

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

Кафедра інженерії програмного забезпечення

Пояснювальна записка

до магістерської роботи

на ступінь вищої освіти магістр

на тему: **«РОЗРОБКА CRM СИСТЕМИ ОБЛІКУ ТОВАРІВ НА
ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ»**

Виконав: студент 7 курсу,

групи ППЗМ–71 спеціальності

121 Інженерія програмного забезпечення

(шифр і назва спеціальності)

Погребняк М. М.

(прізвище та ініціали)

Керівник Трінтіна Н. А.

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ**

Кафедра – Інженерії програмного забезпечення

Ступінь вищої освіти - «Магістр»

Спеціальність - 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного забезпечення

Негоденко О.В.

« ____ » _____ 2021 року

**З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ
ПОГРЕБНЯКУ МИКОЛІ МИХАЙЛОВИЧУ**

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка автоматизованої персональної хмарної системи з використанням мікрокомп'ютера Raspberry PI для IoT»

Керівник роботи: Трінтіна Н.А., к.т.н., доцент кафедри ІІЗ
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом вищого навчального закладу від «12» березня 2021 року № 65

2. Строк подання студентом роботи «01» червня 2021 року

3. Вхідні дані до роботи:

3.1. Вимоги до кваліфікаційної роботи магістра з актуальних завдань спеціальності;

3.2. Нормативні матеріали (стандарти, ГОСТи);

3.3. Технічні вимоги;

3.4. Науково-технічна література з питань, пов'язаних з темою роботи.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

- 4.1. Порівняльний аналіз результатів, отриманих іншими авторами;
 - 4.2. Методика дослідження;
 - 4.3. Результати дослідження;
 - 4.4. Висновки.
5. Перелік графічного матеріалу:
 - 5.1. Статистика росту ШІ
 - 5.2. Переваги технологій ШІ
 - 5.3. Структура рекомендаційної системи
 - 5.4. Класифікація рекомендаційних систем
 - 5.5. Фільтрація вмісту
 - 5.6. Комбінування методів
 - 5.7. Відповідальності сутностей
 - 5.6. Приклад реалізації
6. Дата видачі завдання «19» квітня 2021 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Аналіз літератури та джерел за темою дипломного проекту.	19.04.2021	Виконано
2	Розроблення та затвердження плану дипломного проекту.	21.04.2021	Виконано
3	Проведення консультації з науковим керівником щодо створення першого розділу.	22.04.2021	Виконано
4	Розробка розділу 1	23.04.2021	Виконано
5	Розробка розділу 2	29.04.2021	Виконано
6	Розробка розділу 3	05.05.2021	Виконано
7	Розробка розділу 4	08.05.2021	Виконано
8	Висновки та оформлення пояснювальної записки дипломного проекту.	10.05.2021	Виконано
9	Підписання необхідних документів у встановленому порядку.	07.06.2021	Виконано
10	Підготовка до захисту та попередній захист дипломного проекту на випусковій кафедрі дипломного проекту		Виконано

Студент _____ Погребняк М. М.

(підпис) (прізвище та ініціали)

Керівник роботи _____ Трінтіна Н. А.

(підпис) (прізвище та ініціали)

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ**

**ПОДАННЯ ГОЛОВІ ДЕРЖАВНОЇ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ
ЩОДО ЗАХИСТУ МАГІСТЕРСЬКОЇ РОБОТИ**

Направляється студент Погребняк М. М. до захисту магістерської роботи
(прізвище та ініціали)

за спеціальністю 121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

на тему: «Розробка CRM системи обліку товарів на основі нейронної мережі»
Магістерська робота і рецензія додаються.

Директор інституту _____ А.П. Бондарчук
(підпис)

Довідка про успішність

Трінтіна Н. А. за період навчання в Навчально-науковому інституті інформаційних технологій з 2020 року до 2021 року повністю виконав навчальний план за спеціальністю, з таким розподілом оцінок за:

національною шкалою: відмінно %, добре %, задовільно %;
шкалою ECTS: A %; B %; C %; D %; E %.

Методист інституту _____
(підпис) (прізвище та ініціали)

Висновок керівника магістерської роботи

Студент Погребняк М. М.

Все це дозволяє оцінити виконану магістерську роботу студента _____
Погребняка М. М. на оцінку « » та присвоїти йому кваліфікацію

Керівник роботи _____ Трінтіна Н. А.
(підпис)

« » 2021 року

Висновок кафедри про магістерську роботу

Магістерську роботу розглянуто(а). Студент Погребняк М. М. допускається до захисту даної роботи в Державній екзаменаційній комісії. Завідувач кафедри Комп'ютерної інженерії

Ткаченко О.М.

(прізвище та ініціали)

РЕФЕРАТ

Тестова частина магістерської роботи 80с, 34 рис, 22 джерела.

ВЕБ-СИСТЕМА, ОБРОБКА ДАНИХ, CRM, ШТУЧНА НЕЙРОНА МЕРЕЖА, C# ASP MVC.NET, MS SQL SERVER 2014, БАЗА ДАНИХ.

Об'єкт дослідження – процес ділового взаємодії з клієнтами на основі штучних нейронних мереж і система рекомендацій товарів.

Предмет дослідження – програмне забезпечення для торговельної компанії, призначене для автоматизації взаємодії з клієнтами.

Мета дипломної роботи – застосування нейронних мереж у CRM системах обліку товарів.

Методи дослідження – аналіз методів навчання штучної нейронної мережі, методи запам'ятовування, засоби розробки та технології розробки системи для його подальшого розгортання на веб-сервері.

Під час роботи над дипломною роботою було розроблено сайт інтернет-магазину з системою управління взаємовідносинами з клієнтами на основі застосування нейронних мереж у системі обліку товарів.

Робота містить опис програмної частини сайту інтернет-магазину:

- введення даних для авторизації товару;
- відправка даних на сервер;
- обробка вхідних значень та вивід рекомендаційних даних до клієнтської сторони;
- проведення тестування готового продукту.

При розробці інтернет-магазину було використано мову C# ML.NET у середовищі програмування Microsoft Visual Studio.

Результати дипломної роботи рекомендується використовувати при розробці CRM систем для існуючих інтернет-магазинів з метою підвищення ефективності взаємодії з клієнтами.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	10
ВСТУП.....	11
1 ВВЕДЕННЯ В ШТУЧНІ НЕЙРОННІ МЕРЕЖІ	13
1.1 Архітектура та задачі CRM систем	13
1.1.1 Принципи CRM-системи	13
1.1.2 Аналітика.....	15
1.1.3 Переваги впровадження	16
1.2 Поняття «штучна нейронна мережа».....	17
1.3 Види штучних нейронних мереж	19
1.4 Введення в машинне навчання.....	28
1.5 Типи задач машинного навчання	30
1.5.1 Регресія.....	30
1.5.2 Класифікація	31
1.5.3 Кластиризація.....	32
1.5.4 Прогнозування	33
1.5.5 Зменшення розмірності	34
1.5.6 Зменшення розмірності	35
Висновок до 1 розділу	37
2 АНАЛІЗ МЕТОДІВ НАВЧАННЯ НЕЙРОМЕРЕЖ.....	38
2.1 Навчання нейронних мереж	38
2.2 Застосування нейронних мереж	38
2.3 Модель механізму запам'ятовування	40
2.4 Нейросимулятор.....	46
Висновки до 2 розділу	49
3 ЗАСОБИ РОЗРОБИ ТА ТЕХНОЛОГІЇ	50
3.1 Вимоги до розробки бази даних для автоматизованої системи обліку товару.....	50
3.2 Використані технології.....	51

3.2.1 C# ASP.NET MVC 4.0.....	51
3.2.2 Веб-сервер IIS	54
3.2.3 MS SQL Server 2014.....	54
Висновки до 3 розділу	57
4 ПРОЕКТУВАННЯ CRM СИСТЕМИ ОБЛІКУ ТОВАРІВ НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ	59
4.1 Створення технічного завдання	59
4.2 Етапи розробки автоматизованої системи управління взаємовідносинами з клієнтами з застосуванням нейромережі.....	59
4.2.1 Розробка програмного коду на мові C# в Visual Studio 2015	61
4.2.2 Детальний опис роботи додатку	63
Висновки до 4 розділу	75
ВИСНОВКИ	76
СПИСОК ЛІТЕРАТУРИ	78
ДОДАТКИ	80

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

БД – база даних;

ШМ – штучний інтелект;

НМ – нейрона мережа;

ШНМ – штучна нейрона мережа;

СУБД – система управління базами даних;

ТЗ – технічне завдання;

CRM – Customer Relationship Management;

IDE – Integrated Development Environment;

MVC – Model View Controller;

IIS – Internet Information Services.

ВСТУП

На сьогоднішній день людство є свідками активного розвитку технологій інтелектуального аналізу даних. Перш за все, їх поява пов'язана з необхідністю аналітичної обробки великих обсягів інформації, які не завжди можливо ефективно реалізувати використовуючи застарілі методи обробки даних. Для більшості світових компаній основною метою від свого програмного забезпечення є отримання корисної інформації, яку вони надалі зможуть монетизувати. Але через проблему необхідності зберігання та обробки великої кількості даних все частіше дана задача є обтяжувальною.

Як використовувати дані, щоб дізнатися, що найбільш вигідно для клієнтів компанії, як найбільш ефективно розподілити ресурси, або як мінімізувати втрати? Для вирішення цих завдань і розробляють новітні технології інтелектуального аналізу, що використовуються для пошуку моделей та закономірностей, прихованих в середовищі даних, які неможливо отримати звичайними методами.

У сучасному світі існує величезна кількість компаній головним фактором успіху яких є автоматизація всіх бізнес-процесів. Управління сучасним підприємством у ринковій економіці – це складний процес, що включає в собі вибір і реалізацію комплексу управлінських дій в поточних тимчасових інтервалах з метою виконання завдання забезпечення його стабільного фінансового і соціально-економічного розвитку.

Штучний інтелект – це властивість інтелектуальних систем, якій за принципом дії схожій з інтелектом людини та допомагають у вирішенні різних складних завдань і розширює свої межі пізнання.

Інснючі, на сьогодні, інтелектуальні системи мають досі вузьку область призначення. Наприклад, програми, які здатні обіграти людину в настільну гру, не здатні вести бесіду та відповісти на питання суперника.

На даний момент впровадження нововведень – неоціненний крок до підвищення ефективності взаємовідносин між клієнтами та компанією, що забезпечує прибутковість підприємства. Усі компанії розуміють, що тільки

оптимізація виробничого процесу вже не вирішує проблему виживання продукту в суворих реаліях XXI століття. Особливо це помітно в сфері послуг, де прибутковість компанії залежить від досконалості методів взаємодії з клієнтами.

Метою дипломної роботи є дослідження CRM-систем обліку товарів з використання штучної нейронної мережі. Для досягнення цієї поставленої мети були такі **завдання**:

1. Дослідити CRM системи обліку товарів на основі нейронних мереж.
2. Оцінити та порівняти існуючі готові рішення, визначити сильні і слабкі сторони кожної з розглянутих CRM систем.
3. Проаналізувати алгоритми рекомендацій на основі великої кількості вхідних даних, розглянути архітектуру нейронних мереж.
4. Розглянути основні етапи розвитку CRM системи на основі штучних нейронних мереж і визначити інструменти для її розробки.

Актуальність даної теми виправдана тим, що на сьогоднішній день інформаційні технології стрімко розвивається і існуючі готові рішення (CRM-системи) не завжди відповідають специфічним вимогам, які ставить перед собою підприємство.

Штучний інтелект значно покращив механізми рекомендацій в інтернет-магазинах і сервісах. Алгоритми машинного навчання аналізують вашу поведінку на сайті та порівнюють його з мільйонами інших користувачів. На основі чого і буде подальшу взаємодію з клієнтом і визначає, який продукт або послуга зацікавить клієнта найбільше.

Реферальний механізм забезпечує Amazon 35% продажів. Алгоритм Brain, використовуваний YouTube для рекомендації контенту, дозволив людям знайти майже 70% відео, що переглядаються на сайті. WSJ повідомив, що використання штучного інтелекту для рекомендацій є одним з факторів, які сприяли 10-кратному збільшенню аудиторії за останні декілька років.

1 ВВЕДЕННЯ В ШТУЧНІ НЕЙРОННІ МЕРЕЖІ

1.1 Архітектура та задачі CRM систем

Поняття CRM прийшло з англійської мови – Customer Relationship Management System – і в буквальному сенсі позначає система менеджменту взаємовідносин з клієнтами. Дослівний переклад не може передати всього опису, тому слід розібратися докладніше. Не можна одним словом визначити значення CRM-системи, оскільки це навіть не програмний продукт, а модель бізнес-планування, в центрі якої клієнт.

Для реалізації даного процесу збирається інформація про кожного клієнта компанії, а потім використовується для побудови результативних відносин з ним. Бізнес був би неефективним, якщо б не ставив на перше місце отримання прибутку, а впровадження CRM-system якраз приносить дивіденди від роботи з клієнтами. Індивідуальна програма співпраці з кожним замовником допомагає утримати існуючих клієнтів і розширити базу за рахунок залучення нових.

CRM входить в єдину базу даних компанії і являє собою складну архітектуру. Це допомагає автоматизувати процеси роботи з замовниками, внаслідок чого компанія має можливість запропонувати клієнту певні продукти або послуги точно в той час, коли вони йому необхідні. Така концепція, коли на перший план ставиться споживач, а не товар, робить компанію конкурентоспроможною на ринку. Архітектура CRM-системи складається з наступних модулів:

- фронтальна частина (забезпечує обслуговування на точках продажів);
- оперативна частина;
- сховище даних;
- аналітична підсистема;
- розподілена система підтримки продажів.

1.1.1 Принципи CRM-системи

Взаємодія відділів дозволяє залучити нових клієнтів і не розгубити старих. Це відбувається тому, що кожен співробітник, звертаючись до єдиної бази, має можливість бачити детальну і повну картину про покупця, на основі якої приймається рішення, яке буде відображено в цій базі. Все це можливо, коли дотримуються основні принципи CRM-систем:

- Існування загального центру, де відбувається зберігання інформації;
- Можливість взаємодії з клієнтами за допомогою всіх наявних каналів зв'язку: від телефону до соціальних мереж;
- Проведення постійного аналізу зібраної інформації для прийняття рішення про подальшу роботу компанії.

Друге глобальне питання стосується основних функцій і завдань CRM-системи, які вона покликана вирішувати на підприємстві.

Створення і ведення клієнтської бази. В CRM-системах створюється обліковий запис або картка з контактною інформацією для конкретного замовника, в якій відображаються всі взаємодії з контрагентом. Завдяки цій функції CRM-система вирішує цілий ряд важливих завдань:

- Збереження контактів і історії спілкування з клієнтами при звільненні працівника. Його наступник отримає всі необхідні дані про контрагентів і зможе без зволікань приступити до роботи.

- Можливість розвивати відносини з потенційними замовниками і нагадувати про себе діючим клієнтам. За допомогою CRM-системи ваші менеджери зможуть розсилати інформацію про нові продукти або послуги, а також привітання з датами, важливими для компанії замовника.

- Організація ефективної взаємодії між підрозділами вашої компанії. Спільна робота з єдиною клієнтською базою клієнтів дає можливість вашим співробітникам з різних відділів підходити до завдання комплексно в рамках єдиної маркетингової стратегії. Наприклад, поєднуючи персоналізовані пропозиції щодо товарів і послуг при пошуку нових замовників з PR-акціями або event-заходами.

– Підвищення ефективності продажів. Вирішення цього завдання в CRM-системах здійснюється за допомогою додавання в загальну базу даних відомостей про потенційні угоди і супровід їх на всіх етапах воронки продажів. У картці клієнта відбивається повна інформація про всі його взаємодії з компанією, починаючи з «холодного» дзвінка або вхідної заявки.

– Систематизація роботи менеджера в процесі ведення клієнта по всіх етапах воронки продажів збільшує ймовірність успішної операції.

Підвищення якості обслуговування клієнтів. CRM-система надає великі можливості задоволення потреб замовника. Наприклад, на основі даних, зібраних в CRM по кожному клієнту, можна сформувати для нього найцікавіші пропозиції, не пропустити зручний для клієнта час доставки, а також отримати зворотній зв'язок у вигляді подяк або претензій. Завдяки інформації, що зберігається в CRM-системі, ваші менеджери не забудуть привітати представників замовника з днем народження або ювілеєм компанії, запросити на корпоративний вечір, що також працює на підвищення лояльності до вашої організації.

1.1.2 Аналітика

До функції аналітики відноситься все, що пов'язано з отриманням і обробкою інформації про ефективність роботи з контрагентами та комерційну діяльність компанії в цілому. CRM-система дає можливість проводити аналіз за різними напрямками і тимчасовими проміжками. Керівник отримує відразу кілька зведених звітів: статистику по продажах і операціях, перелік незавершених проектів, інформацію про виконання завдань співробітниками, відгуки від клієнтів. Отримані дані наочно покажуть загальний стан справ в компанії за цікавий для нього період часу, ефективність роботи кожного співробітника, якість сервісу. На основі цих відомостей керівник зможе спрогнозувати рівень продажів і побачити точки зростання для комерційної діяльності організації. Зведені звіти також будуть корисні при розробці довгострокової стратегії розвитку вашої компанії.

Розмежування прав. Щоб перераховані вище функції могли найбільш

ефективно працювати, в CRM-системі підтримується розмежування прав. Керівник зможе особисто давати доступ до ділових операцій певним фахівцям, стежити за реалізацією проектів і контролювати виконання завдань. CRM-система дозволить визначити і зафіксувати зони відповідальності, обов'язки як кожного члена команди, так і цілих підрозділів.

