

Tkachenko Maxim Gennadievich

Scientific Research Institute of Multiprocessor Computer Systems Southern Federal University.

E-mail: TkachenkoMG@yandex.ru.

2, Chehov Street, Taganrog, 347928, Russia.

Phone: +78634315941.

The Programmer.

Korovin Yakov Sergeevich

E-mail: korovin_yakov@mail.ru.

Head the Laboratory.

УДК 519.7

С.Е. Бублей

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ОБУЧЕНИЯ НЕЙРОНЕЧЕТКИХ СИСТЕМ ДЛЯ ЗАДАЧ УПРАВЛЕНИЯ

Рассмотрены алгоритмы обучение нейронных систем в условиях отсутствия априорной информации. Приведены схемы алгоритмов. Определены условия их применения и существующие недостатки. Рассмотрены генетические алгоритмы. Приведены определения операторов генетических алгоритмов. Предложен генетический алгоритм для обучения адаптивной автоматной системы управления.

Обучение; генетические алгоритмы.

S.E. Bublely

RESEARCH OF ALGORITHMS OF TRAINING OF NEURAL INDISTINCT SYSTEMS FOR MANAGEMENT PROBLEMS

Algorithms training of neural systems in the conditions of absence of the aprioristic information are considered. Schemes of algorithms are resulted. Conditions of their application and existing lacks are defined. Genetic algorithms are considered. Definitions of operators of genetic algorithms are resulted. The genetic algorithm for training of an adaptive automatic control system is offered.

Training; genetic algorithms.

Известны алгоритмы обучение нейро-нечетких систем (ННС): алгоритм обратного распространения ошибки [1,2]; эволюционный гибридный алгоритм [3,2]; алгоритм обучения Хэбба [4]. Рассмотрим эти алгоритмы.

Рассмотрим алгоритм обратного распространения ошибки применяется для многослойных ННС. Имеется множество входов x_1, \dots, x_n , множество выходов и множество внутренних узлов. При описании ННС все узлы определены числами от 1 до L . Определим $w_{i,j}$ – вес линии, соединяющей i -й и j -й узлы, а через y_i – выход i -го узла. Если в системе известен результат обучения, то функция ошибки, полученная по методу наименьших квадратов, определится выражением:

$$w_i^m(t+1) = w_i^m(t) - \eta \frac{\partial e_r^{(m)}}{\partial y_k^{(m)}} \Big|_{(t)}, \quad (1)$$

где w_i^m – вес i -го нейрона m -го слоя; η – множитель, задающий скорость обучения, $0 < \eta < 1$; e – среднеквадратическая ошибка обучения r -го множества m -го слоя; y – выходное воздействие. Структура алгоритм обратного распространения ошибки при обучении ННС показана на рис. 1.

Алгоритм обратного распространения ошибки на практике дает хорошие результаты для систем с простыми структурами [5-6]. В процессе обучения трех ННС в отдельности из обучение занимает время от 8 до 168 с, а количество итерации от 83 до 1200, выдавая значения управляющих воздействий со среднеквадратической ошибкой от 0,001 до 0,93 с°. Полученные результаты получены при допустимой погрешности от желаемого значения на выходе ННС и НСФД, равной 0,5 %, имеют требуемые значения среднеквадратической ошибки.

Алгоритмы обучения, построенные по методу Хэбба, реализуют обучение исключительно, как локальное явление, охватывающее только два нейрона и соединяющий их синапс, т.е. не требуется глобальной системы обратной связи для развития нейронных образований.

По существу, алгоритм Хэбба предполагает, что синаптическое соединение двух нейронов усиливается, если оба эти нейрона возбуждены. Это можно представить как усиление синапса в соответствии с корреляцией уровней возбужденных нейронов, соединяемых данным синапсом, поэтому алгоритм обучения Хэбба называется корреляционным алгоритмом.

