

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління
Кафедра Комп'ютерних інтелектуальних технологій та систем

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

рівень вищої освіти другий (магістерський)

Інтелектуальна система прогнозування результату спортивних змагань.

Виконав:

студент 2 курсу, групи КІТм-21-2

Овчаренко Є.С.

Спеціальність 123 Комп'ютерна інженерія

Тип програми освітньо-професійна

Освітня програма Комп'ютерні
інтелектуальні технології

Керівник проф. Аксак Н.Г.

Допускається до захисту

Зав. кафедри

проф. Руденко О.Г.

2022 р.

Харківський національний університет радіоелектроніки

| | |
|---------------------|---|
| Факультет | Комп'ютерної інженерії та управління |
| Кафедра | Комп'ютерних інтелектуальних технологій та систем |
| Рівень вищої освіти | другий (магістерський) |
| Спеціальність | 123 Комп'ютерна інженерія |
| Тип програми | освітньо-професійна |
| Освітня програма | Комп'ютерні інтелектуальні технології |

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« _____ » _____ 202_ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _____ Овчаренку Євгенію Сергійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Інтелектуальна система прогнозування результату спортивних змагань

затверджена наказом по університету від “ 07 ” листопада 2022 р. № 1455 Ст _____

2. Термін подання студентом роботи до екзаменаційної комісії 10.12.2022

3. Вхідні дані до роботи _____

1) виявлення основних умов працездатності гроху-адрес;

2) побудова тестової моделі нейронної мережі для прогнозування результатів змагань;

3) середовище моделювання – Eclipse;

4) мова програмування – Python;

5) нейронні мережі

4. Перелік питань, що потрібно опрацювати в роботі _____

1) огляд предметної області;

2) аналіз предмету дослідження;

3) дослідження нейронних мереж;

4) дослідження інтелектуальних систем прогнозування;

5) дослідження методів прогнозування;

6) розробка інтелектуальної системи оцінки за попередніми даними;

7) експериментальні дослідження;

8) висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 12 слайдів

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

| Найменування розділу | Консультант (посада, прізвище, ім'я, по батькові) | Позначка консультанта про виконання розділу | |
|----------------------|--|---|------|
| | | підпис | дата |
| | | | |
| | | | |

КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Термін виконання етапів роботи | Примітка |
|----|---|--------------------------------|----------|
| 1 | Видача та узгодження теми проєкту | 08.11.2021 | 1 |
| 2 | Огляд стану проблеми та постановка задачі | 01.09-14.09 | 2 |
| 3 | Аналіз літератури за напрямком магістерської роботи | 14.09-21.09 | 3 |
| 4 | Аналіз методу випадкового лісу | 21.09-28.09 | 4 |
| 5 | Аналіз методу один проти всіх | 21.09-28.09 | 5 |
| 6 | Розробка тестової нейронної мережі | 28.09-12.10 | 6 |
| 7 | Експериментальні дослідження | 12.10-02.11 | 7 |
| 8 | Підготовка графічного матеріалу | 23.11-07.12 | 8 |
| 9 | Перевірка виконаного проєкту керівником | 10.12.2022 | 9 |
| 10 | Захист проєкту | 22.12.2022 | 10 |

Дата видачі завдання 07 листопада 2022 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Аксак Н.Г.

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: с.73, 15 рис., 4 табл., 39 джерел, додаток 1

ПРОГНОЗУВАННЯ, РЕЗУЛЬТАТИ СПОРТИВНИХ ЗМАГАНЬ, НЕЙРОННА МЕРЕЖА, МЕРЕЖА LSTM

Метою кваліфікаційної роботи є розробка інтелектуальної системи для прогнозування результатів спортивних змагань.

Об'єктом дослідження є системи для прогнозування результатів спортивних змагань

Предмет дослідження – методи прогнозування на основі нейронних мереж.

У ході виконання кваліфікаційної роботи проводився аналіз методів випадковий ліс, один проти всіх, наївного Байєса. Проаналізовано архітектури та типи нейронних мереж. Використана модель LSTM на основі уваги для прогнозування результатів матчу. Розроблений сервіс для прогнозування результатів з використанням нейронної мережі та перевірений на вирішення поставлених цілей. Представлені результати експериментальних досліджень, які підтвердили ефективність запропонованих методів прогнозування.

ABSTRACT

Explanatory note of attestation work: 73 pages, 15 figures, 4 tables, 39 sources.

FORECASTING, RESULTS OF SPORTS COMPETITIONS, NEURAL NETWORK, NETWORK LSTM

The purpose of the qualification work is to develop an intelligent system for predicting the results of sports competitions

The object of research is systems for predicting the results of sports competitions

In the course of the qualification work, an analysis of random forest, one against all, naive Bayes methods was carried out. Architectures and types of neural networks are analyzed. A service for predicting results using a neural network has been developed and tested for solving the set goals. The results of experimental studies are presented, which confirmed the effectiveness of the proposed forecasting methods.

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління
Кафедра Комп'ютерних інтелектуальних технологій та систем

АНОТАЦІЯ

КВАЛІФІКАЦІЙНОЇ РОБОТИ

рівень вищої освіти другий (магістерський)

Інтелектуальна система для прогнозування
результатів спортивних змагань

Виконав:

студент 2 курсу, групи КІТм-21-2

Овчаренко Є.С.

Спеціальність 123 Комп'ютерна інженерія

Тип програми освітньо-професійна

Освітня програма Комп'ютерні інтелектуальні
технології

Керівник проф. Аксак Н.Г.

2022 р.

АНОТАЦІЯ

Овчаренко Є.С. Інтелектуальна система для прогнозування результатів спортивних змагань. – Магістерська кваліфікаційна робота.

Актуальність теми дослідження. Інтелектуальні системи підтримки прийняття рішень стають корисними для практичних і важливих застосувань і використовують різноманітні методи штучного інтелекту. Застосування варіюються від підтримки охорони здоров'я до прийняття бізнес-рішень, і всі вони покращують процес прийняття рішень людиною.

Оскільки індустрія ставок на спорт і технології набули значних масштабів, прогнозування результату спортивного матчу з використанням технологічного підходу зараз має вирішальне значення. В реальності людина має певні обмеження при обробці великого масиву інформації. Однак технології штучного інтелекту дозволяють подолати цю проблему. Крім того, спорт має величезну кількість даних, які потрібно враховувати, а отже, є чудовим прикладом проблеми ШІ. У цій статті представлено огляд деяких досліджень з використанням різних методів ШІ для прогнозування спортивних результатів. Були проаналізовані різні види спорту, такі як футбол, метання списа, баскетбол і кінні перегони, і продемонстровані різні підходи до прогнозування результатів. Нарешті, запропоновано основу для розробки системи прогнозування результатів футбольних матчів на основі штучного інтелекту з урахуванням поточного огляду досліджень. Спорт є частиною людського життя протягом тисячоліть, і інтерес бути чимось більшим, ніж просто глядачем, повертає нас до більш ніж 2000 років тому, коли греки ходили до Колізею, щоб зробити ставку на обраного гладіатора. З того часу зростає інтерес до спроб прочитати майбутнє в спортивних матчах. Наприклад, коментатори на телебаченні показують, як вони вгадують, яка команда виграє Суперкубок, або намагаються визначити результат матчу Прем'єр-ліги, люди в соціальних мережах обговорюють, хто виграє Чемпіонат світу з тенісу. Це частина людського життя. Крім того, індустрія азартних ігор розрослася настільки, що до 2017 року загальний валовий

дохід гральної індустрії склав \$17,8 млрд. Інтерес до отримання результатів спортивних матчів до їх завершення зараз очевидний і незамінний для індустрії спортивних азартних ігор.

В магістерській роботі досліджується науково-прикладна задача створення інтелектуальної системи прогнозування закінчення спортивних змагань для використання в розважальному бізнесі.

Об'єкт дослідження - системи прогнозування результатів спортивних змагань

Предмет дослідження - методи прогнозування на основі нейронних мереж.

Дослідження базується на системному аналізі результатів сучасних теоретичних і прикладних розробок вітчизняних і зарубіжних вчених в галузі ІТ. Для вирішення поставлених завдань використано: методи системного аналізу, методи побудови нейронних мереж, методи побудови інтелектуальних обчислювальних систем, методи об'єктно-орієнтованого програмування, методи побудови програмних додатків з командним інтерфейсом.

Метою даної роботи є розробка проекту інтелектуальної системи розрахунку результатів спортивних змагань, здатної забезпечити кожному користувачеві розрахунок результатів з високою точністю. Вимоги до системи такі:

- аналіз баз даних з великим обсягом інформації з минулими подіями та результатами;
- розрахунок результатів за різними методиками;
- аналіз точності результатів при розрахунку за різними методиками;
- для надання користувачам максимально точного результату.

У першому розділі "Інтелектуальні системи підтримки прийняття рішень стають корисними для практичних і важливих застосувань і використовують різноманітні методи штучного інтелекту". Застосування варіюється від підтримки охорони здоров'я до прийняття бізнес-рішень, і всі вони покращують процес прийняття рішень людиною.

У цьому підрозділі розглядаються основи найбільш поширених методів штучного інтелекту та їх внесок у прийняття рішень. Штучні нейронні мережі - це сукупність тісно пов'язаних між собою процесорних одиниць, які називаються

нейронами, що працюють разом для вирішення проблеми. Нейронні мережі були натхненні тим, як мозок обробляє інформацію, і зазвичай складаються з шарів нейронів. Перевагою НМ є їх здатність представляти будь-яку обмежену неперервну функцію з будь-якою малою похибкою апроксимації. Дві найпоширеніші топології НМ - це мережі прямого поширення та рекурентні мережі або мережі зі зворотним зв'язком. Як випливає з назви, в нейронній мережі прямого поширення сигнали проходять від входів до виходів і, можливо, через кілька прихованих шарів. Наприклад, на рисунку 1.3 показано чотири входи, які проходять через прихований шар, що складається з трьох вузлів, до вихідного шару. ШНМ прямого поширення найчастіше використовуються в задачах прийняття рішень, оскільки потік інформації узгоджується з процесом прийняття рішень. Пряма нейронна мережа може мати сигнали, що рухаються в обох напрямках.

НМ відрізняється від послідовних, логічних підходів, які припускають певний зв'язок між входами та виходами. Їх фундаментальною перевагою є те, що вони можуть природним чином відображати нелінійність у процесі зважування. НМ пропонують підтримку для вирішення важливих проблем, таких як виявлення шахрайства, які майже неможливо змоделювати за допомогою логічних підходів. З іншого боку, НМ не підходять для таких операцій, як обробка даних.

Нейронні мережі "вивчають" основну функцію в даних, використовуючи одну з трьох стратегій з "навчальними даними": неконтрольоване, контрольоване або навчання з підкріпленням. Неконтрольоване навчання відбувається, коли НМ отримують лише вхідні дані і не мають відповідних вихідних даних. Метою є визначення базової структури даних. При контрольованому навчанні НМ отримує вхідні дані та відповідні вихідні дані, щоб вона намагалася відрегулювати ваги різних вхідних даних таким чином, щоб вихідні дані, які виробляє НМ, були в межах бажаної похибки апроксимації вихідних даних вибірки (цілей). Визначення ваг вхідних даних дозволяє використовувати НМ для прогнозування з новим набором вхідних даних. Однак у багатьох практичних ситуаціях детальні дані п-кортежів "вхід-вихід" є недоступними, і може бути лише обмежена або навіть рідкісна кількість виходів після великої кількості входів.

У другому розділі розглянуто основні методи, що використовуються для розрахунку закінчення спортивних змагань. Досліджено типи нейромережових моделей. На основі цих досліджень були обрані найбільш оптимальні моделі - випадковий ліс, один проти всіх, алгоритм Байєса. Розглянуто методи навчання нейронної мережі. На основі цих досліджень були визначені основні умови, завдяки яким можна навчити інтелектуальну систему. В результаті була побудована нейромережева модель на основі трьох алгоритмів, які показали найвищу точність.