Розробка CRM-системи передбачає повну автоматизацію бізнес-процесів під потреби конкретної організації. Цей варіант більш затребуваний середнім і великим бізнесом, котрі вважають, що вигідніше зробити систему під себе і отримати більш високий зріст продажів, ніж обмежуватися гарантованими витратами на версію коробочки з не таким високим ростом.

В нашій роботі ми розглянемо розробку CRM системи на базі нейронної мережі.

1.1.3 Переваги впровадження

До переваг впровадження CRM системи належать:

1. Зниження витрат на продаж. Привабливо для інтернет-компанії, оскільки правильне використання CRM системи допомагає збільшити продуктивність, автоматизувати типові операції. В інтернет-магазині можна привести відразу кілька прикладів даної оптимізації: автоматичне формування відвантажувальних документів, автоматичне інформування клієнта про статус замовлення, автоматичне створення транспортної накладної для відвантаження замовлення кур'єрською компанією.

2. Скорочення неефективних відділів. Оптимізація процесів допоможе виключити людський фактор і налагодити постійні, безперебійні потоки руху інформації між клієнтів і постачальником, а також всередині самої компанії, яка обробляє замовлення.

3. Формування спеціальної пропозиції для клієнта. Цей пункт актуальний в рівній мірі, як для інтернет-магазину, так і для оптового, наприклад, підприємства. В рамках інтернет-магазину можливо відстеження

повторного замовлення для надання знижки або іншого спеціального пропозиції для клієнтів. Так само, це дає можливість виявити проблеми, які виникали з клієнтом в процесі обробки замовлення, це дозволяє, по-перше, вести роботу з даним клієнтом різними менеджерами, що дозволяє співробітникам бути більш багатофункціональними і ефективними. По-друге, дана інформація дозволить зробити додаткову пропозицію, яке компенсує можливі негативні враження у клієнта.

1.2 Поняття «штучна нейронна мережа»

Теорія нейронних мереж виникла з досліджень мозку і пов'язана із спробами відтворення здатності нервових біологічних систем до навчання і виправлення помилок, моделюючи низькорівневу структуру мозку [1].

Ще в середні віки люди міркували про можливість сконструювати механічну людиноподібну мислячу машину, а алхіміки розробляли рецепти вирощування гомункулів - штучних людей з пробірки. Трохи пізніше з'явилися і цілком наукові обґрунтування можливості створення штучного інтелекту – в XVII столітті французький мислитель Рене Декарт сформулював механістичну теорію, припустивши, що тварина є якимось аналогом складного механізму.

Близькі сучасним уявлення про завдання штучного інтелекту сформувалися в XIX столітті. У 1830-і роки англійський математик Чарльз Беббідж придумав концепцію аналітичної машини, яка, як стверджував розробник, могла б розраховувати ходи для гри в шахи. Менш ніж через сторіччя - в 1914 році - концепція втілилася в життя: директор одного з іспанських технічних інститутів Леонардо Торрес де Кеведо виготовив електромеханічний пристрій, здатний розігрувати найпростіші шахові ендшпілі.

Втім періодом народження поняття «штучний інтелект» прийнято вважати 1930-ті роки, а його творцем - англійського математика Алана Тьюринга. Запропонована ним в 1936 році абстрактна обчислювальна машина Тьюринга дозволила формалізувати поняття алгоритму, яке до сих пір використовується в

безлічі теоретичних і практичних досліджень. Пізніше - в 1950 році - Тьюринг запропонував вважати інтелектуальною такою машину, яку випробувач в процесі спілкування з нею не зможе відрізнити від людини, і розробив для оцінки машинного інтелекту спеціальний тест.

Сам термін «штучний інтелект» (artificial intelligence, AI) з'явився вже після смерті Тьюринга - в 1956 році в ході конференції в Університеті Дартмута (США), що зібрала провідних вчених, які працюють в цьому напрямку, а його безпосереднім автором вважається Джон МакКарті, основоположник функціонального програмування і винахідник мови Lisp.

У 1965 році Джозеф Вайценбаум з Массачусетського технологічного інституту створив комп'ютерну програму «Еліза», яка могла вести нескладні діалоги. Можна сказати, це був перший чат-бот, прообраз сучасної Siri. Робот Фредді, розроблений в 1973-м дослідниками з Единбурзького університету, використовував зір для пошуку і складання моделей. А в 1979-му з'явилася «Стенфордський візок» - перший контрольований комп'ютером автономний автомобіль, подоба сучасних безпілотників. Після деякого спаду новий сплеск інтересу до ІІ стався в середині 1990-х. Мабуть, найвідомішим успіхом цього періоду можна вважати перемогу, яку в 1997 році суперкомп'ютер IBM Deep Blue здобув над чемпіоном світу з шахів Гаррі Каспаровим.

Втім, яких би успіхів ні досягали розробники, завжди знаходилися скептики, які заявляли, що навіть якщо комп'ютер навчився добре вирішувати якісь завдання, це ще не означає, що машина дійсно самостійно мислить. Велика частина сучасних вчених сходяться на думці, що штучний інтелект - це сукупність методів та інструментів вирішення різних складних прикладних задач, що використовують принципи і підходи, аналогічні роздумам людини над їх вирішенням (такі технології ще називають когнітивними).

Сьогодні зазвичай мова йде про програми, які в міру накопичення даних самі вчаться їх класифікувати, розпізнаючи зображення, тексти, аудіофайли та будь-які інші об'єкти, з кожним разом вдосконалюючись і виконуючи поставлене завдання все краще. Тобто штучний інтелект набуває здатності, яка до сих пір дійсно

вважалася винятково людською прерогативою, – здатність до навчання.

Основою досліджень штучного інтелекту в 60-80-і рр. були експертні системи. Такі системи гуртувалися на високорівневому моделюванні процесу мислення (зокрема, на представленні, що процес нашого мислення побудований на маніпуляціях з символами). [2]

Нейронні мережі включають в собі широкий спектр питань з різноманітних областей науки, таких як: математики, фізика, інформатики. Самому поняттю "нейронні мережі" важко дати чітке визначення, оскільки не має точного критерію досягнення комп'ютером «інтелекту».

Штучні нейронні мережі (ШНМ) – сукупність моделей біологічних нейронних мереж, що являють собою мережу елементів пов'язаних між собою синоптичними сполуками, що обробляє вхідні дані та формує набір вихідних сигналів. Зазвичай нейронна мережа оперує цифровими, а не символічними величинами.[14]

Під час розробки штучного інтелекту велику частку становить машинне навчання, яка вивчає методи побудови алгоритмів, що здатні самонавчатися. Це є необхідно для побудови чіткого рішення задачі. У даній ситуації простіше буде не шукати вже готовий алгоритм рішення, а створити механізм, який сам відшукає метод для його пошуку. У наукових журналах можна часто зустріти термін «глибинне» або «глибоке» навчання. Під цим , у більшості випадків, розуміють "нейронні мережі".

1.3 Види штучних нейронних мереж

У загальних рисах ми визначилися з тим, що ж таке нейронна мережа. Тепер прийшов час поговорити про їх різновиди і типи, тобто про класифікацію. Але тут потрібно невелике уточнення. Кожна нейронна мережа включає в себе перший шар нейронів, званий вхідним. Цей шар не виконує будь-яких перетворень і обчислень, його завдання в іншому: приймати і розподіляти вхідні сигнали по інших нейронах. І цей шар єдиний, який є загальним для всіх типів нейромереж, а критерієм для

поділу є вже подальша структура:

1. Одношарова структура нейронної мережі. Являє собою структуру взаємодії нейронів, в якій сигнали зі вхідного шару відразу направляються на вихідний шар, який, власне кажучи, не тільки перетворює сигнал, але і відразу ж видає відповідь. Як вже було сказано, 1-й вхідний шар тільки приймає і розподіляє сигнали, а потрібні обчислення відбуваються вже в другому шарі. Вхідні нейрони є об'єднаними з основним шаром за допомогою синапсів з різними вагами, що забезпечують якість зв'язків.

2. Багатошарова нейронна мережа. Тут, крім вихідного і вхідного шарів, є ще кілька прихованих проміжних шарів. Число цих шарів залежить від ступеня складності нейронної мережі. Вона більшою мірою нагадує структуру біологічної нейронної мережі. Такі види були розроблені зовсім недавно, до цього всі процеси були реалізовані за допомогою одношарових нейронних мереж. Відповідні рішення мають більші можливості, якщо порівнювати з одношаровими, адже в процесі обробки даних кожен проміжний шар - це проміжний етап, на якому здійснюється обробка і розподіл інформації.

Крім кількості шарів, нейронні мережі можна класифікувати у напрямку розподілу інформації по синапсах між нейронами:

1. Нейромережі прямого поширення (односпрямовані). У цій структурі сигнал переміщається строго по напрямку від вхідного шару до вихідного. Рух сигналу в зворотному напрямку не здійснюється і в принципі неможливо. Сьогодні розробки цього плану поширені широко і успішно вирішують завдання розпізнавання образів, прогнозування та кластеризації.

2. Рекурентні нейронні мережі (із зворотними зв'язками). Тут сигнал рухається і в прямому, і в зворотному напрямку. У підсумку результат виходу здатний повертатися на вхід. Вихід нейрона визначається ваговими характеристиками і вхідними сигналами, плюс доповнюється попередніми виходами, що знову повернулися на вхід. Цим нейромережам властива функція короткочасної пам'яті, на підставі чого сигнали відновлюються і доповнюються під час їх обробки.

3. Радіально-базисні функції.

4. Самоорганізуючі карти.

Але це далеко не всі варіанти класифікації та види нейронних мереж. Також їх ділять:

1. Залежно від типів нейронів:

- однорідні;
- гібридні.

2. Залежно від методу нейронних мереж з навчання:

- навчання з учителем;
- без вчителя;
- з підкріпленням.

3. За типом вхідної інформації нейронні мережі бувають:

- аналогові;
- виконавчі;
- образні.

4. За характером налаштування синапсів:

- з фіксованими зв'язками;
- з динамічними зв'язками.

Ще існують поняття гетероасоціативної або автоасоціативної нейромережі.

Розглянемо конкретніше характеристику нейромереж.

Перцептрон – тип штучного нейрона, який було розроблено Френком Розенблатом в 1950-их – 1960-их роках. Хоча у сучасних роботах частіше використовують сигмоїдальну модель штучного нейрона, щоб зрозуміти, як працює сигмоїдальний нейрон, необхідно розглянути структуру і принцип роботи перцептрону. Перцептрон приймає на вхід певні значення $x_1, x_2 \dots x_n$, а на виході видає бінарний результат (рисунок 3.2). Розенблат запропонував використання, так званої, ваги ($w_1, w_2, \dots w_n$) – чисел, що відображають важливість вкладу кожного із входів нейрона для кінцевого результату [13].

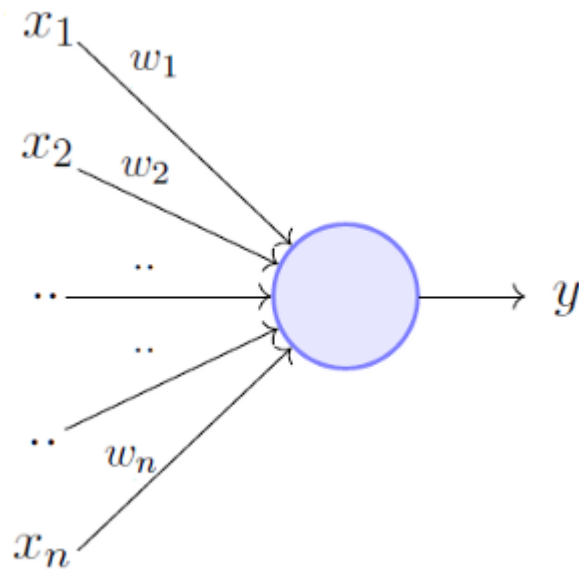


Рисунок 1.1 - Нейронна мережа Хопфілда

Розенблатт висунув ініціативу використовувати ваги – числа, що виражають вагомість вложення усіх входів в кінцевий результат. Зважена вага зіставляється з крайнім значенням (threshold), в кінці, як результат буде 0 або 1. Межа також є параметром нейрона.

Перцептрони можуть бути визначені як штучні нейронні мережі:

- Маючи тільки один прихованих шар;
- Маючи порогову активувальну функцію;
- Маючи прямий розповсюджуваний сигнал.

Навчання перцептрона – зміна ваг у матриці в процесі навчання. Існують 4 історично сформованих види:

- Перцептрон з наявністю одного прихованого шару;
- Одношаровий перцептрон: входні дані пов'язані з вихідними при допомозі систем ваг (найпростіша мережа прямого поширення);
- Багатошаровий перцептрон за Розенблаттом з присутністю додаткових прихованих шарі;
- Багатошаровий перцептрон за Румельхардом з присутністю додаткових прихованих шарі, але навчання відбувається за допомогою метода зворотного поширення помилки.

Якби невелика зміна ваг (або зсуву) викликало невелику зміну на виході мережі, то бажана поведінка нейромережі можна було б отримати за допомогою простих модифікацій зсувів і ваг в процесі навчання. Однак навчання не так просто здійснити, якщо нейронна мережа складається з перцептронів.

Невелика зміна ваг або зсуву одного з перцептронів мережі може кардинально змінити вихідне значення перцептрону, наприклад, з 0 на 1. Тому сама незначна зміна значень одного з елементів мережі може створити значні труднощі в розумінні зміни поведінки мережі. Оскільки завдання навчання нейронної мережі є завданням пошуку мінімуму функції помилки в просторі навчання, то для вирішення даної проблематики можуть використовуватися стандартизовані методи оптимізації. А у випадку перцептрона з одним шаром у якого x входів та y виходів мова йде про пошук мінімуму в y -мережевому просторі.

Залежно від критерію класифікації, можна розділити нейронні мережі на основні види:

- Нейронні мережі прямого поширення і перцептрон - прямолінійні, передають інформацію від входу до виходу. Нейронні мережі часто описуються у вигляді листкового торта, де кожен шар складається з вхідних, прихованих або вихідних клітин. Клітини одного шару не пов'язані між собою, а сусідні шари зазвичай повністю пов'язані. Найпростіша нейронна мережа має дві вхідних клітини і одну вихідну, і може використовуватися в якості моделі логічних вентилів. Нейронні мережі прямого поширення зазвичай навчається за методом зворотного поширення помилки, в якому мережа отримує безлічі вхідних і вихідних даних. Цей процес називається навчанням з учителем, і він відрізняється від навчання без учителя тим, що в другому випадку безліч вихідних даних мережу становить самостійно. Вищезазначена помилка є різницею між введенням і висновком. Якщо у мережі є достатня кількість прихованих нейронів, вона теоретично здатна змоделювати взаємодію між вхідним і вихідними даними. Практично такі мережі використовуються рідко, але їх часто комбінують з іншими типами для отримання нових [3].

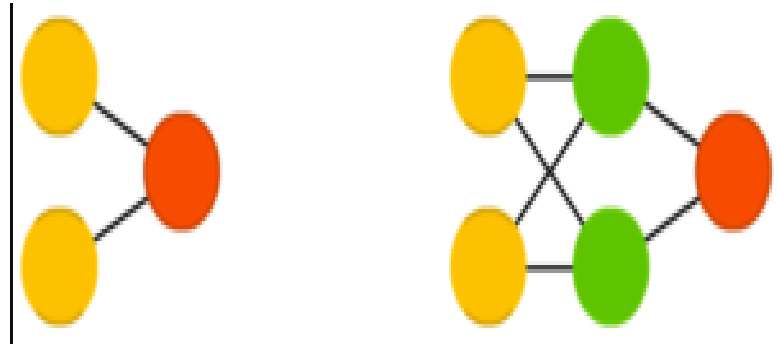


Рисунок 1.2 - Нейронні мережі прямого поширення і перцептор

- Нейронна мережа Хопфілда (рис.1.3) - це нейронна мережа із симетричною матрицею зв'язків. Кожен вузол є входом, яких отримує вхідні параметри, під час навчання стає прихованим, а в кінці стає виходом. Мережа навчається так: значення нейронів встановлюються відповідно до бажаного шаблону, після чого обчислюються ваги, які в подальшому не змінюються. Після того, як мережа навчилася на одному або декількох шаблонах, вона завжди буде зводитися до одного з них (але не завжди - до бажаного). Вона стабілізується в залежності від загальної «енергії» і «температури» мережі. У кожного нейрона є свій поріг активації, що залежить від температури, при проходженні якого нейрон приймає одне з двох значень (зазвичай -1 або 1, іноді 0 або 1). Така мережа часто називається мережею з асоціативною пам'яттю; як людина, бачити половину таблиці, може представити другу половину таблиці, так і ця мережа, отримуючи таблицю, наполовину зашумлену, відновлює її до повної [4].



Рисунок 1.3 - Нейронна мережа Хопфілда

- Ланцюги Маркова – є попередником машини Больцмана та мереж Хопфілда. Зміст цієї архітектури можна описати так: наскільки великі шанси потратити в будь-який наступний вузол за умови, що я перебуваю в даному? Кожен наступний стан залежить тільки від попереднього. Насправді ланцюг Маркова не є повноцінною нейронною мережею, але вони доволі схожі.



Рисунок 1.4 - Ланцюг Маркова

- Машина Больцмана за своїм принципом дуже схожа на мережу Хопфілда, але відмінність в ній полягає у тому, деякі нейрони відмічаються вхідними, а деякі прихованими. Вхідні нейрони надалі стають вихідними. Навчання проходить за алгоритмом порівняльної розбіжності, процес навчання дещо схожий на процес навчання у мережі Хопфілда.



