

---

# НЕЙРОИНФОРМАТИКА ТА ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ

## НЕЙРОИНФОРМАТИКА И ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

### NEUROINFORMATICS AND INTELLIGENT SYSTEMS

---

УДК 519.168:004.658

Асеев Г. Г.

*Д-р техн. наук, профессор, заведующий кафедрой Харьковской государственной академии культуры*

---

#### МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ В ЭЛЕКТРОННЫХ ХРАНИЛИЩАХ: ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ

---

Представлен один из возможных методов интеллектуального анализа данных в электронных хранилищах большого объема – генетические алгоритмы и их модификация.

**Ключевые слова:** интеллектуальный анализ, электронное хранилище, нейронная сеть, генетические алгоритмы.

#### ВВЕДЕНИЕ

В настоящее время в электронных хранилищах данных (ХД) корпоративных информационных систем хранятся терабайты различной текстовой и числовой информации. Для обнаружения, извлечения и интеллектуального анализа этих данных используются методы Knowledge Discovery in Databases и Data mining [1]. В [1] были описаны некоторые рекомендации, следуя которым можно подготовить качественные данные в нужном объеме для анализа: первичные источники данных, хранение данных, подготовка исходного набора данных, предобработка и очистка исходных данных [2], трансформация, нормализация, выдвижение гипотез и построение модели Data Mining [3]. Данная работа продолжает цикл статей, посвященных методам интеллектуального анализа данных в электронных хранилищах большого объема, в частности модификации генетических алгоритмов.

#### ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ – МАТЕМАТИЧЕСКИЙ АППАРАТ

Такие свойства генетических алгоритмов, как адаптивность, робастность, возможность распараллеливания вычислений и отыскание глобального экстремума принятой функции приспособленности, обеспечили их эф-

фективное использование для решения различных задач в пространствах высокой размерности в ХД. Примером подобной задачи может служить обучение нейросети, то есть подбора таких значений весов, при которых достигается минимальная ошибка.

Из биологии мы знаем, что любой организм может быть представлен своим *фенотипом*, который фактически определяет, чем является объект в реальном мире, и *генотипом*, который содержит всю информацию об объекте на уровне хромосомного набора. При этом каждый ген, то есть элемент информации генотипа, имеет свое отражение в фенотипе. Разработчик генетических алгоритмов выступает в данном случае как «создатель», который должен правильно установить законы эволюции, чтобы достичь желаемой цели как можно быстрее. Впервые эти нестандартные идеи были применены к решению оптимизационных задач в середине 70-х годов [4]. Примерно через десять лет появились первые теоретические обоснования этого подхода [5, 6]. В дальнейшем генетические алгоритмы доказали свою конкурентоспособность при решении многих *NP*-трудных задач [7] и особенно в практических приложениях, где математические модели имеют сложную структуру и применение стандартных методов типа ветвей и границ, динамического или линейного программирования крайне затруднено.

В наиболее часто встречающейся разновидности генетического алгоритма для представления генотипа объекта применяются битовые строки. При этом каждому атрибуту объекта в фенотипе соответствует один ген в генотипе объекта. Ген представляет собой битовую строку, чаще всего фиксированной длины, которая представляет собой значение этого признака.

Генетический алгоритм работает с представленными в конечном алфавите строками  $S$  конечной длины  $l$ , которые используются для кодировки исходного множества альтернатив  $W$ . Строки представляют собой упорядоченные наборы из  $l$  элементов:  $S=(s_1, s_2, \dots, s_l)$ , каждый из которых может быть задан в своем собственном алфавите  $V_i, i = \overline{1, L}$ , где алфавит  $V_i$  является множеством из  $r_i$  символов:  $V_i = \{v_{ij}, j = \overline{1, r_i}\}$ . Для решения конкретной задачи требуется однозначно отобразить конечное множество альтернатив  $W$  на множество строк подходящей длины (очевидно, что длина строк зависит от алфавитов, используемых для их задания).

