МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний університет радіоелектроніки
Факультет Комп'ютерних наук
Кафедра Програмної інженерії

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

другий (магістерський)
(рівень вищої освіти)

Дослідження методів прогнозування результатів спортивних матчів. Нейронні мережи. Повнозв'язна нейронна мережа.

Виконав:
студент 2 курсу групи ІПЗм-20-1
Пилипець Д.О.
(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного забезпечення

Тип програми Освітньо-наукова

Керівник доц. Чуприна А. С.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ З.В.Дудар

2022 р.

Харківський національний університет радіоелектроніки

Факультет Комп’ютерних наук
(повна назва)

Кафедра Програмної інженерії
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 121 – Інженерія програмного забезпечення
(код і повна назва спеціальності)

Тип програми освітньо-наукова програма
(освітньо-професійна або освітньо-наукова)

Освітня програма Інженерія програмного забезпечення
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____20 ____ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студента *Пилипця Даниїла Олеговича*
(прізвище, ім’я, по батькові)

1. Тема роботи «Дослідження методів прогнозування результатів спортивних матчів. Нейронні мережи. Повнозв'язна нейронна мережа.»

затверджена наказом університету від «24» березня 2022 р. № 412 Ст

2. Термін подання роботи до екзаменаційної комісії «__» _____ 2022 р.

3. Вихідні дані до роботи *нейронні мережі, метрики, аналіз датасету, критерії їх оцінки, логістична регресія, k-найближчих сусідів, повнозв'язна нейронна мережа, Python, NumPy, SciKit Learn.*

4. Перелік питань, що потрібно опрацювати в роботі *вступ, аналіз предметної області, аналіз нейронних мереж для прогнозування результатів, критерії їх оцінки, постановка задачі, проведення експериментів та аналіз їх результатів, формування подальших рекомендації.*

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз предметної області	25.02.2022	виконано
2	Постановка задачі	10.03.2022	виконано
3	Проведення дослідження	01.04.2022	виконано
4	Підготовка пояснювальної записки	03.05.2022	виконано
5	Підготовка презентації та доповіді	06.05.2022	виконано
6	Попередній захист	10.05.2022	виконано
7	Перевірка на академічний плагіат	10.05.2022	виконано
8	Нормоконтроль	12.05.2022	виконано
9	Рецензування	13.05.2022	виконано
10	Знесення диплома в електронний архів	15.05.2022	виконано
11	Допуск до захисту у зав. кафедри	15.05.2022	виконано

Дата видачі завдання 17.01.2022 р.

Студент _____
(підпис)

Керівник роботи_____ доцент Чуприна А.С.
(підпис)

РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота магістра містить: 63 стор., 12 рис., 1 табл., 1 лістинг, 20 джерел, 3 додаток.

МАТЧ, СПОРТ, СПОРТСМЕНИ, ТУРНІР, ДЕРЕВА РІШЕНЬ, ВИПАДКОВІ ЛІСИ, ГРАДІЄНТНИЙ БУСТИНГ, МАШИННЕ НАВЧАННЯ, НАУКА ПРО ДАНІ.

Об’єктом дослідження є методи прогнозування на основі дерев рішень.

Метою дослідження є аналіз різних за складністю та структурою класифікаційних методів прогнозування для виявлення найбільш доцільного для передбачення результатів спортивних матчів.

Методи розробки базуються на основі архітектурних підходів науки про дані, мові програмування Python та її бібліотек.

У результаті роботи було проведено ретельний аналіз існуючих методів прогнозування та виявлено найбільш влучний для поставленої задачі передбачення результату спортивного матчу, результати були продемонстровані на твореному прототипі.

MATCH, SPORTS, ATHLETES, TOURNAMENT, DECISION TREES, RANDOM FORESTS, GRADIENT BUSTING, MACHINE LEARNING, DATA SCIENCE.

The object of research is forecasting methods based on decision trees.

The aim of the study is to analyze different in complexity and structure of classification methods of forecasting to identify the most appropriate for predicting the results of sports matches.

Development methods are based on architectural approaches to data science, the Python programming language and its libraries.

As a result, a thorough analysis of existing forecasting methods was conducted and the most accurate for the task of predicting the outcome of a sports match was found, the results were demonstrated on the created prototype.

Я, Пилипець Даниїл Олегович, студент групи ІПЗм-20-1, здобувач вищої освіти на другому (магістерському) рівні, кафедра Програмної інженерії, заявляю: моя кваліфікаційна робота на тему «Дослідження методів прогнозування результатів спортивних матчів. Нейронні мережи. Повнозв'язна нейронна мережа.», що буде представлена до ЕК для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу ElArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ML	Machine learning, машинне навчання
DNN	Dense neural network, повнозв'язна нейронна мережа
API	Application Programming Interface, прикладний програмний інтерфейс
HTTP	HyperText Transfer Protocol, протокол передачі даних
PCA	Principal component analysis, метод головних компонент

ВСТУП

Ми всі очікуємо здорового життя, тому що здорове життя є запорукою будь-якого щастя. Тільки завдяки фізичному та психічному благополуччю ми можемо прагнути до здорового життя. Одним з найефективніших способів досягнення фізичного та психічного благополуччя є, безсумнівно, спорт. Ось чому спорт важливий у нашому житті. Спорт не тільки гарантує здорове життя, але й насправді є універсальною необхідністю в нашому житті. Спорт може зробити наше життя сповненим задоволення та успіху.

Завдання роботи − надання аналітичної та статистичної інформації щодо спортивних матчів. Аналіз історії матчів конкретного гравця, яка може бути використана для покращення результатів та виявлення слабких сторін.

Метою роботи є інформаційна система за допомогою якої можна прогнозувати результати спортивних матчів, аналізувати стан та рівень гравця..

Актуальність системи полягає в тому, що у наші дні системний підхід ширше використовується в організаційному аналізі, оскільки цей підхід може врахувати більше змінних і взаємозв’язків, розглядаючи організаційну проблему в рамках більшої системи. Іншим важливим припущенням системного підходу є безперервна взаємна взаємодія між системою та її середовищем. Системний підхід також забезпечує корисну структуру для розуміння того, як елементи організації взаємодіють між собою зі своїм середовищем.

Останнє десятиліття принесло величезний прогрес у захоплюючому вимірі штучного інтелекту − машинному навчанні. Ця техніка для введення даних і перетворення їх у передбачення значно покращує системи.

Компанії використовують машинне навчання, щоб розпізнавати закономірності, а потім робити прогнози щодо того, що сподобається клієнтам, покращить роботу чи допоможе покращити продукт. Однак перш ніж ви зможете побудувати стратегію на основі таких прогнозів, ви повинні зрозуміти вхідні дані, необхідні для процесу прогнозування, проблеми, пов’язані з отриманням цих вхідних даних, і роль зворотного зв’язку, що дозволяє алгоритму робити кращі

прогнози з часом. Тому первинний аналіз вхідних даних та розуміння значення кожної з ознак об'єктів є важливою частиною завдання прогнозування. Таким чином проаналізуємо предметну область і виділимо значущі ознаки.

Безсумнівно, що професійний спорт став бізнесом. Майкл Джордан вже двадцять років виступає за «Чикаго Буллз» і є одним із найвідоміших і найвідоміших спортсменів у світі. За рік він заробив 37 мільйонів доларів, що більше, ніж внутрішній дохід багатьох країн. Ринок спорту у світі став набагато ширшим, ніж раніше, завдяки існуванню великих телевізійних програм і спонсорів. Тому прогнозування результатів матчів привернуло багатьох уболівальників.

Існують різні елементи даних, які впливатимуть на результат матчу. Деякі елементи даних, які безпосередньо впливають на результат, включаючи цілі гравця, вид спорту, середовище тощо. Деякі команди використовують експертів-людей для аналізу та прогнозування результатів спорту. Використання автоматизованих методів і алгоритмів аналізу даних для прогнозування результатів може дати більш точні результати. Інтелектуальний аналіз даних має на меті допомогти тренерам і гравцям передбачити результати, а також оцінити результативність гравців, передбачити травми гравців та оцінити стратегію гри. Знання, отримані зі спортивних даних, можна використовувати для опису ситуацій або прогнозування ситуацій.

Тому багато спортивних тренерів і менеджерів використовують аналіз даних для розвитку спорту. Результати показують, що вони досягли значного прогресу в цій галузі. Пілотна програма була проведена в 2002 році італійським футбольним клубом Мілан, збираючи дані з тренувань, щоб передбачити травми гравців за певний період часу. Це було зроблено з метою прогнозування травм гравців, а також запобігання травмам, що в кінцевому підсумку призвело до великої фінансової вигоди від запобігання травмам гравців на користь клубу. Деталі програми тестування були такими: керівник медичної команди футбольної команди «Мілан» попросив 18 гравців підключити датчики, щоб передбачити можливість травми. Ці датчики були підключені до комп’ютера і зберігали та обробляли інформацію про гравців.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Загальний аналіз галузі

Системи машинного навчання все частіше використовуються в різних сферах і стають все більш спроможними вирішувати різні завдання, починаючи від повсякденного помічника і закінчуючи прийняттям рішень у галузях з високими ставками. Ось деякі приклади: у повсякденному житті людини системи машинного навчання можуть розпізнавати об’єкти в зображеннях, вони можуть транскрибувати мовлення в текст, вони можуть перекладати між мовами, вони можуть розпізнавати емоції в зображенні обличь чи мови, системи машинного навчання дозволяють безпілотникам літати автономно. У медицині системи машинного навчання можуть відкривати нові способи використання існуючих ліків, виявляти поганий стан за зображеннями, дають можливість точної медицини та персоналізованої медицини. У сільському господарстві системи машинного навчання може виявляти хвороби сільськогосподарських культур і розпилювати пестициди на сільськогосподарські культури з точністю, а також може допомогти зберегти наші лісові екосистеми, а в наукових дослідженнях машинне навчання може об’єднати різнорідну бібліографічну інформацію, щоб ефективно витягати знання.

Однак через природу моделей машинного навчання, які мають чорний ящик, розгортання алгоритмів машинного навчання, особливо в сферах високого рівня ставок, таких як медична діагностика, кримінальне правосуддя, прийняття фінансових рішень та інші регульовані критичні сфери безпеки, вимагає перевірки та тестування на правдоподібність експертами в області не лише для безпеки, але й з правових причин. Користувачі також хочуть зрозуміти причини прийняття конкретних рішень на моделях. Такі вимоги призводять до високих суспільних та етичних вимог пояснення таких систем. Пояснення стають незамінними для інтерпретації результатів чорної скриньки та надання користувачам можливості отримати уявлення про прийняття рішень системою. Процес, який є ключовим компонентом у зміцненні довіри та довіри до систем машинного навчання.

Однак більшість роботи в області дослідження пояснення машинного навчання зосереджено на створенні нових методів і прийомів для покращення пояснення в задачах прогнозування та спробі мінімізувати зниження точності передбачення. Отже, існує величезна потреба в оцінках для кількісної оцінки якості пояснювальних методів, а також підходів до оцінки та вибору найбільш відповідного пояснення на практиці. Як результат, існуючі дослідження почали спроби сформулювати деякі підходи до оцінки пояснюваності. Однак немає узгоджених показників для якість методів пояснення, а порівняння складні.

Однією з причин невирішеної проблеми є те, що зрозумілість за своєю суттю є дуже суб'єктивною концепцією, а сприйнята якість пояснення є контекстною і залежить від користувачів, самого пояснення, а також від типу інформації, яка цікавить користувачів. Наприклад, ви можете особисто знати основні причини, чому банк відхилив іпотеку, але за юридичним сценарієм може знадобитися повне пояснення зі списком усіх факторів.

Індустрія професійного спорту − це багатомільярдний бізнес. Кожні вихідні тисячі вболівальників збираються на стадіони, щоб подивитися гру улюбленої команди, будь то футбольний матч в англійській Прем'єр-лізі чи баскетбольний матч у Національній баскетбольній асоціації. Мільйони інших людей також налаштовуються на ігри вдома, що робить ринок професійного спорту дуже прибутковим як для мовників, так і для рекламодавців. У промисловості, що все більш комерціалізована інформація стає все більш затребуваною та професійні гравці та тренери готові користуватися системами, які можуть надати інформаційну перевагу перед іншими.

Розвиток технологій дозволив легко збирати докладні дані, що призвело до активного використання аналітики даних та технологій машинного навчання у сфері професійного спорту. Це допомагає спортивним компаніям проводити симуляції матчів, які їм ще належить зіграти. Аналітика даних — невід'ємна частина будь-якої успішної великої спортивної команди.

