

УДК 004.891

Подбор маршрутов полетов на основе нейро-нечетких моделей

Андросова Е.Е., студент

*Россия, 105005, г. Москва, МГТУ им. Н.Э.Баумана,
кафедра «Программное обеспечение ЭВМ и Информационные технологии»*

Научный руководитель: Rogozin O.B., к.т.н., доцент

*Россия, 105005, г. Москва, МГТУ им. Н.Э.Баумана
кафедра «Программное обеспечение ЭВМ и Информационные технологии»*

bauman@bmstu.ru

Введение. Поиск оптимальных предложений для перелетов между двумя пунктами и выбор пунктов отдыха (как конечных точек перелетов) – немаловажная проблема для современного человека. Подбор маршрутов полетов и пунктов отдыха состоит в анализе существующих в базе данных вариантов полетов и характеристик пунктов прилета, и выборе наиболее подходящего маршрута на основе критериев, заданных пользователем. В данной проблеме можно выделить две задачи.

Первую задачу с формальной точки зрения можно представить, как задачу поиска оптимального (имеется в виду некий критерий оптимальности) или кратчайшего пути между двумя вершинами графа. Алгоритм решения этой задачи хорошо известен. Многие сайты-агрегаторы, сотрудничающие с системами бронирования и продаж билетов для бюджетных авиакомпаний (low-cost airlines) имеют опцию Multiple destinations, с помощью которой можно получить десятки вариантов полетов (пусть с пересадками), что дает возможность пользователям самим выбрать оптимальный вариант для перелета по критерию общего времени в пути или по критерию стоимости перелета.

Вторая задача менее формализована. В качестве критериев подбора полетов могут быть: даты путешествия и характер отдыха или путешествия. Даты путешествия могут быть заданы в числовом виде, что упрощает поиск маршрутов до простого поиска в таблице. Но критерий «характер путешествия» может иметь значения «пляжный», «походный», «экскурсионный», что затрудняет поиск наилучших рейсов, поскольку эти понятия зависят от целого ряда факторов и не могут быть всегда однозначно закреплены за определенным пунктом отдыха. Если каждый пункт отдыха охарактеризовать рядом параметров, то задача состоит в том, чтобы определить, подходит ли этот пункт под каждый из предложенных видов отдыха в зависимости от этих параметров. Наиболее

близким к задаче подбора пункта отдыха на основе предпочтений пользователей является веб-сервис сайта booking.com, позволяющий подобрать пункт отдыха и отель в нем, исходя из рекомендаций предыдущих пользователей.

Для решения второй задачи можно привлечь аппарат нейронных и нейро-нечетких сетей. В настоящее время на рынке программных средств имеется большое количество разнообразных пакетов для конструирования как нейронных, так и нейро-нечетких сетей [1]. Эти пакеты хорошо приспособлены для автономной работы. Однако, при построении заказных систем возникают трудности при попытке встроить в приложение существующую универсальную систему. Такая проблема возникла при разработке базы данных и веб-приложения бронирования авиабилетов и подбора пунктов отдыха на основе анализа предпочтений пользователей.

Постановка задачи. Пусть в базе данных хранится информация, необходимая для бронирования авиабилетов и пусть для каждого пункта отдыха, совпадающего с пунктом прибытия самолета, определен набор параметров, характеризующий данный пункт. В качестве примера ограничимся рассмотрением трех параметров: двух климатических (средняя месячная температура воздуха и средняя месячная температура воды, статистически полученные из многолетнего ряда наблюдений) и параметром «количество достопримечательностей», значение которого можно извлечь из справочников. Эти параметры также должны храниться в базе данных для каждого пункта назначения, т.е. пункта отдыха. Требуется определить, какой тип отдыха будет наиболее вероятным на указанную дату для любого пункта, чтобы согласовать этот тип с предпочтениями пользователя. Решение этой задачи будет последовательно рассматриваться для трех моделей: модели нечеткого вывода, нейронной сети и нейро-нечеткой модели.

