

Тбилисский государственный университет имени Ив. Джавахишвили
Факультет прикладной математики и компьютерных наук
Кафедра математического обеспечения компьютеров и информационных технологий

Ирина Хуцишвили

**Компьютерная система принятия решений, основанная
на статистическом и нечетком анализе**

05.13.11 – Математическое и программное обеспечение вычислительных машин, систем,
комплексов и сетей

ДИССЕРТАЦИЯ

на соискание ученой степени кандидата
физико-математических наук

Научный руководитель: доктор физико-математических наук,
профессор **ТАМАЗ ГАЧЕЧИЛАДЗЕ**

2006

Содержание

Введение.

Глава I. Обзор компьютерных систем принятия решений.

§1.1. Понятие системы поддержки принятия решений.

§1.2. Сосредоточенные и распределенные системы поддержки принятия решений.

§1.3. Понятие экспертной системы.

§1.4. Инженерия знаний.

§1.5. Архитектура ЭС: основные компоненты.

§1.6. Представление знаний в экспертных системах.

§1.7. Представление и обработка нечетких данных в ЭС.

§1.8. Программное обеспечение ЭС.

Глава II. Нечеткие статистические методы принятия решения.

§2.1. Нечеткое множество.

§2.2. Функции принадлежности нечеткому множеству.

§2.3. Статистический метод анализа нечетких классов.

§2.4. Дискриминационный анализ.

§2.5. Метод рассуждений на основе прецедентов.

§2.6. Метод экспертонов.

Глава III. Описание нечеткой компьютерной системы принятия решений.

§3.1. Модель нечеткой компьютерной системы и ее основные элементы.

§3.2. Нечеткие статистические методы для конкретной задачи прогноза.

§3.3. Общее описание программного продукта.

§3.4. Описание основных каталогов.

§3.5. Библиотечный Каталог Lib.

§3.6. Каталог Fuzzy_Functions.

§3.7. Описание каталога Admin.

§3.8. Описание каталога Main.

Вывод и основные результаты.9

Литература.

Приложение.

Введение

Проблемы принятия решений являются одними из самых важных проблем человеческой деятельности. Средством, помогающим человеку в сложных задачах выбора являются компьютерные системы принятия решений. Если вывод решения компьютерной системы включает элемент участия эксперта – специалиста в исследуемой области (или группы экспертов), то такая компьютерная система представляет собой экспертную систему (ЭС). Экспертные системы, использующие эвристические знания, применяются в тех случаях, когда сформулировать решение задачи в традиционных математических терминах не удается. Такие задачи получили название плохо формализованных задач, и они обладают одной или несколькими из следующих особенностей[76]:

- задачи не могут быть заданы в числовой форме;
- цели не могут быть выражены в терминах точно определенной целевой функции;
- не существует алгоритмического решения задачи.

В связи с использованием эвристических знаний возникла необходимость:

- в компьютерной обработке данных, по своей природе являющихся нечеткими;
- в исследовании объектов настолько сложных, что их описание невозможно без введения нечетких представлений об их природе.

Это обусловило распространение компьютерных систем принятия решений, основанных на применении нечеткой логики, в частности, стало целесообразным применение нечетких статистических методов обработки данных, т. к. в подобных случаях применение методов классической статистики не дает достоверных результатов.

Это, в свою очередь, вызывает необходимость разработки методов представления в компьютере нечеткой информации, а также разработки программного обеспечения: эффективных и быстро реализуемых алгоритмов обработки нечеткой информации нечеткими статистическими методами принятия решения.

В настоящее время в мире создание компьютерных систем, основанных на применении «базы знаний», или экспертных систем, переживает настоящий бум. Такие системы создаются и успешно функционируют в различных областях науки: в медицине (диагностика) ([7], [29], [37], [40], [41], [45], [48], [54], [58]), в экономике, финансах и бизнесе ([8], [16], [23], [26], [38], [39], [53], [55], [66], [70], [74], [81]), в образовании, политике и социологии ([52], [72], [73], [79]) и др. ([67]). Значительно меньше компьютерные системы применяются в геологии, геофизике, метеорологии, однако, есть и такие. В основном это – системы-советчики, системы-помощники в деле составления различного рода прогнозов([68], [71]).

Почти во всех перечисленных выше экспертных системах основной компонент их архитектуры – база знаний – представляет собой основанную на правилах базу знаний ([37],[47],[49]). Однако существуют и компьютерные системы принятия решений, имеющие таблично-цифровую базу знаний ([13], [14]). Экспертные системы различают не только по структуре базы знаний, но и по способу ее проектирования, получения из нее информации и дальнейшей обработки этой информации.

Для целей создания и дальнейшего применения базы знаний применяется как логическое программирование (PROLOG, LIPS, Mercury и др.), основанное на исчислении предикатов ([5], [36]) так и объектно-ориентированное программирование (Object Pascal, C++, C#) ([4], [17]).

Особо следует выделить класс компьютерных систем, применяющих для принятия решений теорию нечетких множеств, нечеткую логику. Разработано и реализовано множество нечетких методов как для создания нечеткой базы знаний и получения из нее информации, так и для обработки этой нечеткой информации ([1], [21], [28], [29], [30], [33], [46], [51], [61]).

В последние десятилетия в мире получили большое распространение компьютерные системы поддержки принятия решений (СППР), как средство для доставки квалифицированных узкоспециализированных знаний в то место, где они необходимы для принятия решений. С широким распространением Internet появилась возможность централизованного хранения этих знаний (экспертных систем) и предоставления доступа к ним через каналы связи. Различают два вида СППР: сосредоточенные (одна локализованная экспертная система) и распределенные

(множество распределенных функционально и/или пространственно экспертных систем) ([18], [20], [24]).

Для обеспечения функционирования сосредоточенных компьютерных систем поддержки принятия решений, как правило, применяется Web-программирование (технологии PHP, ASP, CGI, языки программирования JavaScript, Perl) и, как минимум, технология клиент-сервер. Работа же распределенных СППР предполагает наличие многопользовательского интерфейса, хорошо продуманной организации компьютерного взаимодействия (т.н. агентно-ориентированные системы) и соответствующее программное обеспечение ([42],[60]). Что касается программного обеспечения, точнее, языков программирования для подобных систем, то средств процедурных языков параллельного программирования оказалось недостаточно. На смену им пришел объектно-ориентированный, переносимый и многопоточный язык программирования Java ([63]).

Применение компьютерных систем принятия решений последнего типа особенно важно в тех случаях, когда получение консультации у эксперта затруднено, например, из-за его территориальной удаленности. Или необходимо коллективное решение группы экспертов, собрать вместе которых – дорогостоящий проект. В этих случаях компьютерная система может сыграть роль первичного консультанта. Поэтому разработка и внедрение компьютерных систем принятия решений в нашей стране является актуальной проблемой.

Цель работы

В рамках проекта Intas-9702126 «A new Approach to Analysing Fuzzy Data and Decision-making Regarding the Possibility of Earthquake Occurrence» [12] для задач принятия решений на основе оценок параметров землетрясения была проделана большая теоретическая работа по анализу таких методов нечеткой статистики, как статистика нечетких классов, дискриминационный анализ, теория экспертонов, анализ связности и др. В результате была подготовлена теоретическая база для практического создания прогнозирующей компьютерной системы.

Целью настоящей работы является теоретический анализ методов нечеткой статистики и построение моделей следующих систем, способствующих принятию

решения для конкретного случая: статистического метода анализа нечетких классов, дискриминационного анализа, метода Case Based Reasoning: дословно – рассуждения, основанные на прецедентах.

Кроме этого, анализ и построение математической модели системы для окончательного принятия решения, т.е. для выбора наиболее возможного (оптимального) решения из предложенных вышеперечисленными методами: метода экспертонов.

После теоретического построения математических моделей систем принятия решения необходима их практическая реализация. С этой целью нужно разработать и реализовать алгоритмы представления в компьютере нечеткой информации, алгоритмы создания и использования таблично-цифровой базы знаний, а также алгоритмы всех используемых для принятия решения нечетких статистических методов.

Для множества активностей при нечетком прогнозируемом объекте с учетом исследованных нечетких методов создана модель компьютерной системы принятия решений, основанная на таблично-цифровой базе знаний.

Эта модель реализована с использованием средств Web-программирования (технология PHP, языки программирования JavaScript, Perl) и технологии клиент-сервер. В качестве базы данных выбрана СУБД MySQL.

Новизна и основные результаты

- В работе обосновано применение средств Web-программирования для создания компьютерной системы принятия решения в том случае, когда как данные, так и знания размещены централизованно.
- Исследовано создание таблично-цифровой базы знаний и ее применение для 3-х нечетких методов – статистического анализа нечетких классов, дискриминационного анализа, метода CBR. Рассмотрены как положительные, так и отрицательные стороны этих методов и, с целью выбора наиболее возможного решения, исследован 4-тый нечеткий метод, основанный на теории экспертонов и предполагающий учет предпочтений экспертов при окончательном принятии решения (поддержка принятия решения).

- Для метода статистики нечетких классов исследованы различные модели функций совместимости, как то модель Л. Заде, кусочно-линейная модель, модель Цисно и Циммермана. Применяемая в работе функция совместимости является новой модификацией модели Заде[61] и опробованной автором работы.

- Нечеткие методы дискриминационного анализа и метода рассуждений, основанного на прецедентах адаптированы к конкретному виду задачи принятия решения. Предложена их определенная модификация.

- Предложена новая модель компьютерной системы принятия решения (практически, модель сосредоточенной компьютерной системы поддержки принятия решений), в которой роль инженера знаний сведена до минимума, т.к. база знаний создается самой системой. Информация, необходимая для этого, выбирается из базы данных общего назначения (в нашем случае MySQL), в которой представлены известные на данный момент случаи правильно принятых решений вместе со значениями определяющих их активностей. Информация обрабатывается по определенным экспертами правилам.

- Разработаны алгоритмы создания базы знаний, получения из нее информации, представления нечетких понятий и реализации нечетких методов. Эти алгоритмы могут также успешно применяться специалистами нечеткой логики, т.к. в них реализованы механизмы вычисления некоторых нечетких характеристик (понятий).

- Разработанные и описанные классы, каталоги и модули для обработки нечеткой информации, модули для создания базы знаний и доступа к ней (выборки из нее необходимой информации при принятии решения), модули реализации статистических нечетких методов принятия решения и другие представляют программное обеспечение, на основе которого становится возможным практическое создание компьютерной системы принятия решений, основанной на статистическом и нечетком анализе. Необходимо отметить, что предложенная компьютерная система носит обобщенный характер, т.к. созданные каталоги и модули возможно использовать в различных областях исследования (например, прогноз погоды, медицинская диагностика), а саму систему в результате определенной модификации - для других задач прогноза.

Практическое значение

Работа обладает и практическим значением, т.к. предложенное программное обеспечение позволяет реализовать модель компьютерной системы принятия решений. На его основе создан пакет программ, представляющий собой прогнозирующую компьютерную систему для принятия решения о возможности землетрясения.

Система предполагает на основе известных случаев т.н. «шумов», «средних » и «сильных» землетрясений (статистика 1967-1992 гг., станция Душетского района, Грузия) и характеризующих их активностей (некоторые геофизические данные атмосферы), после построения образов нечетких классов объекта прогнозирования в каждый из нескольких дней подготовки землетрясения (за 2 дня, за 1 день и в день) и определения соответствующих функций совместимости прогнозируемых понятий для вновь поступающих данных получать прогноз о возможности землетрясения через несколько дней (через 2 дня, через день и в день соответственно).

В компьютерной системе предусмотрено обновление базы данных общего назначения при накоплении новой информации о правильно принятых решениях и, как следствие, обновление базы знаний. Работа системы протестирована для 20 случаев землетрясения и 60 случаев «шумов»: из 20 случаев землетрясения система правильно составила прогноз в 15 случаях, при этом в 11 случаях был точно установлен интервал интенсивности (лучше прогнозируются т.н. «сильные » землетрясения); из 60 случаев «шумов» система приняла правильное решение в 38-ти. Результаты можно считать удовлетворительными, тем более если учесть, что рассматриваемые активности (напряженность электрического поля атмосферы, температура воздуха, температура почвы, влажность, облачность, сила ветра, атмосферное давление) являются не самыми главными предвестниками землетрясения.

Система расположена на сервере (Apache) локальной/глобальной сети, ядро оболочки реализовано на языке PHP, применяется скриптовый язык JavaScript, используется технология клиент/сервер. База данных компьютерной системы реализована в СУБД (система управления базами данных) MySQL.

Апробация работы

Результаты диссертации опубликованы в 4 научных статьях и доложены на различных научных собраниях и семинарах: на нескольких семинарах кафедры математического обеспечения компьютеров и информационных технологий факультета

прикладной математики и компьютерных наук ТГУ (2004, 2005, 2006 гг.), на семинаре кафедры теории случайных процессов факультета прикладной математики и компьютерных наук ТГУ(2005 г.), на семинаре ИПМ им. И.Векуа (2004 г.).

Объем и структура работы

Диссертация состоит из введения, трех глав и списка цитируемой литературы (82 наименования). Работа содержит 130 страниц машинописного текста. Отдельно дается приложение, которое содержит информацию о первичных данных: численных значениях активностей объекта исследования.

Краткое содержание диссертации по главам

Введение содержит обзор литературы, связанной с тематикой работы, в нем сформулированы цель работы и приведено краткое содержание диссертации по главам.

Первая глава диссертации - «Обзор компьютерных систем принятия решений» посвящена описанию и классификации существующих компьютерных систем принятия решений как самых современных и мощных (компьютерные системы поддержки принятия решений), так и экспертных систем; рассмотрению их назначения, т.е. определению задач, решаемых такими системами и функций, которые они должны выполнять.

В первом параграфе приведена классификация тех задач, решение которых требует применения компьютерных систем, определены факторы человеко-машинного процесса принятия решения.

Во втором параграфе приведена классификация компьютерных систем поддержки принятия решений по типу их распространенности. Сформулированы те объективные причины, которые обусловили широкое распространение этих систем в современном мире.

Во третьем параграфе рассматривается экспертная система, как основной компонент компьютерной системы поддержки принятия решений. Очерчены области применения экспертных систем, сформулированы критерии их применения.

В четвертом параграфе поставлена проблема приобретения знаний и их аккумуляции в экспертной системе. Определено главное назначение инженерии знаний

(knowledge engineering), описаны этапы создания базы знаний: инженер знаний взаимодействует с экспертом и заполняет базу знаний. Он также выбирает оболочку экспертной системы (оболочка – базовый элемент экспертной системы, определяющий интерпретацию команд и действий пользователя). Созданием же оболочки и механизма вывода решений занимается программист.

В пятом параграфе дается описание архитектуры экспертной системы, ее основных элементов и определяется роль и место каждого из них в оболочке системы. Приводится классификация экспертных систем по типу базы знаний: основанные на правилах ЭС и имеющие таблично-цифровую базу знаний.

В шестом параграфе описаны формы представления знаний в экспертных системах и те средства, которые обеспечивают эти представления. Различают несколько основных форм представления знаний. Для представления математических знаний используют языки формальной логики. Формальная продукционная система представляет знание эксперта в виде предложений типа «Если A , То B », что можно записать в виде логического выражения $A \rightarrow B$. Это представление используется как для создания базы знаний, основанной на правилах, так и для создания таблично-цифровой базы знаний. Для представления знаний применяются также фреймы и сетевые модели. Рассмотрение представлений в общем контексте делает базу знаний экспертной системы более гибкой и мощной.

В седьмом параграфе рассматриваются вопросы представления нечетких данных и их обработки. Исследуются объективные причины недостаточной определенности знаний и определяются подходы для разрешения этой проблемы: вероятностный и основанный на нечеткой логике. Рассмотрена нечеткая экспертная система и этапы, из которых состоит процесс принятия решений в нечеткой системе:

1. Фазификация – на этапе фазификации происходит вычисление совместимых значений исходных данных на основе применения функций совместимости нечетких понятий. Если исходные данные изначально задаются совместимыми значениями (от 0 до 1), необходимость в этапе фазификации отпадает.

2. Вывод – для каждого выводимого понятия (решения) вычисляется возможные доверительные значения.

3. Композиция – в случае применения более чем одного метода вывода

решения необходима их композиция, чтобы было получено единственное нечеткое подмножество выводимых значений.

4. Дефазафикация – используется для того, чтобы из нечеткого подмножества выводимых значений выбрать единственное, которое и определит окончательное принятое решение.

В восьмом параграфе проведен обзор тех средств программирования, которые используются при проектировании экспертных систем. Самыми актуальными средствами программирования экспертных систем являются логическое программирование и объектно-ориентированное программирование. В последнее время внедрение компьютерных систем типа СППР потребовало применение новых средств программирования. Программным инструментарием локализованной экспертной системы могут быть, например, технологии PHP, ASP, Flash (для визуализации окончательного решения), язык программирования Perl и, как минимум технология клиент/сервер. Программным инструментарием же распределенной компьютерной системы поддержки принятия решений является технология ASP.NET, которая дает возможность применения объектно-ориентированных языков C#, C++, а также объектно-ориентированный и многопоточный язык программирования Java.

Вторая глава диссертации – «Нечеткие статистические методы принятия решения» - посвящены рассмотрению 4 нечетких методов принятия решения. Применение этих методов возможно в различных задачах принятия решения (прогноз погоды, экономическое планирование, медицинская диагностика). Предполагается, что на основе работы этих методов определенное решение принимается на основании существующих правильно принятых решений и проявленных при них активностей.

В первом параграфе дается определение нечеткого множества и приводятся примеры нечетких множеств.

Во втором параграфе рассматриваются различные модели функций совместимости нечетких множеств, приводятся свойства функций совместимости.

Во третьем параграфе описан статистический метод анализа нечетких классов, во время применения которого принятие решения характеризуется следующим подходом: по значениям активностей, характерным для известных прогнозов, после построения

образов нечетких классов классификации прогнозируемого объекта и функций совместимости прогнозируемых понятий оценивается «близость» исследуемого случая одному из образов.

В методе определяются нечеткие выборочные частоты, нечеткие относительные частоты и вычисляются нечеткие веса каждого интервала прогнозирующего фактора по следующим формулам:

$$\tilde{n}_{kj}^m = \sum_i \mu_i^m \cdot n_{kj}^i, \quad \tilde{f}_{kj}^m = \frac{\tilde{n}_{kj}^m}{\sum_i \tilde{n}_{kj}^i}, \quad w_{kj} = \frac{\sum_i \tilde{n}_{kj}^i}{\sum_j \sum_i \tilde{n}_{kj}^i},$$

где n_{kj}^i обозначает выборочные частоты j -того класса X_k -той активности соответствующего i -того прогнозируемого класса, μ_i^m – среднее значение функции совместимости, когда прогнозируемая величина из i -того прогнозируемого интервала принадлежит m -тому прогнозируемому классу.

После этого для определенной выборки активностей прогнозируемой величины (прогнозирующих факторов) уже можно сделать прогноз: необходимо только определить нечеткие веса каждого прогнозирующего фактора в соответствии с его интервалом и осуществить многофакторный линейный синтез нечетких весов и нечетких относительных частот, в результате чего получается обобщенное решения (взвешенный вектор возможных решений) $\vec{D}_\alpha = \vec{w}_\alpha \cdot \vec{f}_\alpha$, которому соответствует мера

возможности $\overrightarrow{Poss}_\alpha = \frac{\vec{D}_\alpha}{\max_j(D_\alpha(j))}$, где $D_\alpha(j)$ – j -тая компонента вектора \vec{D}_α .

Для получения единственного, классического решения вводится дополнительный принцип, например, принцип максимума возможностей, исходя из которого $D_{Class}^{(\alpha)} = \max_i(Poss_\alpha(i))$, где $Poss_\alpha(i)$ – это i -тая компонента вектора $\overrightarrow{Poss}_\alpha$.

В четвертом параграфе описан метод дискриминационного анализа, который характеризуется следующим подходом: активности оцениваются по поводу того, насколько представительными они являются для конкретного прогноза по сравнению с другим.

Вычисляются значения положительной и отрицательной дискриминаций по следующим формулам:

$$p_{ij} = \frac{1}{C_D - 1} \sum_{\substack{k \in D \\ k \neq j}} \chi_{Large-ratio} \left(\frac{f_{ij}}{f_{ik}} \right), \quad n_{ij} = \frac{1}{C_D - 1} \sum_{\substack{k \in D \\ k \neq j}} \chi_{Large-ratio} \left(\frac{f_{ik}}{f_{ij}} \right),$$

где $p_{ij}, n_{ij} \in [0;1]$, i – активность, j – решение, f_{ij} – относительная частота i -той активности, которая проявилась при j -том решении, C_D – мощность множества решений: $C_D = CardD = Card\{D_1, \dots, D_n\} = n$, *Large-ratio* означает нечеткое подмножество множества R^+ с функцией принадлежности: $\chi_{Large-ratio} : R^+ \rightarrow [0;1]$, которая сопоставляет отношения (действительные числа) интервалу $[0,1]$.

Теперь, если новые рассматриваемые данные описываются множеством активностей (A'_1, \dots, A'_r) , из таблиц $\{p_{ij}\}$ и $\{n_{ij}\}$ выбираются только те строки, которые соответствуют $\{A'_j\}$ и составляются новые $\{p'_{ij}\}$ и $\{n'_{ij}\}$ таблицы.

Обобщенное решение представлено нечетким подмножеством множества возможных решений со следующей функцией принадлежности:

$$\delta(D_j) = \frac{1}{2} (\chi_{Large}(\pi_j) + \chi_{Small}(v_j)), \quad j \in D, \text{ где}$$

$$\pi_j = \frac{1}{C_{A'}} \sum_{i=1}^r p'_{ij}, \quad v_j = \frac{1}{C_{A'}} \sum_{i=1}^r n'_{ij}, \quad C_{A'} = CardA' = Card\{A'_1, \dots, A'_r\} = r.$$

π_j и v_j представляют собой, средние значения (средние меры) положительной и отрицательной дискриминаций для j -того решения соответственно. χ_{Large} – функция принадлежности (совместимости) нечеткого подмножества *Large* (L), χ_{Small} – функция принадлежности нечеткого подмножества *Small* (S). Функция χ_L – монотонно возрастающая, а χ_S – монотонно убывающая функция: $\chi_L, \chi_S : [0;1] \rightarrow [0;1]$.

Для получения окончательного «классического» решения можно выбрать принцип максимальной возможности: решение принимается на основании максимального значения функции принадлежности:

$$\delta^{Class} = \max_{j \in D} \delta(D_j), \text{ где } \delta(D_j) - j\text{-тое решение.}$$

В качестве этого окончательного решения получается четкое подмножество, ближайшее к нечеткому $\delta(D_j)$, другими словами, решение δ^{Class} с максимальным значением в $\delta(D_j)$ считается самым достоверным решением.

В пятом параграфе описан т.н. метод рассуждений, основанный на прецедентах (Case Based Reasoning), идея которого достаточно проста. Для того, чтобы было принято новое правильное решение или сделан новый прогноз, среди прошлых случаев ищутся аналогии с новой ситуацией и принимается такое же решение, которое было верным для аналогов. Вычисляются меры сходства между новым и всеми существующими случаями:

1. мера сходства между двумя i -тыми активностями двух случаев вычисляется при помощи следующих равенств:

$$\text{если активности четкие величины} \quad DV_i = \min(CB_i, ND_i), \quad i = \overline{1, n}.$$

$$\text{в случае нечеткости активностей} \quad DV_i = 1 - |CB_i - ND_i|, \quad i = \overline{1, n}.$$

Здесь n – число всех активностей, DV_i – расстояние между двумя i -тыми активностями, CB_i – значение i -той активности существующего случая, ND_i – значение i -той активности нового случая.

2. мера сходства между двумя случаями

Пусть SV_j представляет собой величину сходства между новым и j -тым из существующих случаев. Тогда ее можно вычислить при помощи следующей формулы

$$SV_j = \frac{1}{n} \sum_{i=1}^n w_i \cdot DV_i, \quad j = \overline{1, k}, \quad i = \overline{1, n},$$

где \vec{w} представляет собой вектор весов, каждый i -тый компонент которого показывает, каков уровень значимости i -той активности для принимаемого решения. $w_i \in [0, 1]$, при этом, $w_i = 0$ означает, что i -тая активность для принятия решения значения не имеет, $w_i = 1$, что i -тая активность для принятия решения очень значительна.

Компоненты вектора весов определяются на уровне экспертов. k – количество всех существующих случаев (прецедентов).

Для получения окончательного решения применяется принцип максимума, т.е. ищется такое число r , которое удовлетворяет условию

$$SV_r = \max(SV_j), \quad j = \overline{1, k}.$$

если мера сходства SV_r больше заранее установленного экспертом порога, тогда принимается решение, характерное для этого прецедента. Если же она меньше, результат неудовлетворительный и похожее решение неубедительно.

В шестом параграфе описан метод экспертонов. Экспертон – это обобщение понятия вероятности случайного нечеткого события, когда вероятность случайного события каждого α -уровня заменяется на доверительный интервал. Эти интервалы, в свою очередь, статистически определяются группой экспертов.

Пусть E – множество определенных объектов, конечное или бесконечное. Группе из r экспертов предлагается высказать свое субъективное мнение относительно каждого элемента из E в виде интервала доверия.

$$\forall P \in E : [a_*^j(P), a_*^*(P)] \subset [0,1], \text{ где } j \text{ – номер эксперта.}$$

Рассматривается статистика, когда каждому элементу $P \in E$ ставится в соответствие как нижняя, так и верхняя границы интервалов доверия. Кумулятивный закон распределения $F_*(\alpha, P)$ построен на основе $a_*^j(P)$, а $F^*(\alpha, P)$ – на основе $a_*^*(P)$.

Отсюда получается

$$\forall P \in E, \quad \forall \alpha \in [0,1]: \quad \tilde{A}(P) = [F_*(\alpha, P), F^*(\alpha, P)], \text{ где } \tilde{A} \text{ означает экспертон.}$$

В теории экспертонов доказывается, что экспертон удовлетворяет тем же алгебраическим свойствам, что и вероятность, если соблюдены специальные правила монотонности. Монотонность здесь подразумевается в смысле интервальной монотонности. Проводятся следующие преобразования экспертона \tilde{A} :

- экспертон приводится к вероятностному множеству взятием среднего арифметического границ интервала;
- вероятностное множество приводится к нечеткому множеству с помощью математического ожидания.

- при необходимости находится ближайшее к нечеткому четкое множество.

Если множество E содержит i элементов, в результате приведенных выше преобразований каждому P_i будет соответствовать определенное число, которое представляет собой значение доверия, установленное с учетом общего мнения экспертов. Для получения единственного решения применяется принцип максимума:

$$\delta(P_i) = \max(P_i).$$

В условиях, когда по поводу одного и того же прогноза было принято несколько решений, например, разными методами, для вывода единственного решения можно воспользоваться методом экспертов. Пусть нечеткое множество всех решений E' , полученное в результате применения метода экспертов, состоит из k элементов: $E' = \{\delta_1, \delta_2, \dots, \delta_k\}$. Тогда, исходя из метода экспертов, можно найти единственное самое достоверное решение:

$$\delta^{Class} = \max_j(\delta_j), j = \overline{1, k}, \text{ где } \delta_j - j\text{-тое возможное решение.}$$

Третья глава диссертации – «Описание нечеткой компьютерной системы принятия решений» – содержит описание модели компьютерной системы и ее основных частей. Здесь же даны математические модели нечетких методов с учетом их модификации для конкретной задачи прогноза (принятие решения о возможности землетрясения по данным о геофизических активностях атмосферы). Приведено описание основного класса, каталогов и модулей пакета программ.

