

DOI 10. 23859/1994-0637-2019-1-88-17
УДК 004.855

© Касторнов А. Ф., Касторнова В. А., 2019

Касторнов Анатолий Федосеевич

Кандидат педагогических наук, профессор,
Череповецкий государственный
университет
(Череповец, Россия)
E-mail: a_kastornov@mail.ru

Kastornov Anatoly Fedoseevich

PhD in Pedagogical Sciences, Professor,
Cherepovets State University
(Cherepovets, Russia)
E-mail: a_kastornov@mail.ru

Касторнова Василина Анатольевна

Кандидат педагогических наук, доцент,
Институт управления образованием
Российской академии образования
(Москва, Россия)
E-mail: kastornova_vasya@mail.ru

Kastornova Vasilina Anatol'evna

PhD in Pedagogical Sciences,
Associate Professor,
Institute of Management
of Education of The Russian Academy
of Education
(Moscow, Russia)
E-mail: kastornova_vasya@mail.ru

**ИНФОРМАЦИОННОЕ
МОДЕЛИРОВАНИЕ НА ОСНОВЕ
ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ
КАК ИНСТРУМЕНТ
СИСТЕМАТИЗАЦИИ ЗНАНИЙ
В ПРЕДМЕТНОЙ ОБЛАСТИ
«ИНФОРМАТИКА»**

**INFORMATION MODELLING
ON THE BASIS OF INTELLECTUAL
SYSTEMS AS THE TOOL
OF SYSTEMATIZATION
OF KNOWLEDGE
IN THE SUBJECT DOMAIN
OF “COMPUTER SCIENCE”**

Аннотация. Рассматриваются основные этапы построения информационных моделей как инструмента систематизации знаний при решении задач классификации объектов предметной области «Информатика» с помощью экспертных систем и нейронных сетей. При построении информационных моделей определяются методы минимизации числа атрибутов созданной модели на предмет выявления существенных характерных свойств распознаваемых объектов. При решении этой проблемы выделяются эффективность и преимущества нейронных сетей.

Abstract. The authors consider the main stages of creation of information models as the tool of systematization of knowledge at the solution of classification problems of objects of subject domain of “Computer science” by means of expert systems and neural networks. At creation of information models methods of minimization of number of attributes of the created model regarding identification of essential characteristic properties of the recognizable objects are defined. At the solution of this problem the efficiency and advantages of neural networks are distinguished.

Ключевые слова: информационное моделирование, атрибуты информационной модели, систематизация знаний, классификация (распознавание) объектов, интеллектуальные системы, нейроэмуляторы, обучение экспертных систем и нейронных сетей

Keywords: information modeling, attributes of information model, systematization of knowledge, classification (recognition) of objects, intellectual systems, neuroemulators, training of expert systems and neural networks

Введение

Человечество в своей деятельности (научной, образовательной, технологической, художественной и др.) постоянно создает и использует модели окружающего мира, т.е. формирует образ того объекта (процесса или явления), с которым ему приходится иметь дело. Каждый объект имеет большое количество различных свойств. В процессе построения модели выделяются главные, наиболее существенные для проводимого исследования свойства. Каждый объект имеет большое количество различных свойств. В математике и информатике можно строить информационные модели некоторых множеств однородных объектов, таких как геометрические фигуры, классы компьютеров, типы программного обеспечения и т.п. В процессе построения модели выделяются главные, наиболее существенные для проводимого исследования свойства. Таким образом, основной целью моделирования является изучение и исследование объекта или явления, для которого модель построена.

Моделирование как одна из технологий решения практических задач имеет различные сферы приложения, одной из которых является распознавание образов, понимаемое в широком смысле этого слова. При этом под образами мы понимаем некоторое множество объектов определенной предметной области (например, геометрические фигуры: треугольник, трапеция, параллелограмм, окружность, эллипс, пирамида, призма, конус, цилиндр и пр.). Все эти фигуры можно разбить на классы: принадлежащие двумерному или трехмерному пространству; имеющие в своей основе прямые или кривые линии. Каждый из объектов имеет свой набор характерных признаков (атрибутов), отличающих их друг от друга. Изучение этих отличительных признаков составляет одно из направлений (векторов) исследования характерных особенностей этих фигур.

