**Содержание**

# 1. Анализ исходных данных и разработка ТЗ

## 1.1 Основание и назначение разработки

## 1.2 Классификация решаемой задачи

## 1.3 Предварительный выбор класса НС

## 1.4 Предварительный выбор структуры НС

## 1.5 Выбор пакета НС

## 1.6 Минимальные требования к информационной и программной совместимости

## 1.7 Минимальные требования к составу и параметрам технических средств

# 2. Обучение НС

## 2.1 Формирование исходных данных

## 2.2. Окончательный выбор модели, структуры и параметров обучения НС

## 2.3 Блок-схема алгоритма обучения

## 3. Анализ качества обучения

4. Тестовый пример

5. Выводы

**1. Анализ исходных данных и разработка технического задания**

* 1. **Основание и назначение разработки**

Основанием данной работы является необходимость выполнения курсовой работы по дисциплине "Представление знаний в информационных системах".

Назначением работы является освоение нейронной сети.

**1.2 Классификация решаемой задачи**

В таблице 1 представлены некоторые проблемы, решаемые нейронными сетями [5].

**Таблица 1.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Проблема | Исходные данные | Выходные данные | Задача | Применение |
| Классификация | вектор признаков | вектор, значения всех координат которого должны быть равными 0, за исключением координаты, соответствующей выходному элементу, представляющему искомый класс(значение этой координаты должно быть равным 1) | указание принадлежности входного образа одному или нескольким предварительно определенным классам | распознавание букв, распознавание речи, классификация сигнала электрокардиограммы, классификация клеток крови |
| Кластеризация(классификация образа без учителя) | вектор признаков, в котором отсутствуют метки классов | - | Алгоритм основан на подобии образов и размещает близкие образы в один кластер | извлечение знаний, сжатие данных и исследование свойств данных |
| Апроксимация функций(имеется обучающая выборка ((xl yi), (х2, у2), ..., (xn, yw)), которая генерируется неизвестной функцией, искаженной шумом) | набор обучающих векторов | рассчитанное сетью значение функции | нахождение оценки функции |  |
| Прогнозирование ( заданы N дискретных отсчетов {(y(f1), y(f2), …, y(fN)} в последовательные моменты времени t1, t2, …, tN ) | вектора данных по M признакам за T периодов времени | вектора данных по M признакам за периоды времени T+L | предсказание значения y(tN+1) в момент tN+1 | принятие решений в бизнесе, науке и технике |
| Управление(рассматривается динамическая система, заданная совокупностью {u(t), y(t)}) | u(t) - входное управляющее воздействие | (t) - выход системы в момент времени t | расчет такого входного воздействия u(t), при котором система следует по желаемой траектории, диктуемой эталонной моделью |  |

Исходными данными для данной задачи является вектора(из нулей и единиц) размерности 60, которые описывают каждую из 10 арабских цифр, выбранных для обучения. Исходные данные предлагаются в прилагаемом файле "cifri.xls"

В этом же файле содержатся и выходные вектора размерности 10, где единица в одном из 10 положений означает принадлежность классифицируемого образца к той или иной цифре.

Сеть, принимая входной вектор, должна в соответствии с ним выдать соответствующий данной последовательности выходной вектор.

Исходя из исходных, выходных данных, задачам, которые необходимо решить, данная задача является задачей классификации.

**1.3 Предварительный выбор класса нейронной сети**

Список классов НС, в которых существует возможность решения задачи, отражён в таблице 2 [5].

**Таблица 2.**

|  |
| --- |
| Класс НС |
| сети прямого распространения (персептрон, однослойный и многослойный) |
| реккурентные (однослойные и многослойные, в том числе сети Хопфилда и двунаправленная ассоциативная память) |

Из классов, которые не подходят для решения задач классификации можно указать многослойные динамические сети с возбуждением и торможением и динамикой второго порядка.

Строго и однозначно определить класс НС для этой задачи не представляется возможным, так как большинство из них в той или иной мере подходят для ее решения.

