

с помощью командных файлов. Результаты записываются в формате CVS, что позволяет использовать для анализа данных табличный редактор.

Программа вызывается помощью команды `lmodgen`. Ей передаются три параметра: имя файла описания алфавита языка; имя файла с текстом, на основании которого будет построена модель; и имя файла, в котором будет сохранён результат.

Все файлы должны быть в формате Unicode, в кодировке UCS2 с прямым порядком байтов.

Файл описания алфавита представляет собой обычный текстовый файл, каждая строка которого представляет собой определённый символ в алфавите языка. Все варианты символа перечисляются в той же строке (подряд, без каких-либо разделителей). Файл текстом, по которому будет построена модель, также должен быть сохранён как обычный текст в формате Unicode (UCS2).

Результат будет сохранён как файл в формате CVS с двумя столбцами (сочетание трёх символов, частота) в кодировке UCS-2.

ЗАКЛЮЧЕНИЕ

В зависимости от поставленной задачи используются разные виды языковых моделей: для коррекции опечаток или распознавания голоса используются модели, работающие на уровне слов.

В своей работе мы написали программу, которая позволяет построить языковую модель на основании данных математической статистики и теории вероятностей. Программа позволит в будущем определить, насколько эффективен данный вид моделей, что требует дальнейшего исследования.

Литература

1. Интернет-адрес: http://language-model.narod.ru/full_report.doc.
2. *Зубов А. В., Зубова И. И.* Основы искусственного интеллекта для лингвистов. — М.: Университетская книга, Логос, 2007.
3. *Марчук Ю. Н.* Компьютерная лингвистика — М.: АСТ, Восток–запад, 2007.
4. Интернет-адрес: <http://language-model.narod.ru/lmodgen.zip>.

МЕХАНИЗМ ПРИНЯТИЯ РЕШЕНИЙ КЛЕТОЧНЫМ АВТОМАТОМ, ОСНОВАННЫМ НА НЕЧЁТКОЙ ЛОГИКЕ

Д. А. Кушнарёв, С. А. Филимонов

ВВЕДЕНИЕ

Наверное, самой впечатляющей способностью человеческого интеллекта является способность принимать правильные решения в условиях

неполной и нечеткой информации. Построение моделей приближенных размышлений человека и использование их в компьютерных системах представляет сегодня одну из важнейших проблем науки.

В последнее время нечеткая технология завоевывает все больше сторонников среди разработчиков систем управления. Простота и дешевизна разработки заставляет проектировщиков все чаще прибегать к этой технологии. Таким образом, «нечеткие» разработки стали весьма актуальны в наше время.

В результате объединения нескольких технологий искусственного интеллекта появился специальный термин – «мягкие вычисления» (*soft computing*), который ввел Л. Заде в 1994 году. В настоящее время мягкие вычисления объединяют такие области, как: нечеткая логика, искусственные нейронные сети, вероятностные рассуждения и эволюционные алгоритмы [1, 2, 3].

