

**Филатова Наталья Николаевна** – Тверской государственный технический университет; e-mail: nfilatova99@mail.ru; 170026, г. Тверь, наб. Аф. Никитина, 22, тел.: 84822444488; кафедра автоматизации технологических процессов; д.т.н.; профессор.

**Сидоров Константин Владимирович** – e-mail: bmisidorov@rambler.ru; тел.: 89607038330; кафедра автоматизации технологических процессов; аспирант.

**Filatova Natalia Nicolaevna** – Tver State Technical University; e-mail: nfilatova99@mail.ru; 22, Afanasy nikitin street, Tver, 170026, Russia; phone: +74822444488; the department of automation technological processes; dr. of eng. sc.; professor.

**Sidorov Constantine Vladimirovich** – e-mail: bmisidorov@rambler.ru; phone: +79607038330; the department of automation technological processes; postgraduate student.

УДК 004.93'1

**Д.М. Ханеев, Н.Н. Филатова**

### **ПИРАМИДАЛЬНАЯ СЕТЬ ДЛЯ КЛАССИФИКАЦИИ ОБЪЕКТОВ, ПРЕДСТАВЛЕННЫХ НЕЧЕТКИМИ ПРИЗНАКАМИ**

*Приводится результат исследования пирамидальной сети применительно к задаче классификации объектов, представленных нечеткими признаками в области биомедицинских систем. Рассмотрена методика построения пирамидальной сети и способ выделения из ее структуры нечетких описаний классов объектов, представленных в обучающей выборке. Полученные с помощью сети описания классов могут быть использованы для последующей классификации объектов рабочей выборки. Рассмотрено применение пирамидальной сети в задаче автоматической классификации образцов дыхательных шумов, заданных спектральными признаками.*

*Классификация; распознавание; дыхательные шумы; нечеткая логика; теория графов.*

**D.M. Khaneyev, N.N. Filatova**

### **THE PYRAMIDAL NETWORK FOR CLASSIFICATION OF OBJECTS, PRESENTED BY FUZZY FEATURES**

*This paper presents the research results of the pyramidal network applied to the problem of classification of objects presented fuzzy features in the field of biomedical systems. The technique of building pyramidal network and an extraction method from this structure the fuzzy descriptions of object classes presented in training set is considered. Classes' descriptions which obtained through the network may be used for the working sample objects classifying. The application of a pyramidal network in problem of automatic classification of respiratory noise samples, presented by spectra, is considered.*

*Classification; pattern recognition; respiratory noises; fuzzy logic; graph theory.*

Автоматическая классификация объектов в области биомедицинских систем является сложной задачей в силу множества зависимостей от различных факторов, поэтому ориентироваться на точные значения признаков объектов нельзя. Следовательно, для классификации сложных нестационарных объектов необходимо использовать аппарат нечетких множеств. Такой подход предполагает составление нечетких описаний классов, на основании которых можно судить о степени принадлежности объектов рабочей выборки к тому или иному классу. В качестве структуры, генерирующей нечеткие описания классов, предлагается нейроподобная сеть аналогичная растущей пирамидальной сети (РПС) [1, 2]. Из достоинств пирамидальных сетей следует отметить, что структуру обученной сети, в отличие от большинства искусственных нейронных сетей, можно легко интерпретировать в

набор логических правил, пригодных для дальнейшего анализа, сеть устойчива к переобучению, а также позволяет работать с признаковыми пространствами больших размеров.

Пирамидальные сети ориентированы на обработку фиксированных значений признаков объектов. В нашем случае признаки представлены лингвистическими переменными (ЛП), принимающими значения из заданного терм-множества. Пирамидальная сеть представляет собой ориентированный ациклический граф в ярусно параллельной форме (ЯПФ), не имеющий вершин с одной заходящей дугой, все дуги ориентированы от нижних уровней сети к верхним. Вершины сети разделены на два типа, вершины первого типа называются рецепторами, вершины второго типа – ассоциативными элементами. Рецепторы являются входами сети, каждому нечеткому значению признака (терму) в соответствие ставится свой рецептор сети, в результате чего количество рецепторов равно сумме мощностей терм-множеств среди всех ЛП-признаков. Ассоциативные элементы по своему функционированию аналогичны логическому «И». Все вершины сети могут принимать только бинарные значения (активен / неактивен). Для возбуждения рецептора требуется внешний сигнал, ассоциативный элемент возбуждается, если на всех его входах присутствует сигнал возбуждения [3].