Для работы алгоритма необходимо на множестве строк  $U^m(V_1, V_2, \dots, V_m)$  задать неотрицательную функцию  $F(S)$ , определяющую показатель качества, «ценность» строки  $SO U^m(V_1, V_2, \dots, V_m)$ . Алгоритм производит поиск строки, для которой

$$F^*(S) = \arg \max_{S \in U^m(V_1, V_2, \dots, V_m)} F(S)$$

Если на множестве  $W$  задана целевая функция  $f(w)$ , то функцию  $F(S)$  на множестве строк  $U^m(V_1, V_2, \dots, V_m)$  можем определить следующим образом:  $F(S) = f(w)$ , если элемент  $w$  при отображении исходного множества  $W$  на множество строк был сопоставлен строке  $S$ .

Генетический алгоритм за один шаг производит обработку некоторой популяции строк. Популяция  $G(t)$  на шаге  $t$  представляет собой конечный набор строк:

$$G(t) = (S_1^t, S_2^t, \dots, S_N^t), S_k^t \in U^m(V_1, V_2, \dots, V_m), k = \overline{1, N},$$

где  $N$  – размер популяции, причем строки в популяции могут повторяться.

Анализ работы алгоритма удобно производить, используя аппарат схем. Схемой в генетическом алгоритме называют описание некоторого подмножества строк. Схема  $H=(h_1, h_2, \dots, h_m)$  может рассматриваться как строка, алфавиты для элементов которой дополнены специальным символом «#»:

$$H \in U^m(V_1^H, V_2^H, \dots, V_m^H), V_i^H = V_i \cup \{ \# \}$$

Если в некоторой позиции  $r$  схемы  $H$  присутствует символ «#», то такая позиция называется свободной, а сам символ «#» интерпретируется как произвольный символ из алфавита  $V_r$ . Позиция  $q$  схемы  $H$  называется

фиксированной, если в этой позиции присутствует один из символов алфавита  $V_q$ . Схема  $H$ , в которой определены фиксированные и свободные позиции, описывает подмножество  $U_H \subseteq U^m(V_1, V_2, \dots, V_m)$ , содержащее такие строки, у которых элементы, соответствующие фиксированным позициям схемы, совпадают с символами схемы, а элементы, соответствующие свободным позициям схемы, являются произвольно заданными в соответствующих алфавитах:

$$U_H = \left\{ S \mid S \in U^m(V_1, V_2, \dots, V_m) \wedge (\forall i (i \in I_{[1, m]} \wedge h_i \neq \{ \# \} \rightarrow (s_i = h_i))) \right\},$$

где  $I_{[1, m]}$  – множество целых чисел отрезка  $[1, m]$ .

Например, для множества строк  $U_{(V_1, V_2, V_3, V_4, V_5)}^5$ , где  $U_i = \{0, 1\}$ ,  $V_i = \overline{1, 5}$ , схема  $H_1 = \langle 1 \# \# \# 0 \rangle$  задает такое множество строк, у которых первым элементом является символ «1», пятым – «0», а остальные – либо «0», либо «1». Строки «10010», «11110» являются примерами строк, принадлежащих множеству  $U_{H_1}$ .

Часть популяции  $G(t) = (S_1^t, S_2^t, \dots, S_N^t)$ , строки которой удовлетворяют схеме  $H$ , обозначают  $G_H(t) = (S_1^{H,t}, S_2^{H,t}, \dots, S_{n(H,t)}^{H,t})$ , где  $n(H, t)$  – число строк схемы  $H$  в популяции  $G(t)$ , и называют подпопуляцией, соответствующей схеме  $H$ .

### ПРОЦЕДУРА ОПТИМИЗАЦИИ

В общем случае процедура оптимизации на основе обычного последовательного комплекс-метода выглядит следующим образом: требуется отыскать минимум некоторой функции, как правило, многоэкстремальной:

$$E(x) \rightarrow \min_{x \in R^n}$$

достаточно общего вида, при этом, о характере этой функции не делается практически никаких априорных предположений.