Рисунок 1.5 - Машина Больцмана

- Обмежена машина Больцмана – сильно схожа на машину Больцмана і на мережу Хопфілда, але різницею є її обмеженість: нейрони однакового типу не пов'язані між собою. Обмежену машину Больцмана можна навчати як і нейронну мережу прямого поширення, але є нюанс: усі помилки необхідно передавати у першу чергу в прямому напрямку, потім в зворотному напрямку. Після цього проходить навчання за методом прямого і зворотного поширення помилки.

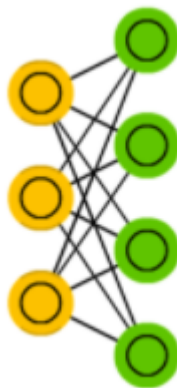


Рисунок 1.6 - Обмежена машина Больцмана

- Автокодувальник – основною ідеєю цієї архітектури є автоматичне кодування інформації (в сенсі стиснення даних). Дана мережа за формою нагадує пісочний годинник, в якому приховані шари значно менші за вхідні та вихідні,

система є семантичною. Мережа може навчати зворотного поширення помилки шляхом передачі вхідних даних і задаючи помилку, що буде дорівнювати різниці між виходом та входом.

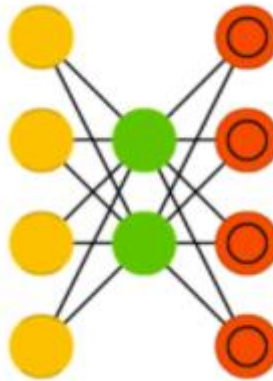


Рисунок 1.7 - Автокодувальник

- Розріджений автокодувальник – можна сказати, що він є протилежністю звичайній архітектурі автокодувальника. Різниця полягає у тому, щоб замість навчати нейромережу відображати дані в меншому обсязі вузлів, буде збільшено кількість вузлів. Замість того, щоб звужитись до центру, мережа там буде роздуватися. Така мережа використовується для роботи з великою кількістю дрібних властивостей набору інформації. Якщо мережу навчати так само, як і автокодувальник, то нічого критичного не відбудеться. Тому окрім вхідних даних додається ще й спеціальний фільтр розрідженості, який пропускає тільки деякі помилки.



Рисунок 1.8 - Розріджений автокодувальник

- Варіаційні автокодувальник – має також схожу архітектуру з звичайним автокодувальником, але тут принцип навчання відбувається по іншому: наближеною імовірністю розподілу вхідних зразків. Мережа може нагадувати машину Больцмана, оскільки бере початок від неї, але спираються на Байєсову математику в ситуаціях, коли мова йде про ймовірних висновках та незалежності. Якщо підвести підсумок – нейромережа приймає до уваги вплив нейронів, у випадку коли одне відбувається в одному місці, а щось інше в іншому, то ці події не обов'язково будуть пов'язані і цей факт потрібно враховувати.

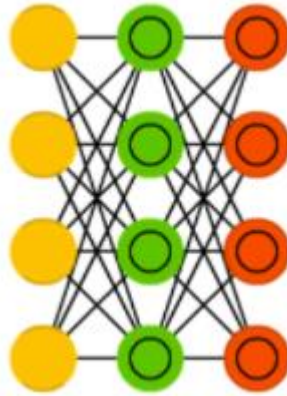


Рисунок 1.9 - Варіаційні автокодувальник

1.4 Введення в машинне навчання

Машинне навчання (МН) – це базова частина штучного інтелекту, жорстка математична дисципліна, яка використовує математичної алгоритми, статистику, теорії ймовірностей, яку можна умовно поділити на два типи:

1. індуктивне навчання – процес навчання зі застосуванням програмування в представленні універсального прикладу вхідних параметрів, гіпотез, прикладів;
2. дедуктивне навчання – засноване на математичних аспектах з завчасно створеними алгоритмами обробки інформації.

МН, як невід'ємна частина штучного інтелекту бере свій початок в результаті

поділу науки про нейронні мережі на методи навчання мереж і види топології архітектури мереж. Вирішувати поставлені завдання можна шляхом трьох основних груп:

1. Навчання з учителем – для кожного випадку є пара (ситуація, рішення) (рис 1.10);
2. Навчання без учителя – система згруповує сутності в кластери і зменшує розмірність вхідних даних (рис 1.11);
3. Навчання з підкріпленням – для кожної ситуації існує пара (ситуація, реакція середовища) (рис 1.12).

Але існують і «не популярні» методи навчання. Наприклад, часткове залучення вчителя або активне навчання (різниця методів несуттєва). Всі вище зазначені способи можна використовувати для класифікації класичних методів МШ і також для алгоритмів навчання нейромереж.

Сьогоднішнє МШ зустрічаються з серйозною проблемою практичності, з причини, що майже немає однорідного простору алгоритмів і методу загального вирішення проблеми індукції.

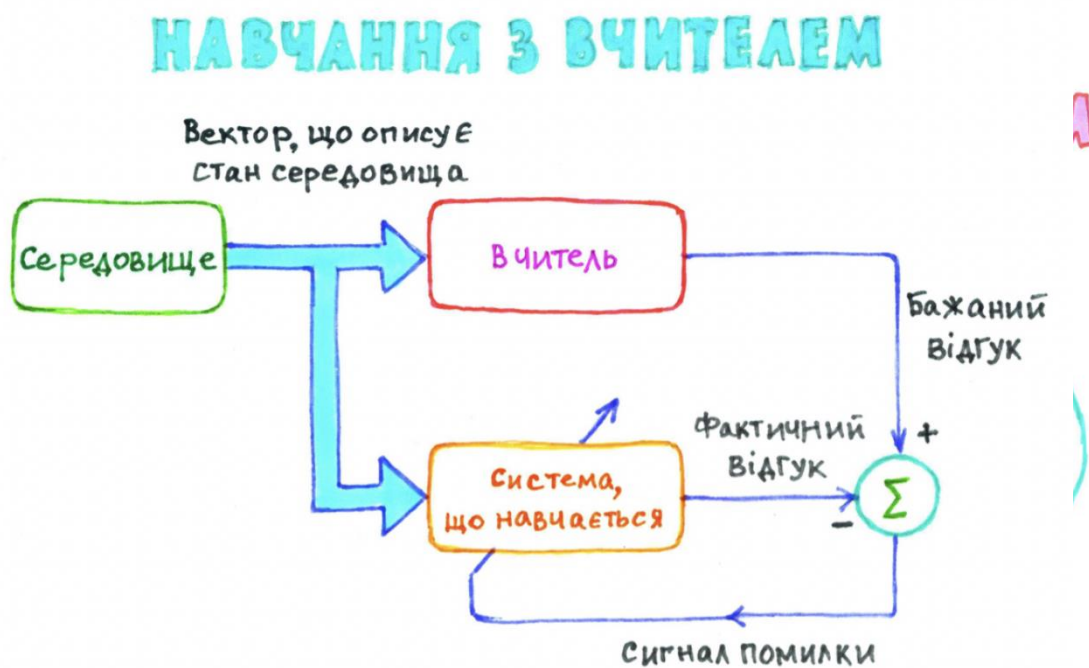


Рисунок 1.10 - Навчання з учителем

1.5 Типи задач машинного навчання

1.5.1 Регресія

Ця модель показує залежність між факторами процесу. Логіка машини полягає у тому, що вона намагається нарисувати криву на графіку, яка б відображала залежність. Модель використовується, щоб передбачити взаємозв'язок між процесом та можливий наслідок такого процесу. Побудований графік регресії відображає позитивний, негативний або відсутній вид зв'язку з фактором. Наприклад, якщо на графіку зображена горизонтальна або вертикальна лінія регресії, то це відображає відсутність будь-якого зв'язку.

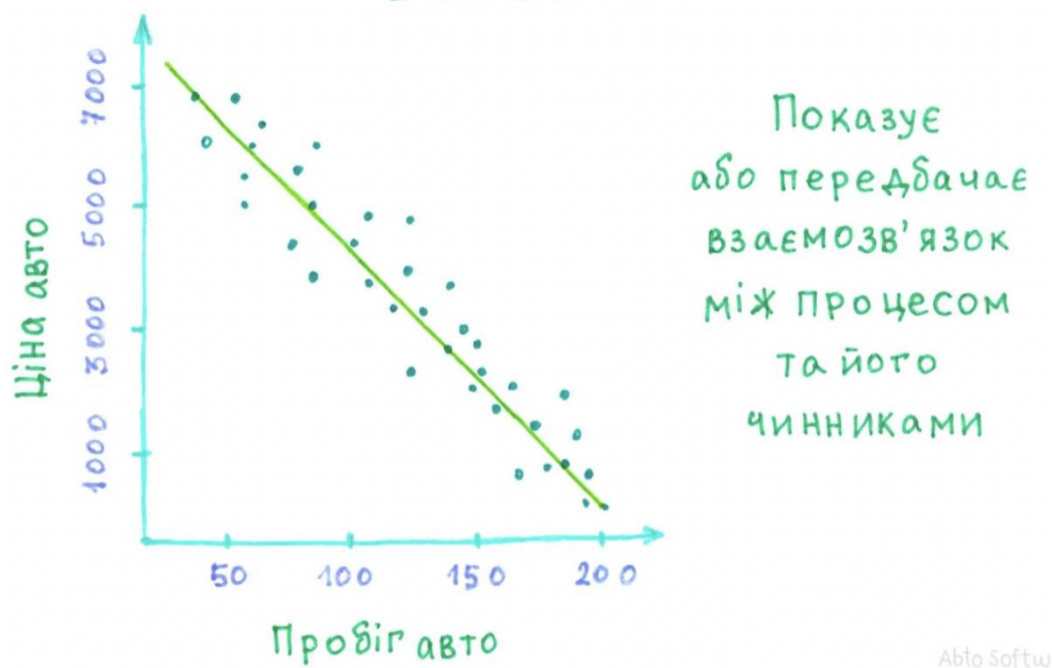


Рисунок 1.13 - Графік регресії

1.5.2 Класифікація

Модель класифікації розділяє об'єкти залежно від заздалегідь вказаних класів. Наприклад, автомобілі та мотоцикли, фарбу за кліром і т.д. На сьогоднішній день роль класифікація виконує величезну кількість завдань, таких як: визначення мови, пошук схожих картинок, застосовується у спам-фільтрах.

Для роботи моделі потрібно заздалегідь розмічену інформацію з обраної категорії та її ознаки, за якими машина буде навчатися.

Одним із класичних прикладів класифікації можна виділити бінарну класифікацію - розділити та відфільтрувати об'єкти за класами. Наприклад, на вході ми отримуємо транспортні засоби, але нам заздалегідь відомо, що це буде або автомобіль, або велосипед. Машина, в свою чергу, розділяє кожен транспортних засіб до одного з описаних класів (рис. 1.14).

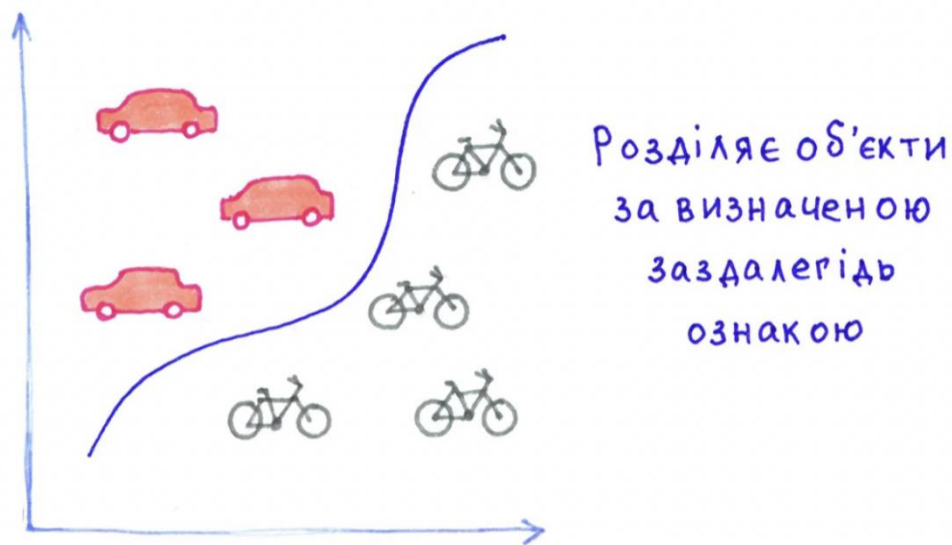


Рисунок 1.14 - Графік класифікації

1.5.3 Кластиризація

Даний алгоритм дає змогу розділити об'єкти в ситуації, коли класи заздалегідь невідомі, а кластери повинні бути сформовані за схожістю елементів. Вона використовує у своїх завданнях сегментації ринку для стиснення зображення, аналізу нової інформації. Машина знаходить схожі ознаки об'єктів та згруповує їх за кластерами. Ми також можемо задати ознаки, за якими необхідно поділити вибірку, вказати кількість класів або все довірити алгоритму, яким сам буде визначати ознаки та кількість класів.

КЛАСТЕРИЗАЦІЯ

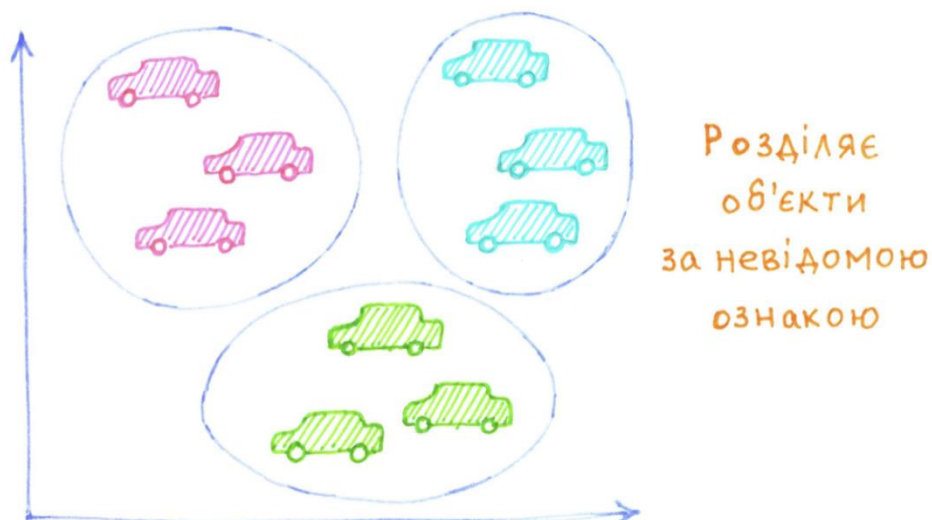


Рисунок 1.15 - Графік кластеризації

1.5.4 Прогнозування

Варто розуміти, що алгоритмів прогнозування безліч і всі вони вирішують завдання пошуку майбутніх даних залежно від часу. При побудові прогнозу алгоритм включає наступні елементи:

3. Аналіз часової послідовності;
4. Формування тренду та його типу;
5. Аналіз послідовності (чи необхідна попередня обробка інформації);
6. В залежності від даних обрати модель та визначити їх параметри;
7. Оцінка точності прогнозування;
8. Аналіз похибки.



Рисунок 1.15 - Графік прогнозування

1.5.5 Зменшення розмірності

Зменшення розмірності – це зведення більшого числа ознак до меншого для зручності їх подальшого використання. Використовується для побудови рекомендаційних систем, пошуку схожих документів чи візуалізації [12].

Навантаження по зменшенню розмірності – певна задача у якій машина знаходить закономірності серед великої кількості вхідних даних. Основною практичною користю метода можна виділити абстракцію (об'єднати декілька ознак в одну). Наприклад, транспортний засіб менший 1м із двома невеликими колесами – мотоцикл.

Також варто відмітити ще одне завдання – алгоритми або рекомендаційні системи, які мають лише певні дані про клієнта, можуть передбачити попит на об'єкти, які могли бути цікавими. При цьому система не буде використовувати спільну оцінку всіх клієнтів, а буде спиратися лише на прерогативи самого клієнта, або у випадку клієнта чий інтереси були би схожі.

ЗМЕНШЕННЯ РОЗМІРНОСТІ

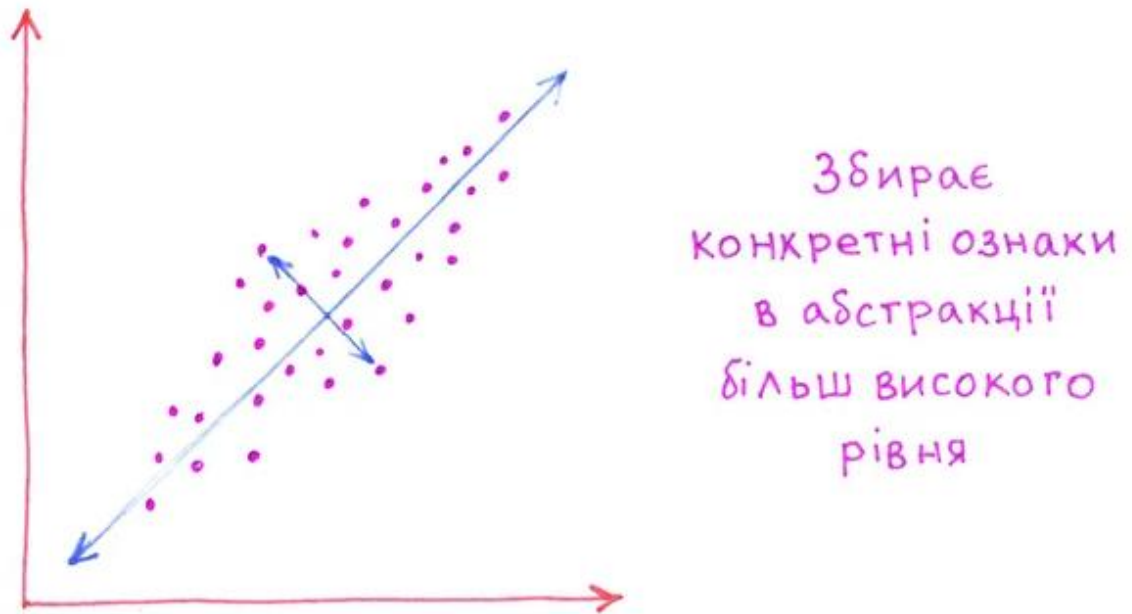


Рисунок 1.17 - Графік зменшення розмірності

1.5.6 Зменшення розмірності

Завдання цього типу полягає у виявленні аномальних відхилень від типових випадків. Може задатися, що даний тип схожий на завдання класифікації, проте це не так. Оскільки виявлення аномалій – доволі рідке явище, тому ситуацій, на основі яких можна було б навчити машину дуже мало.