Решение задачи с помощью системы нечеткого вывода. Идея нечеткой логики связана с введением, так называемых лингвистических переменных, описывающих неточное (нечеткое) отражение человеком окружающего мира. Для того чтобы лингвистические переменные стали полноправными математическими объектами, разработана теория нечетких множеств, включившая в себя обычные множества как частный случай [2]. Отображение между реальными входными данными и выходными параметрами происходит с помощью нечетких правил (или нечеткой логики).

Такой подход дает возможность задать в виде лингвистических переменных понятия «жарко», «умеренно», «холодно», «пляжный отдых», «поход» и др. и оперировать ими в системе. Принятие решения о принадлежности того или иного пункта к определенному типу отдыха происходит благодаря нечеткому выводу. В основе нечеткого

вывода лежит база знаний, представляющая собой совокупность нечетких продукционных правил вида: «если x есть A , то y есть B », где x — входная нечеткая переменная, y — переменная вывода, A и B — значения функций принадлежности, определенные для x и y соответственно. В общем случае логический вывод принято делить на 4 этапа: фазификацию, логический вывод, композицию и дефазификацию. Последний этап является дополнительным и используется, когда полезно преобразовать набор нечетких выводов к четкому числу. В настоящее время в нечетком выводе наиболее часто применяются алгоритм Мамдани и алгоритм Тагаки-Сугено, подробно описанные в работе [2]. На рисунке 1 представлена модель системы нечеткого вывода.



Рис. 1. Модель системы нечеткого вывода

Функционирование модели осуществляется в два этапа. На первом этапе происходит: а) формирование базы знаний, б) задание нечетких множеств путем описания термов и соответствующих им нечетких множеств для каждого входного параметра, в) задание правил нечеткого вывода (условий и следствий). На втором этапе полученная база знаний используется для выполнения нечеткого вывода и получения результатов. Алгоритм логического вывода, представляющий собой модифицированный алгоритм Мамдани, имеет вид:

1. **for** (для) каждой входной переменной x
2. **do for** (для) каждого нечеткого множества этой переменной
3. **do** определить $\mu(x)$;
4. **if** $\mu(x) \neq 0$
5. **then** добавить к списку фактов соответствующий терм и значение $\mu(x)$;
6. **for** (для) каждого правила
7. **do** $\min$ = степень истинности правила;

8. добавить *min* в ячейку *koefs*, соответствующую следствию правила;
9. **return** *maxIndex(koefs)*.

В алгоритме *tu(x)* соответствует значению функции принадлежности для четкой переменной *x*, степень истинности условия равна минимуму от значений функций принадлежности его составляющих, а степень истинности правила равна степени истинности правила. Результат (тип отдыха) кодируется тремя значениями в массиве *koefs*. Каждый элемент массива соответствует одному из типов отдыха. В нем накапливаются значения истинности правил (или их следствий). Поэтому для того, чтобы получить конечный результат, берется тип отдыха, соответствующих ячейке с максимальным значением, что обеспечивается функцией *maxIndex*.

Решение задачи с помощью нейронной сети. Искусственная нейронная сеть — это совокупность искусственных нейронов, составляющих вычислительную систему, которая преобразует информацию по образу процессов, происходящих в человеческом мозге. На рисунке 2 показано, как искусственный нейрон преобразует векторный вход *x* в скалярный выход *y*.