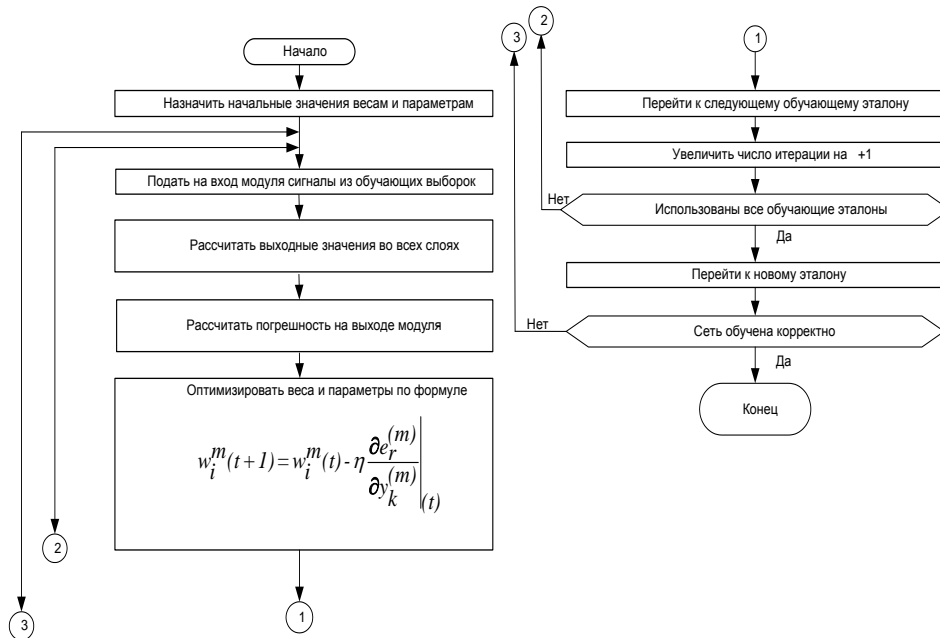


Рис. 1. Алгоритм обратного распространения ошибки обучения ННС

Алгоритм Хэбба выражается формулой:

$$w_{ij}(t+1) = w_{ij}(t) + y_i y_j, \tag{2}$$

где w_{ij} – вес или связь от i -го нейрона к j -му нейрону; y_{ij} – выходы i -го и j -го нейронов.

Структура модуля алгоритма обучения Хэбба показана на рис. 2.

Использование метода Хэбба для обучения нейро-нечетких систем дает хорошие результаты, но существует ограниченность метода, т.к. некоторые образы просто не могут использоваться для обучения этим методом.

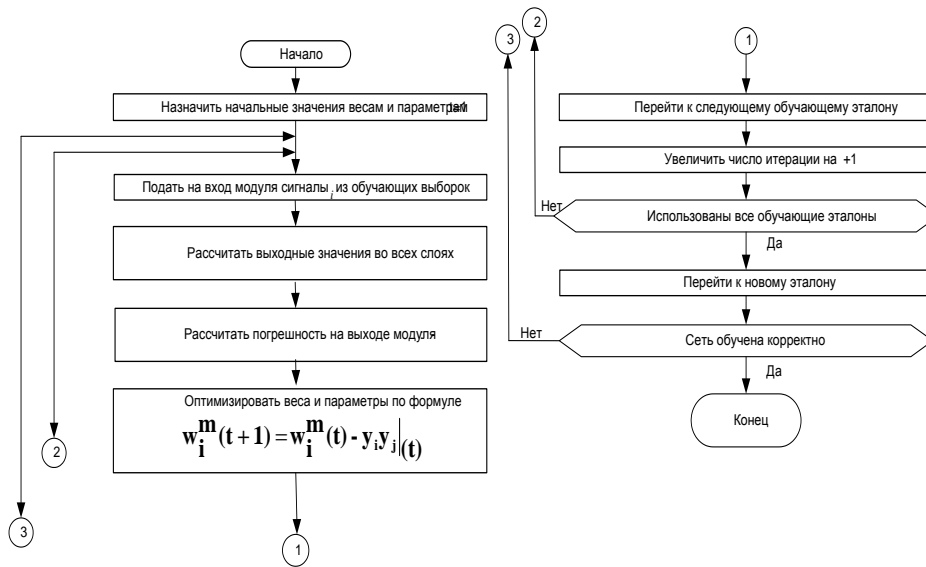


Рис. 2. Алгоритм обучения Хэбба ННС

Алгоритм обучения Хэбба, как алгоритм обратного распространения ошибки следует применять для систем с простыми структурами, где время обучения является незначительным.

