирамиды, так и между уровнями. Фактически на этом шаге даются имена тем связям, которые обнаруживаются на шагах 4 и 5, а также обозначаются причинно-следственные, лингвистические, временные и другие виды отношений.

8. Определение стратегии принятия решений (Sf). Определение стратегии принятия решений, то есть выявление цепочек рассуждений, связывает все сформированные ранее понятия и отношения в динамическую систему поля знаний. Именно стратегии придают активность знаниям, именно они «перетрачивают» модель M в поиске от X к Y .

Однако на практике при использовании данного алгоритма можно столкнуться с непредвиденными трудностями, связанными с ошибками на стадии извлечения знаний и с особенностями знаний различных предметных областей.

Тогда возможно привлечение других, более «прицельных» методов структурирования. При этом на разных этапах схемы (рисунок 5) возможно использование различных методик.



Рисунок 5 – Алгоритм структурирования знаний

Используя представленный на рисунок 5 алгоритм, инженер познаниям может испытывать необходимость в применении специальных методов структурирования на разных шагах алгоритма. При этом, естественно, для таких простых и очевидных шагов, как определение входных и выходных понятий или составление словаря, никаких искусственных методов предлагаться на будет.

3.2.2 Методы выявления объектов, понятий и их атрибутов.

Понятие или концепт – это обобщение предметов некоторого класса по их специфическим признакам. Обобщенность является сквозной характеристикой всех когнитивных психических структур, начиная от простейших сенсорных образов. Так, понятие «автомобиль» объединяет множество различных предметов, но все они имеют четыре колеса, двигатель и массу других деталей, позволяющих перевозить на них грузы и людей. Существует ряд методов выявления понятий предметной области в общем словаре терминов, который составлен на основании сеансов извлечения знаний. При этом важно выявление не только самих понятий, но и их признаков. На этом этапе определяются также интенционалы и экстенционалы понятий предметной области. интенционал очерчива-

ет понятие через взаимосвязь значимых признаков, а экстенционал – через перечисление конкретных экземпляров объекта.

Если задача выделения реальных объектов А связана только с наблюдательностью и лингвистическими способностями эксперта и инженера по знаниям, то определение метапонятий В требует от них умения проводить операции обобщения и классификации, которые никогда не считались тривиальными.

Поспелов Д.А. [11] предложил ряд подходов к созданию теории обобщения и классификации применительно к ситуационному управлению и искусственному интеллекту в целом, а также выделил ряд особенностей задач формирования понятий. Среди них особое место занимает выявление прагматически значимых признаков для формирования понятий, способствующих решению задачи.

Сложность заключается в том, что для многих понятий практически невозможно однозначно определить их признаки, это связано с различными формами репрезентации понятий в памяти человека.

Все методы выявления понятий мы разделим на:

- традиционные, основанные на математическом аппарате распознавания образов и классификации;
- нетрадиционные, основанные на методологии инженерии знаний.

Первые хорошо освещены в литературе, вторые пока менее известны.

В качестве примера приведем описанный в работе [10] интересный эксперимент.

Тридцати студентам, имеющим права на вождение автомобиля, предложили составить словарь терминов предметной области с помощью четырех методов:

1. Формирование перечня понятий (17%);
2. Интервьюирование специалистов (35%);
3. Составление списка элементарных действий (18%);
4. Составление оглавления учебника (30%).

Числа в скобках характеризуют продуктивность соответствующего метода, то есть показывают, какой процент понятий из общего выявленного списка (702 термина) был получен соответствующим методом. Для классификации понятий к участию в эксперименте были привлечены еще два человека, которые разделили 702 выявленных понятия на семь категорий методом сортировки карточек. Таблица 2 отражает численные данные концептуализации.

Таблица 2. Численные данные концептуализации предметной области

Категории	%	% от общего числа терминов, полученный соответствующим методом			
		Перечень понятий	Интервью	Список операций	Составление оглавления
Объяснение	6	5,5	7,2	7	4,9
Общие	22	43,6	18,9	36,8	4,9

правила					
Режимные правила	9	9,8	8,4	11,6	6,6
Понятия	42	15,4	38,9	8,5	77,7
Процедуры	9	5,1	9,5	25,6	1,2
Факты	9	15,0	12,5	8,9	1,2
Прочие понятия	3	2,6	4,6	1,6	3,5

В целом, результаты показали, что для выявления непосредственно концептов наиболее результативными оказались методы интервьюирования и составления оглавления учебника. Однако наибольшее число общих правил было порождено в списка действий. Таким образом, еще раз подтвердилось утверждение о том, что нет «лучшего» метода, есть методы, подходящие для тех или иных ситуаций или типов знаний.

Интересно, что число правил – продукций «если – то» – составило небольшой процент во всех четырех методах. Это говорит о том, что популярная продукционная модель вряд ли является естественной для человеческих моделей репрезентации знаний.

3.2.3 Методы выявления связей между понятиями

Концепты не существуют независимо, они включены в общую понятийную структуру с помощью отношений. Выявление связей между понятиями при разработке баз знаний доставляет инженеру по знаниям немало проблем. То, что знания в памяти – это некоторые связанные структуры, а не отдельные фрагменты, общеизвестно и очевидно. Тем не менее основной упор в существующих моделях представлений знаний делается на понятий, а связи вводят весьма примитивные, в основном причинно-следственные.

В последних работах по теории ИИ все большее внимание уделяется взаимосвязности структур знаний. Так в работе [8] введено понятие сценария (script) как некоторой структуры представления знаний. Основу сценария составляет КОП (концептуальная организация памяти) и метаКОПы – некоторые обобщающие структуры.

Сценарии, в свою очередь, делятся на фрагменты или сцены (chunks). Связи между фрагментами – временные или пространственные, внутри фрагмента – самые различные: ситуативные, ассоциативные, функциональные и т.д.

Все методы выявления таких связей можно разбить на две группы:

- формальные;
- неформальные, основанные на дополнительной работе с экспертом.

Неформальные методы выявления связей придумывает инженер по знаниям для того, чтобы вынудить эксперта указать явные и неявные связи между понятиями. Наиболее распространенным является метод «сортировка карточек» в группы, широко применяемый и для формирования понятий. Другим неформальным методом является построение замкнутых кривых. В этом случае эксперта просят обвести замкнутой кривой связанные друг с другом понятия. Этот

метод может быть реализован как на бумаге, так и на экране дисплея. В этом случае можно говорить о применении элементов когнитивной графики. [12].

После того, как определены связи между понятиями, все понятия как бы распадаются на группы. Такого рода группы представляют собой метапонятия, присвоение имен которым происходит на следующей стадии процесса структурирования.

3.2.4 Методы выделения метапонятий и детализация понятий (пирамида знаний)

Процесс образования метапонятий, то есть интерпретации групп понятий, полученных на предыдущей стадии, как и обратная процедура – детализация (разукрупнение) понятий, – видимо, принципиально не поддающиеся формализации операции. Они требуют высокой квалификации экспертов, а также наличия способностей к «наклеиванию» лингвистических ярлыков.

Если для повседневных знаний и выделенных в них концептах обозначения таких знаний процедура определения метапонятий достаточно тривиальна (рисунок 6), то в реальных предметных областях эта задача оказывается весьма трудоемкой. При этом независимо от того, формальными или неформальными методами были выявлены понятия или деталь понятий, присвоение им имен которым или интерпретация их – всегда неформальный процесс, в котором инженер по знаниям просит эксперта дать название некоторой группе понятий или отдельных признаков.

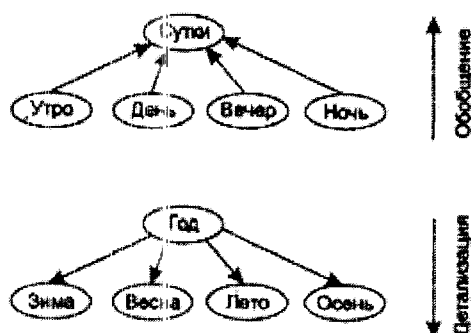


Рисунок 6. Обобщение и детализация понятий

Это не всегда удается. Так, в системе АВАНТЕСТ при образовании метапонятий, полученных методами кластерного анализа, интерпретация заняла несколько месяцев и не может считаться удовлетворительной. Это связано с тем, что формальные методы иногда выделяют «искусственные» концепты, в то время как неформальные обычно – практически используемые и поэтому легко узнаваемые понятия.

Методы построения пирамиды знаний обязательно включают использование наглядного материала – рисунков, схем, кубиков. Уровни пирамиды чаще возникают в сознании инженера по знаниям именно как некоторые образы.

Построение пирамиды знаний может быть основано и на естественной иерархии предметной области, например связанной с организационной структурой предприятия или с уровнем компетентности специалистов (рисунок 7).

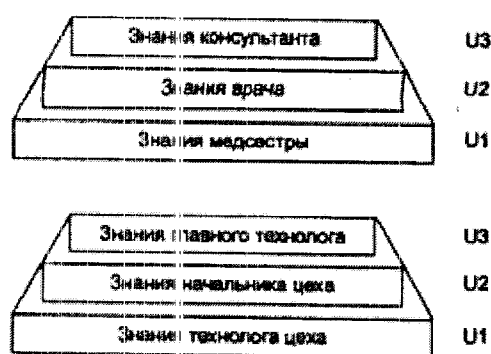


Рисунок 7. Пирамиды знаний

3.2.5 Методы определения отношений

Если на стадии 4 рисунка 5 мы выявили связи между понятиями и использовали их на стадиях 5 и 6 для получения пирамиды знаний, то на стадии 7 мы даем имена связям, то есть превращаем их в отношения.

В работе [13] указывается на наличие более 200 базовых видов различных отношений, существующих между понятиями. Помимо универсальных отношений (пространственных, временных, причинно-следственных) существуют еще и специфические отношения, присущие той или иной предметной области.

Интересные возможности к структурированию знаний добавляют системы когнитивной графики [12]. Так, в системе OPAL эксперт может манипулировать на экране дисплея изображениями простейших понятий и строить схемы лечения заболеваний, обозначая отношения явными линиями, которые затем именуются.

Реализованная в рамках данного исследования методология опирается на современные представления о структуре человеческой памяти о формах репрезентации в ней информации [9].

Скудность методов структурирования объясняется тем, что методологическая база инженерии знаний только закладывается, а большинство инженеров по знаниям проводит концептуализацию, руководствуясь наиболее дорогими и неэффективными способами – «проб и ошибок» и «по наитию», то есть исходя из соображений здравого смысла.

3.3 Многомерные методы обработки данных. Многомерное шкалирование

3.3.1 Назначение метода. Основная цель многомерного шкалирования (МШ) – выявление структуры исследуемого множества объектов. Под структурой понимается набор основных факторов (в данном случае – шкал), по которым различаются и могут быть описаны эти объекты. Исходной информацией для МШ являются данные о различии или близости объектов [14].

В психологии чаще всего исходными данными для МШ являются субъективные суждения испытуемых о различии или сходстве стимулов (объектов). Центральное положение МШ заключается в том, что в основе таких суждений лежит ограниченное число субъективных признаков (критериев), определяющих различие стимулов, и человек, вынося свои суждения, явно или неявно учитывает эти критерии. Основываясь на этом положении, решается главная

задача МШ – реконструкция психологического пространства, заданного небольшим числом измерений-шкал, и расположение в нем точек-стимулов таким образом, чтобы расстояния между ними наилучшим образом соответствовали исходным субъективным различиям. Таким образом, шкала в МШ интерпретируется как критерий, лежащий в основе различий стимулов.

Геометрические представления МШ основаны на аналогии между понятием различия в психологии и понятием расстояния в пространстве. Чем более субъективно сходны между собой два объекта, тем ближе в реконструируемом пространстве признаков должны находиться соответствующие этим объектам точки. Исходя из такой дистанционной модели, по субъективным данным о различии одного объекта от другого реконструируется их взаимное расположение в пространстве нескольких признаков. Эти признаки трактуются как субъективные шкалы – критерии, которыми пользуется человек при различении объектов. А расстояние между объектами в этом пространстве есть определенная функция от исходных оценок различия.

Общая схема МШ формально может быть представлена следующим образом. На основе суждений экспертов (испытуемых) в отношении интересующих исследователя объектов вначале составляется симметричная матрица попарных различий (или матрицы – по одной для каждого эксперта). Допускается и использование данных о предпочтениях, содержащих упорядочивание каждым экспертом совокупности объектов по степени их предпочтения. Сравнимаемыми объектами могут быть члены коллектива, предметы домашнего обихода, литературные отрывки, цветовые оттенки и т. д. Модель МШ предполагает, что эксперт производит сравнение, осознанно или нет пользуясь одним или несколькими признаками этих объектов. В отношении сотрудников подразделения такими признаками могут быть должностной статус, профессионализм, доброжелательность и т. д.

В процессе МШ определяется, сколько признаков-шкал необходимо и достаточно для построения координатного пространства и размещения в нем точек-объектов. Если δ_{ij} – это оценка экспертом различия между объектами i и j , а число признаков, которыми пользуется эксперт при сравнении, – K , то задача многомерного шкалирования сводится к определению всех x_{ik} и x_{jk} как координат этих объектов в пространстве K признаков. При этом предполагается, что число критериев, которыми пользуется эксперт, значительно меньше числа сравниваемых объектов. Если, например, i и j – сотрудники, а признак k – доброжелательность, то x_{ik} и x_{jk} – доброжелательность этих сотрудников. Важно отметить, что исследователю эмпирически даны только оценки различий δ_{ij} . Величины значений признаков x_{ik} и x_{jk} непосредственно не даны, но оцениваются в результате МШ в виде матрицы:

$$X = \begin{pmatrix} x_{11} & \dots & x_{1K} \\ \vdots & \ddots & \vdots \\ x_{P1} & \dots & x_{PK} \end{pmatrix}$$

где P – количество сравниваемых объектов, K – количество шкал.

Элементы x_{ij} указанной матрицы рассматриваются как координаты P объектов в пространстве K признаков. Пространство определено так, что чем больше

исходное различие между объектами, тем дальше друг от друга расположены объекты в этом пространстве. Каждая шкала результирующего пространства получает интерпретацию через объекты, находящиеся на противоположных полюсах шкалы.

Следует отметить, что исходными данными для МШ могут являться не только субъективные оценки различий, но и обычные данные типа «объект-признак». Но поскольку МШ предназначено для анализа различий, то для данных типа «объект-признак» необходимо, во-первых, определить, что будет подлежать шкалированию – сами объекты (строки) или признаки (столбцы). Во-вторых, необходимо задать метрику различий – то, как будут определяться различия между всеми парами изучаемых элементов.

Выбирая МШ, исследователь должен отдавать себе отчет в том, что это довольно сложный метод, применение которого к тому же связано с неизбежными потерями исходной информации о различии объектов. Поэтому, если задача исследования ограничивается классификацией объектов и нет оснований полагать, что эта классификация обусловлена небольшим числом независимых причин – критериев различий, то целесообразнее воспользоваться более простым методом – кластерным анализом

Если от каждого эксперта получена матрица попарных различий P объектов, то для таких данных используется многомерное шкалирование индивидуальных различий. При этом предполагается, что существует общее групповое пространство координат объектов в пространстве K общих признаков. Эксперты же отличаются тем, какой «индивидуальный вес» каждый из них придает тому или иному из K признаков при сравнении объектов. Соответственно, помимо групповой матрицы координат объектов, результатом этого варианта МШ является матрица индивидуальных весов размерностью $K \times N$.

Если эксперты не оценивают различия, а упорядочивают объекты по степени предпочтения, то применяется модель предпочтения. Исходными данными для МШ предпочтений является матрица размером $N \times P$, где N строк соответствуют экспертам, а P – это столбцы, содержащие ранговые места предпочтений объектов экспертами.