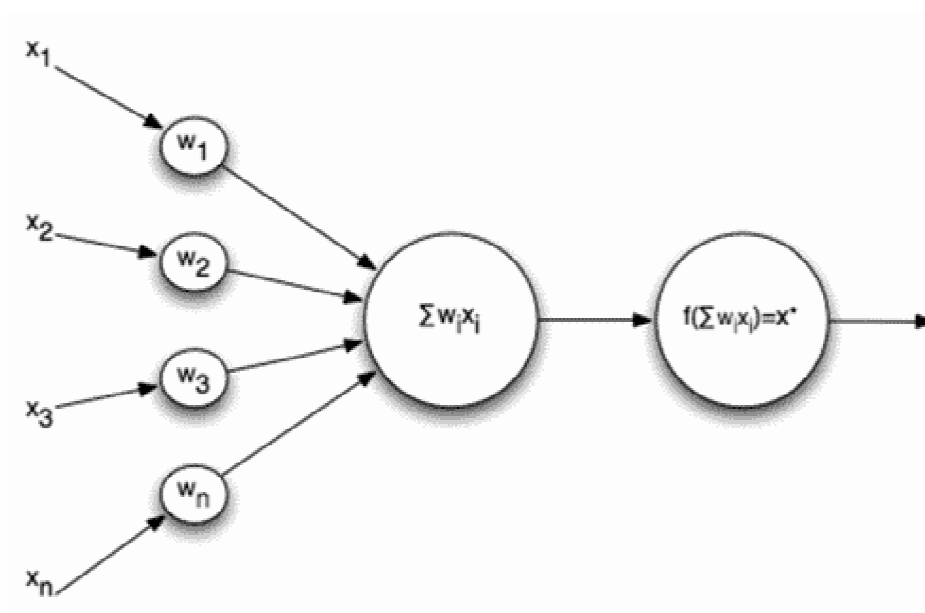


Рис. 2. Искусственный нейрон

Преобразование осуществляется в два этапа. На первом этапе вычисляется уровень активности нейрона — скалярное произведение входного вектора $x = [x_1, x_2, \dots, x_n]^T$ и вектора весов нейрона $w = [w_1, w_2, \dots, w_n]^T$. На втором этапе к рассчитанному значению *A* применяется функция активации.

В многослойных нейронных сетях нейроны объединяются в слои. Каждый слой содержит совокупность нейронов с едиными входными сигналами. Число нейронов в слое

может быть любым. В общем случае сеть состоит из N слоёв, пронумерованных слева направо. Внешние входные сигналы подаются на входы нейронов первого слоя (входной сигнал часто нумеруют, как нулевой), а выходами сети являются выходные сигналы последнего слоя. Кроме входного и выходного слоёв в многослойной нейронной сети есть один или несколько промежуточных (скрытых) слоёв. Схема многослойной нейронной сети, используемой в работе, представлена на рисунке 3.