Примерами технической реализации построения информационных моделей являются интеллектуальные системы: *экспертные системы* и *нейронные сети*. В статье рассматриваются характерные особенности использования в сфере образования этих двух технологических решений информационного моделирования как инструмента систематизации знаний в информационно-образовательном пространстве предметной области «Информатика». С помощью этих систем можно решать неформализованные задачи, среди которых можно выделить распознавание объектов по их характерным признакам. Примером использования указанных средств является изучение вопросов аппаратного и программного обеспечения ЭВМ – умение определять тип ЭВМ, вид программного обеспечения (базового и прикладного), тип языка программирования, а точнее системы программирования. При решении таких задач основным является выделение характерных свойств (признаков, атрибутов) этих объектов. Эту проблему можно решить путем построения соответствующей

информационной модели, с помощью которой можно было бы «узнавать» геометрические фигуры по их характерным признакам. Информационную модель необходимо построить так, чтобы свести набор этих признаков к минимальному.

Основная часть

Современные компьютерные (информационные) технологии предоставляют широкие возможности построения информационных моделей, используя при этом как традиционные (алгоритмические) подходы, так и технологии искусственного интеллекта, ярким проявлением которых являются экспертные системы (ЭС) и нейронные сети (НС) [3, 5]. И те и другие можно использовать для построения информационных моделей, ориентированных на распознавание объектов или их кластеризацию. В этой работе используется оболочка ЭС, созданная в среде языка Паскаль (рис. 1), и эмулятор НС Deductor Academic (рис. 2).

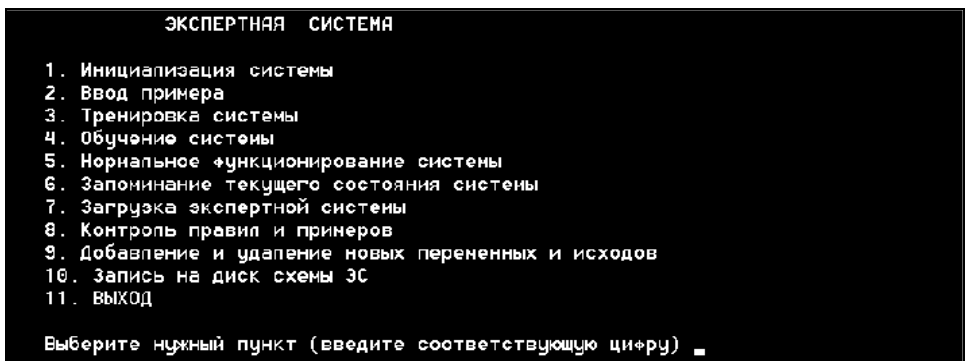


Рис. 1. Главное меню оболочки ЭС



Рис. 2. Эмулятор НС Deductor Academic

Рассмотрим указанные подходы построения информационных моделей по распознаванию объектов с помощью этих программных средств на примере классификации и распознавания видов ЭВМ.

Экспертные системы

Для распознавания видов ЭВМ построим двухузловую экспертную систему [2], [4], которая сначала будет определять принадлежность узнаваемого вида ЭВМ к его типу (суперЭВМ, большие ЭВМ, мини-ЭВМ, микроЭВМ и компьютерная телефония), а затем – собственно сам вид ЭВМ (рис. 3).

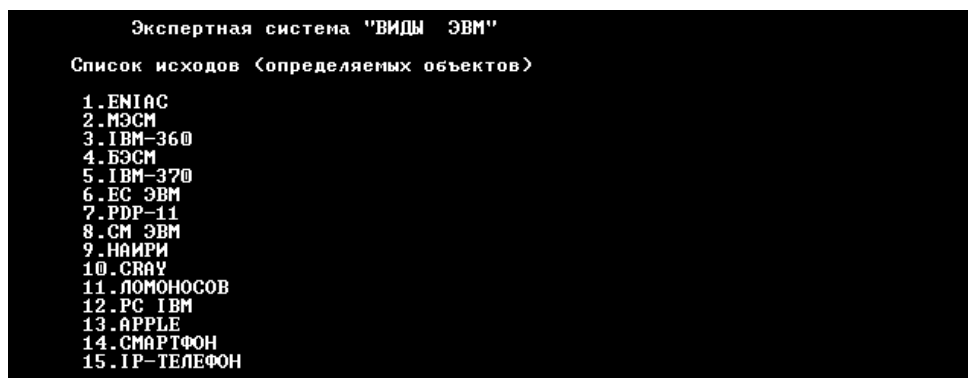


Рис. 3. Список объектов распознавания

Для каждого из этих узлов необходимо определить список его атрибутов (характерных свойств), которые в терминологии ЭС называются *переменными*. Их число в списках может быть произвольным, но все переменные каждого узла должны описывать отличительные признаки всех объектов (*исходов*). Одни из этих признаков могут быть уникальными, т.е. характеризующими только один исход, другие – общими для нескольких объектов. Например, свойство «Блочная конструкция» присуще как большим ЭВМ, так и суперкомпьютерам, а «Открытость архитектуры» – только РС IBM.