Перечисленные алгоритмы обучения ННС применяются для настройки таких параметров известных ННС, как ANFIS, ТСК, НС. Алгоритм Хэбба дает хорошие решения, но время обучения сложных по структуре систем достаточно большое [8].

Недостатком алгоритма обратного распространения ошибки является введение еще одного глобального настроечного параметра, в то время как необходимо стремиться к отсутствию таких навязываемых алгоритму извне параметров. Идеальной является ситуация, когда все параметры обучения сами настраиваются в процессе обучения, извлекая информацию о характере рельефа функции ошибки из самого хода обучения. Еще один недостаток – это большое время обучения за счет пересчета параметров обучения при каждом неправильном решении.

Однако проблему времени обучения и числа итерации в процессе настройки параметров ННС можно решать с применением генетических вычислений [9,10] – синтез, настройка и оптимизация с помощью систематизированного случайного поиска и эволюции.

Рассмотрим возможности применение генетических алгоритмов для обучения нейронечетких систем управления. Генетические алгоритмы [13,14] представляют альтернативу традиционным методам оптимизации, с применением случайного поиска, позволяющим получить набор оптимальных решений. Популяции всех решений оцениваются для определения наилучшего решения. Гибридная система комбинирует систему нейронного нечеткого интерфейса и генетические алгоритмы. Генетические алгоритмы на основе ННС позволяют настраивать параметры гибридной сети и в состоянии достичь высоких классификационных показателей по сравнению с ННС с классическим методом обучения.

Генетические алгоритмы – это стохастические, эвристические оптимизационные методы, которые основываются на идее эволюции с помощью естественного отбора.

Генетические алгоритмы работают с совокупностью «особей» – популяцией, каждая из которых представляет возможное решение данной задачи. Каждая особь оценивается мерой ее «приспособленности» согласно тому, насколько «хорошо» соответствующее ей решение задачи.

Наиболее приспособленные особи получают возможность «воспроизводить» потомство с помощью «перекрестного скрещивания» с другими особями популяции. Это приводит к появлению новых особей, которые сочетают в себе некоторые характеристики, наследуемые ими от родителей. Наименее приспособленные особи с меньшей вероятностью смогут воспроизвести потомков, так что те свойства, которыми они обладали, будут постепенно исчезать из популяции в процессе эволюции. Иногда происходят мутации, или спонтанные изменения в генах. Таким образом, из поколения в поколение хорошие характеристики распространяются по всей популяции. Скрещивание наиболее приспособленных особей приводит к тому, что исследуются наиболее перспективные участки пространства поиска. В конечном итоге популяция будет сходиться к оптимальному решению задачи. Преимущество генетических алгоритмов состоит в том, что они находят приближенные оптимальные решения за относительно короткое время.

Генетический алгоритм состоит из следующих компонент:

- ◆ хромосома – решение рассматриваемой проблемы, состоит из генов;
- ◆ начальная популяция хромосом;
- ◆ набор операторов для генерации новых решений из предыдущей популяции;
- ◆ целевая функция для оценки приспособленности (fitness) решений.

В начале работы генетического алгоритма популяция формируется, как правило, случайным образом. Чтобы оценить качество закодированных решений, используют функцию приспособленности. По результатам оценивания особей наиболее приспособленные из них выбираются для скрещивания. В результате скрещивания выбранных особей посредством применения генетического оператора кроссовера создаются потомки, генетическая информация которых формируется в результате обмена хромосомной информацией между родительскими особями. Созданные потомки формируют новую популяцию, причем часть потомков мутирует (с использованием генетического оператора мутации), что выражается в случайном изменении их генотипов. Этап, включающий последовательность операций, обозначенных блоками «Оценивание популяции» – «Селекция» – «Скрещивание» – «Мутация», называется поколением [11].

Окончание эволюции может определяться следующими факторами:

- ◆ нахождение решения в результате эволюционного поиска;
- ◆ ограничение на число поколений, количество вычислений целевой функции, общее время работы генетического алгоритма;
- ◆ вырождение популяции, когда степень разнородности хромосом в популяции становится меньше установленного значения.