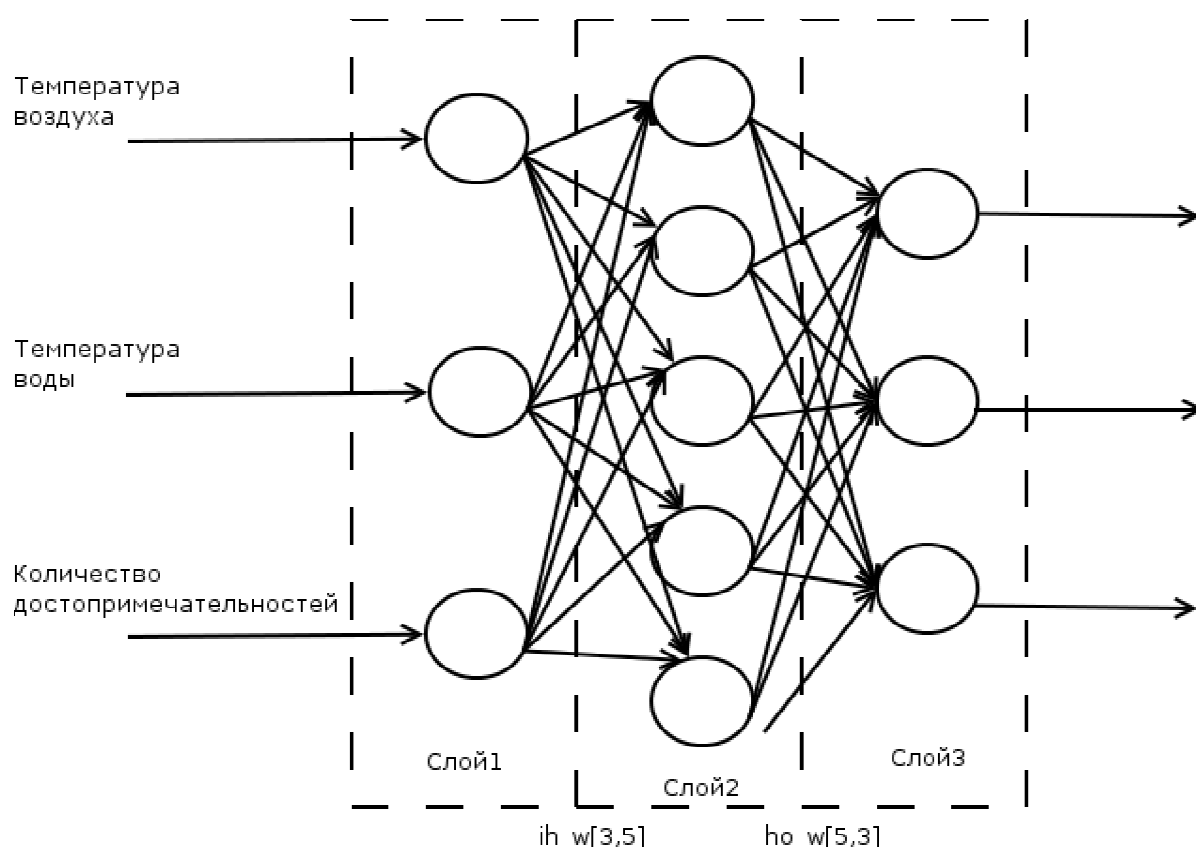


Рис. 3. Схема нейронной сети

В сети три слоя — входной, выходной и один скрытый. На входной слой подаются входные параметры (температуры воздуха, воды и количество достопримечательностей). Выходные данные кодируются значениями трех выходных нейронов способом, аналогичным способу кодирования выходных данных в системе нечеткого вывода. Веса хранятся в двух массивах: ih_w и ho_w . Диапазон значений выходных нейронов (от 0 до 1) определяет функции активации во втором и третьем слоях. Во втором слое используется Сигмоидальная функция, а в третьем слое — функция *Softmax*, которая масштабирует входные данные, используя экспоненту.

Для использования нейронной сети в качестве механизма подбора рейсов в первую очередь следует ее обучить. Для обучения нейронных сетей использовались наиболее

значимые с практической точки зрения методы обучения с учителем: метод обратного распространения ошибки (Back Propagation, далее BP) [3] и метод роя частиц (Particle Swarm Optimization, далее PSO) [4].

Алгоритм вычисления типа отдыха по входным параметрам имеет вид:

1. обучить нейронную сеть;
2. загрузить входные данные на входной слой нейронов;
3. вычислить значения нейронов выходного слоя *koefs*;
4. **return** *maxIndex(koefs)*.

Решение задачи с помощью нейро-нечеткой модели. Нейро-нечеткая модель — это гибрид нейронной сети и нечеткого вывода. Модель такого подхода изображена на рисунке 4.