Алгоритм построения пирамидальной сети будем рассматривать применительно к задаче классификации дыхательных шумов (ДШ). Обучающая и рабочая выборки сети заданы множеством векторов спектральных характеристик образцов ДШ, рассчитанных с помощью быстрого преобразования Фурье. Каждый вектор представляет один объект выборки и имеет вид  $X = \{x_1, x_2, \dots, x_i, \dots, x_m\}$ , где  $x_i$  – ордината спектра на частоте  $f_i = \Delta x_i \cdot i$ , характеризующая выделение энергии на определенном интервале частот, соответствует одному признаку;  $m$  – размерность признакового пространства. Каждый объект обучающей выборки (ОВ) помечен принадлежностью к классу «норма» или «патология».

Каждому спектральному признаку ставится в соответствие своя ЛП «выделение энергии на частоте  $f_i$ », границы базового множества которой определены экстремумами значений признака среди объектов ОВ. Терм-множество каждой лингвистической переменной содержит три термина: «Низкий», «Средний», «Высокий», функции принадлежности которых имеют вид, представленный на рис. 1, где

$d_{ki\max} = \max_{j \in K} (x_{ji})$ ;  $d_{ki\min} = \min_{j \in K} (x_{ji})$ ;  $b = d_{i\min} + \frac{d_{i\max} - d_{i\min}}{2}$ ;  $i = \overline{1, n}$ ;  $K$  – количество объектов в ОВ;  $n$  – количество признаков.

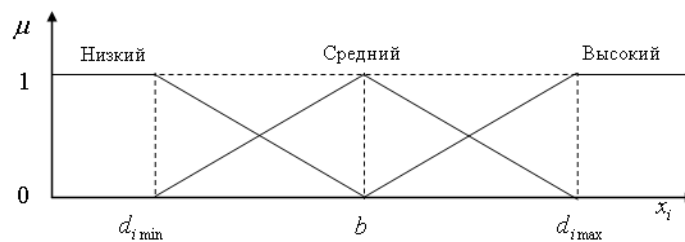


Рис. 1. Функции принадлежности термов ЛП «выделение энергии на частоте  $f_i$ »

В алгоритме построения сети используются следующие термины:

- ◆ субмножество вершины – множество вершин, из которых существует путь к данной вершине;

- ◆ 0-субмножество вершины – множество вершин смежных с данной вершиной, из которых существует путь к данной вершине;
- ◆ супермножество вершины – множество вершин, к которым имеются пути из данной вершины;
- ◆ 0-супермножество вершины – множество вершин смежных с данной, к которым имеются пути из данной вершины;
- ◆ пирамида вершины  $v_l$  – подграф пирамидальной сети, включающий вершину  $v_l$  и все вершины, от которых имеются пути к вершине  $v_l$ .

Каждый ассоциативный элемент характеризуется параметрами:

- ◆ уровнем вершины в иерархии сети;
- ◆ состоянием (возбужден или нет);
- ◆ классом объекта ОВ, породившего вершину;
- ◆ параметром  $l$ , характеризующим число рецепторов в пирамиде ассоциативного элемента;
- ◆ значениями счетчиков возбуждений (счетчики  $m_{cl_1} - m_{cl_k}$ , определяют реакцию вершины на входные объекты ОВ классов  $cl_1 - cl_k$  соответственно, где  $k$  – количество классов, представленных в ОВ).

Не обученная сеть состоит из рецепторов, соединенных исходящими дугами с заключительной вершины верхнего уровня  $Y$ . Вершина  $Y$  выполняет вспомогательную роль в процессе генерации структуры сети. Построение сети выполняется в два последовательных этапа. На первом этапе формируются конъюнктивные зависимости среди значений признаков. На втором этапе выделяется множество контрольных элементов. Контрольные элементы представляют собой ассоциативные элементы с наиболее часто встречающимся набором значений признаков (паттерном), характерных для объектов определенного класса. Рецепторы, из которых существует путь к контрольному элементу, представляют собой паттерн. Паттерн – часто встречающийся характерный набор значений признаков вида  $p_1 \text{ is } T_1 \wedge p_2 \text{ is } T_1 \wedge \dots \wedge p_n \text{ is } T_k$ , где  $p_i | i = \overline{1, n}$  – признаки;  $n$  – количество признаков;  $T_j | j = \overline{1, k}$ ;  $k$  – количество термов лингвистической переменной признака.