Третій розділ присвячений спортивним матчам, які користуються великою популярністю у всьому світі. Прогноз на спортивний матч допомагає вчасно зрозуміти стан команди і скорегувати стратегію по ходу матчу. Прогнозування спортивного матчу є складним завданням. Тому досліджується метод передбачення результату наступного матчу з використанням даних попереднього матчу. Для прогнозування результату спортивного матчу було використано поєднання моделі довготривалої короткочасної пам'яті (LSTM) з механізмом уваги. Для забезпечення своєчасності прогнозу було додано ковзаюче вікно, що дозволило покращити прогноз.

Четвертий розділ присвячено експериментальним дослідженням. Моделювання нейронної мережі проводилося в середовищі PhyCarm.

Всі завдання були виконані, після чотирьох проходів точність результатів була найвищою.

ПРОГНОЗУВАННЯ, СПОРТИВНІ РЕЗУЛЬТАТИ, НЕЙРОННА МЕРЕЖА, LSTM NETWORK

Публікації здобувача за темою роботи:

1. Аксак Н. Г., Сердюк Н. М., Овчаренко Є. Модель системи дистанційного моніторингу та прогнозування стану здоров'я працівника в процесі виробничої діяльності. *Інформаційні технології та системи. Матеріали Міжнародної науково-практичної конференції: тези доповідей, 9-10 квітня 2020 р. - Х.: ХНУМГ ім. О. М. Бекетова, 2020.*

2. Овчаренко Є. Прийняття рішень з використанням мультиагентних систем. *Методи та засоби обчислювального інтелекту: матеріали XXIV Міжнародного молодіжного форуму, Харків, 7-9 квітня. 2020 р. Харків, 2020. С. 173-174.*

3. Аксак Н., Овчаренко Є. Огляд мультиагентних систем для електронного навчання. *Інформаційні технології в сучасному світі: дослідження молодих вчених. Матеріали Міжнародної науково-практичної конференції молодих вчених, аспірантів та студентів: тези доповідей, 17 - 18 лютого 2022 р. - Х.: ХНЕУ ім. С. Кузнеця, 2012.*

Список праць наукового керівника за темою магістерської роботи:

1. Сучасні інформаційні технології та системи: монографія / В. П. Бурдаєв, Н. Г. Аксак, М. В. Кушнар'єв та ін.; за ред. В. С. Пономаренка. - Харків: Видавничий дім "Стиль", 2021. - 182 с...

ЗМІСТ

| | |
|--|----|
| Перелік умовних позначень, символів, одиниць, скорочень і термінів | 13 |
| Вступ | 14 |
| 1 Аналіз предметної області та постановка задачі дослідження | 16 |
| 1.1 Методи штучного інтелекту, які використовуються | 16 |
| 1.2 Штучні нейронні мережі для підтримки інтелектуальних рішень | 16 |
| 1.3 Нечітка логіка для підтримки інтелектуальних рішень | 18 |
| 1.4 Експертні системи для підтримки інтелектуальних рішень | 20 |
| 1.5 Еволюційні обчислення для підтримки інтелектуальних рішень | 21 |
| 1.6 Інтелектуальні агенти для підтримки інтелектуальних рішень | 22 |
| 1.7 Мультиколінеарність | 24 |
| 1.8 Коефіцієнт варіації інфляції (VIF) | 26 |
| 1.9 Огляд літератури | 27 |
| 1.10 Постановка проблеми дослідження | 30 |
| 2 Обґрунтування вибору технологій прогнозування результатів матчів | 31 |
| 2.1 Огляд нейромережевих моделей | 31 |
| 2.2 Хі-квадрат тест для машинного навчання | 33 |
| 2.3 Наївні байєсівські класифікатори | 34 |
| 2.3.1 Теорема Байєса | 36 |
| 2.4 Алгоритм випадкового лісу | 36 |
| 2.6 Дерево рішень | 37 |
| 2.6.1 Класифікатор випадкових лісів | 37 |
| 3 Модель прогнозування спортивних матчів з використанням LSTM | 39 |
| 3.1 Мережа LSTM | 39 |
| 3.2 Механізм уваги | 42 |
| 3.3 LSTM модель для прогнозування спортивних результатів | 44 |
| 4 Експериментальні дослідження | 48 |

| | |
|---|----|
| 4.1 Підготовка даних | 48 |
| 4.2 Опис умов проведення експериментальних досліджень | 50 |
| 4.3 Аналіз даних | 51 |
| 4.4 Навчання та тестування моделей | 54 |
| Висновки | 58 |
| Список використаних джерел | 59 |
| Додаток А | 63 |

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І
ТЕРМІНІВ

| | |
|-------|--|
| ЕС | Експертна система |
| ІА | Інтелектуальний агент |
| NM | Нейронна мережа |
| ОПР | Особа, яка приймає рішення |
| Ш | Штучний інтелект |
| FF | Feed-Forward Neural Network, нейронна мережа з прямим зв'язком |
| HTTP | HyperText Transfer Protocol, протокол передачі даних |
| MAS | Мультиагентні системи, мультиагентна система |
| RNN | Рекурентні нейронні мережі, рекурентні нейронні мережі |
| ШКІРК | мережевий протокол сеансового рівня моделі OSI, який дозволяє |
| И | передавати пакети від клієнта до сервера через проксі-сервер |
| VIF | відхилення темпів інфляції |
| LSTM | мережа довготривалої короткочасної пам'яті |

ВСТУП

Оскільки індустрія ставок на спорт і технології набули значних масштабів, прогнозування результату спортивного матчу з використанням технологічного підходу зараз має вирішальне значення. В реальності людина має певні обмеження при обробці великого масиву інформації. Однак технології штучного інтелекту дозволяють подолати цю проблему. Крім того, спорт має величезну кількість даних, які потрібно враховувати, а отже, є чудовим прикладом проблеми ШІ. У цій статті представлено огляд деяких досліджень з використанням різних методів ШІ для прогнозування спортивних результатів. Були проаналізовані різні види спорту, такі як футбол, метання списа, баскетбол і кінні перегони, і продемонстровані різні підходи до прогнозування результатів. Нарешті, запропоновано основу для розробки системи прогнозування результатів футбольних матчів на основі штучного інтелекту з урахуванням поточного огляду досліджень. Спорт є частиною людського життя протягом тисячоліть, і інтерес бути чимось більшим, ніж просто глядачем, повертає нас до більш ніж 2000 років тому, коли греки ходили до Колізею, щоб зробити ставку на обраного гладіатора. З того часу зростає інтерес до спроб прочитати майбутнє в спортивних матчах. Наприклад, коментатори на телебаченні показують, як вони вгадують, яка команда виграє Суперкубок, або намагаються визначити результат матчу Прем'єр-ліги, люди в соціальних мережах обговорюють, хто виграє Чемпіонат світу з тенісу. Це частина людського життя. Крім того, індустрія азартних ігор розрослася настільки, що до 2017 року загальний валовий дохід гравальної індустрії склав \$17,8 млрд. Інтерес до отримання результатів спортивних матчів до їх завершення зараз очевидний і незамінний для індустрії спортивних азартних ігор. Наприклад, технології розширилися в геометричній прогресії і вивели нас на новий рівень розуміння, а штучний інтелект змінює те, як ми сприймаємо спортивні прогнози.

Алгоритми штучного інтелекту охоплюють кілька галузей інформатики. Деякі приклади: розпізнавання образів, системи прогнозування, логічний висновок та

аналіз даних. Останні кілька років були дуже важливими для технологій машинного навчання, вони мали агресивне розширення своєї точності, і зараз штучні нейронні мережі можуть перевершити людину в багатьох сферах. Однією зі сфер, в якій комп'ютери здолали людину, є прогнозування, наприклад, багато людей використовують цей інструмент для передбачення спортивних результатів. Оскільки спорт має надзвичайно велику кількість характеристик, які безпосередньо пов'язані з результатом, людині складно врахувати всі особливості і передбачити спортивний матч з високою точністю. У таких ситуаціях потрібна високопродуктивна машина для роботи з усіма даними, і саме тут на допомогу приходить штучний інтелект.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

1.1 Методи штучного інтелекту, які використовуються

Інтелектуальні системи підтримки прийняття рішень служать практичним та важливим додатком та використовують різні методи штучного інтелекту. Програми варіюються від медичної підтримки до прийняття бізнес-рішень, і всі вони покращують процес прийняття рішень людиною.

У цьому підрозділі розглядаються основи найбільш поширених методів штучного інтелекту та їх внесок у прийняття рішень.

1.2 Штучні нейронні мережі для підтримки інтелектуальних рішень

Штучна нейронна мережа є набір тісно пов'язаних блоків обробки, званих нейронами, які працюють разом для вирішення проблеми. Нейронні мережі засновані на тому, як мозок обробляє інформацію, і зазвичай складаються з шарів нейронів, як показано на рис. 1.1. Перевага НМ у цьому, що може представляти обмежені безперервні функції з малими помилками апроксимації.

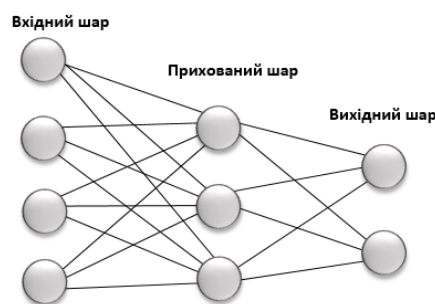


Рисунок 1.1 - Нейронна мережа з прихованим шаром

Основною одиницею нейронної мережі є нейрон чи вузол. Кожен нейрон отримує вхідний сигнал як стимулу з відповідним зважуванням. Це відносна важливість від іншого нейрона або зовнішнього джерела. Вага може бути позитивною або негативною, збудливою або гальмівною. Нейрон обчислює

виважену суму всіх вхідних сигналів нейрона. Тобто для нейрона. де - Рівень активації нейрона (формула 1.1).

$$Y_j = \sum_i x_i w_{ij} \quad (1.1)$$

Інформація може бути змінена та передана від нейронів з використанням декількох різних типів передавальних функцій, які можуть бути нелінійними. Ваги важливі для навчання ЯМ, тому що вони коригуються в міру надходження нових даних. Як правило, NM ітеративно піддаються набору навчальних даних, що містить фактичні відповідні вихідні дані, створюючи набір терезів до тих пір, поки вихідні дані NM не відповідатимуть фактичному результату з бажаним рівнем точності. Потім NM можна використовувати для прогнозування майбутніх станів з використанням набору вхідних даних або "навчитися" коригувати ваги за наявності додаткового набору вхідних/вихідних даних. Ця функція дозволяє нейронним мережам знаходити закономірності, узагальнювати з урахуванням попередніх випадків і класифікувати з урахуванням спостережень. Це схоже на те, як люди використовують емпіричні спостереження, щоб вивести наслідки з минулої історії або поведінки, що спостерігається.

Двома найбільш поширеними топологіями NM є мережі з прямим поширенням та рекурсивні мережі або мережі зі зворотним зв'язком. Як впливає з назви, в нейронних мережах з прямим зв'язком сигнали проходять від входу до виходу, можливо, через кілька прихованих шарів. Наприклад, на рис. 1.3 показані чотири входи, що проходять через прихований шар, що складається з трьох вузлів для досягнення вихідного шару. ІНС із зворотним зв'язком найчастіше використовуються у завданнях прийняття рішень, тому що потік інформації узгоджується з процесом прийняття рішень. У нейронній мережі з прямим зв'язком

сигнали можуть поширюватися в обох напрямках.

NM відрізняється від послідовних логічних підходів, які передбачають фіксований зв'язок між входами та виходами. Їхня ключова перевага полягає в тому, що процес коригування вагових коефіцієнтів природним чином відображає нелінійність. NM підтримує критично важливі завдання, такі як виявлення шахрайства, які неможливо змоделювати за допомогою логічного підходу. З іншого боку, NM не підходить для таких операцій як обробка даних.

Нейронні мережі «вивчають» основні характеристики даних, використовуючи одну з трьох стратегій: ненавчене, навчене та навчання з підкріпленням. Неконтрольоване навчання відбувається, коли NM отримує лише вхідні дані та відповідні вихідні дані. Ціль полягає в тому, щоб визначити базову структуру даних. При навчанні з учителем NM отримує вхідні дані та відповідні вихідні дані, тому він намагається відрегулювати ваги різних вхідних даних так, щоб вихідні дані, створювані NM, знаходилися в межах бажаної помилки апроксимації вибіркового (цільового) вихідного сигналу. Як тільки вага вхідних даних визначена, NM можна використовувати для прогнозування нового набору вхідних даних. Однак у багатьох практичних ситуаціях не можна використовувати докладний набір з n входів і виходів, і за багатьма входами може бути обмежена чи рідкісна кількість виходів. Ми використовуємо навчання з підкріпленням, щоб вирішити цю ситуацію, і надаємо зворотній зв'язок NM, щоб оцінити, чи правильно вибрані ваги. При навчанні NM завданням прийняття рішень слід уникати перенавчання (або перенавчання на навчальних даних), щоб NM можна було узагальнити та використовувати з метою прогнозування.