НЕЧЕТКАЯ ЛОГИКА

1. «Нечеткие» правила, ввод и управление

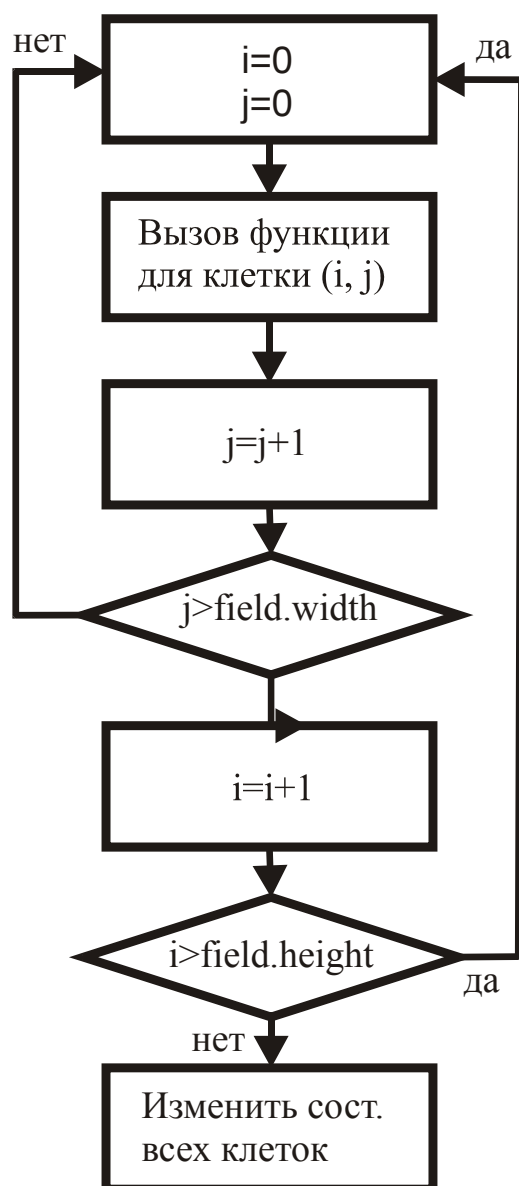
Классическая логика оперирует только двумя понятиями: *истина* и *ложь*, и исключая любые промежуточные значения. Аналогично этому булева логика не признает ничего кроме единиц и нулей. Решить эту проблему и призвана нечеткая логика.

С термином «*лингвистическая переменная*» можно связать любую физическую величину, для которой нужно иметь различные значения, а не только «да» и «нет». Вы можете определить необходимое число термов. Например, можно ввести переменную «*возраст*» и определить для нее термы «*юношеский*», «*средний*» и «*преклонный*» [4].

Одним из основных методов представления знаний в экспертных системах являются продукционные правила, позволяющие приблизиться к стилю мышления человека. Любое правило продукции состоит из посылок и заключения. Возможно наличие нескольких посылок в правиле, в этом случае они объединяются посредством логических связок **и**, **или**. В полном тексте работы процесс нечеткого управления можно ознакомиться более подробно [5].

КЛЕТОЧНЫЙ АВТОМАТ

Данная работа призвана проиллюстрировать пошаговое «принятие решений» простейшим клеточным автоматом. Развитие автомата будет осуществляться согласно специальным правилами, сформулированными пользователем.



Блок-схема. Алгоритм одного шага клеточного автомата

Клеточный автомат состоит из набора объектов (ячеек), обычно образующих регулярную решетку. Состояние отдельно взятого i -го объекта (или ячейки) в момент времени n характеризуется некоторой переменной. Состояния ячеек изменяются синхронным образом в соответствии с правилами, которые могут зависеть от состояния переменных в ближайших соседних узлах.

Отметим основные свойства классической модели клеточных автоматов: локальность правил; однородность системы; конечность множества возможных состояний клетки; одновременность изменения значений во всех клетках.

1. Программная реализация

Нами написала программа [6], которая является реализацией клеточного автомата с возможностью задания пользовательских правил, в том числе на основании нечёткой логики [7].

Правила перехода описываются пользователем на языке C, который затем компилируется в машинные коды, что позволяет добиться высокой скорости и гибкости работы. Для компиляции используется Tiny C Compiler.

Пользователь задаёт некоторую функцию, которая будет вызываться на каждом шагу (блок-схема) для каждой из клеток поля. Она возвращает значение типа **float** от 0 до 1.0.

В функции можно использовать следующие параметры, константы и функции: **int x** (координата текущей клетки на поле по оси Ox; нумерация начинается с нуля, что соответствует левому краю поля), **int y** (координата текущей клетки на поле по оси Oy; нумерация начинается с нуля, что соответствует верхнему краю поля), **int fw** (ширина поля), **int fh** (высота поля), **float f(int x, int y)** (значение клетки с заданными координатами). При

обращении к клетке за границами поля возвращаемое значение в данной реализации всегда равно 0.

2. Описание интерфейса

Центральное место в интерфейсе программы занимает поле, т. е. визуальное отображение текущего состояния решетки. Пользователь может в процессе работы изменять состояние поля левой кнопкой мыши. При этом в клетке, по которой будет произведён щелчок, будет устанавливаться определённое значение (по умолчанию 0), которое можно указать (в процентах) в поле внизу.