Первоначально создается прототип информационной модели, в который входят 2–3 переменные для каждого узнаваемого объекта. Подбор этих переменных осуществляется из знаний эксперта (преподавателя) данной предметной области. Рассмотрим пример создания экспертной системы.

ЭКСПЕРТНАЯ СИСТЕМА «ВИДЫ ЭВМ»

УЗЕЛ 1

Переменные:

1. МНОГОПРОЦЕССОРНАЯ ЭВМ
2. МНОГОПОЛЬЗОВАТЕЛЬСКИЙ РЕЖИМ РАБОТЫ
3. БЛОЧНАЯ КОНСТРУКЦИЯ
4. ДВУХПРОЦЕССОРНАЯ АРХИТЕКТУРА
5. СЛУЖИТ СЕРВЕРОМ
6. СИНТЕЗ ПК И ТЕЛЕФОНА
7. МОБИЛЬНОСТЬ
8. НАСТОЛЬНЫЙ ПК
9. МНОГОТЕРМИНАЛЬНОСТЬ

Исходы:

1. БОЛЬШИЕ ЭВМ
2. МИНИ-ЭВМ
3. МИКРОЭВМ
4. СУПЕРКОМПЬЮТЕРЫ
5. ИНТЕРНЕТ-ТЕЛЕФОНИЯ

УЗЕЛ 2

Переменные:

1. ПРЕДШЕСТВЕННИК ПК
2. ОТКРЫТОСТЬ АРХИТЕКТУРЫ
3. ПУЛЬТ УПРАВЛЕНИЯ
4. ПЕРВАЯ СОВЕТСКАЯ ЭВМ 3-го ПОКОЛЕНИЯ
5. WI-FI ДОСТУП К СЕТИ
6. ШТУЧНОЕ ПРОИЗВОДСТВО
7. КАРМАННАЯ ВЕРСИЯ ПК
8. ОТЕЧЕСТВЕННАЯ ЭВМ
9. ЦВЕТНОЙ ДИСПЛЕЙ
10. НАЛИЧИЕ ОС
11. НАЛИЧИЕ СЕТЕВОЙ ОС
12. НАЛИЧИЕ ОПТИЧЕСКОГО ДИСКОВОДА
13. ПОСТРОЕНА НА ТРАНЗИСТОРАХ

Исходы (см. рис. 3)

Представленные списки переменных и исходов составляют систему фактов информационной модели ЭС и помещаются в нее с помощью пункта меню *Инициализация системы* оболочки ЭС (см. рис. 1). Затем, используя пункт *Ввод примеров*, в каждом исходе переменным присваивают значение 0 или 1 в зависимости от их соответствующих признаков. Например, исходу СУПЕРКОМПЬЮТЕРЫ узла 1 присущи свойства (переменные) 1, 2, 3, 5 и 9, а исходу ЛОМОНОСОВ узла 2 – признаки 6, 8, 9, 10, 11, 12.

Далее с помощью пункта меню *Тренировка системы* формируют систему правил создаваемой ЭС (табл. 1) как составную часть базы знаний, представляющую собой матрицу весовых коэффициентов, следуя которой, заложенный в ЭС алгоритм (распознаватель) определяет объект по его характерным признакам.

Таблица 1

Система правил

Переменные	Номер исхода				
	1	2	3	4	5
1. МНОГОПРОЦЕССОРНАЯ ЭВМ	-1	-1	-1	1	-1
2. МНОГОПОЛЬЗОВАТЕЛЬСКИЙ РЕЖИМ РАБОТЫ	-1	-1	-1	1	-1
3. БЛОЧНАЯ КОНСТРУКЦИЯ	1	-2	-2	-1	-2
4. ДВУХПРОЦЕССОРНАЯ АРХИТЕКТУРА	-1	1	-1	-1	-1
5. СЕРВЕР	-3	-1	-1	-1	-3
6. СИНТЕЗ ПК И ТЕЛЕФОНА	-1	-1	-1	-1	1
7. МОБИЛЬНОСТЬ	-1	-1	-1	-1	1
8. НАСТОЛЬНЫЙ ПК	-1	-1	1	-1	-1
9. МНОГОТЕРМИНАЛЬНОСТЬ	-2	0	-2	0	-2

Из табл. 1 видно, что переменные 1 и 2 присущи исходу СУПЕРКОМПЬЮТЕРЫ (значение «1»). В строке НАСТОЛЬНЫЙ ПК значение номера исхода равно 1, что соответствует исходу МИКРОКОМПЬЮТЕР.