Как правило, в хромосоме кодируются параметры решения. Для этого используют целочисленное либо вещественное кодирование. Определение приспособленности i -й особи в популяции производится в соответствии с оценкой решения, которое, в свою очередь, определяется набором параметров, закодированных в хромосоме i -й особи, и зависит от рассматриваемой задачи.

Важным фактором при использовании генетического алгоритма для решения любых прикладных оптимизационных задач является предположение, что в ре-

зультате неоднократно повторяющегося отбора наиболее приспособленных особей, их последующего скрещивания и мутации будет производиться отсев неудовлетворительных решений (по некоторому критерию) и постепенное повышение качества (приспособленности) существующих «хороших» решений.

Упрощенная структура системы управления с применением эволюционного генетического алгоритма, обучающего систему управления ННС, представлена на рис. 3.

В задачах обучения нейронных и нейронечетких систем управления начальная популяция выбирается случайно, значения весов лежат в промежутке $[-1,0, 1,0]$. Для обучения ННС к начальной популяции применяются простые операции: селекция, скрещивание, мутация, в результате чего генерируются новые популяции.

Рассмотрим операторы генетического алгоритма.

Оператор селекции осуществляет отбор хромосом в соответствии со значениями их функции приспособленности. Селекцию можно выполнять разными способами. Созданы и используются альтернативные алгоритмы селекции, например турнирная и ранговая селекция. При турнирной селекции все особи популяции разбиваются на подгруппы с последующим выбором в каждой из них особи с наилучшей приспособленностью. Подгруппы могут иметь произвольный размер, но чаще всего популяция разделяется на подгруппы по 2-3 особи в каждой.

При ранговой селекции особи популяции ранжируются по значениям их функции приспособленности. Это можно представить себе как отсортированный список особей, упорядоченных по направлению от наиболее приспособленных к наименее приспособленным (или наоборот), в котором каждой особи приписывается число, определяющее ее место в списке и называемое рангом. Количество копий каждой особи, введенных в родительскую популяцию, рассчитывается по априорно заданной функции в зависимости от ранга особи.

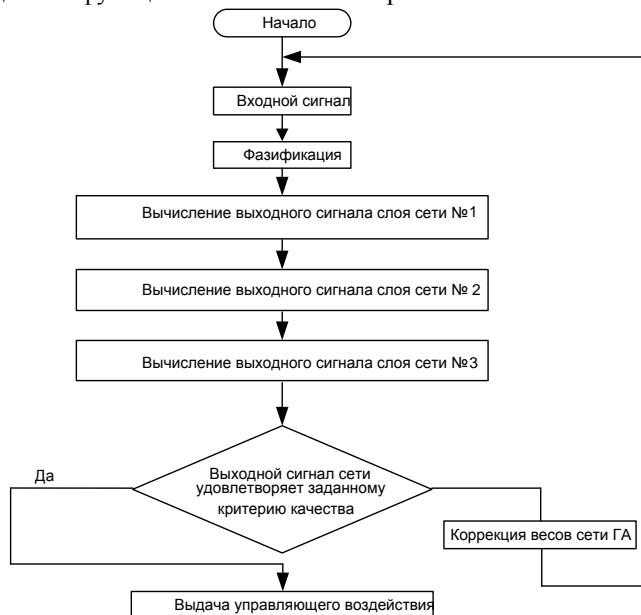


Рис. 3. Упрощенная структура системы управления с применением эволюционного генетического алгоритма

Элитные методы отбора гарантируют, что при отборе обязательно будут выживать лучший или лучшие члены популяции совокупности. При этом часть самых лучших особей без каких-либо изменений переходит в следующие поколения. Размер i -й селекции пропорционален соответствующей величине $P_i(x)$, вычисляемой по формуле

$$P_i = \frac{f_i(x)}{\sum f_i(x)}, \quad (3)$$

где $f_i(x)$ – значение целевой функции или функции приспособленности i -й хромосомы в популяции.

Оператор кроссинговера, иногда называемый оператором скрещивания, является основным генетическим оператором, за счет которого производится обмен частями хромосом между двумя (может быть и больше) хромосомами в популяции. Оператор кроссинговера может быть одноточечным или многоточечным. Одноточечным называется кроссинговер, если при нем родительские хромосомы разрезаются только в одной случайной точке. Для реализации N -точечного кроссинговера можно использовать два подхода:

- ◆ точек разрыва меньше, чем генов в хромосоме;
- ◆ если длина хромосомы L битов, то число точек разрыва равно $(L-1)$, при этом потомки наследуют биты следующим образом: первому потомку достаются нечетные биты первого родителя и четные биты второго; у второго же потомка все наоборот.