В первом параграфе определено, какими свойствами должна обладать нечеткая компьютерная система, основанная на таблично-цифровой базе знаний. Приведено описание основных частей системы и дана схема ее модели. В компьютерной системе вывод решения обеспечивают нечеткие статистические методы, процесс принятия решения которых, а также выбор СУБД, предопределил создание базы знаний самой компьютерной системой.

Во втором параграфе рассмотрена модификация нечетких статистических методов с учетом конкретной задачи прогноза. Приведены определения применяемых функций совместимости, в случае метода статистики нечетких классов использована новая модель функции совместимости.

В третьем параграфе дано общее описание программного продукта, сформулированы факторы выбора таблично-цифровой базы данных, обоснован выбор базы данных MySQL: одна из главных причин выбора MySQL в качестве СУБД для основанной на таблично-цифровой базе знаний нечеткой компьютерной системы состоит в том, что в MySQL осуществлен механизм нечетких запросов (fuzzy queries, flexible queries). Это, в свою очередь, дает возможность не только представления нечеткой информации, но и задания интервалов выбора в виде нечетких множеств. Например, представление нечеткого понятия «землетрясение» возможно в виде таблицы

Key	Name	Magnitude	
		Init_value	Upper_value
1	слабое	0	< 3
2	среднее	>=3	<=5
3	сильное	>5	<=8

Ядро программного продукта реализовано на основе технологии PHP. Соединение с базой данных и осуществление запроса является для PHP простой задачей, которую можно выполнить в двух-трех строках программы. Процессор сценариев PHP хорошо оптимизирован для времени отклика, необходимого веб-приложениям. Еще одно положительное свойство PHP в том, что код выполняется на сервере что, в свою очередь обеспечивает защиту программного пакета от неавторизованного изменения. PHP может использоваться на всех крупных операционных системах, включая Linux, многие варианты Unix (HP-UX, Solaris и OpenBSD), Microsoft Windows, Mac OS X, RISC OS и, возможно, другие. PHP имеет поддержку для большинства существующих web-серверов. Это Apache, Microsoft Internet Information Server, Personal Web Server, Netscape и iPlanet-серверы, O'reilly Website Pro, Caudium, Xitami, OmniHTTPd и многие другие. Для большинства этих серверов PHP имеет стандартный пакет модулей. В других, поддерживающих стандарт CGI, PHP может работать как CGI-процессор.

Выходная информация (результат прогнозирования) представляется программным продуктом в виде сообщения – вывода наименования интервала интенсивности.

В четвертом параграфе дана иерархия программного продукта, описание каталогов. В силу особенностей реализации языка PHP его оболочка состоит из множества мелких модулей (файлов). В нашем случае модули объединены в 4 основных каталога:

- Каталог библиотеки Lib,
- Каталог сценариев по администрированию Admin,
- Каталог нечетких методов Fuzzy_Functions,
- Каталог клиентских сценариев Main.

Нужно отметить, что объединенные во всех трех каталогах Admin, Fuzzy_Functions и Main скрипты используют те модули библиотеки, которые представляют собой набор сценариев взаимодействия с MySQL: сценарии извлечения данных из базы, их обработки и помещения в базу. Кроме того, из каталога Lib применяется набор часто используемых т.н. генерализованных функций. Каталоги могут содержать подкаталоги, в которых объединены модули по их назначению. Модули, в свою очередь, представляют собой объединение функций.

В пятом параграфе описан библиотечный каталог Lib. Он состоит из двух основных базовых модулей

- модуль взаимодействия с системой управления базами данных Cl.php,
- модуль генерализованных функций GGen_Functions.php.

Модуль Cl.php обеспечивает взаимодействие с MySQL, в частности, открытие сокетов, установку параметров соединения, соединение с базой, извлечение данных и их помещение в базу, а также вывод сообщений об ошибках, которые произошли во время выполнения тех или иных операций с данными.

Здесь же описан класс Earthquake_MySql, в котором объединены одиннадцать функций установления диалога с MySQL.

```
class Earthquake_MySql{  
    var $con; var $result; var $record = array();  
    function f_OpenConnection($host, $user, $pass, $db) { ... }
```

```

function f_GetSqlError() { ... }

function f_CloseConnection() { ... }

function f_ExecuteSql($sql = "") { ... }

function f_GetSelectedRows($query_id = 0) { ... }

function f_GetAffectedRows(){ ... }

function f_GetRecord($query_id = 0, $param=0) { ... }

function f_FreeResult($query_id = 0) { ... }

function f_GetTable_List($query_id = 0) { ... }

function f_SetRecordPointer($recordnumber, $query_id = 0) { ... }

function f_GetNextId(){ ... }

}

```

Функция `f_OpenConnection` получает в качестве фактических параметров адрес хоста(сервера), на котором запущена СУБД, имя пользователя, пароль пользователя и собственно название базы с которой следует установить соединение. Посредством SQL-запроса, функция передает данные параметры MySQL и устанавливает соединение. В случае ошибки при соединении, функция возвращает значение `false`.

Функция `f_GetSqlError` возвращает код и описание ошибки, произошедшей при попытке соединения с MySQL, что существенно помогает элиминировать все возможные трудности во время работы программного пакета.

Функция `f_CloseConnection`, как следует из ее названия, просто закрывает соединение и высвобождает ресурсы СУБД.

Функция `f_ExecuteSql` – наиболее важная функция описанного класса `Earthquake_MySql`, которая выполняет конкретные запросы.

Остальные функции класса – утилиты, которые распределяют память, выводят и запоминают данные в результирующие переменные или массивы и высвобождают память, временно выделенную на хранение данных и т.д..

В модуле `GGen_Functions.php` описаны все те функции-утилиты, которые неоднократно используются клиентскими сценариями и сценариями

администрирования базы знаний. Все функции, которые учитывают работу с СУБД, в свою очередь, нуждаются в обращении к классу Cl.php, поэтому вызов класса происходит сразу после активации данного модуля, что обеспечивается первой строкой модуля Gen_Functions.php – require_once("Cl.php").

В шестом параграфе описывается каталог Fuzzy_Functions, который представляет собой библиотеку нечетких методов. Каталог состоит из 5 модулей. Первые 4 соответствуют нечетким методам:

- SFC.php – статистический метод анализа нечетких классов
- DSC.php – метод дискриминационного анализа
- CBR.php – метод рассуждений на основе прецедентов
- Experton.php – метод экспертов

В соответствующих модулях объединены функции и процедуры получения возможного решения. Это блок принятия решения.

В каталоге Fuzzy_Functions содержится также модуль Make_Values.php, работа которого обеспечивает создание базы знаний (объединение различных таблиц). Из таблицы начальных данных выбираются значения активностей и по определенным экспертом правилам происходит заполнение соответствующей таблицы базы знаний.

Формирование базы знаний, также как извлечение из нее информации производится по простым правилам продукционного типа «ЕСЛИ А и В и С и ... , ТО К». Каждый из первых трех описанных нечетких методов применяет правила этого типа соответственно на этапах фазификации и вывода.

В седьмом параграфе описан каталог сценариев по администрированию Admin, который состоит из двух подкаталогов

- подкаталог сценариев по администрированию базы Knowledge_BD_Administration
- подкаталог сценариев по администрированию доступа к данным User_Administration

Несмотря на то, что базу знаний создает сама компьютерная система, в оболочке системы предусмотрено участие инженера знаний в формировании и обновлении базы знаний.

Каталог Knowledge_BD_Administration. Пользователь с наивысшим уровнем доступа посредством данного сценария может изменять значения тех или иных полей базы знаний, преобразовывать, обновлять и/или дополнять ее. Каталог состоит из 2-х модулей:

1. knbd_manual_amendment.php – данный модуль состоит из процедур и функций, позволяющих инженеру знаний изменять заданные в базе знаний значения в любой момент времени по собственному усмотрению.

2. knbd_amendment.php – модуль, состоящий из сценариев, которые автоматически переводят администратора в режим обновления базы данных общего назначения новыми значениями. Разработанный программный продукт предусматривает накопление информации о правильно принятых решениях и сохранение значимых данных до момента, пока не будут получены данные обо всех нечетких подмножествах прогнозируемого понятия. В администраторском интерфейсе предусмотрено отражение информации о накоплении. Когда вся необходимая информация будет накоплена, администратор оповещается и ему предлагается обновить базу данных. Администратор, по своему усмотрению, либо соглашается и обновляет базу данных новыми значениями, либо удаляет накопленные данные целиком или частично.

Существование модуля knbd_manual_amendment.php было predeterminedено обобщенным характером предложенной компьютерной системы, в том смысле, что эта система может быть использована не только для задачи прогноза землетрясений. В случае же применения системы для другой задачи возникнет необходимость в изменении содержимого базы знаний.

Каталог User_Administration предназначен для добавления пользователей или изменения их привилегий. Данный сценарий вступает в диалог с таблицей Users в базе

данных и предоставляет возможность администратору добавлять новых пользователей, изменять уровень доступа, накладывать ограничение на использование программного продукта.

В восьмом параграфе рассмотрен каталог Main, который содержит пользовательские сценарии.

Например, на одной из веб-страниц пользовательского интерфейса, которая как раз и вызывается из каталога Main, предоставляется возможность ввода значений активностей для получения решения о новом конкретном случае. После ввода при нажатии на соответствующую клавишу происходит активизация модулей каталога Fuzzy_Functions и начинаются реальные вычисления. Результаты этой работы передаются модулю results.php, который также находится в каталоге Main. Модуль results.php ответственен за вывод результата. Возможна форма вывода результата прогноза названием интервала интенсивности прогнозируемого объекта (выводом названия нечеткой лингвистической переменной).

Приложение предоставляет первичные данные об известных случаях землетрясений вместе с характеризующими их активностями. Информация оформлена в виде таблиц.

Глава I.

Обзор компьютерных систем принятия решений

В данной главе рассматриваются компьютерные системы принятия решений: как самые мощные и современные - компьютерные системы поддержки принятия решений (СППР), так и экспертные системы. Дается определение СППР; определяется каким условиям должна удовлетворять задача, чтобы для ее решения целесообразно было создание такой системы; приводится их классификация в зависимости от рассматриваемой задачи. Рассматривается главная составляющая часть компьютерных систем поддержки принятия решений – экспертная система (ЭС); приводится классификация ЭС в зависимости как от решаемой задачи, так и от методов создания

базы знаний и методов принятия решений; перечисляются как положительные, так и отрицательные стороны экспертных систем, основанных на правилах; определяются основные этапы работы нечетких экспертных систем; проводится обзор ЭС, применяющих нечеткие методы для решения задач в различных сферах.

Также обосновывается и описывается тот программный инструментарий, который используется для представления нечетких понятий и методов в компьютере в различного рода экспертных системах.

§ 1.1. Понятие системы поддержки принятия решений.

Термин «система поддержки принятия решений» появился в 70-х годах 20-го столетия [19]. Например, дается следующее определение: «система поддержки принятия решений – это компьютерная система, позволяющая ЛПП¹⁾ сочетать собственные субъективные предпочтения с компьютерным анализом ситуации при выработке рекомендаций в процессе принятия решения» [50]. Как видно из определения, основная идея функционирования подобных систем – это сочетание субъективных предпочтений лица, принимающего решение и компьютерных методов. Процесс принятия решения с помощью СППР подразумевает взаимодействие человека и компьютера, который можно разделить на 2 фазы: 1). анализ и постановка задачи для компьютера, которые должен выполнить принимающий решение субъект; 2). поиск решения, который реализуется компьютером.

Системы поддержки принятия решений служат следующим задачам:

1. Помогают произвести оценку обстановки (ситуаций) или объекта, для которого принимается решение; осуществить выбор критериев и оценить их относительную важность;
2. Генерируют возможные решения (сценарии действий);
3. Проводят оценку решений и выбирают лучшие;
4. Осуществляют обмен информацией по ходу принимаемых решений и помогают согласовывать групповые решения;

¹⁾ лицо, принимающее решение

5. Моделируют принимаемые решения (в тех случаях, когда это возможно);
6. Осуществляют динамический компьютерный анализ возможных последствий принимаемых решений
7. Проводят сбор данных о результатах реализации принятых решений и их оценку.

Существует ряд факторов, характеризующих человеко-машинный процесс поддержки принятия решений с помощью СППР. К ним относятся:

- характер распределенности СППР. Различают групповой или индивидуальный процесс принятия решения;
- типы проблем, решаемых с помощью СППР. Они определяют возможность использования аналитических моделей, численных оценок или только качественных характеристик.

Системы поддержки принятия решений могут быть сосредоточенными и распределенными.

§ 1.2. Сосредоточенные и распределенные системы поддержки принятия решений.

Сосредоточенные СППР включают в себя одну *экспертную систему*, установленную на одном компьютере. Такие системы выполняют функции 1-3 и 5-7, перечисленные выше (или часть их), помогая одному ЛПР, или небольшой группе специалистов, оценивать обстановку и принимать решения.

Различают следующие *типы* сосредоточенных СППР:

1. *Решение принимает сама система в автоматическом режиме.* Она состоит из ЭВМ, системы автоматического, ручного или обоих видов ввода информации и средства представления решения. Например, система тушения пожара. В такой системе автоматические датчики передают в ЭВМ результаты измерений температуры, экспертная система определяет возможность возникновения пожара и, если она есть, место его возникновения, а затем передает сигнал тревоги, или включает средства тушения пожара, или осуществляет и то и другое. Это пример системы, принимающей решение, а не поддерживающей принятие решений.

2. *Решение принимает специалист, имеющий в своем распоряжении СППР.* Система может включать в себя экспертную систему (или несколько экспертных систем), моделирующие программы, средства оценки принятых решений. Примером подобной системы может служить СППР для управления подвижным объектом, когда командиру корабля предлагаются несколько вариантов решения, а он выбирает и реализует один из них.

3. На одном компьютере располагают несколько ЭС, которые оценивают ситуацию *по разным критериям.* Окончательное же решение принимает специалист, который должен согласовать и откорректировать предложенные решения.

Под *распределенными СППР* понимают распределенные пространственно, функционально или и пространственно, и функционально. Пространственно и функционально распределенные СППР состоят из локальных систем поддержки принятия решения, которые расположены в связанных узлах вычислительной сети. Каждый узел независимо решает свои частные задачи, но для решения общей проблемы ни один из них не обладает достаточной информацией и ресурсами. Для решения общей проблемы им необходимо объединить свои локальные возможности и согласовать принятые частные решения.

Функционально распределенные системы состоят из нескольких ЭС (или СППР), связанных между собой информационно, но расположенные на одной вычислительной машине, т.е. пространственно они сосредоточены.

Распределенные системы могут выполнять все перечисленные выше функции (1-7) или часть их.

В последнее время большое распространение получили *иерархические системы поддержки принятия решений* (ИСППР). Такие системы состоят из нескольких ЭС или СППР, расположенных в разных узлах вычислительной сети. Узлы такой системы с точки зрения принятия решения являются неравноправными. Например, система состоит из подсистем W_1, W_2, \dots, W_k первого уровня и одной подсистемы W_0 второго, более высокого уровня. Подсистема W_0 осуществляет координацию работы подсистем 1-го уровня, определяет стратегию действий или распределения ресурсов. Степень централизации системы может быть различной: от полного подчинения

подсистем 1-го уровня подсистеме 2-го уровня, до разработки согласованных решений взаимодействия. Описанная система – самый простой пример ИСППР, как правило, такая система является частью более сложных систем принятия решений.

Существуют объективные причины, которые обусловили широкое распространение распределенных систем [80]:

- Дальнейшее развитие технологии производства вычислительной техники позволяет объединить большое число мощных вычислительных машин в единую сеть, что делает возможным выполнение параллельных вычислений и обмен информацией.

- Многие области, в которых используются СППР по своей природе являются распределенными. Одни из них распределены функционально (например, многие системы диагностики в медицине), другие как пространственно, так и функционально (например, системы автоматизации проектирования и управления сложных технических объектов).

- Распределенные системы облегчают принятие согласованных решений и обмен информацией группами специалистов, совместно решающих одну задачу.

- Они дают возможность создания системы для решения сложных проблем из относительно простых и автономных подсистем (принцип модульного построения).

Подводя итоги, можно сделать следующие выводы: сосредоточенные СППР решают более простые задачи, чем распределенные системы. Они и сами более просты, т.к. в них отсутствует проблема обмена информацией. Тем не менее, такие системы часто проектируются как локализованные узлы имеющие выход в Internet, когда необходимая для принятия экспертной системой решения информация располагается на сервере (технология клиент-сервер).

§ 1.3. Понятие экспертной системы. Области применения, критерии использования

Системы компьютерного анализа данных могут основываться на двух подходах. Первый заключается в том, что в системе фиксируется опыт эксперта, который и используется для оценки создавшейся ситуации. На этом подходе основывается построение экспертных систем.

Второй подход базируется на анализе исторических данных, описывающих поведение изучаемого объекта, принятых в прошлом решениях, их результатах (например, анализе временных рядов стоимости валют и акций, статистики продаж различного рода товаров, результатов выборов и т.п.).

Наконец, существует третий подход – комбинация первых двух: результаты, полученные при анализе исторических данных, оцениваются на основе опыта эксперта.

Экспертная система - это набор компьютерных программ или программное обеспечение, которое моделирует рассуждения человека-эксперта в некоторой определенной области его компетенции. Знания о предметной области, необходимые для работы ЭС, определенным образом формализованы и представлены в памяти ЭВМ в виде базы знаний, которая может изменяться и дополняться в процессе развития системы. Экспертная система также обязательно включает в себя некоторую процедуру принятия решения на основе имеющейся в базе знаний информации.

Области применения систем, основанных на знаниях, могут быть сгруппированы в несколько основных классов по типу исследуемых задач:

1) *Диагностика* – сопоставление имеющимся конкретным данным определенного решения. Например, медицинская диагностика (системы используются для установления связи между нарушениями деятельности организма и их возможными причинами), диагностика неисправностей в механических и электрических устройствах (системы, основанные на знаниях, используются при ремонте механических и электрических машин, а также при устранении неисправностей и ошибок в аппаратном и программном обеспечении компьютеров).

2) *Прогнозирование* - предсказание возможных событий или результатов на основе данных о прошлом или текущем состоянии объекта. Например, прогноз погоды, прогноз землетрясения, различного рода экономические прогнозы.

3) *Планирование* – выявление величин параметров объекта планирования для достижения желаемой цели. Например, финансово-экономическое планирование, проектирования сложных объектов (космических станций, крупных предприятий).

4) *Интерпретация* - способностью делать определенные заключения на основе результатов наблюдений. Например, распознавание речи, распознавание образов (определение местоположения и типов судов по данным акустических систем слежения).

5) *Контроль и управление* – способность принимать решения на основе анализа данных. Например, управление воздушным движением, медицинский контроль, регулирование финансовой деятельности предприятия, выработка решений в критических ситуациях.

6) *Обучение* – накопление знаний на основе анализа поведения объекта. Например, деловые игры, компьютерные системы обучения, компьютерные игры.

Большинство ЭС включают знания, по содержанию которых их можно отнести одновременно к нескольким классам. Например, обучающая система может также обладать знаниями, позволяющими выполнять диагностику и планирование (например, она определяет способности обучаемого по основным направлениям курса, а затем с учетом полученных данных составляет учебный план). Управляющая система может применяться для целей контроля, диагностики, прогнозирования и планирования.

Из всего сказанного выше явствует, как велика популярность экспертных систем. Исторически первыми экспертными системами были Heuristic DENDRAL и Meta-DENDRAL, разработанные в середине 60-х годов XX века в Станфордском университете и предназначенные для порождения формул химических соединений на основе спектрального анализа. В последние десятилетия создаются экспертные системы для решения различных задач почти во всех областях деятельности человека. Многие из них функционируют весьма успешно, с большой отдачей. Однако немало и таких случаев, когда экспертная система, создание которой длительный и дорогостоящий процесс, не окупается. Анализ накопленного опыта позволил сформулировать ряд критериев, которым должна удовлетворять задача, чтобы для ее решения целесообразно было проектирование ЭС. Для решения поставленной задачи лучше применять экспертную систему, если:

- затраты на создание ЭС соответствуют ее полезности, т.е. необходимо сопоставить временные и денежные затраты с выигрышем от использования ЭС ;
- обращение к эксперту более длительный или дорогостоящий процесс, чем обращение к экспертной системе, применяющей опыт и знания соответствующего эксперта;
- не существует эффективных алгоритмических методов ее решения, но

существуют эвристические методы решения, т.е. по своему характеру задача относится к области диагностики, интерпретации или прогнозирования (не носят вычислительный характер);

- задача решается методом формальных рассуждений, а не процедурными методами, с помощью аналогии или интуитивно, когда эксперт обращается к своему здравому смыслу;

- знания статичны, т.е. не меняются во время принятия решения;

- пространство возможных решений относительно невелико, т.е. предметная область достаточно узка и специфична, и эксперт может принять решение за время от нескольких минут до нескольких часов;

- есть по крайней мере один эксперт, который способен явно сформулировать свои знания и объяснить свои методы применения этих знаний для решения задачи, т.е. в тех областях, где отсутствуют эксперты (например, в астрологии) или знания о методах решения задачи отсутствуют (невозможно построить базу знаний) применение ЭС оказывается невозможным;

- возможно приведение знаний, полученных от эксперта, к виду, обеспечивающему их эффективную машинную реализацию.

Системы, основанные на знаниях, оказываются неэффективными при необходимости проведения скрупулезного анализа, когда число «решений» зависит от тысяч различных возможностей и многих переменных, которые изменяются во времени. В таких случаях лучше использовать базы данных с интерфейсом на естественном языке.

§ 1.4. Инженерия знаний.

С 70-х годов ЭС стали ведущим направлением в области искусственного интеллекта. При их разработке нашли применение методы ИИ, разработанные ранее: методы представления знаний, логического вывода, эвристического поиска, распознавания предложений на естественном языке и др. Однако уже на начальных этапах выявились серьезные принципиальные трудности, препятствующие более широкому распространению ЭС и серьезно замедляющие и осложняющие их разработку. Они вполне естественны и вытекают из самих принципов разработки ЭС.

Основная трудность - проблема приобретения (усвоения) знаний. Эта проблема возникает при «передаче» знаний, которыми обладают эксперты-люди, ЭС. Оказалось, что большинство экспертов, успешно используя в повседневной деятельности свои обширные знания, испытывают большие затруднения при попытке сформулировать и представить в системном виде хотя бы основную их часть: иерархию используемых понятий, эвристики, алгоритмы, связи между ними. Кроме того, необходимы, с одной стороны, знания в области математической логики и методов представления знаний, с другой - знания возможностей ЭВМ, их программного обеспечения, в частности, языков и систем программирования.

Таким образом, для разработки ЭС необходимо участие в ней особого рода специалистов, обладающих указанной совокупностью знаний и выполняющих функции «посредников» между экспертами в предметной области и компьютерными (экспертными) системами. Они получили название инженеров знаний (knowledge engineers), а сам процесс разработки ЭС и других интеллектуальных программ, основанных на представлении и обработке знаний - инженерией знаний (knowledge engineering). Функции эксперта и инженера знаний редко совмещаются в одном лице. Чаще функции инженера знаний выполняет разработчик ЭС. Как показал опыт многих разработок, для первоначального приобретения знаний, в которых участвуют эксперты, инженеры знаний и разработчики ЭС, требуется активная работа всех трех категорий специалистов. Она может длиться от нескольких недель до нескольких месяцев. Кроме того, в этой работе должен участвовать и пользователь (заказчик) экспертной системы, т.к. создаваемая система прежде всего должна удовлетворять его требованиям.

В том случае, когда инженер знаний намеревается создать систему, основанную на правилах, он предоставляет эксперту примеры проблемы, получает от него объяснения по поводу решения им этой проблемы, формирует правила, а затем проверяет эти правила вместе с экспертом.

Тогда же, когда создается ЭС, основанная на таблично-цифровой базе знаний, эксперт предоставляет инженеру знаний документацию по поводу принятых им решений вместе с предпосылками, определившими это решение, а разъяснения делает только в том случае, если в документации что-либо неясно (в случае задач прогнозирования в качестве документации может выступать информация о правильно

поставленных прогнозах вместе с описывающими объект прогнозирования активностями). Обработка документации уже дело самой системы, т.е. таблично-цифровую базу знаний создает сама экспертная система.

Процесс создания ЭС достаточно трудоемок: требуется разработать средства управления базой знаний, логического вывода, диалогового взаимодействия с пользователем и т.д. Этап создания программного продукта – дело программиста. Если инженер знаний проектирует оболочку системы, то программист реализует ее и создает механизм вывода и возврата решений.

Объем программирования столь велик, а программы столь сложны и нетрадиционны, что имеет смысл, как это принято сейчас при разработке больших программ, сначала создать демонстрационный прототип системы - предварительный вариант, в котором в упрощенном виде реализованы лишь ее основные планируемые возможности и которая будет служить для заказчика подтверждением того, что разработка ЭС для решения данной задачи принципиально возможна, а для разработчиков - основой для последующего улучшения и развития системы. Процесс развития же заключается в циклическом тестировании и модификации прототипа системы при активном участии эксперта с целью выявления возможных ошибок в применяемых правилах или таблично-цифровой базе. Кроме эксперта в этом процессе участвует и пользователь, который проверяет, верные ли решения принимает система и облегчает ли ему работу ее помощь[75].

В последнее время компьютерные системы с базой знаний тесно переплетались с существующими деловыми системами. Их используют в здравоохранении, страховании, банковском деле и других областях, чтобы с помощью правил и объектов накапливать опыт, повышать качество принимаемых решений. Базы знаний встроены сегодня в наиболее современные крупные системы. Они находятся в самой сердцевине программ-агентов, осуществляющих поиск в сети Internet, и помогают коллективам пользователей получать нужную информации. Это, в свою очередь, обусловило применение средств Internet-программирования, таких как технологии ASP, PHP, языков программирования Perl, Java и др. и технологии клиент-сервер .

§ 1.5. Архитектура экспертной системы: основные компоненты

Одним из главных достоинств ЭС можно считать то, что они разделяют специфические знания и средства более общего назначения принятия решения. Часть общего назначения называется оболочкой системы. Оболочка экспертной системы - пакет программ, предназначенных для проектирования экспертных систем. Оболочки отнюдь не являются универсальной архитектурой [56]. Дело в том, что не существует настолько четкого представления о классификации задач, решаемых экспертными системами, чтобы можно было точно представить, к какому именно классу следует отнести конкретную систему. Нужно отметить, что большинство коммерческих продуктов этого типа подходит только для тех проблем, в которых пространство поиска невелико. Простота языков представления знаний, применяемых в большинстве оболочек, является, с одной стороны, достоинством, а с другой — недостатком такого рода систем[2].

По сравнению с первыми разработками современные оболочки более гибкие, они без особого труда могут быть интегрированы в большинство операционных сред, доступных на рынке программного обеспечения, и оснащены достаточно развитыми средствами пользовательского интерфейса. Оболочка, как правило, поддерживает индивидуальную настройку пользовательского интерфейса (например, применение меню) и возможность формирования пояснений при ответах на вопрос «почему».

Основными элементами экспертной системы являются:

База данных. База данных предназначена для хранения фактов или гипотез, являющихся результатом общения системы с внешней средой, в качестве которой обычно выступает человек, ведущий диалог с экспертной системой.

База знаний. База знаний предназначена для хранения экспертных знаний о предметной области, используемых при решении задач экспертной системой. База знаний содержит факты (или утверждения) и правила. Факты представляют собой краткосрочную информацию в том отношении, что они могут изменяться, например, в ходе консультации. Правила представляют более долговременную информацию о том, как порождать новые факты или гипотезы из того, что сейчас известно. Эти знания представляют собой данные для высокоуровневого интерпретатора, а именно - для

машины вывода. Различают два типа баз знаний: основанную на правилах и таблично-цифровую.