Исходя из выше сказанного остановим свой выбор на следующих классах: Персептрон, однослойный и многослойный; Сети Хопфилда; Сети Ворда; Сети Кохонена; Двунаправленная ассоциативная память(ДАП);

Проведем сравнительный анализ данных классов[5]:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Класс | Модель | Область применения | Достоинства | Недостатки |
| Однослойный персептрон | отдельный персептронный нейрон вычисляет взвешенную сумму элементов входного сигнала, вычитает значение смещения и пропускает результат через жесткую пороговую функцию, выход которой равен +1 или -1 в зависимости от принадлежности входного сигнала к одному из двух классов | распознавание образов, классификация | программные или аппаратные реализации модели очень просты. Простой и быстрый алгоритм обучения | простые разделяющие поверхности (гиперплоскости) дают возможность решать лишь несложные задачи распознавания |
| Многослойный персептрон с обучением по методу обратного распространения ошибки | используется алгоритм обратного распространения ошибки. Тип входных сигналов – целые и действительные, тип выходных сигналов – действительные из интервала, заданного передаточной функцией нейронов. Тип передаточной функции – сигмоидальная | распознавание образов, классификация, прогнозирование, распознавание речи. Контроль, адаптивное управление, построение экспертных систем | первый эффективный алгоритм обучения многослойных нейронных сетей | этот метод относится к алгоритмам с минимальной скоростью сходимости. Для увеличения скорости сходимости необходимо использовать матрицы вторых производных функции ошибки |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Сеть Хопфилда | используется как автоассоциативная память. Исходные данные – векторы-образцы классов. Выход каждого из нейронов подаётся на вход всех остальных нейронов.Тип входных и выходных сигналов – биполярные. Тип передаточной функции – жёсткая пороговая. | ассоциативная память, адресуемая по содержанию, распознавание образов, задачи оптимизации(в том числе, комбинаторной оптимизации). | позволяет восстановить искажённые сигналы | размерность и тип входных сигналов совпадают с размерностью и типом выходных сигналов. Это существенно ограничивает применение сети в задачах распознавания образов. При использовании сильно коррелированных векторов-образцов возможно зацикливание сети в процессе функционирования. Небольшая ёмкость, квадратичный рост числа синапсов при увеличении размерности входного сигнала |
| Сети Ворда | обычная трехслойная сеть с обратным распространением ошибки с разными передаточными функциями в блоках скрытого слоя | Классифика-ция | обучение, хорошее обобщение на зашумленных данных |  |
| Сети Кохонена | сеть состоит из М нейронов, образующих прямоугольную решетку на плоскости. Элементы входных сигналов подаются на входы всех нейронов сети. В процессе работы алгоритма настраиваются синаптические веса нейронов. Входные сигналы (вектора действительных чисел) последовательно предъявляются сети, при этом требуемые выходные сигналы не определяются. После предъявления достаточного числа входных векторов, синаптические веса сети определяют кластеры. Кроме того, веса организуются так, что топологически близкие нейроны чувствительны к похожим входным сигналам | кластерный анализ, распознавание образов, классификация | сеть может быть использована для кластерного анализа только в случае, если заранее известно число кластеров | способна функционировать в условиях помех, так как число классов фиксировано, веса модифицируются медленно, и настройка весов заканчивается после обучения |
| Двунаправленная ассоциативная память | является гетероассоциативной. Входной вектор поступает на один набор нейронов, а соответствующий выходной вектор вырабатывается на другом наборе нейронов. Входные образы ассоциируются с выходными | ассоциативная память, распознавание образов | сеть может строить ассоциации между входными и выходными векторами, имеющими разные размерности | Емкость жестко ограничена, возможны ложные ответы |

Исходя из выше сказанного сужаем выбор класса до многослойного персептрона и сети Ворда.

**1.4 Предварительный выбор структуры НС**

**Рекомендации по выбору структуры:**

На вход сети подается вектор размерности 60, так как в нашей математической модели растровое изображение делится сеткой 6х10 и представляется вектором.

На выходе сети вектор из нулей и единиц размерности 10, так как число классов равно 10.