1.3 Нечітка логіка для підтримки інтелектуальних рішень

Нечітка логіка розширює можливості підтримки прийняття рішень, дозволяючи представляти вхідні дані або змінні проблеми прийняття рішень у вигляді людських висновків. Особи, які приймають рішення, часто стикаються з проблемами, коли вхідні дані є неточними або невизначеними. Наприклад, погода

може бути сонячною, частково сонячною, переважно хмарною чи похмурою. Булева логіка, з іншого боку, являє собою систему символічної логіки, яка управляє логічними функціями на комп'ютерах і заснована на двійковій системі 0 (цілком неправильно) і 1 (цілком вірно). Нечітка логіка враховує невизначеність, задаючи вхідний діапазон від 0 (цілком неправильно) до 1 (повністю правильно). Нечітка логіка допомагає в ISPRS, надаючи:

- Гнучкість, що дозволяє враховувати несподіване.;
- Інтуїтивні варіанти, такі як "ймовірно" або "дуже добре";
- Здатність представляти сценарії "що, якщо";
- Низький ризик неправильного вибору, тому що є діапазон можливих значень вибору;
- Підхід до моделювання проблем із невизначеностями, які важко висловити в математичних моделях.

Нечітка логіка гнучкіша у своєму вираженні, надаючи особам, що приймають рішення, широкий вибір та свободу для оцінки цінності вхідних даних. Оскільки нечітка логіка не має внутрішньої структури, вона може природно інкапсулювати нелінійні відносини без попереднього планування. Значення можна легко уточнювати та змінювати у міру надходження нової інформації, що дозволяє особам, які приймають рішення, природно справлятися з невизначеністю. Нечітка логіка надає спосіб вираження заснованого на правилах поведінки, такого як досвід, який дозволяє накопичувати досвід і робити його доступним для осіб, які приймають рішення, у потрібний час. Нечітка логіка також може бути з'єднана з NM, щоб зробити інтерпретацію змінних рішення більш ясною. Наприклад, вхідна змінна може бути представлена трійками, такими як максимум, мінімум та максимальна ймовірність. Це описи природною мовою, які покращують здатність осіб, що приймають рішення, передавати знання про предметну область моделі та інтерпретувати вихідні дані.

Нейронні мережі з нечіткою логікою - це категорія моделей прийняття рішень, які можуть забезпечити багаторівневі мережі прямими зв'язками зі зрозумілим для людини значенням. Сукупні нейрони виконують логіку АБО, тоді як опорні нейрони

підтримують операції, засновані на таких предикатах, як "менше ніж", "більше ніж" і "рівно", виражених мовою нечіткої логіки. Нечітка логіка ІНС допомагає подолати деякі недоліки ІНС з погляду прозорості та покращити процес прийняття рішень.

1.4 Експертні системи для підтримки інтелектуальних рішень

Експертна система (ЕС) — це комп'ютерна система, яка намагається вирішити проблеми, які зазвичай вирішують експерти-люди. Термін ЕС часто використовується для опису системи, яка містить інформацію одного чи кількох призначених експертів. Розробники систем повинні дізнатися, як люди-експерти приймають рішення, та впровадити ці знання у комп'ютерні системи.

Компоненти ЕС показані малюнку 1.2. Він показує експертів у предметній галузі, які роблять свій внесок у модуль придбання знань. Ці знання зазвичай кодуються основою знань як частину процесу розробки.

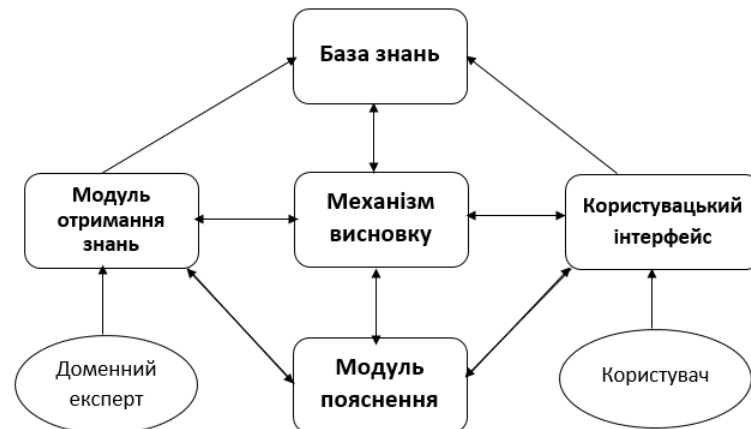


Рисунок 1.2 - Складові експертної системи

Користувач або особа, яка ухвалює рішення, входить до системи через інтерфейс. Потім користувачі можуть отримати доступ до бази знань минулих випадків або механізму логічного висновку для отримання нових випадків попередніх випадків. Користувач може «деталізувати» та пояснити висновки модуля пояснення. Таким чином, ES відповідає за отримання, збирання та уточнення знань

від експертів у предметній галузі та передачу цих знань особам, які приймають рішення.

1.5 Еволюційні обчислення для підтримки інтелектуальних рішень

Еволюційні обчислення натхненні природною еволюцією, у ході якої популяції еволюціонують, щоб стати більш придатними задля досягнення мети, цим збільшуючи виживання. Методи штучного інтелекту намагаються імітувати ці властивості адаптації до довкілля, моделюючи появу, виживання та поліпшення популяції. Генетичні алгоритми є одними з найбільш широко використовуються для вирішення задач. Як тільки популяція ініціалізована, наступні покоління взаємодіють, спілкуються та впливають один на одного, щоб краще адаптуватися до навколишнього середовища.

Узагальнена блок-схема генетичного алгоритму показано малюнку 1.3.

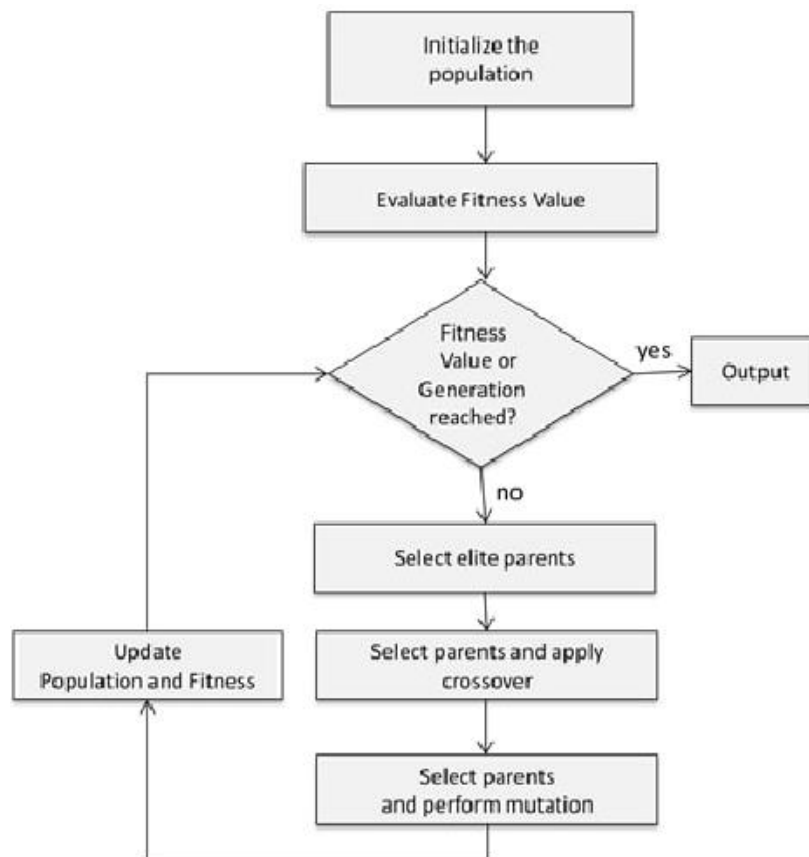


Рисунок 1.3 - Узагальнена структурна схема генетичного алгоритму

Кінцева популяція ініціалізується випадковим чином і отримує ціль за раз. Кожна особистість оцінюється за цілями та характеризується величиною адаптації, що відображає пристосованість особистості до навколишнього середовища. Чим вища пристосованість, тим більша ймовірність того, що людина виживе і стане батьком майбутніх поколінь. Слабкі люди також можуть стати батьками. Населення продовжує зростати до тих пір, поки або не буде досягнуто критерій зупинки, або не буде досягнуто кількості поколінь. Згодом населення стане більш здоровим та цілеспрямованим. Для поліпшення популяцій використовуються два методи: розмноження та мутація. Розмноження вимагає, щоб люди обмінювалися частинами свого коду, щоб зробити потомство. Мутації викликають невеликі зміни деяких компонентах коду людини.

1.6 Інтелектуальні агенти для підтримки інтелектуальних рішень

Серед різних методів штучного інтелекту на вирішення завдань найширше використовуються інтелектуальні агенти (II). За визначенням, агент — це об'єкт у системі, який «перебуває у середовищі і може діяти автономно у цьому середовищі задля досягнення мети». Ця невід'ємна здатність діяти автономно є відмінною рисою людської поведінки, заснованої на контексті (або прийняття рішень), а не логіки, що вказує на комп'ютерні програми «що, якщо». У літературі використовуються концепції слабого агентства та сильного агентства, щоб розрізнити агентів та інтелектуальних агентів. До слабких агентів належать такі навички, як автономія, реактивність, адаптивність, агресія та соціальні навички. Сильні агенти додають до цих навичок більш просунуті риси, такі як спілкування, наполегливість, мобільність, раціональність та здатність до навчання. Автономія – це здатність ухвалювати рішення незалежно від інших. Реактивні та адаптивні агенти сприймають навколишнє середовище та реагують на його зміни. Агресія означає, що NAR можуть діяти для досягнення своєї мети після отримання конкретних інструкцій.

Соціальні функції дозволяють агентам взаємодіяти з іншими. Спілкування з іншими агентами представлено соціальними здібностями агента, що призводить до таких рис, як співробітництво та переговори. Постійність означає, що агент зберігає свій стан протягом тривалого періоду часу, а мобільність дозволяє агенту переміщатися системою для виконання завдань або пошуку нових знань. Раціональність - це здатність приймати рішення на основі навколишнього середовища, а навчання означає, що НП можуть змінювати свої реакції на основі минулих взаємодій та ситуацій. Людські терміни, такі як знання, наміри та переконання, використовувалися для опису П, показуючи, що він здатний до складної поведінки. Ідея полягає в тому, щоб створити багатоагентну систему (MAS), що складається з агентів, які не мають повної інформації про свої середовища чи інших агентів. Кожен МА може мати певні характеристики, і окремі агенти можуть поєднуватися з іншими агентами, щоб діяти від імені осіб, які приймають рішення, або інших агентів з іншими цілями та завданнями. Агентам може знадобитися взаємодіяти, координувати свої дії, вести переговори, навчатися і навіть довіряти іншим агентам для досягнення своїх цілей. Особа, яка приймає рішення, може отримувати інструкції від IAS або уповноважувати агента діяти від імені особи без спеціального дозволу. MAS – це активна галузь досліджень у галузі складних додатків, що охоплює питання, пов'язані з поданням проблем прийняття рішень, розміром команди та реструктуризацією команди під час вирішення проблем.

Динамічна та невизначена середовище особливо підходить для MAS. Команди агентів можуть сприймати своє середовище, адаптуватися до змін, ділитися знаннями між членами команди та розробляти узгоджені плани дій. Агенти можуть навчатися під наглядом, без нагляду або за винагороду. Подібно до NM, навчання з учителем надає системі правильні вхідні дані для управління нею. При неконтрольованому навчанні вихідні дані не надаються, тому система має визначати структуру вхідних даних. Під час навчання з підкріпленням ІНС отримує позитивне або негативне підкріплення, яке приводить у рух систему.