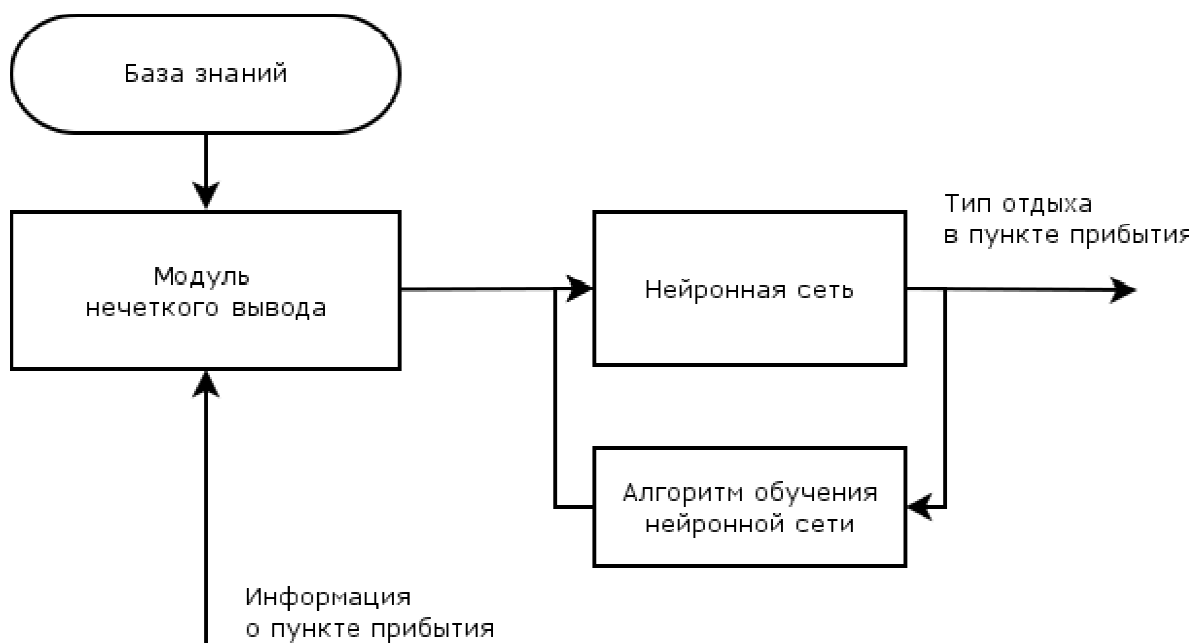


Рис. 4. Нейро-нечеткая модель

На входы модуля нечеткого вывода подается информация о пункте прибытия (температура воздуха, температура воды и количество достопримечательностей). Алгоритм работы модуля нечеткого вывода аналогичен алгоритму, описанному ранее. Используя базу знаний с заданными в ней нечеткими множествами и правилами вывода, модуль нечеткого вывода накапливает значения вероятностей каждого типа отдыха в массиве *koefs*. Этот массив подается на входной слой нейронов нейронной сети. Алгоритм работы нейронной сети в данной модели аналогичен алгоритму работы нейронной сети, описанному ранее. В качестве результата модель представляет один из типов отдыха,

который является наиболее подходящим при заданных значениях входных параметров.

Алгоритм определения типа отдыха с помощью нейро-нечеткой модели имеет вид:

1. сформировать базу знаний;
2. обучить нейронную сеть;
3. подать на вход модуля нечеткого вывода параметры пункта прибытия;
4. получить на выходе модуля нечеткого вывода массив *koefs*;
5. подать на входной слой нейронной сети массив *koefs*;
6. вычислить значения нейронов выходного слоя *leisureTypes*;
7. **return** *maxIndex(leisureTypes)*.

Вычислительные эксперименты. Для того чтобы оценить, какие преимущества дает нейро-нечеткая модель по сравнению с моделью нечеткого вывода и моделью нейронной сети, были проведены два эксперимента. Первый эксперимент состоял в оценке времени, необходимого для подсчета результата. Рассматривалась задача подбора рейсов для метода нечеткого вывода, нейронной сети и нейро-нечеткого метода без учета алгоритмов обучения. Время, необходимое для подсчета результатов по каждому из исследованных методов, показано на рисунке 7.