В дистанционной модели предпочтений каждый эксперт характеризуется координатами своей идеальной точки в пространстве K признаков. Эти координаты – такая комбинация характеристик объектов, которую эксперт считает идеальной. Результирующая матрица содержит, помимо координат объектов, и координаты идеальных точек – по одной для каждого эксперта.

Итак, МШ в своих основных трех модификациях позволяет решать три группы задач:

– Исходные данные – прямые оценки субъектом различий между стимулами или вычисленные расстояния между объектами, характеризующимися совокупностью признаков. Примером второго типа данных могут являться расстояния между ролями (объектами), вычисленные по совокупности конструкторов (репертуарные решетки Келли). МШ позволяет реконструировать психологическое пространство субъекта, как конфигурацию стимулов в осях существенных признаков, по которым эти стимулы различаются субъектом.

– Исходные данные – те же, что и в предыдущем случае субъективные различия между стимулами (оцененные прямо или вычисленные), но полученные не от одного, а от группы субъектов. Взвешенная модель индивидуальных различий позволяет получить групповое психологическое пространство стимулов в осях общих для данной группы существенных признаков. Дополнительно к этому для каждого субъекта – индивидуальные веса признаков как меру учета соответствующих точек зрения при различении стимулов.

– Исходные данные – результаты упорядочивания каждым из группы субъектов набора стимулов по степени предпочтения. Модель анализа предпочтений позволяет получить групповое психологическое пространство стимулов в осях существенных признаков и размещенные в этом же пространстве идеальные точки для каждого субъекта.

3.3.2 Меры различия

Непосредственными данными для многомерного шкалирования является информация о различиях между объектами. Эти различия должны быть определены между всеми парами объектов и иметь числовое выражение. Первое требование означает, что анализируется матрица попарных различий $N \times N$ (где N – количество изучаемых объектов). Второе условие требует измерения различия, то есть введения метрики. Существуют четыре стандартных критерия, которым должна удовлетворять мера различия, чтобы быть метрикой:

1. Симметрия. Даны два объекта: x и y . Расстояние от x до y должно быть равно расстоянию от y до x .

2. Неразличимость идентичных объектов. Расстояние между двумя идентичными объектами равно 0.

3. Различимость нетождественных объектов. Расстояние между двумя различающимися объектами не равно 0.

4. Неравенство треугольника. Даны три объекта: x, y и z . Расстояние от x до y меньше или равно сумме расстояний от x до z и от z до y (длина любой стороны треугольника меньше или равна сумме двух других сторон).

Эти очевидные требования, однако, не всегда выполняются в отношении некоторых показателей различия. Например, в случае социальных взаимных предпочтений: Петру нравится Маша, но Маше Петр не нравится. В этом случае симметрия не соблюдена, показатель не может быть метрикой, и такие данные должны быть преобразованы к симметричному виду (разумеется, с потерей информации об асимметрии).

Из критерия симметрии следует, что матрица попарных различий $N \times N$ является симметричной относительно главной диагонали. Из критерия неразличимости идентичных объектов следует, что на главной диагонали этой матрицы различий расположены нули.

Выбор меры различия является органической частью исследования, определяется его процедурой и характером получаемых данных. По процедуре исследования можно выделить три типичные ситуации: испытуемые непосредственно оценивают степень различия (близости) изучаемых объектов; имеются данные об условной или совместной встречаемости событий (объектов); имеется совокупность оценок для множества объектов – используются меры различия профилей оценок. Отметим, что для МШ чаще всего используют непосредст-

венные оценки различий. Остальные меры различий более специфичны для кластерного анализа.

3.3.3 Непосредственная оценка различий

Существует большое количество достаточно хорошо разработанных процедур субъективной оценки различия между объектами: это и прямая оценка различия пары объектов при помощи заданной шкалы (от 5 до 10 градаций) или графически, и сортировка пар по категориям различия (от идентичных до максимально различных). Основная сложность этих процедур – большое количество пар объектов, предлагаемых испытуемым для сравнения. Для N объектов должно быть оценено количество пар, равное $N(N-1)/2$. Например, если объектов 20, испытуемый должен оценить различия 190 пар объектов.

Самый простой способ организации предъявления всех возможных пар объектов – табличный $N \times N$, когда испытуемый, оценивая различия, заполняет нижний (верхний) треугольник таблицы. Однако при таком предъявлении на оценку различий между объектами влияют не только сами стимулы, но и порядок их предъявления. Речь идет о так называемых пространственных и временных искажениях. Пространственные искажения – это влияние порядка следования объектов в каждой паре (второй объект в паре воспринимается иначе, чем первый). Временные искажения – это влияние порядка следования пар (пары в начале предъявления воспринимаются иначе, чем в конце ряда). Эти искажения могут быть компенсированы путем случайного упорядочивания порядка следования как объектов в паре, так и самих пар.

При большом числе объектов можно воспользоваться неполным планом, когда каждая пара оценивается не всеми испытуемыми. В неполных планах пары делятся на подмножества, и каждый испытуемый оценивает пары только в одном подмножестве. Каждое из подмножеств оценивается одинаковым числом испытуемых.

Результаты социометрии – матрицу взаимоотношений – тоже допустимо рассматривать как вариант прямой оценки различия. Ясно, что такие матрицы – несимметричные, следовательно, получаемые данные требуют предварительной оцифровки и приведения к симметричному виду.

3.3.4 Меры различия профилей для количественных переменных

Профиль – это набор оценок (количественных признаков) объекта. Если объекты – это испытуемые, то профилем могут быть их оценки по каким-либо тестам. Сравниваемыми объектами могут быть и сами признаки, и объекты, оцениваемые испытуемыми, например, по степени предпочтения. Тогда профилем будет совокупность индивидуальных значений соответственно для признака или оцениваемого объекта. Таким образом, меры различия профилей могут применяться к наиболее типичным психологическим данным – типа «объект-признак».

Меры расстояния – наиболее специфичные показатели различия именно для профилей, то есть множества K признаков, измеренных для каждого объекта. Различие между двумя объектами, выраженное мерой расстояния, показывает, насколько в среднем различаются совокупности оценок для двух объектов, или

насколько в среднем удалены друг от друга профили оценок. Другая трактовка меры расстояния – геометрическая. Это расстояние между двумя точками (объектами) в пространстве, размерность которого равна числу признаков. Значения признаков для объекта трактуются как координаты, задающие его положение в этом пространстве.

Есть два способа задания расстояний, и выбор способа зависит от задачи исследования. Обычно исследователь имеет таблицу (матрицу) данных $N \times P$, где N – строки (испытуемые, объекты), P – столбцы (признаки, оцениваемые объекты). Если исследователя интересуют различия между объектами или испытуемыми (с точки зрения, например, их классификации), то ему необходимо задавать меры различия (расстояния) между строками. Если же ему важны основания классификации признаков, то выбираются меры различия (расстояния) между столбцами.

Существует множество мер расстояния, но наиболее часто используются две: евклидово (или его квадрат) и «метрика города».

Евклидово расстояние легко представить себе как кратчайшее расстояние (по прямой) между двумя точками (объектами) в пространстве двух, трех и более координат – в соответствии с числом признаков. Наиболее подходит эта метрика для метрических данных (в шкале интервалов или отношений). Если число признаков P , то евклидово расстояние между объектами с номерами i и j выражается формулой:

$$d_{ij} = \left[\sum_{p=1}^P (X_{ip} - X_{jp})^2 \right]^{1/2},$$

где X_{ip} и X_{jp} – значения признака p для объектов с номерами i и j соответственно. В пространственном представлении эти значения трактуются как координаты объектов i и j по оси p .

«Метрика города» (синонимы – «Манхаттан», «городских кварталов» или city-block) – получила свое название по аналогии с длиной пути пешехода, движущегося от одного объекта к другому в городе. Пешеход вынужден обходить кварталы домов по ломаной линии. Каждый «квартал» при этом – это абсолютная разность между значениями двух объектов по одному из признаков. «Метрика города» больше подходит для неметрических данных (порядковых и, отчасти, бинарных) и вычисляется по формуле:

$$d_{ij} = \sum_{p=1}^P |X_{ip} - X_{jp}|$$

Закljučая обзор мер различия, следует отметить одно обстоятельство. Наиболее частый вид данных в психологии – это множество признаков, измеренных у множества испытуемых. И наиболее вероятно, что исследователь предпочтет, выбирая многомерное шкалирование или кластерный анализ, опираться на меры различия профилей. В этой ситуации исследователь должен отдавать себе отчет в том, что получаемые показатели различия между объектами не несут никакой иной информации, кроме констатации того, что субъект А по измеренным признакам более похож на субъекта В, чем на субъекта С. А информация о том, по каким признакам различия больше, а по каким – меньше, утрачивается при вычислении расстояний. Таким образом, одни и те же значения

различий между парами объектов могут быть обусловлены разницей в значениях по разным признакам. Это обстоятельство сильно затрудняет интерпретацию мер расстояния. Иначе дело обстоит, если меры различия профилей используются в отношении данных о субъективных предпочтениях, когда вычисляются меры различия между объектами предпочтения по множеству субъективных оценок (между столбцами – оцениваемыми объектами). Здесь интерпретация достаточно очевидна: объекты будут тем ближе, чем ближе в среднем их располагают субъекты при упорядочивании по степени предпочтения. Вообще говоря, если у исследователя есть выбор при планировании инструментария, для классификации объектов следует предпочесть непосредственные оценки различия, данные о предпочтениях или условные и совместные вероятности.

3.3.5 Неметрическая модель

Это основной вариант многомерного шкалирования, применяемый в настоящее время. Он лежит в основе всех остальных вариантов метода. Исходные данные для этого метода – матрица размерностью $P \times P$, каждый элемент которой – мера (оценка) различия между двумя объектами из P . Рассмотрим кратко основные математико-статистические идеи метода, необходимые для его использования на практике.

Многомерное шкалирование и исторически, как новый шаг в математике, и процессуально – как последовательность обработки данных компьютерной программой, начинается с метрического шкалирования, предложенного в 50-х годах У.Торгерсоном. В модели Торгерсона вводится жесткое предположение о том, что оценки различия между объектами равны линейному расстоянию между ними в евклидовом пространстве.

Пусть δ_{ij} – имеющаяся в распоряжении исследователя оценка различия между объектами i и j . x_{ik} и x_{jk} – координаты этих объектов по оси k , одной из осей искомого пространства размерностью K . Расстояние между объектами в искомом пространстве обозначим как d_{ij} . Тогда основное предположение Торгерсона можно выразить формулой:

$$\delta_{ij} = d_{ij} = \left[\sum_{k=1}^K (x_{ik} - x_{jk})^2 \right]^{1/2}.$$

Торгерсон показал, что при соблюдении этого условия возможен переход от исходной матрицы различий между стимулами к их координатам в пространстве K признаков. Для этого необходимо прежде всего пересчитать исходную матрицу различий в матрицу «скалярных произведений» – путем двойного центрирования, чтобы среднее значение элементов каждой строки и каждого столбца было равно 0. Элементы такой матрицы обозначим как δ^*_{ij} . Тогда, по Торгерсону, справедливо выражение:

$$\delta^*_{ij} = \sum_{k=1}^K x_{ik} x_{jk} \quad \text{или в матричной форме: } \Delta^* = X X^T,$$

где X – матрица координат стимулов, размерностью $P \times K$.

Это уравнение аналогично главному уравнению факторного анализа, и решается оно относительно X методом главных компонент с заданным числом K .

В современных алгоритмах МШ метод Торгерсона используется на этапе предварительной оценки координат объектов по матрице исходных различий. Далее следует неметрический этап, соответствующий неметричности исходных данных. На этом этапе исходят из требования соответствия рангового порядка расстояний между объектами в результирующем пространстве ранговому порядку исходных различий, то есть, используя принятые обозначения:

$$d_{ij} \rightarrow \delta_{ij}, \delta_{ij} \leq \delta_{lm} \text{ для любых } i, j, l, m \text{ -- номеров объектов.}$$

Основной мерой выполнения этого требования является специальный показатель, который называется стресс – мера отклонения итоговой конфигурации объектов от исходных оценок различия в смысле указанного требования рангового соответствия. Иногда дополнительно применяют коэффициент отчуждения тоже как меру подгонки неметрической модели к данным о различии.

Не рассматривая подробно вычислительные проблемы многомерного шкалирования, укажем, что его алгоритм направлен на нахождение оценок координат объектов, минимизирующих значение стресса. Построен этот алгоритм как градиентная процедура. Первый шаг алгоритма – получение стартовой конфигурации, как правило, методом Торгерсона. На каждом последующем шаге, или итерации, координаты стимулов изменяются в сторону уменьшения значения стресса, вычисленного на предыдущем этапе. Итерации повторяются многократно, до выполнения одного из трех заданных изначально условий: достижения минимального значения стресса; достижения минимальной разницы между последним и предыдущим значениями стресса; выполнения максимального заданного числа итераций. Уменьшая пороговые величины стресса и его изменения, увеличивая максимальное число итераций, исследователь может добиться повышения точности окончательного решения. Показателем точности является конечная величина стресса. Наиболее приемлемые величины стресса находятся в диапазоне от 0,05 до 0,2.

Одна из основных проблем, возникающих перед исследователем в МШ – это проблема размерности K . В МШ требуется предварительное определение числа шкал. Поэтому от исследователя требуется получить несколько решений в пространствах разной размерности и выбрать из них лучшее. Один из критериев размерности, применяемый для предварительной оценки числа шкал: строится график зависимости стресса от числа шкал по результатам решения в разных размерностях. Истинная размерность соответствует точке перегиба графика после резкого его спада.

Другой критерий числа шкал – абсолютная величина стресса. Если решение одномерно, то приемлемая величина стресса – менее 0,1. Если решение размерностью 2 и выше, то приемлемы значения стресса, меньшие 0,1-0,15. Однако если уровень ошибок измерения или выборки высок, то можно признать решение и с более высокими значениями стресса. Дополнительно вычисляется величина R^2 (RSQ), которая показывает долю дисперсии исходных различий (от единичной), учтенную выделенными шкалами. Чем ближе RSQ к единице, тем полнее данные шкалы воспроизводят исходные различия между объектами.

Окончательный выбор размерности решения определяется на основе критериев интерпретируемости и воспроизводимости. Тем не менее, при размерно-

сти 2 и выше, следует избегать решений с величиной стресса выше 0,2. Обычный путь для этого – повышение размерности и исключение объектов.

Результаты применения метода – таблица координат объектов в пространстве K шкал-признаков, величины стресса и RSQ, интерпретация шкал и взаимного расположения объектов по таблице координат.

3.3.6 Модель индивидуальных различий

Этот метод – расширение метода неметрического МШ путем включения в основную модель субъективных параметров. Оценки индивидуальных различий дают, наряду с координатами стимулов, количественное координатное описание субъектов, аналогичное координатному описанию стимулов, получаемому с помощью неметрического МШ. Исходными данными для МШ индивидуальных различий является N матриц оценок различий P объектов-стимулов, то есть результат оценки попарных различий P объектов каждым из N испытуемых.

Индивидуальная матрица различий для каждого испытуемого может быть получена и в результате вычисления матрицы $P \times P$ по результатам сравнения испытуемым P объектов по ряду признаков. Типичный пример последнего – обработка репертуарных решеток Келли, когда исходными данными являются оценки объектов (например, ролей) по совокупности конструкторов.

В модели предполагается, что существует общая для всех испытуемых групповая матрица координат объектов – X , размерностью $P \times K$ (P – число объектов, K – число координат). При этом для каждого испытуемого имеется индивидуальная матрица координат X , той же размерности, но по содержанию отличающаяся от групповой. Можно представить себе координаты групповой матрицы как совокупность общих (групповых) точек зрения на объекты. Тогда индивидуальная матрица будет отличаться от групповой степенью учета данным испытуемым этих общих точек зрения, иначе – индивидуальным весом общих координат.