Первый этап алгоритма заключается в последовательном просмотре всех объектов ОВ сначала по правилу РК1, а затем по правилу РК2. После каждой подачи на вход сети нового объекта, либо после изменения структуры сети, происходит распространение возбуждения по всем вершинам.

Правило РК1 (рис. 2). Поочередно просматриваются все невозбужденные вершины сети, начиная с нижнего уровня иерархии и заканчивая верхним. У просматриваемой вершины  $v_i$  выбираются активные вершины ее 0-субмножества и заносятся во множество  $Va$ . Если количество элементов множества  $Va$  меньше, чем 2, то выполняется переход к следующей просматриваемой вершине  $v_{i+1}$ , иначе – в сеть вводится новый ассоциативный элемент  $v_n$ , промаркированный классом, к которому принадлежит поданный на вход сети объект ОВ. Дуги от вершин из множества  $Va$  к  $v_i$  ликвидируются, вместо них вводятся дуги, соединяющие вершины из  $Va$  с новой вершиной  $v_n$ . Выход нового элемента  $v_n$  соединяется исходящей дугой с входом  $v_i$ . После этого уровни всех вершин, к которым существуют пути из нового ассоциативного элемента  $v_n$ , рекурсивно повышаются на 1 (граф приводится в ЯПФ), и выполняется переход к следующей просматриваемой вершине  $v_{i+1}$ .

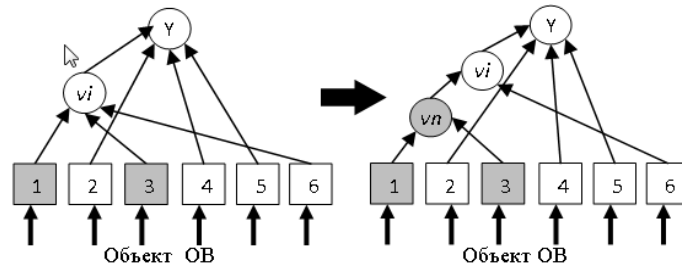


Рис. 2. Правило PK1

Правило PK2 (рис. 3). Поочередно просматриваются все возбужденные вершины сети, кроме заключительной вершины верхнего уровня  $Y$ . У просматриваемой вершины  $v_i$  находится 0-супермножество. Если в 0-супермноестве нет ни одной возбужденной вершины, то  $v_i$  добавляется во множество  $S$ , и выполняется переход к следующей просматриваемой вершине. После просмотра всех вершин и формирования множества  $S$  в сеть добавляется новый ассоциативный элемент  $v_n$ , промаркированный классом, к которому принадлежит поданный на вход сети объект ОВ. На вход элемента  $v_n$  заводятся дуги от вершин множества  $S$ , а исходящая дуга от  $v_n$  соединяется с заключительной вершиной верхнего уровня сети  $Y$ , причем у элементов из множества  $S$  ликвидируются исходящие дуги к вершине  $Y$ . После добавления новой вершины распространяется возбуждение по сети и обновляются счетчики у каждого возбуждившегося ассоциативного элемента.

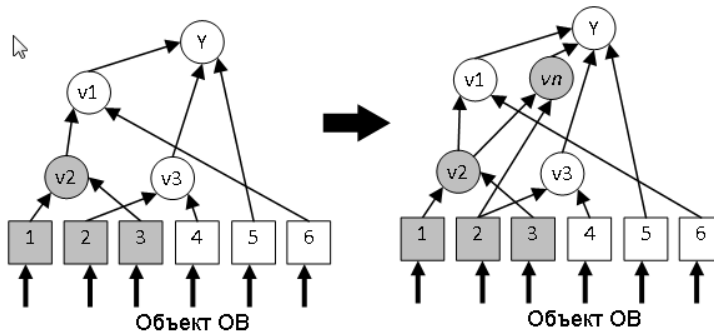


Рис. 3. Правило PK2

В результате построений первого этапа ассоциативные элементы, смежные с вершиной  $Y$ , образуют множество  $V_c$ , с каждым элементом которого ассоциируется объект из обучающей выборки с уникальными комбинациями значений нечетких признаков.