Основні поняття агентної технології:

Спілкування необхідне передачі намірів АН. Зв'язок може бути прямим або

непрямим. Прямий зв'язок може здійснюватися шляхом передачі повідомлень або загальних змінних. Непряма комунікація може відбуватися шляхом логічного висновку, наприклад коли агенти спостерігають за діями інших агентів.

Організація АН, її ресурси та завдання потребують коригування. Узгодження використовується для вирішення конфліктів та підвищення ефективності агента.

Спільна організація необхідна для кодифікації поведінки, такої як спілкування та координація між НА, щоб вони могли взаємодіяти та діяти як команда для досягнення спільних цілей.

Агенти, орієнтовані на людину, необхідні повної взаємодії між комп'ютерними системами та особами, які приймають рішення. Наприклад, особам, які приймають рішення, можуть бути представлені підзавдання, що вимагають людського судження.

Навчання необхідно, щоб зробити ХА більш гуманним та полегшити взаємодію між ІАС та особами, що приймають рішення.

1.7 Мультиколінеарність

Статистичні методи кількісно оцінюють тест та визначають, чи є значні ефекти серед незалежних змінних (керовані та модифіковані змінні — уявіть елементи на осі X графіка) і як вони впливають на залежну змінну. Це відмінний інструмент, щоб побачити, чи це впливає ти. (Змінні, що змінюються при зміні незалежної змінної елемента осі Y графіка).

Наприклад, $y=2x$ означає, що y змінюється на 2 (зміна залежної змінної) при кожній зміні x (незалежної змінної). Це дуже просто, якщо у вас є одна незалежна змінна для однієї залежної змінної. Однак, коли ваш набір даних містить кілька змінних, які називаються функціями, які можуть впливати на вашу цільову змінну (змінну, яку ви намагаєтеся передбачити за допомогою лінійної регресії), все стає трохи складніше.

Подальші складнощі виникають, коли атрибути залежать один від одного. Іншими словами, якщо ви зміните незалежну змінну та очікуєте, що зміниться

залежна змінна, ви виявите, що інші незалежні змінні також змінюються. Ці дві незалежні змінні співзалежні чи колінеарні один одному. Додавання більшої кількості об'єктів, колінеарних один одному, дає мультиколінеарність.

Мультиколлінеарність виникає, коли незалежні змінні моделі регресії корелюють. Існує два основних типи мультиколлінеарності. Перший — структурний (незалежна змінна x зводиться у квадрат), який є просто побічним продуктом, який майже завжди створюється з використанням існуючих незалежних змінних, тому його можна відстежувати. Я маю набір даних, і я хотів би використовувати журнали для масштабування або нормалізації всіх функцій. Це приклад структурної мультиколлінеарності. Другий небезпечніший, на мій погляд, мультиколлінеарність даних. Це вже вбудовано в набір функцій кадру даних `pandas`, і його набагато складніше спостерігати.

Які проблеми це може викликати. Це може зменшити загальні коефіцієнти та p -значення (звані значеннями значущості) та викликати несподівані відхилення. Це може призвести до переоснащення, коли модель відмінно працює на відомому навчальному наборі, але не працює на невідомому наборі тестів. Мультиколлінеарність призводить до високої стандартної помилки навіть за низького значення статистичної значущості, що ускладнює визначення важливості ознаки для цільової змінної. Невелике значення значущості також унеможлиблює відкидання нулів, що призводить до помилки 2-го роду під час перевірки гіпотез.

Наступне питання, яке мені поставили під час дискусії, було таким же, як і Ваше. Як ми їх виявляємо? Ну, найкращий спосіб, який я знайшов, це перевірити кожен незалежну функцію одна проти одної. Це чудово і не так багато роботи, якщо у вас обмежені можливості для роботи, що я і зробив для одного з моїх проєктів, який використовував лінійну базову модель. Але коли у вас є сотні і більше елементів, це набагато складніше. На цьому етапі виникає проблема розмірності (відома як прокляття розмірності). Найкраще на цьому етапі використовувати PCA (аналіз головних компонент), щоб зменшити кількість ознак.

Інший метод виявлення - подивитися на коефіцієнт.

Якщо ви можете визначити, на які змінні впливає мультиколлінеарність та силу

кореляції, ви вже на шляху до визначення того, чи потрібно її коригувати. На щастя, існує дуже простий тест для оцінки мультиколінеарності у вашій регресійній моделі. Дисперсійний коефіцієнт інфляції (VIF) вимірює кореляцію між незалежними змінними та силу цієї кореляції.

За допомогою дисперсійного коефіцієнта інфляції (VIF) можна визначити, чи є дві незалежні змінні колінеарними одна до одної. При вимірюванні, якщо VIF двох змінних дорівнює 1, то вони не є колінеарними одна до одної (тобто кореляція між двома змінними відсутня). Однак, чим більше число, тим більше вони корелюють між собою. Якщо VIF повертає число більше 5, дві ознаки повинні бути зведені до однієї за допомогою ПКА.

Існує ще один останній статистичний інструмент, який чудово підходить для аналізу різниці між функціями. Це ANOVA, що означає дисперсійний аналіз. Як правило, чим вища дисперсія між змінними, тим менша ймовірність того, що вони пов'язані між собою (або корелюють). Але, можливо, це буде темою для подальшої дискусії. Нижче я надам посилання на своє джерело для цієї дискусії, яке також містить чудовий приклад, який ви можете спробувати.

1.8 Коефіцієнт варіації інфляції (VIF)

Для виявлення наявності мультиколінеарності використовується дисперсійний коефіцієнт інфляції (VIF). Дисперсійні коефіцієнти інфляції (VIF) вимірюють, наскільки дисперсія оцінених коефіцієнтів регресії переоцінена порівняно з тим, коли предикторні змінні не є лінійно пов'язаними.

Його отримують шляхом регресії кожної незалежної змінної, скажімо X, на решту незалежних змінних (скажімо Y та Z) і перевірки того, наскільки вона (X) пояснюється цими змінними.

$$vif = \frac{1}{(1 - R \cdot R)} \quad (1.2)$$

З формули (1.2) видно, що чим вище VIF, тим вище R², а це означає, що змінна X колінеарна зі змінними Y та Z. Якщо всі змінні повністю ортогональні, то R² буде дорівнювати 0, в результаті чого VIF дорівнюватиме 1.

1.9 Огляд літератури

В роботі [1] розглядається система, реалізована на основі трирівневої архітектури сервіс-орієнтованих систем, що характеризується поєднанням розподілених методів і засобів збору, зберігання, обробки різнорідних даних.

Огляд мультиагентних систем для прийняття рішень в складних системах зроблено в [2, 3, 4].

Нейронні мережі показали хороші результати в прогнозуванні проблем. Існує багато видів нейронних мереж, і найпростішою з них є глибока нейронна мережа (DNN). В [10] запропоновано систему прогнозування футбольних матчів на основі ШНМ, яка показує, що глибоке навчання може успішно прогнозувати результати футбольних матчів або будь-яких інших спортивних подій. В роботі [11] ШНМ використовуються для обробки спортивних новин, вилучення новин, пов'язаних зі спортсменами в новинних статтях, і пропонується модель для оцінки спортивних спортсменів. Штучні нейронні мережі (ШНМ) також використовуються для прогнозування спортивних результатів. У роботі [12] вони спочатку зібрали дані про спортивні змагання, а потім використали свій дизайн ШНМ для порівняння свого методу з лінійною регресією та лінійним композитом.

Згортова нейронна мережа (Convolutional Neural Network, CNN) є одним з типових алгоритмів глибокого навчання. Вона має здатність представляти навчання та обробку багатовимірних даних. В роботі [13] була використана технологія проектування тактичної функції для стиснення простору-часу і полегшення тонкої настройки трьох попередньо навчених ШНМ. Вони використовували просторово-часову модель ШНМ для прогнозування сил реакції ґрунту та моментів спортсменів.

Крім того, в [1 4] запропоновано метод семантичної ідентифікації часу в баскетбольному відео. Вони використовували двопотокову 3D CNN-структуру для прогнозування групової активності розпізнавання режимів руху та виділяли ознаки зовнішності спортсменів за допомогою CNN-структури. Для оцінки успіху або невдачі командного змагання експеримент довів, що цей метод має найбільш досконалу продуктивність. В роботі [1 5] вони запропонували метод побудови інкрементальної трубки дій для просторово-часового контекстно-залежного виявлення та прогнозування онлайн дій. Вони використовували згорткову рекурентну нейронну мережу та модель кодера-декодера для прогнозування дій об'єкта заздалегідь. Хоча вищезазначені методи можуть працювати з багатовимірними даними, важко обробляти дані часових рядів. Однак результати спортивних змагань часто змінюються з часом. Наприклад, фізичний стан мобілізаційної дистанції, польова обстановка та інші фактори залежать від стану часу.

Рекурентна нейронна мережа також є одним з типових алгоритмів глибокого навчання. В даний час рекурентні нейронні мережі також використовуються в різних моделях прогнозування. В роботі [1 6] рекурентні нейронні мережі були використані для прогнозування пози людського тіла та оцінки безпеки людського тіла за допомогою зібраних даних датчиків руху. Цей метод допомагає людям з групи ризику. У сучасному світі спортивні ігри надають багато статистичної інформації по кожному гравцю, команді, грі та сезону [1 7]. Іншими словами, вже зараз існує багато доступних даних про спорт. На прикладі [1 8] взяли до уваги безперервність гри, послідовність дій гравців та місце розташування дій і використали CNN та RNN для прогнозування результатів гри, досягнувши хорошої точності прогнозування. На прикладі футболу, футбольна гра - це групова поведінка на шумовому фоні. У статті [1 9] багатошаровий перцептрон (MLP) та RNN були використані для прогнозування групової поведінки, щоб оцінити відвідуваність футбольного матчу. Існує багато сценаріїв, де RNN використовується для прогнозування. Наприклад, автор роботи [20] запропонував багатозадачну ШНМ з

марковськими випадковими полями високого порядку для прогнозування тренду ціни на акції. Багатозадачна структура ШНМ може досягти високої якості результатів прогнозування, витягуючи інформаційні характеристики лише з вихідних ринкових даних однієї акції без знання іншої предметної області. Крім того, в роботі [21] використано ШНМ для визначення інтересів користувачів в соціальних мережах на основі визначення місцезнаходження для прогнозування наступного об'єкту інтересу. Нейронна мережа зі зворотним поширенням та LSTM були використані для прогнозування результатів гандболісток [22]. Крім прогнозування виступу спортсменок, глибоке навчання також досягло значного прогресу в інших областях, таких як зменшення шуму зображення [23] та оптимізація затримки завантаження в периферійних обчисленнях [2 4].

Однак рекурентна нейронна мережа має і певні недоліки. Наприклад, вона не може вловити довгострокові зв'язки між історичними даними [25]. У реальному процесі навчання деяка важлива історична інформація може бути проігнорована. Тому для прогнозування даних часових рядів почали використовувати LSTM, відмінний варіант RNN, для прогнозування даних часових рядів. Щоб допомогти баскетбольній команді розробити тактику, в [2 6] запропоновано метод, що поєднує глибокий Bi-LSTM з мережею щільності суміші (MDN). Тренуючи реальну модель траєкторії баскетбольного м'яча, він допомагає тренерам і гравцям вирішувати, коли і куди кидати, щоб виграти баскетбольний матч.

Оскільки багато моделей не є досконалими, деякі стратегії оптимізації надають ідеї для вдосконалення моделей [28] , [29] , [30]. Механізм уваги став важливою концепцією в нейронних мережах і був повністю вивчений в різних областях [31, 32]. Після цього дослідники почали поєднувати механізм уваги з нейронною мережею для прогнозування проблем. В [33] запропоновано модель CNN-LSTM-BiLSTM, що базується на увазі, для короткострокового прогнозування навантаження. Враховуючи притаманний мовним сигналам часовий зв'язок, автор статті [3 4] використав LSTM на основі уваги для виділення емоційних ознак у

людській мові та класифікації голосових емоцій. Застосування механізму уваги покращує продуктивність моделі та інтерпретованість нейромережевої моделі.