Наприклад, людина здійснює схожі покупки онлайн використовуючи один додаток, з приблизно однаковими сумами, знаходячись в місті Києві, проте, якщо людина завітає до іншого міста та повторить покупку, то така ситуація з великою ймовірністю буде вважатися аномалією та потребуватиме додаткового підтвердження.

ВІЯВЛЕННЯ АНОМАЛІЙ

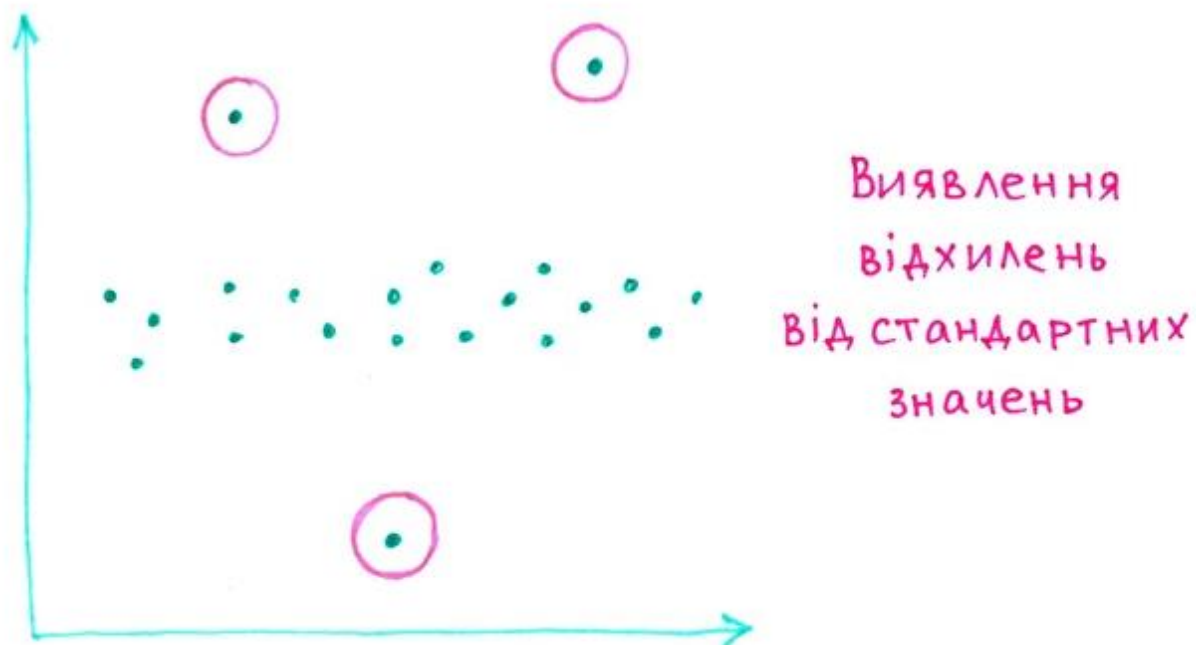


Рисунок 1.18 - Графік виявлення аномалій

Висновок до 1 розділу

Таким чином було описано основні характеристики штучних нейронних мереж та машинного навчання. Незважаючи на те, що штучний інтелект не є досконалим і не може змагатися з людським мозком, у якості обробки інформації, неможливо уявити сучасний світ без використання сучасних технологій.

Кількість інформації, що накопичилась у базах даних різних підприємств вже неможливо обробляти звичними інструментами. І нейронні мережі та машинне навчання - потужний інструмент в руках людини, який використовуються для поліпшення рутинної роботи з оброблення даних та прийняття рішень.

2 АНАЛІЗ МЕТОДІВ НАВЧАННЯ НЕЙРОМЕРЕЖ

2.1 Навчання нейронних мереж

Основною можливістю мозку людини є здатність до навчання, але коли ми розглядаємо на прикладі штучної нейронної мережі – навчання розглядається, як процес налагодження ваг, зв'язків мережі, заради результативного вирішення поставленої цілі. У багатьох ситуаціях нейронна мережа налагоджує вагу зв'язків за заздалегідь описаним принципом навчання. Мережа з можливістю самостійно навчатися на підручних прикладах значно цікавіше за альтернативні системи, які працюють за заздалегідь прописаним алгоритм дій.

Для побудови процесу самонавчання мережі варто мати прототип зовнішнього середовища з нейромережею та знайти потрібні дані в ній. Також варто розуміти, як саме змінити вагові границі в даній мережі.

Можна виділити три базові моделі навчання: «з учителем», «без вчителя» і «змішана». Модель навчання «з учителем» використовує заздалегідь прописані правильні відповіді на всілякий вхідний запит. Ваги задаються таким чином, щоб мережа вирішувала поставлену задачу мінімальним коефіцієнтом погрешності. У варіанті навчання, де не використовується учитель, не має необхідності у володінні знань з правильними відповідями. В даному варіанті йде робота з внутрішньою структурою інформації, що дає змогу відфільтрувати усі вхідні показники за категоріями. У варіанті з використання змішаного типу навчання, деякі ваги задаються шляхом навчання з учителем.

2.2 Застосування нейронних мереж

Основним етапом аналізу нейромереж є класифікації та розпізнавання образів. Розпізнавальні образи можуть бути різні за своїми характеристиками: зображення, звук, символи. В момент навчання їх показується кілька образів із прив'язкою до. Комплекс ознак повинен точно класифікувати зразок за класом. В

ситуації, коли ознак недостатньо для одностайного рішення, мережа може допустити погрішність та пов'язати зразок з кількома класами. Після закінчення навчання мережі їй можна пред'являти невідомі раніше образи і отримувати відповідь про належність до певного класу [5].

Топологія комп'ютерної мережі описується кількістю нейронів у вихідному шарі та буде дорівнювати кількості визначених класів, встановлюється відповідність між класом представником та між виходом нейронної мережі. В той момент, коли нейрона мережа висуває певний образ, на виході повинна з'явитися показник, що образ належить саме даному класу. Водночас на незадіяних виходах повинна з'явитися показник того, що образ класу не залежить мережі. У випадку, коли на двох або більше виходах є показник, що належить до класу – мережа «не впевнена» у своєму рішенні.

Прийняття рішень і управління – це завдання близьке до задачі класифікації. Класифікації підлягають ситуації, характеристики яких надходять на вхід нейронної мережі. На виході нейронної мережі з'являється ознака рішення, яке вона прийняла, в якості вхідних сигналів використовуються різні критерії опису стану керованої системи [22].

Кластеризація процедура під час роботи якої вхідні сигнали розділяються на класи, за умови, що властивості класів завчасно не відомі. Наприкінці процесу навчання нейромережа може з'ясувати клас до якого належить вхідний сигнал. Крім того, у мережі є можливість сповістити про те, що вхідний сигнал не належить до жодного з вищеперерахованих класів і це буде властивістю наступних класів, яких немає у початковому реєстрі даних. Можна зробити висновок, що така мережа може знаходити нові ще невідомі класи сигналів.

Потенціал нейромережі до прогнозування швидко впливає на можливість до типізування та виявлення прихованих залежностей між вхідними та вихідними сигналами. В кінці навчання мережа може самостійно бачити наперед значення будь-якої послідовності на базі деяких попередніх висновків. Необхідно звернути увагу, що прогнозування можливо тільки за умов, коли попередні зміни без сумніву

визначають майбутні. Як приклад, прогнозування котирувань певної акції на основі котирувань за минулий квартал може виявитися успішним (а може і не виявитися), тоді як прогнозування результатів завтрашньої лотереї на основі даних за останній період часу може майже не дати ніяких результатів.

Здатність мережі до знаходження спільними зв'язками між різними параметрами дає можливість висловити дані великої ритмічності більш стисло, при умові, що дані тісно взаємопов'язані один з одним.

Асоціативна пам'ять – це процес відновлення вихідного списку даних з фрагменту інформації. Асоціативна пам'ять дає можливість відновлювати вихідний сигнал з пошкоджених вхідних даних.

Neural Network Toolbox – це пакет розширення MATLAB, що містить засоби для проектування, моделювання, розробки та візуалізації нейронних мереж [7].

Мережеві технології дозволяють вирішувати поставлені завдання, рішення яких базовими методами неможливо або, у випадку, коли трапляється перешкод через яку неможливо вирішення поставленої цілі. Пакет забезпечує всебічну підтримку типових непромережених парадигм і має відкриту модульну архітектуру. Пакет містить функції командного рядка і графічний інтерфейс користувача для швидкого покрокового створення нейромереж.

2.3 Модель механізму запам'ятовування

Існує думка про те, що нейромережа формується під конкретні вимоги. Однак в природі є ідеальна, універсальна, уніфікована нейромережа — наш мозок. Кожен нейрон можна вважати своєрідним процесором: він підсумовує з відповідними вагами сигнали, що приходять від інших нейронів, виконує нелінійну (наприклад, порогову) функцію і передає результуюче значення пов'язаним з ним нейронам. Відповідно до чинного правила "все або нічого" в найпростіших моделях нейронів вихідний сигнал приймає двійкові значення: 0 або 1. Значення 1 відповідає перевищенню порога збудження нейрона, а значення 0 - порушення нижче

порогового рівня. В одній з перших моделей нейрона, так званої моделі МакКаллока-Питтса (рис. 2.1), запропонованої в 1943 р., нейрон вважається бінарним елементом. Вхідні сигнали підсумовуються з урахуванням відповідних ваг в суматорі, після чого результат порівнюється з граничним значенням. Модель МакКаллока-Питтса — це дискретна модель, в якій стан нейрона в даний момент розраховується за значеннями його вхідних сигналів в попередній момент.

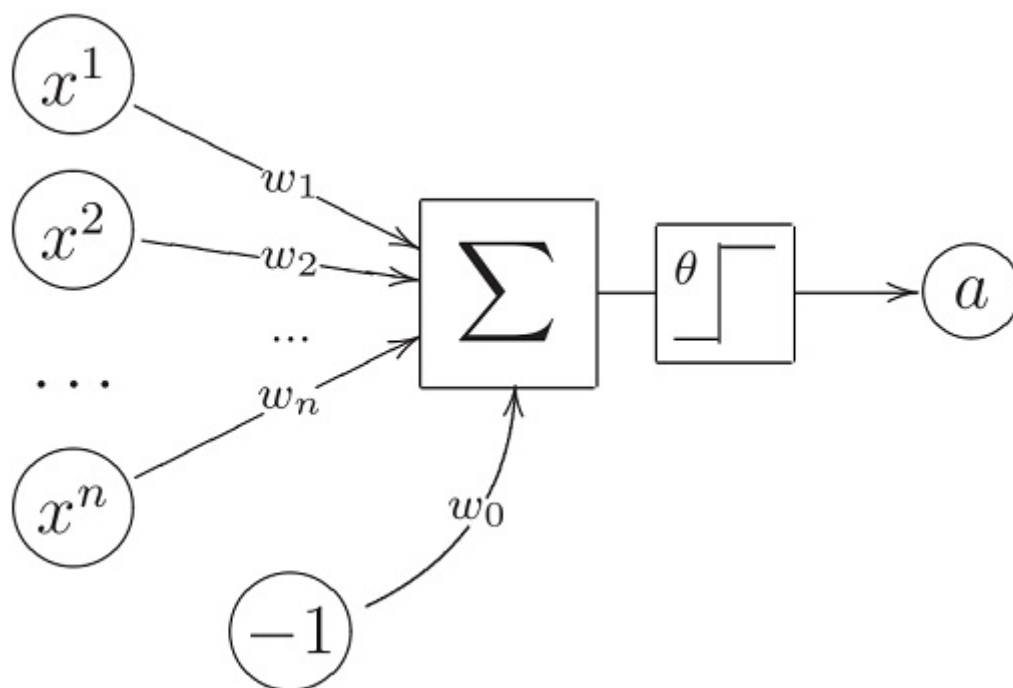


Рисунок 2.1 - Модель нейрона МакКаллока-Питтса

При вирішенні задач ідентифікації та керування динамічними про-процесами нейронна мережа виконує кілька функцій: являє собою нелінійну модель цього процесу, що забезпечує вироблення відповідного керуючого впливу. Мережа також діє як система стеження, яка адаптується до мінливих умов навколишнього середовища. Дуже важливою, особливо при управлінні роботами, є функція класифікації, яка реалізується при виробленні рішення про подальший розвиток процесу.

Основний механізм запам'ятовування, реалізований в природі, можна представити таким чином: імпульс збудження, проходячи через синапс,

"нагрівається" і зменшує його опір, збільшуючи масу синапсу. На наступних етапах, при подальшому пред'явленні еталона імпульсу збудження, шлях порушення долається більш впевнено, вказуючи відповідне зображення з більшою визначеністю, а використовувані в цьому процесі синапси, "розігріваючись", зберігають, а можливо, і збільшують вагу. Тут працює відоме правило Хебба: синаптична вага зв'язку двох порушених нейронів збільшується.

Таким чином, досягається навіть ефект локалізації і максимізації збудження на останньому шарі, що дублює, а можливо, і усуває необхідність взаємодії сусідніх нейронів.

Відзначимо і важливу роль уяви: еталони на вхідному шарі підтримуються достатньо довго, поновлюються або моделюються. Тут велике значення має епіфіз, «третє око» — орган уявлень і медитації, пам'ять і генератор видінь.

Нейромережі мають змогу змінювати власну поведінку, що обумовлюється навколишнім середовищем. Завершальною стадією є аналіз усіх вхідних даних, які самостійно налагоджуються і самонавчаються, з метою гарантувати запланований результат. Після того, як етап навчання нашої мережі повністю завершений, вона стає невідчутливою до незначних відхилень вхідних сигналів, що дає змогу точніше відчувати образ.

Існує велика кількість всіляких конфігурацій мереж з різними принципами функціонування, які орієнтовані на рішення найрізноманітніших завдань. Як приклад, розглянемо багат шарову нейронну мережу прямого поширення (рис 2.2), яка широко використовується для пошуку закономірностей і класифікації образів. Повносвязною нейронною мережею називається структура з багатьма шарами, де будь-який нейрон будь-якого шару зв'язаний з іншими нейронами перехідного шару, а в ситуації з першим шаром — пов'язаний з всіма входами мережі. Пряме поширення сигналу означає, що така нейромережа не містить петель.

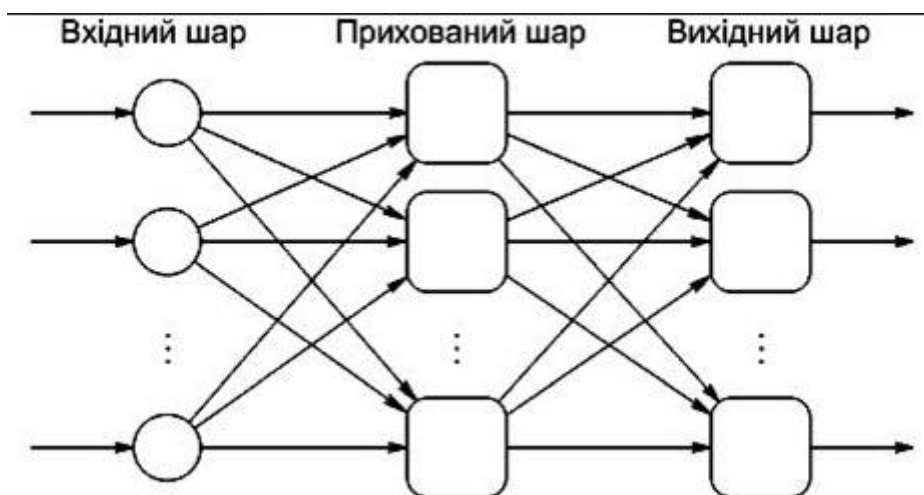


Рисунок 2.2 - Приклад багат шарової повнозв'язної нейронної мережі

Базовою ознакою мозку є здатність до навчання. Коли ми розглядаємо штучну нейромережу, то розглядається рух налагодження конструкції зв'язків між нейронами, та вагами синаптичних зв'язків для досягнення більш дієвого вирішення ситуації. Процес навчання нейромережі проходить за деякою вибіркою, що виконується за чітким алгоритмом.

Серед великої кількості алгоритмів навчання можна виділити метода зворотного поширення помилки, що є найуспішнішим рішенням серед сьогоденних альтернативних методів навчання. Базова ідея зводиться до того, що зміна ваг проходить з урахуванням локального градієнта функції помилки. Різниця між реальними і правильними відповідями нейронної мережі, визначеними на вихідному шарі, поширюється в зворотному напрямку (рис 2.3) — назустріч потоку сигналів. В результаті кожен нейрон здатний визначити внесок ваги кожного окрема нейрона в сумарну помилку мережі. Простіше правило навчання відповідає методу найшвидшого спуску, тобто зміни синаптичних ваг пропорційна їхньому внеску в загальну помилку [9].