1. **количество слоев: 1**, как правило, начинают с одного скрытого слоя, а затем экспериментально при необходимости увеличивают это число
2. **количество нейронов:**
3. **во входном слое: 60**, таккак размер изображения 6x10;
4. **в скрытом слое: 30-50% от 60**, также определяется экспериментально в процессе моделирования;
5. **в выходном слое: 10**;

3. **вид функций активации:** определим далее экспериментально

Скорость обучения выберем исходя из того, что при очень большой скорости сеть может обучиться неверно, а при очень маленькой процесс обучения может быть долгим.

**1.5 Выбор пакета НС**

Требования для выбора пакета НС:

1. наличие в пакете выбранной модели НС
2. возможность создания выбранной структуры
3. возможность задания исходных данных для обучения и расчета
4. доступность пакета
5. трудоемкость по его освоению
6. наличие русификации
7. интерфейс
8. другие достоинства
9. другие недостатки

В таблице 2 приведены характеристики некоторых пакетов [http://alife.narod.ru/].

**Таблица 2:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Название пакета | 1 | 2 | 3 | 4 | 5 | 7 | 6 |
| Neural10 | нет | нет | да | freeware | легко | неудобный | есть |
| NeuroPro | нет | нет | да | freeware | легко | неудобный | есть |
| Neural Planner | да | да | да | - | легко | слишком упрощённый | есть |
| BrainMaker 3.10 | да | да | да | - | легко | очень удобный | есть |
| NeuroShell 2 | да | да | да | freeware | очень легко | очень удобный | есть |
| Название пакета | 8 |
| Neural10 | допускаются любые расширения файлов с обучающими парами |
| NeuroPro | возможности упрощения сети |
| Neural Planner | встроенный графический редактор, аналог табличного редактора |
| BrainMaker 3.10 | набор утилит широкого назначения |
| NeuroShell 2 | очень хорошая справка, собственные алгоритмы обучения |
| Название пакета | 9 |
| Neural10 | ограниченность настройки параметров сети |
| NeuroPro | невозможность сохранения результатов опроса обученной сети |
| Neural Planner | ограниченность по числу алгоритмов обучения |
| BrainMaker 3.10 | ограниченные функциональные возможности |
| NeuroShell 2 | возможность импорта только 5 типов входных файлов |

Исходя из данных таблицы 2, наиболее подходящим пакетом является пакет NeuroShell 2.

**1.6 Минимальные требования к информационной и программной совместимости**

Microsoft Office 97 и выше

Пакет NeuroShell 2

**1.7 Минимальные требования к составу и параметрам технических средств**

MS Windows 9х/2000/Ме/NT/XP

32 Mb RAM

IBM – совместимый персональный компьютер класса 486/66 или выше (желательно Pentium II или выше)

Видеоадаптер SVGA

Монитор VGA и выше

Клавиатура, мышь

**2. Обучение нейронной сети**

Под *обучением* искусственной нейронной сети понимается процесс настройки весовых коэффициентов ее базовых процессорных элементов, результатом чего является выполнение сетью конкретных задач — распознавания, оптимизации, аппроксимации, управления. Достижение подобных целей формализуется критерием качества Q, минимальное значение *minWQ=Q\** которого соответствует наилучшему решению поставленной задачи [5].

## 2.1 Формирование исходных данных

Рассматривается классификация цифр арабского алфавита, написанных шрифтом Arial 10ым размером. Для формирования исходных данных (вектор признаков) использовалось растровое представление цифр (в виде массива из 0 и 1). Размер изображения, исходя из шрифта написания, составил 6х10 квадратов. Отсюда размерность входного вектора – 60.

Выходной вектор (-вектор, значения всех координат которого должны быть равными 0, за исключением координаты, соответствующей выходному элементу, представляющему искомый класс(значение этой координаты должно быть равным 1)) имеет размерность 10, в нем номер позиции единицы соответствует номеру цифры. Таким образом, обучающая пара содержит 60 + 10 = 70 значений.

Для каждого желаемого образа формируется соответствующий массив, записанный в одну строку (сверху вниз слева направо), и также в процессе обучения используются реальные выходные значения, которые записаны в конце строки сформированного массива.