Далее с помощью пункта меню *Нормальное функционирование системы* следует протестировать ЭС на предмет правильности ее работы по распознаванию объектов. Как правило, на этом этапе обнаруживаются неправильно опознаваемые объекты. В этом случае рекомендуется дообучить систему с помощью пункта *Обучение системы*. Эта опция оболочки работает так же, как и *Ввод примеров*, но имеет существенные отличия: при вводе примеров выкладываются все переменные, а при обучении – только некоторые из них. При обучении коэффициенты массива системы правил модифицируются и ЭС перестает ошибаться.

Иногда многократное дообучение не приводит к устранению ошибки. Случается, что система не отличает два близких друг к другу объекта. Например, большие ЭВМ и суперкомпьютеры (по размерам и другим особенностям) имеют много общего, поэтому для их разграничения нужно добавить дополнительное свойство (*Многопроцессорность* или *Наличие ОС*). Расширение структуры информационной модели можно произвести с помощью пункта *Добавление и удаление новых переменных и исходов*. После этой операции систему следует переобучить (см. рис. 1, пункт 2) или дообучить (см. рис. 1, пункт 4).

Если система обучена и работает безошибочно, следует подумать о ее минимизации, об удалении из нее незначимых переменных, т.е. привести информационную модель к формату содержания в ней только существенных свойств распознаваемых объектов. В контексте нашей задачи термин «существенные» надо рассматривать не с точки зрения сущности того или иного объекта, а как показатель взаимосвязи данного объекта с другими объектами изучаемой предметной области. Поиск лишних переменных можно вести и «вручную», и с помощью самой оболочки экспертной системы.

Ручной поиск этих переменных можно осуществлять путем детального изучения их списка. Свойства *Многопользовательский режим работы* и *Многотерминальность* равнозначны свойству *Суперкомпьютеры*, поэтому их можно заменить одним свойством – *Многопроцессорность*. Эти предположения надо проверять путем их удаления из базы фактов системы с последующим ее переобучением. Лишние переменные можно также обнаружить в процессе изучения протокола работы системы в режиме ее нормальной работы. В протоколе фиксируются все задаваемые системой вопросы. В исходном списке можно обнаружить те переменные, которых нет в протоколе.

Более надежный путь обнаружения и удаления лишних переменных – обучение системы. Например, при обучении распознаванию объекта ENIAC система задает вопросы, отличающие его от других объектов путем отсутствия всех свойств, кроме свойства *Штучное производство*, которое присуще только большим ЭВМ. Система не задает вопросов о наличии свойств *Пульт управления*, *Цветной дисплей* и *Наличие оптического дисковода* (рис. 4).

```

Узел 2
Свойство "ОТЕЧЕСТВЕННАЯ ЭВМ" есть у объекта "" <1/0>? 0
Свойство "ПРЕДШЕСТВЕННИК ПК" есть у объекта "" <1/0>? 0
Свойство "ШТУЧНОЕ ПРОИЗВОДСТВО" есть у объекта "" <1/0>? 1
Свойство "МОБИЛЬНОСТЬ" есть у объекта "" <1/0>? 0
Свойство "КАРМАННАЯ ВЕРСИЯ ПК" есть у объекта "" <1/0>? 0
Свойство "ПЕРВАЯ СОВЕТСКАЯ ЭВМ 3-го ПОКОЛЕНИЯ" есть у объекта "" <1/0>? 0
Свойство "ОТКРЫТОСТЬ АРХИТЕКТУРЫ" есть у объекта "" <1/0>? 0
Свойство "ПОСТРОЕНА НА ТРАНЗИСТОРАХ" есть у объекта "" <1/0>? 0
Свойство "НАЛИЧИЕ ОС" есть у объекта "" <1/0>? 0
Свойство "WI-FI ДОСТУП К СЕТИ" есть у объекта "" <1/0>? 0
Могу я вывести возможный исход : ENIAC [y/n]
    
```

Рис. 4. Режим обучения исходу ENIAC

При распознавании других объектов наблюдается то же самое. Это позволяет обнаружить лишние переменные во втором узле базы знаний ЭС. В первом узле лишними являются *Многопользовательский режим работы*, *Сервер* и *Многотерминальность*. В результате удаления этих переменных в базе фактов ЭС создается информационная модель, содержащая только существенные свойства видов и типов ЭВМ. В результате получается следующий укороченный список переменных:

УЗЕЛ 1

Переменные:

1. МНОГОПРОЦЕССОРНАЯ ЭВМ
2. БЛОЧНАЯ КОНСТРУКЦИЯ
3. ДВУХПРОЦЕССОРНАЯ АРХИТЕКТУРА
4. СИНТЕЗ ПК И ТЕЛЕФОНА
5. МОБИЛЬНОСТЬ
6. НАСТОЛЬНЫЙ ПК

УЗЕЛ 2

Переменные:

1. ПРЕДШЕСТВЕННИК ПК
2. ОТКРЫТОСТЬ АРХИТЕКТУРЫ
3. ПЕРВАЯ СОВЕТСКАЯ ЭВМ 3-го ПОКОЛЕНИЯ
4. WI-FI ДОСТУП К СЕТИ
5. ШТУЧНОЕ ПРОИЗВОДСТВО
6. КАРМАННАЯ ВЕРСИЯ ПК
7. ОТЕЧЕСТВЕННАЯ ЭВМ
8. НАЛИЧИЕ ОС
9. НАЛИЧИЕ СЕТЕВОЙ ОС
10. ПОСТРОЕНА НА ТРАНЗИСТОРАХ

Таким образом, представленная оболочка ЭС не только создает информационную модель по распознаванию объектов, но и позволяет совершенствовать ее с точки зрения минимизации числа необходимых для этой цели характерных свойств.

Нейронные сети

Информационные модели распознавания объектов можно создавать по технологии нейронных сетей с помощью эмулятора Deductor Academic [1]. Эта технология имеет много схожего с созданием ЭС – в обоих случаях формируется список исходных данных, имеющих следующие названия: переменные – *входы (атрибуты)*; исходы – *выходы*. Меняется также способ их ввода в базу знаний. В ЭС переменные и исходы вводятся отдельно с помощью пункта *Ввод примеров* (см. рис. 1), а в НС формируется текстовый файл в виде таблицы (атрибуты – столбцы, выходы – строки), которая импортируется в Deductor.

Информационная модель в Deductor строится, как нейронная сеть, с помощью так называемого *Мастера обработки* (например, *Дерево решений*, рис. 5).

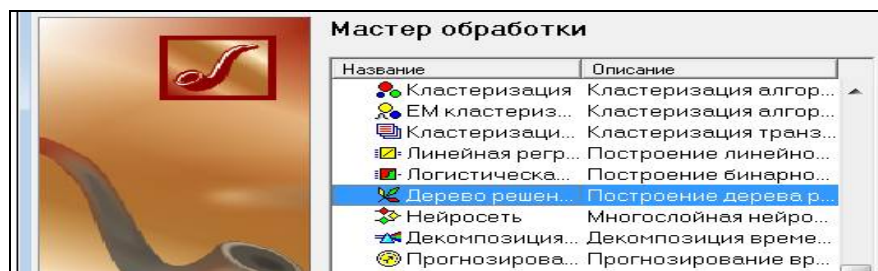


Рис. 5. Мастера обработки Deductor

Мастер *Дерево решений* (decision trees) – одно из самых мощных средств решения задачи отнесения какого-либо объекта (строки набора данных) к одному из заранее известных классов. *Дерево решений* – это классификатор, полученный из обучающего множества, содержащего объекты и их характеристики.

На примере классификации видов ЭВМ рассмотрим вопросы поиска атрибутов для распознавания объектов и минимизацию их числа с использованием *Дерева решений*. Информационная модель нейронной сети для классификации объектов дает результат в виде дерева, построенного с использованием управляющей алгоритмической структуры «Если-то-иначе» (рис. 6).

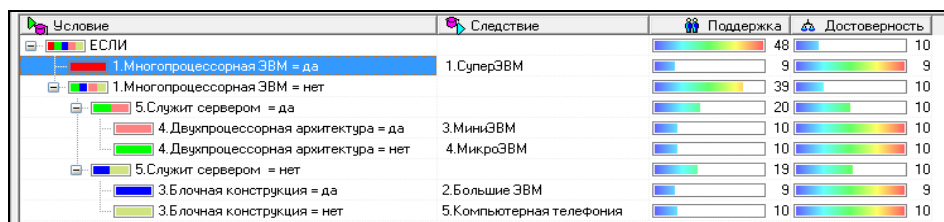


Рис. 6. Дерево решений классификации типов ЭВМ

Мастер *Дерево решений* в качестве отображения результатов построения НС позволяет получать визуализаторы: *Значимость атрибутов*, *Таблица сопряженности*, *Что-если* и *Правила*. С помощью визуализатора *Значимость атрибутов* можно сразу же увидеть лишние атрибуты (рис. 7).