Аналіз даних допомагає спортивним організаціям оцінювати результати своїх спортсменів та оцінювати набір, необхідний для поліпшення результатів команди. Він також оцінює сильні та слабкі сторони суперника, що дозволяє

тренерам приймати правильні рішення щодо тактики. Використання даних допомагає збільшити дохід, знизити експлуатаційні витрати та гарантувати високу віддачу від інвестицій.

Аналітики допомагають командам отримувати цінну інформацію з даних та застосовувати їх для підвищення продуктивності команди. Аналітики використовують дані окремих гравців для вимірювання їхньої продуктивності та прийняття обґрунтованих рішень про те, як використовувати їх у своїй команді. Вони також надають інформацію, яка допомагає у наборі спортсменів.

Стратегія є невід'ємною частиною будь-якого виду спорту, чи то індивідуального чи командного виду спорту. Професійні спортсмени та команди залежать від цих стратегій, щоб конкурувати та перемагати своїх суперників. Сучасний коучинг використовує набори великих даних для створення виграшних стратегій, які допомагають окремим спортсменам та команді загалом.

Травми у професійному спорті – частина життя кожного спортсмена[1]. Вони роблять великий внесок у його стан і рівень гри. Тому цьому аспекту буде приділено окрему увагу під час розгляду завдання прогнозування. Спортивні травми зазвичай спричиняються надмірним навантаженням, прямим ударом або застосуванням сили, більшої, ніж частина тіла може структурно витримати. Існує два види спортивних травм: гострі та хронічні. Травма, яка виникає раптово, наприклад розтягнення щиколотки, спричинена незручним приземленням, відома як гостра травма.

Хронічні травми викликані повторним перевантаженням груп м’язів або суглобів. Погана техніка та структурні аномалії також можуть сприяти розвитку хронічних травм. Медичне розслідування будь-якої спортивної травми має важливе значення, тому що ви можете отримати більш серйозні травми, ніж ви думаєте. Наприклад, те, що здається розтягненням щиколотки, насправді може бути переломом кістки.

Лікування залежить від типу та тяжкості травми. Завжди звертайтеся до лікаря, якщо біль не проходить через пару днів. Те, що, на вашу думку, є простим розтягненням, насправді може бути переломом кістки. Фізіотерапія може допомогти відновити пошкоджене місце і, залежно від травми, може включати

вправи для підвищення сили та гнучкості. Повернення до спорту після травми залежить від оцінки вашого лікаря або фізіотерапевта.

Багатьом спортсменам і командам найбільш потрібна інформація про те, коли умови можуть підвищити ризик травмування. Для команд травми можуть мати фінансовий вплив на отримання доходу через медичні витрати, відновлення, спонсорство та продаж квитків через програш змагань на ранніх етапах. Так само для гравців інформація про запобігання травмам може допомогти продовжити їхню кар'єру, заробіток і підвищити їхню цінність до найвищого потенціалу. Більш ефективне прогнозування травм потребує заходів, які допомагають збалансувати навантаження та напругу з належною кількістю часу відновлення, живлення та сну.

Моделі логістичної регресії можуть допомогти визначити, як гравці реагують на конкретний тренувальний стимул, та визначити ймовірність потенційної травми. Тренувальне навантаження може бути скориговано відповідним чином, щоб уникнути ризику травми[2].

Спроба грати до того, як травма буде належним чином залікована, лише призведе до подальшої шкоди та затримки відновлення. Найбільшим фактором ризику травми м’яких тканин є попередня травма. Поки травма заживає, ви можете підтримувати свою фізичну форму, якщо це можливо, вибираючи такі види вправ, які не задіють цю частину вашого тіла.

1.2 Виявлення проблем

На даний момент інформація стала невід'ємною частиною спорту. Трекінг різних показників спортсмена під час тренувань та матчів стає звичайною справою. Ці статистичні дані використовуються для покращення процесу тренування, аналізу сильних та слабких сторін кожного гравця. Збір інформації під час дружніх та професійних матчів та її постобробка дозволяється тренеру виділяти та покращувати показники гравців.

Традиційно аналіз спортивних результатів визначається як завдання аналізу

спостережень, що починаючи від збору даних і закінчуючи наданням зворотного зв’язку, і має на меті покращити спортивні результати, залучаючи всіх тренерів, гравців та самих аналітиків. Спостереження за продуктивністю здійснюється або в прямому ефірі під час спортивної події, або після змагань за допомогою відеозйомки та зібраної статистики.

Більшість методів аналізу, що використовуються у спорті вищих досягнень, походять від методів дослідження, що використовуються в академічному світі, де використовується широкий спектр інструментів аналізу для систематичного вивчення техніки, рухів, навичок, прийняття рішень.

Проблема цього аналізу у цьому, що аналіз за своєю природою деструктивний. Аналіз розбиває дії, прийоми, навички тощо на складові частини чи вимірні події. Він покликаний визначити, що пішло не так зі спортсменом чи командою і які проблеми, недоліки та помилки призвели до поганої роботи.

Аналітики продуктивності тепер можуть бути помічені на стадіонах, чи то в тренерській ложі, чи в окремому гарному оглядовому місці на трибунах, відзначаючи події та дії з матчу за допомогою спеціалізованого програмного забезпечення. У цьому процесі вони розробляють статистичні звіти, які можна надсилати в режимі реального часу на пристрої, якими користуються тренери, і відображати їм підсумок ключових показників ефективності, а також короткі відео канали з ключовими моментами. Однак додатковий час, доступний для аналізу після матчу, дозволяє більш детально оцінити результати, використовуючи додаткові додаткові джерела даних. Дані, які використовуються під час післяматчевого аналізу, можуть надходити з джерел, які не залежать від спостережень аналітика, таких як якісні дані, відеоряди і навіть вимірювання навантаження спортсменів, частоти серцевих скорочень, рівня лактату в крові, прискорення, швидкості та місцезнаходження, зібраних за допомогою носових пристроїв[3].

Так як зараз захоплення спортом розвивається з великою швидкістю, як професіонали так і аматори бажають відстежувати свій прогрес та фізичний стан. А також багато спортсменів бажають мати можливість спостерігати розвиток та досягнення.

Дані в спортивній організації зазвичай складаються зі зведень гравців і команд, статистики виступів, відеокліпів і т. д., але тепер дані надходять з багатьох джерел, кількість яких за останні кілька років значно збільшилася. Вибух спортивної науки зробив відстеження здоров'я та харчування гравців набагато складнішим. Тренери та медичний персонал тепер ведуть власні набори даних, які стали важливим активом для оцінки гравців.

Способів трекінгу є безліч. Найрозповсюдженими є трекінг серцевого ритму, рухи (кроки), або трекінг за відео. Я вважаю, що трекінг за відео потоком є найбільш цікавим серед цих методів. Проаналізувавши поточний ринок, я виділив для себе декілька цікавих програмних продуктів трекінгу спортсменів.

Першим з них є iSports Analysis. Цей продукт надає аналіз спортивних результатів. а саме можливість створення уявлення результатів команди, суперників та гравців. Підхід до аналізу, орієнтований на гравця, використовує останні досягнення в аналізі відео (див рис. 1.1).

Рисунок 1.1 Трекінг відео ігри у баскетбол за допомогою iSports Analysis.

В системі надається гравцям можливість переглянути та зрозуміти власну гру. Здатність утримувати інформацію значно зростає, коли учень більше залучений у процес. Швидкість і ступінь навчання значно збільшуються, тому система шукає та застосовує нові інноваційні способи підвищення рівня залучення

гравців. А також за допомогою цього продукту можна створювати звіти та ділитися з тренерами та членами команди безпечно в Інтернеті. Наступною функцією є система аналізу тренерів. Вона дозволяє клубам і тренерам аналізувати тренерську поведінку. Також можна отримати аналітику залученості гравця.

Сервіс дозволяє аналізувати баскетбольні матчі (див рис. 1.2), щоб покращити результати команд та гравців та виділяти важливі моменти, ключові дії під час гри та надавати тренерам, командам та гравцям відгуки про ситуації [4].

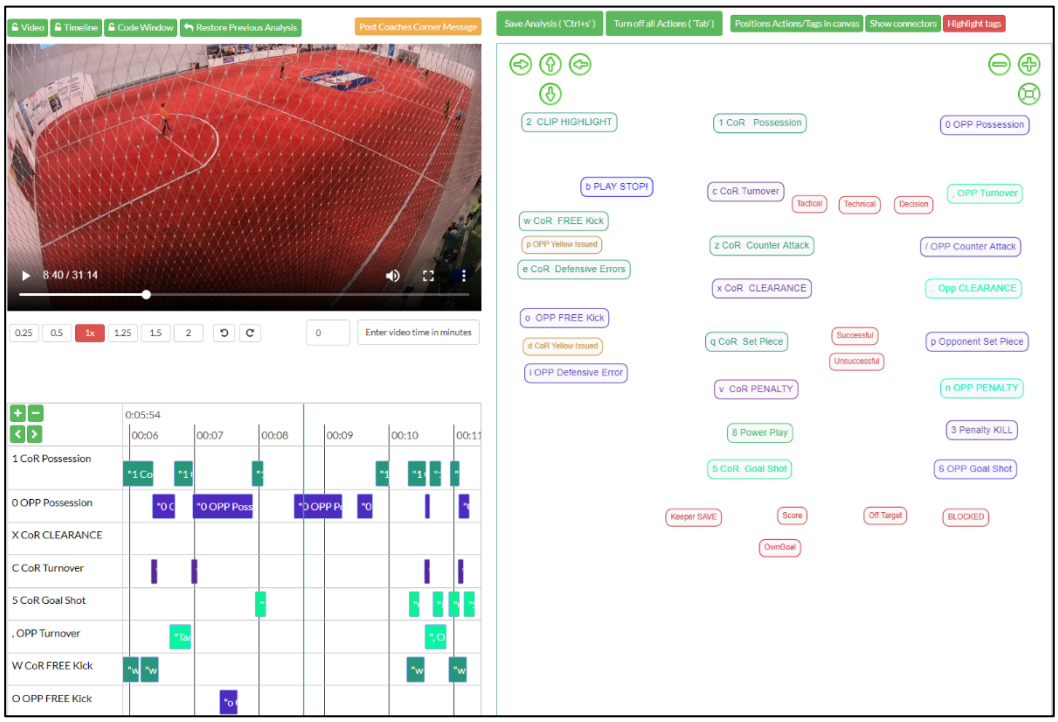

Рисунок 1.2 Аналіз гри у баскетбол за допомогою iSports Analysis.

Підхід до аналізу, орієнтований на гравця, використовує найновіші досягнення в аналізі відео та хмарних технологій, щоб надати гравцям можливість переглядати та розуміти власну продуктивність ефективнішим способом, ніж будь-коли раніше – прискорюючи навчання та максимізуючи розвиток. Онлайн-платформа iSportsAnalysis є першим у світі повністю онлайн-тренерським освітнім ресурсом. Організаціям більше не потрібно створювати власні дорого вартісні онлайн-платформи рішень [5]. Система дозволяє персоналу планувати, контролювати та адаптувати інтенсивність конкретних тренувань/сеансів, бо розроблена одним із провідних у світі спортивних вчених/фітнес-тренерів.

Розширювати можливості команд, покращувати їх тактичні знання та допомагати гравцям приймати більш виважені рішення.

Його конкурентом та аналогом є сервіс Once Video Analyser. Він робить позначення дій та гравців простим та інтуїтивно зрозумілим. Це робить аналіз швидшим і ефективнішим. Він допомагає тренерам зробити їх роботу легшою, швидшою та ефективнішою. Завдяки міжнародному досвіду роботи з телевізійними станціями, ЗМІ, національними командами, великими клубами та меншими регіональними командами. Приклад роботи сервісу Once Video Analyser показаний на рисунку 1.3 [6].