**2.2 Окончательный выбор модели, структуры и параметров обучения НС**

В NeuroShell 2 предлагается несколько моделей НС.

По рекомендациям разработчиков пакета критерием остановки обучения будет: события после минимума > 20000, так как с использованием встроенной калибровки этот критерий позволяет избежать переучивания сети и запоминания тестовых примеров.

**О**кончательный выбор нейронной сети основан на анализе полученных экспериментально результатов для средней ошибки < 0,000001.

Таблица 1 Окончательный выбор нейронной сети

|  |  |  |
| --- | --- | --- |
| Нейронная сеть | Время обучения | Количество эпох |
| Многослойный персептрон | 0:09 | 4047 |
| Реккурентная сеть | 2:13 | 29274 |
| Сеть Ворда | 0:00 | 224 |
| Сеть Ворда с обходным соединением | 0:00 | 87 |
| Трехслойная сеть с обходным соединением | 0:01 | 149 |

Из анализа таблицы 1 следует, что наилучшей моделью является сеть Ворда с двумя блоками скрытого слоя, имеющими разные передаточные функции, с обходным соединением.

Окончательный выбор структуры нейронной сети основан на анализе полученных экспериментально результатов для средней ошибки < 0,000001.

Изменяем количество скрытых нейронов и передаточные функции скрытых блоков для выбора оптимальной структуры сети.

Передаточная функция блока 1 – Б1.

Передаточная функция блока 2 – Б2.

Таблица 2 Окончательный выбор структуры нейронной сети

|  |  |  |
| --- | --- | --- |
| Параметр сети | Время обучения | Количество эпох |
| Количество нейронов - 30 | 0:00 | 99 |
| Количество нейронов - 40 | 0:00 | 87 |
| Количество нейронов - 50 | 0:00 | 86 |
| Б1 – Гаусса, Б2 - Гаусса | 0:00 | 87 |
| Б1 – Гаусса, Б2 – комп. Гаусса | 0:00 | 87 |
| Б1 – Гаусса, Б2 – sin | 0:00 | 87 |
| Б1 – комп. Гаусса, Б2 – комп. Гаусса | 0:00 | 86 |
| Б1 – sin, Б2 - sin | 0:00 | 87 |
| Б1 – комп. Гаусса, Б2 - логистическая | 0:00 | 112 |

Из анализа таблицы 2 следует, что оптимальной структурой сети является:

• комплем. Гауссова функция для блока 1;

• комплем. Гауссова функция для блока 2;

• количество скрытых нейронов – 50(25 в каждом из блоков);

Выбор параметров обучения основан на анализе результатов эксперимента для сети Ворда с двумя скрытыми блоками, имеющими разные передаточные функции, с обходным соединением, с количеством скрытых нейронов равном 50.

Таблица 3 Выбор скорости обучения

|  |  |  |
| --- | --- | --- |
| Скорость обучения | Время обучения | Количество эпох |
| 0,1 | 0:00 | 86 |
| 0,2 | 0:00 | 62 |
| 0,3 | 0:00 | 49 |
| 0,4 | 0:00 | 37 |
| 0,5 | 0:00 | 37 |
| 0,6 | 0:00 | 37 |
| 0,7 | 0:00 | 36 |

Из анализа таблицы 3 следует, что оптимальная скорость обучения равна 0,7.

Проанализируем теперь зависимость обучаемости сети от момента при выбранной скорости обучения.

Таблица 4 Выбор момента

|  |  |  |
| --- | --- | --- |
| Момент | Время обучения | Количество эпох |
| 0,1 | 0:00 | 36 |
| 0,2 | 0:00 | 37 |
| 0,3 | 0:00 | 37 |
| 0,4 | 0:00 | 37 |

Из анализа таблицы 4 следует, что оптимальное значение момента равно 0,1.