Таким образом, предполагается, что элементы каждой субъективной матрицы координат отличаются от элементов групповой матрицы индивидуальными весовыми коэффициентами – мерами того, насколько учитываются данным субъектом общие точки зрения. А каждый субъект может быть охарактеризован набором индивидуальных весов – по одному для каждой координаты (точки зрения). Индивидуальные веса иногда называют весами важности, весами характеристик или масштабными коэффициентами. При прочих равных условиях, при увеличении индивидуального веса данной координаты разность между стимулами по этой координате вносит все больший и больший вклад в оцениваемое различие между этими стимулами.

Изложенное выше можно представить в виде формальных соотношений. Обозначим через x_{ik} координату объекта i по шкале k в общем групповом пространстве объектов. Предполагается, что соответствующий элемент субъективной матрицы координат x_{iks} для испытуемого s связан с элементом групповой матрицы соотношением $x_{iks} = x_{ik} w_{ks}$ или в матричных обозначениях: $X_s = X W_s$, где w_{ks} – вес координаты k для субъекта s ; W_s – матрица субъективных весов для

всех K координат, преобразующая групповую матрицу в индивидуальную. В соответствии с моделью индивидуальных различий исходные данные о субъективном различии объектов i и j субъектом s выражаются формулой

$$\delta_{ijs} = \left[\sum_{k=1}^K (x_{iks} - x_{jks})^2 \right]^{\frac{1}{2}} = \left[\sum_{k=1}^K w_{ks}^2 (x_{ik} - x_{jk})^2 \right]^{\frac{1}{2}}.$$

Эта формула выражает основное предположение модели МШ индивидуальных различий и связывает исходные данные о субъективных оценках различий δ с результатами применения метода: групповой матрицей координат объектов X , размерностью $P \times K$, и матрицей субъективных (индивидуальных) весов W , размерностью $N \times K$. При этом матрица K аналогична результатам неметрического МШ, а матрица W содержит N строк – для каждого субъекта и K столбцов – для каждой шкалы-координаты.

Как упоминалось, исходными данными для модели индивидуальных различий являются N матриц субъективных оценок различий между P объектами-стимулами. Мерами соответствия результата исходным данным, или мерами качества решения, являются величины стресса, как и в неметрическом МШ. Дополнительно вычисляются квадраты коэффициентов корреляции (RSQ) между фактическими и оцененными скалярными произведениями. Они показывают степень согласованности расстояний (вычисленных по результирующим координатам стимулов с учетом субъективных весов) с исходными оценками субъективных различий между стимулами. Общий стресс и общий RSQ – меры соответствия для результирующей групповой матрицы координат стимулов. Они служат, в частности, для оценки числа координат, как и в случае неметрического МШ.

Помимо этого, вычисляются аналогичные меры для каждого субъекта. Величина стресса и RSQ для субъекта – это меры соответствия групповой матрицы координат исходным данным для этого субъекта. Чем ниже величина стресса и выше RSQ, тем выше соответствие индивидуальной точки зрения групповой. Если стресс значительно превышает 0,15 и RSQ меньше 0,7, то индивидуальные данные не соответствуют групповым и этого субъекта следует рассматривать отдельно либо выделить в отдельную группу таких же субъектов.

Если величины мер соответствия удовлетворяют исследователя, он может приступить к интерпретации групповой матрицы координат стимулов и матрицы индивидуальных весов. Матрица координат стимулов интерпретируется аналогично результатам неметрического МШ: каждая шкала интерпретируется через стимулы, имеющие по этой оси наибольшие абсолютные значения координат. Визуализация матрицы координат стимулов в пространстве двух или трех измерений-шкал позволяет обнаружить содержательно важные группировки объектов.

Матрица индивидуальных весов показывает то, насколько каждый субъект учитывает или разделяет групповые точки зрения при различении объектов. Чем выше для субъекта вес одной из координат, тем существеннее для него соответствующая групповая точка зрения. Матрица индивидуальных весов задает пространство субъектов, каждая ось которого соответствует оси пространства объектов. Каждый субъект в этом пространстве характеризуется вектором из

начала координат в точку с координатами, заданными строкой матрицы. Длина каждого вектора прямо пропорциональна степени соответствия индивидуальных данных групповым и пропорциональна RSQ: чем длиннее вектор, тем в большей степени данный субъект учитывает групповые точки зрения. Группировки в пространстве субъектов соответствуют испытуемым со сходными точками зрения.

Отметим, что метод МШ индивидуальных различий предоставляет уникальную возможность сочетания идеографического и нормативного подходов в одном исследовании. Нормативный подход осуществляется путем соотнесения индивидуальных результатов с общими для группы – по индивидуальным субъективным весам общих для группы точек зрения. Одновременно возможно изучение качественного своеобразия каждой индивидуальной точки зрения – по индивидуальной матрице различий.

3.4 Многомерные методы обработки данных. Кластерный анализ

3.4.1 Назначение

Кластерный анализ решает задачу построения классификации, то есть разделения исходного множества объектов на группы (классы, кластеры). При этом предполагается, что у исследователя нет исходных допущений ни о составе классов, ни об их отличии друг от друга. Приступая к кластерному анализу, исследователь располагает лишь информацией о характеристиках (признаках) для объектов, позволяющей судить о сходстве (различии) объектов, либо только данными об их попарном сходстве (различии). В литературе часто встречаются синонимы кластерного анализа: автоматическая классификация, таксономический анализ, анализ образов (без обучения).

Варианты кластерного анализа – это множество простых вычислительных процедур, используемых для классификации объектов. Классификация объектов – это группирование их в классы так, чтобы объекты в каждом классе были более похожи друг на друга, чем на объекты из других классов. Более точно, кластерный анализ – это процедура упорядочивания объектов в сравнительно однородные классы на основе попарного сравнения этих объектов по предварительно определенным и измеренным критериям.

Существует множество вариантов кластерного анализа, но наиболее широко используются методы, объединенные общим названием иерархический кластерный анализ {Hierarchical Cluster Analysis). В дальнейшем под кластерным анализом мы будем подразумевать именно эту группу методов.

Кластерный анализ – это комбинаторная процедура, имеющая простой и наглядный результат. Широта возможного применения кластерного анализа очевидна настолько же, насколько очевиден и его смысл. Классифицирование или разделение исходного множества объектов на различающиеся группы – всегда первый шаг в любой умственной деятельности, предваряющий поиск причин обнаруженных различий.

Можно указать ряд задач, при решении которых кластерный анализ является более эффективным, чем другие многомерные методы:

- разбиение совокупности испытуемых на группы по измеренным признакам с целью дальнейшей проверки причин межгрупповых разли-

чий по внешним критериям, например, проверка гипотез о том, проявляются ли типологические различия между испытуемыми по измеренным признакам;

- применение кластерного анализа как значительно более простого и наглядного аналога факторного анализа, когда ставится только задача группировки признаков на основе их корреляции;

- классификация объектов на основе непосредственных оценок различий между ними (например, исследование социальной структуры коллектива по данным социометрии – по выявленным межличностным предпочтениям).

Несмотря на различие целей проведения кластерного анализа, можно выделить общую его последовательность как ряд относительно самостоятельных шагов, играющих существенную роль в прикладном исследовании:

1. Отбор объектов для кластеризации. Объектами могут быть, в зависимости от цели исследования: а) испытуемые; б) объекты, которые оцениваются испытуемыми; в) признаки, измеренные на выборке испытуемых.

2. Определение множества переменных, по которым будут различаться объекты кластеризации. Для испытуемых – это набор измеренных признаков, для оцениваемых объектов – субъекты оценки, для признаков – испытуемые. Если в качестве исходных данных предполагается использовать результаты попарного сравнения объектов, необходимо четко определить критерии этого сравнения испытуемыми (экспертами).

3. Определение меры различия между объектами кластеризации. Это первая проблема, которая является специфичной для методов анализа различий: многомерного шкалирования и кластерного анализа.

4. Выбор и применение метода классификации для создания групп сходных объектов. Это вторая и центральная проблема кластерного анализа. Ее весомость связана с тем, что разные методы кластеризации порождают разные группировки для одних и тех же данных. Хотя анализ и заключается в обнаружении структуры, наделе в процессе кластеризации структура привносится в данные, и эта привнесенная структура может не совпадать с реальной.

5. Проверка достоверности разбиения на классы.

Последний этап не всегда необходим, например, при выявлении социальной структуры группы. Тем не менее следует помнить, что кластерный анализ всегда разобьет совокупность объектов на классы, независимо от того, существуют ли они на самом деле. Поэтому бесполезно доказывать существенность разбиения на классы, например, на основании достоверности различий между классами по признакам, включенным в анализ. Обычно проверяют устойчивость группировки – на повторной идентичной выборке объектов. Значимость разбиения проверяют по внешним критериям – признакам, не вошедшим в анализ.

3.4.2 Методы кластерного анализа

Непосредственными данными для применения любого метода кластеризации является матрица различий между всеми парами объектов. Определение или задание меры различия является первым и необходимым шагом кластерного анализа.

Из всего множества методов кластеризации наиболее распространены так называемые иерархические агломеративные методы. Название указывает на то, что классификация осуществляется путем последовательного объединения (агломерации) объектов в группы, оказывающиеся в результате иерархически организованными. Эти методы – очень простые комбинаторные процедуры, отличающиеся критерием объединения объектов в кластеры.

Критерий объединения многократно применяется ко всей матрице попарных расстояний между объектами. На первых шагах объединяются наиболее близкие объекты, находящиеся на одном уровне сходства. Затем поочередно присоединяются остальные объекты, пока все они не объединятся в один большой кластер. Результат работы метода представляется графически в виде дендрограммы – ветвистого древовидного графика.

Существуют различные методы иерархического кластерного анализа, в частности, в программе SPSS предлагается 7 методов. Каждый метод дает свои результаты кластеризации, но три из них являются наиболее типичными.

Метод одиночной связи – наиболее понятный метод, который часто называют методом ближайшего соседа. Алгоритм начинается с поиска двух наиболее близких объектов, пара которых образует первичный кластер. Каждый последующий объект присоединяется к тому кластеру, к одному из объектов которого он ближе. Этот метод имеет тенденцию образовывать небольшое число крупных кластеров. К его особенностям можно отнести и то, что результаты его применения часто не дают возможности определить, как много кластеров находится в данных.

Метод полной связи часто называют методом «дальнего соседа». Правило объединения этого метода подразумевает, что новый объект присоединяется к тому кластеру, самый далекий элемент которого находится ближе к новому объекту, чем самые далекие элементы других кластеров. Это правило является противоположным предыдущему и более жестким. Поэтому здесь наблюдается тенденция к выделению большего числа компактных кластеров, состоящих из более похожих элементов.

Метод средней связи, или межгрупповой связи, занимает промежуточное положение относительно крайностей методов одиночной и полной связи. На каждом шаге вычисляется среднее арифметическое расстояние между каждым объектом из одного кластера и каждым объектом из другого кластера. Объект присоединяется к данному кластеру, если это среднее расстояние меньше, чем среднее расстояние до любого другого кластера. По своему принципу этот метод должен давать более точные результаты классификации, чем остальные методы. То, что объединение кластеров в методе средней связи происходит при расстоянии большем, чем в методе одиночной связи, но меньшем, чем в методе полной связи, и объясняет промежуточное положение этого метода.

В реальных исследованиях обычно имеются десятки классифицируемых объектов, и применение каждого из указанных методов дает существенно разные результаты для одних и тех же данных. Опыт и литературные данные свидетельствуют, что наиболее близкий к реальной группировке результат позволяет получить метод средней связи. Но это не означает бесполезность применения двух других методов. Метод одиночной связи «сжимает» пространство, об-

тов свою структуру и представляет собой как бы свою точку зрения на реальность. Исследователь, в зависимости от стоящей перед ним задачи, вправе выбрать тот метод, который ему больше подходит.

Численность классов является отдельной проблемой в кластерном анализе. Сложность заключается в том, что не существует формальных критериев, позволяющих определить оптимальное число классов. В конечном итоге это определяется самим исследователем исходя из содержательных соображений.

3.4.3 Кластерный анализ и многомерное шкалирование

Многомерное шкалирование и кластерный анализ – это методы, основанные на дистантной модели: непосредственными исходными данными для них является информация о различии объектов. Поэтому представляет интерес сравнение результатов применения этих методов в отношении одних и тех же данных.

Напомним, что многомерное шкалирование, как и факторный анализ, направлено на выявление небольшого числа шкал. Эти шкалы трактуются как критерии, лежащие в основе различий объектов, и интерпретируются через объекты, поляризованные по этим шкалам. Особое значение при интерпретации шкал, в отличие от факторов, уделяется визуализации координатных представлений объектов в пространстве шкал, что связано с невозможностью поворота шкал относительно объектов (как факторов относительно признаков в факторном анализе). Поэтому особую ценность при шкалировании имеют двух-, максимум – трехшкальные решения. Так же, как и в факторном анализе, при многомерном шкалировании получение решения с малым числом шкал неизбежно влечет потерю исходной информации о различии объектов.

Таким образом, наиболее серьезным ограничением многомерного шкалирования является, по-видимому, предположение о том, что в основе всех различий между объектами лежит небольшое число критериев (шкал). Если есть серьезные сомнения на этот счет, то применение многомерного шкалирования вряд ли оправданно. И тогда целесообразнее применять кластерный анализ.

3.5 Основные статистические понятия организации эксперимента

В настоящий момент не существует полной и точной методики построения пирамиды знаний и выявления семантического пространства предметной области. В рамках данной работы было проведено статистическое исследование, в котором использовались основные понятия статистики, описанные ниже.

Исследование начинается с некоторого предположения, требующего проверки с привлечением фактов. Это предположение – гипотеза – формулируется в отношении связи явлений или свойств некоторой совокупности объектов [14].

Для проверки гипотез на фактах необходимо измерить соответствующие свойства у их носителей. Но невозможно охватить всех носителей изучаемого свойства. Поэтому при проведении исследований ограничиваются лишь

относительно небольшой группой представителей соответствующих совокупностей людей.

Генеральная совокупность – это все множество объектов, в отношении которого формулируется исследовательская гипотеза. Обычно это хотя и не бесконечное по численности, но, как правило, недоступное для сплошного исследования множество потенциальных испытуемых.

Выборка – это ограниченная по численности группа объектов (испытуемых, респондентов), специально отбираемая из генеральной совокупности для изучения ее свойств. Соответственно, изучение на выборке свойств генеральной совокупности называется выборочным исследованием. Практически все исследования, относящиеся к человеческим сообществам, являются выборочными, а их выводы распространяются на генеральные совокупности.

Выборка должна быть такой, чтобы была обоснована генерализация выводов выборочного исследования – обобщение, распространение их на генеральную совокупность. Основные критерии обоснованности выводов исследования – это репрезентативность выборки и статистическая достоверность (эмпирических) результатов.

Репрезентативность выборки – иными словами, ее представительность – это способность выборки представлять изучаемые явления достаточно полно с точки зрения их изменчивости в генеральной совокупности.

Конечно, полное представление об изучаемом явлении, во всем его диапазоне и нюансах изменчивости, может дать только генеральная совокупность. Поэтому репрезентативность всегда ограничена в той мере, в какой ограничена выборка. И именно репрезентативность выборки является одним из критериев при определении границ генерализации выводов исследования. Тем не менее, существуют приемы, позволяющие получить достаточную для исследователя репрезентативность выборки.

Первый и основной прием – это простой случайный (рандомизированный) отбор. Он предполагает обеспечение таких условий, чтоб каждый член генеральной совокупности имел равные с другими шансы попасть в выборку. Случайный отбор обеспечивает возможность попадания в выборку самых разных представителей генеральной совокупности. При этом принимаются специальные меры, исключающие появление какой-либо закономерности при отборе. И это позволяет надеяться на то, что в конечном итоге в выборке изучаемое свойство будет представлено если и не во всем, то в максимально возможном его многообразии.