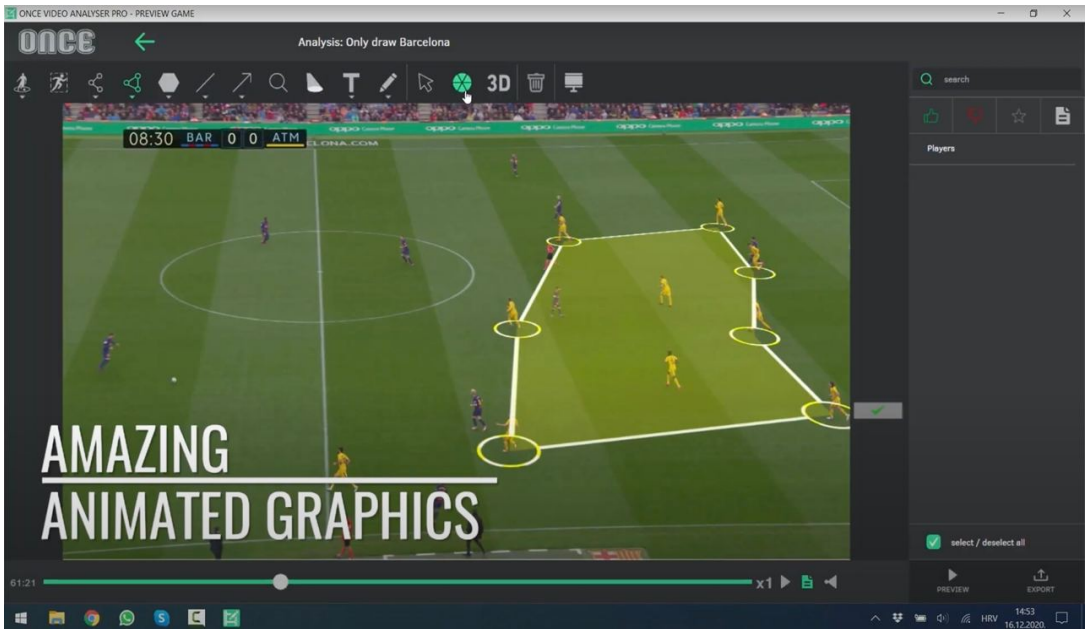

Рисунок 1.3 Аналіз гри у футбол за допомогою Once Video Analyser.

Серед переваг цього сервісу слід відзначити великий інструментарій візуального аналізу та розмітки ігрового процесу [7].

Наведені вище програмні продукти є цілком влучними для заданих цілей, але і вони мають недоліки:

– вони платні,
– аналізують лише за відео,
– не мають прогнозування майбутньої гри,
– не враховують травмування гравців.

Саме тому нам слід поставити задачу, з урахуванням усіх цих недоліків.

2 ПОСТАНОВКА ЗАДАЧІ

Після аналізу предметної області та виявлення основних проблем у ній можна назвати основне завдання поточного проекту. Необхідно розробити та реалізувати інформаційну систему збору та аналізу даних з спортивних матчів.

Завдання прогнозування результату матчу зводитиметься до класифікації об'єкта. У ролі об'єкта виступає матч, а ознаки характеризують гравців, які в ньому беруть участь. Ознаковий опис гравця є формалізованою історією його кар'єри[8].

Процес прогнозування з використанням машинного навчання має великий потенціал, а спортивних іграх існує безліч правил, тому спортивні ігри можна як багатофакторну модель прогнозування з часовими рядами. У порівнянні з традиційними алгоритмами прогнозування алгоритми машинного навчання мають певні організаційні та адаптивні можливості.

Прогнозування, насправді, дисципліна статистики. Процес прогнозування включає збирання даних, обробку даних та прогнозування даних, що виражаються за допомогою математичних моделей. Завдяки прогнозу даних можна ефективно аналізувати суб'єктивну тенденцію розвитку у майбутньому та будувати математичну модель, відповідну розвитку майбутніх подій. Модель спортивного прогнозування також є своєрідною моделлю прогнозування. Традиційний метод спортивного прогнозування полягає у прогнозуванні результатів спортивних змагань за допомогою математичної статистики, яка переважно використовується у професійних спортивних тренуваннях.

Для реалізації мети на основі проведеного аналізу предметної області можна чітко сформулювати етапи виконання роботи:

- провести детальний аналіз області, побудова моделі даних;
- провести всебічний аналіз існуючих алгоритмів прогнозування;
- розробити і порівняти алгоритми прогнозування результатів матчів та підсумкової турнірної таблиці.

Спочатку потрібно провести попередній аналіз ознак, які присутні в даній предметній області і виділити серед них ті, що роблять максимальний внесок у

результати матчів. Для цього буде використовуватися візуальний аналіз даних і збір статистики датасету, що використовується.

Необхідно провести дослідження існуючих алгоритмів прогнозування, виділити їх переваги та недоліки. Використовуючи отриману інформацію вибрати найбільш підходящий і результативний алгоритм, виходячи з отриманих в ході дослідження даних. Розробити архітектуру обраної моделі машинного навчання з урахуванням специфіки предметної галузі та провести її навчання з використанням відкритого датасету.

Існує велика кількість різних видів спорту на професійній сцені. Кожен із яких відрізняється правилами гри, необхідними навичками, умовами перемоги. Розробка універсального рішення, які можна було б застосувати для всіх видів спорту являє собою дуже комплексне завдання. У рамках дослідницької роботи було обрано великий теніс.

Система буде розгорнута в хмарному сервісі, що необхідно для прискорення процесу навчання та забезпечення стабільності її роботи. За такого підходу відповідальність за стабільну роботу системи переходить на провайдерів хмарного сервісу.

Необхідно вжити заходи для забезпечення безпеки та захисту від способів злому, таких як SQL ін'єкцій і XSS.

В даному розділі було проаналізовано проблеми аналогічних систем та поставлено завдання для реалізації інформаційної системи для прогнозування спортивних матчів.

3 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ І АЛГОРИТМІВ

3.1 Первинна обробка даних

Для навчання моделі прогнозування спортивних матчів та вирішення завдання класифікації було знайдено відкритий датасет, який містить у собі необхідні дані. Одним з важливих етапів побудови моделі машинного навчання є попередній аналіз даних, який включає такі пункти:

– очищення та дослідження даних,
– функціональна інженерія,
– моделювання даних та налаштування гіперпараметрів.

Однією з етапів очищення є видалення порожніх даних, т.к. відсутність даних у датасеті може знизити відповідність моделі або може призвести до усунення моделі, оскільки ми неправильно проаналізували поведінку та взаємозв'язок з іншими змінними. Це може призвести до неправильного прогнозу чи класифікації[9].

Виявлення та обробка викидів. Викид - це спостереження, яке здається далеким і розходиться із загальною закономірністю у вибірці. Щоразу, коли ми стикаємося з викидами, ідеальний спосіб їх вирішення – це з'ясувати причину появи цих викидів. Тоді спосіб боротьби з ними залежатиме від причини їх виникнення. Причини викидів можна розділити на великі категорії: неприродні і природні. Найчастіше використовуваний метод виявлення викидів – це візуалізація. Ми використовуємо різні методи візуалізації, такі як коробчаста діаграма, гістограма, точкова діаграма.

Матриця кореляції дозволить нам зрозуміти відносини між різними ознаками. Кореляційна матриця використовується у вигляді теплової карти, щоб зменшити інформаційне навантаження при відображенні занадто великої кількості чисел. Теплова карта дозволяє нам з першого погляду вловити будь-які цікаві стосунки.

У датасеті присутньо досить багато параметрів матчу, турніру і гравця, тому перше завданням, яке ми поставили для себе, − це виділити основні параметри, які

в подальшому будуть використовуватися в якості представлення гравця, тобто, це набір якихось полів, які є значущими для перемоги гравця в матчі. Проблема, яка з'явилася при виділенні цього набору полів полягає в тому, що багато параметрів залежать не тільки від одного гравця і їх неможливо ізолювати.

Наприклад, використовувати для прогнозу особистий рахунок двох гравців між собою не вийде, так як при навчанні ми б отримали, що гравець залежить від всіх гравців, з якими він коли-небудь грав.

Виходячи з цієї проблеми, ми прийняли такі значущі для перемоги параметри:

– відношення перемог до поразок за останній рік,
– відношення перемог до поразок за останній місяць,
– відношення перемог до поразок за весь час,
– максимальна серія перемог,
– максимальна серія поразок.

3.2 Опис існуючих алгоритмів

У задачах класифікації базова вихідна змінна є дискретною. Ця змінна класифікується на різні групи або категорії, наприклад, «переможець» або «програвший». Відповідна вихідна змінна є реальним значенням у проблемах регресії, наприклад, шанс на перемогу у матчі. Далі буде наведено опис алгоритмів машинного навчання з учителем для прогнозування спортивних матчів[10].

Логістична регресія є потужним і добре встановленим методом контрольованої класифікації. Його можна розглядати як розширення звичайної регресії і може моделювати лише дихотомічну змінну. Логістична регресія допомагає знайти ймовірність того, що новий екземпляр належить до певного класу. Оскільки це ймовірність, результат знаходиться між 0 і 1. Отже, щоб використовувати логістичну регресію як бінарний класифікатор, необхідно призначити поріг для диференціації двох класів. Наприклад, значення ймовірності

вище 0,5 для вхідного екземпляра класифікує його як «клас А»; інакше «клас В». Модель логістичної регресії може бути узагальнена для моделювання категоріальної змінної з більш ніж двома значеннями.

Логістична регресія діє дещо дуже схоже на лінійну регресію[11]. Вона також обчислює лінійний вихід, за яким слідує функція обробки результата регресії. Сигмовидна функція є часто використовуваною логістичною функцією. На малюнку нижче показано розподіл сигмовидної функції (рис. 3.1).

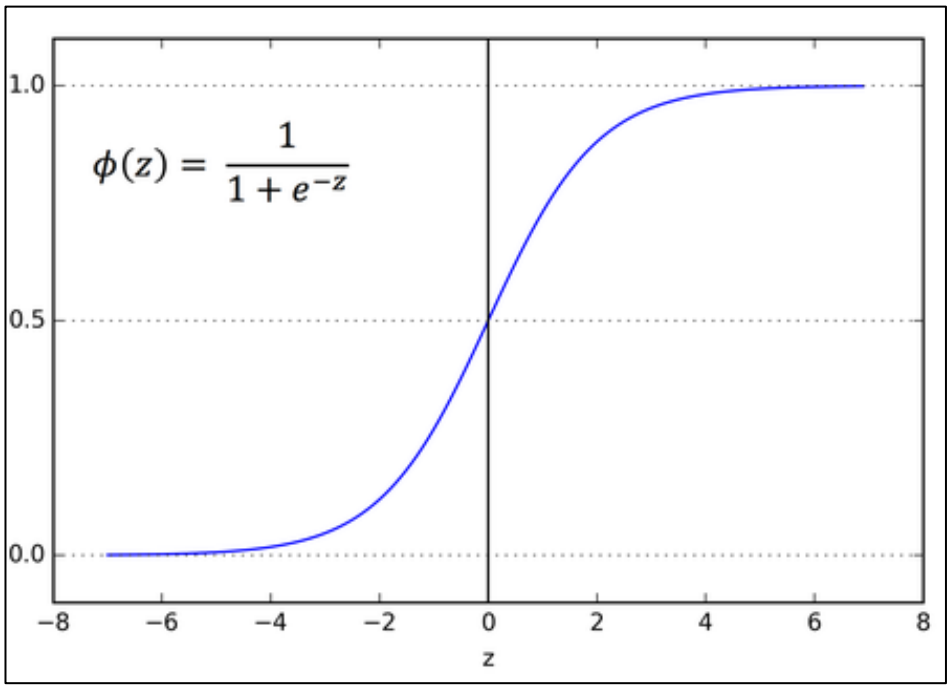

Рисунок 3.1 Розподіл сигмовидної функції.

Цей метод має такі переваги:

- легкий, швидкий і простий метод класифікації.;
- параметри пояснює напрям і інтенсивність значущості незалежних змінних над залежною змінною;
- можна також використовувати для багатокласових класифікацій.

Недоліками логістичної регресії є:

- не можна застосувати до задач нелінійної класифікації;
- потрібен правильний підбір функцій;
- викиди впливають на точність моделі.

Гіперпараметри логістичної регресії подібні до лінійної регресії. Швидкість

навчання та параметр регуляризації повинні бути правильно налаштовані для досягнення високої точності.

Тепер розглянемо метод k найближчих сусідів. У ньому відбувається пошук k сусідів і складається прогноз. У випадку класифікації методом найближчих сусідів, більшість голосів використовується для k найближчих точок даних, тоді як у регресії середнє значення k найближчих точок даних обчислюється як вихід. Як правило, ми вибираємо непарні числа як k. Метод найближчих сусідів − це модель відкладеного навчання, де обчислення відбуваються лише під час виконання.

Нижче зображена графічна інтерпретація алгоритму (рис. 3.2) сині та сірі точки відповідають класам A та B у даних навчання. Жовтий ромб вказує на дані тесту, які підлягають класифікації. коли k = 3, ми прогнозуємо клас B як вихід, а коли K=6, ми прогнозуємо клас A як вихід.