Будем использовать функции приспособленности следующего вида:

$$E(x) = \sum_{cl} E(cl) = \sum_{cl} \sum_d w_{cl,d} = \sum_{cl} \sum_d sf_{cl,d} \times \times ids_{cl,d} = \sum_{cl} \sum_d sf_{cl,d} \times \log \frac{N}{ds_{cl}},$$

где  $E(cl)$  – функция приспособленности для хромосомы  $cl$ ;  $w_{cl,d}$  – нормализованные данные о хромосоме  $c$  для документа  $d$ ;  $sf_{cl,d}$  – частота встречаемости терма (или набора термов), представленного хромосомой  $cl$ ;  $ids_{cl,d}$  – инверсная частота встречаемости терма (или

набора термов), представленного хромосомой  $cl$ ;  $ds_{cl}$  – число документов, содержащих комбинации хромосомы  $cl$ ;  $N$  – общее число документов в ХД.

Все данные должны быть представлены в двоичном коде (1 – если терм (или набор термов) содержится в документе, 0 – в противном случае). В ХД используются хромосомы, максимальная длина которых составляет до нескольких сотен генов, причем некоторые из них могут быть пустыми. Согласно этому утверждению, число повторяющихся термов в решении может варьироваться от 2 до нескольких сотен. Поскольку пространство решений очень большое, предлагается использовать мутацию, фиксированную между 50 и 70 процентами, и в конечном итоге каждая хромосома будет подвержена мутации для новой популяции.

Работа алгоритма начинается с формирования начального комплекса

$$x_i(0) = (x_{i1}(0), x_{i2}(0), \dots, x_{ij}(0), \dots, x_{in}(0))^T, \quad i = 1, 2, \dots, N \geq n + 1,$$

представляющего собой «облако» (популяцию) точек (векторов), достаточно произвольно расположенных в  $n$ -мерном пространстве факторов. Среди множества этих точек находится «наихудшая»  $x_N(0)$ , в которой значение функции  $E(x_N(0))$  максимально, после чего эта точка отражается через центр тяжести всех остальных вершин-точек, формируя новый комплекс  $x_N(1)$ ,  $i = 1, 2, \dots, N$ . Такое отражение вместе с растяжением и сжатием обеспечивают движение комплекса к экстремуму функции  $E(x)$ , при этом, благодаря достаточно случайному распределению точек «облака», поиск имеет глобальный характер.

С формальной точки зрения, рассмотрим процесс оптимизации на  $k$ -й итерации поиска, когда сформирован комплекс  $x_i(k)$ ,  $i = 1, 2, \dots, N$ . Среди множества точек  $x_i(k)$  находится «наихудшая», такая, что

$$E(x_N(k)) = \max_i \{E(x_1(k)), \dots, E(x_N(k))\},$$

после чего определяется центр тяжести «облака» без наихудшей точки:

$$x_C(k) = \frac{1}{N-1} \left( \sum_{i=1}^N x_i(k) - x_N(k) \right).$$

Далее  $x_N(k)$  отражается через центр тяжести  $x_C(k)$ , формируя новую вершину комплекса  $x_R(k)$ , которая теоретически расположена ближе к экстремуму, чем  $x_N(k)$  и  $x_C(k)$ , т. е.

$$E(x_R(k)) < E(x_C(k)) < E(x_N(k)).$$

Операция отражения формально имеет следующий вид:

$$x_R(k) = x_C(k) + \eta_R(x_C(k) - x_N(k)) =$$

$$= \frac{1}{N-1} x_1(k) + \dots + \frac{1}{N-1} x_{N-1}(k) + \frac{\eta_R}{N-1} x_1(k) + \dots \\ \dots + \frac{\eta_R}{N-1} x_{N-1}(k) - \eta_R x_N(k) = X(k)R,$$

где  $\eta_R$  – параметр шага отражения, часто полагаемый равным единице,  $X(k) = (x_1(k), x_2(k), \dots, x_{N-1}(k))$  –  $(n \times N)$  – матрица координат вершин комплекса,

$$R = \left( -\eta_R, \frac{1+\eta_R}{N-1}, \dots, \frac{1+\eta_R}{N-1} \right)^T - (N \times 1) - \text{вектор.}$$