Незважаючи на те, що існує багато методів прогнозування, передбачити спортивні результати все ще складно. Якщо взяти для прикладу футбольний матч, то на перемогу в матчі впливає багато факторів, наприклад, перевага домашнього поля. Команда має перевагу домашнього майданчика. Команда господарів має більше вболівальників на грі і спортсмени мають більше шансів для натхнення. Крім того, суддя може мати суб'єктивну упередженість. Команда гостей виснажена дорогою. Вони не можуть заручитися підтримкою великої кількості вболівальників, та й фізичний стан у них буде гіршим, ніж у господарів. Крім того, перемога в грі тісно пов'язана з голами і ударами команди, наприклад, удари дають можливості для голів. Чим більше голів, тим більше шансів виграти гру. Іншими словами, кількість голів пов'язана з перемогою в грі, однак це не обов'язково означає, що команда може виграти з більшою кількістю кидків або голів. Показник володіння м'ячем також може вплинути на результат гри. Навіть якщо команда забиває кілька голів, вона може виграти гру за рахунок вміння гравців контролювати м'яч. У такій ситуації прогнозування результатів спортивних змагань залишається складним завданням. Тому нам потрібно знайти спосіб передбачити результат заздалегідь, щоб команда могла заздалегідь скоригувати свою стратегію. І це може полегшити тренеру підготовку команди.

1.10 Постановка проблеми дослідження

Метою магістерської роботи є розробка інтелектуальної системи прогнозування закінчення спортивних змагань

Для досягнення мети дослідження необхідно вирішити наступні завдання:

- аналізувати архітектуру нейронних мереж;
- розробити нейронну мережу для прогнозування результатів;
- протестувати сформовану нейронну мережу на тестових даних;

- створити додаток на основі тестової НМ.

Вирішення цих завдань дозволить створити додаток на основі нейронних мереж, який можна буде використовувати для прогнозування результату на основі попередніх результатів спортивних матчів.

2 ОБҐРУНТУВАННЯ ВИБОРУ ТЕХНОЛОГІЙ ПРОГНОЗУВАННЯ РЕЗУЛЬТАТІВ МАТЧІВ

2.1 Огляд нейромережових моделей

Нейронні мережі (НС) зазвичай поділяються на два типи: зі зворотним і без. Існують нейронні мережі з прямим зв'язком (FF) без зворотного зв'язку та рекурентні нейронні мережі (RNN) або мережі зі зворотним зв'язком.

До основних типів нейронних мереж відносяться:

- багатошаровий або одношаровий перцептрон;
- згорткові нейронні мережі;
- рекурентні нейронні мережі.

- Одношаровий перцептрон - перша запропонована модель нейрона, в якій вміст локальної пам'яті нейрона складається з вектора вагових коефіцієнтів. Обчислення одношарового перцептрона виконується шляхом обчислення суми вхідних векторів із значеннями, помноженими на відповідні елементи вагових векторів.

- Багатошаровий перцептрон є нейронною мережею, в якій взаємозв'язок введення-виведення нелінійний.

- Багатошарові перцептрони мають вхідні та вихідні шари, а також один або кілька прихованих шарів, що складаються з багатьох нейронів. Перцептрони також містять нейрони, які повинні мати функцію активації, яка встановлює поріг проходження.

- Багатошарові перцептрони належать до категорії алгоритмів прямого розповсюдження (FF). Це з тим, що вхідні дані об'єднуються з початковими вагами у зважену суму і підпорядковуються функції активації. Але різниця у тому, що кожна лінійна комбінація поширюється на наступний шар.

- Кожен шар передає результати обчислень наступному шару. Тобто внутрішнє подання даних. Це відбувається по всьому шляху від прихованого шару

до вихідного шару.

- Якщо алгоритм обчислює лише зважену суму кожного нейрона, множить результат на вихідний шар і зупиняється на ньому, він не може дізнатися ваги, що мінімізують функцію вартості. Якщо алгоритм обчислюється однією ітерацією, реального навчання немає.

Нейрони у багатошарових перцептронах можуть використовувати довільні функції активації, як і одношарових перцептронах.

Функції активації:

- сигмовидної кишки;
- функція порогової активації;
- лінійна функція активації;
- гіперболічний тангенс;
- Релу;
- Дірявий РеЛю.

Згорткова нейронна мережа (CNN) - це алгоритм глибокого навчання, який бере вхідне зображення і привласнює важливість (навчені ваги та зміщення) різним аспектам або об'єктам на зображенні, щоб їх можна було відрізнити один від одного. Попередня обробка згорткових мереж набагато нижча порівняно з іншими алгоритмами класифікації.

Архітектура згорткових мереж нагадує архітектуру нейронів людського мозку. Окремі нейрони реагують на подразники лише в обмеженій області зору, відомої як рецептивне поле. Набір таких полів, що перекриваються, покриває все поле зору.

Згортки можуть успішно фіксувати просторові та часові залежності у зображеннях, застосовуючи відповідні фільтри. Ця архітектура зменшує кількість взаємопов'язаних та повторно використовуваних параметрів, щоб забезпечити кращу відповідність набору зображень.

Рекурентна нейронна мережа (RNN) – це нейронна мережа, призначена для маніпулювання текстом. Рекурентні мережі корисні вивчення безперервної структури тексту, де кожне слово залежить від попереднього слова чи слів у попередньому реченні.

Кожна осередок мережі може бути представлена як чорна скринька, яка приймає прихований стан (вектор), вектор слів, вихідний вектор та наступне значення прихованого стану. Ця комірка має деякі ваги, які необхідно скоригувати за допомогою зворотного розповсюдження помилки. Крім того, один і той же осередок застосовується до всіх слів, таким чином розподіляючи вагу за словами у реченні.

Кожну комірку мережі можна представити як чорний ящик, який приймає значення прихованого стану (вектор), вектор слова, вектор виходу і наступний прихований стан. У цій комірці є деякі ваги, які необхідно скоригувати за допомогою зворотного розповсюдження. Також ця ж комірка застосовується до всіх слів, так що ваги розподіляються на слова в реченні.

2.2 Хі-квадрат тест для машинного навчання

Поширеною проблемою в прикладному машинному навчанні є визначення відповідності вхідних ознак прогнозованому результату. Це проблема вибору ознак. У випадку проблем класифікації, де вхідні змінні також є категоріальними, ми можемо використовувати статистичні тести, щоб визначити, чи є вихідна змінна залежною або незалежною від вхідних змінних. Якщо вона є незалежною, то вхідна змінна є кандидатом на функцію, яка може не мати відношення до проблеми і може бути видалена з набору даних. Категоріальна змінна - це змінна, яка може приймати одне з багатьох значень. Прикладом може бути стать, яка може бути узагальнена як чоловіча або жіноча. Змінною або фактором є "стать", а позначеннями або рівнями змінної в цьому випадку є "чоловік" і "жінка". Можливо, ми захочемо розглянути підсумок категоріальної змінної, оскільки вона пов'язана з іншою категоріальною змінною.

Наприклад, стать та інтерес, де інтерес може бути позначений як "наука", "математика" або "мистецтво". Ми можемо збирати спостереження від людей щодо цих двох категоріальних змінних. Тест хі-квадрат Пірсона, або просто тест хі-

квадрат для стислості, названий на честь Карла Пірсона, хоча тест має варіації.

Тест χ^2 -квадрат - це перевірка статистичної гіпотези, яка передбачає (нульова гіпотеза), що спостережувані частоти для категоріальної змінної збігаються з очікуваними частотами для категоріальної змінної. Тест обчислює статистику, яка має розподіл χ^2 -квадрат, названий на честь грецької великої літери χ (X), яка вимовляється "кі", як у слові "повітряний змій". Враховуючи наведений вище приклад щодо статі/інтересу, кількість спостережень для категорії (наприклад, чоловіки та жінки) може бути однаковою, а може і не бути однаковою.

Тим не менш, ми можемо розрахувати очікувану частоту спостережень у кожній групі інтересів і побачити, чи поділ інтересів за статтю призводить до однакових чи різних частот. Тест χ^2 -квадрат робить це для таблиці непередбачених обставин, спочатку розраховуючи очікувані частоти для груп, а потім визначаючи, чи відповідає поділ груп, який називається спостережуваними частотами, очікуваним частотам.

Результатом тесту є тестова статистика, яка має розподіл χ^2 -квадрат і може бути інтерпретована як відхилення або не відхилення припущення або нульової гіпотези про те, що спостережувані та очікувані частоти збігаються. Змінні вважаються незалежними, якщо спостережувані та очікувані частоти подібні, що рівні змінних не взаємодіють, не залежать. Інтерпретувати статистику тесту в контексті розподілу χ^2 -квадрат з необхідним числом ступенів свободи можна наступним чином:

- Якщо статистичне значення \geq критичному значенню : значущий результат, відхилити нульову гіпотезу (H_0) про наявність залежності.

- Якщо статистичне значення $<$ критичного значення: незначущий результат, то нульову гіпотезу (H_0) не можна відкидати незалежно від того, яке значення має статистичне значення.

2.3 Наївні байєсівські класифікатори

Наївні байєсівські класифікатори - це сукупність алгоритмів класифікації,

заснованих на теоремі Байєса. Це не один алгоритм, а сімейство алгоритмів, всі вони мають спільний принцип, тобто кожна пара ознак, що підлягають класифікації, є незалежною одна від одної. Спочатку розглянемо набір даних. Розглянемо вигаданий набір даних, що описує погодні умови для гри в гольф. Кожен кортеж класифікує погодні умови як придатні (Так) або непридатні (Ні) для гри в гольф. Наведемо табличне представлення нашого набору даних у таблиці (2.1).

Таблиця 2.1 - Матриця ознак та вектор відгуку

| | Погода | Температура | Вітряність | Пограти в гольф | Перспектива |
|----|----------|-------------|------------|-----------------|-------------|
| 0 | Дощова | Гарячі | Високий | Брехня. | Ні. |
| 1 | Дощова | Гарячі | Високий | Правда | Ні. |
| 2 | Похмуро. | Гарячі | Високий | Брехня. | Так. |
| 3 | Сонячний | Незначний | Високий | Брехня. | Так. |
| 4 | Сонячний | Усе гаразд. | Як завжди | Брехня. | Так. |
| 5 | Сонячний | Усе гаразд. | Як завжди | Правда | Ні. |
| 6 | Похмуро. | Усе гаразд. | Як завжди | Правда | Так. |
| 7 | Дощова | Незначний | Високий | Брехня. | Ні. |
| 8 | Дощова | Усе гаразд. | Як завжди | Брехня. | Так. |
| 9 | Сонячний | Незначний | Як завжди | Брехня. | Так. |
| 10 | Дощова | Незначний | Як завжди | Правда | Так. |
| 11 | Похмуро. | Незначний | Високий | Правда | Так. |
| 12 | Похмуро. | Гарячі | Як завжди | Брехня. | Так. |
| 13 | Сонячний | Незначний | Високий | Правда | Ні. |

Набір даних поділяється на дві частини, а саме: матрицю ознак та вектор відгуку. Матриця ознак містить всі вектори (рядки) набору даних, в яких кожен вектор складається зі значень залежних ознак. У наведеному вище наборі даних ознаками є "Погода", "Температура", "Вологість" та "Вітер". Вектор відгуку містить значення змінної класу (прогноз або вихід) для кожного рядка матриці ознак. У

наведеному вище наборі даних ім'я змінної класу - Play Golf.

Стосовно до набору даних це поняття можна розуміти так: Будемо вважати, що жодна пара ознак не є залежною. Наприклад, температура "Жарко" ніяк не пов'язана з вологістю, а прогноз "Дощ" ніяк не впливає на вітер. Отже, ознаки вважаються незалежними. По-друге, кожній ознаці надається рівна вага. Наприклад, знаючи тільки температуру і вологість, не можна точно передбачити результат. Жодна з ознак не є несуттєвою і вважається, що вона в рівній мірі впливає на результат. Припущення, зроблені за допомогою наївного Байєса, зазвичай невірні в реальних ситуаціях.

2.3.1 Теорема Байєса

Теорема Байєса визначає ймовірність того, що подія відбудеться, якщо відома ймовірність іншої події, яка вже відбулася. Математично теорема Байєса формулюється у вигляді наступного рівняння:

$$P\left(A \mid (B)\right) = P\left(B \mid (A)\right) \cdot \frac{P(A)}{P(B)}$$

де A і B - події, а $P(B) \neq 0$. По суті, ми намагаємося знайти ймовірність події A за умови, що подія B істинна. Подія B також називається доказом. $P(A)$ - це апіорне значення події A (апіорна ймовірність, тобто ймовірність події до появи доказу).