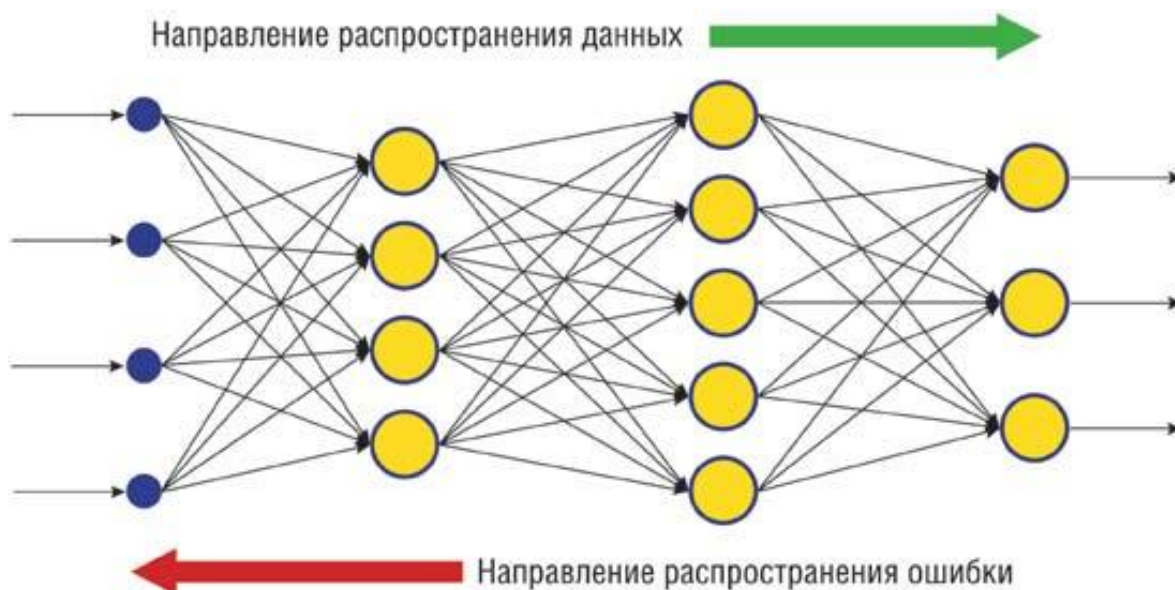


Рисунок 2.3 - Метод зворотного поширення помилки

ШНМ вже використовуються сьогодні в багатьох областях, але, спершу вони можуть бути застосовані там, де мова йде здоров'я людини або у питаннях значних матеріальних потреб, необхідно вирішити серйозні питання, що чіпають стабільність роботи таких мереж. Отже рівень дозволених похибок повинен визначатися виходячи з характеру самої проблематики. Окремі труднощі з аналізом питань надійності виникають через гіпотезу, що комп'ютери повністю безпомилкові, в той час як ШНМ можуть бути сумнівними, хоч вони функціонують за запланованим сценарієм. Ніхто не застрахований від помилки, як людина може помилятися, так і комп'ютер. По-перше, через різні технічні неполадки або похибок в алгоритмі, по-друге, через неувагу, втому або халатність. Тому для завдань з великим пріоритетом необхідно, щоб ці системи дублювали і страхували один одного. З цього випливає, що мережі не повинні розглядатися як додаток, а не єдиний засіб для рішення поставлених задач, і працювати в сукупності з іншими програмними додатками, що могли би запобігати форс-мажорній ситуації, або ситуація передавалась людині для більш детальної аналітики та пошуків вирішення проблематики.

Решта проблем з використанням нейронних мереж залучається в тому, що класичні нейромережі не можуть розтлумачити чому вони вирішують певне

завдання саме таким шляхом. Представлення наслідків навчання нерідко буває непомірно складним, що його неможливо проаналізувати, за винятком приватний ситуації, які здебільшого не є цікавими.

Просто зараз активно здійснюються спроби об'єднати ШНМ, МН та експертні системи. Якщо задіяти повний комплекс, то машинне навчання виконує роботу з правильною вибіркою вхідної інформації, нейронна мережа реагує на значну кількість простих ситуацій, а всі форс-мажорні ситуації, що не змогла обробити система — передаємо на експертизу до експертної систему. Як висновок, більш складні ситуації розглядаються на високому рівні.

Пакети прикладних програм для нейронних мереж, розроблені компаніями, дозволяють користувачам працювати з різними типами нейромереж і з різними способами їх навчання. Вони можуть бути або спеціалізованими (наприклад, для прогнозування ціни акцій), або цілком універсальними.

Нейромережі на фінансових ринках. Американський Citibank використовує нейромережеві передбачення з 1990 року. У 2002 році, за свідченням журналу The Economist, автоматичний ділінг показував прибутковість 25% річних, що набагато перевищує показники більшості брокерів. Chemical Bank використовує нейро-систему фірми Neural Data для попередньої обробки транзакцій на валютних біржах 23 країн, фільтруючи «підозрілі» операції. Fidelity of Boston використовує нейромережі при управлінні портфелями з сумарним об'ємом 3 мільярда доларів. Повністю автоматизовані системи ведення портфелів з використанням нейромереж застосовують, наприклад, Deere & Co - на суму \$100 млн і LBS Capital - на суму \$400 млн. У останньому випадку експертна система об'єднується з приблизно 900 нейромережами. [8]

Таким чином, основний механізм запам'ятовування, реалізований в природі, можна представити таким чином: імпульс збудження, проходячи через синапс, нагрівається і зменшує його опір, збільшує масу синапсу. На наступних етапах, при подальшому пред'явленні еталона імпульсу збудження, шлях порушення долається більш впевнено, вказуючи відповідне зображення з більшою виразністю, а

використовувані в цьому процесі синапси, розігріваючи, зберігають, а можливо, і збільшують вагу.

Отже, досягається навіть ефект локалізації і максимізації збудження на вихідному шарі, що дублює, а можливо і усуває необхідність взаємодії сусідніх нейронів.

2.4 Нейросимулятор

Сьогодні на ринку програмного забезпечення існує безліч різних нейросимуляторів (наприклад, рис 2.4). Розумний нейросимулятор легко пристосувати для вирішення практичних завдань на персональному комп'ютері, якщо вам не потрібно виконувати обчислення якомога швидше. Існують тренажери, які підтримують нейроакселераторні плати з процесорами цифрової обробки сигналів (DSP). Ознакою солідного симулятора є докладний опис програми з введенням з теорії нейронних мереж з технічною підтримкою. Найсучасніші тренажери дозволяють проектувати нейромережі з екзотичними архітекторами і застосовувати розроблені користувачем навчальні алгоритми, а також забезпечують потужне графічне представлення поведінки нейромереж під час навчання і, відповідно до реклами, генерують вихідний код на мові C++. У просунутих нейросимуляторах, як і слід було очікувати, єдиний недолік — захмарна ціна.

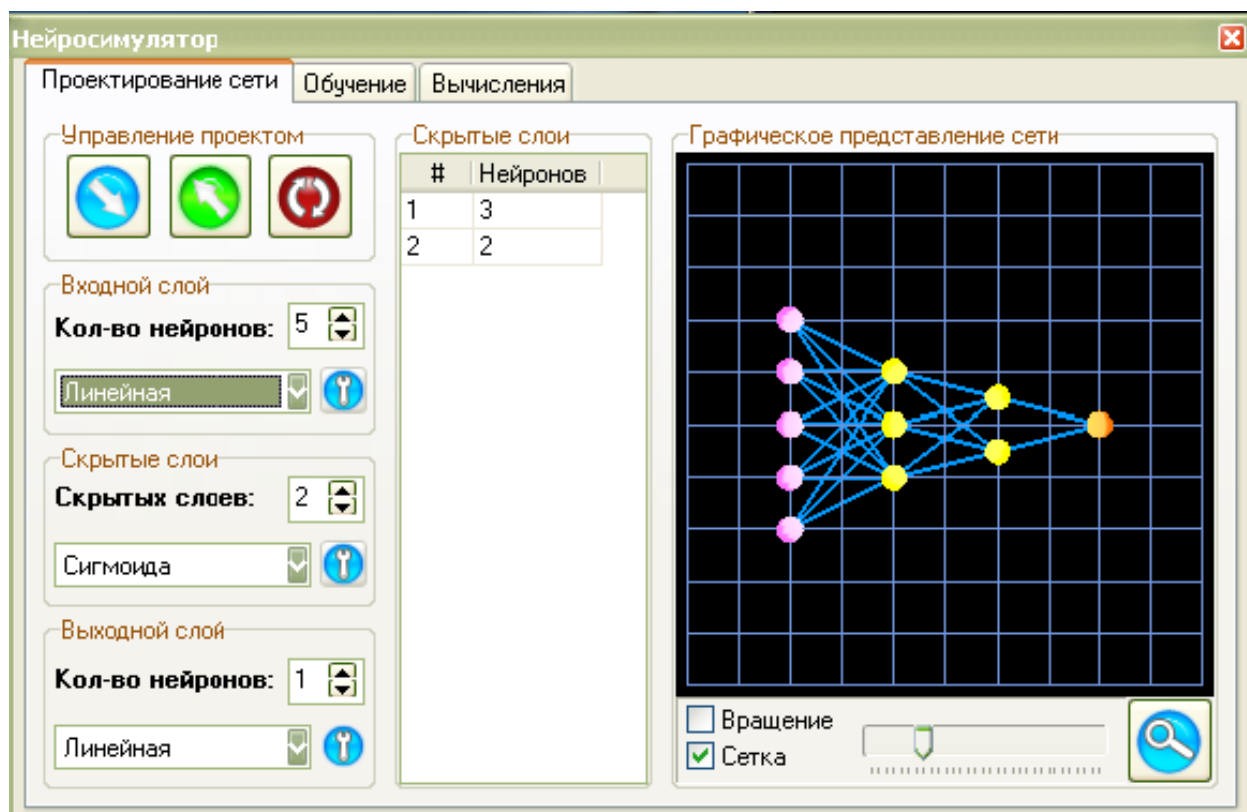


Рисунок 2.4 - Пример нейросимулятора

На вкладці "Проектирование сети" вказується: тип функції активації для кожного мережевого шару, кількість нейронів у вхідному і вихідному мережевих шарах, а також кількість прихованих шарів. На вкладці "Обучение" завантажуються зразок, який використовується для навчання, тут задається кількість навчальних ітерацій і швидкість навчання, а на вкладці "Вычисления" вводяться вхідні параметри для перевірки роботи нейросимулятора. Основний принцип роботи цього нейросимулятора — сигнали, що подаються на входи X_1, \dots, X_n , множаться на коефіцієнти (ваги, синапси), відповідні кожному входу, і визначають рівень збудження нейрона. Вихідний сигнал виходить шляхом пропускання сумарного сигналу збуджених нейронів прихованого шару через нелінійну функцію.

Ще один загін нейромережевого «персонального» програмного забезпечення орієнтований на вузькі гаманці і оптимізований під конкретні завдання. Найбільш поширеними є нейронні предиктори платоспроможності, банкрутства, призначені

для фінансистів. Крім того, існує промислове нейромережеве програмне забезпечення, ексклюзивне за своєю суттю і призначенням, тісно пов'язане зі спеціалізованим апаратним забезпеченням. Останній здійснює попередню обробку що надходять ззовні сигналів, передає отримані дані на вхід нейронної мережі, прискорює функціонування самої нейромережі і виводить готову інформацію в необхідному вигляді. Можливість роботи в режимі реального часу досягається за рахунок паралельних обчислень з використанням нейропроцесорів або шляхом розпаралелювання декількох стандартних DSP. У деяких випадках достатньо одного DSP, так як ці процесори буквально «народжуються» для нейромереж, виконуючи операцію множення з накопиченням суми за один такт — найбільш часто використовуючи операцію в алгоритмі будь нейронної мережі. Додаткові DSP беруть на себе прискорення обчислень на стадії попередньої обробки інформації. Однак на даному етапі нейросимулятори все ж використовуються в більшій мірі в дослідницьких цілях, оскільки здійснення спільної роботи декількох нейросимуляторів різних типів не так просто, як здавалося раніше.

Висновки до 2 розділу

Нейронні мережі здатні знаходити приховані динамічні закономірності в тимчасових економічних рядах, будувати оптимальні стратегії для їх передбачення. Така здатність особливо важлива при вирішенні економічних і фінансових завдань, оскільки внаслідок їх динамічної природи функції приналежності неминуче повинні адаптуватися до змін умов.

Розроблена методика створення і навчання нечітких нейронних мереж дозволяє побудувати якісний прогноз. Нейронні мережі використовувалися для налаштування функції приналежності нечіткої системи прийняття рішень, автоматизували процес подання знань експерта і скоротили час витрат, поліпшивши при цьому параметри системи.

Нечіткі нейронні мережі дають більш довготривалий і якісний прогноз, а, значить, є ефективним інструментом аналізу і передбачення поведінки різних ринків. Можна очікувати, що в недалекому майбутньому реалізовані на основі нечітких нейронних мереж підсистеми аналізу і прогнозу займуть гідне місце в різних сферах застосування.

3 ЗАСОБИ РОЗРОБИ ТА ТЕХНОЛОГІЇ

3.1 Вимоги до розробки бази даних для автоматизованої системи обліку товару

Проектування бази даних – процес створення схеми бази даних і визначення необхідних обмежень цілісності. При проектуванні треба враховувати, що база даних повинна задовольняти комплекс вимог:

- цілісність бази даних (вимога повноти і несуперечності даних);
- багаторазове використання даних;
- швидкий пошук і отримання інформації за запитам користувачів;
- простота оновлення даних;
- зменшення зайвої надмірності даних;
- захист даних від спотворення і знищення.

На цій фазі розробки передбачаються усі необхідні вимоги сукупності користувачів майбутнього додатку. Було обрано реляційну базу даних, так як вона має наступні переваги перед іншими:

- простота, легка в розумінні;
- вибагливі правила проектування;
- інформаційна автономія;
- для організації запитів і написання прикладного програмного забезпечення немає необхідності знати конкретну організацію бази даних у зовнішній пам'яті.

Під завданнями обробки даних зазвичай розуміється спеціальний клас розв'язуваних на ЕОМ завдань, пов'язаних з видом, зберіганням, сортуванням, відбором по заданій умові і групуванням записів однорідної структури [14].

Для програмної реалізації БД були обрані наступні технології: SQL і C#, Microsoft SQL Server 2014 і Microsoft Visual Studio 2015, застосована непроцедурна мова запитів SQL, що орієнтована на операції з даними, представленими у вигляді логічно взаємопов'язаних сукупностей таблиць. Особливість пропозицій цієї мови

полягає в тому, що вони орієнтовані більшою мірою на кінцевий результат обробки даних, ніж на процедуру цієї обробки. SQL сам визначає, де знаходяться дані, які індекси і навіть які найбільш ефективні послідовності операцій слід використовувати для їх отримання. Основні переваги мови SQL, які і стали причиною для вибору мною саме цієї мови, полягають в наступному:

- стандартність – використання мови SQL в програмах стандартизовано міжнародними організаціями;

- незалежність від конкретних СУБД – реляційну базу даних можна перенести з однієї СУБД на іншу;

- таблична структура реляційної БД добре зрозуміла, а тому мова *SQL* проста для вивчення;

- можливість створення інтерактивних запитів – SQL забезпечує користувачам негайний доступ до даних без написання програми;

- можливість програмного доступу до БД – мову SQL легко використовувати в додатках, яким необхідно звертатися до баз даних;

- забезпечення різного представлення даних – дані з різних частин БД можуть бути скомбіновані і представлені у вигляді однієї простої таблиці.

Крім того, мова роботи з базами даних повинна вирішувати всі зазначені вище завдання при мінімальних зусиллях з боку користувача системи [15].

3.2 Використані технології

3.2.1 C# ASP.NET MVC 4.0

Програмний код CRM системи та всього сайту загалом написаний на мові C# на платформі .NET за допомогою IDE Visual Studio 2015.

Використання цих технологій обумовлено легкістю розробки, масштабованості та підтримки. Найголовнішою причиною є постійна підтримка і вдосконалення платформи розробниками. У моєму випадку ці технології

розроблюються компанією Майкрософт.

Таким чином, перевагами цього стеку технологій є:

1. Швидкість розробки - швидкість розробки на *C#* на початкових етапах проекту значно вище в порівнянні з іншими мовами, такими як *C++*, *PHP*, що дозволяє отримати готовий прототип рішення швидше, в коротких мало бюджетних проектах *C#* має значну перевагу.

2. Готові бібліотеки - наявність великої кількості бібліотек, *.NET* та безліч вільно доступних бібліотек, це покриває практично всі першорядні завдання розробки для веб-рішень та *windows*-додатків.

3. Мова і синтаксис – синтаксис постійно розширюється. Код на *C#*, як правило, виглядає простіше і лаконічніше, ніж код на інших мовах.

4. Зручність відладки – *Visual studio 2015* має багато функцій для швидкої відладки та знаходження помилок на ранній стадії.

Синтаксис *C#* близький до *C++* і *Java*. Мова має строгу статичну типізацію, підтримує поліморфізм, перевантаження операторів, вказівники на функції-члени класів, атрибути, події, властивості, винятки, коментарі у форматі *XML*.

Платформа *ASP.NET MVC* являє собою фреймворк для створення сайтів і веб-додатків за допомогою реалізації партерна *MVC*.

Концепція партерна *MVC (model - view - controller)* передбачає поділ додатка на три компоненти:

1. Контролер (*controller*) представляє клас, що забезпечує зв'язок між користувачем і системою, поданням і сховищем даних. Він отримує дані, щ введені користувачем і обробляє їх. І в залежності від результатів обробки відправляє користувачеві певний висновок, наприклад, у вигляді подання.

2. Представлення (*view*) – це власне візуальна частина або призначений для користувача інтерфейс програми. Як правило, *html*-сторінка, яку користувач бачить, зайшовши на сайт.