Второй этап построения пирамидальной сети состоит из однократного последовательного выполнения правил R1 и R2. В процессе выполнения правил второго этапа корректируются значения параметров вершин  $m_{cl1} - m_{clk}$  и  $l$ .

**Правило R1.** Просматривается каждая вершина  $vc$  из множества  $V_c$ . У просматриваемой вершины  $vc$  находится подмножество  $V_{cs}$ , класс объекта, ассоциированного с  $vc$ , запоминается как  $cli$ . Из  $V_{cs}$  выделяются вершины с наибольшим количеством рецепторов в своей пирамиде и заносятся во множество  $V_{sk}$ , из кото-

рого выделяется подмножество вершин  $Vskm$  с максимальными значениями счетчика срабатываний  $m_{cli}$  для класса  $cli$ . Вершины из множества  $Vskm$  помечаются как контрольные элементы класса  $cli$  и добавляются во множество контрольных элементов сети.

**Правило R2.** Просматривается каждая вершина  $vc$  из множества  $Vcs$ . У просматриваемой вершины  $vc$  находится подмножество  $Vcs$ , класс объекта, ассоциированного с  $vc$ , запоминается как  $cli$ . Из  $Vcs$  выделяется множество вершин  $Vf$ , являвшихся контрольными элементами классов отличных от  $cli$ . У каждой вершины из  $Vf$  находятся супермножества, из которых выбираются в множество  $Vft$  вершины, состоящие в  $Vcs$ . Если в  $Vft$  нет ни одного контрольного элемента класса  $cli$ , то из него выделяется подмножество вершин  $Vftm$  с максимальными значениями счетчика срабатываний  $m_{cli}$ . Затем из  $Vftm$  выделяется подмножество вершин  $Vftmk$ , имеющих наибольшее количество рецепторов в своих пирамидах. Элементы из  $Vftmk$ , входящие в  $Vcs$ , и не являющиеся контрольными, помечаются как контрольные элементы класса  $cli$  и заносятся во множество контрольных элементов сети.

По завершению второго этапа из всех ассоциативных элементов сети будет выделено множество контрольных элементов.

Путем интерпретации структуры обученной пирамидальной сети, для каждого класса, представленного в ОБ, строится продукционное правило вида *IF antecedent THEN consequent*. Где *antecedent* есть конъюнктивно-дизъюнктивное высказывание вида  $паттерн_1 \vee паттерн_2 \vee \dots паттерн_n$ ; *consequent* – заключение об отнесении объекта к определенному классу вида  $CLASS\ is\ B_c$ , где  $CLASS$  – лингвистическая переменная нечеткого вывода;  $B_c$  – терм, представляющий  $c$ -й класс;  $c = \overline{1, L}$ ;  $L$  – количество классов в обучающей выборке.

Полученные продукционные правила позволяют посредством системы нечеткого вывода судить о степени принадлежности объектов тестовой или рабочей выборки к классам, представленным в ОБ.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Гладун В.П.* Растущие пирамидальные сети // *Новости искусственного интеллекта.* – 2004. – № 1. – С. 30-40.
2. *Гладун В.П.* Партнерство с компьютером – Киев: Port-Royal, 2000. – С. 17-44.
3. *Поспелов Д.А.* Ситуационное моделирование. Теория и практика. – М.: Наука, 1986. – С. 174-182.

Статью рекомендовал к опубликованию к.т.н. Ф.Ф. Филиппов.

**Филатова Наталья Николаевна** – Тверской государственный технический университет; e-mail: nfilatova99@mail.ru; 170026, г. Тверь, наб. Аф. Никитина, 22, тел.: 84822444488; кафедра автоматизации технологических процессов; д.т.н.; профессор.

**Ханеев Дмитрий Михайлович** – e-mail: t\_ran@mail.ru; 170021, г. Тверь, ул. Хрустальная, 43/161; тел.: 89201560175; кафедра автоматизации технологических процессов; аспирант.

**Filatova Natalia Nicolaevna** – Tver State Technical University; e-mail: nfilatova99@mail.ru; 22, Afanasy Nikitin Street, Tver, 170026, Russia; phone: +74822444488; the department of automation technological processes; dr. of eng. sc.; professor.

**Khaneyev Dmitry Michailovich** – e-mail: t\_ran@mail.ru; 43/161, Khrustalnaya street, Tver, 170021, Russia; phone: +7201560175; the department of automation technological processes; postgraduate student.