Доказом є значення ознаки невідомого екземпляра (тут це подія B). $P(A|B)$ - це апостеріорна ймовірність B , тобто ймовірність події після того, як з'явився доказ.

2.4 Алгоритм випадкового лісу

Як працює алгоритм і чому він такий ефективний Більша частина машинного навчання полягає в класифікації - бажанні знати, до якого класу (також відомого як група) належить спостереження. Здатність точно класифікувати спостереження надзвичайно цінна для різних бізнес-додатків, таких як прогнозування того, чи

купити конкретний користувач продукт, або прогнозування того, чи буде дефолт за певним кредитом. Наука про дані надає багато алгоритмів класифікації, таких як логістична регресія, машина опорних векторів, наївний класифікатор Байєса та дерева рішень.

Але на вершині ієрархії класифікаторів знаходиться класифікатор випадкового лісу (існує також регресор випадкового лісу, але це тема для іншого дня). У цій статті ми розглянемо, як працюють базові дерева рішень, як окремі дерева рішень об'єднуються для створення випадкового лісу, і, зрештою, з'ясуємо, чому випадкові ліси настільки хороші в тому, що вони роблять.

2.6 Дерево рішень

Дерева класифікації - це метод, який дозволяє прогнозувати приналежність спостережень або об'єктів до певного класу категоріальної залежної змінної в залежності від відповідних значень однієї або декількох змінних-предикторів.

Якщо залежна змінна, тобто цільова змінна приймає дискретні значення, то метод дерева рішень вирішує задачу класифікації.

Якщо залежна змінна приймає неперервні значення, то дерево рішень встановлює залежність цієї змінної від незалежних змінних, тобто вирішує задачу чисельного прогнозування.

Вперше дерева рішень були запропоновані Ховелендом та Хантом наприкінці 50-х років минулого століття.

2.6.1 Класифікатор випадкових лісів

Випадковий ліс, як випливає з його назви, складається з великої кількості окремих дерев рішень, які працюють як ансамбль. Кожне окреме дерево у випадковому лісі виробляє прогноз класу, і клас, що набрав найбільшу кількість голосів, стає модельним прогнозом.

Фундаментальна концепція випадкового лісу проста, але потужна - мудрість

натовпу. З точки зору науки про дані, причина, чому модель випадкового лісу працює так добре, полягає в наступному: Велика кількість відносно некорельованих моделей (дерев), що працюють як комітет, перевершить будь-яку з моделей, що складаються з окремих компонентів. Ключовим моментом є низька кореляція між моделями.

Подібно до того, як інвестиції з низькою кореляцією (наприклад, акції та облігації) об'єднуються в портфель, який є більшим, ніж сума його частин, некорельовані моделі можуть давати ансамблеві прогнози, які є більш точними, ніж будь-які індивідуальні прогнози. Причиною цього чудового ефекту є те, що дерева захищають одне одного від своїх індивідуальних помилок (якщо вони не помиляються послідовно в одному напрямку). Хоча деякі дерева можуть помилятися, багато інших дерев будуть мати рацію, тому дерева групи можуть рухатися в правильному напрямку. Таким чином, створюються передумови для того, щоб випадковий ліс працював добре:

- У функціях повинен бути якийсь реальний сигнал, щоб моделі, побудовані з використанням цих функцій, працювали краще, ніж випадкові припущення.

- Прогнози (і, відповідно, помилки), зроблені окремими деревами, повинні мати низьку кореляцію між собою.

3 МОДЕЛЬ ПРОГНОЗУВАННЯ СПОРТИВНИХ МАТЧІВ З ВИКОРИСТАННЯМ LSTM

3.1 Мережа LSTM

Long Short-Term Memory (LSTM) - це особлива категорія рекурентних нейронних мереж, призначених для обробки даних часових рядів. LSTM - це мережа з довгою короткочасною пам'яттю, яка здатна вибірково запам'ятовувати закономірності протягом тривалого часу.

Вона є ідеальним вибором для моделювання послідовних даних і тому використовується для вивчення складної динаміки людської діяльності. Довгострокова пам'ять називається станом клітини. Завдяки рекурсивній природі комірок в них зберігається попередня інформація. Для зміни станів комірки використовуються вентиля забування, розміщені нижче стану комірки. Вентиль забування виводить значення, які вказують, яку інформацію потрібно забути, шляхом множення 0 на позицію в матриці. Якщо на виході воріт забування виходить 1, то інформація зберігається в комірці. Вхідний вентиль визначає, яка інформація повинна надходити до станів комірки. Нарешті, вихідний вентиль повідомляє, яка інформація повинна бути передана в наступний прихований стан.

Рахунок футбольного матчу команди є функцією $F_{tk}^u = (u, g, l, c, s, h, t_k)$. Це означає, що команда u грає матч у локації h у момент часу t_k .

Історія футбольного матчу - це набір рахунків команд у футбольному матчі. Вона може позначатися як

$$FH_u = (u, g_1, l_1, c_1, s_1, h_1, t_1), (u, g_2, l_2, c_2, s_2, h_2, t_2), \dots, (u, g_k, l_k, c_k, s_k, h_k, t_k),$$

Таким чином, всі історії футбольних матчів команд можна позначити через $AFH = \{FH_{u1}, FH_{u2}, \dots, FH_{u|U|}\}$, де U - кількість всіх команд.

Прогнозування футбольних матчів: враховуючи історію футбольних матчів

команди FH_u , метою є прогнозування наступних можливих результатів гри команди, включаючи результати змагань, кількість голів, швидкість контролю м'яча та кількість пропущених голів. Це допомагає тренеру вчасно визначати стан команди та тренувати команду. У таблиці 3.1 показані використовувані позначення.

Таблиця 3.1 - Використані позначення

| Символ | Опис. |
|----------------------------------|--|
| u | команда |
| $U = \{ u_1, u_2, \dots, u_M \}$ | збір всіх команд |
| F_{tk}^u | інкасація |
| h_t | прихований вектор блоку LSTM |
| $P_{tN+1,FK}^u$ | Ймовірність результату матчу команди u при t_{N+1} |
| f_t, i_t, o_t | відображений вектор затвора, вхідний вектор затвора та вихідний вектор затвора блоків LSTM |
| C_t | статус товару |
| KT | кандидатський елемент |
| g | цілі |
| l | втрачені голи |
| c | швидкість управління кулькою |
| s | результати конкурсу |
| h | перевага домашнього майданчика |
| t_k | час гри |
| FH_u | набір оцінок на футбольний матч |
| AFH | добірка репортажів про футбольні матчі всіх команд |
| $P_{tN+1,FK}^u$ | LSTM вихід |
| $O_{tN+1,Ck}^u$ | Вихід LSTM для спортивного прогнозування |
| σ | сигмоїдна функція |
| | набір вагових матриць для LSTM моделі |
| | набір векторів зміщень для LSTM моделі |

Мережа LSTM використовується для отримання історичної інформації про

спортивні тренування. Хоча ШНМ може обробляти дані часових рядів, базова ШНМ дуже схильна до зникнення градієнта і вибуху градієнта в процесі навчання моделі, і вона не може добре вловити довгострокові відносини. LSTM є хорошим варіантом RNN, і він відрізняється від загального RNN оновленням методу стану. Структура блоку LSTM показана на рисунку 3.1 .

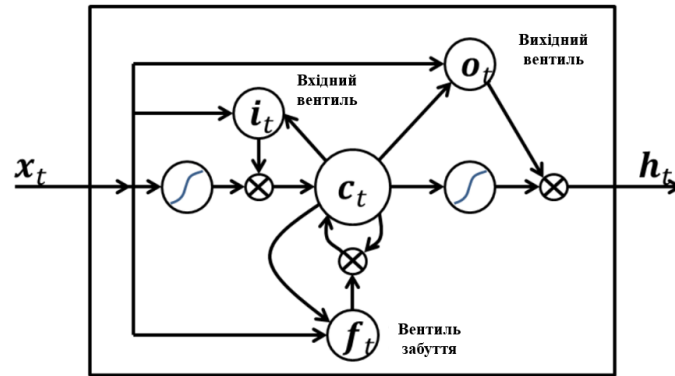


Рисунок 3.1 - Блокова структура LSTM

Члени f_t визначають, яку інформацію слід відкинути, а яку зберегти. Через сигмоїдну функцію пропускається інформація з попереднього прихованого стану h_{t-1} з поточного входу x_t . Далі f_t знаходиться в межах від 0 до 1. Якщо значення близьке до 0, то це означає, що інформація буде забута. Інакше, чим ближче до 1, тим більше інформації буде збережено. А формула розрахунку f_t виглядає наступним чином:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (3.1)$$

де σ - сигмоїдальна функція активації, яка обмежує значення між 0 та 1. h_{t-1} - вихід попереднього елемента, x_t - вхід поточного елемента, W_f - вага, b_f - член зсуву. Потім попередній прихований стан h_{t-1} та поточний вхід x_t підставляються у сигмоїдну функцію. Це визначає, які значення оновлюються шляхом перетворення значень від 0 до 1. Серед них 0 означає неважливе, а 1 - важливе. Потім у функцію \tanh передається прихований стан h_{t-1} і поточний вхід x_t . А формула розрахунку виглядає наступним чином:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (3.2)$$

де W_i - вага, а b_i - член переміщення. Тепер стан елемента розраховано і ми оновлюємо нове значення до стану елемента.

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (3.3)$$

$$C_t = f_t^* C_{t-1} + i_t^* \tilde{C}_t \quad (3.4)$$

де \tanh - функція активації \tanh , W_C - вага стану елемента, а b_C - член зсуву.

Нарешті, вихідний елемент визначає, яким має бути наступний прихований стан. Потім новий стан елемента C_t і новий прихований стан h_t переносяться на наступний часовий крок. А формула розрахунку виглядає наступним чином:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (3.5)$$

$$h_t = o_t^* \tanh(C_t) \quad (3.6)$$

де W_o - вага стану, а b_o - член зміщення. Таким чином, є можливість використовувати LSTM для аналізу виступу спортивної команди. За допомогою відповідного забування та пригадування можна спрогнозувати додаткові результати команди, що допоможе тренеру вчасно зрозуміти поточний стан команди та допомогти її підготовці.

3.2 Механізм уваги

Механізм уваги по суті є похідним від механізму зорової уваги людини. Коли людський зір сприймає речі, він зазвичай не бачить сцену від початку до кінця, а спостерігає певну частину в міру необхідності [3 5]. В історичних тренуваннях і навчаннях спортивної команди завжди є більш важливі ігри, які мають більший

вплив на наступні команди. А деякі види спорту або тренування можуть не привертати надто багато уваги і не мати великого впливу на подальший прогноз виступу. Перевага механізму уваги полягає в тому, що він може зосередитися на впливових частинах. В даний час механізми уваги поділяються на кілька типів.

Суть функції уваги можна описати як відображення запиту на ряд пар "ключ-значення", як показано на рисунку 3.2 .

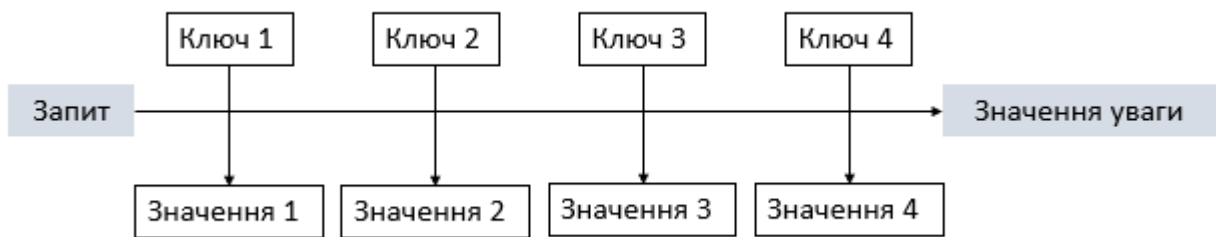


Рисунок 3.2 - Сутність функції уваги

Порівнюючи схожість між ключем та значенням, отримували бал уваги. Чим вищий бал, тим більше вони схожі. Як показано на рисунку 3.3 , розрахунок уваги складається з трьох кроків.