Целевой атрибут: ТИПЫ ЭВМ			
№²	Номер	Атрибут	Значимость, %
1	5	5. Служит сервером	35,004
2	1	1. Многопроцессорная ЭВМ	30,009
3	4	4. Двухпроцессорная архитектура	17,959
4	3	3. Блочная конструкция	17,027
5	8	8. Настольный ПК	0,000
6	9	9. Многотерминальность	0,000
7	7	7. Мобильность	0,000
8	2	2. Многопользовательский режим ...	0,000
9	6	6. Синтез ПК и телефона	0,000

Рис. 7. Значимость атрибутов классификации типов ЭВМ

Информационная модель должна включать в себя только 4 атрибута (см. рис. 7), что почти соответствует результатам анализа определения излишних переменных, проведенного при создании ЭС. После импорта в *Deductor* таблицы с укороченным списком получается дерево (рис. 8), в котором значимость атрибутов содержит уже меньшее число незначащих свойств объектов (рис. 9).

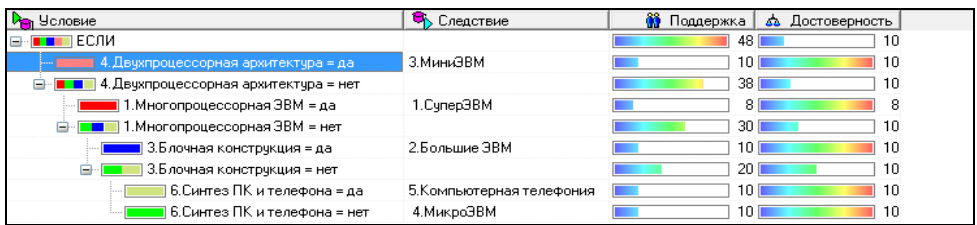


Рис. 8. Дерево решений классификации типов ЭВМ (укороченный список атрибутов)

Целевой атрибут: ТИПЫ ЭВМ			
№²	Номер	Атрибут	Значимость, %
1	3	4. Двухпроцессорная архитектура	31,868
2	1	1. Многопроцессорная ЭВМ	25,373
3	2	3. Блочная конструкция	24,774
4	4	6. Синтез ПК и телефона	17,985
5	6	8. Настольный ПК	0,000
6	5	7. Мобильность	0,000

Рис. 9. Значимость атрибутов классификации типов ЭВМ (укороченный список атрибутов)

При классификации видов ЭВМ (на основе второго узла ЭС) проведенный ранее анализ излишних переменных и удаление их из модели полностью подтверждаются опцией *Значимость атрибутов* (рис. 10–13).

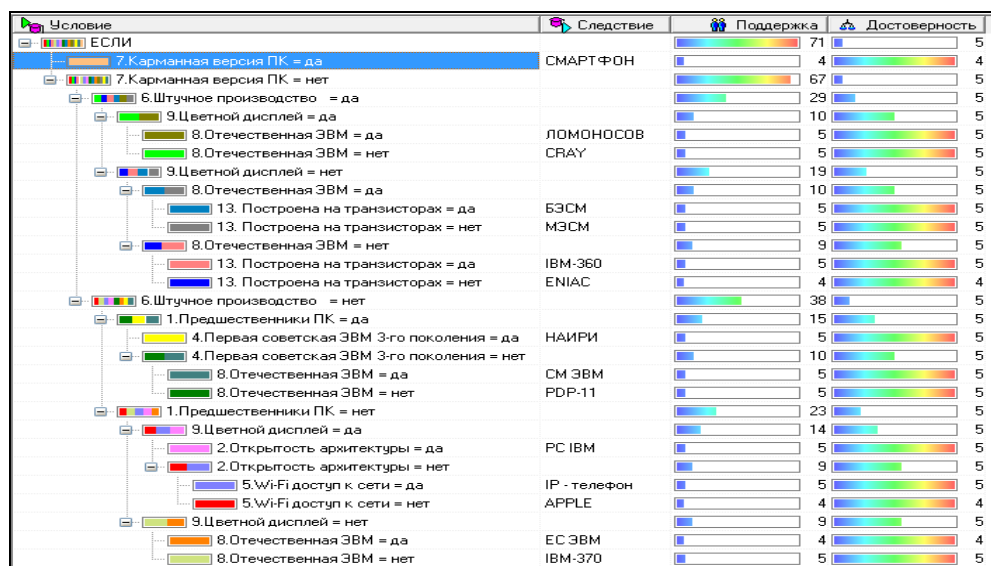


Рис. 10. Дерево решений классификации видов ЭВМ

Целевой атрибут: Виды ЭВМ				
№	Номер	Атрибут	Значимость, %	/
1	6	6. Штучное производство	23,878	
2	9	9. Цветной дисплей	17,752	
3	8	8. Отечественная ЭВМ	17,290	
4	1	1. Предшественники ПК	13,280	
5	7	7. Карманная версия ПК	8,018	
6	13	13. Построена на транзисторах	6,832	
7	4	4. Первая советская ЭВМ 3-го пок...	4,974	
8	2	2. Открытость архитектуры	4,754	
9	5	5. Wi-Fi доступ к сети	3,221	
10	12	12. Наличие оптического диска	0,000	
11	11	11. Наличие сетевой ОС	0,000	
12	3	3. Пульт управления	0,000	
13	10	10. Наличие ОС	0,000	