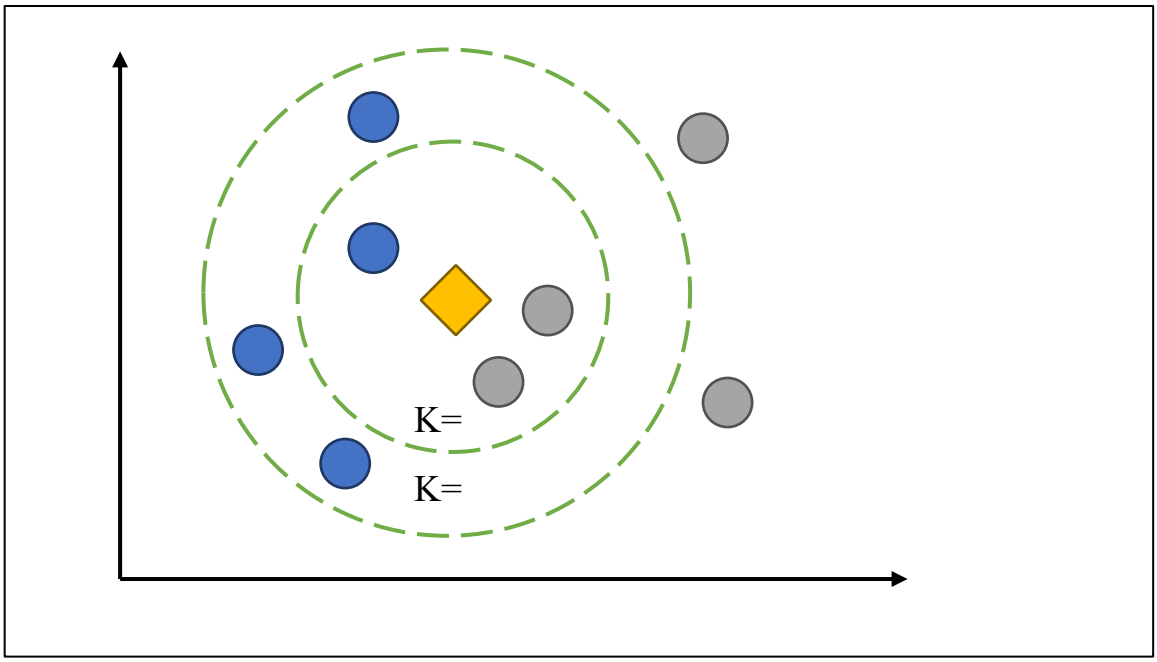

Рисунок 3.2 − Графічна ілюстрація алгоритму k найближчих сусідів.

Цей метод має такі переваги:

- легка та проста модель машинного навчання;
- кілька гіперпараметрів для налаштування.

Недоліками алгоритму k найближчих сусідів є:

- k слід вибирати з розумом;

- великі витрати на обчислення, якщо розмір вибірки великий;
- необхідно забезпечити належне масштабування для справедливого ставлення між функціями.

Метод k найближчих сусідів в основному включає два гіперпараметри, значення K і функцію відстані. Значення k відображає скільки сусідів братиме участь в алгоритмі. Евклідова відстань є найбільш використовуваною функцією.

Тепер розглянемо штучні нейронні мережі, які є набором алгоритмів машинного навчання, які створені на основі функціонування нейронних мереж людського мозку. У біологічному мозку нейрони з'єднані один з одним за допомогою кількох аксонних з'єднань, утворюючи діаграму, подібну до архітектури. Аналогічно, алгоритми штучних нейронних мереж можна представити як взаємопов'язану групу вузлів. Вихід одного вузла надходить як вхід до іншого вузла для подальшої обробки відповідно до взаємозв'язку. Вузли зазвичай групуються в матрицю, яка називається шаром, залежно від перетворення, яке вони виконують.

Крім вхідного та вихідного шарів, у структурі штучних нейронних мереж може бути один або кілька прихованих шарів. Вузли та ребра мають ваги, які дозволяють регулювати потужність сигналу зв'язку, яку можна посилити або послабити за допомогою повторного навчання. На основі навчання та подальшої адаптації матриць, ваг вузлів і ребер штучні нейронні мережі можуть робити прогноз для даних тесту.

Dense neural network − це нейронна мережа, в якій шари повністю пов'язані (щільно) нейронами в мережевому шарі[12]. Кожен нейрон в шарі отримує вхід від усіх нейронів, присутніх в попередньому шарі − таким чином, вони щільно пов'язані. Щільна нейронна мережа (DNN), яка є найпростішою архітектура нейронної мережі є біологічно натхненною обчислювальною моделлю, призначена для моделювання шляху в які людський мозок обробляє. DNN – це нейронна мережа, що складається з нейронів, які самооптимізуються за допомогою навчального процесу. І, як випливає з назви DNN, шари є щільно з'єднані, що означає кожен нейрон у шарі отримує вхід від усіх нейронів попереднього шару.

У будь-якій нейронній мережі dense шар – це шар, який глибоко пов'язаний з

попереднім шаром, що означає, що нейрони шару з'єднані з кожним нейроном його попереднього шару. Цей шар є найбільш часто використовуваним шаром у штучних нейронних мережах.

Нейрон щільного шару в моделі отримує вихідні дані від кожного нейрона свого попереднього шару, де нейрони щільного шару здійснюють множення матриці-вектора[13].

Приклад DNN та взаємозв’язків нейронної мережі зображений нижче на малюнку 3.3.

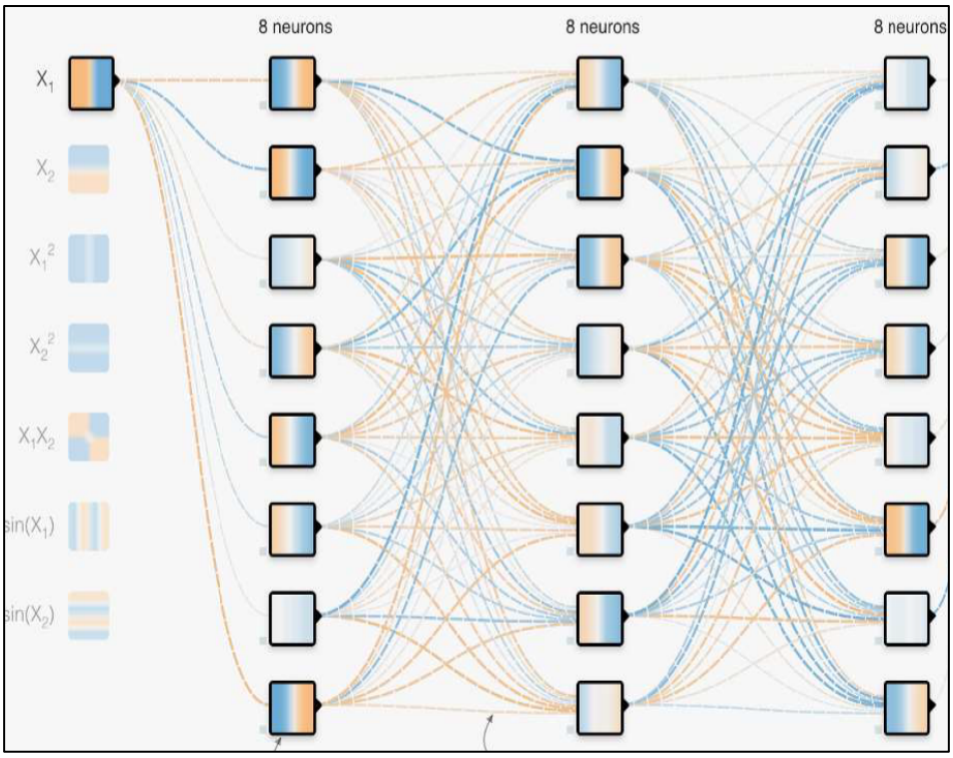

Рисунок 3.3 – Приклад DNN нейронної мережі

Порівняємо представлені алгоритми та виберемо найбільш підходящий для вирішення поставленого завдання.

Логістична регресія та нейронна мережа:

– нейронна мережа може підтримувати нелінійні рішення, а логістична регресія ні;

– логістична регресія має опуклу функцію втрат, тому він не зависає в локальних мінімумах, тоді як нейронна мережа може зависнути;

– логістична регресія перевершує нейронну мережу, коли кількість

навчальних даних невелике, а кількість параметрів велике, тоді як нейронна мережа потребує велику кількість навчальних даних.

Логістична регресія та метод k найближчих сусідів:

- метод k найближчих сусідів – це непараметрична модель, а логістична регресія – параметрична модель;
- метод k найближчих сусідів порівняно повільніше, ніж регресія;
- метод k найближчих сусідів підтримує нелінійні рішення, а логістична регресія підтримує лише лінійні рішення;

Метод k найближчих сусідів та нейронна мережа:

- нейронним мережам потрібні великі навчальні дані порівняно з методом k найближчих сусідів, щоб досягти достатньої точності;
- нейронна мережа потребує значної настройки гіперпараметрів порівняно з методом k найближчих сусідів.

Проаналізувавши порівняння представлених моделей машинного навчання і прийнявши факт, що в обраному в датасеті достатньо даних і нам потрібне нелінійне рішення. нейронна мережа є кращим варіантом.

4 МЕТОД ПРОГНОЗУВАННЯ ПОВНОЗВ’ЯЗНА НЕЙРОНА МЕРЕЖА

4.1 Повнозв’язна нейрона мережа

Як алгоритм для вирішення задачі була обрана нейронна мережа в якій шари повністю пов'язані (щільно) нейронами в мережевому шарі (DNN).

Нейрон щільного шару в моделі отримує вихідні дані від кожного нейрона свого попереднього шару, де нейрони щільного шару здійснюють множення матриці-вектора.

Множення вектора матриці – це процедура, де вектор-рядок виводу з попередніх шарів дорівнює вектору-стовпцю щільного шару. Загальне правило множення матриці-вектора полягає в тому, що вектор-рядок повинен мати стільки стовпців, як і вектор-стовпець. Була спроектована архітектура моделі (рис. 4.1)[14].

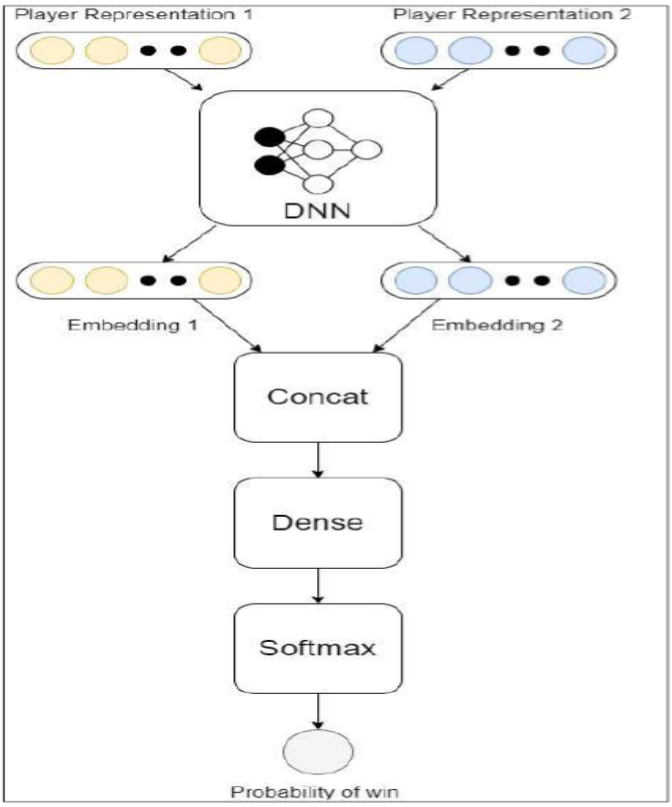

Рисунок 4.1 – Архітектура моделі DNN

На вхід у нас надходить вектор представлення гравця. На схемі зображено, що ми передаємо в нейронну мережу двох гравців, але на ділі це означає, що ми

послідовно передаємо вектор представлення першого і другого гравця і отримуємо два більш високорівневих уявлення. Після чого ми склеюємо два цих вектора, щоб отримати представлення про матч між двома гравцями.

Далі нам потрібно перетворити цей вектор, використовуючи ще один повнозв'язний шар.

І останній етап − це функція softmax, яка дозволяє отримати значення ймовірності перемоги[15].

Використовуючи спроектовану модель ми навчимо її показувати ймовірність виграшу гравців на даних датасету заздалегідь перетворених в уявлення гравців.

4.2 Очищення даних

Набір даних — це сукупність пов’язаних дискретних елементів, пов’язаних даних, доступ до яких можна отримати окремо або в поєднанні.

Набір даних організовано в структуру даних певного типу. У базі даних, наприклад, набір даних може містити колекцію бізнес-даних (імена, зарплати, контактну інформацію, дані про продажі тощо). Саму базу даних можна вважати набором даних, так само як і масиви даних у ній, пов’язані з певним типом інформації, наприклад дані про продажі для певного корпоративного відділу.