Таким образом оптимальными параметрами обучения для данной сети являются:

* Скорость обучения – 0,7
* Момент – 0,1

**2.3 Блок-схема алгоритма обучения**

Блок-схема алгоритма обучения с обратным распространением ошибки[6]:

начало

прочитать входной образец и соответствующий ему выходной образец

для входного слоя установить совокупный ввод каждого элемента равным соответствующему элементу входного вектора. значение вывода каждого элемента установить равным вводу

для элементов скрытого слоя вычислить совокупный ввод и вывод *netj=w0+Σxiwij oj=1/(1+exp(-netj))*

для элементов выходного слоя вычислить совокупный ввод и вывод *netj=w0+Σxiwij oj=1/(1+exp(-netj))*

Сходство = true

попадает ли разность между целевым выходным образцом и реальным выводом сети в допустимые рамки?

для каждого выходного элемента вычислить его ошибку*.*

*δj=(tj-oj)oj(1-oj)*

для последнего скрытого слоя вычислить ошибку каждого элемента *δk=oj(1-oj)Σ δkwkj*

для всех слоев обновить значение весов каждого элемента

*△wij(n+1)=η(δjoi)+α △wij(n)*

Последний образец?

конец

сходство=true

Сходство=true?

нет

нет

есть ещё скрытые слои?

нет

да

да

есть ещё скрытые слои?

да

нет

да

нет

**3 Анализ качества обучения**

Для проверки способностей к обобщению на вход сети подаются зашумленные последовательности входных сигналов. Процент зашумления показывает, какое количество битов входного вектора было инвертировано по отношению к размерности входного вектора.

Исходные и выходные файлы для анализа качества обучения [6,7]:

Листы Шум 7, Шум 13 и Шум 20 в файле cifri.xls для 7%,13% и 20% соответственно.

В силу большой размерности таблиц, содержащих входные и выходные вектора, данные файлы прилагаются к пояснительной записке в электронном виде.

При зашумлении до 13% сеть хорошо решает предложенную задачу. При вводе сигналов с уровнем шума 20% обученная сеть выдает неверные значения на 40% экзаменационных наборах (т.е. сеть не может отдать предпочтение одной из нескольких цифр).

Например, при подаче зашумленного вектора, соответствующего цифре 8, выход сети, соответствующий 8 равен 0,61, выход 1 – 0,73.

При подаче цифры 3 соответствующий выход равен 0,34, выход 8 – 0,51.

Понизить ошибку можно увеличив размерность сетки, которой разбиваются цифры, например 16х16, тогда размерность входного вектора 256.

Кроме того можно понизить среднюю ошибку при обучении сети.

Но, несмотря на все выше сказанное, можно сказать, что сеть хорошо обучилась и обладает хорошими способностями к обобщению.

**4. Тестовый пример**

Для тестирования сети используем экзаменационные данные из файла "test.xls" с уровнем шума 7 %.

Результаты отражены в файле "test.out", который прилагается к пояснительной записке в электронном виде.

Из анализа содержимого файла можно сделать вывод, что обученная сеть справляется с поставленной задачей и обладает хорошей способностью к обобщению.

**5. Выводы**

В данной курсовой работе был решен частный случай задачи классификации цифр арабского алфавита на основе нейронных сетей. Задача была решена только для цифр, написанных шрифтом Arial, 10 размера.

Для решения задачи на других данных (других алфавитов, шрифтов, размеров шрифтов) потребуется ввести некоторые изменения:

• Увеличить количество записей в файле обучения;

• Увеличить количество ячеек, на которые разбивается символ (обычно применяется растр размером 16х16 пикселей);

• Уменьшить среднюю ошибку при обучении нейронной сети;

• Для применения данной схемы к другим размерам букв нужно нормализовать поступающие на вход сети символы к единому размеру, например 16х16.

**Список использованных источников**

1. Стандарт предприятия СТП 1–У–НГТУ–98

2. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика. – М.: Горячая линия – Телеком, 2001. – 382 с.:ил.

3. Уоссермен Ф. Нейрокомпьютерная техника – М.: Мир 1992

4. Электронный учебник по NeuroShell 2

5. Терехов В.А. Управление на основе нейронных сетей. – М.: Высшая школа,2002. – 153 с.: ил.

1. Каллан Р. Основные концепции нейронных сетей
2. Ресурсы сети Интернет(http://alife.narod.ru/)