База данных и база знаний являются информационной основой ЭС. База данных содержит формализованную информацию об отдельных фактах, характеризующих объекты, процессы и явления предметной области, а также их свойствах (например, данные физико-химических свойствах веществ и т.д.). На данных, полученных практическим путем, основываются знания, представляющие собой структурированные данные (метаданные).

Машина вывода. Машина вывода - механизм, который необходим для построения логических вычислений (механизм рассуждений, оперирующий знаниями и данными с целью получения новых данных). Для этого обычно используется реализованный программно механизм логического вывода (какая-либо его разновидность) или механизм поиска решения в сети фреймов или семантической сети. В большинстве ЭС используются 2 режима вывода: прямая цепочка рассуждений (использование фактов) и обратная цепочка рассуждений (подтверждение или опровержение фактов). Наиболее удачные системы используют комбинации этих способов рассуждений.

Модуль приобретение знаний. Модуль приобретения знаний - это компонент, который автоматизирует процесс наполнения ЭС знаниями, осуществляемый пользователем-экспертом. Он необходим для получения знаний от эксперта, поддержки базы знаний и дополнения ее при необходимости.

Интерфейс пользователя. Интерфейс пользователя - диалоговый компонент, который ориентирован на организацию дружественного общения с пользователем как в ходе решения задач, так и в процессе приобретения знаний и объяснения результатов работы.

Система разьяснения. Позволяет объяснить пользователю принятое решение.

Ниже показана схема экспертной системы. Возможны те или иные модификации этой схемы в зависимости от решаемой задачи.

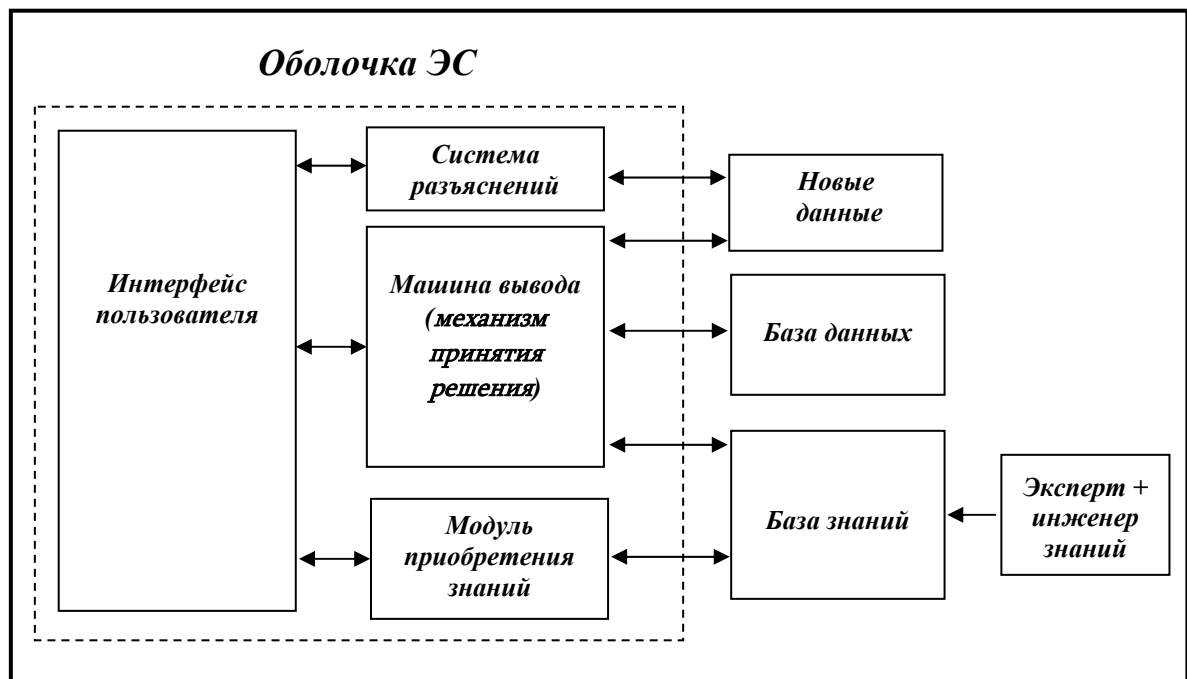


Схема экспертной системы

Как видно из приведенной схемы, оболочка обеспечивает механизм вывода решения (машина вывода), интерфейс пользователя, систему разъяснений, модуль приобретения знаний. В то же время база знаний не является компонентом оболочки ЭС. Отсюда явствует, что для новой задачи возможно применение готовой оболочки, нужно лишь наполнить содержанием базу знаний. В этом случае вся работа по созданию ЭС ложится на инженера знаний и эксперта.

Объединение всех видов программных продуктов и их отдельных компонентов в единую ЭС признано экономически выгодным, так как применение ЭС позволяет существенно сократить расходы на подготовку квалифицированного персонала, дальнейшую проверку работоспособности и надежности разрабатываемых и исследовательских систем, а также уменьшить время проектирования и(или) исследования.

§ 1.6. Представление знаний в экспертных системах

Все предметы и события, составляющие основу необходимой для решения задачи информации, называются *предметной областью*. Предметная область представляется состоящей из реальных компонент, называемых *сущностями*. Сущности каждой конкретной предметной области находятся между собой в определенных отношениях. Отношения между сущностями описывают посредством *суждений*. Суждения в каждом языке (как естественном, так и формальном) выражают в виде *предложений*. Языки предназначенные для описания предметных областей называются *языками представления знаний*. Для представления математического знания пользуются формальными логическими языками – исчислением высказываний и исчислением предикатов. Эти языки имеют ясную формальную семантику и для них разработаны формальные методы логического вывода. Поэтому исчисление предикатов было первым логическим языком, который стали применять для формального описания пригодных для этого предметных областей. Описания предметных областей, выполненные в логических языках, называются логическими моделями. Логические модели, построенные с применением языков логического программирования, широко применяются в базах знаний экспертных систем.

В настоящее время основными классами моделей представления знаний в экспертных системах являются фреймы, семантические сети, продукционные модели и модели нечетких знаний.

Один из наиболее популярных способов представления знаний основан на понятии «*формальной продукционной системы*», которое в 1943 г. ввел в обращение американский логик Пост. Он показал, что любая формальная система может быть представлена как система продукций, т.е. как совокупность правил $\{ Ri : \Phi i \rightarrow \Psi i \}$, где выражение $Ri : \Phi i \rightarrow \Psi i$ означает, что в данной формальной системе по правилу Ri из посылки Φi непосредственно выводится заключение Ψi .

В 1972-1973 гг. Ньюэл и Саймон показали, что человек в ходе рассуждений и деятельности использует правила, подобные продукциям, т.е. делает заключения по схеме «*условие->следствие*» и действует по схеме «*ситуация->действие*». Тогда же было предложено использовать продукционные системы для моделирования процессов принятия решений.

В общем виде под продукцией понимают утверждение типа «ЕСЛИ <перечень условий>, ТО <перечень действий >», или проще, «если A , то B », обозначаемое логическим выражением вида $A \rightarrow B$.

Продукция $A \rightarrow B$ может истолковываться в логическом смысле (как следование истинности B из истинности A). Возможны и другие интерпретации продукции $A \rightarrow B$ (например, A может быть описанием некоторого условия, выполнение которого необходимо, чтобы можно было совершить действие B). В зависимости от количества условий и действий в соответствующих перечнях различают следующие типы правил: простое – одно условие и одно действие, составное – много условий и действий, фиксирующее – много условий и одно действие, разветвляющееся – одно условие и много действий.

При использовании продукционных моделей для представления знаний создается продукционная система – набор продукций, организованный по определенному принципу. Для систем продукций создаются также определенные процедуры управления их функционированием, которые обычно решают следующие задачи:

- а) выявление совокупности активных правил – продукций, условия для применения которых выполнены (правил, которые могут действовать);
- б) разрешение конфликтов между правилами (с учетом приоритетов, эффективности, эвристик) и выбор правила для применения (правила, которое должно действовать);
- в) применение выбранного правила, то есть выполнение действий, предписываемых правой частью продукции (исполнение действия).

В различных оболочках используются различные способы управления системами продукций, определяющие и способы оформления самих продукций. В самом общем виде продукционное правило представляет собой набор $(I, Q, P, A \rightarrow B, N)$ следующих компонент:

I – имя (или порядковый номер) правила, имя может выражать суть правила, смысл выполняемого им действия;

Q – область применения правила – часто система продукций может быть разделена на подсистемы, каждая из которых связана с решением определенного типа задачи в комплексе задач;

P - предусловие применения правила - общие условия, при которых разрешено или имеет смысл обращаться к данному правилу;

$A \rightarrow B$ - ядро правила (*«если A , то B »*), где A и B , соответственно, либо посылка и заключение, либо ситуация и действие;

N - постусловие применения правила - другие последствия и изменения, обусловленные применением данного правила, не учтенные в правой части B его ядра.

Наиболее ориентированными на применение продукционных моделей представления знаний являются компьютерные системы, основанные на использовании глобальной базы данных. В состав такой системы входит база правил (множество продукций), глобальная база данных и система управления. База правил представляет собой совокупность знаний в форме правил вида $A \rightarrow B$ (*«если A , то B »*). Глобальная база данных содержит знания, представляющие собой данные или уже установленные факты. Система управления формирует заключения, используя базу правил и глобальную базу данных. Существуют два способа вывода заключений – путем построения *прямой* или *обратной цепочки рассуждений*. В концепции глобальной базы данных может быть реализован каждый из этих способов. При этом в прямых цепочках рассуждений вывод очередного заключения осуществляется путем сопоставления данных, содержащихся в базе данных, с левыми частями правил, и если находится правило, где сопоставление происходит, правая часть этого правила в качестве заключения по этому правилу помещается в базу данных (или исполняется предписываемое правой частью этого правила действие, соответствующим образом изменяющее содержимое базы данных). В обратных цепочках рассуждений вывод начинается от поставленной цели. Если находится правило, правая часть (заключение) которого сопоставляется с этой целью, то левая часть (посылка) правила принимается за подцель. Этот процесс обратного вывода повторяется до тех пор, пока не будет получено совпадение подцелей с данными. Концепция глобальной базы данных проста и поэтому удобна для теоретического исследования, но при большом числе продукций в такой базе знаний быстро усложняется анализ и разрешение противоречий между правилами. Поэтому число продукций, с которыми работают современные компьютерные системы такого типа, обычно не превышает тысячи. Глобальная база данных – центральная структура данных, используемая продукционной системой. В зависимости от

конкретной задачи эта база данных может быть *простой*, как обычная матрица чисел, или *сложной*, как большая реляционная индексированная файловая структура. Правила (продукции) применяются к глобальной базе данных. Каждое правило имеет предусловие, которому эта база данных в каждый момент либо удовлетворяет, либо нет. Если предусловие выполняется, то правило может быть применено. Применение этого правила изменяет базу данных. Система управления выбирает, какое именно из применимых правил следует использовать, и прекращает вычисления, когда глобальная база данных удовлетворяет терминальному условию (условию останова).

Продукционные модели (наряду с фреймами) являются наиболее распространенными средствами представления знаний (особенно в промышленных экспертных системах), так как обеспечивают простоту смысловой интерпретации, модульность, легкость корректировки и логического вывода. Применение продукционных правил способствует «прозрачности» экспертной системы, т.е. ее способности к объяснению принятых решений и полученных результатов. Недостатками продукционных моделей являются трудности при добавлении правил, зависящих от уже имеющихся в базе знаний, а также при отладке программ.

Теперь рассмотрим сетевую модель представления данных. В самом общем случае сетевая модель - это информационная модель предметной области. В сетевой модели задается множество информационных единиц (объекты и их свойства, классы объектов и их свойств) и отношения между этими единицами. В зависимости от типов отношений между информационными единицами сети можно подразделить на:

- классификационные (отношения типа часть-целое, род, вид, индивид); В классификационных сетях используются отношения, позволяющие описывать структуру предметной области, что позволяет отражать в базах знаний разные иерархические отношения между информационными единицами.

- функциональные (преобразование информационных единиц); Функциональные сети часто называют вычислительными моделями, т.к. они позволяют описывать процедуры «вычислений» одних информационных единиц через другие.

- каузальные (причинно-следственные отношения); В каузальных сетях, называемых также сценариями, используются причинно-следственные отношения, а также отношения типов «средство – результат», «орудие – действие» и т.п.

- смешанные (использующие разнообразные типы отношений).

Если в сетевой модели допускаются отношения различного типа, то ее обычно называют семантической сетью. Обычно сетевая модель представляется в виде ориентированного графа, вершины которого представляют понятия (объекты, события, явления), а дуги изображают отношения между понятиями (например, класс – элемент класса, свойство – значение).

Наибольшую известность в системах искусственного интеллекта получили сети смешанного типа (семантические сети и их разновидность - сети фреймов), использующие, в зависимости от области применения, самые разные типы отношений. Семантические сети находят применение в системах понимания естественного языка, в вопросно-ответных системах, в других различных предметно – ориентированных системах. Важной чертой семантических сетей является возможность представлять знания более естественным и структурированным образом, чем это делается с помощью других формализмов.

Семантические сети можно классифицировать по количеству типов отношений (однородные, неоднородные) и по типам отношений (бинарные, N-арные). Они обеспечивают наглядность отображения объектов, связей и отношений, а также гибкость представления знаний за счет ассоциативности и иерархичности. Недостатком семантических сетей и фреймов как моделей знаний является отсутствие универсальной процедуры их обработки, а также сложность организации процедур поиска вывода, что может привести к неэффективному использованию ресурсов вычислительной техники.

Фреймы - это фрагменты знания, предназначенные для представления стандартных ситуаций. Характерными для этого подхода являются: представление знаний в виде достаточно крупных, содержательно завершенных единиц, называемых фреймами; иерархическая структура фреймов, где иерархия основана на степени абстрактности фреймов; совмещение в фреймах декларативных и процедурных знаний.

Автором теории фреймов является М. Минский. В основе этой теории лежат психологические представления о том, как мы видим, слышим и концентрируем воспринимаемое. Сам Минский излагал суть теории фреймов следующим образом: каждый раз, попадая в некую ситуацию, человек вызывает из своей памяти соответствующую ситуации структуру, именуемую фреймом (frame - рамка). Фрейм -

это единица представления знания, заполненная в прошлом, детали которой по необходимости изменяются и уточняются применительно к ситуации. Каждый фрейм может быть дополнен различной информацией, касающейся способов применения данного фрейма, последствий этого применения и т.п. Например, образ жизни каждого человека - это, большей частью, череда типовых ситуаций, различающихся каждый раз в деталях, но в общем и целом повторяющихся.

Структуру фрейма можно представить в виде:

$$N : \{ \langle S_1, V_1, P_1 \rangle, \dots, \langle S_k, V_k, P_k \rangle, \dots, \langle S_n, V_n, P_n \rangle \},$$

где N - имя фрейма; $\langle S_k, V_k, P_k \rangle$ ($k=1, \dots, n$) - слот; S_k - имя слота; V_k - значение слота; P_k - процедура. Из этого определения видно, что фрейм имеет иерархическую структуру: на верхнем уровне располагаются фиксированные характеристики ситуации, на последующих уровнях (в т.н. «слотах» - отсеках) - уточняющая и конкретизирующая информация. Таким образом, слоты - это структурные элементы фрейма, заполнение которых приводит к тому, что фрейм ставится в соответствие некоторой ситуации, явлению, объекту или процессу. Имена фреймов и слотов обеспечивают интерпретируемость хранящихся во фреймах значений, поэтому имя фрейма должно быть единственным во всей фреймовой системе, а имена слотов должны быть различны в пределах фрейма. Фреймы имеют свойство вложенности, т.к. значениями слота могут быть имена слотов более глубокого уровня или имена других фреймов.

Знания предметной области представляются фреймами-объектами, фреймами-сценариями, фреймами-ролями, фреймами-ситуациями и др. Например, фреймами-объектами можно представить знания о веществах, процессах и т.д.

Различают также пользовательское и машинное представление фреймов. С точки зрения пользователя различают три уровня общности фреймов: а) скелетный, пустой фрейм (*шаблон*), превращаемый после его заполнения в общее или конкретное понятие; б) фрейм общего понятия (*прототип*) - шаблон, заполненный не конкретными значениями, константами, а переменными; в) фрейм конкретного понятия (экземпляр) - прототип, заполненный конкретными значениями, константами.

Каждый слот предназначен для заполнения определенной структурой данных (в скелетном фрейме все они пусты, кроме первого, который имеет значение). Значением слота может быть практически все, что угодно (числа или математические соотношения,

тексты на естественном языке или программы, правила вывода или ссылки на другие слоты данного фрейма или других фреймов). При конкретизации фрейма ему и его слотам присваиваются конкретные имена и происходит заполнение слотов их значениями. Переход от исходного фрейма-прототипа к фрейму-экземпляру может быть многошаговым (за счет постепенного уточнения значений слотов).

Внутреннее (машинное) представление фрейма имеет более сложную организацию и содержит средства для создания иерархии фреймов, их взаимодействия, обмена информацией, порождения конкретных фреймов из общих и общих из скелетных.

Фреймовые модели часто рассматриваются в общем контексте с сетевыми моделями. В частности, сеть фреймов можно рассматривать как семантическую сеть с блочной структурой. Фрейм в такой сети содержит информационный и процедурный элементы, которые обеспечивают преобразование информации внутри фрейма и его связь с другими фреймами. Слоты фрейма заполняются конкретной информацией в процессе функционирования фрейма. В сети фреймов могут быть также реализованы и логические связки, и кванторы общности и существования.

Итак, можно сделать вывод что на некотором глубинном уровне все формы представления знания равносильны (в том смысле, что они универсальны, и знания, представленные в одной форме, могут быть преобразованы в другую), но не равноценны (в том смысле, что для различных предметных областей и различных задач более удобными и эффективными в вычислительном отношении оказываются различные формы представления знания).

§ 1.7. Представление и обработка нечетких данных в ЭС

Модели представления нечетких знаний используются для формализации человеческих знаний, описывающих качественные характеристики (например, большой, сильный, очень сильный, высокий и т.п.) объектов предметной области, которые могут интерпретироваться неоднозначно, но содержат важную информацию.

В реальных условиях знания, которыми располагает человек, всегда в какой-то степени размыты. Тем не менее, людям на основе таких знаний все же удастся делать

достаточно обоснованные выводы и принимать разумные решения. Следовательно, чтобы интеллектуальные системы были действительно полезны, они должны быть способны учитывать неполную определенность знаний и успешно действовать в таких условиях. Неопределенность может иметь различную природу. Наиболее распространенный тип недостаточной определенности знаний обусловлен объективными причинами: 1). действием случайных и неучтенных обстоятельств; 2). неточностью измерительных приборов; 3). ограниченными способностями органов чувств человека; 4). отсутствием возможности получения необходимых свидетельств. В таких случаях люди в оценках и рассуждениях прибегают к использованию вероятностей, допусков и шансов (например, шансов победить на выборах).

Другой тип неопределенности, можно сказать, обусловлен субъективными причинами: 1).неоднозначностью смысла слов и высказываний (например, "ключ"); 2).нечеткостью содержания используемых человеком понятий (например, «толпа»). Неоднозначность смысла слов и высказываний часто удается устранить, приняв во внимание контекст, в котором они употребляются, но это тоже получается не всегда или не полностью.

Таким образом, неполная определенность и нечеткость имеющихся знаний - скорее типичная картина при анализе и оценке положения вещей, при построении выводов и рекомендаций, чем исключение. В процессе исследований по искусственному интеллекту для решения этой проблемы выработано несколько подходов.

Самое серьезное внимание этой проблеме стали уделять при создании экспертных систем и первым здесь был применен вероятностный подход, поскольку теория вероятностей и математическая статистика в тот период были уже достаточно развиты и весьма популярны. Примером систем, использующих вероятностный подход, могут служить экспертные системы MYCIN и PROSPECTOR[10]. Функционирование этих систем основано на хорошо организованных базах знаний в предметной области: MYCIN – в области диагностирования и лечения инфекционных заболеваний крови, PROSPECTOR – в области геологии (автоматизированная оценка данных геологической разведки - прогнозирование залежей полезных ископаемых). Обе ЭС построены на основе правил продукций в следующем формате: «ЕСЛИ предпосылки, ТО

действие». Обработка же недостоверных данных и знаний производится с применением некоторого вида весовых оценок, используются вероятностные формы вывода заключений. Большое значение в обеих системах имеет установление меры доверия (т.н. «доверительности»), т.е. вероятности справедливости полученного заключения. В MYCIN мера доверия выражается с помощью т.н. «коэффициентов достоверности», в PROSPECTOR используются условные вероятности и теорема Байеса.

В дальнейшем исследования в этой области привели к разработке особой (нечеткой) логики (Fuzzy Logic). Основы ее были заложены Лотфи Заде, который ввел понятия «частичной принадлежности» элемента подмножеству [59] (отсюда - понятие нечеткого подмножества). Создание нечеткой логики привело к новому этапу развития компьютерных систем, основанных на базах знаний. Это и понятно, т.к. в случаях, когда либо сами данные, либо объект исследования имели нечеткую природу, применение Булевой логики давало неточные результаты [47].

Процесс принятия решений в нечетких экспертных системах можно подразделить на 4 этапа [13]:

5. Фазификация (Fuzzification) – на этапе фазификации происходит вычисление совместимых значений исходных данных на основе применения функций совместимости нечетких понятий. Если исходные данные изначально задаются совместимыми значениями (от 0 до 1), необходимость в этапе фазификации отпадает.

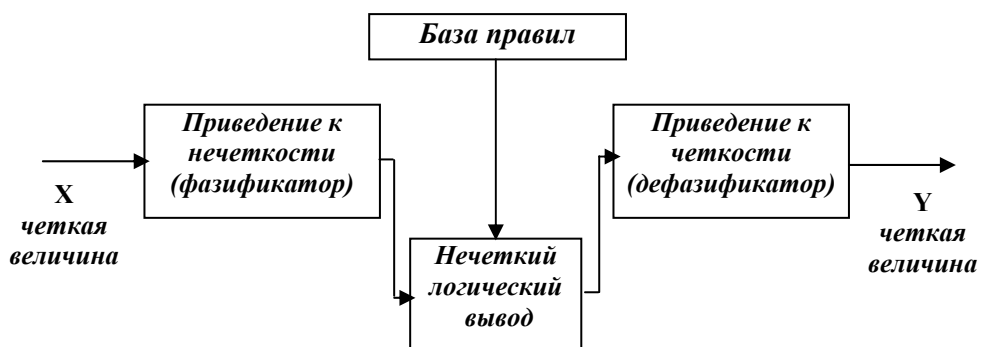
6. Вывод (Inference) – для каждого выводимого понятия (решения) вычисляется возможные доверительные значения.

7. Композиция (Composition) – в случае применения более чем одного метода вывода решения необходима их композиция, чтобы было получено единственное нечеткое подмножество выводимых значений.

При этом можно использовать опять-таки нечеткий метод (например, метод экспертонов), который позволит учесть предпочтения группы экспертов.

8. Дефазафикация (Defuzzification) – используется для того, чтобы из нечеткого подмножества выводимых значений выбрать единственное, которое и определит окончательно принятое решение. Как правило, при этом используются принцип максимума возможностей, нечеткие операции минимума или умножения.

Процесс дефазификации заканчивается приведением нечеткого понятия к четкому.



В настоящее время известны нечеткие компьютерные системы принятия решения с базами знаний, основанными как на правилах [6], так и с таблично-цифровой архитектурой (их значительно меньше). В компьютерных системах принятия решения с таблично-цифровой базой знаний механизм вывода решения, как правило, применяет нечеткие статистические методы, использование же именно таблично-цифровой базы знаний обусловлено самими алгоритмами принятия решения этих методов [43]. База знаний в этом случае может содержать четкие и нечеткие значения, получаемые с помощью определенных экспертами правил и функций совместимости, а также некоторые вычисляемые нечеткие значения, необходимые в дальнейшем на этапе вывода решения. К таким методам нечеткой статистики относятся дискриминационный анализ, анализ связности[44]. Новшеством является применение статистического метода анализа нечетких классов[62], метода CBR[25], а также метода экспертонов([32], [33]).

§ 1.8. Программное обеспечение ЭС

Как отмечалось выше, программное обеспечение для экспертных систем, а в частности и для систем, применяющих нечеткую логику, создается с учетом, во-первых, решаемой задачи, а во-вторых, с учетом типа базы знаний. Наиболее актуальными средствами программирования в нечетких системах являются логическое программирование ([5], [36], [42]) и объектно-ориентированное программирование ([4], [17], [76]).

Объектная технология прекрасно подходит аналитикам и программистам, т.к. очень напоминает стратегию решения проблем и соответствует мыслительным

процессам людей, считающихся экспертами в своей области. Методика объектно-ориентированного программирования основана на модели, напоминающей образы, возникающие в мозгу аналитика, которая представляет предметы и процессы в виде объектов и связей между ними. Наблюдая событие, эксперт легко выделяет знакомые образы. Для решения проблем он испытывает конкретные правила, рассматривая при этом исследуемую проблему под определенным ракурсом.

В последние два десятилетия широкое распространение получили экспертные системы как средство для тиражирования и доставки квалифицированных узкоспециализированных знаний в то место, где они необходимы для принятия решений. С широким распространением Internet появилась возможность централизованного хранения этих знаний и предоставления доступа к ним через каналы связи. Отсюда и использование новых средств программирования. В основном это применяемые для локализованной экспертной системы технологии веб-программирования - PHP, ASP, JavaScript, Flash (для визуализации результатов принятых решений), язык интернет-программирования Perl и технология клиент/сервер, позволяющая пользователю осуществлять доступ к базе данных и базе знаний, располагающейся на стороне сервера. Технология клиент/сервер применительно к СУБД(система управления базами данных) сводится к разделению системы на две части - приложение-клиент и сервер базы данных, которые могут располагаться на одном компьютерном узле.

Когда же процесс принятия решений организован в виде распределенной компьютерной системы поддержки принятия решений, то программными средствами могут служить технология ASP.NET, с применением которой возможно использование объектно-ориентированных языков программирования, таких как C#, C++ и VB.NET, а также объектно-ориентированный, динамический, переносимый, высокопроизводительный и многопоточный язык программирования Java.

Г л а в а II .

Нечеткие статистические методы принятия решения

В этой главе будут введены основные понятия нечеткого анализа, описаны 4 нечетких метода принятия решения, такие как: статистический метод анализа нечетких

классов, дискриминационный метод, метод рассуждений, основанный на прецедентах (Case Based Reasoning) и метод экспертов. В работе перечисленные методы используются для принятия решения в задаче прогнозирования, однако их, с таким же успехом, можно применять и в других областях (например, в медицинской диагностике). Целью задачи прогноза является на основании множества выявленных активностей (факторов, определяющих состояние объекта исследования) принять решение о возможности того или иного прогноза. Объект, для которого принимается решение, должен быть представлен соответствующей прогнозируемой величиной. В нашем случае прогнозируемую величину можно описать с помощью лингвистической переменной [69] со значениями: «слабое», «среднее», «сильное», «разрушительное» (землетрясение) и т.д. Очевидно, что в этом случае невозможно провести строгих границ между классами классификации, т. е. прогнозируемое понятие содержит нечеткость. Если классы классификации оказываются нечеткими множествами, применение методов классической статистики не дает достоверных результатов. Это и обусловило применение нечетких статистических методов.