Основним завданням моделі, щоб вона була точною та точною в прогнозах, є те, що алгоритм повинен мати можливість легко інтерпретувати характеристики даних.

Попередня обробка та аналіз датасета – це більша частина роботи при розробці та дослідженні моделі машинного навчання (див. рис. 4.2). Попередня обробка даних включає кроки, які нам потрібно виконати, щоб перетворити або закодувати дані, щоб їх можна було легко проаналізувати машиною[16].

Більшість реальних наборів даних дуже сприйнятливі до відсутніх, суперечливих і шумних даних. Застосування алгоритмів аналізу даних до цих шумних даних не дасть якісних результатів, оскільки вони не можуть ефективно

ідентифікувати закономірності. Тому обробка даних важлива для покращення загальної якості даних:

– повторювані або відсутні значення можуть дати неправильне уявлення про загальну статистику даних;

– викиди й суперечливі точки даних часто мають тенденцію порушувати загальне навчання моделі, що призводить до помилкових прогнозів.

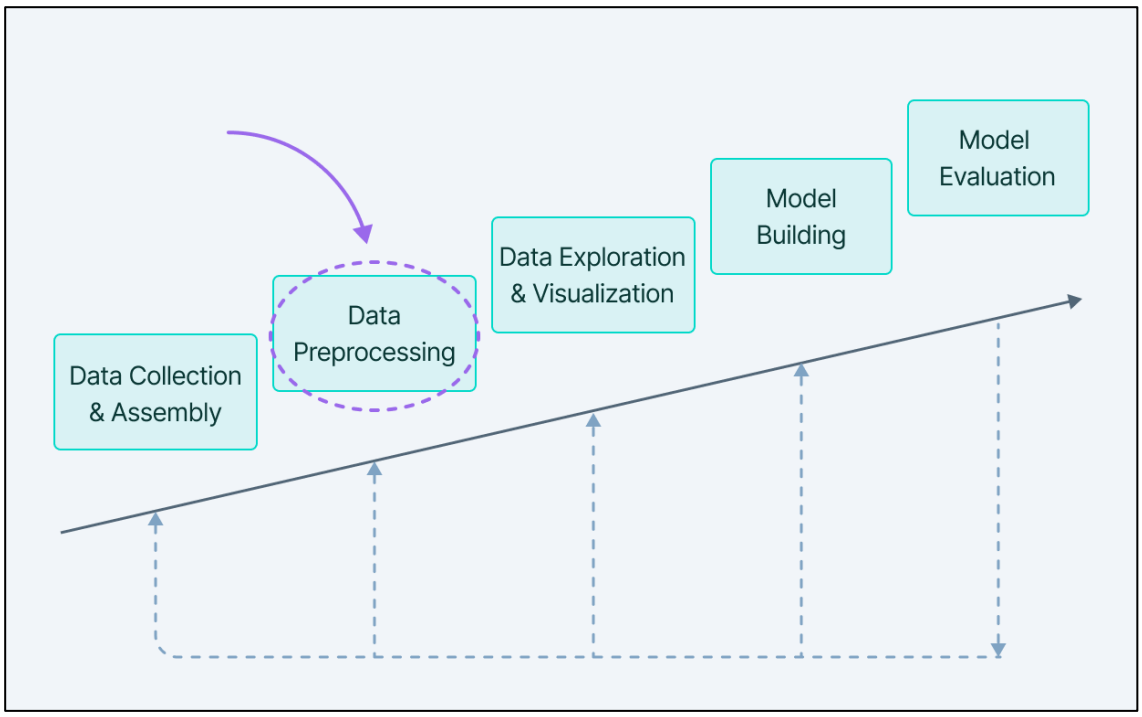

Рисунок 4.2 – Зображення етапу попередньої обробки даних

Окремі незалежні змінні, які є вхідними даними в нашій моделі машинного навчання, називаються ознаками. Їх можна розглядати як уявлення або атрибути, які описують дані та допомагають моделям передбачити класи-мітки.

Очищення даних, зокрема, виконується як частина попередньої обробки даних шляхом заповнення відсутніх значень, згладжування шумних даних, вирішення невідповідності та видалення викидів. Були виділені такі основні проблеми під час роботи з відкритими наборами даних:

– відсутні значення. Ось кілька способів вирішення цієї проблеми:

Ігнорувати ці кортежи. Цей метод слід розглянути, коли набір даних величезний і в кортежі присутні численні відсутні значення. Заповнити пропущені значення. Для

цього існує багато методів, наприклад заповнення значень вручну, передбачення відсутніх значень за допомогою методу регресії або числових методів;

− шумні дані. Це передбачає видалення випадкової помилки або дисперсії у вимірювальній змінній. Це можна зробити за допомогою таких технік: Бінінг − це техніка, яка працює з відсортованими значеннями даних, щоб згладити будь-який шум, присутній у них. Дані поділяються на контейнери однакового розміру, і кожен обробляється незалежно. Усі дані в сегменті можна замінити його середнім, медіаною або граничними значеннями. Регресія − це метод аналізу даних зазвичай використовується для прогнозування. Це допомагає згладити шум, вписуючи всі точки даних у функцію регресії. Рівняння лінійної регресії використовується, якщо є лише один незалежний атрибут; інакше використовуються поліноміальні рівняння. Кластеризація або створення груп із даних, що мають подібні значення. Значення, які не лежать у кластері, можна розглядати як дані з шумом і їх можна видалити;

− видалення викидів. Методи кластеризації об’єднують подібні точки даних. Кортежи, які лежать за межами кластера, є викидами або непослідовними даними.

Інтеграція даних − це один із кроків попередньої обробки даних, який використовується для об’єднання даних, наявних у кількох джерелах, в єдине сховище даних, наприклад сховище даних[17].

4.3 Трансформація даних

Після очищення даних нам потрібно об’єднати якісні дані в альтернативні форми, змінивши значення, структуру або формат даних за допомогою наведених нижче стратегій перетворення даних:

− узагальнення – низькорівневі або детальні дані, які ми перетворили на

інформацію високого рівня за допомогою ієрархії концепцій. Ми можемо перетворити примітивні дані в адресі, як місто, в інформацію вищого рівня, як країна;

− нормалізація − це найбільш важливий метод перетворення даних, який широко використовується. Числові атрибути масштабуються, щоб помістити в заданий діапазон. У цьому підході ми обмежуємо наш атрибут даних певним контейнером, щоб створити кореляцію між різними точками даних. Нормалізацію можна зробити кількома способами, які висвітлені тут: мінімальна-максимальна нормалізація, нормалізація Z-Score, нормалізація десяткового масштабування;

− вибір атрибутів – нові властивості даних створюються з наявних атрибутів, щоб допомогти в процесі аналізу даних. Наприклад, дату народження, можна трансформувати в іншу властивість, що безпосередньо вплине на прогнозування;

− агрегація – це метод зберігання та представлення даних у зведеному форматі. Наприклад, дані про продажі можна агрегувати та трансформувати у форматі місяця та року.

4.4 Зменшення даних

Розмір набору даних у сховищі даних може бути занадто великим, щоб його могли обробляти алгоритми аналізу та інтелекту даних.

Одним з можливих рішень є отримання зменшеного представлення набору даних, який є набагато меншим за обсягом, але дає таку ж якість аналітичних результатів.

Зменшення розмірності застосовують для вилучення ознак використовуються методи зменшення розмірності. Розмірність набору даних відноситься до атрибутів або окремих ознак даних. Ця техніка спрямована на зменшення кількості зайвих функцій, які ми розглядаємо в алгоритмах машинного

навчання. Зменшення розмірності можна зробити за допомогою таких методів, як аналіз основних компонентів тощо.

Стиснення даних досягають використовуючи технології кодування, адже саме так можна значно зменшити розмір даних. Але стиснення даних може бути як із втратами, так і без втрат. Якщо вихідні дані можна отримати після реконструкції зі стиснутих даних, це називається скороченням без втрат, інакше це називається зменшенням втрат.

Дискретизація даних використовується для поділу атрибутів безперервного характеру на дані з інтервалами. Це робиться тому, що безперервні ознаки, як правило, мають менший шанс кореляції з цільовою змінною. Таким чином, може бути важче інтерпретувати результати. Після дискретизації змінної можна інтерпретувати групи, що відповідають цілі.

Вибір підмножини атрибутів не менш важливий, адже необхідно бути конкретним у виборі атрибутів. Інакше це може призвести до отримання даних великої розмірності, які важко навчити через проблеми з недостатнім/переобладнанням. Слід враховувати лише атрибути, які додають більше значення для навчання моделі, а всі інші можна відкинути.

4.5 Візуалізація даних за великої кількості ознак.

Паралельний графік координат використовується для побудови багатовимірних числових даних. Графіки паралельних координат ідеально підходять для порівняння багатьох змінних разом і перегляду взаємозв’язків між ними. Наприклад, якщо вам довелося порівняти низку продуктів з однаковими атрибутами (порівняти характеристики комп’ютера чи автомобілів різних моделей).

На графіку паралельних координат кожній змінній надається своя вісь, і всі осі розміщуються паралельно одна одній.

Кожна вісь може мати різний масштаб, оскільки кожна змінна працює з іншою одиницею вимірювання, або всі осі можна нормалізувати, щоб усі шкали були однаковими.Значення зображуються у вигляді ряду ліній, які з’єднуються через усі осі. Це означає, що кожна лінія являє собою сукупність точок, розміщених на кожній осі, які всі були з’єднані разом.

Порядок розташування осей може вплинути на те, як читач розуміє дані. Однією з причин цього є те, що відносини між сусідніми змінними легше сприймати, ніж для несуміжних змінних.

Таким чином, зміна порядку осей може допомогти виявити закономірності або кореляції між змінними.Графі паралельних координат обраного датасету зображений на рисунку 4.3 та побудований за допомогою бібліотеки Plotly.

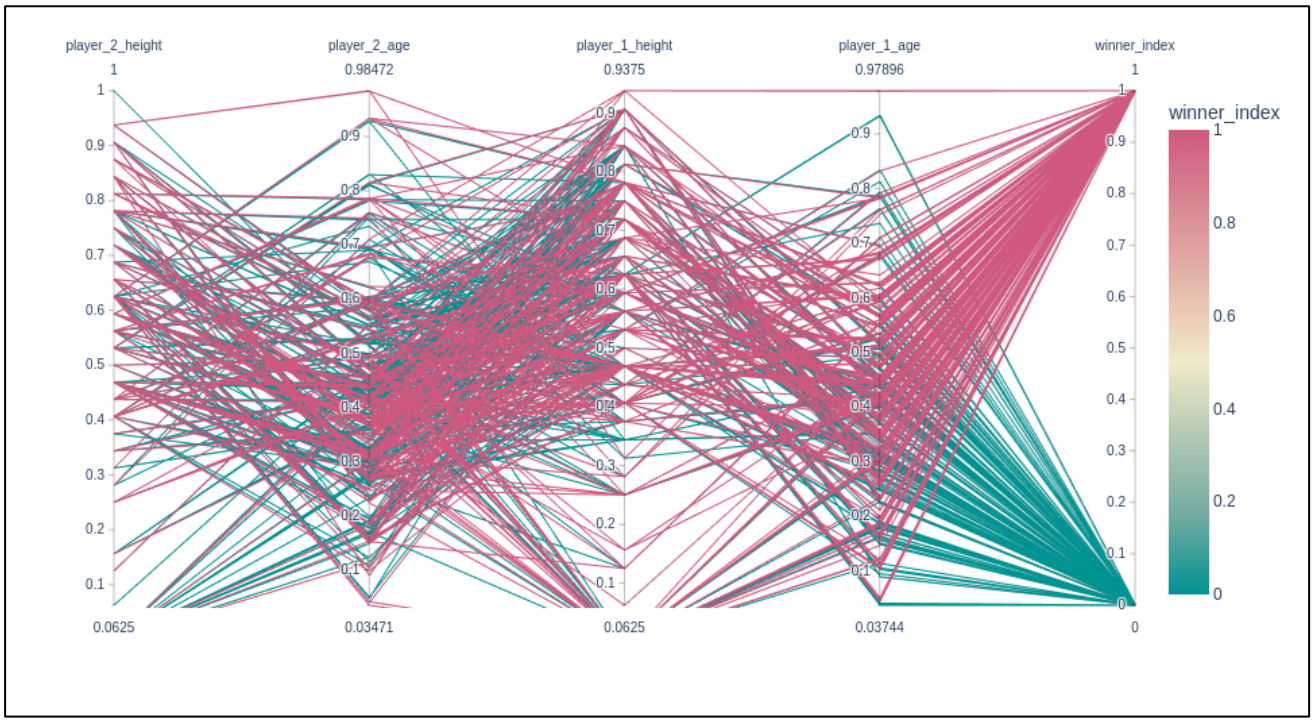

Рисунок 4.3 Паралельний графік координат.