В случае, если отраженная вершина  $x_R(k)$  окажется «наилучшей» среди всех остальных точек комплекса, т. е.:

$$E(x_R(k)) < E(x_C(k)) < E(x_N(k)), \quad i = 1, 2, \dots, N-1,$$

производится операция растяжения комплекса в направлении от центра тяжести  $x_C(k)$  до  $x_R(k)$  согласно выражению

$$x_E(k) = x_C(k) + \eta_E(x_R(k) - x_C(k)) = X(k)E,$$

где  $\eta_E$  – параметр шага растяжения, часто полагаемый равным двум:

$$E = \left( -\eta_E \eta_R, \frac{1-\eta_E(1-\eta_R)}{N-1}, \dots, \frac{1-\eta_E(1-\eta_R)}{N-1} \right)^T.$$

Если же  $x_R(k)$  окажется наихудшей среди всех  $x_i(k)$ , комплекс сжимается согласно соотношению

$$x_S(k) = x_C(k) + \eta_S(x_R(k) - x_C(k)) = X(k)S,$$

где  $\eta_S$  – параметр шага сжатия, обычно полагаемый равным 0,5:

$$S = \left( -\eta_S \eta_R, \frac{1-\eta_S(1-\eta_R)}{N-1}, \dots, \frac{1-\eta_S(1-\eta_R)}{N-1} \right)^T.$$

При  $\eta_S = 1$ ,  $\eta_E = 2$ ,  $\eta_S = 0,5$  приходим к простым выражениям:

$$R = \left( -1, \frac{2}{N-1}, \dots, \frac{2}{N-1} \right)^T, \quad E = \left( -2, \frac{1}{N-1}, \dots, \frac{1}{N-1} \right)^T,$$

$$S = \left( -0,5, \frac{1}{N-1}, \dots, \frac{1}{N-1} \right)^T.$$

Таким образом, в процессе своего движения к экстремуму оптимизируемой функции комплекс на каждой итерации теряет одну наихудшую вершину и приобретает одну новую точку так, что на  $(k+1)$ -й итерации новый комплекс также имеет  $N$  точек-вершин.

В генетических алгоритмах в результате селекции из популяции одновременно исключаются несколько особей с наихудшими (максимальными) значениями функции приспособленности. В связи с этим представляется целесообразным ввести алгоритм комплекс-метода с отражением, растяжением и сжатием сразу нескольких вершин [8, 9].

Итак, пусть на  $k$ -й итерации процесса оптимизации имеется комплекс  $x_i(k), i = 1, 2, \dots, N$  с  $P < N$  наихудшими вершинами  $x_{H_p}(k), p = 1, 2, \dots, P$ . Тогда координаты центра тяжести комплекса без вершин  $x_{H_p}(k)$  задаются выражением

$$x_C(k) = \frac{1}{N-P} \left( \sum_{i=1}^N x_i(k) - \sum_{p=1}^P x_{H_p}(k) \right),$$

а процедура отражения описывается системой уравнений

$$\begin{cases} x_{R_1}(k) = x_C(k) + \eta_R(x_C(k) - x_{H_1}(k)), \\ \vdots \\ x_{R_P}(k) = x_C(k) + \eta_R(x_C(k) - x_{H_P}(k)). \end{cases}$$

В случае, если среди отраженных вершин оказывается  $Q \leq P$  наилучших, комплекс растягивается в их направлении согласно уравнениям

$$\begin{cases} x_{E_1}(k) = x_C(k) + \eta_E(x_{R_1}(k) - x_C(k)), \\ \vdots \\ x_{E_Q}(k) = x_C(k) + \eta_E(x_{R_Q}(k) - x_C(k)). \end{cases}$$

Если, далее, среди отражаемых вершин окажется  $U \leq P$  наихудших, комплекс сжимается в их направлении согласно уравнениям

$$\begin{cases} x_{S_1}(k) = x_C(k) + \eta_S(x_{R_1}(k) - x_C(k)), \\ \vdots \\ x_{S_U}(k) = x_C(k) + \eta_S(x_{R_U}(k) - x_C(k)). \end{cases}$$

Таким образом, комплекс-метод приобретает черты генетического алгоритма, у которого в результате селекции на каждой итерации из популяции удаляется несколько наихудших особей.