3. Модель (*model*) представляє клас, що описує логіку використовуваних даних.

Загальна схема взаємодії цих компонентів представлена на рис. 3.1.

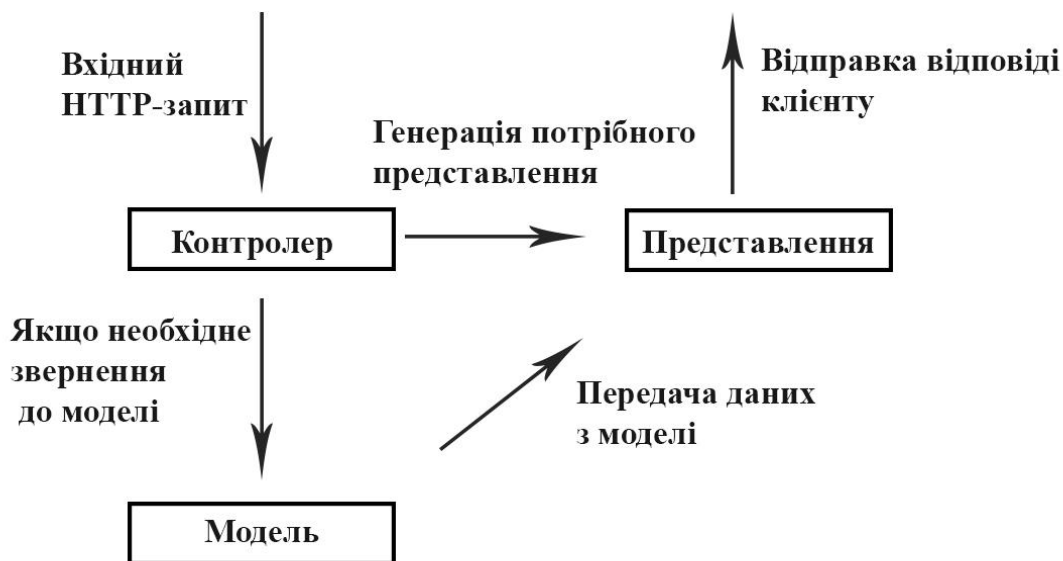


Рисунок 3.1 - Загальне представлення шаблону MVC протоколу

Переваги платформи .NET MVC:

- потужна система маршрутизації (удосконалення технології побудови веб-додатків супроводжувалося розвитком стилю *URL*-адрес);
- можливість тестування окремих компонентів (розподіл різних відповідальностей додатки з незалежних один від одного частин програмного забезпечення, яке підтримується архітектурою MVC, дозволяє спочатку будувати легко супроводжувані і протестовані додатки);
- контроль над HTML і HTTP (інфраструктура ASP.NET MVC генерує ясний і відповідний стандартам код розмітки та працює в тісній співпраці з HTTP. При цьому є контроль над запитом, що передаються між браузером і сервером, що дозволяє дуже точно налаштувати користувацький інтерфейс на свій розсуд);
- можливість розширення (інфраструктура MVC Framework побудована у вигляді набору незалежних компонентів, які задовольняють інтерфейс .NET або створені на основі абстрактного базового класу. Компоненти, подібні системі

маршрутизації, механізми візуалізації і фабрики контролерів, можна легко замінювати іншими компонентами з власної реалізацією) [16].

3.2.2 Веб-сервер IIS

Заключним етапом розробки веб-системи обслуговування клієнтів є його перенесення на веб-сервер.

IIS – набір серверів для декількох служб Інтернету від компанії Майкрософт. IIS поширюється з операційними системами сімейства Windows NT. Основним компонентом IIS є веб-сервер, який дозволяє розміщувати в Інтернеті сайти. IIS підтримує протоколи HTTP, HTTPS, FTP, POP3, SMTP, NNTP. Основним компонентом IIS є веб-сервер – служба WWW (звана також W3SVC), яка надає клієнтам доступ до сайтів за протоколами HTTP і, якщо проведене налаштування, HTTPS [20].

ASP.NET – розроблена Microsoft технологія; для IIS це - основний на сьогоднішній день засіб створення веб-додатків і веб-служб. IIS поставляється разом з операційними системами, в які також спочатку входить .NET Framework, так що підтримка ASP.NET начебто вже вбудована в IIS 6.0 і вище; для більш ранніх версій необхідно окремо завантажити і встановити .NET Framework [17].

3.2.3 MS SQL Server 2014

З огляду на попередні технології і платформи можна побачити, що було використано стек технологій від компанії Microsoft, тому мій вибір програмного забезпечення для розробки бази даних зупинився на Microsoft SQL Server 2014.

Найголовніша причина вибору MS SQL Server – це легкість підключення бази за допомогою LINQ to SQL – API-інтерфейсу для роботи з базами даних SQL Server. Він призначений виключно для SQL Server, однак існують реалізації LINQ від незалежних постачальників, які підтримують найбільш поширені бази даних,

включаючи Oracle, DB2, MySQL, SqlLite і інші.

LINQ to SQL – інструмент ORM початкового рівня, що дозволяє виконувати потужні SQL-запити.

Microsoft SQL Server – система управління реляційними базами даних, розроблена корпорацією Microsoft. Основна використовувана мова запитів – Transact-SQL, створена спільно з Microsoft і Sybase. Transact-SQL – це реалізація стандарту ANSI/ISO по структурованій мові запитів (SQL) з розширеннями. Використовується для роботи з базами даних розміром від персональних до великих баз даних масштабу підприємства; конкурує з іншими СУБД у цьому сегменті ринку [21].

Transact-SQL (T-SQL) – процедурне розширення мови SQL, створене компанією Microsoft (для Microsoft SQL Server) і Sybase (для Sybase ASE) [22].

Мова Transact-SQL є ключем до використання MS SQL Server. Всі додатки, які взаємодіють з екземпляром MS SQL Server, незалежно від їх реалізації і призначеного для користувача інтерфейсу, відправляють серверу інструкції Transact-SQL [18].

Для більш зручної роботи з базою MSSQL необхідно встановити *SQL Server Management Studio* – утиліта для управління і адміністрування всіх компонентів серверу. Утиліта включає скриптовий редактор і графічну програму, яка працює з об'єктами і налаштуваннями сервера як зображено на рис. 3.2.

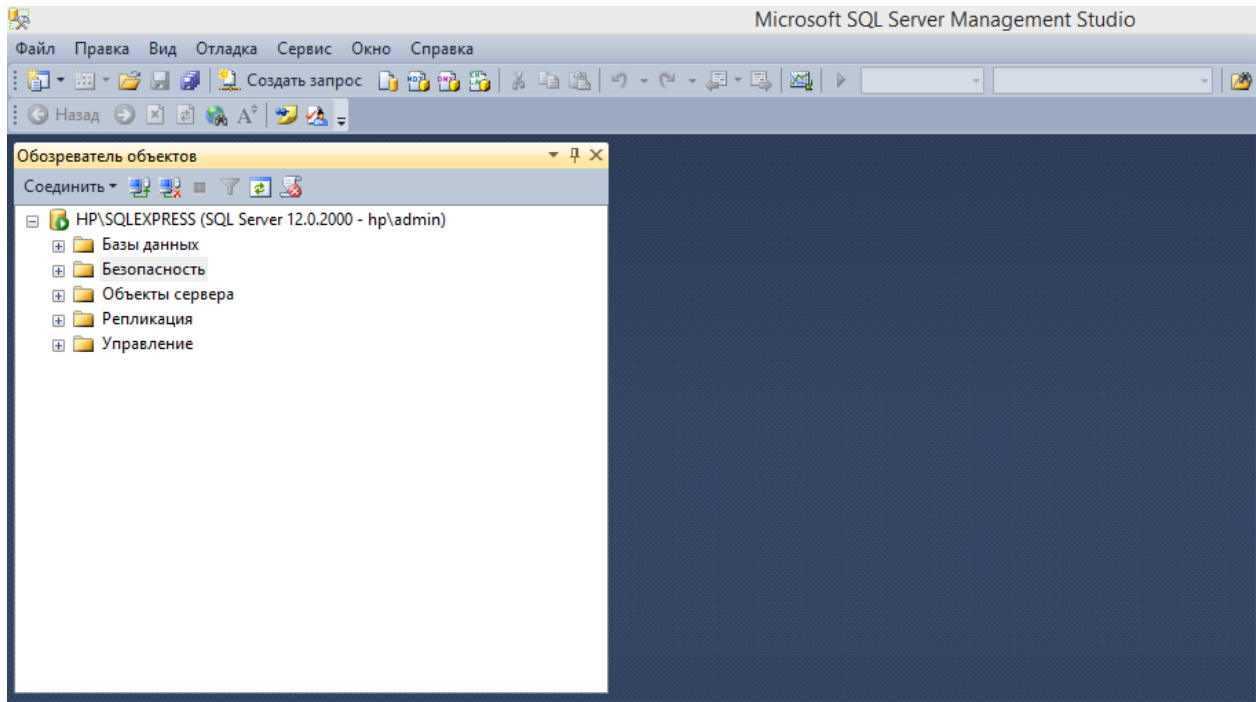


Рисунок 3.2 - *SQL Server Management Studio*

Головним інструментом SQL Server Management Studio є Object Explorer, який дозволяє користувачеві переглядати, отримувати об'єкти сервера, а також повністю ними управляти.

Висновки до 3 розділу

Замислюючись про розробку та впровадження CRM-системи, зазвичай порівнюють такі варіанти:

- використання CRM-системи, яка поширюється за принципами Open Source (програмне забезпечення з відкритим вихідним кодом);
- використання універсальної версії коробочки CRM-системи - комерційної програми з обмеженим функціоналом і набором ліцензій, які дозволяють використовувати її на декількох комп'ютерах або смартфонах;
- розробка CRM-системи під потреби конкретної компанії з урахуванням всіх потреб і всієї специфіки бізнесу організації.

Вибір в більшості випадків роблять в залежності від бюджету і розміру комерційної структури або держпідприємства. Потрібно розуміти, що функціональність і структура CRM-систем коробкового типу орієнтована на усереднені потреби.

Безкоштовні CRM-системи найчастіше з'являються в світ після того, як велика компанія розробила CRM-систему для власних потреб і відкрила свою програму світу. Тому такі рішення актуальні в першу чергу для тих компанії, бізнес-моделі яких нескладні або нагадують бізнес-модель вихідної організації. Цей варіант найкраще підійде для початківців компаній, зацікавлених в першу чергу в обробці лідів (залученні потенційних клієнтів).

При цьому удосконалення використання безкоштовної CRM-системи вимагає залучення програмістів.

Коробкові CRM-системи вважаються бюджетним варіантом (в порівнянні з розробкою CRM), найбільш підходящим для малого і середнього бізнесу, зацікавленого в певній автоматизації бізнес-процесів. Потрібно розуміти тільки, що використання коробкових версій програм часто має на увазі необхідність оплати за підтримку, настройку та удосконалення системи. Платити доведеться і за

збільшення кількості ліцензій, і за вдосконалення функціоналу, і за навчання персоналу.

Як і у випадку з безкоштовними CRM-системами, структура універсальних коробкових версій програм не завжди відповідає потребам конкретної організації. Оплата за версію коробочки може стягуватися у вигляді одноразової безстрокової ліцензії на програму без додаткової підтримки, підписки (тимчасової ліцензії) на певний термін з можливістю різних індивідуальних послуг (добробка, консультація і т.д.), ліцензії на копію програми з можливістю її настройки і зміни і т.д.

Розробка CRM-системи передбачає повну автоматизацію бізнес-процесів під потреби конкретної організації. Цей варіант більш затребуваний середнім і великим бізнесом, котрі вважають, що вигідніше зробити систему під себе і отримати більш високий зріст продажів, ніж обмежуватися гарантованими витратами на версію коробочки з не таким високим ростом.

4 ПРОЕКТУВАННЯ CRM СИСТЕМИ ОБЛІКУ ТОВАРІВ НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ

4.1 Створення технічного завдання

Була поставлена задача: розробити структурну схему системи управління обліком товарів на складі з додатковим модулем штучного інтелекту, яким міг би робити рекомендації для користувачів. CRM система повинна володіти такими особливостями:

1. Побудова і редагування моделі штучної нейронної мережі;
2. Отримання вихідних даних з облікової системи;
3. Заповнення первинних ваг зв'язків нейронів мережі;
4. Можливість запуску навчання нейронної мережі фоновим завданням;
5. робити зв'язку товарів, які краще було б застосовувати разом;
6. Висновок передбачуваних результатів безпосередньо в обробці або збереження в табличний файл.

4.2 Етапи розробки автоматизованої системи управління взаємовідносинами з клієнтами з застосуванням нейромережі

Існує декілька етапів розробки автоматизованої системи управління обліку товарів з використання штучної нейронної мережі. Початковий етап проектування та впровадження системи – проектування майбутнього графічного інтерфейсу нашої CRM системи та створення ТЗ (технічне завдання). Правильно прописане ТЗ дає гарантію на уникнення плутанин між виконавцем і замовником, що в майбутньому, свою чергу, заощаджує нам час на доопрацювання. За умови виконання всіх прописаних вимог до розробки, що зазначені у технічному завданні, розробляємий додаток вважається закінченим. Наступним кроком буде погодження технічного завдання з замовником на розробку програмного додатку,

що буде підписано з обох сторін.

Також дуже важливо врахувати проблеми, які ставляться розробкою додатку. Докладно моделюється «карта» CRM системи для обліку товару, де детально розписується всі розділи, їх можливості та, можливо ще, описуються побажання по відображенню інформації. Особлива увага в технічному завданні присвячується функціональним можливостям системи, що детально прописуються, описується за яким саме принципом потрібно розробити модуль штучного інтелекту для обліку товару на складі. Технічне завдання передбачає також зазначення вимог до серверу, верстання сайту, мови програмування тощо. Також ТЗ повинне містити й терміни виконання робіт. Звичайно ж, ця частина технічного завдання прописується, коли описи всіх розділів, функціональності і вимоги сформовані [10].

Після затвердження макету системи сайту здійснюється їх верстка. На етапі верстки здійснюється створення html структури елементів. Результат, отриманий після закінчення робіт етапу – зверстані html-шаблони всіх сторінок розробляємої системи.

На основі ТЗ і html-макетів сторінок сайту здійснюється програмування функціоналу, розроблення модулю штучного інтелекту, на основі нейронної мережі. Результатом етапу є готовий сайт та CRM система, яка самостійно може прогнозувати попит на товари, зв'язку товарів (який товар з яким доповнюють один одного), створювати рекомендаційний список товарів, які найближчим часом можуть закінчитися, для подальшої аналітики.

Після того, як роботи зі створення веб-сервісу на сервері завершуються, розпочинається тестування.

Заключним етапом розробки сайту є його перенесення на веб-сервер. Для сервера необхідною умовою є задоволення вимогам програмного забезпечення, на якому розроблявся сайт [11].

4.2.1 Розробка програмного коду на мові C# в Visual Studio 2015

Платформа ASP.NET MVC являє собою фреймворк для створення сайтів і веб-додатків за допомогою реалізації патерна MVC. Так як була вибрана мова C#, ми будемо працювати в IDE Visual Studio 2015.

Visual Studio – лінійка продуктів компанії Microsoft, що включають інтегроване середовище розробки програмного забезпечення і ряд інших інструментальних засобів [19].

Було обрано Visual Studio 2015 через ряд переваг над іншими середовищами програмування, а саме:

- підтримка розробки різних типів додатків (консольні, веб-додатки, веб-сервіси);

- редактор коду з підтримкою технології IntelliSense (технологія автодоповнення Microsoft);

- можливістю найпростішого рефакторінга коду;

Структура проекту представлена на рис. 4.1.

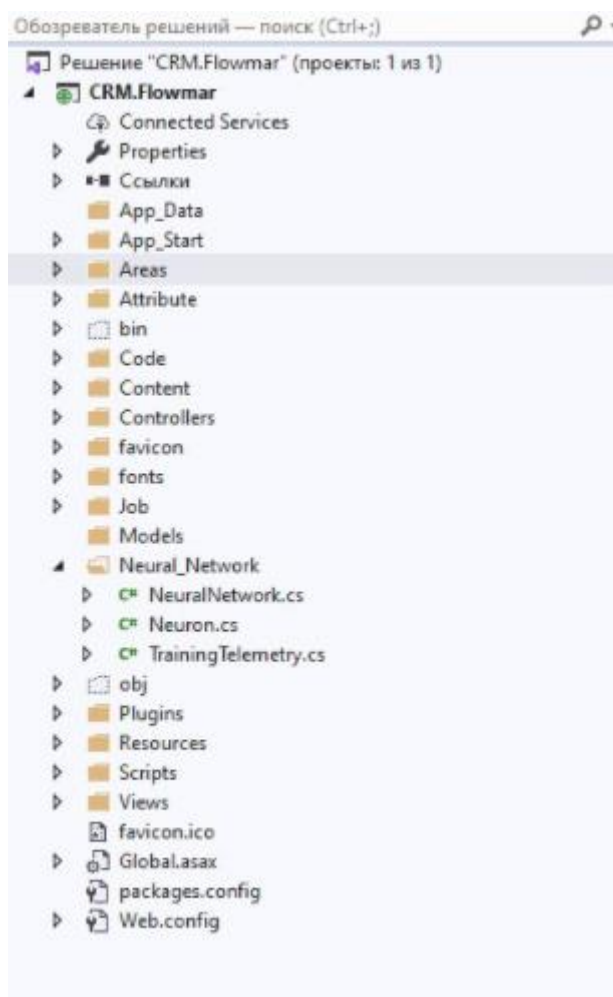


Рисунок 4.1 - Структура проекту в IDE Visual Studio 2015

Розглянемо, для чого потрібні всі ці папки і файли:

- App_Data (містить файли, ресурси і бази даних, що використовуються додатком);
- Ari (містить класи для роботи з веб-сервісом);
- App_Start (зберігає ряд статичних файлів, які містять логіку ініціалізації програми при запуску);
 - CSS (містить допоміжні файли, які не включають код на C# або Javascript, і які розгортаються разом з додатком, наприклад, файли стилів CSS);
 - Controllers (містить файли класів контролерів);
 - Areas (незважаючи на те, що проект MVC за замовчуванням має чітку структуру поділу на окремі функціональні частини – контролери, моделі, уявлення,

іноді для більш зручної роботи над додатком, особливо над великими додатками, додаток ділиться на ряд областей);

- Code (містить файли класів для роботи веб-сайту);
- Images (зберігає картинки);
- Models (містить файли моделей. За замовчуванням Visual Studio додає пару моделей, що описують обліковий запис і службовців для аутентифікації користувача);
- Js (каталог зі скриптами і бібліотеками на мові Javascript);
- Views (тут зберігаються уявлення. Всі вистави групуються по папках, кожна з яких відповідає одному контролеру. Після обробки запиту контролер відправляє одне з цих уявлень клієнта. Також тут є каталог Shared, який містить загальні для всіх уявлення);
- Global.asax (файл, що запускається при старті програми та виконує початкову ініціалізацію. Як правило, тут спрацьовують методи класів, визначених в папці App_Start);
- Web.config (файл конфігурації програми).
- Neural_Network

4.2.2 Детальний опис роботи додатку

4.2.2.1 Проблематика

Прогнозування – стартова точка для планування рішень в управлінні. Зокрема, для прийняття рішення щодо запасів товарів в оптово-роздрібній мережі.