Значения активностей, представляющие первичные данные (результаты наблюдений, измерений и экспертных оценок) вместе с правильно принятыми решениями сконцентрированы в базе данных общего назначения (MySQL). При создании базы знаний применены следующие подходы:

- *статистика нечетких классов* – по значениям активностей, характерным для известных прогнозов, после построения образов нечетких классов классификации прогнозируемого объекта и функций совместимости прогнозируемых понятий оценивается «близость» исследуемого случая одному из образов;

- *дискриминационный анализ* – активности оцениваются по признаку того, насколько представительными они являются для конкретного прогноза по сравнению с другим;

- *рассуждения, основанные на прецедентах* – выявляют наиболее возможное решение после оценки «близости» активностей к активностям какого-нибудь известного случая прогноза.

4-тый метод – метод экспертов – используется для выбора из решений, предложенных первыми тремя методами, наиболее возможного решения на основании предпочтений группы экспертов.

Прежде, чем описать каждый из нечетких методов, введем несколько определений теории нечетких множеств.

§ 2.1. Нечеткое множество

Строгое определение нечеткого множества по Заде [61] выглядит следующим образом: Пусть задано множество X (которое может быть набором произвольных объектов или математических конструкций), а x - его элемент. Конечным нечетким подмножеством \tilde{A} множества X называется множество упорядоченных пар вида: $\tilde{A} = \{ \mu_A(x) | x \}$, где $x \in X, \mu_A(x) \in [0,1]$. $\mu_A(x)$ определяет меру членства, которая указывает предполагаемую степень принадлежности элемента x нечеткому множеству \tilde{A} . При этом множество X называется базовым множеством или базовой шкалой (если множество X линейно упорядочено). Принято, что в нечеткое множество \tilde{A} не входят элементы $x \in X$, имеющие $\mu_A(x) = 0$. Подмножество $A \subseteq X$, содержащее все те элементы $x \in X$, для которых $\mu_A(x) > 0$, называется носителем нечеткого множества \tilde{A} . Функция $\mu_A(x): X \rightarrow [0,1]$ называется функцией принадлежности (Membership Function) или совместимости множества \tilde{A} и представляет собой обобщение понятия характеристической функции обычного множества. Значение $\mu_A(x) = 0$ означает отсутствие принадлежности к множеству, $\mu_A(x) = 1$ – полную принадлежность. Если все $\mu_A(x) \in \{0,1\}$, т.е. равны 0 либо 1, то нечеткое множество становится «обычным» четким множеством, а функция $\mu_A(x)$ - обычной булевой функцией.

Рассмотрим простой пример: формализуем неточное определение «горячий». В качестве X (область рассуждений) будет выступать шкала температуры в градусах Цельсия. Пусть она изменяется от 0 до 100 градусов. Нечеткое множество для понятия «горячий» может выглядеть следующим образом:

$$\tilde{A} = \{0|0; 0|10; 0|20; 0.1|30; 0.3|40; 0.6|50; 0.8|60; 0.9|70; 1|80; 1|90; 1|100\}.$$

Так, при температуре 60°C уровень принадлежности к множеству «горячий» равен 0.8: для одного человека жидкость при температуре 60°C может оказаться горячей, для другого – не слишком горячей. Именно в этом и проявляется нечеткость задания соответствующего множества.

Для качественного описания нечетких объектов вводятся понятия нечеткой переменной и лингвистической переменной [69]. Рассмотрим, например, такое нечеткое понятие как «Землетрясение». Это и есть название лингвистической переменной. Сформируем для нее базовое множество X , которое будет состоять из трех нечетких переменных: «слабое», «среднее», «сильное», и количественно зададим область рассуждений в виде $X = [1,8]$ (единиц). Для нечетких переменных также можно ввести количественные характеристики: «слабое» $\Rightarrow [1,3]$, «среднее» $\Rightarrow [3,5]$, «сильное» $\Rightarrow [5,8]$ тех же единиц. Последнее, что осталось сделать – построить функции совместимости (принадлежности) для каждого элемента из базового множества X .

Для нечетких множеств, как и для обычных, определены основные логические операции. Самыми распространенными являются отношение вложенности, определение дополнительного нечеткого множества, произведения и суммы нечетких множеств:

отношение вложенности: $A \subset B: \mu_A(x) \leq \mu_B(x), \forall x \in X$;

дополнительное нечеткое множество: $\mu_{\bar{A}}(x) = 1 - \mu_A(x), \forall x \in X$, где

$\mu_{\bar{A}}(x)$ - отношение дополнения;

произведение (нечеткое «И»): $\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)), \forall x \in X$, или

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x), \forall x \in X;$$

сумма (нечеткое «ИЛИ»): $\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)), \forall x \in X$, или

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x), \forall x \in X.$$

Вводится еще целый ряд операций над нечеткими множествами такие как унарная операция степени α нечеткого множества, бинарные операции: алгебраическое произведение, алгебраическая и ограниченная сумма, разность и абсолютная разность и др.

§ 2.2. Функции принадлежности (совместимости) нечеткому множеству

Как уже отмечалось, функция совместимости представляет собой обобщение понятия характеристической функции обычного множества. Функция совместимости строится исходя из субъективных предпочтений экспертов, поэтому ее вид может быть произвольным. Однако при определении функции принадлежности нечеткому множеству стараются достичь определенного уровня стандартизации [9]. Например, представление таких нечетких подмножеств, как «нуль», «малое», «среднее», «большое», «очень большое» при базовом множестве из диапазона $[0;10]$ и один из вариантов функций принадлежности, построенной для этих подмножеств таков:

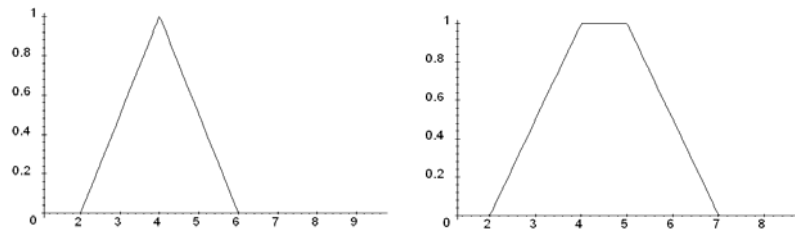
<i>Уровень</i>	<i>Нуль</i>	<i>Малое</i>	<i>Среднее</i>	<i>Большое</i>	<i>Очень большое</i>
0.00	0.00	0.00	0.00	0.00	0.00
0.50	0.25	0.00	0.00	0.00	0.00
1.00	0.50	0.00	0.00	0.00	0.00
1.50	0.75	0.00	0.00	0.00	0.00
2.00	1.00	0.00	0.00	0.00	0.00
2.50	0.75	0.25	0.00	0.00	0.00
3.00	0.50	0.50	0.00	0.00	0.00
3.50	0.25	0.75	0.00	0.00	0.00
4.00	0.00	1.00	0.00	0.00	0.00
4.50	0.00	0.75	0.25	0.00	0.00
5.00	0.00	0.50	0.50	0.00	0.00
5.50	0.00	0.25	0.75	0.00	0.00
6.00	0.00	0.00	1.00	0.00	0.00
6.50	0.00	0.00	0.75	0.25	0.00
7.00	0.00	0.00	0.50	0.50	0.00
7.50	0.00	0.00	0.25	0.75	0.00
8.00	0.00	0.00	0.00	1.00	0.00
8.50	0.00	0.00	0.00	0.75	0.25
9.00	0.00	0.00	0.00	0.50	0.50
9.50	0.00	0.00	0.00	0.25	0.75
10.00	0.00	0.00	0.00	0.00	1.00

Существует свыше десятка типовых форм кривых для задания функций принадлежности. Наибольшее распространение получили: треугольная, трапецидальная и гауссова функции принадлежности.

Функции согласованности треугольного и трапецидального вида описываются следующими формулами

$$\mu(x) = \begin{cases} 1 - \frac{b-x}{b-a}, & a \leq x \leq b, \\ 1 - \frac{x-b}{c-b}, & b \leq x \leq c, \\ 0, & \text{в остальных случаях.} \end{cases} \quad \mu(x) = \begin{cases} 1 - \frac{b-x}{b-a}, & a \leq x \leq b, \\ 1, & b \leq x \leq c, \\ 1 - \frac{x-c}{d-c}, & c \leq x \leq d, \\ 0, & \text{в остальных случаях.} \end{cases}$$

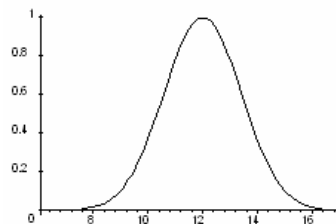
и имеют вид



Функция принадлежности гауссова типа описывается формулой

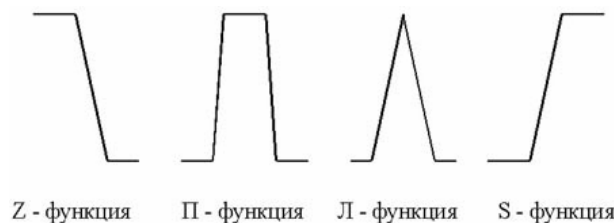
$$\mu(x) = \exp\left[-\left(\frac{x-c}{\sigma}\right)^2\right]$$

и оперирует двумя параметрами. Параметр c обозначает центр нечеткого множества, а параметр σ определяет крутизну функции.



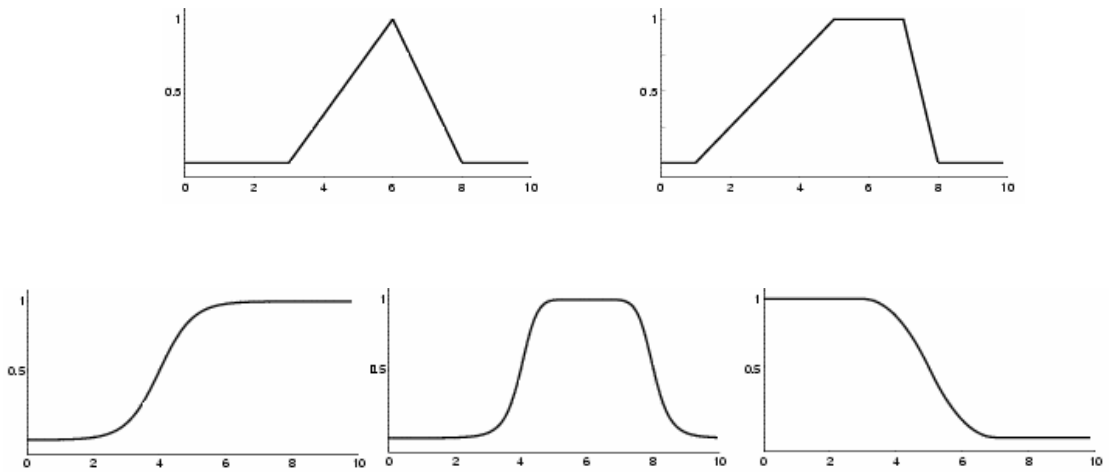
Гауссова функция принадлежности ($\sigma = 2$).

В последнее время появилось понятие т.н. стандартных кусочно-линейных функций принадлежности:



Стандартные кусочно-линейные функции принадлежности

Применяются также различные модификации кусочно-линейных и гауссовых моделей.



Построению моделей функций совместимости посвящено большое количество специальной литературы ([3],[15]). Самые известные из них: модель Заде, различные кусочно-линейные модели, модель Димитру и Любана, функции Сваровена, обобщенная модель Цисно и Циммерманна и др.

Эвристическая модель Заде, различные модификации которой позднее применялись разными авторами, определяет функции принадлежности для нечеткого множества «возраст» и описывается следующими формулами:

$$\mu_{Young}(x) = \begin{cases} 1, & x \leq 25, \\ \frac{1}{1 + \left(\frac{x-25}{5}\right)^2}, & x > 25. \end{cases} \quad \mu_{Old}(x) = \begin{cases} 0, & x < 50, \\ \frac{1}{1 + \left(\frac{x-50}{5}\right)^2}, & x \geq 50. \end{cases}$$

Возможно выделить ряд общих свойств, которым должна удовлетворять функция принадлежности:

1. любая функция принадлежности должна быть непрерывной;
2. любая функция принадлежности отображает интервал $[a, b]$ в $[0, 1]$:

$$\mu : [a, b] \rightarrow [0, 1];$$

3. функции принадлежности или
 - а). монотонно возрастающие или
 - б). монотонно убывающие или
 - в). частично монотонно возрастающие и монотонно убывающие;
4. на всем интервале определения функции принадлежности или

а). выпуклые или б). вогнутые или в). выпукло-вогнутые, т.е. существует такое значение $c \in [a, b]$, что на интервале $[a, c]$ функция выпуклая, а на интервале $[c, b]$ – вогнутая;

5. для монотонно возрастающих функций $\mu(a) = 0, \mu(b) = 1$, для монотонно убывающих – $\mu(a) = 1, \mu(b) = 0$.

В частности, если рассматривать функцию совместимости $[0,1] \rightarrow [0,1]$, то она должна удовлетворять следующим свойствам:

- монотонно возрастающая

$\mu(x)$ непрерывно возрастающая функция $[0,1] \rightarrow [0,1]$

$$\mu(0) = 0, \mu(1) = 1$$

$$\mu'(0) = \mu'(1) = 0;$$

- монотонно убывающая

$\mu(x)$ непрерывно убывающая функция $[0,1] \rightarrow [0,1]$

$$\mu(0) = 1, \mu(1) = 0$$

$$\mu'(0) = \mu'(1) = 0;$$

- унимодальная функция совместимости

$\mu(x)$ непрерывная функция $[0,1] \rightarrow [0,1]$, при этом на интервале

$[0, 1/2]$ она возрастающая, а на интервале $[1/2, 1]$ – убывающая

$$\mu(0) = \mu(1) = 0, \mu(1/2) = 1$$

$$\mu'(0) = \mu'(1) = \mu'(1/2) = 0.$$

§ 2.3. Статистический метод анализа нечетких классов

Опишем математическую модель прогноза, построенную с применением статистического метода анализа нечетких классов (статистики нечетких классов)[34, 35]

В математической модели объект прогнозирования представлен соответствующей прогнозируемой величиной, вся область значений которой делится на прогнозируемые классы: M_1, M_2, \dots, M_n . Каждому классу ставится в соответствие численный

интервал. Определяются соответствующие функции совместимости прогнозируемых понятий (функции принадлежности классу): $\mu_1, \mu_2, \dots, \mu_n$. Определение этих функций всегда содержит в себе субъективный момент – мнение эксперта о степени принадлежности прогнозируемого объекта прогнозируемому классу. «Хорошее» определение функций совместимости прогнозируемых понятий является основной гарантией успеха работы метода. Т. к. упомянутые классы являются нечеткими, суппорты функций совместимости пересекаются.

Значение прогнозируемой величины зависит от определенных параметров, или прогнозирующих (помогающих составить прогноз) факторов: X_1, X_2, \dots, X_p . Каждый из факторов, в свою очередь, делится на классы (подфакторы): $X_{k1}, X_{k2}, \dots, X_{km}$, где $k = \overline{1, p}$; $X_k = \bigcup_{i=1}^m X_{ki}$. Количество прогнозирующих факторов и их классов, а также пределы их численных интервалов могут быть выбраны произвольно, образуя нечеткие подмножества.

Для выборочных частот j -того класса X_k –того фактора соответствующего i -того прогнозируемого класса вводятся обозначения n_{kj}^i . В математической модели эти величины представляют первичную информацию (первичные данные), и их получают непосредственным наблюдением и измерениями. При помощи величин n_{kj}^i и μ_i по известным формулам определяются нечеткие выборочные частоты и нечеткие относительные частоты : [62]

$$\tilde{n}_{kj}^m = \sum_i \mu_i^m \cdot n_{kj}^i, \quad \tilde{f}_{kj}^m = \frac{\tilde{n}_{kj}^m}{\sum_i \tilde{n}_{kj}^i}, \quad (1)$$

где μ_i^m – среднее значение функции совместимости, когда прогнозируемая величина из i -того прогнозируемого интервала принадлежит m -тому прогнозируемому классу.

Вычисляются также нечеткие веса для каждого интервала прогнозирующего фактора:

$$w_{kj} = \frac{\sum_i \tilde{n}_{kj}^i}{\sum_j \sum_i \tilde{n}_{kj}^i}. \quad (2)$$

После этого для определенной выборки активностей прогнозируемой величины (прогнозирующих факторов) уже можно сделать прогноз: необходимо только определить нечеткие веса каждого прогнозирующего фактора в соответствии с его интервалом и осуществить многофакторный линейный синтез нечетких весов и нечетких относительных частот, в результате чего получается соответствующее обобщенное решение (взвешенный вектор возможных решений) $\vec{D}_\alpha = \vec{w}_\alpha \cdot \vec{f}_\alpha$, которому соответствует мера возможности:

$$\overrightarrow{Poss}_\alpha = \frac{\vec{D}_\alpha}{\max_j(D_\alpha(j))}, \text{ где } D_\alpha(j) - j\text{-тая компонента вектора } \vec{D}_\alpha.$$

Для получения классического, единственного решения необходимо ввести дополнительный принцип. Например, можно воспользоваться принципом максимума возможностей. Тогда окончательное решение имеет вид[57]:

$$D_{Class}^{(\alpha)} = \max_i(Poss_\alpha(i)), \text{ где } Poss_\alpha(i) \text{ есть } i\text{-тая компонента вектора } \overrightarrow{Poss}_\alpha.$$

Используя предложенный метод следует помнить, что между прогнозирующими активностями и прогнозируемым объектом должна существовать заметная корреляция, а между прогнозирующими факторами - зависимость. Кроме того, необходимо следить, чтобы в выборке первичных классических частот не встречалось большого количества нулевых значений. Иначе это будет иметь статистический эффект. И последнее, в пользу предложенного метода говорит тот факт, что для получения удовлетворительного результата прогноза достаточно сравнительно небольшого количества исходных данных[27].

§ 2.4. Дискриминационный анализ

Дискриминационный анализ основан на таблично-цифровой базе и не требует характерных для теории вероятностей и математической статистики вычислительных методов. Он наиболее подходит для моделирования интеллектуальной активности

эксперта в процессе принятия решений. Суть метода дискриминационного анализа состоит в следующем[64, 65]:

На основе информации, полученной из базы первичных данных создается таблица распределений частот $\{f_{ij}\}$. По горизонтали этой таблицы расположены возможные решения D_j (факт подготовки или наличия землетрясения), а по вертикали – активности A_i : (в нашем случае, факторы, которые проявляются во время или при подготовке землетрясения, т.е. его предвестники). f_{ij} – относительная частота активности A_i , которая проявляется при решении D_j .

$$f_{ij} = \frac{m_{ij}}{N}, \quad (1)$$

где m_{ij} - число тех правильно принятых решений D_j , при которых проявилась активность A_i , а N - общее число случаев.

	D_1	D_2	...	D_n
A_1	f_{11}	f_{12}	...	f_{1n}
...
A_m	f_{m1}	f_{m2}	...	f_{mn}

На основании таблицы распределений частот строятся две новые таблицы: положительной $\{p_{ij}\}$ и отрицательной $\{n_{ij}\}$ дискриминаций по следующим формулам:

$$p_{ij} = \frac{1}{C_D - 1} \sum_{\substack{k \in D \\ k \neq j}} \chi_{Large-ratio} \left(\frac{f_{ij}}{f_{ik}} \right), \quad (2)$$

$$n_{ij} = \frac{1}{C_D - 1} \sum_{\substack{k \in D \\ k \neq j}} \chi_{Large-ratio} \left(\frac{f_{ik}}{f_{ij}} \right),$$

где $p_{ij}, n_{ij} \in [0;1]$ и C_D означает мощность множества решений: $C_D = CardD = Card\{D_1, \dots, D_n\} = n$. *Large-ratio* означает нечеткое подмножество множества R^+ с функцией принадлежности: $\chi_{Large-ratio} : R^+ \rightarrow [0;1]$, которая сопоставляет отношения (действительные числа) интервалу $[0,1]$.

Смысл положительной и отрицательной дискриминации следующий: p_{ij} это достоверность того, что i -тая активность более представительна для j -того решения, чем для какого-либо другого, а n_{ij} это достоверность того, что i -тая активность менее представительна для j -того решения, чем для какого-либо другого.

Теперь, если новые рассматриваемые данные описываются множеством активностей (A'_1, \dots, A'_r) , из таблиц $\{p_{ij}\}$ и $\{n_{ij}\}$ выбираются только те строки, которые соответствуют $\{A'_j\}$ и составляются новые таблицы $\{p'_{ij}\}$ и $\{n'_{ij}\}$.

Обобщенное решение представлено нечетким подмножеством множества возможных решений со следующей функцией принадлежности:

$$\delta(D_j) = \frac{1}{2} (\chi_{Large}(\pi_j) + \chi_{Small}(v_j)), \quad j \in D, \quad (3)$$

где

$$\pi_j = \frac{1}{C_{A'}} \sum_{i=1}^r p'_{ij}, \quad v_j = \frac{1}{C_{A'}} \sum_{i=1}^r n'_{ij}, \quad (4)$$

$$C_{A'} = Card A' = Card \{A'_1, \dots, A'_r\} = r$$

p'_{ij} и n'_{ij} – элементы таблиц $\{p'_{ij}\}$ и $\{n'_{ij}\}$ соответственно.

π_j и v_j представляют собой, средние значения (средние меры) положительной и отрицательной дискриминаций для j -того решения соответственно.

χ_{Large} – функция принадлежности (совместимости) нечеткого подмножества $Large(L)$, χ_{Small} – функция принадлежности нечеткого подмножества $Small(S)$. Функция χ_L – монотонно возрастающая, а χ_S – монотонно убывающая функция:

$$\chi_L, \chi_S : [0;1] \rightarrow [0;1].$$

Для получения окончательного «классического» решения можно выбрать принцип максимальной возможности: решение принимается на основании максимального значения функции принадлежности:

$$\delta^{Class} = \max_{j \in D} \delta(D_j), \text{ где } \delta(D_j) \text{ – } j\text{-тое решение.}$$

В качестве этого окончательного решения получается четкое подмножество, ближайшее к нечеткому $\delta(D_j)$, другими словами, решение δ^{Class} с максимальным значением в $\delta(D_j)$ считается самым достоверным решением[11].

§ 2.5. Метод рассуждений на основе прецедентов (Case Based Reasoning)

Идея метода достаточно проста. Для того, чтобы было принято новое правильное решение или сделан новый прогноз, среди прошлых случаев ищутся аналогии с новой ситуацией и принимается такое же решение, которое было верным для аналогов[25]. Поэтому этот метод часто называют методом «ближайшего соседа» (nearest neighbour). Метод CBR имеет ряд преимуществ:

- он близок к реальным процессам принятия решения экспертами, т.е. проблема разрешается с помощью сравнения с прошлыми похожими ситуациями,
- автоматизация процесса объединения исторических (базовых) и новых знаний достаточно проста, в этом случае применяется таблично-цифровая база знаний,
- возможно лучшее объяснение и обоснование принятого решения на основе рассмотрения предыдущих случаев и др.

Из базы данных общего назначения выбираются атрибуты (активности) прецедентов, соответствующие правильно принятым решениям и строится соответствующая таблица базы знаний.

Вычисляются меры сходства между новым и всеми существующими случаями:

3. мера сходства между двумя i -тыми активностями двух случаев вычисляется при помощи следующих равенств:

если активности четкие величины $DV_i = \min(CB_i, ND_i), i = \overline{1, n}$.

в случае нечеткости активностей $DV_i = 1 - |CB_i - ND_i|, i = \overline{1, n}$.

Здесь n – число всех активностей, CB_i – значение i -той активности существующего случая, ND_i – значение i -той активности исследуемого случая, DV_i – расстояние между двумя i -тыми активностями существующего и нового случая.

4. мера сходства между двумя случаями

Пусть SV_j представляет собой величину сходства между новым и j -тым из существующих случаев. Тогда ее можно вычислить при помощи следующей формулы

$$SV_j = \frac{1}{n} \sum_{i=1}^n w_i \cdot DV_i, \quad j = \overline{1, k}, \quad i = \overline{1, n},$$

где \vec{w} представляет собой вектор весов, каждый i -тый компонент которого показывает, каков уровень значимости i -той активности для принимаемого решения. $w_i \in [0, 1]$, при этом, $w_i = 0$ означает, что i -тая активность для принятия решения значения не имеет, $w_i = 1$, что i -тая активность для принятия решения очень значительна. Компоненты вектора весов определяются на уровне экспертов. k – количество всех существующих случаев (прецедентов).

Для получения окончательного решения применяется принцип максимума, т.е. ищется такое число r , которое удовлетворяет условию

$$SV_r = \max(SV_j), \quad j = \overline{1, k}.$$

Если мера сходства SV_r больше заранее установленного экспертами порога, тогда принимается решение, характерное для этого прецедента. Если же она меньше, результат неудовлетворительный, похожее решение неубедительно и следует искать другие прецеденты.

§ 2.6. Метод экспертов

Экспертон – это обобщение понятия вероятности случайного нечеткого события, когда вероятность случайного события каждого α -уровня заменяется на доверительный интервал. Эти интервалы, в свою очередь, статистически определяются группой экспертов.

Пусть E - множество определенных объектов, конечное или бесконечное. Группе из r экспертов предлагается высказать свое субъективное мнение относительно каждого элемента из E в виде интервала доверия.

$$\forall P \in E : [a_j^-(P), a_j^+(P)] \subset [0, 1], \quad \text{где } j \text{ – номер эксперта.}$$

Рассматривается статистика, когда каждому элементу $P \in E$ ставится в соответствие как нижняя, так и верхняя границы интервалов доверия. Кумулятивный закон распределения $F_*(\alpha, P)$ построен на основе $a_*^j(P)$, а $F^*(\alpha, P)$ – на основе $a_j^*(P)$.

Отсюда получается

$$\forall P \in E, \quad \forall \alpha \in [0,1]: \quad \tilde{A}(P) = [F_*(\alpha, P), F^*(\alpha, P)],$$

где \tilde{A} означает экспертон.

Приведем пример экспертона: допустим, 10-ти экспертам предложено высказать свое мнение о некоторых свойствах P_1, P_2 . Вместо того, чтобы свои суждения выразить числом $\alpha \in [0,1]$, они оценивают свойства P_1 и P_2 интервалом доверия $[a_1, a_2] \subset [0,1]$. Рассмотрим 11 α -уровней от 0 до 1. Оценки каждого эксперта в виде интервала доверия сведены в таблицу

Эксперт	P_1	P_2
1	[0.2, 0.3]	[0.6, 0.7]
2	[0.5, 0.6]	[0.5, 0.6]
3	[0.1, 0.7]	[0.8, 0.9]
4	[0.3, 0.4]	1
5	0.6	[0.6, 0.8]
6	[0.8, 1]	[0.2, 0.3]
7	[0.4, 0.8]	[0, 0.1]
8	[0.4, 0.5]	1
9	[0, 0.1]	[0.9, 1]
10	[0.2, 0.4]	[0.4, 0.6]

Для каждого свойства $P_i, i=1,2$ вычислим две статистики каждого уровня: одну для нижней границы интервала, другую – для верхней. Если взять эти статистики на множестве уровней $\{0, 0.1, 0.2, \dots, 0.9, 1\}$, получим таблицу, которая и представляет собой экспертон:

Уровень	P_1	P_2
0	1	1
0.1	[0.9, 1]	[0.9, 1]
0.2	[0.8, 1]	0.9
0.3	[0.6, 0.9]	[0.8, 0.9]
0.4	[0.5, 0.8]	0.8
0.5	[0.3, 0.6]	[0.7, 0.8]
0.6	[0.2, 0.3]	[0.6, 0.8]
0.7	[0.1, 0.3]	[0.4, 0.6]
0.8	[0.1, 0.2]	[0.4, 0.5]
0.9	[0, 0.1]	[0.3, 0.4]
1	[0, 0.1]	[0.2, 0.3]

В теории экспертонов доказывається [22,31,32], что экспертн удовлетворяет тем же алгебраическим свойствам, что и вероятность, если соблюдены специальные правила монотонности. Монотонность здесь подразумевается в смысле интервальной монотонности. Проводятся следующие преобразования экспертона \tilde{A} :

- экспертн приводится к вероятностному множеству взятием среднего арифметического границ интервала;
- вероятностное множество приводится к нечеткому множеству с помощью математического ожидания.
- при необходимости находится ближайшее к нечеткому четкое множество.