Второй способ обеспечения репрезентативности – это стратифицированный случайный отбор, или отбор по свойствам генеральной совокупности. Он предполагает предварительное определение тех качеств, которые могут влиять на изменчивость изучаемого свойства (это может быть пол, уровень доходов, образование и т.д.). Затем определяется процентное соотношение численности различающихся по этим качествам групп (страт) в генеральной совокупности и обеспечивается идентичное процентное соотношение соответствующих групп в выборке. Далее в каждую подгруппу выборки испытуемые подбираются по принципу простого случайного отбора.

Статистическая достоверность, или статистическая значимость, результатов исследования определяется при помощи методов статистического вывода. Эти методы предъявляют определенные требования к численности, или объему выборки.

Строгих рекомендаций по предварительному определению требуемого объема выборки не существует. Более того, ответ на вопрос о необходимой и достаточной ее численности исследователь обычно получает слишком поздно – только после анализа данных уже обследованной выборки. Тем не менее, можно сформулировать наиболее важные рекомендации:

- наибольший объем выборки необходим при разработке диагностирующей методики – от 200 до 1000-25000 человек.

- Если необходимо сравнивать две выборки, их общая численность должна быть не менее 50 человек; численность сравниваемых выборок должна быть приблизительно одинаковой.

- Если изучается взаимосвязь между какими-либо свойствами, то объем выборки должен быть не меньше 20-25 человек.

- Чем больше изменчивость изучаемого свойства, тем больше должен быть объем выборки. Поэтому изменчивость можно уменьшить, увеличивая однородность выборки. При этом, естественно, уменьшается возможность генерализации выводов.

Зависимые и независимые выборки. Обычна ситуация, когда интересующее исследователя свойство изучается на двух или более выборках с целью их дальнейшего сравнения. Эти выборки могут находиться в различных соотношениях – в зависимости от процедуры их организации. Независимые выборки характеризуются тем, что вероятность отбора любого испытуемого одной выборки не зависит от отбора любого из испытуемых другой выборки. Напротив, зависимые выборки характеризуются тем, что каждому испытуемому одной выборки поставлен в соответствие по определенному критерию испытуемый из другой выборки.

Следует отметить, что случаи «частично зависимых» выборок недопустимы: это непредсказуемым образом нарушает их репрезентативность.

Измерения и шкалы.

Любое эмпирическое научное исследование начинается с того, что исследователь фиксирует выраженность интересующего его свойства или свойств у объекта или объектов исследования, как правило при помощи чисел. Таким образом, следует различать объекты исследования (испытуемые), их свойства (то, что интересует исследователя) и признаки, отражающие в числовой шкале выраженность свойств.

Измерение в терминах производимых исследователем операций – это приписывание объекту числа по определенному правилу. Это правило устанавливает соответствие между измеряемым свойством объекта и результатом измерения – признаком.

В научном исследовании нам исключительно важно отдавать себе отчет в том, что точность, с которой признак отражает измеряемое свойство, зависит от процедуры (операции) измерения.

В зависимости от того, какая операция лежит в основе измерения признака, выделяют так называемые измерительные шкалы. Эти шкалы устанавливают определенные соотношения между свойствами чисел и измеряемым свойством объектов. Шкалы разделяют на метрические (если есть или может быть установлена единица измерения) и неметрические (если единицы измерения не могут быть установлены).

В основе номинативной шкалы (неметрической), или шкалы наименований (номинальное измерение), лежит процедура, обычно не ассоциируемая с измерением. Пользуясь определенным правилом, объекты группируются по различным классам так, чтобы внутри класса они были идентичны по измеряемому свойству. Каждому классу дается наименование и обозначение, обычно числовое. Затем каждому объекту присваивается соответствующее обозначение. Например, номинативный признак «пол» имеет значения 1 – мужской и 0 – женский.

Ранговая, или порядковая шкала (неметрическая) получается в результате ранжирования. Измерение в этой шкале предполагает приписывание объектам чисел в зависимости от степени выраженности измеряемого свойства.

Интервальная шкала (метрическая). Это такое измерение, при котором числа отражают не только различия между объектами в уровне выраженности свойства (характеристика порядковой шкалы), но и то, насколько больше или меньше выражено свойство. Измерение в этой шкале предполагает возможность применения единицы измерения (метрики). Объекту присваивается число единиц измерения, пропорциональное выраженности измеряемого свойства. Важная особенность интервальной шкалы – произвольность выбора нулевой точки, что означает, что измерение в этой шкале не соответствует абсолютному значению измеряемого свойства. Следовательно, применяя эту шкалу, мы можем судить, насколько больше или меньше выражено это свойство при сравнении объектов, но не можем судить о том, во сколько раз больше или меньше выражено свойство.

Измерения в абсолютной шкале, или шкале отношений (метрической), отличается от интервального в том, что в ней устанавливается нулевая точка, соответствующая полному отсутствию выраженности измеряемого свойства.

Перечисленные шкалы полезно характеризовать еще по признаку их дифференцирующей способности (мощности). В этом отношении шкалы по мере возрастания мощности располагаются следующим образом: номинативная, ранговая, интервальная, абсолютная. Неметрические шкалы заведомо менее мощные – они отражают меньше информации о различии объектов (испытуемых) по измеренному свойству, и, напротив, метрические шкалы более мощные, они лучше дифференцируют испытуемых. Поэтому, если у исследователя есть возможность выбора, следует применять более мощную шкалу. Другое дело, что чаще такого выбора нет, и приходится использовать доступную измерительную шкалу. Более того, часто исследователю даже трудно определить, какую шкалу он применяет.

3 ПРОВЕДЕНИЕ ЭКСПЕРИМЕНТА И ОБРАБОТКА ПОЛУЧЕННЫХ ДАННЫХ

3.1 План эксперимента

Для проверки гипотезы о возможности использования латентных структур для контроля знаний необходимо провести эксперимент среди студентов специальности «Вычислительная техника и программирование» по дисциплине «Программирование на алгоритмических языках» [21]. Подтверждением гипотезы будем считать совпадение результатов исследования и оценок, полученных студентами во время контроля знаний традиционными методами, хотя бы на 80%.

За эталон взяты семантические пространства преподавателей, ведущих указанную дисциплину. Выявление уровней семантических пространств респондентов будем выполнять в соответствии с алгоритмом построения пирамиды знаний.

Общее описание этапов эксперимента [16] приведено в таблице 3.

3.1.1 Участники эксперимента.

Генеральной совокупностью являются все студенты, изучающие дисциплину «Программирование на алгоритмических языках». Исследование выборочное. Обе выборки формируются случайным образом: первая – из числа студентов, вторая – из числа преподавателей-экспертов. Статистическую достоверность обеспечиваем за счет объема выборок: объем первой выборки – 25 студентов, второй – 9 преподавателей и экспертов. Для обеспечения репрезентативности в обеих выборках должны присутствовать представители нескольких страт.

Для студенческой выборки выделены страты Форма обучения, Курс обучения, Преподаватель дисциплины «Программирование на алгоритмических языках». Выбор таких страт обусловлен следующими соображениями. Дисциплину «Программирование на алгоритмических языках» изучают студенты всех форм обучения. На сегодняшний день наиболее значимая группа студентов, работающих в дальнейшем по специальности, – очники, поэтому в выборке должно быть большинство студентов именно такой формы обучения. Студенты заочной и дистанционной форм обучения также должны быть представлены в выборке соответственно с реальным соотношением студентов таких форм в генеральной совокупности.

Страта «Курс» отражает уровень владения языком программирования. Предполагается, что студенты первого курса являются новичками, студенты выпускных курсов должны иметь сформированные практические навыки программирования, студенты третьего курса должны занимать среднее положение между новичками и выпускниками.

Страта «Преподаватель» введена для того, чтобы увидеть в процессе исследования, зависит ли формирующееся семантическое пространство от лектора (преподавателя), читавшего студентам основы программирования, или

Таблица 3. Описание этапов эксперимента

Этап	Участники	Методы	Результаты
Этап 1. Выявление семантических пространств экспертов			
Шаг 1. Формирование перечня понятий предметной области	Эксперты (5 человек)	Составление списка терминов (концептов) предметной области	Перечень понятий
Шаг 2. Выявление взаимосвязей между выявленными понятиями	Эксперты (5 человек)	Метод замкнутых кривых	Семантические сети экспертов
Шаг 3. Выявление метапонятий предметной области	Эксперты (5 человек)	Метод карточек	Разбиение концептов по группам в соответствии их принадлежностью к метапонятию
Этап 2. Выявление семантических пространств испытуемых			
Шаг 1. Формирование перечня понятий предметной области	Студенты (25 человек)	Составление оглавления учебника по программированию	Перечень понятий
Шаг 2. Выявление взаимосвязей между выявленными понятиями метапонятий предметной области	Студенты (25 человек)	Метод замкнутых кривых	Семантические сети испытуемых
Этап 3. Сравнение и оценка выявленных семантических пространств			
Шаг 1. Анализ семантических пространств	Исследователь	Визуальный субъективный анализ	Перечень отличий семантических сетей экспертов и испытуемых. Критерии оценки студентов по уровню знаний
Шаг 2. Верификация семантических сетей экспертов	Эксперты (4 человека)	Многомерное шкалирование, кластерный анализ	Субъективное шкалированное пространство эксперта, группировка понятий предметной области по кластерам (метапонятиям)
Этап 4. Сравнение полученных результатов с итогами традиционных методов контроля знаний студентов	Исследователь		Процент совпадения традиционных оценок и оценок по семантическому пространству

же оно будет примерно одинаковым у большинства студентов независимо от этого.

Подробная характеристика выборок представлена ниже в таблице 4.

Таблица 4. Характеристика выборки студентов (объем – 25).

Страты студенческой выборки	Группы страт	Количество студентов	Процентное соотношение
Форма обучения	Очная	15	60%
	Заочная	7	28%
	Дистанционная	3	12%
Курс	Выпускной	7	28%
	Третий	6	24%
	Первый	12	48%
Преподаватель	Преподаватель 1	7	28%
	Преподаватель 2	8	32%
	Преподаватель 3	10	40%

Для обеспечения репрезентативности и статистической значимости экспертной выборки в качестве экспертов привлекались специалисты, обладающими определенными характеристиками (таблица 5). В их выборке присутствуют страты «опытный лектор», «эксперт практического программирования», «лектор и программист». Так как основной упор в эксперименте делается именно на знания, полученные в процессе обучения, большая часть экспертов должна быть лекторами или преподавателями, ведущими дисциплины программирования. Эксперты практической направленности должны быть представлены в выборке, так как знания, привносимые в семантическую сеть практическим опытом, составляют ее немаловажную часть. Различные эксперты привлекаются к эксперименту на различных стадиях.

Таблица 5. Характеристики участников эксперимента – экспертов

Эксперт	Квалификационная характеристика	Языки или среды программирования, другие специальные знания
1	Опытный лектор	C++, Паскаль
2	Опытный лектор	Паскаль, Delphi
3	Лектор со средним опытом	Паскаль, знание теории построения языков программирования и компиляторов
4	Начинающий лектор, опытный программист	Delphi
5	Опытный программист	Паскаль, Delphi, C++, Visual C.
6	Опытный лектор	Паскаль, Delphi
7	Лектор со средним	Паскаль, Delphi, знание теории информации

	ОПЫТОМ	
8	Лектор со средним опытом	Assembler
9	Опытный программист	Паскаль, Delphi

3.1.2 Ход эксперимента

Весь эксперимент состоит из следующих этапов:

1. выявление семантических пространств экспертов;
2. выявление семантических пространств испытуемых;
3. сравнение и оценка выявленных пространств;
4. сравнение полученных результатов с итогами традиционных методов контроля знаний студентов.

Начальные этапы наиболее трудоемки и похожи между собой. Они включают в себя одинаковые процедуры и отличаются только респондентами: преподаватели в первом случае и студенты во втором.

Первый этап будем проводить в соответствии с теорией инженерии знаний, в частности, с помощью методов извлечения знаний, применяемых при построении баз знаний экспертных систем. В этом процессе участвуют инженер по знаниям (аналитик, когнитолог) и один или несколько экспертов предметной области. В рамках данного эксперимента в качестве когнитолога выступает исследователь, в качестве экспертов – преподаватели на первом этапе и студенты на втором.

Как известно, структурный подход [15, 27] к построению модели предметной области предполагает выделение следующих когнитивных элементов знаний:

- понятия;
- взаимосвязи;
- метапонятия;
- семантические отношения.

Выделяемые понятия предметной области должны образовывать систему, под которой понимается совокупность понятий, обладающая следующими свойствами:

- уникальностью (отсутствием избыточности);
- полнотой (достаточно полным описанием различных процессов, фактов, явлений и т.д. предметной области);
- достоверностью (валидностью – соответствием выделенных единиц смысловой информации их реальным наименованиям);
- непротиворечивостью (отсутствие омонимии).

В рамках данного эксперимента основные концепты семантического пространства «Программирование на алгоритмических языках» можно выявить с помощью простейшего метода формирования перечня понятий. Он заключается в том, что экспертам дается задание составить список понятий, относящихся к исследуемой предметной области. Понятия, выделенные всеми экспертами, включаются в систему понятий (т.е. являются бесспорными концептами семантического пространства), остальные подлежат обсуждению.

Взаимосвязи между выявленными понятиями будут определены методом замкнутых кривых, метапонятия будут определены с помощью метода карточек. Метод замкнутых кривых предполагает, что выделенные концепты располагаются на листе в произвольном порядке, эксперту необходимо соединить их с помощью поименованных связей. В итоге формируется семантическая сеть, отражающая субъективное (когнитивное) пространство предметной области «Программирование» эксперта.

Этап 2, респондентами которого являются студенты, будет проходить следующим образом. Для выявления списка понятий им предложен метод составления оглавления, для выявления метапонятий и взаимосвязей – метод замкнутых кривых.

В результате первого и второго этапа будут получены графические представления семантических пространств экспертов и испытуемых. Для удобства дальнейшего анализа экспертные представления предметной области могут быть сведены в одно «усредненное», включающее в себя только бесспорные концепты и все возможные их взаимосвязи.

Третий этап предполагает визуальное сравнение полученных графических представлений с процентной оценкой их совпадения. Из всех перечисленных выше этапов эксперимента, этот этап, пожалуй, носит самый субъективный характер, т.к. степень совпадения эталонного и проверяемого семантических пространств определяет исследователь, исходя из собственных представлений о схожести схем. Поэтому для получения более объективных результатов необходимо выработать четкие критерии оценки совпадения или различия эталонного и тестируемого графиков.

Для уменьшения субъективности и верификации полученных графических представлений семантической сети на третьем этапе предполагается применение методов многомерного шкалирования и кластерного анализа с обработкой полученных результатов в программе Attestat, предназначенной для всестороннего математико-статистического анализа данных различной природы. Ряд методов, реализованных в программе, не основан на статистических идеях. Еще ряд методов допускают как статистическую, так и физико-математическую интерпретацию. AtteStat имеет модульную структуру и реализует большое число прикладных алгоритмов.

Программное обеспечение AtteStat является модульной программой анализа и обработки данных. Оно предназначено для работы в среде электронных таблиц Microsoft Excel, входящих в комплект офисных приложений Microsoft Office под управлением операционных систем Microsoft Windows. Программное обеспечение AtteStat не оснащено собственным интерфейсом. Вместо этого оно в полной мере использует возможности интерфейса электронных таблиц Microsoft Excel.