Недоліком паралельних координатних графіків є те, що вони можуть стати нерозбірливими, коли на них дуже багато даних. Найкращий спосіб вирішити цю проблему – це інтерактивність і техніка, відома як «чистка». За допомогою пензля виділяється вибрана лінія або сукупність ліній, а всі інші згасають. Це дозволяє вам ізолювати ділянки, які вас цікавлять, одночасно відфільтровуючи шум[18].

5 КРИТЕРІЇ ОЦІНЮВАННЯ ТА МЕТРИКИ

Показники класифікації дозволяють оцінити продуктивність моделей машинного навчання, але їх дуже багато, кожна з яких має свої переваги та недоліки, і вибір показника оцінки, який підходить для вашої проблеми, іноді може бути дуже складним. Найважливішим завданням у побудові будь-якої моделі машинного навчання є оцінка її продуктивності. Отже, виникає питання, як можна виміряти успіх моделі машинного навчання. Звідки ми дізнаємося, що коли припинити навчання та оцінювання.

У типовій задачі класифікації даних метрика оцінки використовується у двох етапах, етапі навчання та етапі тестування. На етапі навчання метрика оцінки була використовується для оптимізації алгоритму класифікації. Іншими словами, метрика оцінки використовується як дискримінатор для розрізнення та вибору оптимального рішення, яке може дати більш точне прогнозування майбутньої оцінки конкретного класифікатора. Тим часом на етапі тестування метрика оцінки використовувалася як оцінювач для вимірювання ефективності створеного класифікатора під час тестування з невидимими даними.

Показники оцінювання прив’язані до завдань машинного навчання. Існують різні метрики для завдань класифікації та регресії. Деякі показники, як точність відкликання, корисні для виконання кількох завдань. Класифікація є прикладом навчання з учителем, яке становить більшість програм машинного навчання. Використовуючи різні показники для оцінки продуктивності, ми повинні мати можливість покращити загальну передбачувану силу нашої моделі, перш ніж запровадити її для виробництва на невидимих даних. Невиконання належної оцінки моделі машинного навчання з використанням різних показників оцінки залежно від точності може призвести до проблеми, коли відповідна модель буде розгорнута на невидимих даних, і може закінчитися поганими прогнозами.

Показник класифікації – це число, яке вимірює ефективність вашої моделі машинного навчання, коли справа доходить до призначення спостережень певним класом. Бінарна класифікація – це особлива ситуація, коли вам просто потрібно

класи: позитивні та негативні. Зазвичай продуктивність представлена в діапазоні від 0 до 1, де оцінка 1 зарезервована для ідеальної моделі.

Точність просто вимірює, наскільки часто класифікатор правильно прогнозує. Ми можемо визначити точність як відношення кількості правильних прогнозів до загальної кількості передбачень. Коли будь-яка модель дає точність 99%, ви можете подумати, що модель працює дуже добре, але це не завжди так. Розглянемо задачу бінарної класифікації, де модель може досягти лише двох результатів, будь-яка модель дає правильний або неправильний прогноз. Тепер уявіть, що у нас є завдання класифікації, щоб передбачити, чи переможе гравець у матчі. У алгоритмі навчання з наглядом ми спочатку навчаємо модель на навчальних даних, а потім тестуємо модель на даних тестування. Отримавши передбачення моделі з даних, ми порівнюємо їх із справжніми значеннями.

Точність корисна, коли цільовий клас добре збалансований, але не є хорошим вибором для незбалансованих класів. Отже, якщо ми хочемо зробити кращу оцінку моделі та отримати повну картину оцінки моделі, слід також враховувати інші показники, такі як запам’ятовування та точність.

Відкликання розраховується як відношення між кількістю позитивних зразків, правильно класифікованих як позитивні, до загальної кількості позитивних зразків. Відкликання вимірює здатність моделі виявляти позитивні зразки. Чим вище відкликання, тим більше було виявлено позитивних зразків. Відкликання хвилює лише те, як класифікуються позитивні зразки. Це не залежить від того, як класифікуються негативні зразки для точності. Коли модель класифікує всі позитивні зразки як позитивні, тоді відкликання буде 100%, навіть якщо всі негативні зразки були неправильно класифіковані як позитивні.

Незалежно від того, як класифікуються негативні зразки, відкликання стосується лише позитивних зразків. Оскільки не має значення, чи класифікуються негативні зразки як позитивні чи негативні, краще взагалі знехтувати негативними зразками. Під час розрахунку відкликання потрібно враховувати лише позитивні зразки. Коли відкликання високе, це означає, що модель може правильно класифікувати всі позитивні зразки як позитивні. Таким чином, моделі можна довіряти в її здатності виявляти позитивні зразки.

Оскільки відкликання нехтує тим, як класифікуються негативні зразки, все ще може бути багато негативних зразків, класифікованих як позитивні (тобто високий рівень хибнопозитивних). Відкликання цього не враховує.

Рішення про використання точності чи відкликання залежить від типу проблеми, що вирішується. Якщо мета полягає в тому, щоб виявити всі позитивні зразки не піклуючись про те, що негативні зразки будуть неправильно класифіковані як позитивні, треба використовувати відкликання. Потрібно використовувати точність, якщо проблема чутлива до класифікації зразка як позитивного в цілому, тобто включаючи негативні зразки, які були помилково класифіковані як позитивні.

Для задач бінарної класифікації оцінка дискримінації найкращого рішення під час навчання може бути визначена на основі матриці плутанини. Рядок таблиці представляє прогнозований клас, а стовпець – фактичний клас.

Таблиця 5.1 Матриця плутанини.

	фактичний позитивний клас	фактичний негативний клас
прогнозований позитивний клас	True positive (tp)	False negative (fn)
прогнозований негативний клас	False positive (fp)	True negative (tn)

Матриця плутанини забезпечує більш повне уявлення про продуктивність моделі, надаючи нам точні пропорції правильно і неправильно передбачених класів у вигляді таблиці або матриці.

Найважливіше, що можно зробити, щоб правильно оцінити модель – це не тренувати модель на всьому наборі даних. Типовим поділом буде використання 70% даних для навчання і 30% даних для тестування.

Важливо використовувати нові дані під час оцінки нашої моделі, щоб

запобігти ймовірності переобладнання навчального набору. Однак іноді буває корисно оцінити нашу модель під час її створення, щоб знайти найкращі параметри моделі, але ми не можемо використовувати тестовий набір для цієї оцінки, інакше ми виберемо параметри, які найкраще працюють на тестових даних, але не параметри, які найкраще узагальнюють. Щоб оцінити модель під час створення й налаштування моделі, ми створюємо третю підмножину даних, відому як набір перевірки. Типовим розподілом буде використання 60% даних для навчання, 20% даних для перевірки та 20% даних для тестування.

6 ТЕХНОЛОГІІ ТА ІНСТРУМЕНТИ

Існують різні інструменти, програмне забезпечення та платформи, доступні для машинного навчання, а також нове програмне забезпечення та інструменти розвиваються з кожним днем. Хоча існує багато варіантів і доступність інструментів машинного навчання, вибір найкращого інструменту для вашої моделі є складним завданням. Якщо ви виберете правильний інструмент для своєї моделі, ви зможете зробити його швидшим і ефективнішим.

Точніше, основна функціональність scikit-learn включає класифікацію, регресію, кластеризацію, зменшення розмірності, вибір моделі та попередню обробку. Бібліотека дуже проста у використанні та головне ефективна, оскільки вона побудована на NumPy.

Основними перевагами цієї бібліотеки є:

− підвищена швидкість обчислень: scikit-learn спеціалізується на обробці складних математичних задач, включаючи чисельну інтерполяцію, інтегрування, лінійну алгебру та статистику. Оскільки бібліотека призначена для керування різними математичними операціями, вона робить це швидко. В результаті значно зростає загальна обчислювальна швидкість розробки моделі та її інтеграції в існуючу систему;

− підходить для роботи з класичними алгоритмами машинного навчання: типові класичні алгоритми включають алгоритми класифікації, регресії та кластеризації. Ці алгоритми використовуються для вирішення конкретних програм, які покращують загальну точність обчислень;

− проста інтеграція з інструментами SciPy: бібліотека Scikit-Learn доповнює вже існуючий набір наукових і числових бібліотек Python. Він легко інтегрується з іншими бібліотеками та допомагає покращити функції та функціональність цільового програмного продукту.

Для попереднього аналізу даних, їх очищення та виявлення першочергових закономірностей необхідний інструмент візуалізації. Для цього було використано бібліотеку seaborn[19]. Вона розроблена для створення статистичної графіки на

Python та побудована на основі matplotlib і тісно інтегрується зі структурами даних pandas.

Необхідно реалізувати основний код нейроної мережі, для цього використаємо мову програмування Python[20]. Наведемо приклад реалізації моделі.

```
def get_model(input_shape):
    left_input = Input(input_shape)
    right_input = Input(input_shape)
    model = Sequential()
    model.add(Dense(100, activation='sigmoid'))
    encoded_l = model(left_input)
    encoded_r = model(right_input)
    concatted = Concatenate()([encoded_l, encoded_r])
    flatten = Flatten()(concatted)
    prediction = Dense(2, activation='softmax')(flatten)
    model_net = Model(inputs=[left_input, right_input], outputs=prediction)

    return model_net

def get_model_for_train(model_path, show_summary=True):
    model = get_model (input_shape=(1, 2))
    model.load_weights(model_path)
    if show_summary:
        model.summary()

    return model
```

.Лістінг 6.1 – Імплементація нейроної мережі на мові програмування Python.

Як результат, було розроблено архітектуру нейронної мережі та реалізовано алгоритм навчання мережі. Нейронна мережа як інструмент машинного навчання є досить універсальним та потужним для вирішення задач класифікації.

Основні переваги використання бібліотек машинного навчання Python:

− мова Python пропонує описовий та інтерактивний код, який легко вивчати, інтерпретувати та розуміти. Зрозуміла мова робить його придатним для початківців. Більше того, простота бібліотек Python дозволяє програмістам проектувати надійні системи;

− python – це незалежна від платформи мова. Це означає, що Python може запускати програми на таких платформах, як Linux, Windows і macOS, не вимагаючи інтерпретатора Python у відповідних операційних системах;

− бібліотеки Python є безкоштовними та відкритими. Це робить їх

відкритими для постійних удосконалень та оновлень;

− python надає широкий набір бібліотек, які дозволяють користувачам вирішувати кожну існуючу проблему;

− бібліотеку Python легко інтегрувати з іншими інструментами. Крім того, бібліотека доступна будь-якій людині і не вимагає особливих навичок. Спільнота спрощує впровадження бібліотеки Python, оскільки члени спільноти швидко діляться, обговорюють та вирішують проблеми.

7 РЕЗУЛЬТАТИ ТА ЇХ АПРОБАЦІЯ

У результаті дослідження розглядається набір даних із великою кількістю властивостей. Для спрощення процесу візуалізації результатів було застосовано метод PCA зменшення розмірності. Аналіз основних компонентів, або PCA, – це метод зменшення розмірності, який часто використовується для зменшення розмірності великих наборів даних шляхом перетворення великого набору змінних у менший, який все ще містить більшу частину інформації. Зменшення кількості змінних у наборі даних, природно, відбувається за рахунок точності, але хитрість у зменшенні розмірності полягає в тому, щоб поміняти невелику точність на простоту. Тому що менші набори даних легше досліджувати та візуалізувати, а також зробити аналіз даних стає набагато простішим і швидшим для алгоритмів машинного навчання без сторонніх змінних для обробки.