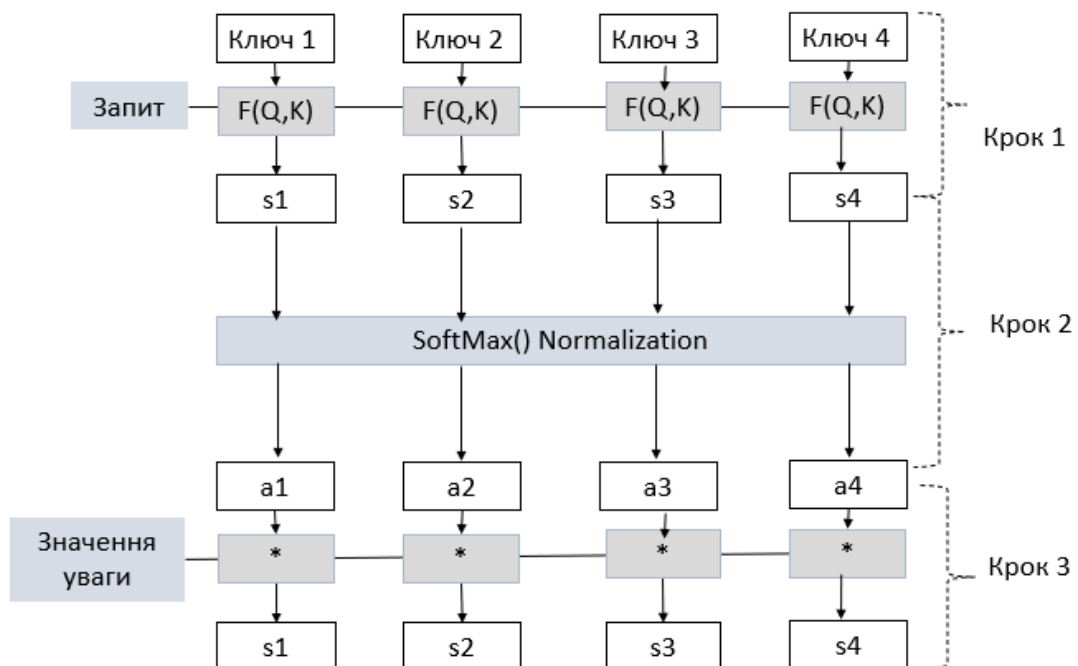


Рисунок 3.3 - Схема розрахунку уваги

Перший крок - розрахунок ваги. Нам потрібно обчислити схожість між запитом і кожним ключем. Існує багато функцій для обчислення схожості, таких як скалярний добуток, сплайсинг, перцептрон тощо. Було вирішено використовувати скалярний добуток через його ефективність та високу швидкість. Формула розрахунку виглядає наступним чином:

$$f_{mul}(Q, K) = QK^T \quad (3.7)$$

Щоб нормалізувати значення ваги, потрібно нормалізувати ці оцінки. Функція softmax може добре реалізувати процес нормалізації. На другому кроці ми використовуємо функцію softmax для нормалізації ваг. На третьому кроці ми отримуємо остаточну увагу:

$$\text{Attention}(Q, K, V_{val}) = \text{softmax}(f(Q, K)) V_{val} \quad (3.8)$$

3.3 LSTM модель для прогнозування спортивних результатів

Ми будемо використовувати LSTM для прогнозування спортивних результатів. LSTM дуже добре справляється з нелінійними даними і фіксує нелінійний зв'язок між історичними досягненнями команди. Стан команди буде змінюватися в залежності від стану членів команди, зовнішнього середовища та багатьох інших факторів. Тому спрогнозувати результативність, використовуючи лише лінійну модель, неможливо. Але головна мета прогнозування результату гри - отримати результат наступного матчу. Прогнозуючи результат матчу заздалегідь, тренер може вчасно скоригувати підготовку команди. Це може зробити команду кращою, і це сприяє проведенню тренувань і навчань команди.

У модель LSTM для прогнозування спортивних результатів ми вводимо історичні результати змагань команди в хронологічному порядку, щоб отримати результат кожного підрозділу. При цьому прогнозується ймовірність успіху і невдачі в кінці гри, а також кількість кидків, голів, рівень володіння м'ячем і пропущених голів у грі. Тренер тренує гравців через результат і стан гри, тим самим зменшуючи ймовірність невдачі і збільшуючи ймовірність успіху. На рисунку 3.4 показана архітектура LSTM.

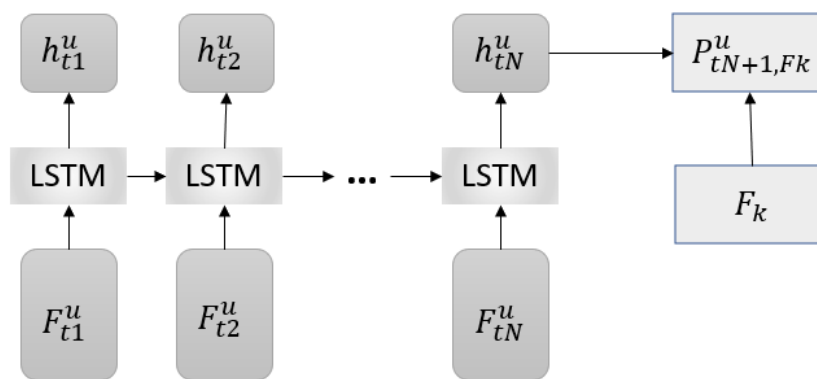


Рисунок 3.4 - Архітектура LSTM

Кортеж F_{ik}^u визначається як вхідні дані в кожний момент часу. Тому є можливість оновлювати кожен прихований вектор h_{ii}^u , використовуючи поточний вхід та попередній прихований вектор h_{ii-1}^u . Тоді, використовуючи вихід блоку LSTM, можна спрогнозувати результати наступних командних змагань:

$$h_{t_i}^u = LSTM(W_F F_{t_i}^u, h_{t_{i-1}}^u) \quad (3.9)$$

де $W_F \in \mathbb{R}^{d \times d}$ та $W_N \in \mathbb{R}^{d \times d}$ - матриці переходів. Якщо ймовірність перемоги більше 50%, то команда має більшу ймовірність перемоги. Коли ймовірність перемоги дорівнює 50%, ймовірність нічиєї вища. В цей час тренер повинен вчасно змінити стратегію команди, щоб збільшити ймовірність перемоги команди. Коли ймовірність перемоги менше 50%, ймовірність поразки команди вище. У цей час

тренеру необхідно підвищувати рівень підготовки команди, звертати увагу на стан спортсменів, відповідно змінювати стратегію і прагнути перетворити поразку в перемогу.

$$P_{t_{N+1}, F_k}^u = (W_N h_{t_N}^u)^T (W_F F_{t_i}^u) \quad (3.10)$$

Для прогнозування спортивних матчів не всі історичні матчі мають однаково важливий вплив на наступну гру. Деякі змагання є більш важливими, тому вплив, безумовно, буде більшим і вплив на наступне змагання буде більшим. Навпаки, деякі звичайні тренувальні ігри мають відносно менший вплив. У реальному тренувальному матчі команда може мати стратегічну надмірність під час змагань. Також членам команди важко змоделювати психологічний стан офіційного змагання, що може призвести до недостатньої підготовки [37]. Тому для прогнозування спортивних матчів необхідно більше уваги приділяти результатам, які мають більший вплив на прогнозування наступних результатів. Механізм уваги добре справляється з цією роботою. Механізм уваги буде присвоювати різну вагу різним історичним записам гри [38]. Хронологічні записи з більшим впливом будуть мати більшу вагу, в той час як записи змагань з меншим впливом будуть мати меншу вагу. Після операції нормалізації ми присвоюємо відповідну вагу кожному історичному запису, а сума всіх ваг дорівнює 1.

Прогнозування результатів спортивних змагань є дуже чутливим до часу. Якщо прогнозований результат проявиться через кілька тижнів або навіть більше, то немає ніякої своєчасності і практичної цінності. Тому нам потрібно спрогнозувати матч команди, яка може з'явитися в наступній грі. Ми використовуємо ковзне часове вікно, щоб зосередитися на короткостроковому стані гри команди. Рідкісні доступні дані про ігри перешкоджають навчання моделі. Різні команди мають різний час і частоту змагань. Якщо припустити, що певний час визначається як стандартне ковзне вікно, то різниця в часовому інтервалі спортивного матчу буде серйозно впливати на навчання моделі. Навпаки, більш стабільним є визначення ковзного вікна на основі кількості матчів. Кількість матчів може краще забезпечити наявність

достатньої кількості безперервних даних для участі в модельних тренуваннях. Коли розмір ковзного вікна дорівнює 5, воно буде зосереджене на перших десяти станах гри користувача, щоб передбачити наступний результат гри користувача. Розмір розсувного вікна потрібно встановлювати відповідно до фактичного стану команди.

LSTM-модель для прогнозування спортивних результатів включає три етапи проектування. Найпростіша LSTM-модель була використана для прогнозування результатів змагань. Також LSTM було поєднано з механізмом уваги. Потім ми замінюємо початковий стан h_{tN}^u прихованого шару LSTM на розраховану вагу уваги, щоб отримати остаточну ймовірність результату матчу:

$$r_{tN}^u = \sum_{i=1}^N a_i h_{t_i}^u \quad (3.11)$$

$$o_{t_{N+1}, C_k}^u = (W_N r_{tN}^u)^T (W_F F_{t_i}^u) \quad (3.12)$$

По-третє, для забезпечення своєчасності прогнозованих результатів, тобто прогнозованим результатом є виступ команди в наступному матчі, використовується ковзне вікно. На рисунку 3.5 зображено архітектуру спортивної моделі LSTM.

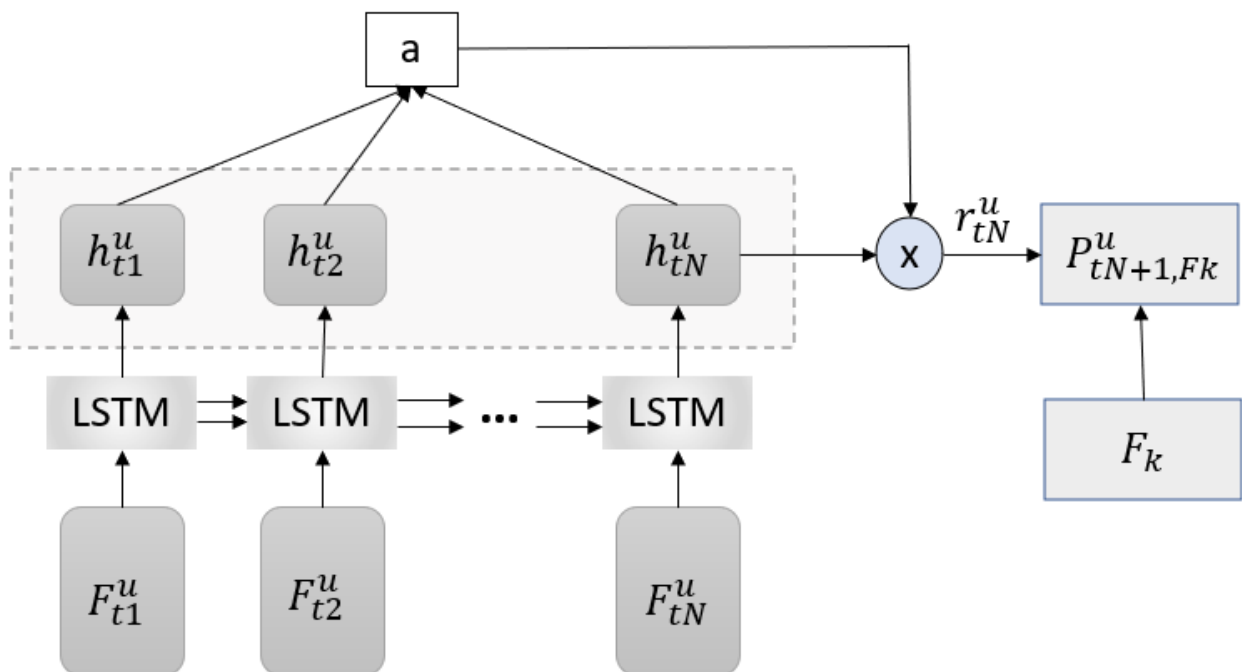


Рисунок 3.5 - Архітектура LSTM-моделі для спортивного прогнозування

Хоча модель враховує багато факторів, які впливають на результат спортивного матчу, все ж є ще деякі інші фактори, які не були враховані, такі як фізичний стан спортсменів і несподівані фактори, які можуть з'явитися в матчі. Крім того, додавання ковзного вікна не може точно передбачити можливі результати кожного матчу.