На четвертом этапе, как видно из его названия, происходит сравнение полученных результатов с итогами традиционных методов контроля знаний студентов. Так как в эксперименте в качестве испытуемых будут принимать участие студенты, уже имеющие итоговые оценки по дисциплине «Программирование на алгоритмических языках», то организация этой стадии исследования не вызывает никаких затруднений. Окончательные результаты

четвертого этапа позволят подтвердить или опровергнуть гипотезу о возможности использования метода анализа скрытых знаний для контроля знаний студентов.

3.1.3 Ожидаемые результаты эксперимента

В результате проводимого эксперимента исследователь надеется получить следующую информацию:

1. Список понятий предметной области «Программирование на алгоритмических языках», характерный для экспертов и студентов.
2. Графическое представление семантической сети предметной области, где будут указаны взаимосвязи между концептами.
3. Перечень основных отличий семантических сетей экспертов и испытуемых, степень выраженности этих отличий в зависимости от уровня теоретических знаний и практических навыков испытуемых.
4. Субъективное семантическое пространство предметной области экспертов, полученное в результате обработки матриц различия методом многомерного шкалирования. Здесь исследователя прежде всего будет интересовать вопрос о группировке концептов в определенные метапонятия.
5. Субъективное семантическое пространство предметной области экспертов, полученное в результате обработки матриц различия методом кластерного анализа, для проверки правильности выделенных многомерным шкалированием групп концептов (метапонятий).

Эта информация позволит подтвердить или опровергнуть следующие гипотезы, указанные в порядке возрастания их значимости:

1. формируемое семантическое пространство предметной области не зависит от способа подачи материала (т.е. от лектора, читавшего дисциплину). Усвоение основных концептов, установление взаимосвязей между ними, их объединение в метапонятия происходит в первую очередь в процессе совместной с преподавателем и самостоятельной практической деятельности студента.
2. семантические пространства отражают уровень знаний и навыков в программировании. На основании этого всех испытуемых можно разбить на группы новичков, средних и сильных студентов. Причем семантические пространства сильных студентов и экспертов имеют много общего.
3. методы многомерной статистической обработки можно использовать для контроля знаний студентов как более адекватные и менее громоздкие по сравнению с визуальным сравнением семантических пространств.
4. латентные структуры знаний, выявленные с помощью нетрадиционных методов тестирования, могут быть использованы для полноценного и достоверного контроля знаний студентов.

3.2 Ход эксперимента

3.2.1 Формирование семантического пространства эксперта.

Выявление понятий предметной области

Выявление понятий предметной области происходило в несколько шагов. На первом шаге пяти экспертам было предложено перечислить основные понятия, относящиеся, на их взгляд, к указанным темам. Общая тема «Программирование на алгоритмическом языке» была разбита на подтемы в соответствии с уже устоявшимся классическим подходом к обучению программированию [29]. Другими словами, это были названия глав традиционного учебника по программированию. Темами, предложенными экспертам для формирования перечня понятий, были:

1. Алгоритмы;
2. Типы данных;
3. Структура программы;
4. Операторы языка программирования.

Каждый эксперт указал в общей сложности от 20 до 40 понятий. Всего было указано 70 различных понятий. Их перечень по темам представлен в таблице 6.

Таблица 6. Перечень концептов предметной области «Программирование на алгоритмических языках»

Понятие	Абсолютная частота	Невхождение в результирующий список
Алгоритмы		
1. Алгоритм	5	
2. Задача	4	
3. Блок-схема	5	
4. программа	5	
5. структурные элементы блок-схемы	1	Не входит
6. Постановка задачи	2	
7. Простые алгоритмы	1	синонимы
8. Линейные алгоритмы	3	
9. Разветвляющиеся алгоритмы	3	
10. циклические алгоритмы	3	
11. стандартные алгоритмы	1	Не входит
12. алгоритм сортировки	1	Не входит
Типы данных		
13. идентификатор	2	
14. переменная	5	
15. имя переменной	5	
16. тип переменной	5	
17. размер переменной	1	синонимы
18. диапазон типа	3	
19. постоянная	1	синонимы
20. константа	5	
21. имя константы	5	

22.	тип константы	3	
23.	целый тип	3	
24.	вещественный тип	3	
25.	логический тип	3	
26.	символьный тип	2	
27.	сложный тип	1	синонимы
28.	составной тип	2	
29.	перечисляемый тип	1	
30.	строки	3	
31.	массив	4	
32.	запись	2	синонимы
33.	структура	1	
34.	множества	1	Не входит
35.	файлы	1	Не входит
36.	динамические переменные	1	Не входит
37.	метка	1	Упоминаются в другой теме
38.	имя метки	1	
39.	комментарии	1	
Структура программы			
40.	программа	4	
41.	стандартный модуль	1	Не входит
42.	подпрограмма	5	Синонимы
43.	модуль	2	
44.	программный блок	1	
45.	процедура	4	
46.	функция	4	
47.	параметр	2	
48.	тип функции	3	
49.	структура программы	5	
50.	блочная структура	2	
51.	разделы описаний	1	Не входит
52.	блок описания модулей	1	Не входит
53.	блок описания меток	2	
54.	блок описания констант	3	
55.	блок описания переменных	5	
56.	блок описания функций и процедур	2	Не входит
57.	исполняемая часть	2	
58.	тело программы	4	
59.	оператор	3	синонимы
60.	ключевое слово	1	
61.	команда	1	
62.	операторные скобки	2	синонимы
63.	логические блоки	1	

64.	комментарии	1	
65.	метка	1	
Операторы языка программирования			
66.	лексика языка	1	Не входит
67.	выражения	2	Не входит
68.	операция	2	Не входит
69.	стандартная функция	2	Не входит
70.	оператор	5	
71.	операнд	3	
72.	операторы ввода-вывода	5	
73.	оператор присваивания	4	
74.	оператор выбора	4	
75.	условный оператор	5	
76.	операторы перехода	3	
77.	операторы цикла	5	

Сформированный перечень понятий можно считать полным и достоверным (валидным) благодаря разностороннему составу привлеченных экспертов.

Уникальность и непротиворечивость система понятий приобрела только после определенной обработки исходного полного списка всех терминов. Как показано в таблице 1, некоторые базовые термины появлялись у всех респондентов (число появлений равно 5), часть терминов появлялась у большинства (число появлений равно 3 и 4), часть появилась только единожды или дважды. Термины с максимально возможным числом вхождений без каких-либо изменений были перенесены в результирующий список, так как, появившись у всех экспертов, они считаются бесспорными понятиями предметной области. Такой же вывод можно сделать и о понятиях, имеющих показатель 4.

Термины, появившиеся в ответах экспертов 1 – 3 раза, были обсуждены с экспертами повторно. В результате этого обсуждения выяснилось, что некоторые различные термины имеют один и тот же смысл, то есть являются синонимами. Их различное написание вызвано такими причинами, как предпочитаемый экспертом язык программирования (записи в Паскале аналогичны структурам в С) и вид профессиональной деятельности. У лекторов-теоретиков понятия обозначены так же, как в учебниках, программисты-практики называли понятия с их функциональной точки зрения. Кроме того, некоторые эксперты сознательно указали синонимы для некоторых терминов, например перечисляемый и составной тип переменных.

Таковыми синонимами являются следующие понятия: простые алгоритмы и линейные алгоритмы; константа и постоянная; размер переменной и диапазон типа; сложный тип, составной тип и перечисляемый тип; структура и запись; программный модуль, модуль, подпрограмма; оператор, ключевое слово, команда; операторные скобки и логический блок.

Синонимия была удалена путем объединения терминов в одно понятие в виде его возможных значений, их показатели появления суммировались. В

результатирующий список было перенесено только одно общее понятие. Этот шаг обеспечил уникальность формируемого перечня понятий.

Первоначальный список понятий содержал и омонимию, то есть был противоречив. Это выражалось в том, что некоторые понятия были упомянуты разными экспертами в разных темах. Так, понятие «программа» появлялся как в разделе «Алгоритмы», так и в разделе «Структура программы»; термины «метка» и «комментарий» упоминаются в «Типах данных» и в «Структуре программы»; термин «операторы» в фигурирует в «Структуре программы», «Операторах языка программирования», «Типах данных».

Непротиворечивым список стал после дополнительного обсуждения с экспертами. Выяснилось, что термины «оператор» и «программа» действительно можно отнести к двум темам, а термины «метка» и «комментарии» правильнее оставить только как понятия темы «Структура программы».

Оставшиеся термины, появившиеся в списках только одного эксперта, были отнесены к незначимым с точки зрения проверки знаний студентов.

В целом, результаты обработки данных таблицы 6 с учетом свойств совокупности понятий представлены в таблице 7.

Таблица 7. Характеристика появления терминов в списке

Количество появлений	Абсолютная частота	Относительная частота	Накопленная частота
5	16	0,24	0,24
4	11	0,16	0,4
3	14	0,21	0,6
2	12	0,18	0,78
1	15	0,22	1
Сумма	68	1	

Результатирующий список понятий предметной области «Программирование на алгоритмических языках» выглядит следующим образом.

Тема «Алгоритмы»

1. Алгоритм
2. Задача
3. Блок-схема
4. программа
5. Постановка задачи
6. Простые (линейные) алгоритмы
7. Разветвляющиеся алгоритмы
8. циклические алгоритмы

Тема «Типы данных»

9. идентификатор
10. переменная
11. имя переменной

12. тип переменной
13. диапазон типа
14. константа (постоянная)
15. имя константы
16. тип константы
17. целый тип
18. вещественный тип
19. логический тип
20. символьный тип
21. составной (сложный, перечисляемый) тип
22. строки
23. массив
24. запись (структура)

Тема «Структура программы»

25. программа
26. модуль (подпрограмма, программный блок)
27. процедура
28. функция
29. параметр
30. тип функции
31. структура программы
32. блочная структура
33. блок описания меток
34. блок описания констант
35. блок описания переменных
36. исполняемая часть
37. тело программы
38. оператор (ключевое слово, команда)
39. операторные скобки (логический блок)
40. комментарии
41. метка

Тема «Операторы языка программирования»

42. оператор
43. операнд
44. операторы ввода-вывода
45. оператор присваивания
46. оператор выбора
47. условный оператор
48. операторы перехода
49. операторы цикла

Выявление метапонятий

Для выявления метапонятий предметной области «Программирование на алгоритмических языках» использовался метод карточек. Экспертам были предложены карточки с написанными на них понятиями из списка, сформированного на предыдущем шаге. Карточки надо было разбить на группы, руководствуясь их собственными знаниями и практическим опытом.

Предполагалось, что после разбиения эксперт сможет назвать один или несколько признаков отбора, благодаря чему можно будет поименовать метапонятие, объединяющее все отнесенные к нему концепты.

В результате этого шага обнаружилось, что все эксперты разбили понятия по тем же группам, что и темы первого шага. Это вполне закономерно, но несколько нарушает чистоту эксперимента. Для избежания подобного положения разумней было бы на первом шаге предложить экспертам сформировать список всех возможных понятий предметной области «Программирование» без разбиения ее на темы. Или для второго шага необходимо было пригласить другую группу экспертов, которые разнесли бы карточки по группам непредвзято.

В любом случае, по мнению экспертов – опытных преподавателей, выделенных даже таким образом метапонятий и составляющих их концептов вполне достаточно, чтобы сформировать семантическую сеть предметной области «Программирование» обучаемых.

Определение отношений

Последний шаг формирования семантической сети предметной области заключается в именовании связей между выделенными концептами и метапонятиями. Для выявления семантических отношений за основу был взят метод, имеющий теоретической основой методологию когнитивной графики.

Экспертам был предложен список понятий, которые надо было соединить между собой в виде направленного графа, дуги которого могут быть ориентированными или неориентированными, но обязательно должны иметь название. Для облегчения задачи экспертам также был предложен примерный перечень имен связей: это (ISA), является, состоит из, включает в себя, должен содержать, может содержать, следует за, предшествует и т.д.

Этот шаг осуществляли четыре эксперта, не принимавшие участия в первых двух шагах. Их привлечение кажется целесообразным для верификации полученного перечня понятий и выделенных метапонятий. Если для построения семантических сетей им будет достаточно предложенного списка концептов и если на сетях будут прослеживаться обнаруженные метапонятия, значит, статическая часть базы знаний предметной области «Программирование» сформирована с учетом всех необходимых свойств: полноты, валидности, избыточности.

В результате этого шага было получено четыре семантические сети, две наиболее наглядные представлены на рисунках 8,9 в приложении А.

Как видно, сеть первого эксперта содержит три из четырех выделенных метапонятий: алгоритм, структура программы, типы данных. Сеть второго эксперта содержит концепты, сгруппированные таким образом, что можно увидеть все четыре метапонятия.

Кроме предложенных экспертам имен связей, также были обозначены: атрибутивная связь «имеет свойство», связь типа «например» в дополнение к связи «это», логические связи «И» и «НЕ».

Интересно, что по способу отображения семантических сетей можно сделать предположения о характере деятельности экспертов. Очевидно, что первый эксперт является опытным лектором именно по дисциплине

«Программирование». Понятие «исполнительная часть» повторяет последовательность выполнения лабораторных работ по этому курсу: концепты «Массив», «Подпрограммы», «Модули» расположены в таком порядке именно по этой причине. Кроме того, в этой сети появляется понятие «лексика языка», исключенное из перечня понятий на первом шаге. Здесь оно не могло не появиться, так как знакомство студентов с любым языком программирования лектор начинает с описания лексики языка.

Сеть второго эксперта говорит о том, что ему знаком способ представления знаний с помощью семантических сетей (использование не только функциональных связей, но и атрибутивных и логических), но вряд ли он читает лекции по дисциплинам программирования.

Таким образом, первый этап эксперимента завершился следующими результатами:

1. для проверки знаний студентов по дисциплине «Программирование на алгоритмических языках» сформирован перечень из 49 понятий, разбитых на четыре метапонятия;
2. полученная статическая часть базы знаний предметной области обладает всеми необходимыми свойствами;
3. полученные семантические сети достаточно наглядны и содержательны, но их обработка требует усилий;
4. все четыре сети сильно отличаются друг от друга, привести их к какому-либо «среднему» графическому виду почти невозможно;
5. необходимо использовать другой метод представления семантического пространства предметной области, подходящий для обработки и практического применения.

3.2.2 Формирование семантического пространства испытуемых.

Второй этап эксперимента – формирование семантического пространства испытуемых – первоначально предполагалось провести, так же как последний шаг этапа 1, т.е. понятия сформированного списка студенты соединят поименованными связями. Но первый этап эксперимента показал, что процесс формирования семантического пространства очень трудоемок. Еще более трудоемким процессом оказался процесс сравнения семантических пространств, на структуру которых не накладывается никаких ограничений. Никаких математических или статистических методов для сравнения ориентированных графов с поименованными дугами не существует.

Кроме того, к эксперименту привлекались студенты разных групп, преподаватели «Программирования» в них были различными. Чтобы проверить подгипотезу о формировании примерно одинакового семантического пространства, независимо от лектора или преподавателя, мы посчитали необходимым провести со студентами и первый шаг выяснения структуры предметной области. Таким образом, второй этап, в котором участвовало 25 студентов, выглядел следующим образом.

Первоначально студентам было предложено составить оглавление учебника по программированию с указанием основных разделов. Трудность для студентов заключалась в том, что они изучали теорию программирования в

среднем около года назад. Им необходимо было не только структурировать свои знания, но хотя бы вспомнить последовательность изучения материала и терминологию. Даже те студенты, которые очень хорошо программируют, затруднились выполнить это задание.

Единственная группа (шесть студентов), которая справилась с составлением содержания, была группа, только закончившая изучение основ программирования. Эксперимент в этой группе проводился сразу после экзамена по этой дисциплине, в памяти студентов еще был свеж порядок выполнения лабораторных работ на Паскале и написанные им оглавления почти полностью повторяли последовательность работ, данную в методических указаниях по изучению предмета.