Кроме описанных типов кроссинговера есть ещё однородный кроссинговер. Его особенность заключается в том, что значение каждого бита в хромосоме потомка определяется случайным образом из соответствующих битов родителей. Для этого вводится некоторая величина $0 < P_0 < 1$, и если случайное число больше p_0 , то на n -ю позицию первого потомка попадает n -й бит первого родителя, а на n -ю позицию второго – n -й бит второго родителя. В противном случае, к первому потомку попадает бит второго родителя, а ко второму потомку – бит первого. Такая операция проводится для всех битов хромосомы. Вероятность кроссинговера самая высокая среди генетических операторов и равна обычно 60 % и больше.

Оператор мутации – это стохастическое изменение части хромосом. Оператор необходим для «выбивания» популяции из локального экстремума и способствует защите от преждевременной сходимости. Достигается это за счет того, что каждый ген строки, которая подвергается мутации, с малой вероятностью P_{mut} заменяется другим геном (добавляется случайная величина между $-1,0$ и $1,0$ к весу). Мутация проводится не только по одной случайной точке. Можно выбирать некоторое количество точек в хромосоме для изменения, причем их число также может быть случайным. Также можно изменять сразу некоторую группу подряд идущих точек. Вероятность мутации значительно меньше вероятности кроссинговера и редко превышает 1 %.

Условие останова – процесс эволюции, может продолжаться до бесконечности. Условием останова может служить заданное количество поколений или схождение (convergence) популяции.

Схождением называется состояние популяции, когда все строки популяции находятся в области некоторого экстремума и почти одинаковы. То есть кроссинговер практически никак не изменяет популяции, а мутирующие особи склонны вымирать, так как менее приспособлены. Таким образом, схождение популяции означает, что достигнуто решение, близкое к оптимальному. Итоговым решением задачи может служить наиболее приспособленная особь последнего поколения.

Как известно из теории обучаемых нейро-нечетких систем, если функции принадлежности не подвергаются обучению (настройке) и их связи между отдельными нечеткими нейронами являются настраиваемыми, то ННС обучается как простая нейронная сеть с прямым распространением сигнала. Настройку параметров адаптивной обучаемой автоматной системы управления можно выполнять поэтапно для всех модулей, входящих в систему. Однако можно настраивать параметры модулей системы управления одновременно, что достигается уменьшением количества шагов настройки, которые необходимо выполнять, прежде чем параметры могут быть включены в систему. Схема предлагаемого генетического алгоритма приведена на рис. 4.