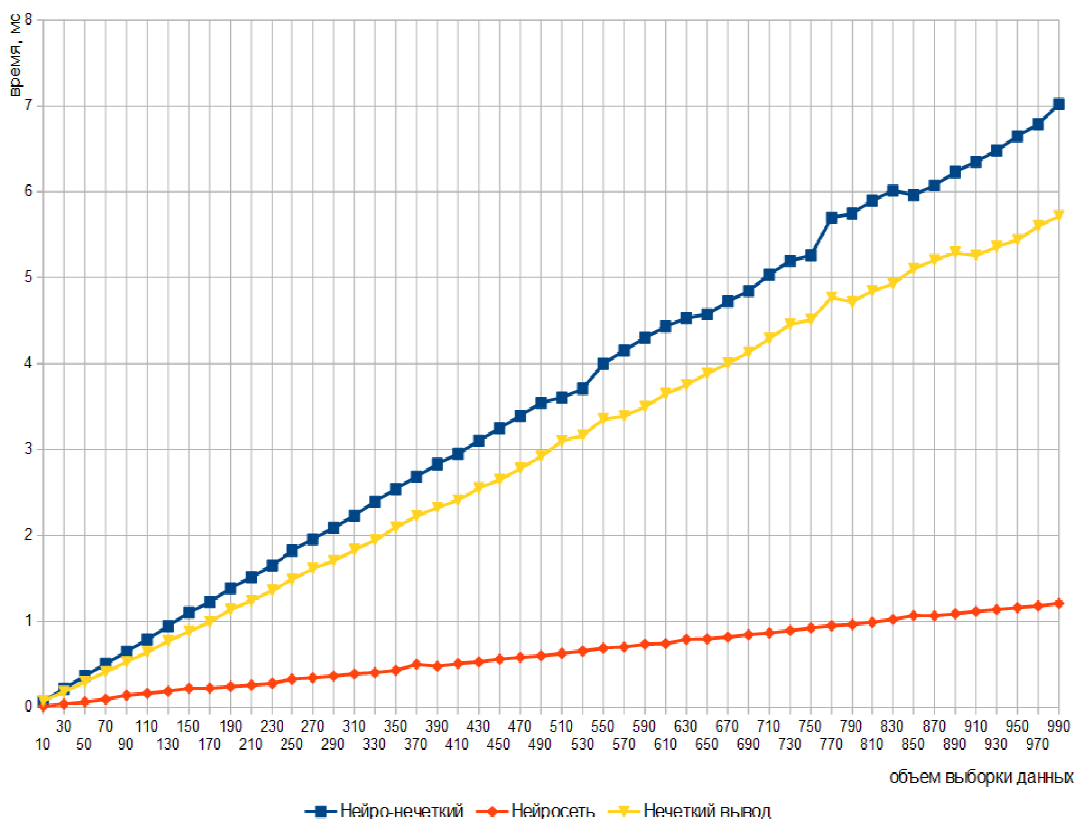


Рис. 5. Время, необходимое для подсчета результатов

Из рисунка видно, что по быстродействию нейро-нечеткая модель уступает другим

анализируемым моделям, но время растет линейно.

Второй эксперимент состоял в проверке адекватности исследуемых моделей. Рассматривалась задача подбора рейсов для метода нечеткого вывода, нейронной сети, обученной по алгоритму PSO, нейронной сети, обученной по алгоритму BP, нейро-нечеткого метода, где нейронная сеть обучена по методу PSO, нейро-нечеткого метода, где нейронная сеть обучена по методу BP. Точности результатов исследованных методов представлены на рисунке 8. На графике по оси x показан объем выборки в виде количества пунктов отдыха, а по оси y – точность метода, исчисляемая в процентах.