Студенты этой группы, обладающие хорошими теоретическими знаниями и практическим опытом разработки программ, упомянули все четыре метапонятия, выявленные в семантических пространствах экспертов. В их работах присутствуют интенционалы, содержание хорошо структурировано.

После того, как не удалось массово выяснить список терминов, оставшихся в памяти студентов, с помощью составления оглавления, студентам было предложено более простое задание. Оно повторяло задание для экспертов. Студентам необходимо было указать термины по определенным темам программирования. Темы остались те же: алгоритмы, структура программы, типы данных и операторы языка программирования. С этим заданием в той или иной степени справились все студенты.

В результате выяснилось следующее:

1. студенты, обладающие хорошими теоретическими знаниями и практическим опытом разработки программ, упомянули почти все понятия, вошедшие в результирующий список понятий экспертов. В их работах присутствуют интенционалы, термины перечислены в порядке их изучения или использования в программе. Например, в теме «Операторы языков программирования» они указывали не сами операторы, а их функциональную принадлежность: операторы ввода - вывода; оператор присваивания; операторы стандартных математических функций; операторы выбора и условия; операторы цикла.

В теме «Типы данных» сильные студенты упомянули не только простейшие, наиболее часто используемые типы данных (целый, вещественный, массивы), но и те, которые необходимы при написании более сложных и крупных программ. Так, в их работах присутствовали такие типы, как файл, множество, запись.

2. студенты среднего уровня предпочитают указывать экстенционалы понятий, т.е. приводить примеры, а не определения [20]. Например, в работах четырех студентов структура программы была показана так: PROGRAM VAR BEGIN END.

Эта запись характерна для правил грамматики произвольного языка в форме Бэкуса-Наура [24, 25]. Ее показывают и объясняют студентам при изучении теории построения языков программирования в рамках дисциплины «Системное программное обеспечение».

В теме «Операторы языков программирования» такие студенты указывали сами операторы: Begin, End, For, While и т.д.

В теме «Типы данных» перечислены типы целый вещественный, логический, символьный, строковый, но не указаны составные типы, кроме массива;

3. слабые студенты в большей части с заданием не справились.

После обработки ответов студентов получился меньший список терминов, чем экспертный, но, по всей видимости, более актуальный для студентов.

Тема «Алгоритмы»

1. Алгоритм
2. Задача
3. Блок-схема
4. программа

Тема «Типы данных»

5. переменная
6. имя переменной
7. тип переменной
8. константа
9. имя константы
10. целый тип
11. вещественный тип
12. логический тип
13. символьный тип
14. составной (сложный, перечисляемый) тип
15. массив

Тема «Структура программы»

16. модуль (подпрограмма, программный блок)
17. процедура
18. функция
19. структура программы
20. блок описания меток
21. блок описания констант
22. блок описания переменных
23. тело программы
24. оператор (ключевое слово, команда)
25. комментарии

Тема «Операторы языка программирования»

26. оператор
27. операторы ввода-вывода
28. оператор присваивания
29. оператор выбора
30. условный оператор
31. операторы цикла

Выявление связей между терминами предметной области.

После выяснения студенческого списка понятий предметной области стало понятно, что проведение эксперимента в запланированном виде является очень громоздким и неинформативным. Поэтому нами было решено в некоторых

моментах эксперимент изменить. В результате были сделаны следующие упрощения:

1. перед экспериментом объяснить, что такое семантическая сеть. Семантическая сеть – это способ представления знаний, основанный на современных представлениях об организации долгосрочной памяти человека. Она представляет собой ориентированный граф, вершины которого – понятия предметной области (концепты, термины), а дуги – это отношения между ними;

2. размерность пространства ограничить изначально 14 терминами. Это термины, встречавшиеся в большей части работ студентов по выявлению понятийного списка и те, которые упоминались экспертами. Термины в списке специально были указаны вразброс, чтобы для студентов не было подсказки в их расположении [14]. Также среди терминов специально были оставлены синонимы.

1. комментарии
2. процедура
3. функция
4. операторные скобки
5. массив
6. тип переменной
7. ключевое слово
8. блок описания переменной
9. структура программы
10. тело программы
11. переменная
12. константа
13. операторы
14. подпрограмма

3. Чтобы этот шаг эксперимента не терял своей познавательной ценности, были оставлены только базовые термины, с одной стороны. С другой стороны, студенты не были ограничены в количестве новых терминов, которые они сочтут необходимым внести в семантическую сеть, чтобы сделать ее более полной. Например, вместо перечисления всех возможных типов (вещественный, целый и т.д.) остался термин «тип переменной». Вместо указания всех возможных функциональных групп операторов (операторы ввода-вывода, оператор присваивания, оператор выбора, условный оператор, операторы цикла) остался только термин «оператор».

4. указать список связей, которые можно использовать при построении семантической сети: это, состоит из, включает, является, должен содержать, может содержать, следует за, предшествует;

5. в процессе объяснения задания перед экспериментом приводить студентам пример семантической сети любой другой предметной области.

На выполнения задания студентам давалось 20 минут. В результате проведения этой части эксперимента было получено 25 семантических сетей предметной области «Программирование» от студентов различных групп различных форм обучения. Они представлены на рисунках в приложении А.

3.2.3 Сравнение семантических пространств эксперта и испытуемых.

После первичного анализа из 25 работ, полученных в результате этапа 2, были исключены 9, авторы которых не справились с заданием. Оставшиеся 16 работ были подробно изучены и проанализированы. В результате даже беглого анализа стала отчетливо видна разница между работами сильных и слабых студентов. Сети сильных студентов насыщены связями, их концепты могут быть объединены несколькими отношениями с другими концептами. Сети более слабых студентов в большинстве бинарны, т.е. каждая связь соединяет только два концепта.

Более подробный анализ представленных семантических сетей приведен в таблице 8. В строках таблицы указаны критерии, по которым можно оценивать уровень программирования автора сети, по столбцам указаны значения критериев для различных групп испытуемых.

В качестве критериев взяты наиболее существенные отличия экспертных и студенческих семантических сетей. Здесь не проводится анализ причин их появления, мы просто констатируем наличие таких отличий. Отличием будем считать отличную от экспертной связь между конкретными отношениями предметной области, причем будем учитывать и степень отличия. Это дает основание разделять испытуемых на группы, обозначенные как сильные студенты, средние студенты, слабые студенты.

Комментарии к каждому из предложенных критериев даны ниже.

1. Алгоритм, блок-схема, задача, программа, структура программы. Как описано в методологии создания программных продуктов, этапу программирования должен предшествовать этап постановки задачи, ее алгоритмизация и построение блок-схемы согласно выбранному алгоритму. В семантических сетях экспертов этот последовательный процесс явно обозначен (рисунок 10). В сетях многих студентов этапы концептуализации задачи и ее последующая детализация либо не разграничиваются, либо меняются местами. В большинстве случаев между этими терминами студенты вообще не видят разницы (рисунок 11).

2. Переменные и константы. В семантических сетях экспертов и переменные, и константы характеризуются тем, что имеют имя и тип. Большинство студентов не выделяют этих признаков. В отдельных работах понятие «константа» приравнивают к «типу переменной» (рисунок 12).

3. Переменная – блок описания переменной, константы – блок описания констант. Слабые студенты не связывают термин «переменная» и «блок описания переменной», «константа» и «блок описания констант». (рисунки 13, 14). Средние студенты указывают эту связь. (рисунок 15)

4. Переменная, массив. К большому удивлению, студенты часто не соотносят термин «массив» с понятием «переменная» (рисунки 16, 17). Но в работе сильных студентов такая связь присутствует (рисунки 18, 12).

5. Массив, составной тип. Сильные студенты указали принадлежность массива к перечисляемому типу переменных (рисунок 18). В работах слабых студентов часто этот концепт вообще не обозначен.

Таблица 8. Критерии оценки знаний студентов

Критерий	Эксперты	Сильные студенты	Средние студенты	Слабые
Алгоритм, блок-схема, задача, программа, структура программы.	Жестко разграничивают, указывают взаимосвязь и последовательность	Разграничивают, указывают взаимосвязь	Почти не разграничивают, воспринимают как синонимы	Не разграничивают, часто не указывают
Переменные и константы.	Жестко разграничивают, указывают такие свойства, как имя и тип	Жестко разграничивают, указывают такие свойства, как имя и тип	Разграничивают, указывают такое свойство, тип	Не разграничивают
Переменная – блок описания переменной, константы – блок описания констант.	Связь типа «описывается» или «принадлежит» всегда присутствует	Связь типа «описывается» или «принадлежит» часто присутствует	Не связывают	Не связывают
Переменная, массив.	Связь типа «это», «состоит из» или «является частью» всегда присутствует	Связь типа «это», «состоит из» или «является частью» часто присутствует	Не связывают, относят массив к другим понятиям	Не связывают
Массив, составной тип.	Связь типа «это» всегда присутствует	Связь типа «это» часто присутствует	Не связывают, концепт «составной тип» относят к другим понятиям	Не связывают, концепт «составной тип» не указывают
Тело программы, блоки описания переменных, констант, меток, исполняемая часть.	Жестко разграничивают, указывают взаимосвязь и	Жестко разграничивают, указывают взаимосвязь и	Частично разграничивают, иногда указывают взаимосвязь	Не разграничивают, не указывают последовательность

	последовательность	последовательность	
Модуль, подпрограмма, функции, процедуры.	Связь типа «это» или «является частью» всегда присутствует	Связь типа «это» или «является частью» всегда присутствует	Связь типа «это» или «является частью» иногда присутствует
Оператор – функции, процедура, программа.	Связь типа «состоит из» всегда присутствует	Связь типа «состоит из» всегда присутствует	Присутствует связь «состоит из», «образуют» между программой и оператором
Функция, процедура – список параметров.	Нет данных	Связь типа «работает с» иногда присутствует	Не связывают
Ключевые слова, операторы.	Связь типа «это» всегда присутствует	Связь типа «это» всегда присутствует	Не связывают
Местонахождение термина «комментарий».	Всегда в «теле программы» или «программе»	Всегда в «теле программы» или «программе»	Относят к другим понятиям
Перечисление возможных типов переменных.	часто присутствует	часто присутствует	Не указывают
Перечисление возможных функциональных типов операторов.	часто присутствует	часто присутствует	Не указывают
Отдельные ветки (гнезда) по метаяпонятиям.	часто присутствует	Нет данных	Не указывают

6. Тело программы, блоки описания переменных, констант, меток, исполняемая часть. Эксперты выделяют в теле программы разделы описаний (блоки описания меток, констант, переменных) и исполняемую часть, причем именно в такой последовательности. В работах сильных студентов последовательность блоков обозначена связями «предшествует за», «следует за», в то время как средние и слабые студенты либо не указывают этих блоков вообще (рисунок 16), либо указывают наличие таких блоков, но не их последовательность (рисунок 19). Исполняемая часть как обособленный участок программы редко появляется в работах студентов.

7. Модуль, подпрограмма, функции, процедуры. Как известно, любая функция или процедура может рассматриваться как модуль или подпрограмма в основной программе. Это отражено в работах экспертов (рисунок 8). Большинство студентов не соотносят эти термины между собой (рисунок 16).

8. Оператор, функции, процедура, программа. Во многих студенческих работах не показана очевидная связь «состоит из», «содержит» для операторов, функций и процедур. Чаще всего указывают, что программа состоит из операторов, но вложенность термина «оператор» в термины «процедура» или «функция» указывали только сильные студенты.

9. Функция, процедура – список параметров. В двух работах появилось более детальное описание функций (рисунки 18, 20), что говорит об умении работать с подпрограммами.

10. Ключевые слова, операторы. Только в работах экспертов и сильных студентов, имеющих практический опыт программирования, появилась связь «это» между понятиями «ключевое слово» и «оператор» (рисунок 18). Слабые студенты ассоциируют термин «ключевое слово» с другими терминами (рисунки 14, 21), или не указывают его в своих семантических сетях.

11. Местонахождение термина «комментарий». Обычно комментариями при написании программ пользуются уже достаточно опытные программисты. Поэтому логично было бы ожидать появления термина «комментарий» в рамках термина «программа» или «структура программы». В работах экспертов и сильных студентов это действительно так (рисунок 22). В работах других студентов, чьи практические навыки еще не сформированы, это понятие относится к чему-либо другому (рисунки 15, 16).

12. Перечисление возможных типов переменных. Так как по условиям эксперимента студенты могли расширять список терминов, один из них дал детализацию понятию «тип переменной» (рисунок 17).

13. Перечисление возможных функциональных типов операторов. Так как по условиям эксперимента студенты могли расширять список терминов, один из них дал детализацию понятию «операторы» (рисунок 17).

14. Отдельные ветки (гнезда) по метапонятиям. Ярko выраженные гнезда метапонятий (алгоритмы, типы данных, структура программы, операторы языка программирования) (рисунок 9) прослеживаются в работах экспертов. Но даже в большинстве работ сильных студентов метапонятия не выражены.

Критерии 9, 12 – 14 являются желательными, но не обязательными, так как при объяснении задачи внимание студентов на этом не заострялось. Их появление в работе студента говорит о достаточно сформированном

семантическом пространстве предметной области и понимании взаимосвязей между ее понятиями.

3.2.4 Сравнение результатов эксперимента с традиционными оценками

Отнесение конкретного студента к определенной группе является подтверждением или опровержением основной гипотезы работы о возможности исследования семантического пространства испытуемого с целью контроля его знаний. Так как проводимый эксперимент не был анонимным, нам известны исходные данные каждого респондента, в том числе и его традиционные оценки по дисциплине «Программирование на алгоритмических языках». Каждый студент также оценивал и свои практические навыки. Сравнение полученных оценок с результатами исследования семантических пространств будет основанием для вывода о приемлемости или неприемлемости подобного рода проверок.

Будем относить студента к сильной группе, если из 10 обязательных критериев в его работе присутствуют все 10 и есть один необязательный. К средней группе отнесем тех, чьи семантические сети совпадают с экспертными по 6 из обязательных критериев. Остальных будем относить к слабым.

Традиционные и полученные в результате эксперимента оценки будем соотносить друг с другом следующим образом (таблица 9):

Таблица 9. Соотнесение традиционных оценок и экспериментальные оценки

Традиционная оценка	Возможные оценки по семантическим сетям
5	Сильный
	Средний, ближе к сильному
4	Средний
	Средний, ближе к слабому
3	Слабый

Все исходные данные о респондентах и результаты анализа их семантических пространств представлены в таблице 10.

Таблица 10. Результаты анализа семантических пространств испытуемых.

№	Форма обучения	Курс	Традиционная оценка	Практические навыки	Группа по семантическому пространству
1	Очная	3	4	3	Слабый
2	Очная	4	5	5	Средний, ближе к сильному
3	Очная	4	5	5	Сильный

4	Очная	4	5	4	Средний, ближе к сильному
5	Очная	1	4	4	Средний
6	Очная	1	5	4+	Средний, ближе к сильному
7	Очная	1	4	3	Слабый
8	Очная	1	5	5	Средний, ближе к сильному
9	Очная	1	4	4	Средний, ближе к слабому
1	Заочная	1	5	3	Средний
1	Заочная	1	4	3	Средний, ближе к слабому
1	Заочная	1	4	3	Слабый
1	Заочная	1	3	3	Слабый
1	Заочная	1	3	2	Слабый
1	Заочная	1	3	3	Слабый
1	Заочная	1	4	2	Слабый
1	Очная	3	3	2	Сеть не сформирована
1	Очная	3	3	2	Сеть не сформирована
1	Очная	3	3	2	Сеть не сформирована
2	Очная	3	3	2	Сеть не сформирована
2	Очная	3	3	2	Сеть не сформирована
2	Очная	3	3	2	Сеть не сформирована
2	Дистанционная	3	4	2	Сеть не сформирована
2	Дистанционная	3	3	3	Сеть не сформирована
2	Дистанционная	3	3	2	Сеть не сформирована

Согласно данным таблиц 9 и 10, оценки совпали для 10 испытуемых из 25. Это составляет меньше 50%. Но если учесть оценки практических знаний студентов, то процент совпадений увеличивается до 90% (21 из 25).