При наявності інформації з показників продажу керівник приймає рішення о модифікуванні своєї CRM системи: розробка додаткових модулів прогнозування товарів, модуля парного зв'язку. Для цього необхідно тимчасовій зупинки поставок. Це дає змогу для керівництва оптимально розподілити фінанси компанії, що витрачаються на закупівлю асортименту товарів і потенційно збільшити

прибуток від кількості популярних товарах, знизити збитки від закупівлі позицій шляхом тимчасово не користуватися попитом покупців.

4.2.2.2 Збір інформації

В додатку, що розробляється, реалізована можливість отримання прогнозу обсягу реалізацій товарів з облікової системи. Далі представлені отримані значення реалізації групи товарів помісячно за 2 роки (рис 3.1).

Розглянувши гістограму можна зробити висновок, що пік продажів припадає на липень та березень, а спад – з листопада по лютий. Необхідно також звернути увагу, що у жовтні 2019 році, на який припадає лівова частка реалізації, вже через рік у той самий місяць було зафіксовано спад попиту.



Рисунок 4.2 - Приклад нейросимулятора

4.2.2.3 Обрання методу прогнозування

Беручи до уваги, розглянуту раніше гістограму (рис 3.1), і враховуючі зауваження зроблені в попередньому пункті виникає висновок, що для дослідження групи товарів неможливо побудувати математичну модель дослідження об'єкта. Таким чином, вибір мною методу прогнозування був зроблений на користь апарату нейронних мереж, оскільки вони здатні до прогнозування безпосередньо впливаючи з здатності до узагальнення та виділення прихованих залежностей між вхідними та вихідними даними. Після закінчення процесу навчання нейромережа буде здатна зробити припущення значень будь-якої структурності виходячи з кількох тодішніх результатів.

4.2.2.4 Проблематика навчання мережі

Для того, щоб нейронна мережа вміла правильно розпізнавати, необхідно спочатку навчити її. Навчання мережі проходить таким чином (рис. 3.2). Вагові коефіцієнти мережі повинні змінюватися до тих пір, поки помилка розпізнавання не стане мінімальною. Слід сказати, що не можна досягти 100% точності розпізнавання, завжди будуть присутні деякі похибки. Процес навчання - процес досить витратний і вимагає великого часу.

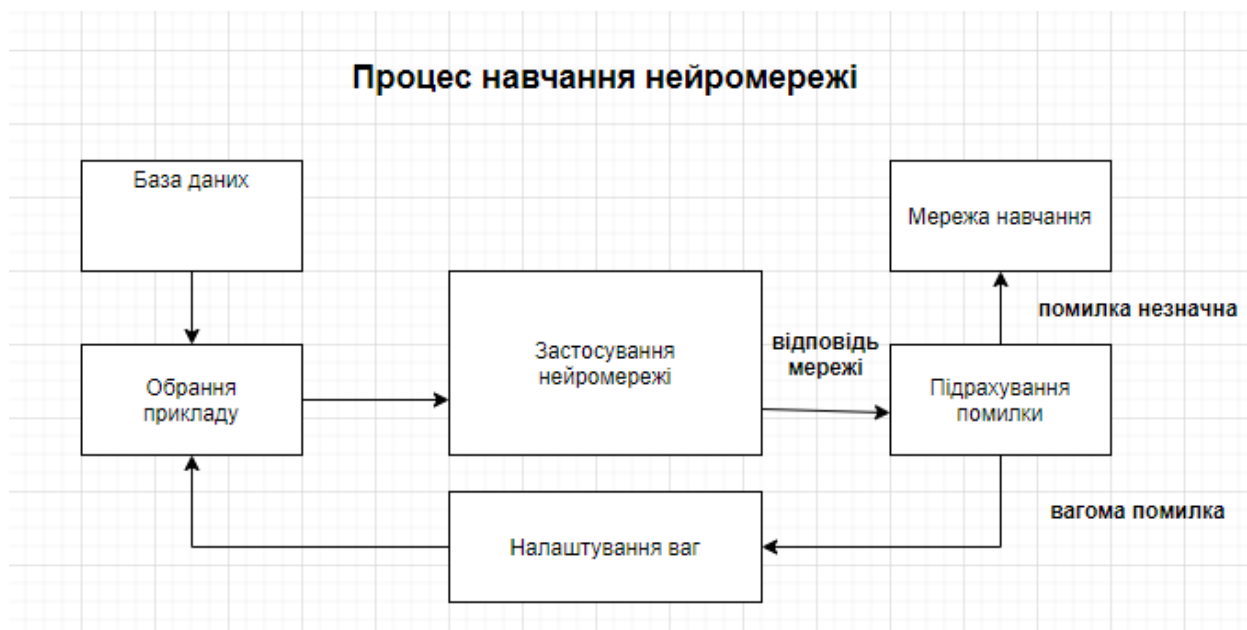


Рисунок 4.3 - Схема процесу навчання неймережі

4.2.2.5 Моделювання додатку

У загальному вигляді програма повинна працювати наступним чином: на вхід отримуємо параметри з облікової системи, передані в якості аргументів данні випадковим шляхом розділяємо, перед початком навчання, на три частини (тренування, тестування, верифікація) у співвідношенні 7:2:1. Це розбиття з початком нового циклу буде змінюватися. Перед початком навчання усі входи зважуються і від середньозважених входів отримуємо значення функції активації на виході. Нейрон повністю описується своїми вагами і функцією активації F . Отримавши набір чисел в якості входів, нейрон видає деяке число на виході. Функція активації, в нашому випадку – логістичний сигмоїд з постійним коефіцієнтом кривизни 2.

Перший шар – це вхідні данні. Кожен нейрон першого шару видає своє значення без перетворень на всі нейрони другого шару. Другий шар вираховує середню, серед тих сигналів, що прийшли до нього, вираховує від неї величину сигмоїд і передає отримане значення до третього шару і т.д. поки в останньому шарі

не збереться на виході набір результатів. Тобто, нейрони першого шару – це фактори, а нейрони останнього – результати, всі внутрішні шари – предмет нашої «фантазії».

Суть процесу навчання полягає в тому, що ми подаємо N-ну кількість відомих наборів факторів на вхід мережі та дивимось, що отримуємо на виході. Залежно від того наскільки розрахункове значення останнього шару (результату) відхилялося від відомого для навчання проводимо корекцію ваг сигналів у кожного нейрона. Метод корекції – це зворотне поширення помилки, тобто, відхилення між фактичним і розрахунковим результатом. І від так, від набору даних до набору, від етапу до етапу мережа налаштовує ваги таким чином, щоб помилка була все меншою. Наприкінці навчання додаємо набір змінних на вхід для якого поки що не відомий результат і обчислюємо його через налагоджену мережу.

На вхід подаємо набір параметрів (P) і відомий результат (R). Наприклад 100 рядків:

$$1). P_1 P_2 \dots P_n = R_1$$

....

$$100) P_1 P_2 \dots P_n = R_{100}$$

На цьому етапі мережа вчиться правильно отримувати результат на основі параметрів, додаємо ще 100 рядків:

$$101) P_1 P_2 \dots P_n = R_{101} \text{ результат мережі із 101 відхилення, із 101-R}_{101}$$

....

$$200) P_1 P_2 \dots P_n = R_{200}$$

Мережа отримані результати порівнює з реальними, звідси і отримуємо значення похибки навчання нейромережі.

У наступному масиві ми використаємо ще 10 рядків:

$$201) P_1 P_2 \dots P_n = ? \text{ результат мережі C}_{201}$$

....

$$210) P_1 P_2 \dots P_n = ?$$

Ми маємо тільки масив параметрів, але результатів не знаємо, а мережа з

виявленою похибкою надає нам свої результати розрахунків для кожного набору (рядки) параметрів.

Після запуску програми в першу чергу необхідно побудувати модель нейронної мережі. Передбачено візуальний інтерфейс для полегшення роботи користувача і наочності (рис.3.2).

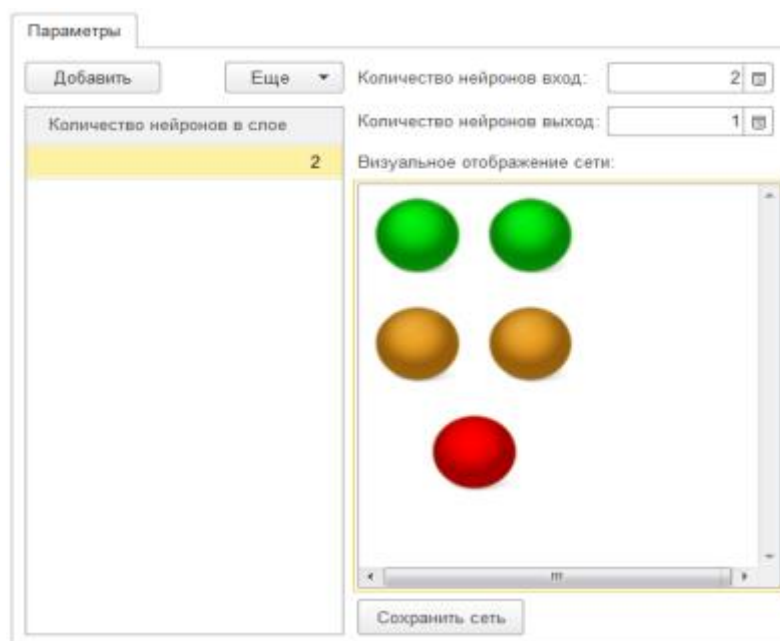


Рисунок 4.4 - Интерфейс форми роботи з нейромережою

Кількість вхідних і вихідних нейронів повинна відповідати кількості типів вхідних даних і результатів відповідно.

Внутрішні шари – це синапси (зв'язку) для кожного з яких мережа підбирає і запам'ятовує вага. Чим більше шарів і нейронів тим більше зв'язків і ваг, і тим більше область пошуку рішення. Для простих завдань досить мереж з одним прихованим шаром, а якщо змінних багато і зв'язків між ними далекі, то гнучкість, і адаптивність рішення підвищується за рахунок збільшення кількості внутрішніх нейронів. Але при цьому збільшується і час навчання. Після збереження моделі мережі необхідно отримати дані про продажі і залишки. В оптових і роздрібних мережах нерідко використовують схему резервування, тому кількість товару на складі може бути менше нуля. Для отримання цих даних так само передбачена

Выберите период

16.06.2016 [календарь] × – 16.06.2016 [календарь] × [Очистить период](#)

Вчера
Сегодня
Завтра

[Показать произвольный период](#) **Выбрать** Отмена

День
Неделя
Декада
Месяц
Квартал
Полугодие
Год
Прочее

Рисунок 4.5 - Обрання періоду часу

Після натискання кнопки «отримати дані» вихідні значення і результати програмно поміщаються в таблицю значень для подальшої роботи з ними (рис. 3.3).

Навчання нейронної мережі відбувається за алгоритмом зворотного поширення помилки.

$$w_{p-q}(i+1) = w_{p-q}(i) + \eta \delta_q OUT_p$$

де:

i – номер поточної ітерації навчання;

w_{p-q} – величина синоптичної ваги, що з'єднує нейрон p з нейроном q ;

η – коефіцієнт «швидкості навчання», управляє величиною зміни ваг;

δ_q – помилка q нейрона;

OUT_p – вихід нейрона p .

На головній формі вказуємо η – «норму швидкості навчання», помилки

апроксимації для навчання і граничне число ітерацій. При дуже маленьких значеннях навчання нейронної мережі буде проходити повільно. При дуже великих значеннях η виникає ймовірність того, що в момент досягнення мінімуму функції помилки $E = f(W)$ нейронна мережа не зможе потрапити в цей мінімум і буде нескінченно довго «стрибати» праворуч і ліворуч від нього, виробляючи перерахунки вагових коефіцієнтів. На страті приймається значення, вказане в параметрах, але по ходу навчання швидкість навчання може змінюватися. Принцип модифікації швидкості полягає в тому, що в разі коли між кроками навчання напрямок градієнта для ребра мережі не змінюється, швидкість навчання множиться на коефіцієнт $K > 1$, а в разі зміни знака множник K стає < 1 . (рис.3.5).

Параметры сети:

Норма скорости обучения:

Критерии завершения обучения

Порог ошибки теста, %:

Порог ошибки тренировки, %:

Предельное число итераций:

Рисунок 4.6 - Форма параметрів навчання мережі

Для навчання нейромережі передбачено 2 режими:

- Тренувальна;
- Версифікаційна.

Тестувальни режим – не бере участь прямо в навчанні, але її помилка видна в процесі. Судячи за значенням тренувальної помилки, а точніше бачити її динаміку в порівнянні з помилкою тренування можна судити про це не перевчити чи мережу (помилки розходяться) і взагалі наскільки адекватно даними побудована модель мережі.

Класичний приклад перенавчання мережі – це коли помилки тренування і тесту спочатку знижуються, а потім розходяться, і помилка тесту нескінченно

зростає. Як правило, це трапляється коли вихідна вибірка занадто мала або занадто однорідна, і / або кількість внутрішніх шарів, і нейронів при цьому занадто велике. Для перегляду статистики навчання необхідно натиснути на кнопку «Перегляд статистики», де буде представлений графік залежності помилок від кількості ітерацій) (рис.4.7).

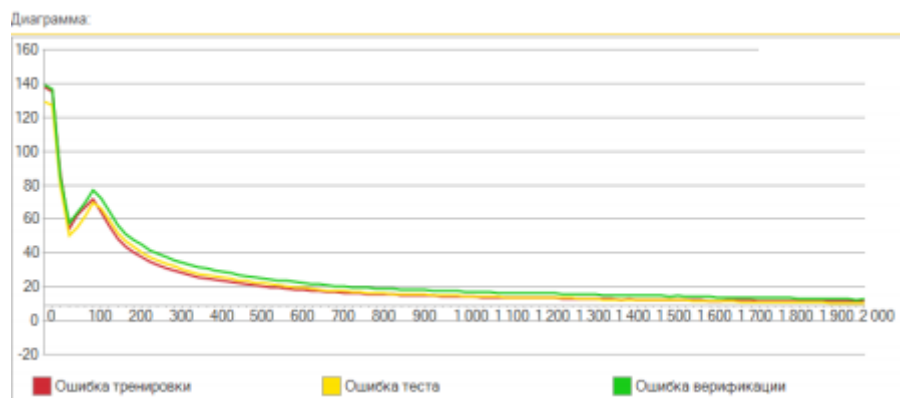


Рисунок 4.7 - Статистика навчання мережі

Для отримання результатів прогнозу з урахуванням зазначеної помилки необхідно натиснути кнопку «розрахунок за даними мережі» (рис. 4.8).

Экспресс расчет

Еще ▾

Переменная	Значение
X0	4
X1	2 518

Рассчитать Изменить аргументы

Рисунок 4.8 - Форма прогнозування значень
Для збереження навченої мережі призначена кнопка «зберегти мережу».

Завантажені в масив внутрішні і зовнішні шари нейронів, ваги для них і

активаційна функція зберігаються, або в текстовий файл, або в довідник «Сховище значень» типових конфігурацій.

3.3.6 Аналіз показників прогнозу

Після навчання нейронної мережі отримане значення прогнозованої величини на минулий місяць, тобто квітень 2021 видається рівному 9571. (рис.4.9).

Экспресс расчет

Еще ▾

Переменная	Значение
X0	4
X1	9 571

Рисунок 4.9 - Прогнозоване значення

Роблячи висновок, що частка реалізованої продукції в кілограмах, у порівнянні з усіма попередніми періодами (рис.4.9) є рекордним, і керівнику при плануванні закупівель даної групи товарів має сенс задуматися над збільшенням кількості поставки досліджуваної продукції.

Однак, сформувавши звіт в обліковій системі, і порівнявши дане значення з фактичною кількістю видно, що (рис.4.10) фактичний обсяг реалізацій за заданий період дорівнює 129661, що значно відрізняється від прогнозованого значення.