В случае нашего экспертона соответствующее вероятностное множество:

P_1	P_2
1	1
0.95	0.95
0.85	0.90
0.75	0.85
0.65	0.80
0.45	0.75
0.35	0.70
0.20	0.50
0.15	0.45
0.05	0.35
0.05	0.25

Для каждого P_i после вычисления математического ожидания получаем нечеткое множество

P_1	P_2
0.495	0.682

Ближайшим же четким множеством будет

P_1	P_2
0	1

Это означает, что соответствуясь с общим мнением экспертов эксперт отдает предпочтение свойству P_2 . Вместо нахождения ближайшего к нечеткому четкого множества можно было упорядочить данные свойства по возрастанию, чтобы установить их приоритет.

Итак, если нечеткое множество E содержит i элементов, тогда в результате приведенных выше преобразований эксперта каждому P_i будет соответствовать определенное число, которое представляет собой значение доверия, установленное с учетом общего мнения экспертов. Для получения единственного решения применяется принцип максимума:

$$\delta(P_i) = \max_i(P_i).$$

В условиях, когда по поводу одного и того же прогноза было принято несколько решений, например, разными методами, для вывода единственного решения возможно использование метода экспертонов. Пусть нечеткое множество всех решений E' , полученное в результате применения метода экспертонов, состоит из k элементов: $E' = \{\delta_1, \delta_2, \dots, \delta_k\}$. Тогда, исходя из метода экспертонов, можно найти единственное самое достоверное решение:

$$\delta^{Class} = \max_j(\delta_j), \quad j = \overline{1, k}, \quad \text{где } \delta_j - j\text{-тое возможное решение.}$$

Глава III.

Описание нечеткой компьютерной системы принятия решений

§ 3.1. Модель нечеткой компьютерной системы и ее основные элементы

Нечеткая компьютерная система, основанная на таблично-цифровой базе знаний, должна удовлетворять следующим свойствам:

- в системе должны быть предусмотрены средства обращения к базе данных общего назначения.
- система должна уметь обрабатывать данные из базы данных общего назначения для создания базы знаний, т.е. в ней должны быть реализованы соответствующие алгоритмы.
- в системе должна быть предусмотрена возможность размещения обработанной информации в базе знаний.
- в системе должен быть предусмотрен механизм получения решения на основании новой информации, т.е. в ней должны быть реализованы алгоритмы выбора информации из базы знаний и принятия решения.
- система должна обладать средствами вывода принятого решения, т.е. его визуализации.

Работу компьютерной системы можно разделить на следующие этапы:

- получение информации из базы данных общего назначения, обработка этой информации, создание и заполнение базы знаний.
- выбор единственного решения из всех возможных на основе вновь поступившей информации и существующей информации базы знаний.
- вывод информации о полученном решении в удобной для пользователя форме.

Приведем модель основанной на таблично-цифровой базе знаний нечеткой компьютерной системы

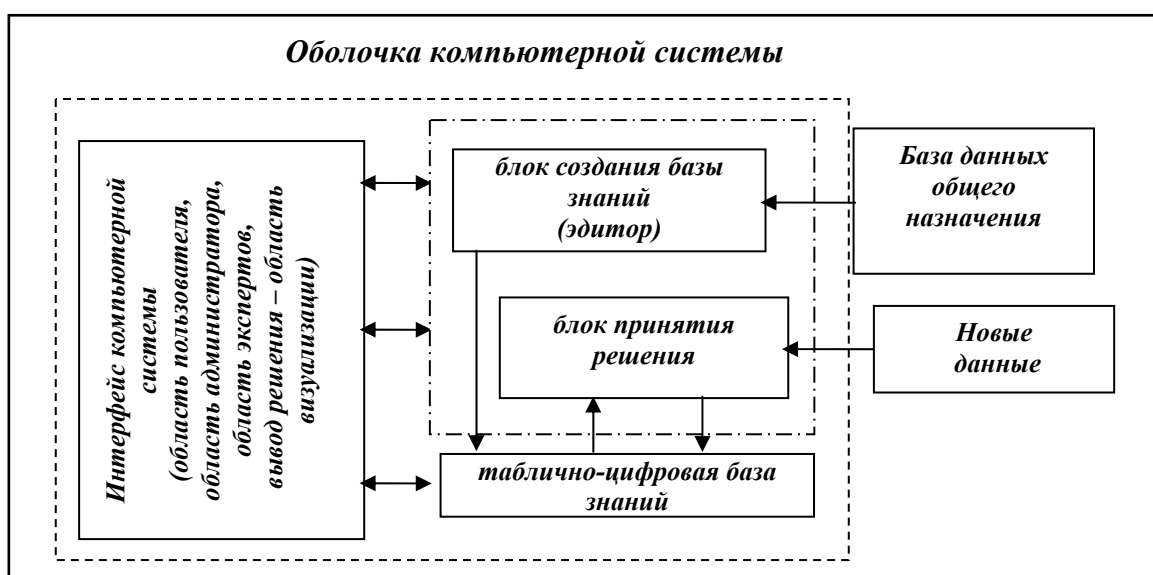


Схема компьютерной системы, основанной на таблично-цифровой базе знаний

Эдитор базы знаний ответственен за выбор информации из базы данных общего назначения, ее обработку и размещение в базе знаний. Поэтому он взаимодействует с обеими базами.

В блоке принятия решения на основании новых данных и информации, размещенной в базе знаний, происходит принятие окончательного решения. Поэтому он взаимодействует с базой знаний и интерфейсом пользователя. На схеме не выделен отдельно блок визуализации решения (посредством которого и происходит отображение принятого решения в интерфейсе пользователя). с которым также должен взаимодействовать блок принятия решения.

В компьютерной системе вывод решения обеспечивают нечеткие статистические методы, процесс принятия решения которых, а также выбор СУБД, predeterminedили создание базы знаний самой компьютерной системой. Поэтому таблично-цифровая база знаний является составной частью оболочки компьютерной системы, что представляет собой новшество по сравнению с подобными моделями.

§ 3.2. Нечеткие статистические методы для конкретной задачи прогноза.

Для конкретной задачи принятия решения о возможности землетрясения в качестве факторов-предвестников рассматриваются некоторые геофизические данные атмосферы: 1). величина напряженности электрического поля (вольт/м); 2). температура воздуха (в градусах Цельсия); 3). температура почвы (в градусах Цельсия); 4). атмосферное давление (в мб); 5). абсолютная влажность (упругость водяного пара в мб); 6). относительная влажность (в %); 7). общая облачность (в баллах); 8). нижняя облачность (в баллах); 9). скорость ветра (в м/сек.).

Значения величин активностей замерены в течение дня с промежутком в 3 часа: 0^{00} , 3^{00} , 6^{00} , 9^{00} , 12^{00} , 15^{00} , 18^{00} , 21^{00} . Величины активностей для известных случаев землетрясения находятся в базе данных общего назначения.

3.2.1. Статистика нечетких классов

Объект прогнозирования – землетрясение – делится на 3 класса классификации (M_0, M_1, M_2) со значениями нечетких переменных «шум», «среднее» и «сильное» и характеризуется численным значением величины магнитуды (M). При $0 \leq M \leq 3$ наблюдается «шум», при $3 < M < 5$ имеем «среднее землетрясение», при $5 \leq M \leq 8$ – «сильное землетрясение».

Модель функции совместимости, применяемая в данном методе, является модификацией модели Заде.

$$\mu_0(M) = \begin{cases} \frac{1}{1 + (\alpha_1 M)^2}, & 0 \leq M \leq 3; \\ 0, & M > 3 \end{cases} \quad \mu_1(M) = \begin{cases} 0, & M < 4.4; \\ \frac{1}{1 + (\alpha_2 (M - 4.9))^2}, & 4.4 \leq M \leq 5.4; \\ 0, & M > 5.4 \end{cases}$$

$$\mu_2(M) = \begin{cases} 0, & M < 4.4; \\ \frac{1}{1 + (\alpha_3 (M - 8))^2}, & 4.4 \leq M \leq 8; \\ 1, & M > 8 \end{cases} ;$$

Коэффициенты α_1 , α_2 и α_3 подбираются эмпирическим путем в соответствии с имеющимися данными и рекомендациями экспертов.

В нашем случае функции совместимости имеют следующий вид:

$$\mu_0(M) = \begin{cases} \frac{1}{1 + (0.15M)^2}, & 0 \leq M \leq 3; \\ 0, & M > 3 \end{cases} \quad \mu_1(M) = \begin{cases} 0, & M < 4.4; \\ \frac{1}{1 + (5.05(M - 4.9))^2}, & 4.4 \leq M \leq 5.4; \\ 0, & M > 5.4 \end{cases}$$

$$\mu_2(M) = \begin{cases} 0, & M < 4.4; \\ \frac{1}{1 + (0.5(M - 8))^2}, & 4.4 \leq M \leq 8; \\ 1, & M > 8 \end{cases}$$

Так как прогнозируемые классы представлены в виде интервалов, необходимо усреднить функции совместимости по этим интервалам. Пусть μ_i^j есть усредненное значение μ_j с учетом пересечения суппорта i -того прогнозируемого нечеткого класса и $\text{supp } \mu_j$:

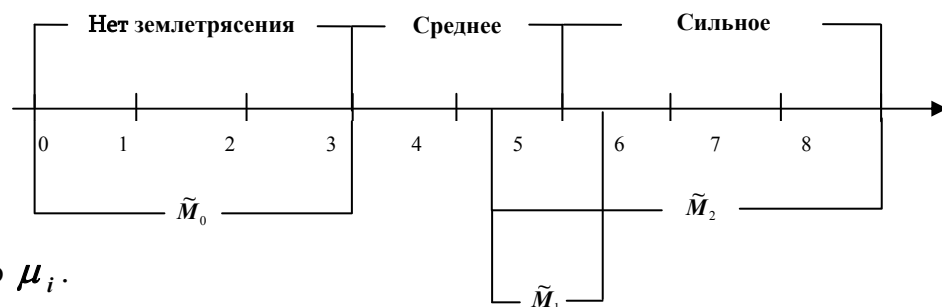
$$\mu_0^0 = \frac{1}{3} \int_0^3 \frac{dM}{1 + (0.15M)^2} \approx 0.93968; \quad \mu_1^0 = 0;$$

$$\mu_0^1 = 0; \quad \mu_1^1 = \frac{1}{5 - 4.4} \int_{4.4}^{5.0} \frac{dM}{1 + (4.99(M - 4.9))^2} \approx 0.55192;$$

$$\mu_2^1 = \frac{1}{0.4} \int_5^{5.4} \frac{dM}{1 + (4.99(M - 4.9))^2} \approx 0.3641;$$

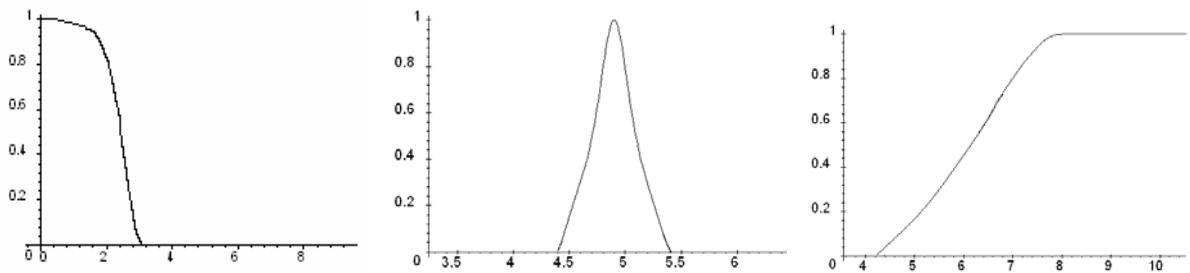
$$\mu_1^2 = \frac{1}{0.6} \int_{4.4}^5 \frac{dM}{1 + (0.5(M - 8))^2} \approx 0.26968; \quad \mu_2^2 = \frac{1}{3} \int_5^8 \frac{dM}{1 + (0.5(M - 8))^2} \approx 0.6552.$$

Приведем схему перекрытия интервалов прогнозируемых классов и суппортов функций совместимости:



где $\tilde{M}_i = \text{supp } \mu_i$.

Графики соответствующих функций совместимости имеют вид



Как видно из схемы перекрытия, считаем, что случаи «шумов» не влияют на «среднее», и наоборот.

Каждый из факторов-предвестников делится на три класса (подфактора). Интервалы классов факторов – нечеткие множества, границы их подбираются эмпирическим путем, согласуясь с имеющимися данными и знаниями экспертов.

Таблица базы знаний создается по правилам типа «Если ..., То ...», которые позволяют подсчитать четкие выборочные частоты n_{kj}^i . Например, «Если значение НЭП ≤ 5 , То число «шумов»=2, число «средних»=0, число «сильных»=1». Таким образом подсчитывается количество случаев «шумов», «средних» и «сильных» из имеющихся в наличии случаев землетрясений, для которых величина активности - напряженность электрического поля - имеет значение меньше или равное пяти. Затем при помощи функций совместимости и четких выборочных частот вычисляются нечеткие выборочные частоты \tilde{n}_{kj}^i . Например,

$$\tilde{n}_{00}^0 = \mu_0^0 \cdot n_{00}^0 + \mu_1^0 \cdot n_{00}^1; \quad \tilde{n}_{00}^1 = \mu_0^1 \cdot n_{00}^0 + \mu_1^1 \cdot n_{00}^1 + \mu_2^1 \cdot n_{00}^2; \quad \tilde{n}_{00}^2 = \mu_1^2 \cdot n_{00}^1 + \mu_2^2 \cdot n_{00}^2.$$

Значения нечетких весов вычисляются как отношение суммы нечетких частот подфактора к сумме нечетких частот всех трех подфакторов. Например,

$$w_{00} = (\tilde{n}_{00}^0 + \tilde{n}_{00}^1 + \tilde{n}_{00}^2) / \sum_{i=0}^2 (\tilde{n}_{0i}^0 + \tilde{n}_{0i}^1 + \tilde{n}_{0i}^2).$$

Нечеткие же относительные частоты – как отношение каждой нечеткой частоты каждого подфактора к сумме всех нечетких частот этого подфактора. Например,

$$\tilde{f}_{00}^0 = \tilde{n}_{00}^0 / (\tilde{n}_{00}^0 + \tilde{n}_{00}^1 + \tilde{n}_{00}^2); \quad \tilde{f}_{00}^1 = \tilde{n}_{00}^1 / (\tilde{n}_{00}^0 + \tilde{n}_{00}^1 + \tilde{n}_{00}^2);$$

$$\tilde{f}_{00}^2 = \tilde{n}_{00}^2 / (\tilde{n}_{00}^0 + \tilde{n}_{00}^1 + \tilde{n}_{00}^2).$$

Введем обозначения для каждой активности и каждого ее класса

X_0 - напряженность электрического поля: x_{00} - НЭП ≤ 5 , x_{01} - НЭП $\in (5, 9]$, x_{02} - НЭП > 9 ; X_1 - температура воздуха: x_{10} - ТВ < 4.3 , x_{11} - ТВ $\in [4.3, 9)$, x_{12} - ТВ ≥ 9 ; X_2 - температура почвы: x_{20} - ТП ≤ 2 , x_{21} - ТП $\in (2, 11)$, x_{22} - ТП ≥ 11 ;

X_3 - атмосферное давление: x_{30} - Атм.д ≤ 911.0 , x_{31} - Атм.д $\in (911, 916.4)$, x_{32} - Атм.д ≥ 916.4 ; X_4 - влажность абсолютная: x_{40} - Вла.абс ≤ 5 , x_{41} - Вла.абс $\in (5, 9]$, x_{42} - Вла.абс > 9 ; X_5 - влажность относительная: x_{50} - Вла.отн < 60 , x_{51} - Вла. отн $\in [60, 81]$, x_{52} - Вла. отн > 81 ; X_6 - облачность общая: x_{60} - Обл.общ. < 6 , x_{61} - Обл.общ. $\in [6, 8]$, x_{62} - Обл.общ. > 8 ;

X_7 - облачность нижняя: x_{70} - Обл.н ≤ 3 , x_{71} - Обл.н $\in (3, 9)$, x_{72} - Обл.н ≥ 9 ;

X_8 - скорость ветра: x_{80} - Ск.в ≤ 1 , x_{81} - Ск.в $\in (1, 2]$, x_{82} - Ск.в > 2 .

Для имеющихся данных получаем таблицу базы знаний:

Фактор		n_{kj}^0	n_{kj}^1	n_{kj}^2	\tilde{n}_{kj}^0	\tilde{n}_{kj}^1	\tilde{n}_{kj}^2	$\sum_i \tilde{n}_{kj}^i$	w_{kj}	\tilde{f}_{kj}^0	\tilde{f}_{kj}^1	\tilde{f}_{kj}^2
X_0	x_{00}	2	0	1	1.8794	0.3641	0.6552	2.8987	0.1489	0.6484	0.1256	0.2260
	x_{01}	5	5	4	4.6984	4.216	3.9692	12.884	0.6619	0.3647	0.3272	0.3081
	x_{02}	0	2	2	0	1.8320	1.8498	3.6818	0.1892	0	0.4976	0.5024
X_1	x_{10}	3	3	2	2.8190	2.3840	2.1194	7.3224	0.3762	0.3850	0.3256	0.2894
	x_{11}	1	1	3	0.9397	1.6442	2.2353	4.8192	0.2476	0.1950	0.3412	0.4638
	x_{12}	3	3	2	2.8190	2.3840	2.1194	7.3224	0.3762	0.3850	0.3256	0.2894
X_2	x_{20}	3	2	3	2.8190	2.1961	2.5050	7.5201	0.3864	0.3749	0.2920	0.3331
	x_{21}	2	2	3	1.8794	2.1961	2.5050	6.5805	0.3381	0.2856	0.3337	0.3807

X ₃	x ₂₂	2	3	1	1.8794	2.0199	1.4642	5.3635	0.2756	0.3504	0.3766	0.2730
	x ₃₀	1	2	2	0.9397	1.8320	1.8498	4.6215	0.2374	0.2033	0.3964	0.4003
	x ₃₁	3	5	3	2.8190	3.8519	3.314	9.9849	0.5130	0.2823	0.3858	0.3319
X ₄	x ₃₂	3	0	2	2.8190	0.7282	1.3104	4.8576	0.2496	0.5803	0.1499	0.2698
	x ₄₀	3	2	1	2.8190	1.4680	1.1946	5.4815	0.2816	0.5143	0.2678	0.2179
	x ₄₁	2	4	4	1.8794	3.6641	3.6995	9.2430	0.4749	0.2033	0.3964	0.4003
X ₅	x ₄₂	2	1	2	1.8794	1.2801	1.5801	4.7396	0.2435	0.3965	0.2701	0.3334
	x ₅₀	2	2	1	1.8794	1.4679	1.1946	4.5419	0.2333	0.4138	0.3232	0.2630
	x ₅₁	4	3	5	3.7587	3.4763	4.0850	11.32	0.5816	0.3320	0.3071	0.3609
X ₆	x ₅₂	1	2	1	0.9397	1.4679	1.1946	3.6022	0.1851	0.2609	0.4075	0.3316
	x ₆₀	4	2	3	3.7587	2.1961	2.5050	8.4598	0.4346	0.4443	0.2596	0.2961
	x ₆₁	2	4	2	1.8794	2.9359	2.3891	7.2044	0.3701	0.2609	0.4075	0.3316
X ₇	x ₆₂	1	1	2	0.9397	1.2801	1.5801	3.7999	0.1952	0.2473	0.3369	0.4158
	x ₇₀	2	3	2	1.8794	2.3840	2.1194	6.3828	0.3279	0.2944	0.3735	0.3321
	x ₇₁	4	4	4	3.7587	3.6641	3.6995	11.1223	0.5714	0.3379	0.3294	0.3326

	x_{72}	1	0	1	0.9397	0.3641	0.6552	1.9590	0.1006	0.4797	0.1859	0.3345
X_8	x_{80}	1	4	4	0.9397	3.6641	3.6995	8.3033	0.4266	0.1132	0.4413	0.4455
	x_{81}	5	2	2	4.6984	1.8320	1.8498	8.3802	0.4305	0.5607	0.2186	0.2207
	x_{82}	1	1	1	0.9397	0.9160	0.9249	2.7806	0.1429	0.3379	0.3294	0.3326

Пусть необходимо исследовать новый случай, описывающие его активности 6.708 21.175 30.125 914.99 20.96 85.5 7.125 6 1. Этому набору активностей соответствуют строки таблицы $x_{01}, x_{12}, x_{22}, x_{31}, x_{42}, x_{52}, x_{61}, x_{71}, x_{80}$. Тогда вектор

нечетких

весов

$$\vec{w} = (0.6619, 0.3762, 0.2756, 0.5130, 0.2435, 0.1851, 0.3701, 0.5714, 0.4266), \quad a$$

матрица нечетких относительных весов

$$\tilde{f} = \begin{pmatrix} 0.3647 & 0.3272 & 0.3081 \\ 0.3850 & 0.3256 & 0.2894 \\ 0.3504 & 0.3766 & 0.2730 \\ 0.2823 & 0.3858 & 0.3319 \\ 0.3965 & 0.2701 & 0.3334 \\ 0.2609 & 0.4075 & 0.3316 \\ 0.2609 & 0.4075 & 0.3316 \\ 0.3379 & 0.3294 & 0.3326 \\ 0.1132 & 0.4413 & 0.4455 \end{pmatrix}$$

Проведя линейный многофакторный синтез, получим взвешенный вектор возможных решений, которому соответствует мера возможности:

$$\vec{D} = \vec{w} \cdot \tilde{f} = (1.11039, 1.30928, 1.20374);$$

$$\overline{Poss} = \frac{\vec{D}}{\max_j D_j} = (0.84809, 1, 0.91939), \text{ где } D_j - j\text{-тая компонента вектора } \vec{D}.$$

Вспользуемся принципом максимума возможностей и, окончательно, получим прогноз

$$D_{Class} = 1 \quad (\Rightarrow M_1 \equiv \text{Среднее землетрясение}).$$

В ходе решения задачи прогноза для построения образов дней подготовки землетрясения (за 2 дня, за 1 день до землетрясения), была также использована трапециадальная модель функции согласованности для интервала интенсивности «среднее»:

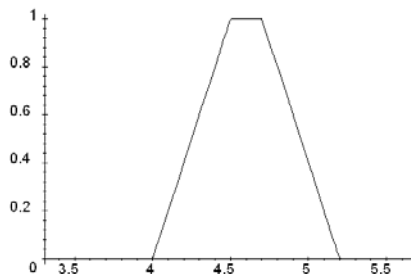
$$\mu_1(M) = \begin{cases} 0, & M < 4, \\ 2M - 8, & M \in [4; 4.5], \\ 1, & M \in (4.5; 4.7], \\ -2M + 10.4, & M \in (4.7; 5.2], \\ 0, & M > 5.2. \end{cases}$$

соответствующие усредненные по интервалам значения которой $\mu_0^1 = 0$;

$$\mu_1^1 = \int_4^5 \mu_1(M) dM = \int_4^{4.5} (2M - 8) dM + \int_{4.5}^{4.7} dM + \int_{4.7}^5 (-2M + 10.4) dM = 0.66;$$

$$\mu_2^1 = \frac{1}{0.2} \int_5^{5.2} (-2M + 10.4) dM = 0.2,$$

а график



3.2.2. Дискриминационный анализ

Все базовые случаи землетрясений по всем атрибутам, как и в случае статистики нечетких классов, делятся на 2 подфактора (класса):

X_0 - напряженность электрического поля: x_{00} - НЭП < 6.5, x_{01} - НЭП ≥ 6.5; X_1 - температура воздуха: x_{10} - ТВ < 8.8, x_{11} - ТВ ≥ 8.8;

X_2 - температура почвы: x_{20} - ТП < 11, x_{21} - ТП ≥ 11;

X_3 - атмосферное давление: x_{30} - Атм.д < 915, x_{31} - Атм.д ≥ 915;

X_4 - влажность абсолютная: x_{40} - Вл.абс < 9, x_{41} - Вл.абс ≥ 9;

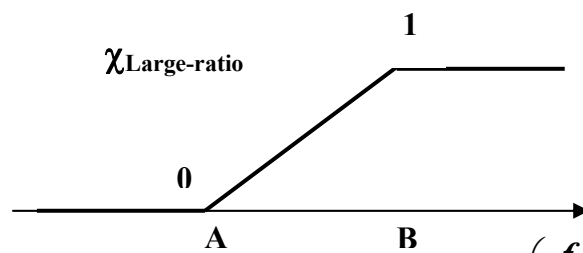
X_5 - влажность относительная: x_{50} - Вл.отн < 75, x_{51} - Вл.отн ≥ 75;

X_6 - облачность общая: x_{60} - Обл.общ. < 6, x_{61} - Обл.общ. ≥ 6;

X_7 - облачность нижняя: x_{70} - Обл.н < 3, x_{71} - Обл.н ≥ 3;

X_8 - скорость ветра: x_{80} - Ск.в ≤ 1.25, x_{81} - Ск.в > 1.25.

Подсчитываются относительные частоты каждого подфактора. Например, пусть для базовых случаев по первой активности – напряженность электрического поля – оказалось, что: 1). для «шумов» в 3-х случаях из 7-ми значение активности принадлежит первому подфактору, в 4-х же – второму; 2). для «средних» в 2-х случаях из 7-ми значение активности принадлежит I классу, в 5-ти - II классу; 3). для «сильных» - в 4-х случаях наблюдается принадлежность I классу, в 3-х - II классу. На основании этих подсчетов (производимых по правилам типа продукций «Если ..., То ...») получаем первую таблицу базы знаний $\{f_{ij}\}$. Затем по известным формулам строятся таблицы положительной и отрицательной дискриминаций, где функция принадлежности $\chi_{Large-ratio} : R^+ \rightarrow [0;1]$ имеет вид



В рассматриваемом случае $\chi_{Large-ratio} \left(\frac{f_{ij}}{f_{ik}} \right) = \frac{1}{3.75} \left(\frac{f_{ij}}{f_{ik}} \right)$. Приведем

полученную таблицу

Фактор		f_{ij}			p_{ij}			n_{ij}		
X_0	x_{00}	3	2	4	0.3	0.1556	0.4444	0.2667	0.4667	0.1667
	x_{01}	4	5	3	0.2844	0.3889	0.18	0.2667	0.1867	0.4
X_1	x_{10}	4	4	5	0.24	0.24	0.3333	0.3	0.3	0.2133

	x_{11}	3	3	2	0.3333	0.3333	0.1778	0.2222	0.2222	0.4
X_2	x_{20}	5	4	6	0.2778	0.1956	0.36	0.2667	0.3667	0.2
	x_{21}	2	3	1	0.3556	0.6	0.1111	0.2667	0.1333	0.6667
X_3	x_{30}	2	6	5	0.0978	0.56	0.4444	0.7333	0.1556	0.2133
	x_{31}	5	1	2	1	0.0933	0.32	0.08	0.9333	0.4
X_4	x_{40}	5	6	5	0.2444	0.32	0.2444	0.2933	0.2222	0.2933
	x_{41}	2	1	2	0.4	0.1333	0.4	0.2	0.5333	0.2
X_5	x_{50}	3	5	5	0.16	0.3556	0.3556	0.4444	0.2133	0.2133
	x_{51}	4	2	2	0.5333	0.2	0.2	0.1333	0.4	0.4
X_6	x_{60}	4	2	3	0.4444	0.1556	0.3	0.1667	0.4667	0.2667
	x_{61}	3	5	4	0.18	0.3889	0.2844	0.4	0.1867	0.2667
X_7	x_{70}	2	3	1	0.3556	0.6	0.1111	0.2667	0.1333	0.6667
	x_{71}	5	4	6	0.2778	0.1956	0.36	0.2667	0.3667	0.2
X_8	x_{80}	3	4	5	0.18	0.2844	0.3889	0.4	0.2667	0.1867
	x_{81}	4	3	2	0.4444	0.3	0.1556	0.1667	0.2667	0.4667

Пусть для конкретного прогнозируемого случая значения активностей 6.708 21.175 30.125 914.99 20.96 85.5 7.125 6 1. Этому набору активностей соответствуют строки таблицы $x_{01}, x_{11}, x_{21}, x_{30}, x_{41}, x_{51}, x_{61}, x_{71}, x_{80}$. Они выбираются из базы знаний и строятся новые таблицы $\{p'_{ij}\}$ и $\{n'_{ij}\}$.