Как видно из таблицы 5, студенты, имеющие 4 по теоретическим знаниям и 3 по практическим навыкам, показывают семантические пространства, характерные для средних или слабых студентов; студенты, у которых 5 по теории и 3 по практике, показывают семантические пространства, характерные для слабых студентов; те студенты, которые получили 3 по теории, но не имеют никаких практических навыков программирования, не смогли сформировать семантическую сеть.

На основании полученных данных можно сделать вывод о том, что на формирование семантического пространства предметной области «Программирование на алгоритмических языках» очень сильно влияет наличие практического опыта.

3.2.5 Обработка результатов эксперимента

Данное исследование затронуло небольшое количество студентов, что, несомненно, влияет на возможность генерализации полученных выводов о возможности применения латентных структур знаний для контроля в учебном процессе. Хотя даже в этой небольшой выборке представлены несколько страт: форма обучения; курс; преподаватель, преподававший дисциплину

«Программирование на алгоритмических языках». В этих стратах студенты выбирались случайным образом, что соответствует статистической методологии эксперимента.

Кроме того, как было сказано в выводах по результатам первого этапа эксперимента, сравнение семантических пространств – процесс очень трудоемкий и в большей степени носит субъективный характер, т.к. желание исследователя увидеть подтверждение собственным гипотезам может привести к искажению восприятия семантических пространств экспертов и испытуемых.

Эти недостатки эксперимента были неизбежны, их появление предполагалось еще на стадии его планирования. Чтобы уменьшить субъективность оценки семантических пространств, можно использовать специальные статистические методы полученных результатов [], такие как многомерное шкалирование и кластерный анализ. Их описание и методы применения приведены в предыдущей главе данной работы.

Также математические методы в рамках проводимого исследования позволяют определить достоверность полученных результатов.

Проверка полученных результатов с помощью многомерных методов.

Исследованию с помощью многомерных методов подверглись семантические пространства только экспертов.

Как показано в плане эксперимента, статистические расчеты выполнялись с помощью специального модуля Attestat, работающего на базе стандартного приложения Excel. К сожалению, эта надстройка недостаточно мощная. В ней предусмотрены только базовые операции многомерного анализа данных, количество модификаций тестов ограничено, а самым большим недостатком является представление результатов в матричном, а не в графическом виде. Этих недостатков лишены более мощные самостоятельные среды статистической обработки, например, SPSS или STATISTICA. Но они являются лицензионным программным обеспечением, причем платным.

Исходными данными для многомерных методов являются матрицы оценок близости определенных концептов выбранной предметной области. Эти матрицы формировали эксперты на основе списка концептов, который предлагался студентам для построения семантической сети.

Исходный список концептов содержал 14 терминов. Размерность пространства была сознательно уменьшена с исходных 68 терминов, т.к. не представляется возможным оценить 2278 пар объектов. Для 14 объектов эксперту необходимо оценить 91 пару.

Так как исходная матрица экспертных оценок должна быть симметричной относительно главной диагонали, предполагаем, что отношения между терминами симметричны. Такое предположение позволит заполнять только нижний (верхний) угол матрицы под (над) главной диагональю. Для этого вводим следующую шкалу оценок близости:

- | | |
|---|----------------------------|
| 1 | Объекты очень похожи |
| 2 | объекты похожи |
| 3 | Объекты имеют много общего |
| 4 | Объекты имеют что-то общее |

По сути, эксперт отвечает на вопросы плана: «Насколько похожи между собой процедура и функция?». При этом термин «процедура» находится в строке, а термин «функция» в столбце матрицы. На пересечении эксперт проставляет свою оценку отличия двух объектов.

Заполненные матрицы двух экспертов представлены на рисунках 24 и 25.

Результаты, полученные с помощью обработки матриц отличия методом многомерного шкалирования, показаны на рисунках 26, 27.

В субъективном пространстве первого эксперта явно прослеживаются группы объектов, которые соответствуют выделенным на первом этапе эксперимента метапонятиям предметной области: метапонятие «Структура программы» включает в себя термины тело программы, процедура, структура программы, функция, подпрограмма. Эта же группа объектов присутствует в субъективном пространстве второго эксперта. Также у обоих экспертов достаточно ярко выражено метапонятие «Тип данных». Сюда входят термины блок описания переменных, переменная, тип переменных.

Хорошо видно, что у двух экспертов противоположное расположение объектов по горизонтальной оси. Субъективное пространство первого эксперта инвертировано по горизонтали по сравнению с субъективным пространством второго эксперта. В то время как по вертикальной оси их пространства совпадают.

Такое отличие всего двух субъективных пространств уже показывает, насколько различными могут быть пространства различных экспертов и испытуемых. Поэтому целесообразно применить более простой способ многомерного шкалирования, который покажет распределение объектов по метапонятиям более явно.

С такой задачей попробуем справиться с помощью кластерного анализа. На рисунках 28, 29 показан результат таксонометрии по тем же исходным данным, как и в многомерном шкалировании. Видно, что те же понятия, которые группировались в многомерном шкалировании, здесь обособливаются в отдельные кластеры. Так как нас интересует только распределение терминов по кластерам (метапонятиям), становится очевидным, что метод кластерного анализа так же информативен, как и многомерное шкалирование, но представление его результатов более наглядно.

3.3 Результаты эксперимента

По окончании эксперимента можно сказать, что в результате была получена вся необходимая информация для разработки программного средства, осуществляющего контроль знаний студентов на основе открытых вопросов:

1. список понятий предметной области «Программирование на алгоритмических языках», характерный для экспертов и студентов.
2. взаимосвязи между концептами.
3. субъективное семантическое пространство предметной области экспертов, полученное в результате статистической обработки данных.

	комментарии	процедура	функция	операторные скобки	массив	тип перемен- ной	ключевое слово	блок описания переменной	структура программы	тело программы	переменная	константа	операторы	подпрограмма
комментарии	0													
процедура	4	0												
функция	4	2	0											
операторные скобки	1	2	3	0										
массив	1	5	5	5	0									
тип переменной	1	5	2	5	2	0								
ключевое слово	1	3	3	4	5	5	0							
блок описания переменной	1	3	5	5	3	2	5	0						
структура программы	4	2	2	5	5	5	2	2	0					
тело программы	4	2	2	2	5	5	2	3	2	0				
переменная	1	4	1	5	2	2	5	2	3	3	0			
константа	1	4	2	5	2	5	5	5	3	3	5	0		
операторы	1	2	2	2	5	5	1	5	2	2	4	4	0	
подпрограмма	4	1	1	2	5	4	2	4	2	3	4	4	5	0

Рисунок 24. Матрица оценок расстояний между концептами первого эксперта

Матрица координат стимулов		
3,41975616	3,81	1. комментарий
-4,254671292	-1,37	2. процедура
-2,300522671	-2,24	3. функция
-2,73731336	2,53	4. операторные скобки
6,901841278	1,19	5. массив
6,086907162	-1,34	6. тип переменной (Тип)
-3,554225277	2,08	7. ключевое слово
4,090148182	-2,03	8. блок описания переменной (БОП)
-2,858994506	-2,26	9. структура программы (СП)
-3,627317418	-0,28	10. тело программы (ТП)
4,109363932	-2,17	11. переменная
1,593809255	1,4	12. константа
-3,407130339	3,32	13. операторы
-3,461651107	-2,64	14. подпрограмма (пп)

♦ оператор	♦ комментарий
оп.скобки ♦	
♦ ключ.слово	♦ константа ♦ массив
♦ ТП	
♦ процедура	тип ♦
СП ♦ функция	БОП ♦ переменная
♦ пп	

Рисунок 26. Матрица координат стимулов и субъективное пространственное пространство первого эксперта

Номер кластера, численность, объекты									
1	1	1							
2	5	2	10	9	14	3	4	7	13
3	3	4	7	13	8				
4	4	5	6	11	8				
5	1	12	11						

Рисунок 28. Результаты кластерного анализа по исходной матрице первого эксперта

	комментарии	процедура	функция	операторные скобки	массив	тип переменной	ключевое слово	блок описания переменной	структура программы	тело программы	переменная	константа	операторы	подпрограмма
Комментарии	0													
Процедура	4	0												
Функция	4	1	0											
операторные скобки	4	5	5	0										
Массив	4	5	5	5	0									
тип переменной	4	5	4	5	3	0								
ключевое слово	4	2	2	5	3	1	0							
блок описания переменной	4	5	5	5	5	1	2	0						
структура программы	1	2	2	2	5	4	2	1	0					
тело программы	1	1	1	1	5	5	3	1	1	0				
Переменная	4	5	5	5	3	1	5	1	4	2	0			
Константа	3	5	5	5	4	5	5	5	4	2	5	0		
Операторы	5	3	3	1	1	3	2	3	2	2	2	4	0	
Подпрограмма	4	1	1	2	4	3	5	4	1	2	3	4	4	0

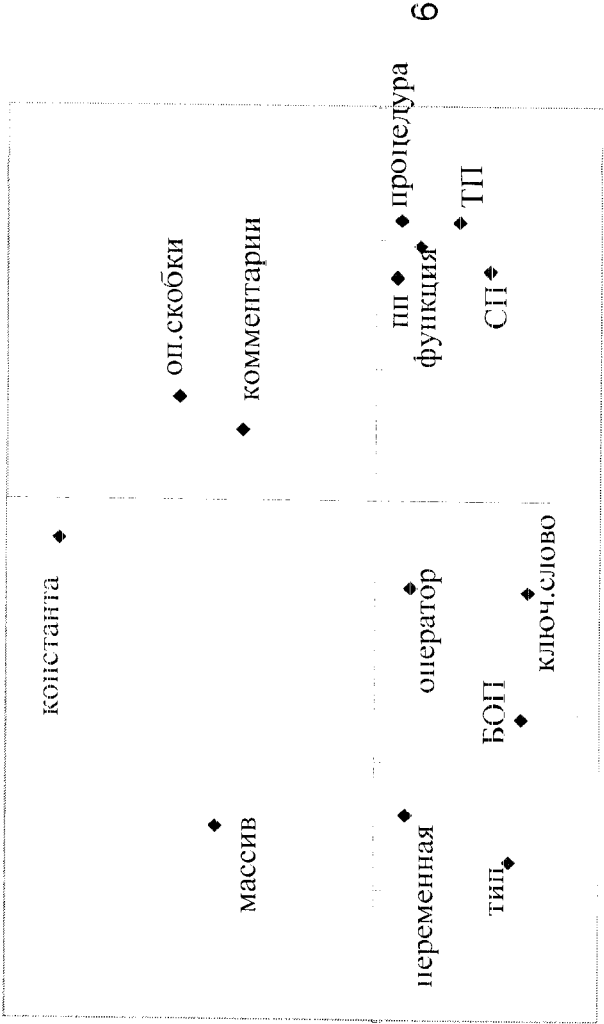
Рисунок 25. Матрица оценок расстояний между концептами второго эксперта

Матрица координат стимулов		
1,08042	2,4766	1. комментарий
4,30491	-0,471	2. процедура
3,90789	-0,863	3. функция
1,56376	3,6765	4. операторные скобки
-5,0325	2,9948	5. массив
-5,5383	-2,577	6. тип переменной
-1,4106	-2,934	7. ключевое слово
-3,3386	-2,799	8. блок описания переменной (БОП)
3,53077	-2,181	9. структура программы (СП)
4,27909	-1,603	10. тело программы (ТП)
-4,8233	-0,598	11. переменная
-0,604	5,9831	12. константа
-1,3509	-0,706	13. операторы
3,43134	-0,401	14. подпрограмма (пп)

Рисунок 27. Матрица координат стимулов и субъективное пространство второго эксперта

Номер кластера, численность, объекты						
1	2	1	12			
2	5	2	3	14	9	10
3	1	4				
4	1	5	6	8	11	7
5	5	6	8	11	7	13

Рисунок 29. Результаты кластерного анализа по исходной матрице первого эксперта



4. перечень основных отличий семантических пространств экспертов и испытуемых, степень выраженности этих отличий в зависимости от уровня теоретических знаний и практических навыков испытуемых.

В соответствии с планом эксперимента, можно сделать следующие выводы. Формируемое семантическое пространство предметной области не зависит от способа подачи материала (т.е. от лектора, читавшего дисциплину). Усвоение основных концептов, установление взаимосвязей между ними, их объединение в метапонятия происходит в первую очередь в процессе совместной с преподавателем и самостоятельной практической деятельности студента. Семантические сети студентов, независимо от преподавателя и формы обучения, имеют ряд общих признаков. Их отличие обуславливается только уровнем теоретической подготовки и, в большей степени, практическими навыками программирования. Говоря другими словами, субъективное семантическое пространство отражают уровень знаний и навыков в программировании его носителя. На основании этого всех испытуемых можно разбить на группы новичков, средних и сильных студентов. Причем семантические пространства сильных студентов и экспертов имеют много общего.

Таким образом, можно сделать вывод, что основная гипотеза исследования о том, что латентные структуры знаний, выявленные с помощью нетрадиционных методов тестирования, могут быть использованы для полноценного и достоверного контроля знаний студентов, – доказана.

3.4 Выводы и рекомендации

Результаты проведенного эксперимента позволяют приступить к разработке специального программного средства, с помощью которого можно проводить контроль знаний не закрытыми тестами, как это в большинстве случаев происходит сейчас, а с помощью открытых вопросов, ответы на которые необходимо давать на естественном языке. Т.е. тестирование будет происходить в условиях контекстно-зависимого естественно-языкового диалога.

Одним из инструментов, предназначенных для реализации контекстно-зависимого естественно-языкового диалога, является автоматизированная обучающая и экспертная система POLARIS. Ее функциональные возможности позволяют создавать компьютерные тьюторы и экспертные программы, которые способны взять на себя большую часть рутинной работы преподавателя по контролю усвоения знаний реальной или виртуальной аудиторией.

Интерфейс большинства современных диалоговых систем организован на основе оконного меню [27, 26, 28]. Этот наиболее простой с точки зрения технической реализации интерфейс между пользователем и программными средствами получил широкое распространение и принят как наиболее удобный. Большое значение в распространении такого интерфейса сыграло наличие стандартных средств его реализации в современном программном обеспечении

компьютеров. Тем не менее, во многих диалоговых системах применение такого интерфейса является не самым эффективным средством организации диалога компьютера и пользователя. Примерами взаимодействия компьютера и пользователя, при которых стандартный интерфейс становится недостаточно эффективным, а часто и вредным с точки зрения некоторых требований к организации диалога могут служить контролирующие, обучающие, экспертные и поисковые системы [16].

Известны шесть обобщенных типов диалога: простой запрос, запрос с синтаксисом для ответа, запрос для выборочного ответа (меню), запрос для выборочно конструируемого ответа, команда, фраза на естественном языке. При организации диалога средствами стандартного оконного интерфейса возможна реализация пяти первых типов диалога. Направление развития диалоговых средств в настоящее время в большей степени основано на применении именно этих типов диалога. Но все эти пять типов – суть простые или сложные формы дихотомического запроса, из которых преимущественное распространение получило меню.

Для управления программными средствами и операционными системами информативных возможностей первых простейших типов диалога вполне достаточно. Но для эффективной реализации педагогических целей и задач в системах учебного назначения и особенно в контролирующих и обучающих программах возможностей применяемых в настоящее время типов диалога явно не достаточно. Меню как основа тестовых вопросов – очень сложный инструмент при кажущейся простоте его применения.