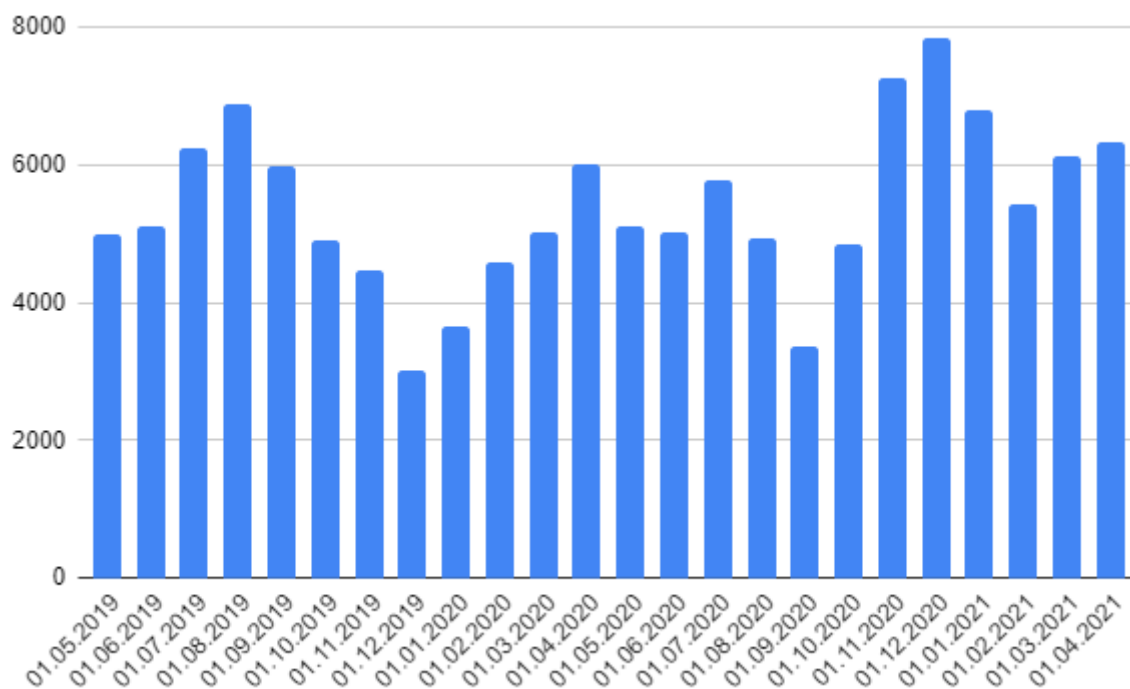


Рисунок 4.10 - Фактична кількість відвантаженого товару

З цього випливає висновок, що спроектована нейронна мережа видає результат, що перевищує задане значення помилки 10%. Це може бути пов'язано з тим, що не зовсім коректно була спроектована нейронна мережа (склад вхідних нейронів і внутрішні шари) або у вихідних даних виявилось недостатньо для створення навчальний вибірки.

Висновки до 4 розділу

У заключенні можна сказати, що впровадження нейронних мереж в реалізацію CRM системи обліку товарів на складі представляється перспективним напрямком. Використання подібних систем в економіці не має обмежень. Вже зараз існують системи підтримки прийняття рішень на базі непромережених технологій застосовуються найбільшими зарубіжними компаніями з метою прогнозування попиту на конкретну послугу або товар в певному часовому відрізку.

В даний час штучні нейронні мережі є важливим розширенням поняття обробки даних, що дали змогу вирішити питання з деякою кількістю нелегких проблем, також є запит на розробку нових ПЗ і апаратів, які могли би виконувати складну роботу замість людини. Сучасні нейрокомп'ютери використовуються, зазвичай, в програмних продуктах і тому зрідка задіють свій потенціал «на повну». Епоха справжніх паралельних нейровчислень почнеться з появою на ринку великої кількості апаратних реалізацій – спеціалізованих нейрочипів і плат розширень.

ВИСНОВКИ

Впровадження CRM-системи є одним з пріоритетних напрямків розвитку будь-якої компанії, оскільки зможе забезпечити підвищення якості обслуговування клієнтів, зменшити трудовитрати на супровід і звільнити співробітників від рутинної роботи. Система автоматизує процес спілкування з клієнтом і обробки його звернень, що позитивно позначається на реалізації клієнтоорієнтованої стратегії компанії.

Світовий ринок CRM-систем дуже різноманітний і майорить великою кількістю компаній, що пропонують свої рішення. Серед них є фірми, які давно зарекомендували себе на ринку, так і маловідомі, а також багато відомих брендів, які тільки відносно недавно представили своє рішення в цьому сегменті програмних рішень.

Поставлені перед нами завдання кваліфікаційної роботи було виконано. Так, впровадження нейронних мереж в реалізацію систем підтримки прийняття рішень представляється перспективним напрямком. Використання подібних систем в економіці не має обмежень. Вже зараз існують системи підтримки прийняття рішень на базі нейромережових технологій, що застосовуються найбільшими зарубіжними компаніями з метою скорочення ризиків у плануванні своєї фінансової діяльності.

Таким чином, підбиваючи підсумок по роботі, можу резюмувати, що в першій частині роботи нами узагальнено теоретичні знання про нейромережі, вивчено принципи їх роботи. У другій частині був проведений аналіз методів навчання, були виявлені основні етапи рішення завдань за допомогою нейромереж і нейросимулятора. У третій частині на практиці були застосовані теоретичні знання про нейромережі на прикладі вже працюючого підприємства та впровадження нових можливостей до CRM системи.

Нейронні мережі досить полегшили процес прийняття управлінського

рішення. Проведена робота показує, що використання штучних нейронних мереж для прийняття рішень ефективно використовувати особливо тоді, коли слід зробити вибір, враховуючи велику кількість вхідних параметрів.

На сьогоднішній день штучні нейромережі є основним перспективним розширенням поняття обчислення. Мережі вже вирішили низку нескладних задач і людство працює над цією темою з метою розроблення та вдосконалення існуючого матеріалу, що, в свою чергу, будуть спроможні на складні завдання, які на даний момент може виконувати тільки людина. Нинішні нейронні комп'ютери застосовуються, здебільшого, в програмних продуктах. Епоха істинних паралельних нейромереж, розрахунки яких починаються з приходом на ринок значної кількості апаратних рішень: спеціалізованих нейрочипів, спеціальних розширень до плат основним призначенням яких було б обробка звуків, відео, графічних ілюстрацій та інших видів даних.

СПИСОК ЛІТЕРАТУРИ

1. http://eprints.library.odeku.edu.ua/577/1/PerelyginBV_TkachTB_Shtuchni_neyronni_merezshi_v_zadachakh_obrobky_danykh_Lab_rob_1_MV_2019.pdf
2. <https://olacom.ru/uk/mozilla-firefox/iskusstvennyi-intellekt-na-osnove-neironnyh-setei/>
3. <https://www.ling.upenn.edu/courses/cogs501/Rosenblatt1958.pdf>
4. http://eprints.library.odeku.edu.ua/6727/1/Goncharenko_Rozr_sys_id_oblycha_2019.pdf
5. <https://krs.chmnu.edu.ua/jspui/bitstream/123456789/1691/1/%D0%90%D0%B2%D1%82%D0%BE%D1%80%D0%B5%D1%84%D0%B5%D1%80%D0%B0%D1%82%20601%20%D0%A1%D1%96%D1%87%D0%B5%D0%B2%D1%81%D1%8C%D0%BA%D0%B8%D0%B9%20%D0%A1%D1%82%D0%B0%D0%BD%D1%96%D1%81%D0%BB%D0%B0%D0%B2%20%D0%92%D1%96%D0%BA%D1%82%D0%BE%D1%80%D0%BE%D0%B2%D0%B8%D1%87.pdf>
6. <http://um.co.ua/8/8-4/8-4561.html>
7. <https://ukrbukva.net/page,2,92419-Ispol-zovanie-neironnyh-seteiy-v-sisteme-Matlab.html>
8. https://moodle.znu.edu.ua/pluginfile.php?file=/481692/mod_resource/content/2/%d%9b%d%b5%d%ba%d1%86%d1%96%d1%8f%203.pdf
9. Нейронні мережі: навчання без вчителя [електронний ресурс]: http://www.codenet.ru/progr/alg/ai/htm/gl3_4.php
10. Етапи веб проектування та їх суть [Електронний ресурс]. – Режим доступу: <https://sites.google.com/site/ka2508addsite/> – Назва з екрану.
11. Вебсайт. Створення веб сайтів [Електронний ресурс]. – Режим доступу: <https://websait.uz.ua/website-development/etapy-rozrobky-sajtu/> – Назва з екрану.
12. <http://specials.kunsht.com.ua/machinelearning2>
13. http://elartu.tntu.edu.ua/bitstream/lib/34121/1/Mag_nayk_2020_CTm_Bass_R_V.pdf

14. Етапи веб проектування та їх суть [Електронний ресурс]. – Режим доступу: <https://sites.google.com/site/ka2508addsite/> – Назва з екрану.

15. Дрінь Б.М. Основи SQL: Навч.посібн. / Б.М.Дрінь, О.В.Козич – Івано-Франківськ, Прикарпатський національний університет, ВДВ ЦІТ «Плай», 2012. – 79 с.

16. Магдануров, Г.И. ASP.NET MVC Framework / Г.И. Магдануров, В.А. Юнев. – СПб.: БХВ-Петербург, 2010. – 320 с.: ил.

17. Станек У. Internet Information Services (IIS) 7.0. Справочник администратора: Пер.с англ. / У. Станек. – М.: Издательство «Русская Редакция»; СПб.: «БХВ-Петербург», 2009. – 528 с.

18. Фленов М.Е. Transact-SQL /М.Е.Фленов. – СПб.: БХВ-Петербург, 2006 – 576 с.

19. <https://thepresentation.ru/informatika/oglyad-seredovishcha-visual-studio-ta-sposobi#:~:text=Microsoft%20Visual%20Studio%20%E2%80%93%20%D0%BB%D1%96%D0%BD%D1%96%D0%B9%D0%BA%D0%B0%20%D0%BF%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82%D1%96%D0%B2,%D1%96%20%D1%80%D1%8F%D0%B4%20%D1%96%D0%BD%D1%88%D0%B8%D1%85%20%D1%96%D0%BD%D1%81%D1%82%D1%80%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D0%BB%D1%8C%D0%BD%D0%B8%D1%85%20%D0%B7%D0%B0%D1%81%D0%BE%D0%B1%D1%96%D0%B2.&text=%D0%9F%D1%80%D0%BE%D0%B5%D0%BA%D1%82%20%2D%20%D1%86%D0%B5%20%D0%B2%D1%96%D0%B4%D0%BF%D1%80%D0%B0%D0%B2%D0%BD%D0%B8%D0%B9%20%D0%BF%D1%83%D0%BD%D0%BA%D1%82%20%D0%B4%D0%BB%D1%8F,%D1%81%D0%BB%D1%83%D0%B6%D0%B1%20%D0%B2%20Visual%20Studio%202010.>

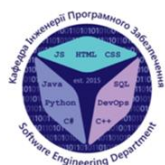
20. https://znaimo.com.ua/Internet_Information_Services

21. <http://lib.mdpu.org.ua/e-book/vstup/L5.htm>

22. <https://learn.ztu.edu.ua/mod/resource/view.php?id=17833>

ДОДАТКИ

Презентація до диплому



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
Навчально-науковий інститут інформаційних технологій
Кафедра Інженерії програмного забезпечення

ПРЕЗЕНТАЦІЯ ДО МАГІСТЕРСЬКОЇ РОБОТИ
на тему:
"РОЗРОБКА CRM СИСТЕМИ ОБЛІКУ ТОВАРІВ НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ"

підготував студент 7 курсу
групи ППЗМ-71

спеціальності 121 Інженерія програмного забезпечення

Погребняк М.М.

Наручний керівник:

доцент Трінтіна Н. А.

1 слайд

МЕТА ТА ЗАВДАННЯ МАГІСТЕРСЬКОЇ РОБОТИ

Мета дипломної роботи – застосування нейронних мереж у CRM системах обліку товарів.

Об'єкт дослідження – процес ділового взаємодії з клієнтами на основі штучних нейронних мереж і система рекомендацій товарів.

Предмет дослідження – програмне забезпечення для торговельної компанії, призначене для автоматизації взаємодії з клієнтами.

2

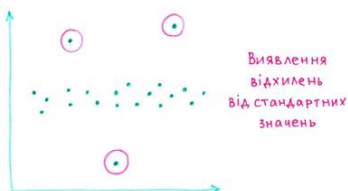
2 слайд

АНАЛОГИ

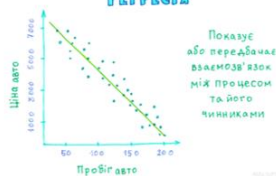


МАШИННЕ НАВЧАННЯ ТА ЇЇ ЗАДАЧІ

Виявлення аномалій



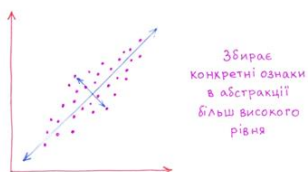
РЕГРЕСІЯ



КЛАСИФІКАЦІЯ



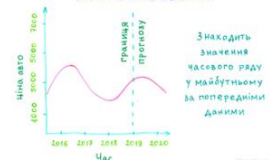
ЗМЕНШЕННЯ РОЗМІРНОСТІ



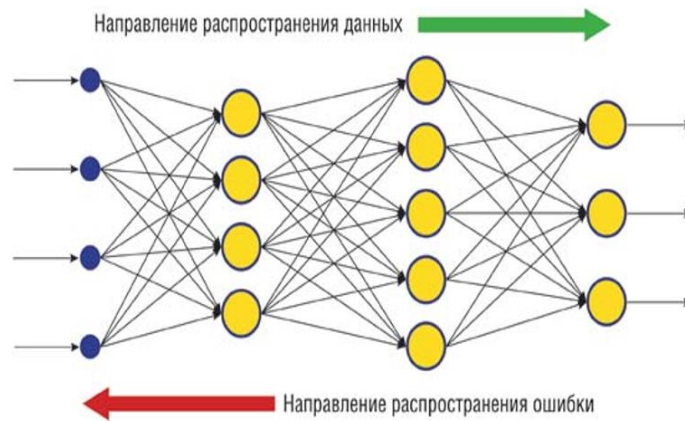
КЛАСТЕРИЗАЦІЯ



ПРОГНОЗУВАННЯ



МОДЕЛЬ МЕХАНІЗМУ ЗАПАМ'ЯТОВУВАННЯ



5

5 слайд

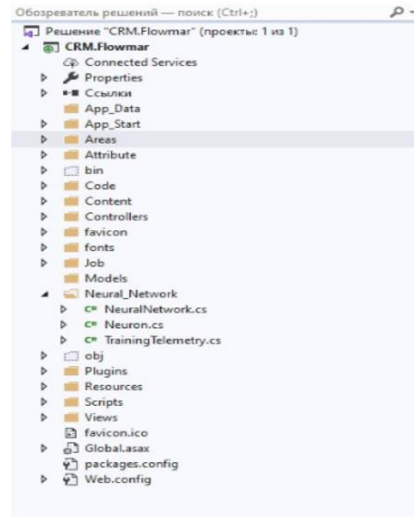
ЕТАПИ РОЗРОБКИ

1. Провести аналіз існуючих аналогів
2. Виділити ключові аспекти майбутньої системи
3. Написати ТЗ
4. Затвердити макет
5. Розробка клієнтського графічного інтерфейсу
6. Програмування функціоналу
7. Тестування
8. Перенесення додатку на веб-сервер

6

6 слайд

ДЕРЕВО ПРОЕКТУ ТА ТІХНОЛОГІЧНІ РІШЕННЯ



7

7 слайд

ПРОЦЕС ПІДГОТОВКИ СИСТЕМИ ДО НАВЧАННЯ

Сформировать данные о продажах и остатках: НейроннаяСеть

Настройки

Тема:

Период:

Отчет:

Выберите период

16.06.2016 - 16.06.2016 [Очистить период](#)

Вчера	День
Сегодня	Неделя
Завтра	Декада
	Месяц
	Квартал
	Полугодие
	Год
	Прочие

[Показать произвольный период](#) [Выбрать](#) [Отмена](#)

Параметры

Добавить Еще Количество нейронов ввода:

Количество нейронов в слое: Количество нейронов выхода:

Визуализация оптимизации сети

[Сохранить сеть](#)

Параметры сети:

Норма скорости обучения:

Критерии завершения обучения

Порог ошибки теста, %:

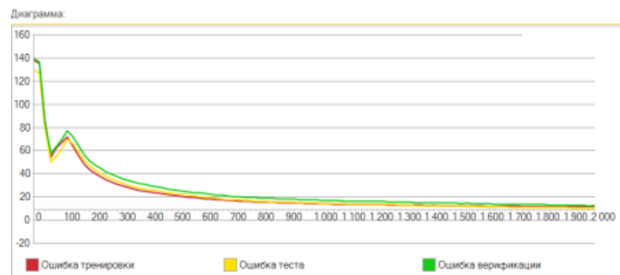
Порог ошибки тренировки, %:

Предельное число итераций:

8

8 слайд

НАВЧАННЯ НЕЙРОННОЇ МЕРЕЖІ



РЕЗУЛЬТАТИ ТРЕНУВАНЬ

ДО



ПІСЛЯ



ВИСНОВКИ

Проведена робота показує, що використання штучних нейронних мереж для прийняття рішень ефективно використовувати особливо тоді, коли слід зробити вибір, враховуючи велику кількість вхідних параметрів.

- Було розроблено CRM систему обліку товарів на складі з модулем нейроної мережі, який проходить процес зворотного навчання з мінімальною допустимою помилкою;
- Проаналізовано, що дана система може збільшити кількість проданого товару на ~10%;

ДЯКУЮ ЗА УВАГУ!