Объединяя введенную модификацию комплекс-метода с голландской генетической процедурой, приходим к алгоритму, реализующему идею искусственного отбора, состоящую в данном случае в том, что из популяции не только удаляются наихудшие особи, но и одновременно создаются их «антиподы», обладающие улучшенными свойствами.

Работа такого алгоритма образована последовательностью следующих шагов:

- создание начальной популяции, образованной  $P(0)$  особями хромосомами – вершинами комплекса;
- операция кроссовера с увеличением популяции  $P_{CR}(0) > P(0)$ ;
- операция мутации  $P_M(0) > P_{CR}(0)$ ;
- операция инверсии  $P_I(0) > P_M(0)$ ;
- первая селекция (определение наихудших особей) без сокращения популяции  $P_{SEL1}(0) = P_I(0)$ ;
- операция отражения с удалением  $P$  наихудших особей  $P_R(0) < P_{SEL1}(0)$ ;
- операция растяжения без увеличения популяции  $P_E(0) = P_R(0)$ ;
- операция сжатия без увеличения популяции  $P_I(0) = P_E(0)$ ;
- вторая селекция с удалением  $P_W(0)$  наихудших особей  $P_{SEL2}(0) = P_I(0) = P(1)$  и формирование популяции  $P(1)$  для следующей итерации алгоритма.

## ВЫВОДЫ

Описанный в разделе математический аппарат голландских генетических алгоритмов имеет ряд недостатков. В частности, они характеризуются низкой скоростью сходимости, не позволяющей им отыскивать решение за приемлемое время. Также генетические алгоритмы являются чувствительными к выбору параметров алгоритма, например, размера популяции, вероятностей кроссовера и мутации и т. п.

Эти и некоторые другие особенности генетических алгоритмов послужили толчком к созданию их различных модификаций. В некоторых модификациях, например, предлагается использовать, кроме классических генетических операторов кроссовера, мутации и инверсии дополнительные операторы. Например, такие как операторы объединения (fusion) и разделения (fission). Операция объединения заключается в том, что два аллеля соединяются в один. Операция разделения предполагает замену одного аллеля другим случайным аллелем. В результате происходит разделение кластеров [10].

В основе рассматриваемого алгоритма лежит синтез обычного эволюционного генетического подхода с идеями адаптивной оптимизации и, прежде всего, последовательного комплекс-метода отыскания экстремума функций многих переменных. При этом в каждый момент времени текущая популяция отождествляется с «облаком» – комплексом точек в пространстве переменных-факторов, а кроме традиционных генетических операторов мутации, кроссовера и инверсии дополнительно вводятся операторы комплекс-поиска, такие как отражение, растяжение и сжатие. Работа предложенного алгоритма протестирована на выборке Reuters-21578 [8, 9]. Было установлено, что предложенный алгоритм работает быстрее и дает более точные результаты (в среднем 8–10 %) по сравнению со стандартными генетическими алгоритмами.