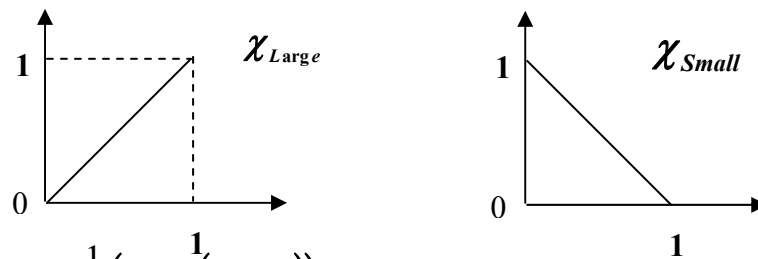
<i>Фактор</i>	p'_{ij}			n'_{ij}		
x_{01}	0.2844	0.3889	0.18	0.2667	0.1867	0.4
x_{11}	0.3333	0.3333	0.1778	0.2222	0.2222	0.4
x_{21}	0.3556	0.6	0.1111	0.2667	0.1333	0.6667
x_{30}	0.0978	0.56	0.4444	0.7333	0.1556	0.2133
x_{41}	0.4	0.1333	0.4	0.2	0.5333	0.2

x_{51}	0.5333	0.2	0.2	0.1333	0.4	0.4
x_{61}	0.18	0.3889	0.2844	0.4	0.1867	0.2667
x_{71}	0.2778	0.1956	0.36	0.2667	0.3667	0.2
x_{80}	0.18	0.2844	0.3889	0.4	0.2667	0.1867

После этого находят средние значения мер положительной π_j и отрицательной ν_j дискриминаций для j -того решения, а нечеткое множество возможных решений имеет функцию принадлежности, определяемую по формуле

$$\delta(D_j) = \frac{1}{2}(\chi_{Large}(\pi_j) + \chi_{Small}(\nu_j)), \quad j \in D,$$

где нечеткие множества *Large* и *Small* характеризуются функциями принадлежности $\chi_{Large}, \chi_{Small} : [0,1] \rightarrow [0,1]$, причем χ_{Large} – монотонно возрастающая функция, а χ_{Small} – монотонно убывающая. Например, можно взять $\chi_{Large}(\pi_j) = \pi_j$, а $\chi_{Small}(\nu_j) = -\nu_j + 1$:



Тогда $\delta_j = \frac{1}{2}(\pi_j + (1 - \nu_j))$, $j \in D$. Таблица для π_j , ν_j и δ_j выглядит

следующим образом

π_j	ν_j	δ_j
0.29358	0.320988	0.486296
0.342716	0.272346	0.535185
0.282963	0.325926	0.478518

Единственное, классическое решение получаем как $\delta^{Class} = \max_j \delta_j$.

В нашем случае $\delta^{Class} = 0.535185$, и это максимальное значение соответствует интервалу интенсивности «среднее». Таковым и будет прогноз.

3.2.3. Метод рассуждений на основе прецедентов

Из базы данных общего назначения выбираем значения активностей, характеризующие известные случаи «шум», «среднее» и «сильное» (для примера, по 7 случаев, как и в предыдущих методах). Все факторы (активности) каждого случая разбиваем на классы, как и в случае нечеткого дискриминационного анализа и с теми же границами интервалов классов факторов. Затем по правилам типа «Если ..., То ...» составляем новую таблицу (базы знаний), где на каждый класс каждой активности приходится значения 0 или 1. Например, если первая активность (напряженность электрического поля) одного из случаев НЭП=6.25, то, учитывая классы этого фактора ($x_{00} < 6.5$; $x_{01} \geq 6.5$), получим 2 значения в базе знаний: $x_{00} = 1$; $x_{01} = 0$.

Соответствующая таблица базы знаний

X_0		X_1		X_2		X_3		X_4		X_5		X_6		X_7		X_8	
x_{00}	x_{01}	x_{10}	x_{11}	x_{20}	x_{21}	x_{30}	x_{31}	x_{40}	x_{41}	x_{50}	x_{51}	x_{60}	x_{61}	x_{70}	x_{71}	x_{80}	x_{81}
0	1	1	0	1	0	0	1	1	0	0	1	0	1	0	1	1	0
1	0	1	0	1	0	0	1	1	0	0	1	1	0	1	0	0	1
1	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	1	1	0	1	0	1	0	1	0	1	0	0	1	0	1
1	0	1	0	1	0	0	1	1	0	0	1	0	1	0	1	1	0
0	1	1	0	1	0	0	1	1	0	1	0	1	0	0	1	0	1
0	1	0	1	0	1	1	0	0	1	1	0	1	0	1	0	1	0
0	1	1	0	1	0	1	0	1	0	1	0	0	1	0	1	1	0
0	1	0	1	0	1	1	0	0	1	0	1	1	0	1	0	1	0
0	1	1	0	1	0	0	1	1	0	1	0	0	1	1	0	0	1
0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	0	1
0	1	0	1	0	1	1	0	1	0	1	0	0	1	0	1	0	1
1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	1	0
1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	1	0

1	0	1	0	1	0	0	1	1	0	0	1	0	1	0	1	0	1
0	1	1	0	1	0	1	0	1	0	1	0	0	1	0	1	1	0
0	1	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0
1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	1	0
1	0	0	1	1	0	1	0	1	0	1	0	1	0	1	0	0	1
1	0	1	0	1	0	0	1	0	1	0	1	0	1	0	1	1	0
0	1	1	0	1	0	1	0	1	0	1	0	1	0	0	1	1	0

Здесь первые 7 строк соответствуют значению землетрясения «шум», вторые 7 - значению землетрясения «среднее», последние 7 - «сильное».

Пусть исследуемый случай описываются значениями параметров 6.25 18.5375 23.75 908.5875 13.9125 69.5 5.375 3.25 0.625. Тогда им ставится в соответствие набор 1 0 0 1 0 1 1 0 0 1 1 0 1 0 0 1 1 0. Если обозначить через CB_i - i -тый элемент строки базы знаний, а через ND_i - i -тый элемент вектора нового случая, то можно вычислить меру сходства между ними по формуле $DV_i = 1 - |CB_i - ND_i|$, где $i = \overline{1, 2n}$, а n - число активностей. В результате получаем таблицу, где в строке заголовка $i \leftrightarrow DV_i$,

i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	i=11	i=12	i=13	i=14	i=15	i=16	i=17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
1	1	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0
0	0	1	1	0	0	1	1	0	0	1	1	1	1	1	1	0	0
1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1
0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1	1	1
0	0	1	1	1	1	1	1	1	1	0	0	1	1	0	0	1	1
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0
1	1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	1	1
1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0

0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1	1	1
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	1	1	1
1	1	1	1	0	0	1	1	0	0	1	1	1	1	0	0	0	0
1	1	0	0	0	0	0	0	1	1	0	0	0	0	1	1	1	1
0	0	0	0	0	0	1	1	0	0	1	1	1	1	1	1	1	1

Затем находим меру сходства между двумя случаями, используя формулу

$$SV_j = \frac{1}{2n} \sum_{i=0}^{2n-1} w_{i/2} \cdot DV_i, \text{ где } SV_j - \text{ мера сходства } j\text{-того известного и нового случаев, а}$$

w_i - вес i -той активности, который указывает, насколько она важна, по мнению экспертов, для принятия решения. В нашем случае для простоты будем считать, что $\vec{w} = (1,1,1,1,1,1,1,1)$. Проведя необходимые вычисления, получаем

	M_0	SV_j	M_1	SV_j	M_2	SV_j
1		0.2222		0.4444		0.2222
2		0.2222		0.6667		0.4444
3		0.5556		0.1111		0.8889
4		0.5556		0.5556		0.5556
5		0.3333		0.5556		0.5556
6		0.3333		0.4444		0.4444
7		0.7778		0.5556		0.5556
max		0.7778		0.6667		0.8889

Для получения окончательного решения, находим $SV_r = \max(SV_j)$, что в нашем случае равно 0.8889, а значит принимается решение – возможно «сильное» землетрясение. Это соответствует историческим данным – 26.07.1989 в 18⁰⁰ произошло «сильное землетрясение» с магнитудой $M = 5.8$.

Во всех 3-х рассмотренных методах в качестве базовых случаев взяты данные геофизических активностей в день землетрясения по всем интервалам интенсивности. Для каждого интервала выбраны в качестве базовых по 7 известных случаев землетрясений: данные таковы

Напряж эл. поля	Темпер возд.	Темпер почвы	Атмосф. давлен.	Влажн. абсолют	Влажн. относит.	Облачн. общая	Облачн. нижняя	Скор. ветра
6.8333	-3.3875	-2.125	919.45	3.7963	80.5	8	6.875	0.875
4.8333	-2.7625	-2.125	915.35	3.6375	75	3.5	2.625	3.625
3.375	12.3875	12.375	916.26	13.4375	93.5	9.75	9.375	1.75
7.3333	10.2	9	912.45	7.2875	59.75	5.75	3.625	1.75
6.1667	8.6875	10.125	917.05	8.9	79.75	6.375	4.625	1.25
7.7083	0.3125	1.25	916.75	2.9125	45	3.375	3.125	1.75
7.375	25.2	31.125	910.05	22.185	70.25	3.125	2.125	1.125
9.95833	0.0875	3.625	911.225	4.25	73.25	6.125	4.875	0.875
7.2083	20.975	26.5	909.4875	20.0125	82.875	3.75	2.375	0.5
12.7917	1.7875	-1.625	916.35	3.3875	50.5	6.375	2.25	2.625
6.6666	15.725	22.25	912.9125	8.8	48.125	4.5	1.625	1.625
7.5833	13.6	15.375	913.975	8.2138	64.875	7.375	6.25	2
5.95833	2.3	1.375	906.425	5.94125	82.875	6.75	6.375	0.75
5.95833	4.4	5.125	914.4375	6.0175	74.75	8.25	6	0.875
5.4583	7.9	-0.25	919.875	3.9225	75	7.625	5	1.5
8.2917	8.7875	3.75	908.325	6.3375	56.25	6	5	0.75
14.20833	14.7	19.25	909.6125	11.6875	73.875	5.625	4.5	1
2.75	4.65	0.375	914.275	6.8063	70.875	9.75	9.5	0.5
5.125	9.675	8.875	913.98	8.12	69	3.25	2.625	6.125
5.125	7.5125	8.625	924.2375	9.60125	92.25	10	8.25	1.125
12.9166	4.0125	-0.125	913.75	5.5438	68.625	3	3	1

3.2.4. Метод экспертов

Так как решения, принятые тремя нечеткими статистическими методами, могут дать неидентичные результаты, необходимо прийти к единственному окончательному решению. Можно выработать экспертные рекомендации, например, в виде: 1). если

принятые всеми тремя методами решения совпадают, то в качестве окончательного прогноза выбирается это одинаковое решение; 2). если из 3-х прогнозов 2 совпадают, то в качестве окончательного прогноза выбирается одинаковое для 2-х решение. Хуже обстоит дело, если все 3 метода дают различные прогнозы, хотя и в случае совпадения 2-х или даже 3-х прогнозов ситуация неоднозначная: совпавшие прогнозы могут быть неверными. Тогда для получения единственного решения применим метод экспертонов.

На основании имеющихся в базе знаний экспертных оценок каждого из методов, заданных в виде доверительных интервалов, строим экспертон и принимаем за верное то решение, которое выработает экспертон, т.е. решение, учитывающее коллективное мнение группы экспертов. В группе 3 эксперта. Каждый из них задает оценки по каждому методу.

Например, для исследуемого случая, характеризующегося набором активностей 5.125 9.675 8.875 913.98 8.12 69 3.25 2.625 6.125:

1. метод статистики нечетких классов получил вектор возможных решений $\overrightarrow{Poss} = (0.96627, 1, 0.99829)$ и принял решение: прогноз \Rightarrow «среднее землетрясение»;

2. дискриминационный анализ получил нечеткое множество решений $\delta_j = (0.516931, 0.527778, 0.455291)$ и выработал прогноз \Rightarrow «среднее землетрясение»;

3. метод CBR получил множество решений $(0.7778, 0.7778, 0.6667)$ и дал прогноз \Rightarrow «нет землетрясения».

Как видим, получен неоднозначный результат и есть предпосылка для применения метода экспертонов.

Доверительные интервальные оценки экспертов даны в таблице:

Эксперт	Метод	M_0	M_1	M_2
1	1	[0.2, 0.3]	[0.4, 0.6]	[0.6, 0.7]
	2	[0.5, 0.6]	[0.6, 0.8]	[0.5, 0.6]
	3	[0.1, 0.7]	1	[0.7, 0.8]
2	1	[0.3, 0.4]	1	1
	2	0.6	[0.7, 1]	[0.6, 0.8]
	3	[0.8, 1]	[0.6, 1]	[0.2, 0.3]
3	1	[0.4, 0.8]	0.6	[0, 0.1]

	2	[0.4, 0.5]	[0.3, 0.7]	1
	3	[0.2, 0.4]	[0.2, 0.9]	[0.2, 0.9]

Для каждого интервала интенсивности M_i , $i = 0,1,2$ вычислим две статистики на множестве уровней $\{0, 0.1, 0.2, \dots, 0.9, 1\}$: одну для нижней границы интервала, другую – для верхней. Таким образом построенный экспертон имеет вид:

Уровень	M_0	M_1	M_2
0	[1, 1]	[1, 1]	[1, 1]
0.1	[1, 1]	[1, 1]	[0.9, 1]
0.2	[0.9, 1]	[1, 1]	[0.9, 0.9]
0.3	[0.7, 1]	[0.9, 1]	[0.7, 0.9]
0.4	[0.6, 0.9]	[0.8, 1]	[0.7, 0.8]
0.5	[0.3, 0.7]	[0.7, 1]	[0.7, 0.8]
0.6	[0.2, 0.6]	[0.7, 1]	[0.6, 0.8]
0.7	[0.1, 0.3]	[0.3, 0.8]	[0.3, 0.7]
0.8	[0.1, 0.2]	[0.2, 0.7]	[0.2, 0.6]
0.9	[0, 0.1]	[0.2, 0.6]	[0.2, 0.3]
1	[0, 0.1]	[0.2, 0.4]	[0.2, 0.2]

Приведем экспертон к вероятностному множеству взятием среднего арифметического границ интервала и получим следующую таблицу:

Уровень	M_0	M_1	M_2
0	1	1	1
0.1	1	1	0.95
0.2	0.95	1	0.9
0.3	0.85	0.95	0.8
0.4	0.75	0.9	0.75
0.5	0.50	0.85	0.75
0.6	0.4	0.85	0.7
0.7	0.2	0.55	0.5
0.8	0.15	0.45	0.4
0.9	0.05	0.4	0.25
1	0.05	0.3	0.2

Для каждого M_i после вычисления среднего получаем нечеткое множество

M_0	M_1	M_2
0.536364	0.7500	0.654545

Ближайшее же четкое множество будет

M_0	M_1	M_2
1	1	1

Так как нахождение ближайшего к нечеткому четкого множества в данном случае не дает однозначного результата, установим приоритет интервалов интенсивности, упорядочив полученные элементы нечеткого множества по возрастанию.

0.7500 0.654545 0.536364

Получили, что соответствуясь с общим мнением экспертов, эксперт отдает предпочтение интервалу интенсивности M_1 , т.е. получили прогноз для данного исследуемого случая – возможно «среднее землетрясение».

§ 3.3. Общее описание программного продукта

В последние два десятилетия компьютерные системы получили широкое распространение как средство для тиражирования и доставки квалифицированных узкоспециализированных знаний в то место, где они необходимы для принятия решений.

Данный параграф рассматривает архитектуру компьютерной системы и описывает программный продукт, разработанный для размещения и использования компьютерной системы в WWW.

Представленный программный продукт располагается на одном из серверов локальной/глобальной сети, ядро оболочки реализовано на языке PHP, применяет скриптовый язык JavaScript, использует технологию клиент/сервер, база данных компьютерной системы реализована в СУБД (система управления базами данных) MySQL [82].

Одним из ведущих факторов выбора MySQL как СУБД для созданного программного продукта является необходимость обеспечить доступ к данным через Internet с рабочих станций, на которых, вполне возможно установлены, различные Операционные Системы.

MySQL – это одна из самых популярных и самых распространенных СУБД в интернете, отличающаяся хорошей скоростью работы, надежностью, гибкостью и

дающая возможность создавать реляционные базы данных и работать с ними. Характерной особенностью реляционной базы данных является структура, выполненная в виде таблиц. Строки таких таблиц соответствуют записям, столбцы - атрибутам (признакам хранимых данных). Такие данные являются ядром реляционной базы. MySQL основана на технологии client/server, которая актуальна главным образом потому, что обеспечивает простое и относительно дешевое решение проблемы коллективного доступа к базе данных в локальной/глобальной сети. Технология «клиент-сервер» применительно к СУБД сводится к разделению системы на две части - приложение-клиент и сервер базы данных, которые могут располагаться на одном компьютерном узле.

Использование реляционных баз данных позволяет:

- собирать и хранить данные в виде таблиц;
- обновлять их содержание;
- получать разнообразную информацию по атрибутам или записям;
- отображать полученные данные в виде диаграмм или таблиц;
- выполнять необходимые расчеты по материалам базы.

Немаловажным условием выбора MySQL, также является реализация в данной СУБД механизма нечетких запросов (fuzzy queries, flexible queries), базирующихся на теории нечетких множеств Заде. Большая часть данных, обрабатываемых в современных информационных системах, носит четкий, числовой характер. Однако в нашем случае, определяющую роль играет семантическая неоднозначность. Ввиду того, что предложенная компьютерная система осуществляет выбор информации из базы данных с использованием качественных критериев и нечетко сформулированных условий, инструмент нечетких запросов позволяет согласовать формальные критерии и неформальные требования к кругу факторов и задавать интервалы выбора потенциальных значений как нечеткие множества.

Например, представление нечеткого понятия «землетрясение» в базе знаний возможно в виде таблицы, состоящей из трех полей и трех столбцов, которые описывают интервалы интенсивности магнитуды.

Ключ magnitude_id	Название Name	Начальное значение Init_value	Конечное значение Upper_value
1	Шум	0	< 3
2	Среднее	>=3	<=5
3	Сильное	>5	<=8

Данный пример существенно упрощает реальное представление нечеткого понятия, сохранен лишь смысл.

Дополнительно, можно отметить следующие положительные стороны пакета MySQL: многопоточность, поддержка нескольких одновременных запросов, оптимизация связей с присоединением многих данных за один проход, записи фиксированной и переменной длины, гибкая система привилегий и паролей, до 16 ключей в таблице, каждый ключ может иметь до 15 полей. Также есть поддержка ключевых полей, поддержка чисел длиной от 1 до 4, строк переменной длины и меток времени, интерфейс с языками PHP, Perl, C. Основанная на потоках, быстрая система памяти, утилита проверки и ремонта таблицы, все данные хранятся в формате ISO8859_1. Все операции работы со строками не обращают внимания на регистр символов в обрабатываемых строках, псевдонимы применимы как к таблицам, так и к отдельным колонкам в таблице, все поля имеют значение по умолчанию. INSERT можно использовать на любом подмножестве полей. Легкость управления таблицей, включая добавление и удаление ключей и полей.

Ядро программного продукта реализовано на основе технологии PHP [77]. Соединение с базой данных и осуществление запроса является для PHP простой задачей, которую можно выполнить в двух-трех строках программы. Процессор сценариев PHP хорошо оптимизирован для времени отклика, необходимого веб-приложениям. Еще одно положительное свойство PHP в том, что код выполняется на сервере что, в свою очередь обеспечивает защиту программного пакета от неавторизованного изменения. PHP может использоваться на всех крупных операционных системах, включая Linux, многие варианты Unix (HP-UX, Solaris и OpenBSD), Microsoft Windows, Mac OS X, RISC OS и, возможно, другие. PHP имеет поддержку для большинства существующих web-серверов. Это Apache, Microsoft Internet Information Server, Personal Web Server, Netscape и iPlanet-серверы, O'reilly Website Pro, Caudium, Xitami, OmniHTTPd и многие другие. Для большинства этих

серверов PHP имеет стандартный пакет модулей. В других, поддерживающих стандарт CGI, PHP может работать как CGI-процессор.

Выходная информация – результат прогнозирования – представляется программным продуктом в виде сообщения. Возможна форма вывода результата прогноза названием интервала интенсивности прогнозируемого объекта (выводом названия нечеткой лингвистической переменной) и анимации на основе технологии Flash [78]. Применение Macromedia Flash избавляет от проблемы совместимости между браузерами: Flash одинаково хорошо работает как в Internet Explorer, так и в Netscape Navigator. В Macromedia Flash 5 применяется специальный событийно-управляемый язык, который поддерживает условные переходы, циклы, массивы, функции и классы. В нем также осуществлен механизм наследования.

Интеграция Flash с базой данных и серверной частью происходит посредством модулей, реализованных на языке программирования Perl. Файл Flash обеспечивает прием информации от сценария Perl. В свою очередь, Perl-сценарии создают сокет и устанавливают соединение для приема информации от базы данных запущенной из под Apache сервера.

§ 3.4. Описание основных каталогов

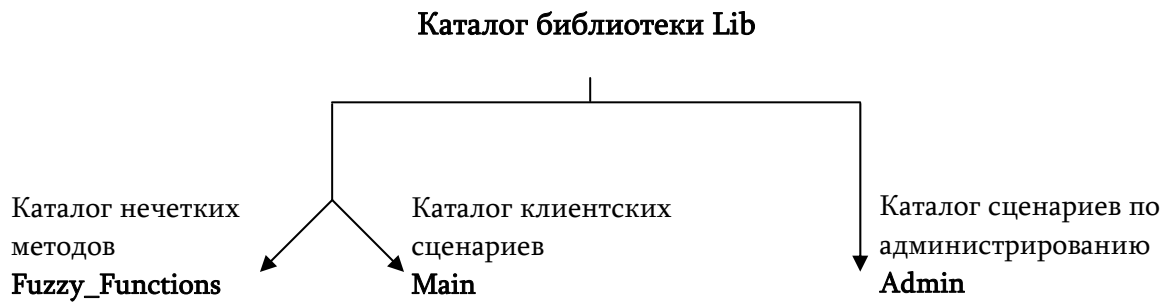
В силу особенностей реализации языка PHP его оболочка состоит из множества мелких модулей (файлов). В нашем случае модули объединены в 4 основных каталога:

- Каталог библиотеки **Lib**,
- Каталог нечетких методов **Fuzzy_Functions**,
- Каталог клиентских сценариев **Main**,
- Каталог сценариев по администрированию **Admin**.

Нужно отметить, что объединенные во всех трех каталогах Admin, Fuzzy_Functions и Main скрипты используют модули библиотеки Lib, которые представляют собой набор сценариев взаимодействия с MySQL: сценарии извлечения данных из базы, их обработки и помещения в базу. Кроме того, из каталога Lib применяется набор часто используемых т.н. генерализованных функций.

Каталоги могут содержать подкаталоги, в которых объединены модули по их назначению. Модули, в свою очередь, представляют собой объединение функций.

Логическое распределение каталогов можно представить в виде схемы:



Таким образом, каталог Lib находится на верхушке иерархии всех каталогов, включенных в программный продукт, так как практически все модули объединенные в остальные каталоги, обращаются к функциям из сценариев данного каталога.

Каталоги Fuzzy_Functions и Main находятся на одной ветви, так как функционально модули из этих каталогов служат выполнению одной задачи - получению возможного решения.

Скрипты из каталога Fuzzy_Functions являют собой программную реализацию математических методов прогнозирования. Так как предложенная, на примере прогнозирования землетрясений, математическая модель включает четыре метода, то и каталог, соответственно, состоит из четырех модулей обработки данных, на базе чего происходит принятие решения. В свою очередь, каталог Main объединяет сценарии ввода прогнозируемых данных и вывода полученного решения, другими словами, модули каталога Main обеспечивают диалог с пользователем.

Модули из каталог Admin используются с целью обновления и преобразования базы знаний. Администратор системы – пользователь с наивысшим уровнем доступа, посредством задействования сценариев из данного каталога, имеет возможность не только обновлять базу знаний новыми значениями, но и менять ее структуру адаптируя базу знаний под новые задачи.

Тут же следует отметить, что реализация данного каталога обосновывает определение разных уровней доступа. В программном продукте реализованы три таких уровня, упорядоченных по возрастанию. Каждый последующий уровень автоматически включает в себя возможности предыдущего:

- уровень «чтения», позволяет наблюдать за прогнозами, использовать поиск с целью нахождения данных и полученного решения по датам и/или по результатам прогноза.
- уровень пользователя, позволяет вводить данные и получать решения на основе вычислений, проведенных по вводным данным.
- администраторский уровень, позволяет работать с базой данных.

При входе в систему с уровнем доступа, допускающим не только считывание, но и внесение данных/изменение структуры базы (пользовательский или администраторский уровни доступа), кроме имени пользователя и пароля, происходит также проверка на соответствие входящего в систему IP адреса с заранее определенным рядом допустимых IP адресов. Данное ограничение, наряду с самим принципом реализация пакета (т.е. конфигурация сервера, установок MySQL и т.д.), повышает степень сохранности данных и увеличивает защищенность системы от несанкционированного вмешательства, что является одним из наиболее важных факторов оценки надежности системы, реализованной в сети.

§ 3.5. Каталог Lib - Библиотечный Каталог

Каталог Lib состоит из двух основных базовых модулей:

- Модуль взаимодействия с системой управления базами данных **Cl.php**,
- Модуль генерализованных функций **Gen_Functions.php**.

Модуль Cl.php обеспечивает взаимодействие с MySQL, в частности, открытие сокетов, установку параметров соединения, соединение с базой, извлечение данных и их помещение в базу, а также вывод сообщений об ошибках, которые произошли во время выполнения тех или иных операций с данными.