Покрокове пояснення PCA:

– Стандартизація. Метою цього кроку є стандартизація діапазону неперервних вихідних змінних, щоб кожна з них вносила однаковий внесок у аналіз. Точніше, причина, чому важливо виконати стандартизацію перед PCA, полягає в тому, що останній є досить чутливим щодо дисперсій початкових змінних. Тобто, якщо є великі відмінності між діапазонами вихідних змінних, ті змінні з більшими діапазонами будуть домінувати над змінними з малими діапазонами, що призведе до упереджених результатів. Таким чином, перетворення даних до порівнянних масштабів може запобігти цій проблемі;

– обчислення коваріаційної матриці. Мета цього кроку полягає в тому, щоб зрозуміти, як змінні вхідного набору даних відрізняються від середнього по відношенню один до одного, або, іншими словами, з’ясувати, чи існує якийсь зв’язок між ними. Тому що іноді змінні дуже корелюють таким чином, що містять зайву інформацію. Отже, щоб ідентифікувати ці кореляції, ми обчислюємо коваріаційну матрицю;

– обчислення власних векторів значень коваріаційної матриці, щоб визначити головні компоненти. Власні вектори та власні значення – це поняття

лінійної алгебри, які нам потрібно обчислити з коваріаційної матриці, щоб визначити головні компоненти даних. Головними компонентами є нові змінні, які будуються як лінійні комбінації або суміші вихідних змінних. Ці комбінації виконуються таким чином, що нові змінні були не корельовані, і більшість інформації в початкових змінних стискається або стискається в перші компоненти. Отже, ідея полягає в тому, що 10-вимірні дані дають вам 10 основних компонентів, але PCA намагається помістити максимальну можливу інформацію в перший компонент, потім максимальну інформацію, що залишилася, у другий і так далі..

Перш ніж перейти до методів налаштування, необхідно розглянути мету поділу наших даних на дані навчання та тестування. Кінцева мета будь-якої моделі машинного навчання полягає в тому, щоб вчитися на прикладах таким чином, щоб модель була здатна узагальнити навчання на нові екземпляри, яких вона ще не бачила. На самому базовому рівні ви повинні тренуватися на підмножині вашого загального набору даних, викладаючи решту даних для оцінки, щоб оцінити здатність моделі до узагальнення. При досліджуванні різних архітектур моделей, також потрібен спосіб оцінити здатність кожної моделі узагальнювати на невидимі дані. Однак, якщо ви використовуєте дані тестування для цієї оцінки, ви в кінцевому підсумку пристосуєте архітектуру моделі до даних тестування, втративши можливість дійсно оцінити, як модель працює з невидимими даними.

Щоб пом’якшити це, в кінцевому підсумку розділимо загальний набір даних на три підмножини: навчальні дані, дані перевірки та дані тестування. Введення набору даних перевірки дозволяє нам оцінити модель на даних, відмінних від того, на яких вона була навчена, і вибрати найкращу архітектуру моделі, зберігаючи при цьому підмножину даних для остаточної оцінки в кінці розробки нашої моделі.

За результатами першого експерименту набір даних матчів з тенісу за останні 5 років з використанням моделів логістичної регресії, K найближчих сусідів та DNN. Порівняння показує, що модель DNN перевершує модель K найближчих сусідів за продуктивністю, простотою та часом обчислень.

Після проведення необхідних перетворень та навчання було отримано результати роботи обраних моделей машинного навчання (рис. 7.1). Як ми можемо бачити, що розподіл даних у просторі ознак є досить кучним і проведення поділу

на класи поблизу центру є досить важким завданням для всіх трьох методів

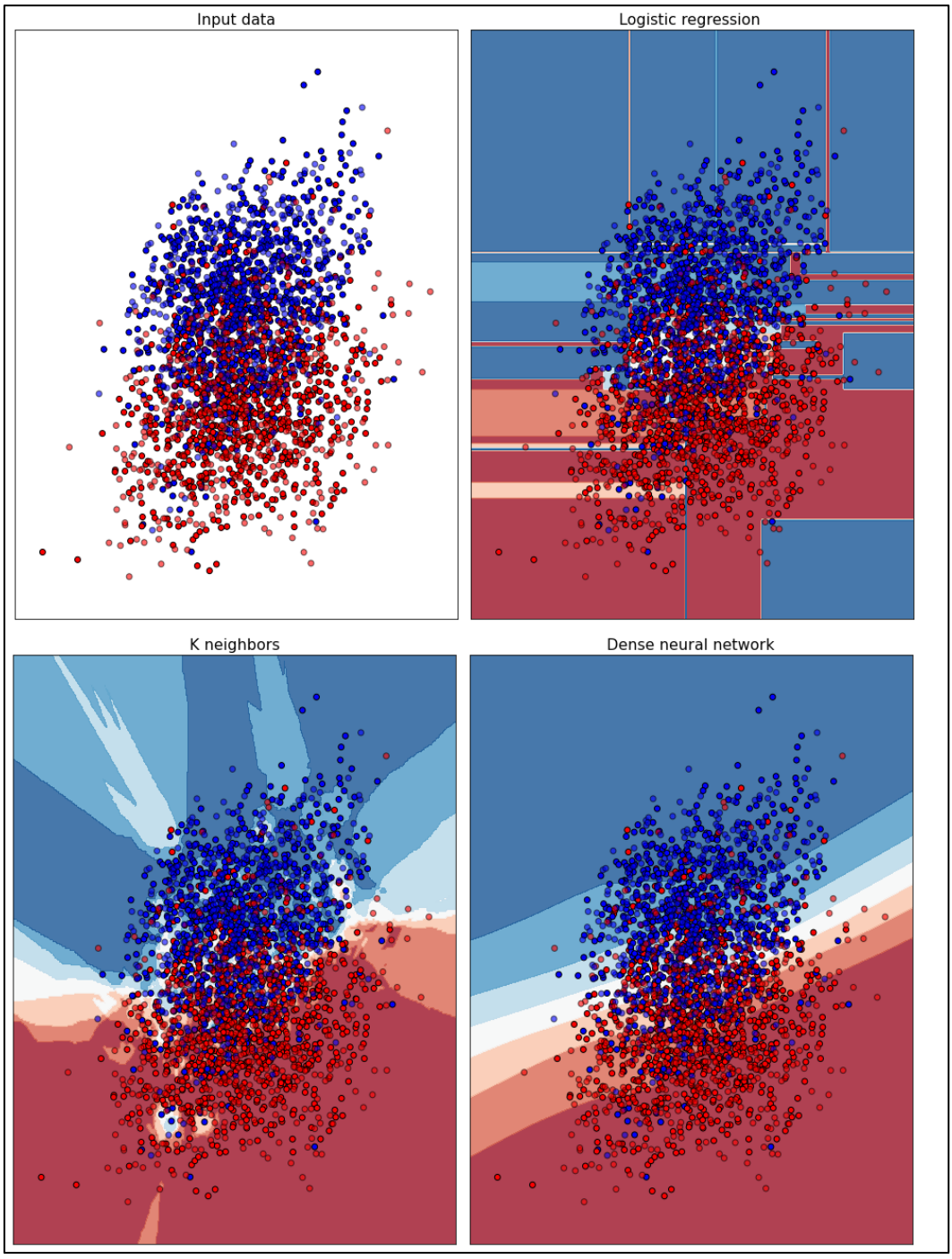

Рисунок 7.1- Візуалізація результатів датасету у двомірному просторі.

Ці результати можна покращити за рахунок збільшення глибина моделі. Наприклад, шляхом збільшення кількість шарів від 64 до 512 шарів, модель досягає кращої оцінки. Однак для порівняння, відповідно до ефективності моделі, кількість шарів для моделі DNN зберігається дорівнює 64.

Слід зазначити що результати є оцінкою, а не визначеним значенням з огляду

на стохастичний характер моделей. Прогноз DNN не показують подальшого значного покращення за рахунок збільшення глибина моделі на основі тестових експериментів. Крім того, у цьому експерименті з меншим розміром набору даних, використання перехресної перевірки не призводить до значного покращення результатів. Результати DNN значно покращуються, коли набір даних збільшується.

Якщо розглядати час навчання різних моделей (рис. 7.2), то можна побачити що час навчання повної нейронної мережі набагато перевищує дві інші моделі.

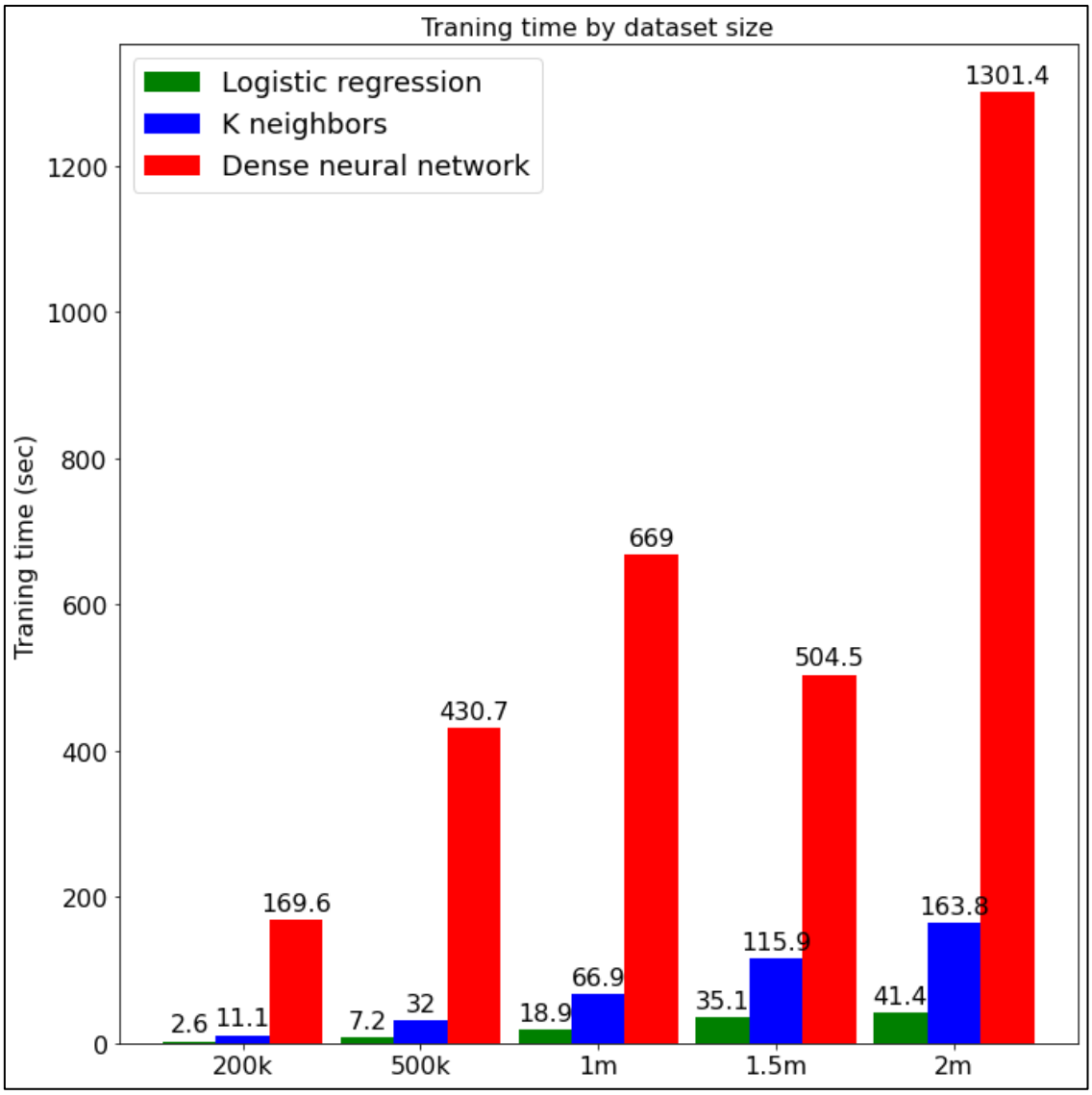

Рисунок 7.2 – Діаграма часу навчання алгоритмів.

Це насамперед пов'язано із складністю алгоритму навчання. Щоб вибрати найкращу модель машинного навчання для даного набору даних, важливо

враховувати особливості або параметри моделі. Параметри допомагають оцінити гнучкість моделі, її припущення.

Якщо порівнювати дві моделі лінійної регресії, одна модель може бути спрямована на зменшення середньоквадратичної помилки, тоді як інша може прагнути зменшити середню абсолютну помилку за допомогою цільових функцій. Щоб зрозуміти, чи підходить друга модель краще, нам потрібно зрозуміти, чи впливають викиди в даних на результати, чи вони не повинні впливати на дані. Якщо необхідно враховувати аномалії або викиди, використання другої моделі з цільовою функцією як середньої абсолютної похибки буде правильним вибором.

Аналогічно для класифікації, якщо розглядаються три моделі, то основою для порівняння буде ступінь узагальнення, якого може досягти модель.