## СПИСОК ЛІТЕРАТУРЫ

1. *Асеев, Г. Г.* Проблема обнаружения нового знания в хранилищах данных методами Knowledge Discovery in Databases / Г. Г. Асеев // Вестник НТУ «ХПИ». – 2006. – № 19. – С. 62–70.
2. *Асеев, Г. Г.* Методы интеллектуальной предобработки данных в электронных хранилищах / Г. Г. Асеев // Радиоелектроніка, інформатика, управління. – 2010. – № 2(23). – С. 106–111.
3. *Асеев, Г. Г.* Методы интеллектуального анализа данных в электронных хранилищах / Г. Г. Асеев // Бионика интеллекта : науч.-техн. журнал. – 2008. – № 1(70). – С. 28–33.
4. *Растринин, Л. А.* Случайный поиск – специфика, этапы истории и предрасудки / Л. А. Растринин // Вопросы кибернетики. – Вып. 33. – 1988. – С. 3–12.
5. *Holland, J. H.* Adaptation in natural and artificial systems / John H. Holland. – Ann Arbor : University of Michigan Press, 1985. – 305 p.
6. *Rechenberg, I.* Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der Biologischen Information / Rechenberg I. – Freiburg : Fromman, 1983. – P. 135–143.
7. *Goldberg, D. E.* Genetic algorithms in search, optimization, and machine learning / David E. Goldberg. – [USA] : Addison-Wesley, 1989. – 752 p.
8. *Волкова, В. В.* Возможностная фаззи-кластеризация текстовых массивов в реальном времени на основе самообучающейся нейронной сети / В. В. Волкова, Б. В. Колчигин // Факультетская научно-практическая молодежная школа-семинар студентов, аспирантов и молодых ученых «Информационные интеллектуальные системы» : тезисы докл. – Харьков : ХНУРЭ, 2008. – С. 30–33.
9. *Волкова, В. В.* Комбинированное обучение самоорганизующихся карт с нечетким выводом / В. В. Волкова, Е. В. Махиборода // Факультетская научно-практическая молодежная школа-семинар студентов, аспирантов и молодых ученых «Информационные интеллектуальные системы» : тезисы докл. – Харьков : ХНУРЭ, 2008. – С. 30–33.
10. *Рассел, С.* Искусственный интеллект. Современный подход / С. Рассел, П. Норвиг. – М. : Вильямс, 2006. – 1408 с.

Стаття надійшла до редакції 21.01.2011.

Асеев Г. Г.

#### МЕТОДИ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ В ЕЛЕКТРОННИХ СХОВИЩАХ: ГЕНЕТИЧНІ АЛГОРИТМИ

Представлено один з можливих методів інтелектуального аналізу даних в електронних сховищах великого об'єму – генетичні алгоритми і їх модифікація.

**Ключові слова:** інтелектуальний аналіз, електронне сховище, нейронна мережа, генетичні алгоритми.

Aseyev G. G.

#### METHODS OF INTELLECTUAL ANALYSIS OF DATA IN ELECTRONIC DEPOSITORIES: GENETIC ALGORITHMS

One of the possible methods of data intellectual analysis in high-volume electronic depositories is presented – genetic algorithms and their modification.

**Key words:** intellectual analysis, electronic depository, neuron network, genetic algorithms.

УДК 004:519.2

Кротких С. С.<sup>1</sup>, Кириченко Л. О.<sup>2</sup>

<sup>1</sup>Аспирант Харьковського національного університету радіоелектроніки  
<sup>2</sup>Канд. техн. наук, доцент Харьковського національного університету радіоелектроніки

## ИССЛЕДОВАНИЕ ВЫЗВАННЫХ ПОТЕНЦИАЛОВ В ЭЭГ ЧЕЛОВЕКА С ПОМОЩЬЮ ДИСКРЕТНОГО ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ

В работе с помощью дискретного вейвлет-преобразования проведен анализ изменения частотной структуры электроэнцефалограммы с вызванными потенциалами после воздействия стимула. Реализован метод определения времени реакции на стимул, основанный на оценивании вейвлет-энтропии и относительной вейвлет-энтропии сегментов энцефалограммы.

**Ключевые слова:** электроэнцефалограмма, вызванные потенциалы, дискретное вейвлет-преобразование, спектр вейвлет-энергии, вейвлет-энтропия, относительная вейвлет-энтропия.

### ВВЕДЕНИЕ

Головной мозг человека обладает электрическим полем, которое характеризуется электрическими скачками от нескольких миллиампер до нескольких сотен миллиампер. Каждый нейрон мозга генерирует изменение электрического потенциала, этот процесс можно измерить. Для анализа функций головного мозга широко используется электрический энцефалограф, который поз-

воляет измерить изменение электрического потенциала в отдельных участках головного мозга человека. Электроэнцефалограмму (ЭЭГ) можно рассматривать как фиксацию последовательности умственных задач, выполняемых мозгом субъекта. Различные умственные задачи или условия, в которых находится человек, имеют свои уникальные отражения в результирующем ЭЭГ сигнале. Таким образом, наличие тех или иных изменений в