Здесь же описан класс Earthquake_MySql, в котором объединены одиннадцать функций установления диалога с MySQL.

```
class Earthquake_MySql{
    var $con;
    var $result;
    var $record = array();
    function f_OpenConnection($host, $user, $pass, $db) { ... }
```

```

function f_GetSqlError() { ... }
function f_CloseConnection() { ... }
function f_ExecuteSql($sql = "") { ... }
function f_GetSelectedRows($query_id = 0) { ... }
function f_GetAffectedRows(){ ... }
function f_GetRecord($query_id = 0, $param=0) { ... }
function f_FreeResult($query_id = 0) { ... }
function f_GetTable_List($query_id = 0) { ... }
function f_SetRecordPointer($recordnumber, $query_id = 0) { ... }
function f_GetNextId(){ ... }
}

```

Функция **f_OpenConnection** получает в качестве фактических параметров адрес хоста(сервера), на котором запущена СУБД, имя пользователя, пароль пользователя и собственно название базы, с которой следует установить соединение. Посредством SQL-запроса, функция передает данные параметры MySQL и устанавливает соединение. В случае ошибки при соединении, функция возвращает значение false.

```

function f_OpenConnection($host, $user, $pass, $db)
{
    $this->con = mysql_connect($this->host, $this->user, $this->password);
    if ($this->con)
        {
            if ($this->dbname)
                {
                    $dbselect = mysql_select_db($this->dbname);
                    if( !$dbselect )
                        {
                            mysql_close($this->db_connect_id);
                            $this->con = false;
                        }
                }
            return $this->con;
        }
    else return false;
}

```

Функция **f_GetSqlError** возвращает код и описание ошибки, произошедшей при попытке соединения с MySQL, что существенно помогает элиминировать все возможные трудности во время работы программного пакета.

```
function f_GetSqlError()
{
    $result['message'] = mysql_error($this->con);
    $result['code'] = mysql_errno($this->con);
    return $result;
}
```

Функция **f_CloseConnection**, как следует из ее названия, просто закрывает соединение и высвобождает ресурсы СУБД.

```
function f_CloseConnection()
{
    if( $this->con )
        return mysql_close($this->con);
    else
        return false;
}
```

Функция **f_ExecuteSql** – наиболее важная функция описанного класса **Earthquake_MySql**, которая выполняет конкретные запросы.

```
function f_ExecuteSql($sql = "")
{
    unset($this->result);
    if ($sql != "")
        $this->result = mysql_query($sql, $this->con);
    if ($this->result)
    {
        unset($this->record[$this->result]);
        return $this->result;
    } else {
        return false;
    }
}
```

Остальные функции класса – утилиты, которые распределяют память, выводят и запоминают данные в результирующие переменные или массивы и высвобождают память, временно выделенную на хранение данных и т.д.:

```
function f_GetSelectedRows($query_id = 0)
{
    if( !$query_id ) $query_id = $this->result;
    return ( $query_id ) ? mysql_num_rows($query_id) : false;
}
function f_GetAffectedRows()
{
    return ( $this->con ) ? mysql_affected_rows($this->con) : false;
}
```

Gen_Functions.php - модуль генерализованных функций

В модуле **Gen_Functions.php** описаны все те функции-утилиты, которые неоднократно используются клиентскими сценариями и сценариями администрирования базы знаний.

Все Функции которые учитывают работу с СУБД, в свою очередь, нуждаются в обращении к классу **Cl.php**, поэтому вызов класса происходит сразу после активации данного модуля, что обеспечивается первой строкой модуля **Gen_Functions.php** – **require_once (“ Cl.php”)**.

Примером часто используемых функций-утилит из данного модуля можно привести функцию формирования SQL запроса, которая принимает параметрами:

\$tabel-название таблицы из, которой следует извлечь данные;

\$param- параметр вида вывода, который определяет коком виде будут переданы данные – в виде значения переменной n-мерного массива, хеш-массива или вывода true/false;

\$what- параметр определяюши значения каких именно полей заданой таблицы следует выводить (по умолчанию выводятся все данные);

\$where- определяет условие выбора данных.

Функция также предусматривает вывод сообщение овозможной ошибке при выполнении операции извлечения данных а также вызывает функцию **\$free_resultset** из класса **Earthquake-MySql** заданого в модуле **Cl. Php**, целью высвобождения временно выделенных ресурсов памяти.


```

function fetch_table_data($table, $param=0, $what="*", $where=" 1")
{
    unset($Resultset);
    global $earthquake;
    $selectSql="SELECT ".$what." FROM ".$table." WHERE".$where." ";
    ...
    connect_earthquake();
    $Result=          $earthquake->f_ExecuteSql($selectSql);
    $Resultset=      $earthquake->f_GetRecord($Result, $param);
    $errors =        $earthquake->f_GetSqlError();
    if($errors['code']) { print_r($errors); }
    if(!empty($Resultset)) {
        $free_resultset= $earthquake->f_FreeResult($Result);
        return $Resultset;
    } else return false;
}

```

Еще одним примером утилитной функции данного модуля можно рассмотреть функцию описания структуры заданной таблицы, с единственным параметром \$table_name, который задает название таблицы, структура которой должна быть выведена. Данная функция неоднократно применяется администратором системы при работе с базой данных и особенно важна в случае изменения структуры базы, или какой-либо ее части.

```

function describe_tbl($table_name) {
    unset($fields);
    global $earthquake;
    connect_earthquake();
    $selectSql= "DESCRIBE $table_name";
    $Result=     $earthquake->f_ExecuteSql($selectSql);
    $fields=     $earthquake->f_GetRecord($Result, 3);
    $errors =    $earthquake->f_GetSqlError();
    if($errors['code']) { print_r($errors); }
    if($fields) {
        $free_resultset= $earthquake->f_FreeResult($Result);
    }
}

```

```

return $fields;
    } else {
        return false;
    }
}

```

- Ветом-же модуле определены такие функции как:
- Функция записи данных в базу;
- Функция записи данных и результатов в файл f-wfile (\$ filename); Функция считывания данных из файла f-rfile (\$filename);

§ 3.6. Каталог Fuzzy_Functions

Каталог **Fuzzy_Functions**, который представляет собой библиотеку нечетких методов, состоит из 5 модулей. Первые 4 соответствуют нечетким методам:

- **SFC.php** – статистический метод анализа нечетких классов
- **DSC.php** – метод дискриминационного анализа
- **CBR.php** – метод рассуждений на основе прецедентов
- **Experton.php** – метод экспертов

В соответствующих модулях объединены функции и процедуры получения возможного решения. Это блок принятия решения. Модули данного блока представляют запрограммированные математические методы, описанные в соответствующей главе.

Как пример, приведем часть программного кода модуля **SFC.php**.

Получая от пользователя вводные данные, функция **generate_res()** сравнивает значения введенных данных со значениями соответствующих параметров из базы знаний, на основе чего, происходит отбор выборочных частот; затем, подключив другие функции данного модуля - **generate_res()** составляет вектор $wect = (w_1, w_2, w_3, \dots, w_n)$, где индекс n (здесь и далее) определяется количеством факторов-предвестников землетрясения определенных в базе в таблице **f_factors**

```

for($i=0;$i<$n; $i++) {
    $wect[$i]=$w[$indi[$i]][$indj[$i]];
}

```

Далее, происходит выбор значений f_n^0, f_n^1 и f_n^2 , и составляется матрица **matrix_f**

```

for($i=0; $i<$n; $i++) {

```

```

for($j=0; $j<3; $j++) {
    $matrix_f[$n][3]=0;
    for($i=0; $i<n; $i++)
    {
        $matrix_f[$i][0]=$f0[$indi[$i]][$indj[$i]];
        $matrix_f[$i][1]=$f1[$indi[$i]][$indj[$i]];
        $matrix_f[$i][2]=$f2[$indi[$i]][$indj[$i]];
    }
}
}

```

В результате перемножения полученных вектора и матрицы – получаем взвешенный вектор возможных решений с тремя значениями, из которых после нормализации выбирается максимальное. Полученное максимальное значение и представляет результат прогноза данного метода:

```

for($j=0; $j<3; $j++) {
    $result[$j]=0;
    for($i=0; $i<$n; $i++)      $result[j]+=$wect[$i]*matrix[$i][$j];
}

$max_val=$result[0]; $ind=0;
if($result[1]>$max_val) {
    $max_val=$result[1];      $ind=1;
} elseif ($result[2]> $max_val) {
    $max_val =$result[2];      $ind=2; }

```

В каталоге **Fuzzy_Functions** содержится также модуль **Make_Values.php**, работа которого обеспечивает создание базы знаний (объединение различных таблиц). Из таблицы начальных данных выбираются значения активностей и по определенным экспертом правилам происходит заполнение соответствующей таблицы базы знаний. Данный модуль вызывается при изначальной активации системы, а также в том случае, если администратор системы производит обновление базы знаний новыми значениями.

Формирование базы знаний, также как извлечение из нее информации, производится по простым правилам продукционного типа «*ЕСЛИ А и В и С и ... , ТО К*». Каждый из первых трех описанных нечетких методов применяет правила этого типа соответственно на этапах фазификации и вывода.

Одна из таблиц базы знаний, созданная для нечеткого метода дискриминационного анализа выглядит следующим образом:

БАЗА ЗНАНИЙ - Метод DSC (дискриминационный анализ)

в день землетрясения

Параметр	Границы	Четкие частоты			Положительные дискриминации			Отрицательные дискриминации		
Напряженность электрического поля	<6.5	3	2	4	0.3	0.155547	0.444453	0.266667	0.466667	0.166667
Напряженность электрического поля	>=6.5	4	5	3	0.284453	0.38888	0.18	0.266667	0.186667	0.4
Температура воздуха	<8.8	4	4	5	0.24	0.24	0.333333	0.3	0.3	0.213333
Температура воздуха	>=8.8	3	3	2	0.333333	0.333333	0.177787	0.222213	0.222213	0.4
Температура почвы	<11	5	4	6	0.277787	0.195547	0.36	0.266667	0.366667	0.2
Температура почвы	>=11	2	3	1	0.355547	0.6	0.11112	0.266667	0.133333	0.666667
Атмосферное давление	<915	2	6	5	0.0977867	0.56	0.444453	0.733333	0.155547	0.213333
Атмосферное давление	>=915	5	1	2	1	0.0933333	0.32	0.08	0.933333	0.4
Влажность абсолютная	<9	5	6	5	0.244453	0.32	0.244453	0.293333	0.222213	0.293333
Влажность абсолютная	>=9	2	1	2	0.4	0.133333	0.4	0.2	0.533333	0.2
Влажность относительная	<75	3	5	5	0.16	0.355547	0.355547	0.444453	0.213333	0.213333
Влажность относительная	>=75	4	2	2	0.533333	0.2	0.2	0.133333	0.4	0.4
Облачность общая	<6	4	2	3	0.444453	0.155547	0.3	0.166667	0.466667	0.266667
Облачность общая	>=6	3	5	4	0.18	0.38888	0.284453	0.4	0.186667	0.266667
Облачность нижняя	<3	2	3	1	0.355547	0.6	0.11112	0.266667	0.133333	0.666667
Облачность нижняя	>=3	5	4	6	0.277787	0.195547	0.36	0.266667	0.366667	0.2

Скорость ветра	<1.26	3	4	5	0.18	0.284453	0.38888	0.4	0.266667	0.186667
Скорость ветра	>=1.26	4	3	2	0.444453	0.3	0.155547	0.166667	0.266667	0.466667

Кроме таблиц для реализации нечетких методов база знаний содержит ряд таблиц, хранящих экспертные данные. Например, фрагмент таблицы, содержащей значения функций совместимости, установленных на основании консультаций с экспертом, выглядит так:

Редактирование m функций		
день	название	значение
в день землетрясения	m00	0.93968
в день землетрясения	m10	0
в день землетрясения	m01	0
в день землетрясения	m11	0.55192
в день землетрясения	m21	0.3641
в день землетрясения	m12	0.26968
в день землетрясения	m22	0.6552

Из заголовка таблицы явствует, что возможно ее редактирование, т.е. при нажатии на одну из ячеек вызывается интерфейс для редактирования таблицы, который позволяет изменить название функции, ее значение и день подготовки землетрясения:

редактирование m функции	
укажите день	в день <input type="text"/>
название	<input type="text" value="m11"/>
значение	<input type="text" value="0.55192"/>
<input type="button" value="ВВОД"/>	

Для первоначального заполнения данной таблицы предусмотрен удобный интерфейс, позволяющий выбрать имя функции, добавить ее значение и указать, для какого из дней подготовки землетрясения функция вводится:

Добавить значение m функции	
укажите день	в день <input type="text"/>
название	<input type="text"/>
значение	<input type="text"/>
<input type="button" value="ввод"/>	

База знаний содержит также таблицу определения классов классификации прогнозируемого объекта с указанием названия лингвистической переменной и численных значений интервалов классов:

Редактирование интервала интенсивности		
название (интервал интенсивности)	нижняя граница	верхняя граница
Нет землетресения	0	3
Среднее	3.1	4.99
Сильное	5	9

В этом случае также предусмотрено редактирование, соответствующий интерфейс вызывается простым нажатием на одну из ячеек таблицы:

редактирование интервала интенсивности	
выберите параметр	среднее <input type="text"/>
нижняя граница	3.1 <input type="text"/>
верхняя граница	4.99 <input type="text"/>
<input type="button" value="ввод"/>	

Возможно также удаление класса классификации, что осуществляется через соответствующий интерфейс:

удалить значение интервала интенсивности		
название (интервал интенсивности)	нижняя граница	верхняя граница
Нет землетресения	0	3
среднее	3.1	4.99
сильное	5	9

А также добавление класса классификации:

Добавить значение интервала интенсивности	
название (интервал интенсивности)	<input type="text"/>
нижняя граница	<input type="text"/>
верхняя граница	<input type="text"/>
<input type="button" value="ввод"/>	

§ 3.7. Описание каталога Admin

Каталог сценариев по администрированию **Admin** состоит из двух подкаталогов

- Подкаталог сценариев по администрированию базы **Knowledge_BD_Administration**
- Подкаталог сценариев по администрированию доступа к данным **User_Administration**

Несмотря на то, что базу знаний заполняет сама компьютерная система, в оболочке системы предусмотрено участие инженера знаний в формировании и обновлении базы знаний.

Например, осуществлена возможность добавления параметров (атрибутов) прогнозируемого объекта с указанием их названия (или их удаления):

Добавить параметр	
название	<input type="text"/>
<input type="button" value="ввод"/>	

3.7.1.Подкаталог Knowledge_BD_Administration.

Пользователь с наивысшим уровнем доступа посредством данного сценария может изменять значения тех или иных полей базы знаний, преобразовывать, обновлять и/или дополнять ее. Каталог состоит из 2-х модулей:

3. knbd_manual_amendment.php
4. knbd_amendment.php

Модуль knbd_manual_amendment.php

Данный модуль состоит из процедур и функций, позволяющих инженеру знаний изменять заданные в базе знаний значения в любой момент времени по собственному усмотрению. Несколько интерфейсов для осуществления этого было приведено выше. Рассмотрим еще один – т.к. во всех применяемых для вывода решения нечетких статистических методах прогнозирующие параметры (активности) также делятся на классы (подфакторы), количество и границы которых выбираются произвольно, соответствуясь с мнением эксперта, в компьютерной системе предусмотрено добавление границ параметров (фактически, это означает увеличение подфакторов), удаление и редактирование этих границ. Приведем интерфейс для добавления:

Добавить значение границы параметра	
выберите параметр	Напряженность электрического поля ▾
укажите метод	dsc ▾
укажите день	в день ▾
строка	1 ▾
нижняя граница	<input type="text"/>
верхняя граница	<input type="text"/>
<input type="button" value="ввод"/>	

С помощью этого же интерфейса осуществляется первоначальное разделение параметров на классы.

Модуль **knbd_amendment.php** состоит из сценариев, которые автоматически переводят администратора в режим обновления базы данных общего назначения новыми значениями. Разработанный программный продукт предусматривает накопление информации о правильно принятых решениях и сохранение значимых данных до момента, пока не будут получены данные обо всех нечетких подмножествах прогнозируемого понятия. В администраторском интерфейсе предусмотрено отражение информации о накоплении.

Система учитывает наличие данных за три дня подготовки землетрясения, включая сам день землетрясения. Все данные и полученные решения сохраняются в отдельной таблице **forecasting_data**. Программный продукт допускает обновление базы данных только в том случае, если в таблице **forecasting_data** накопились данные по всем видам прогнозируемого понятия на все дни рассматриваемые во время принятия решения. Иными словами, так как, на данный момент, мы работаем с тремя значениями прогнозируемого понятия (землетресение) – нет землетресения, среднее землетресение и сильное землетресение – и прослеживаем изменение значений факторов-предвестников за три дня, включая день составления прогноза, обновление базы знаний становится возможным каждый раз, когда в таблице **forecasting_data** фиксируется три набора новых значений:

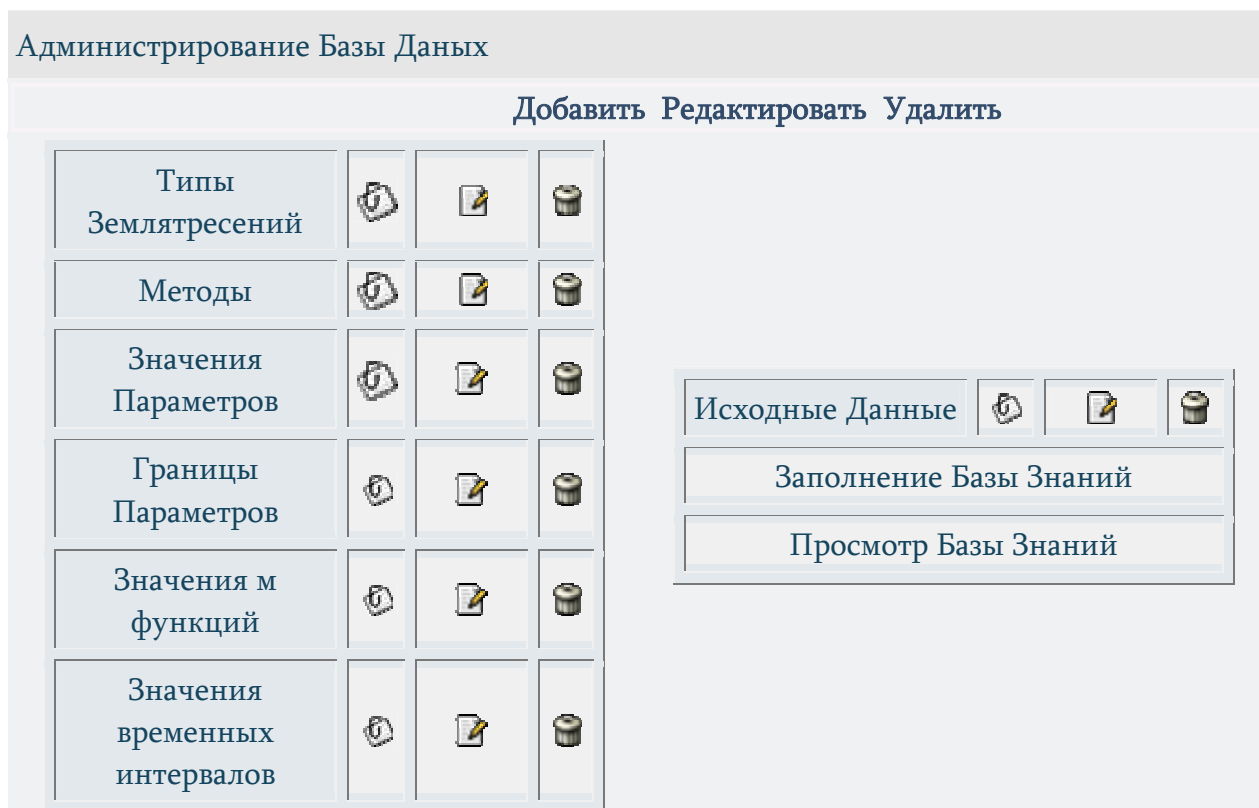
- значения факторов-предвестников за два дня до и в день составления прогноза для случая когда землетресения не наблюдалось;
- накопились данные по трем дням, включая сам день прогнозирования, когда имело место среднее землетресение;
- получен набор данных для случая сильного землетресения.

Когда вся необходимая информация накоплена, программный продукт оповещает администратора и предлагает обновить базу данных. Администратор, по своему усмотрению, либо соглашается и обновляет базу данных новыми значениями, либо удаляет накопленные данные целиком или частично.

Существование модуля **knbd_manual_amendment.php** было predeterminedено обобщенным характером предложенной компьютерной системы, в том смысле, что эта система может быть использована не только для задачи прогноза землетрясений. В

случае же применения системы для другой задачи возникнет необходимость в изменении содержимого базы знаний.

Общий вид интерфейса для администрирования базы данных MySQL, что означает как администрирование базы данных общего назначения, так и базы знаний, приводится ниже:



3.7.2. Подкаталог User_Administration.

Подкаталог **User_Administration** предназначен для добавления пользователей или изменения их привилегий. Сценарий **user_admin.php** вступает в диалог с таблицей **Users** в базе данных и предоставляет возможность администратору добавлять новых, активировать уже существующих, удалять пользователей, изменять уровень доступа, накладывать ограничение на использование программного продукта.

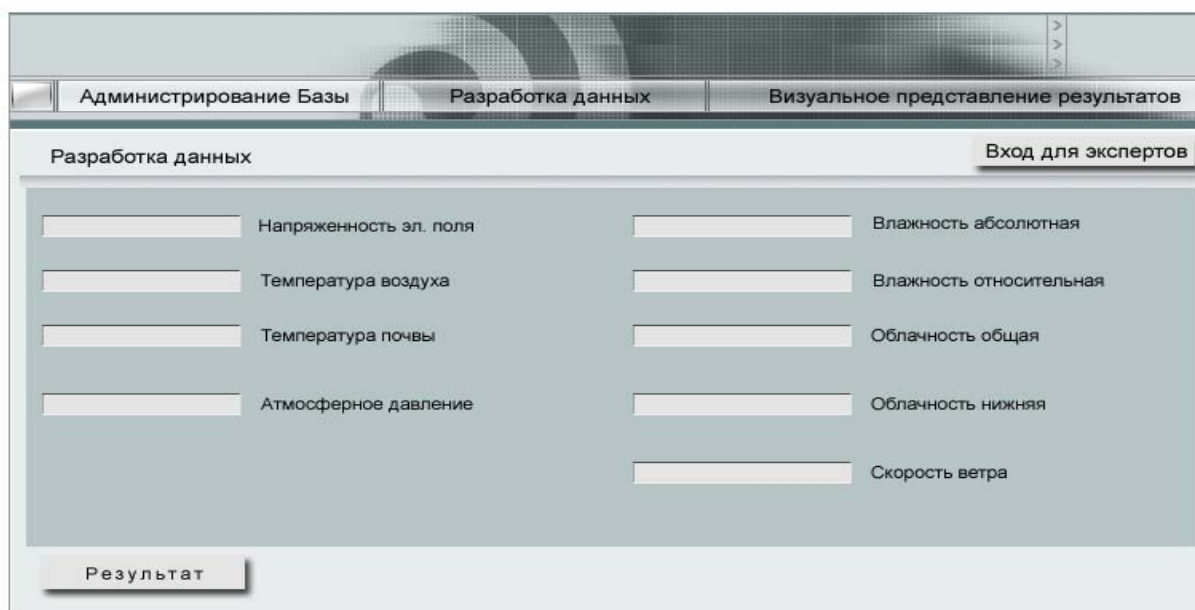
Во время обращения к вышеописанному сценарию, в первую очередь происходит проверка уровня доступа и IP адреса отправителя запроса. Только пользователь с наивысшим уровнем доступа, т.е. администратор системы, допускается до исполнения рассмотренного сценария.

§ 3.8. Описание каталога Main

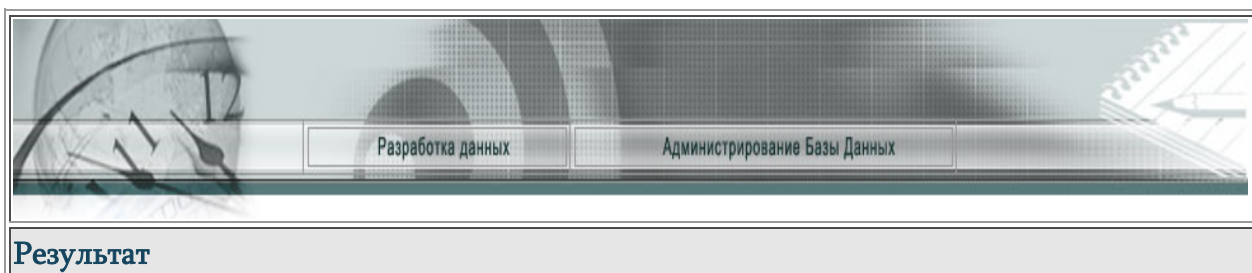
Каталог **Main** содержит пользовательские сценарии.

Например, на одной из веб-страниц пользовательского интерфейса, которая как раз и вызывается из каталога Main, предоставляется возможность ввода значений активностей для получения решения о новом конкретном случае.

На данном этапе разработки программного продукта пользовательский интерфейс имеет простой, но достаточно удобный дизайн. К примеру, веб-страница ввода значений факторов-предвестников для исследуемого случая выглядит следующим образом:



После ввода при нажатии на соответствующую клавишу происходит активизация модулей каталога Fuzzy_Functions и начинаются реальные вычисления. Результаты этой работы передаются модулю results.php, который также находится в каталоге Main. Модуль results.php ответственен за вывод результата. Возможна форма вывода результата прогноза названием интервала интенсивности прогнозируемого объекта (выводом названия нечеткой лингвистической переменной).



Результаты расчета по методу EXPERTON		
Нет землетресения	Среднее	Сильное
<i>0.53636</i>	<i>0.75000</i>	<i>0.65455</i>
Окончательный Прогноз: Среднее Землетрясение		

Вывод и основные результаты

Задача принятия решений, объект исследования которой уже в своем определении содержит нечеткость, должна использовать специальные методы обработки нечеткой информации, например, нечеткие статистические методы. Исходя из этого, в случае наличия подобных задач компьютерная система принятия решений должна обеспечивать возможность представления нечеткой информации и ее обработки. В работе предложена модель компьютерной системы принятия решений, которая применяет статистический и нечеткий анализ и основана на базе знаний т.н. таблично-цифрового типа. Основные результаты работы заключаются в следующем:

- В работе обоснованно применение средств интернет-программирования для создания компьютерной системы принятия решения, когда данные и знания расположены централизованно, описаны разработанные каталоги и модули для обработки нечеткой информации, формирования базы знаний и доступа к ней.

- Исследованы методы разработки таблично-цифровой базы знаний и 3-х нечетких метода принятия решения на ее основе - статистический метод анализа нечетких классов, метод дискриминационного анализа, метод рассуждений на основе прецедентов. Рассмотрены их как положительные, так и отрицательные стороны. Опираясь на эту информацию для получения наиболее возможного единственного решения исследован 4-тый нечеткий метод, основанный на теории экспертов и позволяющий выбрать решение, руководствуясь предпочтениями группы экспертов.

- Исследованы основные свойства построения функций совместимости, различные модели этих функций. Предложена и опробована функция совместимости,

являющаяся новой модификацией модели Заде. Нечеткие статистические методы адаптированы к конкретной задаче, предложена их определенная модификация.

- Предложена новая модель компьютерной системы принятия решений, в которой база знаний создается самой системой на основе информации, получаемой из базы данных общего назначения, где хранятся уже существующие правильно принятые решения вместе со значениями характерных для них активностей. Это обстоятельство предопределили, с одной стороны, алгоритмы методов обработки информации, а с другой – программные средства, выбранные для реализации модели компьютерной системы.