Эта простота во многом и предопределяет низкую эффективность использования меню, а то и серьёзный вред, который наносится процессу обучения при неправильном применении этого инструмента. Основные негативные моменты, присущие тестовым вопросам типа меню, связаны с тем, что они дают лишь очень приближённую информацию о степени усвоения обучающимся учебного материала. Ко многим вопросам сложно подбирать подходящие дистракторы, информация, заключенная в дистракторах серьёзно препятствует правильному усвоению учебного материала, а порой и профанирует суть контроля. Наконец, бывают случаи, когда постановкой вопроса типа меню проконтролировать степень усвоения и понимания учебного материала крайне сложно. Как правило, это вопросы на проверку креативных знаний, ответ на которые показывают ход и логику мышления обучающегося.

Всех этих негативных моментов лишён контроль усвоения знаний с применением открытых вопросов и умением анализировать семантику ответа обучающегося [18, 28]. Для этого необходимо иметь программные средства, позволяющие реализовывать в компьютерных системах диалог на естественном языке.

Традиционно реализации диалога человек-компьютер на естественном языке основываются на попытках решения общей задачи языкового анализа. Несмотря на серьёзные успехи в теории анализа языка, на практике создание таких систем ещё довольно проблематично. В ряде случаев, при решении частных задач диалогового взаимодействия бывает достаточно провести анализ естественно-языкового высказывания лишь в определённом контексте. Именно

такие ситуации часто возникают в системах управления и диспетчеризации, информационного поиска, обучения и экспертных системах.

Эти ситуации характеризуются следующими основными исходными моментами:

- элементарный акт диалогового взаимодействия инициализируется компьютером;

- контекст данного акта диалогового взаимодействия определён заданной ситуацией или запросом на действие пользователя.

В этих случаях для анализа введённого пользователем ответа на поставленный вопрос нет необходимости применять все средства, предназначенные для решения общей задачи языкового анализа. Частичный языковой анализ высказывания в контекстно-зависимой ситуации существенно упрощается [15].

При этом нет необходимости в применении обобщенных баз знаний, правил редуций и словообразования. Достаточно иметь частичную и упрощённую базу форм сочетания и взаиморасположения знаковых символов, предположительно ожидаемых в высказывании, и правил проверки наличия этих форм в строке ответа. Эту базу будем называть семантической базой ожидаемых высказываний.

При таком подходе формы сочетания знаковых символов, которые обязательно должны содержаться (или наоборот отсутствовать) в строке высказывания, будем называть дескриптором. А формы, определяющие, в каком взаиморасположении дескрипторы должны находиться в правильном ответе, будем называть семантическими масками. Последовательную проверку строки ответа на наличие в ней дескрипторов, расположенных в соответствии с маской, будем называть демаскированием строки.

Демаскируя строку ответа различными масками в определённой последовательности, можно сделать вывод о соответствии введённого высказывания той или иной семантике.

Описанный процесс и есть процесс анализа контекстно-зависимого высказывания.

Для того чтобы провести анализ, необходимо выявить, составить и записать определённым образом вышеперечисленные формы для возможных и ожидаемых семантических конструкций в строке высказывания. Процесс определения того, какие дескрипторы должны находиться (или отсутствовать) в строке ответа и как они должны быть взаимно расположены в ней, будем называть маскированием строки ожидаемого ответа.

Дескрипторы и семантические маски являются семантической базой для анализа синтаксиса и семантики высказывания. Процесс построения этой базы для каждого конкретного акта диалогового взаимодействия является сутью построения конкретного анализатора. Записанную по определённым правилам семантическую базу ожидаемых высказываний с указанием порядка и правил демаскирования строки, в которой высказывание записано, будем называть структурой анализатора или просто анализатором.

Задание (описание) в семантической маске взаимного расположения дескрипторов в строке высказывания и последовательности демаскирования

строки совокупностью семантических масок производится на основе применения логических операций и выражений. Поэтому такой подход к анализу высказывания следует называть логико-дескриптивным.

Процесс анализа строки пользователя конкретным анализатором проводится автоматически на основе алгоритма, заложенного в структуре анализатора. Процесс построения структуры анализатора пока автоматизировать не удалось. Он сводится к определению семантической базы ожидаемых высказываний и записи определённым образом её элементов, а так же порядка и правил демаскирования строки высказывания.

При анализе высказывания программные средства используют структуру анализатора как некую программу для проведения определённых действий на введённой пользователем строке.

Программные средства анализатора, а так же средства реализации модели естественно-языкового диалога в компьютерных обучающих системах разработаны и многократно применялись при создании многих контрольно-обучающих программ.

Для формального описания структуры анализатора в каждом элементарном акте диалога автор сценария должен знать правила работы программных средств анализатора и методику его создания. Для описания общего сценария диалога автору необходимо ознакомиться с методикой его формирования и реализации.

Таким образом, несмотря на то, что пока решение вопросов распознавания синтаксиса и семантики высказываний в общем виде еще не достаточно разработано, распознавание семантики свободно конструируемых ответов в контексте заданного вопроса вполне реально и возможно. Разработанная методика создания анализаторов контекстно-зависимых высказываний на базе применения логико-дескриптивного подхода позволяет авторам непрограммистам создавать довольно сложные естественно-языковые вопросно-ответные структуры, используя разработанные для этого программные средства.

Метод логико-дескриптивного анализа контекстно-зависимых высказываний подтвердил свою работоспособность в течение многих лет. Дальнейшее развитие логико-дескриптивного метода позволяет при современном развитии технических и программных средств создавать сетевые диалоговые естественно-языковые программы и, в частности, компьютерные интеллектуальные тьюторы для их применения в системах открытого и дистанционного образования. Под компьютерным интеллектуальным тьютором мы понимаем такую обучающую программу, которая моделирует естественно-языковой диалог с пользователем и ведёт обучающую беседу. Причём, вопросы задаются преимущественно открытого типа. Анализируя свободно-конструируемые ответы пользователя на поставленные вопросы, тьютор определяет полную или частичную верность ответов в контексте диалога, а по результатам анализа даёт пояснения и делает заключения.

Задачи по выявлению уровня незнания и его конкретизации, разъяснению основных положений учебного материала по результатам контроля в каждом элементарном акте диалога индивидуально каждому обучаемому, сбору и

обработке контрольной информации может взять на себя компьютерный тьютор. Разработаны и многократно опробованы методика и технология создания естественно-языковых диалоговых систем на основе применения логико-дескриптивного метода анализа контекстно-зависимых высказываний. Очевидно, что такие системы найдут широкое применение в открытом образовании [16].

По результатам проведенного исследования можно сформировать семантическую базу для проверки знаний по дисциплине «Программирование на алгоритмических языках» нетрадиционным методом. Выявленная методика выявления семантических пространств с определенной доработкой и оформлением может применяться для выявления семантической базы и по другим дисциплинам.

ЗАКЛЮЧЕНИЕ

Магистерская диссертация «Применение ЕЯ-интерфейсов в образовательном процессе» посвящена исследованию возможности применения латентных (скрытых) структур знаний для контроля знаний студентов и последующей разработке специального программного средства, осуществляющего такой контроль.

Естественно-языковые интерфейсы – это комплекс средств, позволяющий неподготовленному пользователю общаться с компьютером на естественном языке. Такие программы также называют естественно-языковыми системами. В теории такое общение ничем не ограничено, на практике же не существует ни одной программы, полностью воспроизводящей человеческое общение. Большинство программ, работающих с естественным языком, имеет массу ограничений. Одно из самых главных – применение конкретного интерфейса в конкретной предметной области.

Если говорить о создании конкретного ЕЯ-интерфейса в конкретной предметной области, то ее выявление можно организовать как процесс извлечения экспертных знаний инженером-когнитологом с последующей концептуализацией и структуризацией выявленных знаний. Используя термины инженерии знаний, для полного описания предметной области необходимо построить ее пирамиду знаний.

Пирамида знаний, или семантическое представление предметной области – это формальное представление концептов (определений, терминов, понятий) предметной области и их взаимосвязей. На более высоком уровне концептуализации понятия объединяются в метапонятия. Несколько метапонятий также могут сливаться в новое на еще более верхнем метакуровне.

В данной работе рассматривается семантическое пространство предметной области «Программирование на алгоритмических языках». Такой выбор обусловлен несколькими причинами. Во-первых, эта предметная область хорошо изучена, ее описание в том или ином виде представлено в множестве учебников, практических руководств, методических указаний. Собственно, уже оглавление хорошо написанного учебника по программированию уже показывает основные метапонятия и концепты этой области. Т.е., по сути, мы имеем эталон, с которым можно соотносить полученные в результате исследования пирамиды знаний (семантические пространства) экспертов и студентов.

Во-вторых, эта дисциплина является одной из базовых для любого программиста, она изучается на первом курсе, а полученные знания и навыки затем оттачиваются студентами в течение всего срока обучения. Таким образом, в распоряжении исследователя были студенты всех четырех курсов и различного уровня подготовки. Этот факт, конечно же, повысил достоверность проведенного исследования и позволил генерализировать полученные выводы.

Уже упоминавшееся исследование проходило в четыре этапа. Его участниками стали 9 экспертов и 25 студентов. В качестве экспертов выступали опытные преподаватели, ведущие дисциплину, и программисты-практики. Студенты привлекались из разных групп разных форм обучения.

В результате эксперимента была получена вся необходимая информация для разработки программного средства, осуществляющего контроль знаний студентов на основе открытых вопросов:

1. список понятий предметной области «Программирование на алгоритмических языках», характерный для экспертов и студентов.

2. взаимосвязи между концептами.

3. субъективное семантическое пространство предметной области экспертов, полученное в результате статистической обработки данных.

4. перечень основных отличий семантических пространств экспертов и испытуемых, степень выраженности этих отличий в зависимости от уровня теоретических знаний и практических навыков испытуемых.

Анализ полученных данных позволил сделать следующие выводы. Формируемое семантическое пространство предметной области не зависит от способа подачи материала (т.е. от лектора, читавшего дисциплину). Усвоение основных концептов, установление взаимосвязей между ними, их объединение в метапонятия происходит в первую очередь в процессе совместной с преподавателем и самостоятельной практической деятельности студента. Семантические сети студентов, независимо от преподавателя и формы обучения, имеют ряд общих признаков. Их отличие обуславливается только уровнем теоретической подготовки и, в большей степени, практическими навыками программирования. Говоря другими словами, субъективное семантическое пространство отражают уровень знаний и навыков в программировании его носителя. На основании этого всех испытуемых можно разбить на группы новичков, средних и сильных студентов. Причем семантические пространства сильных студентов и экспертов имеют много общего.

Выявленная пирамида знаний предметной области «Программирование на алгоритмических языках», по сути, является семантической базой, необходимой для диалоговых естественно-языковых программ-тьюторов. Под компьютерным интеллектуальным тьютором мы понимаем такую обучающую программу, которая моделирует естественно-языковый диалог с пользователем и ведёт обучающую беседу. Причём вопросы задаются преимущественно открытого типа. Анализируя свободно-конструируемые ответы пользователя на поставленные вопросы, тьютор определяет полную или частичную верность ответов в контексте диалога, а по результатам анализа даёт пояснения и делает заключения.

С задачами по выявлению уровня незнания и его конкретизации, разъяснению основных положений учебного материала по результатам контроля в каждом элементарном акте диалога индивидуально каждому обучаемому, сбору и обработке контрольной информации компьютерный тьютор справляется на базе той самой семантической сети, выяснению которой была посвящена данная работа.

Очевидно, что естественно-языковые системы при всем их потенциале и мощи найдут широкое применение в открытом образовании.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. И.С.Евдокимова Естественные-языковые системы: курс лекций. – Улан-Удэ: издательство ВСГТУ, 2006. – 99с.
2. И.П.Карпова Исследование и разработка подсистемы контроля знаний в распределенных автоматизированных обучающих системах. – М.: издательство МГТУ, 2002. – 200с.
3. Искусственный интеллект. Справочник. Книга 1. Системы общения и экспертные системы - М.: Радио и связь, 1987. – 315с.
4. В.В.Корнеев, А.Ф.Гарев, С.В.Васютин, В.В.Райх Базы данных. Интеллектуальная обработка информации. – М.: Нолидж, 2000. – 352с.
5. Д.С.Джахаев Психологическая структура процесса обучения. – издательство КазГУ, 2004. – 207с.
6. Т.А.Гаврилова, А.Ф.Хорошевский базы данных интеллектуальных систем. – СПб.: - Питер, 2000. – 384с.
7. Г.Биркгофф Математика и психология. – М.: Советское радио, 1977. – 96с.
8. Р.Шенк, Л.Бирнбаум, Дж.Мей К интеграции семантики и прагматики // Новое в зарубежной лингвистике. Компьютерная лингвистика. 1989 Вып.14. – М.: - Прогресс
9. И.Н.Горелов. Разговор с компьютером: психолингвистический аспект проблемы. – М.: Наука, 1987
10. Информатика. Энциклопедический словарь для начинающих /Сост. Д.А.Поспелова – М.: Педагогика-Пресс, 1994. – 352с.
11. Н.М.Кук, Дж. Макдональд Формальная методология приобретения и представления экспертных знаний // ТИИЭР 1986 т.74 № 10 с.145-155
12. Д.А.Поспелов Ситуационное управление: теория и практика. – М.: Наука, 1986
13. Б.М.Величковский Современная когнитивная психология. – М.: издательство Московского университета, 1982. – 338с.
14. Б.А.Куркин Проблемы представления информации для принятия решений при управлении процессами в промышленности и бизнесе // Сборник трудов конференции Диалог 2008 с.15-19
15. Д.А.Поспелов Моделирование рассуждений. Опыт анализа мыслительных актов. – М.: Радио и связь, 1989. – 184с.
16. Основы научных исследований /сост. В.И.Крутов, В.В.Попов. – М.: Высшая школа, 1989. – 400с.
17. А.Д.Наследов математические методы психологического исследования. Анализ и интерпретация данных. Учебное пособие. – СПб.: Речь, 2004 – 392 с.
18. Д.Ш. Сулейманов Аналитический обзор отечественных и зарубежных работ обработки естественного языка в аспекте прагматически-ориентированного подхода. – издательство Казанского ГУ, 2000. – 64с.
19. А.И.Стригун, В.А. Стригун Организация контекстно-зависимого естественно-языкового диалога // Сборник трудов международной конференции Диалог 2008 с. 35-43

20. Р. Беллман, Л. Заде Принятие решений в расплывчатых условиях // Сборник переводов Вопросы анализа и процедуры принятия решений М.: Мир, 1976. - с.173-215
21. Л.Н.Изотова The application of latent knowledge structures in the educational process // Материалы международной научно-практической конференции Казахстан и мировые языки, т.2, 2008 с. 175 –177
22. А.Н.Баранов Введение в прикладную лингвистику: учебное пособие. – М.: Едиториал УРСС, 2003 – 360 с.
23. Современная американская лингвистика: Фундаментальные направления . под ред. А.А.Кибрика, И.М.Кобозевой и И.А.Секериной – М.: Едиториал УРСС, 2002 – 480 с.
24. А.В. Гордеев, А.Ю. Молчанов Системное программное обеспечение – Санкт-Петербург: Питер, 2001
25. Л.Бек Введение в системное программирование. – М.: Мир, 1988
26. О.В.Михеев Комплексный контроль знаний о дискретных технических системах //Материалы международной научно-практической конференции Информационные технологии в образовании www.ito.ru
27. Титенко С.В., Гагарін О.О. Семантична модель знань для цілей організації контролю знань у навчальній системі. // Сборник трудов международной конференции «Интеллектуальный анализ информации-2006». – Київ: Просвіта, 2006. – С.298-307.
28. материалы сайта www.setlab.net
29. С.А. Немнюгин Turbo Pascal. Учебник – СПб.: Питер, 2000. – 196с: ил.