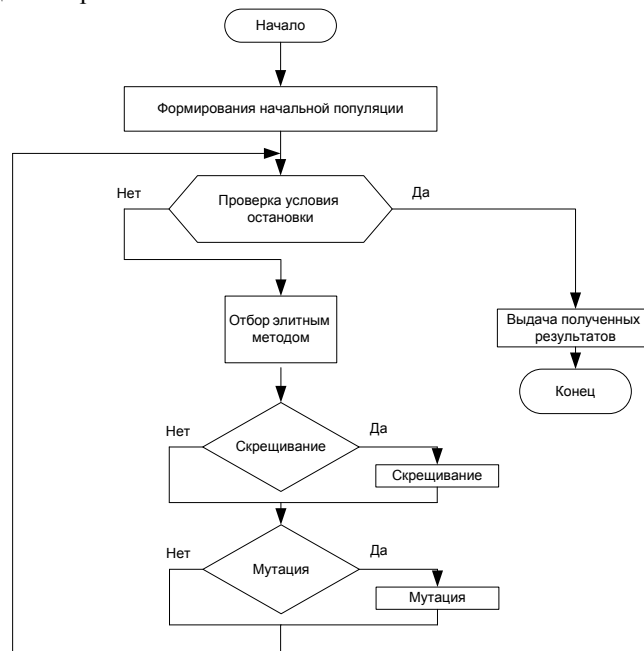


Рис. 4. Схема генетического алгоритма

Предлагаемый подход позволяет выполнять предварительную настройку адаптивных автоматных обучаемых систем управления в условиях отсутствия априорных сведений относительно модели объекта управления [12].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Gang L., Thomas M.* Design for self-organizing fuzzy neural networks based on genetic algorithms // IEEE Transactions on fuzzy systems. – 2006. – Vol. 14, №. 6. – P. 755-776.
2. *Рутковская Д., Пилиньский М., Рутковский Л.* Нейронные сети, генетические алгоритмы и нечеткие системы / Пер. с польск. И.Д. Рудинского. – М.: Горячая линия – Телеком, 2004. – С. 452.
3. *Ridwan M.* A parallel genetic algorithm-based tsk-fuzzy system for dynamic car-following modeling // European Journal of Scientific Research. – 2009. – № 4. – P. 627-641.
4. *Роббинс М.* A stochastic approximation Method // Annals of Math.Statistics, 1951. – С. 223.
5. *Gayadhar P., Sidhartha P., Ardil C.* Hybrid Neuro Fuzzy Approach for Automatic Generation Control of Two-Area Interconnected Power System. International Journal of Computational Intelligence 5:1. 2009.

6. *Gayadhar P., Sidhartha P., Ardil C.* Hybrid Neuro Fuzzy Approach for Automatic Generation Control of Two-Area Interconnected Power System // International Journal of Computational Intelligence 5:1. – 2009.
7. *Petru R., Emil M.* Behavior-Based Neuro-Fuzzy Controller for Mobile Robot Navigation // IEEE Transactions on instrumentation and measurement. – 2003. – Vol. 52, № 4.
8. *Ranadhir G.A* Novel Hybrid Learning Algorithm for Artificial Neural Networks // School of Information Technology. – 2002. – P. 214.
9. *Leu Y., Wei Y., Hsum L.* RGA-based on-line tuning of BMF fuzzy-neural networks for adaptive control of uncertain nonlinear systems // Neurocomputing. – 2009. – № 72. – P. 2636-2642.
10. *Stefan S.* Optimizing a production process by a neural network/genetic algorithm approach // Egnn Applic. Artif. Intell. – Vol. 9, № 6. – P. 681-689.
11. *Ярушкіна Н.Г.* Основы теории нечетких и гибридных систем: Учеб. пособие. – М.: Финансы и статистика, 2004. – С. 320.
12. *Финаев В.И., Молчанов А.Ю.* Метод моделирования самонастраивающихся систем управления // Известия ТРТУ. – 2004. – № 8 (43). – С. 45-49.
13. *Курейчик В.М.* Модифицированные генетические операторы // Известия ЮФУ. Технические науки. – 2009. – № 12 (101). – С. 7-15.
14. *Курейчик В.В., Курейчик В.М., Родзин С.И.* Концепция эволюционных вычислений, инспирированных природными системами // Известия ЮФУ. Технические науки. – 2009. – № 4 (93). – С. 16-24.

Бублей Сергей Евгеньевич

Технологический институт федерального государственного автономного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

E-mail: fin_val_iv@tsure.ru.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 88634371689.

Кафедра систем автоматического управления; аспирант

Bubley Sergey Evgehiievich

Taganrog Institute of Technology – Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education “Southern Federal University”.

E-mail: fin_val_iv@tsure.ru.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Тел.: +78634371689.

The Department of Automatic Control Systems; Postgraduate Student.

УДК 004.89

А.В. Козоброд, В.Е. Мешков, Е.В. Мешкова

**АНАЛИЗ АРХИТЕКТУР ГИБРИДНЫХ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ
В ЗАДАЧАХ АВТОМАТИЧЕСКОЙ КЛАССИФИКАЦИИ ТЕКСТОВОЙ
ИНФОРМАЦИИ**

Проанализированы ключевые моменты разработки архитектуры нейронной сети для гибридной модели с использованием семантических отношений в зависимости от способа их отражения на ассоциативной нейронной сети. Приводится анализ и выбор сетевой архитектуры, исходя из поставленных целей разработки системы классификации текстовых документов.

Гибридная нейросетевая архитектура; автоклассификация; семантические отношения.