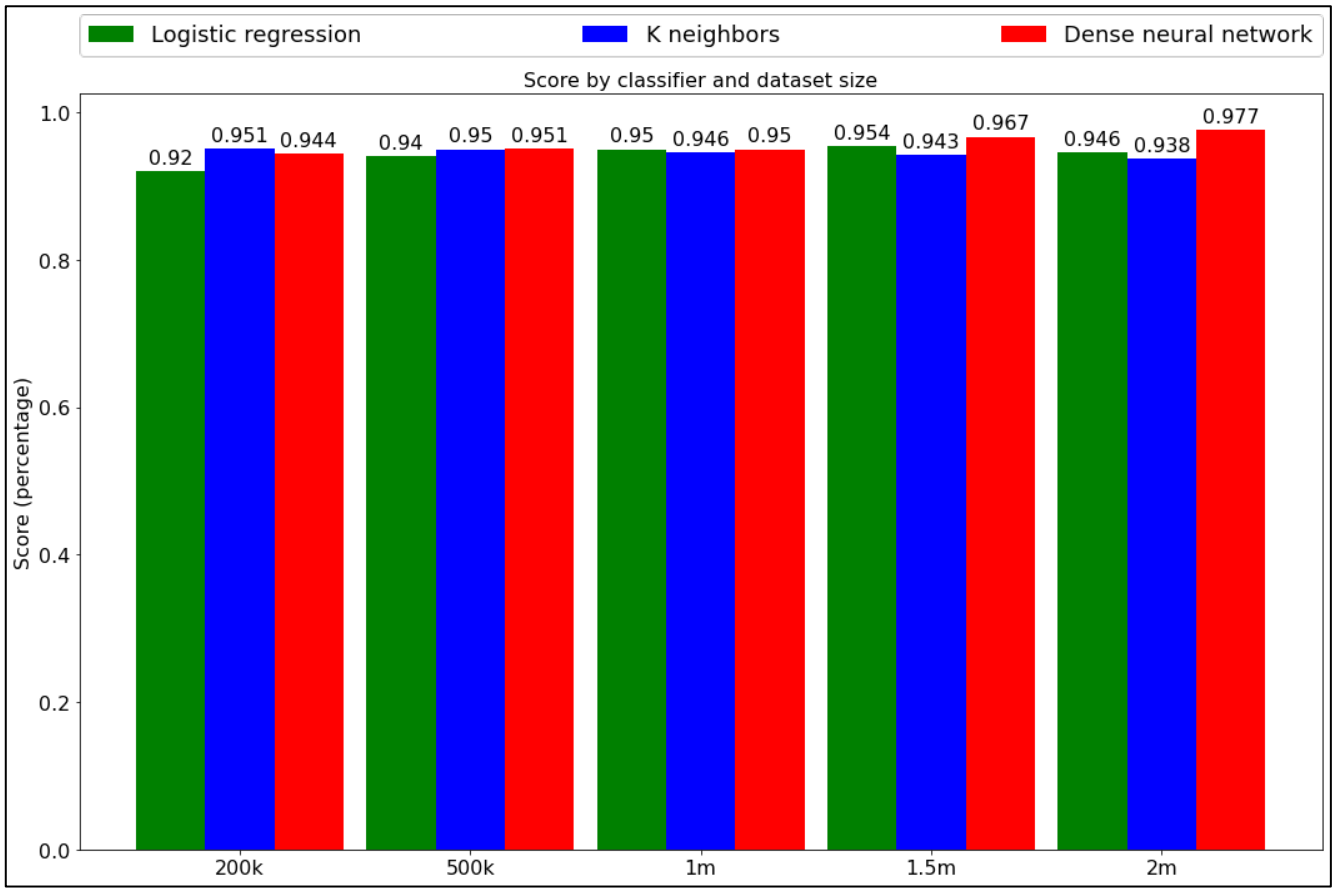

Рисунок 7.3 – Діаграма результатів точності алгоритмів.

Після проведення всіх налаштувань та вибору необхідних значень гіпер параметрів були отримані результати порівняння (рис. 7.3). Як видно з отриманих результатів, повнозв'язна нейронна мережа має більший потенціал для навчання і

починає показувати кращий результат при збільшенні розміру вибірки.

Мабуть, найвідомішим недоліком нейронних мереж є їхня природа чорного ящика, що означає, що ви не знаєте, як і чому ваша DNN отримала певний результат. Такі алгоритми, як дерева рішень, дуже добре інтерпретуються. Це важливо, оскільки в деяких областях інтерпретація є досить важливою.

Нейронні мережі зазвичай вимагають набагато більше даних, ніж традиційні алгоритми машинного навчання, як принаймні тисячі, якщо не мільйони зразків. Вирішити цю проблему непросто, і багато проблем машинного навчання можна добре вирішити з меншою кількістю даних, якщо ви використовуєте інші алгоритми.

8 ПОДАЛЬШІ РЕКОМЕНДАЦІЇ

Одним з головних ключів до успіху є точність і продуктивність моделі. Продуктивність моделі є головним чином технічним фактором, і для низки випадків використання машинного навчання та глибокого навчання розгортання не має сенсу, якщо модель недостатньо точна для даного випадку використання бізнесу.

У контексті вдосконалення існуючих моделей машинного навчання та глибокого навчання не існує універсальної стратегії, яку можна було б послідовно застосовувати. Розглянемо набір рекомендацій і найкращих практик, щоб систематично визначити потенційні джерела підвищення точності та продуктивності моделі.

Наприклад, модель машинного навчання для прогнозування кредитного рейтингу нових роздрібних банківських клієнтів також повинна мати можливість пояснити своє рішення у разі відхилення заявки на отримання кредитної картки. Тут просто оптимізувати технічні показники недостатньо, якщо модель не пропонує пояснення та вказівки для клієнта, щоб зрозуміти та покращити свій кредитний рейтинг.

Розглянемо кілька методів для покращення продуктивності моделі. Зрештою, практика та досвід роботи над різноманітними моделями дають змогу краще зрозуміти найкращі підходи до підвищення точності моделі та встановлення пріоритету цих методів над іншими.

В ідеальному сценарії будь-якому моделюванню або алгоритмічній роботі з машинного навчання передує ретельний аналіз наявної проблеми, включаючи точне визначення варіанту використання та ділових і технічних показників для оптимізації.

Не потрібно втрачати з поля зору попередньо визначені рекомендації щодо анотації даних, стратегії створення наборів даних, показники та критерії успіху, коли починається захоплюючий етап створення машинного навчання або моделей глибокого навчання. Та пам’ятати про ширшу картину корисно для впорядкування

та визначення пріоритетів ітераційного процесу покращення машинного навчання та моделей глибокого навчання.

Якщо модель переобладнана, її можна покращити за допомогою:

– використання більшої кількості навчальних даних,

– зниження складності моделі,

– методами регуляризації,

– ранньої зупинки.

Якщо модель недостатньо підібрана, це можна вирішити, зробивши модель більш складною, тобто додавши більше функцій або шарів, і навчаючи модель для більшої кількості епох.

ВИСНОВКИ

В результаті кваліфікаційної роботи було проведено дослідження та виявлено найбільш відповідний до завдання метод прогнозування результатів спортивних матчів.

Експерименти проводилися з використанням мови програмування Python та таких бібліотек машинного навчання як: NumPy, Pandas, Scikit-learn. Дослідження датасету відбувалося за допомогою бібліотеки візуалізації seaborn. У ході якого було виділено основні ознаки об'єктів та їх взаємозв'язок.

Було проведено дослідження існуючих моделей машинного навчання, призначених для вирішення завдання класифікації з учителем. Серед яких були докладно описані такі алгоритми: логістична регресія, метод k найближчих сусідів та нейронна мережа. При порівнянні їх переваг, недоліків і гіперпараметрів було обрано найбільш підходящий алгоритм вирішення поставленої під час роботи завдачі, саме нейронна мережа(DNN). Вибір був обґрунтований необхідністю вирішення нелінійних завдань класифікації та наявністю достатньої для навчання кількості даних.

Було збудовано архітектуру нейронної мережі та проведено її навчання з використанням відкритого датасету матчів з тенісу.

В подальшому ця система може бути розвинута та адаптована під інші види спорту та роботу з різним набором ознак.

Для наступних релізів даної системи можливе створення великої платформи даних, яка буде взаємодіяти з сервісами зі збору статистичних даних щодо матчів різних видів спорту. Це дозволить зробити передбачення більш точними. Нова платформа прискорить роботу вже існуючих сервісів через те, що буде чітке розподілення між кластерами аналізу, та кластерами збору інформації, це дозволить зробити більш автономну систему.

В якості майбутніх технологій планується використання технологій Big Data(Kafka, Spark). При побудування нової платформи у системі з’явиться можливість більш чітко передбачувати результати матчів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. К. Бочавер, Л. Довжик. Психологія спортивної травми. Litres - 14 січня 2021 - 473 с.

2. B. Min, J. Kim, C. Choe, H. Eom, and R. Ian, “A Compound Framework for Sports Prediction: The Case Study of Football,” undefined, 2007.

3. F. Thabtah, L. Zhang, and N. Abdelhamid, “NBA Game Result Prediction Using Feature Analysis and Machine Learning,” Ann. Data Sci., vol. 6, no. 1, pp. 103–116, Mar. 2019.

4. Sport performance analysis – iSports Analysis [Електронний ресурс] – Режим доступу до ресурсу: https://www.isportsanalysis.com/sport-performance.php (дата звернення: 12.05.2022).

5. Смеляков К., Чуприна А., Пономаренко О. та інші. Search by Image Engine using Local Feature Detectors. 2020 IEEE Open Conference of Electrical, Electronic and Information Sciences, eStream 2020 – Proceedings, 2020.

6. Analyse any sport – once [Електронний ресурс] – Режим доступу до ресурсу: https://www.once.de/analyse-any-sport (дата звернення: 12.05.2022).

7. Смеляков К., Смеляков С., Чуприна А. Adaptive edge detection models and algorithms. Studies in Computational Intelligencethis link is disabled, 2020, 876, pp. 1-51.

8. Т. Фоусет, Ф. Провост. Data Science для бізнесу. Наш формат – 2019 – 400с.

9. Корнієнко Є.В. Методи прогнозування та прийняття рішень. Концеп – Жовтень 13, 2012 – 100 с.

10. J. Otero and L. Sánchez, “Induction of descriptive fuzzy classifiers with the Logitboost algorithm,” Soft Comput., vol. 10, no. 9, pp. 825–835, Jul. 2006.

11. A. Afshar, A. Zahraei, and M. A. Mariño, “Large-Scale Nonlinear Conjunctive Use Optimization Problem: Decomposition Algorithm,” J. Water Resour. Plan. Manag., vol. 136, no. 1, pp. 59–71, Jan. 2010.

12. С. Николенко, А. Кадурін, Є. Архангельська. Глубоке навчання. Пітер – Жовтень 26, 2017 – 480 с.

13. Vishal Meshram, Kailas Patil, Vidula Meshram, Dinesh Hanchate, S.D. Ramkteke. Machine learning in agriculture domain: A state-of-art survey, Artificial Intelligence in the Life Sciences, Volume 1 – 2021 – 11p.

14. К. Берлінський. Основи нейромереж. Litres – Квітень 22, 2020 – 170 с.

15. Federico Filipponi. Sentinel-1 GRD preprocessing workflow – June 2019 – 5p.

16. K. Smelyakov, A. Chupryna, D. Sandrkin, M. Kolisnyk. Search by Image Engine for Big Data Warehouse. IEEE – 2020. P. 1-4.

17. K. Smelyakov, A. Chupryna, O. Bohomolov, N. Hunko. The Neural Network Models Effectiveness for Face Detection and Face Recognition. IEEE – 2021. p. 1-7

18. Y. Freund, R.E. Schapire “Large margin classification using the perceptron algorithm,” Mach. Learn., vol. 37, no. 3, pp. 277–296, Dec. 1999.

19. Seaborn documentation [Електронний ресурс] – Режим доступу до ресурсу: https://seaborn.pydata.org (дата звернення: 12.05.2022).

20. David Beazley, Brian K. Jones. Python Cookbook: Recipes for Mastering Python 3 – June 2011 – 1392 p.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ ЗА НАУКОВИМИ НАПРЯМАМИ КЕРІВНИКА ТА НАУКОВЦІВ КАФЕДРИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ

5. Смеляков К., Чуприна А., Пономаренко О. та інші. Search by Image Engine using Local Feature Detectors. 2020 IEEE Open Conference of Electrical, Electronic and Information Sciences, eStream 2020 – Proceedings, 2020.

7. Смеляков К., Смеляков С., Чуприна А. Adaptive edge detection models and algorithms. Studies in Computational Intelligencethis link is disabled, 2020, 876, pp. 1-51.

16. K. Smelyakov, A. Chupryna, D. Sandrkin, M. Kolisnyk. Search by Image Engine for Big Data Warehouse. IEEE – 2020. P. 1-4.

17. K. Smelyakov, A. Chupryna, O. Bohomolov, N. Hunko. The Neural Network Models Effectiveness for Face Detection and Face Recognition. IEEE – 2021. p. 1-7