- Разработаны алгоритмы создания базы знаний, получения из нее информации, обработки нечетких понятий и реализации нечетких методов.

- Разработанные алгоритмы создания базы знаний, получения из нее информации и работы нечетких статистических методов реализованы.

- Разработанные модули для создания базы знаний и доступа к ней, модули реализации статистических нечетких методов принятия решения и другие представляют программное обеспечение, на основе которого становится возможным практическое создание компьютерной системы принятия решений, основанной на статистическом и нечетком анализе.

- Создан пакет программ, представляющий собой компьютерную систему прогноза возможности землетрясения. Предлагаемая компьютерная система носит общий характер, т.к. построенную иерархию каталогов, модулей и классов возможно использовать в различных сферах исследований, а саму систему, при определенной модификации, для других задач принятия решений.

Л и т е р а т у р а

1. Adlassnig K., Fuzzy set theory in medical diagnosis, IEEE Transactions on Systems, Man and Cybernetics 16, 1986, pp. 260–265.
2. Aikins J. S. (1983). Prototypical knowledge for expert systems. Artificial Intelligence, 10, p. 163-210.
3. Atkin R.H., Mathematical structures in human affairs. Crane, Russak&Go. №4(1974)

4. Baldwin J. and Martin T., Fuzzy classes in object-oriented fuzzy programming, Proceedings of 5th IEEE International Conference on Fuzzy Systems, 1996, pp.1358-1364.
5. Baldwin J., Martin T. and Vargas-Vera M., Fril++: object-based extensions to Fril. Logic Programming and Soft Computing, Research Studies Press, 1998, pp. 223-238.
6. Belmonte M., Sierra C., and López de Mántaras R., Renoir: an expert system using fuzzy logic for rheumatology diagnosis, International Journal of Intelligent Systems 9(11), 1994, pp.985-1000.
7. Bonfa I., Maioli C., Sarti F., Milandri G. and Monte P., HERMES: An Expert System for the Prognosis of Hepatic Diseases, Technical Report UBLCS-93-19, Universiti of Bologna, 1993.
8. Bryan G. Talbot, Bruce B. Whitehead, Lisa M. Talbot, "Metric Estimation via a Fuzzy Grade-of-Membership Model Applied to Analysis of Business Opportunities," p. 431, 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02), 2002.
9. Cao Z., Kandel A., Applicability of some fuzzy implication operators – FSS, 31, N2, 1989
10. Coombs M.J. and Alty J.L., Expert systems: concepts and examples, The National Computing Centre Limited, 1987.
11. Criado F., Gachechiladze T., Fuzzy random events and their corresponding conditional probability measures, Revista de la Real Academia de ciencias Exactas, Fisicas y Naturales de Madrid, tomo LXXXIX (1995), Matematicas.
12. Criado F., Gachechiladze T., Meladze H., Tsertsvadze G.– A new Approach to Analysing Fuzzy Data and Decission-making Regarding the Possibility of Earthquake Occurrence – Intas-9702126 (Final Report), www.cordis.lu/en/home.html , 1999.
13. Cunliffe D., Biomedical Applications of Belief networks, PHd Thesis, Division of Informatics, University of Edinburg, 1996.
14. De Dombal F.T., Leaper D.J., Horrocks J.C. et al., Human and computer-aided diagnosis of abdominal pain, British Med. J.1., 1974, pp. 367-380.
15. Dombi J., Membership function as an evaluation. FSS, v. 35, №1(1990).
16. Droy J. M., Darmoni S. J., Massari P., Blanc T., Moritz F., and Leroy J., SETH: An Expert System for the Management on Acute Drug Poisoning, Comput. Methods Programs

- Biomed 43, 1993, pp. 171-176. <http://www.chu-rousen.fr/dsii/publi/seth.htm>
17. Dubois D., Prade H., and Rossazza J., Vagueness, typicality and uncertainty in class hierarchies, *International Journal of Intelligent Systems* 6, 1991, pp.167-183.
 18. Durfee E.H., Lesser V.R., Gorkill D.D. Trends in cooperative distributed problem solving. - *IEEE Trans. Knowledge and Data Eng.* March, v.1, №1, p.63-83, 1989.
 19. Eom S.B., Decision support systems research: reference disciplines and a cumulative tradition, *The International Journal of Management Science*, 23, 5, October 1995, p. 511-523.
 20. Findler N.V., Lo R. An examination of distributed planning in the world of air traffic control. – *Parallel Distributed Comput.*, v.3, p. 411-431, 1986.
 21. Friedman M., Henne M. and Kandel A., Most typical values for fuzzy sets, *Fuzzy sets and Systems* 87, 1997, pp. 27-37.
 22. Gachechiladze T., Sirbiladze G., Restored fuzzy measures in expert decision-making *Information Sciences: an International Journal archive*, Volume 169, Pages: 71 – 95, 2005
 23. Ghosh D., Sarkar S.and Dardeau P. - Knowledge-based Simulation for Business Process Redesign *International Journal of Intelligent Systems in Accounting, Finance & Management*, Vol 5, Number 1, 1996, 14 Pages
 24. Gpyal S., Worrest R. Expert system applications to network management. - *Expert System Applications to Telecommunications*. New York, v. 1, p.3-44, 1988.
 25. Hirota K., Kurisu H., Yoshino H., A Precedent-base Legal Judgement System Using Fuzzy Database – *Fuzziness and Knowledge Based Systems*, 4-6, pp.573-580,1996
 26. Hussein D., Pepe S., Investment using technical analysis and fuzzy logic. / *Fuzzy Sets and Systems*, 127 (2002) 221–240.
 27. Juzhang Li - Fuzzy Statistics of Classification - *Fuzzy Mathematics*, 2(4) (1988), p. 107.
 28. Kandel A. and Byatt W., Fuzzy sets, fuzzy algebra and fuzzy statistics, *Proc. IEEE* 66, 1978, pp.1619-1639.
 29. Kandel A., Fuzzy Statistics and forecast evaluation, *IEEE Trans. Systems Man Cybernet* 8, 1978, pp. 396-401.
 30. Kandel A., *Fuzzy techniques in pattern recognition*, John Wiley, New York, 1982.
 31. Kaufmann A., *Les expertons*, Hermes, Paris, 1987.

32. Kaufmann A., Expert Appraisements and Counter-Appraisements with Experton Processes - Analysis and Management of Uncertainty: Theory and Applications, North-Holland, Amsterdam, pp. 109-132, 1992.
33. Kaufmann, Arnold, and Gupta, Madan M. Introduction to Fuzzy Arithmetic/ Thomson Computer Press, 1991.
34. Khutsishvili I., Statistical Method of Fuzzy Grades' Analysis for Forecast Modeling, Applied Mathematics, Informatics and Mechanics, v.1, 2006, pp.12-19.
35. Khutsishvili I., An application of the statistical method of fuzzy grades' analysis - Bulletin of the Georgian Academy of Sciences, 173, № 2, 2006, 266-268.
36. Lukasiewicz T., Probabilistic logic programming, Proceedings of the 13th Biennial European Conference on Artificial Intelligence, 1998, pp. 388-392.
37. Miller R.A., Pople H.E., Myers JD., Internist-1, an experimental computer-based diagnostic consultant for general internal medicine, New England Journal of Medicine 307, 1982, pp.190-209.
38. Nedovic L., Devedgic V. Expert system in finance - a cross-section of the field / Expert Systems with Applications, 23 (2002), 49-66.
39. Nedovic L., Devedzic V. - Expert Systems with Applications, International Journal of Intelligent Systems in Accounting Finance and Management, Great Britain, 2002, V. 23; N. 1, pages 49-66.
40. Neves J., Alves V., Nelas L., Romeu A. and Basto S., An information system that supports knowledge discovery and data mining in medical imaging, Machine Learning and Applications: Machine Learning in Medical Applications, Chania, Greece, 1999, pp. 37-42.
41. Norris D., Pilsworth P.W., Baldwin J.F., Medical diagnosis from patient records, Fuzzy Sets and Systems 23, 1987, pp. 73-87.
42. Olson D., Courtney J. Decision supports, models and expert system. Mac Millan Publ. 1992.
43. Popesku D., Fuzzy Architectures for the Implementation of Expert Systems, PhD Thesis in Computer Science, the University Politehnica of Bucharest, Computer Science Department, April 2003.

44. Schneider M., Friedman M., and Kandel A., On fuzzy reasoning in expert systems, Proc. 17th International Symposium on Multiple-valued Logic, Boston, MA, 1987, pp.70-74.
45. Seising R., Schuh C., and Adlassnig K.P., Medical knowledge, fuzzy sets, and Expert Systems, 1st Workshop on [Intelligent and Adaptive Systems in Medicine](#), Prague, March 31 – April 1, 2003. <http://cyber.felk.cvut.cz/EUNITE03-BIO/pdf/Seising.pdf>
46. Shortliffe E. H., Computer programs to support clinical decision making. Journal of the American Medical Association, Vol. 258, No. 1, 1987, pp 61-67.
47. Shortliffe E.H. and Buchanan B.G., A model for inexact reasoning in medicine, Mathematical Biosciences 23, 1975, pp. 351–379.
48. Shortliffe E. H., Computer-based medical consultations: MYCIN. Elsevier, New York, 1976.
49. Sikharulidze A., Application of discrimination and connectivity analysis in psychiatry, Bulletin of the Georgian Academy of Sciences, Vol. 164 , No.1, 2001, pp.39-40.
50. Simonovic A., Slobodan P. Decision support for sustainable water resources development in water resources planning in a changing world - Proceeding of International UNESCO symposium, Karlsruhe, Germany, p. III. 3-13,1994.
51. Sirbiladeze G. and Sikharulidze A., Weighted fuzzy averages in fuzzy environment, part II. Generalized weighted fuzzy expected values in fuzzy environment, International Journal of Uncertainty, Fuzziness and Knowledge-based Systems, Vol. 11, No.2, 2003, pp. 159-172.
52. Smirnov S.V., Tyukavkin D.V., A geoinformation system for decision-making support in social and educational authorities - Журнал "Проблемы управления", №3, 2003.
53. Steve Ku, Yung-Ho Suh and Gheorghe Tecuci - Building An Intelligent Business Process Reengineering System, International Journal of Intelligent Systems in Accounting, Finance & Management, Vol 5, Number 1, 1996.
54. Street W. N., Mangasarian O. L. and Wolberg W. H., Individual and collective prognostic prediction, Technical Report 96-01, Computer Sciences Department, University of Wisconsin, Madison, WI, 1996.
55. Summit Sukar, Ram Sriram and Shibu Joykutty - Belief Networks for Expert System Development in Auditing, International Journal of Intelligent Systems in Accounting

- Finance and Management, Vol 5, Number 3, 1996.
56. van Melle W. J. (1981). System Aids in Constructing Consultation Programs. Ann Arbor MI: UMI Research Press.
 57. Wang Peizhuang – Fuzzy Sets and its Application - Publishing House of Science and Technology, Shanghai, 1983.
 58. Walker, N. J., and Kwon, O., ISS: An expert system for the diagnosis of sexually transmitted diseases, 11th Annual Midwest Computer Conference (MCC'97), Springfield, Illinois, 1997 .
 59. Zadeh, Lotfi. Fuzzy Sets / Information and Control, 8(3), June 1965, pp.338-53.
 60. Zadeh, Lotfi. Outline of a New Approach to the Analysis of Complex Systems and Decision Processes / IEEE Transactions on Systems, Man, and Cybernetics, SMC-3(1), January 1973,pp.28-44.
 61. Zadeh L.A., The roles of fuzzy logic and soft computing in the conception, design and development of intelligent systems, Software Agents and Soft Computing, Springer-Verlag, 1998, pp.183-190.
 62. Zuoying Li, Zhenpei Chen and Jitao Li - A Model of Weather Forecast by Fuzzy Grade Statistics - FSS 26, N 3, June (1988), pp. 275-283.
 63. Барлетт Н., Лесли А., Симкин С. Программирование на Java. Путеводитель. DiaSoft. Киев, 1996.
 64. Гачечиладзе Т., Церцвадзе Г., Математическая модель многошагового процесса поддержки принятия решения в задаче прогноза землетрясений – Proceeding of Iv.Javakhishvili Tbilisi State University, Vol. 330(19), pp. 55-58.
 65. Гачечиладзе Т., Об одном методе нечеткого анализа в задаче оценки параметров землетрясений, Proceeding of the Geophysical Center, Vol. LVIII, 2003.
 66. Горбунов А.Р., Управление финансовыми потоками и новое поколение экспертных систем, 1998, www.tora-centre.ru/library/Reing/expart.htm
 67. Добролюбов И.П., Савченко О.Ф., Альт В.В. Идентификация состояния сельскохозяйственных объектов измерительными экспертными системами. – Новосибирск: СибФТИ, 2003, 208 с.
 68. Дремов С.Н., Каган М.М., Легашов М.А., Олейник О.В., Фридман А.Я., Экспертная

- система диагностики состояния породного массива на подземном руднике, Сб. Региональные информационные системы. Апатиты, 1995, С.77-83.
69. Заде Л., Понятие о лингвистической переменной и его применение к принятию решений. - М.: Мир, 1976.
 70. Исаев Г.Н. Информационные системы в экономике - International ISBN Agency, Berlin - Germany : 5-98119-965-2
 71. Каган М.М. Система сейсмического мониторинга на апатитовых рудниках, опыт разработки и перспективы развития // Физика и механика разрушения горных пород применительно к прогнозу явлений в горном массиве. - Сборник трудов семинара (ВНИМИ, 24-26 октября 1994 г.) - С.-Петербург.
 72. Литвак Б.Г., Экспертные технологии в управлении: интеграция науки и образования / Семинар: Экспертные оценки и анализ данных, Центр "Прикладная прогностика" Института проблем управления РАН, 2004.
 73. Лобанов Ю.И., Брусиловский П.Л., Съедин В.В. Экспертно-обучающие системы. М.,1991. 56с. (Новые информационные технологии в образовании: Обзор. Инф./НИИВО; вып.2).
 74. Масалович А., Нечеткая логика в бизнесе и финансах, 2002, <http://www.tora-centre.ru/library/fuzzy/fuzzy-.htm>
 75. Нейлор К., Как построить свою экспертную систему.- М.: Энергоатомиздат, 1991.
 76. Попов Э.В., Фоминых И.Б., Кисель Е.Б., Шапот М.Д. Статистические и динамические экспертные системы. - М. Финансы и статистика. 1996.
 77. Ратшиллер Т., Геркен Т. PHP: разработка Web-приложений. Библиотека программиста – СПб: Питер, 384 с.(+CD), 2001.
 78. Рейнхард Р., Ленц Дж., Flash 5. Библия пользователя: Пер. с англ. – М: Изд. дом «Вильямс», 1164 с., 2001.
 79. Смолин Д.В., Экспертная система для обучения – VIPES, <http://altnet.ru/~mcsmall/index.htm>
 80. Трахтенгерц Э.А., Методы генерации, оценки и согласования решений в распределенных системах поддержки принятия решений. – АиТ, №4, с.3-52, 1995.
 81. Уткин В.Б., Балдин К.В., Информационные системы в экономике – Академия, 2004,

282.

82. Яргер Р., Риз Дж., Кинг Т., MySQL и mSQL. Базы данных для небольших предприятий и Интернета – СПб: Символ-Плюс, 560 с., 2000.

Приложение

С целью тестирования компьютерной системы были рассмотрены 60 случаев т.н. «шумов» (дней, когда землетрясение не наблюдалось), при этом считается что $0 \leq M \leq 3$, и 20 случаев землетрясерий: 10 т.н. «средних» – со значением магнитуды $3 < M < 5$ и 10 «сильных» – со значением магнитуды $5 \leq M \leq 8$. Рассматривается статистика 1967-1992 гг. – результаты измерений некоторых геофизические данных атмосферы (данные станции Душетского района Грузии и Гидрометцентра). В частности, в качестве факторов-предвестников рассматриваются следующие геофизические данные атмосферы: 1). величина напряженности электрического поля (вольт/м); 2). температура воздуха (в градусах Цельсия); 3). температура почвы (в градусах Цельсия); 4). атмосферное давление (в мб); 5). абсолютная влажность (упругость водяного пара в мб); 6). относительная влажность (в %); 7). общая облачность (в баллах); 8). нижняя облачность (в баллах); 9). скорость ветра (в м/сек.).

Значения величин активностей замерены в течение дня с промежутком в 3 часа: 0^{00} , 3^{00} , 6^{00} , 9^{00} , 12^{00} , 15^{00} , 18^{00} , 21^{00} . Далее приводятся средние значения каждого из предвестников при подготовке землетрясения (за 2 дня до, за 1 день до) и в день землетрясения.

Землетрясения средней силы

02.06.1967, 5^{00} , $M = 4.5$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
31	5.04166	9.6625	12.125	908.275	9.2625	77.75	8.875	7.25	3.875
1	6.91666	11.725	17.0	911.575	6.375	51.375	4.25	1.75	9.875
2	6.66666	15.725	22.25	912.9125	8.8	48.125	4.5	1.625	1.625

03.01.1970, 6⁰⁰, $M = 4.7$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
1	9.29166	-0.9375	-1.75	914.325	4.0125	71.25	4.5	2.5	3.625
2	7.8333	-1.825	-1.125	920.0	4.425	83.625	8.25	5.375	1.5
3	12.7916	1.7875	-1.625	916.35	3.3875	50.5	6.375	2.25	2.625

22.03.1972, 0⁰⁰, $M = 4.5$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
20	10.875	-2.8	1.125	923.25	4.0375	81.625	9.125	7.25	1.625
21	12.4166	-2.15	1.75	917.675	3.425	66.875	8.875	7.125	1.375
22	9.9583	0.0875	3.625	911.225	4.25	73.25	6.125	4.875	0.875

13.11.1974, 2⁰⁰, $M = 4.7$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
11	6.1666	8.6875	10.125	917.05	8.9	79.75	6.375	5.0	1.25
12	5.375	5.525	6.25	923.55	8.75	96.625	9.875	8.5	0.5
13	3.1466	5.075	4.5	923.0375	8.1125	92.625	6.125	4.625	0.875

28.07.1976, 21⁰⁰, $M = 4.7$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
26	9.79166	22.2375	25.375	906.3125	18.35	69.0	1.5	1.25	1.375
27	8.125	21.625	25.125	907.7	21.2875	82.875	2.125	1.625	1.375
28	7.20833	20.975	26.5	909.4875	20.0125	82.875	3.75	2.375	0.5

17.03.1978, 21⁰⁰, $M = 4.4$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
15	10.2083	1.725	1.25	917.425	4.9563	72.625	7.25	6.625	4.25
16	7.625	2.7	3.875	918.4125	5.5363	76.125	6.375	4.625	1.375
17	5.95833	4.4	5.125	914.4375	6.0175	74.75	8.25	6.0	0.875

12.07.1978, 11⁰⁰, $M = 4.4$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
10	7.375	25.2	31.125	910.05	22.1875	70.25	3.125	2.125	1.125
11	7.16666	22.075	31.25	911.4125	22.425	85.875	8.5	7.0	1.25
12	6.70833	21.175	30.125	914.9875	20.9625	85.5	7.125	6.0	1.0

11.12.1980, 0⁰⁰, $M = 4.3$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
9	9.33333	4.25	0.375	912.775	4.7375	58.875	1.5	0.375	1.5
10	8.54166	4.925	0.5	907.3375	5.0075	59.625	2.125	0.75	0.875
11	5.95833	2.3	1.375	906.425	5.94125	82.875	6.75	6.375	0.75

18.10.1981, 18⁰⁰, $M = 4.6$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
16	6.66666	9.375	10.25	916.0125	8.4125	73.0	4.375	4.0	3.125
17	7.95833	9.0	7.625	916.825	7.5163	72.625	2.375	2.0	7.25
18	5.125	9.675	8.875	919.725	8.12	69.0	3.25	2.625	6.125

07.04.1989, 7⁰⁰, $M = 4.6$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
5	16.375	14.5125	13.75	916.8125	8.4375	51.875	2.25	2.125	1.5
6	6.375	13.275	15.625	913.8625	7.9663	51.5	3.875	2.875	1.125
7	7.58333	13.6	15.375	913.975	8.21375	64.875	7.375	6.25	2.0

Землетрясения сильные

26.07.1989, 18⁰⁰, $M = 5.8$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
24	7.16666	16.975	19.25	913.8125	15.4875	81.0	8.5	6.25	1.25
25	6.45833	17.7875	20.75	911.1875	14.2	73.125	3.375	1.5	1.375
26	6.25	18.5375	23.75	908.5875	13.9125	69.5	5.375	3.25	0.625

22.05.1971, 16⁰⁰, $M = 6.8$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
20	18.25	15.4625	17.75	915.675	12.0625	69.875	7.125	5.25	1.875
21	0.66666	13.6625	16.125	910.8375	11.8125	75.875	4.75	3.125	1.5
22	14.2083	14.7	19.25	909.6125	11.6875	73.25	5.625	4.5	1.0

9.01.1975, 23⁰⁰, $M = 5.2$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
7	11.875	0.1375	-2.375	911.25	4.7125	75.125	5.25	3.625	2.125
8	17.125	2.45	0.25	908.975	5.3875	74.5	2.625	0.875	1.25
9	19.375	3.05	-0.25	907.8125	5.5125	73.875	2.75	1.75	0.875

24.11.1976, 12⁰⁰, $M = 7$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
22	10.8333	4.0	0.125	912.7	6.325	78.875	3.5	0.625	0.25

23	10.6666	6.075	2.375	910.3625	5.6625	65.375	2.375	0.75	1.0
24	8.29166	8.7875	3.75	908.325	6.3375	56.25	6.0	5.0	0.75

30.09.1977, 16⁰⁰, $M = 5.4$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
28	2.70833	10.65	11.75	918.975	12.8625	100.0	9.25	8.75	0.5
29	6.875	9.25	10.25	920.1375	10.85	92.875	10.0	8.75	2.25
30	5.125	7.5125	8.625	924.2375	9.60125	92.25	10.0	8.25	1.125

26.02.1978, 13⁰⁰, $M = 5.3$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
24	4.83333	-2.7625	-2.125	915.35	3.6375	75.0	3.5	2.625	3.625
25	7.16666	-1.775	-1.5	918.925	3.72625	70.75	4.25	3.625	2.25
26	5.45833	-1.9375	-0.25	919.875	3.9225	75.0	7.625	5.0	1.5

18.10.1981, 5⁰⁰, $M = 5.4$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
16	6.66666	9.375	10.25	916.0125	8.4125	73.0	4.375	4.0	3.125
17	7.95833	9.0	7.625	916.825	7.5163	72.625	2.375	2.0	7.25
18	5.125	9.675	8.875	913.98	8.12	69.0	3.25	2.625	6.125

30.10.1983, 4⁰⁰, $M = 6.8$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
28	10.9166	10.4875	8.75	922.1125	9.40125	75.625	4.75	3.375	1.0
29	2.08333	9.925	7.375	918.8125	10.64	87.875	5.625	5.625	1.0
30	7.25	13.0875	10.75	914.8125	11.0125	75.375	4.5	4.25	0.875

6.03.1986, 0⁰⁰, $M = 6.1$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
4	24.625	0.825	-2.0	916.275	3.70625	59.125	8.5	8.5	0.375
5	22.75	1.2625	-2.625	915.8625	4.59375	69.625	6.5	5.625	0.375
6	2.75	4.65	0.375	914.275	6.80625	70.875	9.75	9.5	0.5

16.12.1990, 15⁰⁰, $M = 5.1$

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
14	12.2083	0.625	-0.125	914.7125	5.0225	79.25	6.375	6.375	0.625
15	12.5	4.4	-0.125	914.275	5.44375	66.0	0.75	0.625	0.75
16	12.9166	4.0125	-0.125	913.75	5.54375	68.625	3.0	3.0	1.0

Нет землетрясений

16.03.1992 - 18.03.1992

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
16	13.5833	7.35	5.75	919.9125	4.7125	47.625	0	0	1.125
17	9.29166	6.325	6.375	914.9	6.0125	64.25	3.25	1	0.875
18	6.8333	-3.3875	-2.125	919.45	3.7963	80.5	8	6.875	0.875

28.06.1987 - 30.06.1987

Число	Напряж эл. поля	Темпер воздуха	Темп. почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
28	6.5833	10.85	12.25	907.85	7.3625	74.875	7.25	4.875	1.5
29	5.5	11.3875	12.125	907.625	6.85	62.125	6	3	3.25
30	4.8333	-2.7625	-2.125	915.35	3.6375	75	3.5	2.625	3.625

5.12.1980 - 7.12.1980

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облач.о бщая	Облачн. нижняя	Скор. ветра
-------	--------------------	-------------------	-----------------	-------------------	------------------	-------------------	-----------------	-------------------	----------------

5	9.70833	5.175	2.0	913.1875	5.5875	59.375	1.0	0.5	1.25
6	5.41666	4.875	2.625	913.2	4.55625	58.125	1.0	0.625	1.0
7	8.1666	5.775	3.5	915.8375	5.95875	69.125	3.125	1.75	1.25

10.07.1982 - 12.07.1982

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
10	8.04166	22.125	28.125	911.15	19.35	64.75	1.125	0.5	1.25
11	6.58333	23.75	29.0	911.025	18.425	64.375	0.125	0.125	1.125
12	7.16666	22.075	31.25	911.4125	22.425	85.875	8.5	7.0	1.25

12.03.1978 - 14.03.1978

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
12	10.9166	0.3875	1.5	910.225	5.80625	90.125	7.5	6.0	1.875
13	13.9166	0.3375	1.375	908.025	5.9525	94.375	8.625	7.0	0.875
14	7.3333	10.2	9	912.45	7.2875	59.75	5.75	3.625	1.75

7.07.1978 - 9.07.1978

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
7	8.9583	22.1125	28.0	910.2125	19.1375	58.375	1.375	0.625	1.25
8	8.29166	22.2	28.125	911.1	19.0125	62.375	1.5	0.5	1.125
9	3.375	12.3875	12.375	916.26	13.4375	93.5	9.75	9.375	1.75

15.03.1972 - 17.03.1972

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
15	16.4583	1.05	2.625	923.0375	4.1	58.25	0	0	0.5
16	13.5833	7.35	5.75	919.9125	4.7125	47.625	0	0	1.125
17	6.1667	8.6875	10.125	917.05	8.9	79.75	6.375	4.625	1.25

3.03.1986 - 5.03.1986

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
3	27.5	-3.6	-6.0	919.7625	2.64	58.875	8.125	7.25	1.125
4	24.625	0.825	-2.0	916.275	3.70625	59.125	8.5	8.5	0.375
5	22.75	1.2625	-2.625	915.8625	4.59375	69.625	6.5	5.625	0.375

8.12.1980 - 10.12.1980

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
8	10.5833	5.4	2.375	916.4125	6.4938	74.625	2.75	2.75	1.125
9	9.3333	4.25	0.375	912.775	4.7375	58.875	1.5	0.375	1.5
10	7.375	25.2	31.125	910.05	22.185	70.25	3.125	2.125	1.125

13.12.1990 - 15.12.1990

Число	Напряж эл. поля	Темпер воздуха	Темпер почвы	Атмосф давлен.	Влажн. абсол.	Влажн. относит	Облачн. общая	Облачн. нижняя	Скор. ветра
13	8.5	-0.1625	-0.125	920.975	4.6875	78.0	6.875	6.875	1.25
14	12.2083	0.625	-0.125	914.7125	5.0225	79.25	6.375	6.375	0.625
15	12.5	4.4	-0.125	914.275	5.44375	66.0	0.75	0.625	0.75