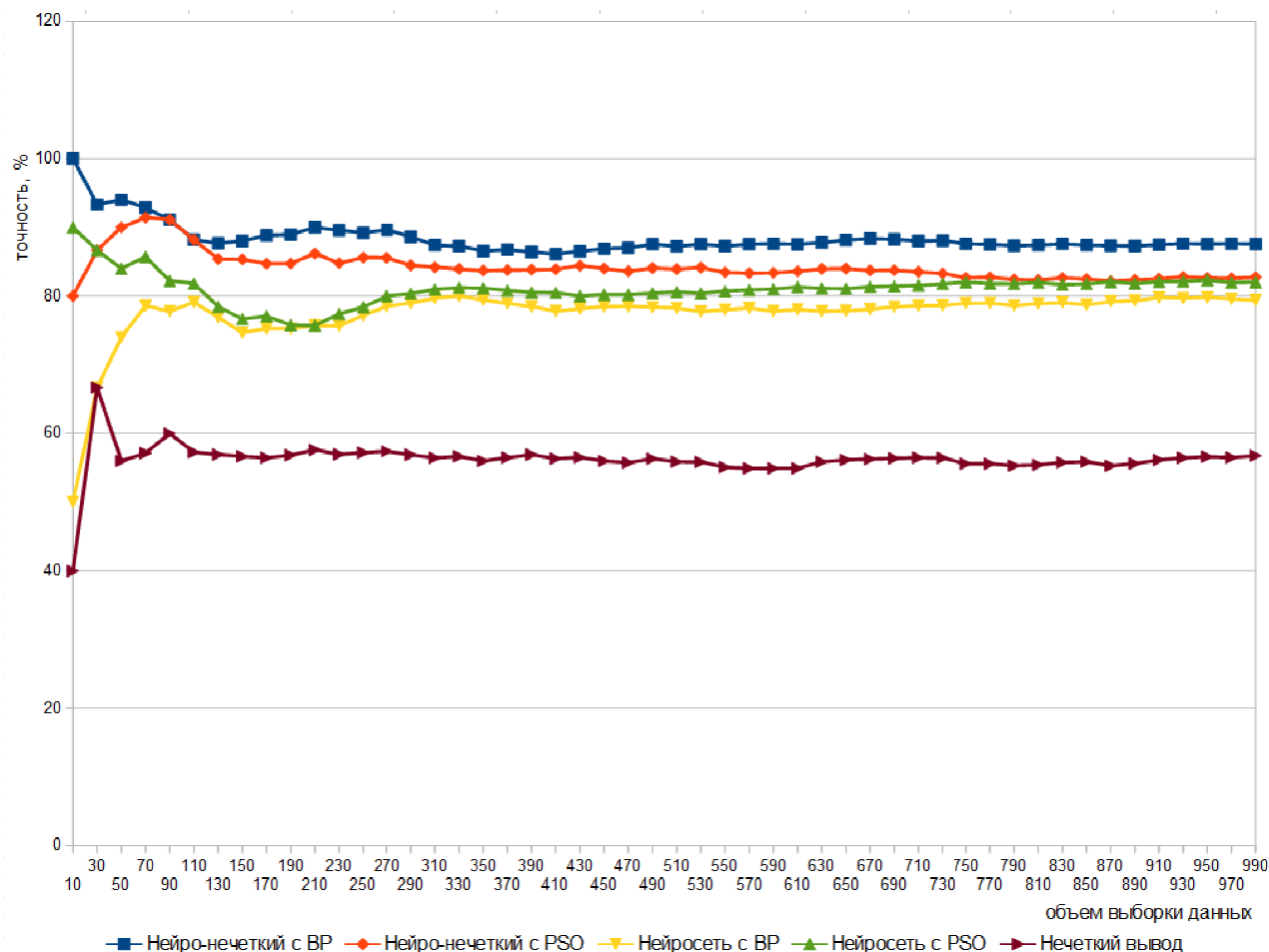


Рис. 6. Точность результатов

Из рисунка видно, что нейро-нечеткая модель с обучением по методу BP обладает наибольшей точностью.

Заключение. В данной работе были реализованы три модели для подбора авиарейсов на основе предпочтений пользователей относительно типа отдыха в пункте прилета. Все три модели были интегрированы в веб-приложение бронирования и подбора авиарейсов для бюджетных авиакомпаний. Проведенные вычислительные эксперименты

позволили уточнить теоретические оценки временной сложности разработанных алгоритмов для задач различной размерности. Во всех случаях временная сложность алгоритмов была линейной. Кроме того преимущество в точности полученных решений для нейро-нечеткой модели с обучением по методу ВР в среднем составило 10% для больших выборок данных. Нейро-нечеткие системы могут являться эффективным средством для построения блоков принятия решений.

Список литературы

1. Neural network software. Available at: http://en.wikipedia.org/wiki/Neural_network_software, accessed:25.11.2014.
2. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. 2-е изд., стереотип. М.: Горячая линия – Телеком, 2013. 384 с.
3. McCaffrey J. Neural Network Training Using Back-Propagation. Available at: <http://visualstudiomagazine.com/articles/2013/09/01/neural-network-training-using-back-propagation.aspx>, accessed: 25.11.2014.
4. McCaffrey J. Neural Network Training Using Particle Swarm Optimization. Available at: <http://visualstudiomagazine.com/articles/2013/12/01/neural-network-training-using-particle-swarm-optimization.asp>, accessed: 25.11.2014.