Пользователь может задать ширину и высоту решетки в клетках. По умолчанию решетка имеет размеры 100x100, после чего можно изменить размеры решетки (поля) и обнулить их значения с помощью кнопки «Создать поле».

Кнопка «Заполнить поле» позволяет заполнить решетку (поле) клетками со степенью заполненности, определенной псевдослучайным законом, чтобы оценить работу правила перехода в условиях, которые не были определены заранее.

Окно формулировки правила развития служит для описания правила развития пользователем. По умолчанию содержит правило постепенного усреднения значений. С помощью этого окна также можно определить начальные свойства поля по какому-то закону: например, чтобы очистить поле, можно использовать правило **return 0;**, после чего выполнить один шаг. Это позволяет проверить работу правила при определённых начальных условиях.

Кнопка «Шаг» позволяет произвести следующую итерацию.

Выводы

На сегодняшний день существует достаточно большое количество программных реализаций клеточных автоматов [8]. Их все объединяет то, что они основаны на принципах булевой логики. В докладе приведена программная реализация клеточного автомата на базе нечёткой логики. Её использование в конечных автоматах расширяет область применения и приближает к практически значимым программным продуктам.

Поскольку для практических задач типичной является неполнота данных, невозможно переоценить преимущества, предоставляемые нечёткой логикой. Дальнейшее исследование позволит оценить круг задач, решение которых может быть эффективным при реализации нечётких логик на машинах с архитектурой фон-Неймана.

Литература

1. *Круглов В. В., Дли М. И.* Интеллектуальные информационные системы: компьютерная поддержка систем нечеткой логики и нечеткого вывода. – М.: Физматлит, 2002.
2. *Рутковская Д., Пилиньский М., Рутковский Л.* Нейронные сети, генетические алгоритмы и нечеткие системы. – М., 2004.
3. *Масалович А.* Нечеткая логика в бизнесе и финансах. – Режим доступа: www.tora-centre.ru/library/fuzzy/fuzzy-.htm.
4. *Заде Л.* Понятие лингвистической переменной и его применение к принятию приближенных решений. – М.: Мир, 1976.
5. Интернет-адрес: <http://cellular-automaton.narod.ru/cellular-automaton.doc>.
6. Интернет-адрес: <http://cellular-automaton.narod.ru/CA.zip>.
7. *Kosko B.* Fuzzy systems as universal approximators // IEEE Transactions on Computers, vol. 43, No. 11, November 1994. – P. 1329–1333.
8. Интернет-адрес: <http://cafaq.com/soft/index.php>.

САМООЦЕНКА ВЛАДЕЛЬЦЕВ ДОМАШНИХ ЖИВОТНЫХ

Е. И. Ласовская

Анималотерапия (или зоотерапия) – это психотерапия, в которой психотерапевтическим средством является взаимодействие человека с животным. Такое взаимодействие существенно отличается от других психотерапевтических средств (взаимодействие с неодушевленными объектами в разных видах психотерапии, с самим терапевтом и участниками групповой терапии). Доктор психологических наук, профессор Г.Г. Филиппова выделяет следующие основные отличия такого взаимодействия в рамках анималотерапии:

1. животное имеет внутренний источник активности, в результате чего становится относительно непредсказуемым как для пациента, так и для терапевта;

2. животное не является носителем смысловой и оценочной функции по отношению к человеку, оно не может быть в ситуации психотерапии «неконгруэнтным», т. е. оно всегда выражает то, что непосредственно чувствует, и это позволяет пациенту освободиться от проявления психологических защит, препятствующих эффективности психотерапии;

3. животное имеет «внутреннюю позицию», собственную мотивацию и включается в процесс взаимодействия с человеком, активно эту позицию проявляя. При психотерапии с помощью «неодушевленных» средств все строится только во взаимодействии позиций пациента и психотерапевта.

Содержание животных в домашних условиях издавна используется человеком, как это принято говорить, «для души», т. е. для эмоциональ-