Рис. 11. Значимость атрибутов классификации видов ЭВМ

Целевой атрибут: Виды ЭВМ				
№	Номер	Атрибут	Значимость, %	/
1	7	8. Отечественная ЭВМ	19,037	
2	5	6. Штучное производство	16,295	
3	1	1. Предшественники ПК	13,791	
4	8	10. Наличие ОС	9,732	
5	3	4. Первая советская ЭВМ 3-го пок...	9,265	
6	6	7. Карманная версия ПК	9,055	
7	2	2. Открытость архитектуры	8,018	
8	10	13. Построена на транзисторах	6,832	
9	4	5. Wi-Fi доступ к сети	4,754	
10	9	11. Наличие сетевой ОС	3,221	

Рис. 12. Значимость атрибутов классификации видов ЭВМ (укороченный список атрибутов)



Рис. 13. Дерево решений классификации видов ЭВМ (укороченный список атрибутов)

Использование построенной информационной модели в виде нейронной сети осуществляется с помощью инструмента *Что-если*, который в интерактивном режиме дает возможность отметить набор характерных признаков объекта и получить распознаваемый вид ЭВМ (рис. 14).

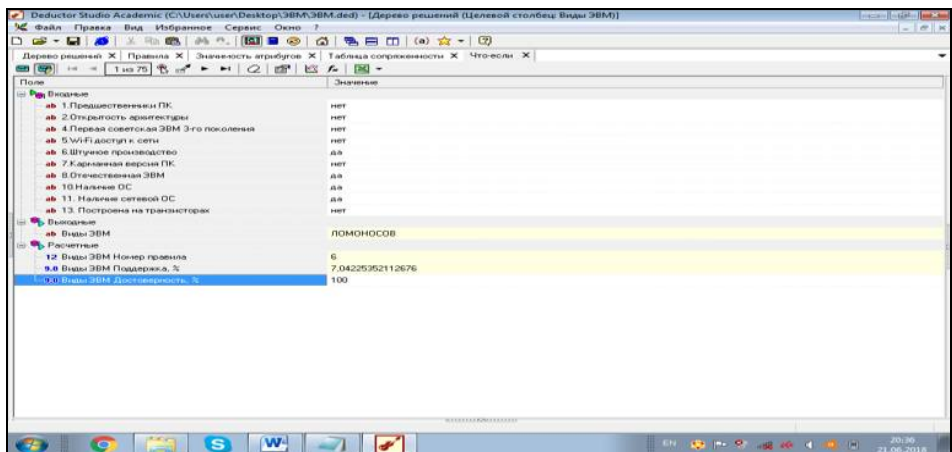


Рис. 14. Распознавание вида ЭВМ ЛОМОНОСОВ

Выводы

Способы решения задач идентификации объектов, рассмотренные с помощью экспертных систем и нейронных сетей, несмотря на их внешние различия, имеют и

много общего. Они предполагают составление списков входных (переменных) и выходных (исходов) данных. Оба программных средства создают базу знаний (систему правил), в основе которых лежит понятие «весовые коэффициенты». Они формируются на этапе обучения, которое осуществляется по-разному: обучение нейронной сети производится нейроэмулятором Deductor Academic после ввода в него обучающей выборки в виде исходной таблицы; экспертных систем – в формате работы трех модулей (ввод примера, тренировка системы и обучение системы).

Имеются различия и в самой архитектуре ЭС и НС. Действительно, экспертная система может быть многоузловой, что обуславливается многоуровневой классификацией объектов. В рассмотренном нами примере в первом узле ЭС идентифицируемая компьютерная программа соотносится по своим признакам с системным, прикладным ПО или с системой программирования, во втором узле определяется название программы по его переменным. Нейросеть не имеет узлов, в ней задается только один список атрибутов, позволяющих сразу определить название узнаваемой программы. Этот список атрибутов может совпадать со списком переменных второго узла ЭС. Таким образом, нейросеть можно формально считать одноузловой экспертной системой.

В данной работе показана технология построения информационных моделей при решении задач распознавания объектов, что позволяет систематизировать знания в различных предметных областях. Программные среды «Оболочка ЭС» и Deductor Academic лучше использовать совместно с целью оптимизации числа атрибутов (характерных признаков) распознаваемых объектов. Действительно, работая в среде ЭС довольно трудно определить значимые и незначимые атрибуты (переменные). С помощью инструмента *Значимость атрибутов* можно быстро определить лишние атрибуты (свойства объектов). При этом ЭС и нейросети ведут себя по-разному. Так, нейросеть может идентифицировать объект по отсутствию у него указанных в базе знаний свойств. Это объясняется спецификой работы мастера НС, использующего управляющую структуру «Если-то-иначе». Для нее отсутствие у объекта какого-либо свойства есть его характеристика. А для ЭС идентификация объекта обязательно требует наличия у него хотя бы одного характерного признака. Такова особенность работы алгоритма, заложенного в оболочку ЭС.

Список признаков, входящих в соответствующее объекту правило, включает в себя не все свойства объектов, которые указаны в исходной таблице для построения нейронной сети. Сюда входят только те свойства, которые позволяют сети отличить один объект от другого, с учетом того, что эти объекты могут иметь одинаковые характерные признаки.

Литература

1. Касторнов А. Ф. Нейрокомпьютинг – современный интеллектуальный инструмент познания // Ученые записки Института информатизации образования РАО. 2017. № 1–2 (61). С. 58–61.
2. Касторнова В. А., Андреев А. Е., Касторнов А. Ф., Яламов Г. Ю. Применение и перспективы развития технологии модульного проектирования экспертных систем контроля знаний в сфере образования // Управление образованием: теория и практика. 2016. № 1 (21). С. 52–63.

3. Молчанов А. А. Использование экспертных систем в системе открытого образования // Гаудеамус. 2014. № 2 (24). С. 57–68.
4. Нейлор К. Как построить свою экспертную систему. М.: Энергоатомиздат, 1991. 286 с.
5. Рыбина Г. В. Интеллектуальные обучающие системы на основе интегрированных экспертных систем: опыт разработки и использования // Информационно-измерительные и управляющие системы. 2011. №10. С. 4–16.

References

1. Kastornov A. F. Nejrokomp'yuting – sovremennyy intellektual'nyj instrument poznaniya [Neurocomputing – modern intellectual tool for learning]. *Uchenye zapiski institute informatizatsii obrazovaniya RAO* [Scientific notes of Institute of education informatization RAO], 2017, no. 1–2 (61), pp. 58–61.
2. Kastornova V. A., Andreev A. E., Kastornov A. F., Yalamov G. YU. Primenenie i perspektivy razvitiya tekhnologii modul'nogo proektirovaniya ehkspertnyh sistem kontrolya znanij v sfere obrazovaniya. [Application and the prospects of development of technology of modular design of expert control systems of knowledge in education]. *Upravlenie obrazovaniem: teoriya i praktika* [Management of education: theory and practice], 2016, no. 1 (21), pp. 52–63.
3. Molchanov A. A. Ispol'zovanie ehkspertnyh sistem v sisteme otkrytogo obrazovaniya [Use of expert systems in the system of open education]. *Gaudeamus* [Gaudeamus], 2014, no. 2 (24), pp. 57–68.
4. Nejlor K. *Kak postroit' svoyu ehkspertnyuyu sistemu* [How to construct the expert system]. Moscow: Energoatomizdat, 1991. 286 p.
5. Rybina G. V. Intellektual'nye obuchayushchie sistemy na osnove integrirovannyh ehkspertnyh sistem: opyt razrabotki i ispol'zovaniya [The intellectual training systems on the basis of the integrated expert systems: experience of development and use]. *Informacionno-izmeritel'nye i upravlyayushchie sistemy* [The information and measuring and operating systems], 2011, no. 10, pp. 4–16.

Для цитирования: Касторнов А. Ф., Касторнова В. А. Информационное моделирование на основе интеллектуальных систем как инструмент систематизации знаний в предметной области «Информатика» // Вестник Череповецкого государственного университета. 2019. № 1 (88). С. 175–187. DOI: 10.23859/1994-0637-2019-1-88-17

For citation: Kastornov A. F., Kastornova V. A. Information modeling on the basis of intellectual systems as the tool of systematization of knowledge in the subject domain of “computer science”. *Bulletin of the Cherepovets State University*, 2019, no. 1 (88), pp. 175–187. DOI: 10.23859/1994-0637-2019-1-88-17