4 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

4.1 Підготовка даних

Було досліджено дев'ять місяців матчів. Дані включають забиті та пропущені м'ячі, відсоток володіння м'ячем, чи має команда домашню перевагу, час та результати матчу. Якщо команда має перевагу домашнього майданчика, то вона записується як 1, в іншому випадку - як 0. Інформація про кожен матч надається у вигляді вектора для подальшого введення в СППР для навчання, є три типи результатів: перемога, поразка та нічия. У процесі навчання моделі в якості навчальної вибірки для навчання моделі використовуються дані перших семи місяців. Дані останніх двох місяців змагань є валідаційною множиною. У табл. 4.1 наведено результати матчів двох команд.

Таблиця 4.1 - Протоколи матчів команд А та Б

| Команда | Воротар | Пропущені голи | Коефіцієнт володіння м'ячем (%) | Своя нива | Дата | Результат |
|---------|---------|----------------|---------------------------------|-----------|------------|--------------|
| А | 0 | 2 | 33 | 1 | 25-07-2020 | Поразка |
| | 3 | 2 | 54 | 1 | 30-07-2020 | Перемога |
| | 1 | 0 | 60 | 1 | 05-08-2020 | Перемога |
| | 2 | 2 | 51 | 1 | 10-08-2020 | Жеребкування |
| | 1 | 0 | 66.3 | 1 | 14-08-2020 | Перемога |
| | 2 | 3 | 45 | 1 | 19-08-2020 | Поразка |
| | 0 | 0 | 47 | 1 | 26-08-2020 | Жеребкування |
| | ... | ... | ... | ... | ... | ... |
| | 2 | 1 | 51.7 | 0 | 21-03-2021 | Перемога |
| В | 2 | 0 | 53 | 0 | 15-09-2021 | Перемога |
| | 1 | 3 | 44 | 0 | 30-07-2021 | Поразка |
| | 0 | 4 | 40 | 0 | 05-08-2021 | Поразка |
| | 1 | 0 | 51 | 0 | 10-08-2021 | Перемога |
| | 1 | 0 | 62 | 0 | 14-08-2021 | Перемога |
| | 1 | 1 | 45 | 0 | 19-08-2021 | Жеребкування |
| | 2 | 1 | 53 | 0 | 09-09-2021 | Перемога |
| | ... | ... | ... | ... | ... | ... |

| Команда | Воротар | Пропущені голи | Коефіцієнт володіння м'ячем (%) | Своя нива | Дата | Результат |
|---------|---------|----------------|---------------------------------|-----------|------------|-----------|
| | 1 | 0 | 51.7 | 1 | 21-03-2022 | Перемога |

Розмір ковзного вікна було встановлено рівним 5 - для прогнозування результатів майбутніх матчів використовувалися результати останніх п'яти ігор команди. Приклад додавання ковзного вікна для команди А наведено на рисунку 4.1. У ковзаючому вікні містяться дані з п'яти матчів команди. Таким чином можна краще натренувати модель для прогнозування виступу команди в наступному матчі. Тобто тренер може заздалегідь надати команді настанови та спеціальну підготовку, виходячи з можливих матчевих умов гри команди.



Рисунок 4.1 - Процес прогнозування

В процесі навчання, після додавання ковзного вікна, в модель LSTM були введені дані, що базуються на увазі. Спочатку дані матчів команди створюють вектор, а після сортування за часом вводиться ковзне вікно. Для кожної команди перше ковзне вікно містить дані перших десяти матчів, а друге ковзне вікно пересувається один раз назад, включаючи дані від другого матчу до останнього матчу.

Тренуючи модель протягом перших семи місяців, можна отримати кращу модель прогнозування, а потім перевірити дані для наступних двох місяців. Під час валідації будуть отримані результати прогнозування моделі, і ми збережемо результати прогнозування для оцінки ефективності моделі.

Прогнозовані результати порівнювалися з реальним значенням для отримання оцінки ефективності роботи. Також була зроблена спроба спрогнозувати результати кількох наступних матчів. Модель може прогнозувати як ефективність матчу, так і його результат. Процес порівняння наведено в таблиці 4.2.

Таблиця 4.2 - Порівняння фактичних результатів з прогнозованими

| Команда | Фактичні результати | Прогнозовані результати | Точність (%) |
|---------|---|--|--------------|
| A | Нічия, поразка, нічия, нічия, перемога | Перемога, поразка, поразка, нічия, перемога | 60 |
| B | Перемога, перемога, нічия, поразка, нічия | Перемога, перемога, нічия, поразка, поразка | 80 |
| ... | ... | ... | ... |
| N | Нічия, перемога, перемога, перемога, поразка, поразка | Перемога, перемога, перемога, перемога, поразка, поразка | 80 |

Даний метод прогнозування результатів спортивних змагань може допомогти командній підготовці та тренуванням. Використання ковзного вікна дозволило забезпечити своєчасність прогнозування спортивної гри.

4.2 Опис умов проведення експериментальних досліджень

Існує величезна кількість інструментів для проведення математичних досліджень. Це можуть бути як сервіси або додатки, так і мови програмування. Однією з таких мов є Python.

Eclipse є де-факто IDE з відкритим вихідним кодом для розробки на Java. Існує безліч розширень і доповнень, які роблять Eclipse корисним для вирішення

Рисунок 4.3 - Таблиця очищення даних

4.3 Аналіз даних

Набір даних містить 62 різні статистичні дані для кожного матчу, але вам потрібно вибрати саме ті, які мали найбільший вплив.

На рис. 4.4 наведено візуалізацію розподілу найбільш корисних функцій.

Аналізуючи розподіл "Домашні/Виїзні", можна побачити, що результати матчів більш сприятливі для "домашніх" команд, ніж для "виїзних".

Дві характеристики "Удари в створ воріт вдома" та "Удари в створ воріт на виїзді" мали дуже незначний, якщо взагалі мали, вплив на остаточні результати. Натомість величезний вплив мали "Удари в створ воріт вдома" та "Удари в створ воріт на виїзді".

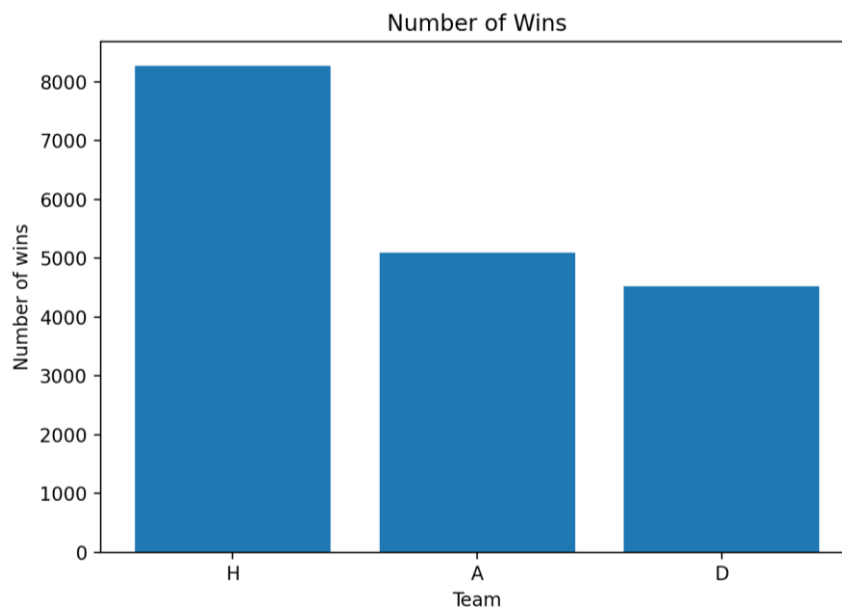


Рисунок 4.4 - Розподіл за місцем проживання

Кількість жовтих карток майже не вплинула на результат. А от кількість червоних карток мала величезний вплив. Це поширена проблема в прикладному машинному навчанні, коли потрібно визначити, чи мають певні вхідні ознаки

відношення до результату.

У випадку проблем класифікації, де вхідні змінні також є категоріальними, статистичні тести можуть бути використані для визначення того, чи є вихідна змінна залежною або незалежною від вхідних змінних. Якщо вона є незалежною, то вхідна змінна є ознакою-кандидатом, яка може не мати відношення до проблеми і може бути вилучена з набору даних.

Одним з таких тестів є статистична гіпотеза хі-квадрат Пірсона.

Іншою проблемою, з якою ми стикаємося, є колінеарність, тобто стан, коли дві змінні є високо корельованими і містять однакову інформацію про дисперсію в межах даного набору даних (рис. 4.5).

```

HC is NOT an important predictor
AC is NOT an important predictor
HY is NOT an important predictor
AY is NOT an important predictor
AF is IMPORTANT for Prediction
AR is IMPORTANT for Prediction
AS is IMPORTANT for Prediction
AST is IMPORTANT for Prediction
HC is IMPORTANT for Prediction
HF is IMPORTANT for Prediction
HR is IMPORTANT for Prediction
HS is IMPORTANT for Prediction
HST is IMPORTANT for Prediction
HTAG is IMPORTANT for Prediction
HTHG is IMPORTANT for Prediction
ht_label is IMPORTANT for Prediction
at_label is IMPORTANT for Prediction

```

Рисунок 4.5 - Результат аналізу хі-квадрат

Одним з методів, який ми можемо використати для перевірки на мультиколінеарність змінних, є розрахунок дисперсійного коефіцієнта інфляції (VIF). Високе значення VIF вказує на те, що відповідна незалежна змінна є сильно колінеарною з іншими змінними в моделі.

Після розрахунку VIF для цього набору даних я виявив, що наступні змінні мають високі значення VIF:

```
HS: 9
AS: 4
HF: 5
AF: 2
```

Рисунок 4.6 - Результат аналізу хі-квадрат

Попереднє спостереження щодо шоків "Додому" та "Вдома" підтверджується цим тестом. Також виявлено більше змінних, які є колінеарними.

Після видалення зайвих функцій отримано наступне (рис. 4.7):

```
home_encoded    non-null float64
away_encoded    non-null float64
HTHG            non-null float64
HTAG            non-null float64
HST             non-null float64
AST             non-null float64
HR              non-null float64
AR              non-null float64
```

Рисунок 4.7 - Видалення непотрібних функцій

4.4 Навчання та тестування моделей

Теорема Байєса дає ймовірність настання події при заданій ймовірності іншої події, яка вже відбулася.

Наївне припущення, зроблене в цьому конкретному класифікаторі, полягає в тому, що всі ознаки є "незалежними" одна від одної. Це полегшує прогнозування, але саме тому прогнози є досить наївними.

Але на практиці існує багато реальних випадків використання цього типу

класифікатора, а саме класифікація документів та фільтрація спаму серед багатьох інших.

Випадкові ліси - це просто набір дерев рішень, де велика кількість дерев рішень виробляє свій власний прогноз, і прогноз, що набрав найбільшу кількість голосів, стає модельним прогнозом.

Дерево рішень, яке є будівельним блоком випадкового лісу, є саме тим, що впливає з назви. Це деревоподібна структура, в якій модель приймає рішення "так" або "ні" в кожному вузлі, щоб пройти по дереву, і врешті-решт досягає одного з вузлів листа, де робить прогноз.

Логістична регресія називається логістичною функцією або сигмоїдною функцією, яка використовується в основі методу.

Він використовує рівняння як представлення, подібно до лінійної регресії, де вхідні дані комбінуються лінійно з використанням ваг або значень коефіцієнтів для прогнозування вихідного значення.

Самі по собі логістичні регресії є лише бінарними класифікаторами, тобто вони не можуть обробляти вхідні дані з більш ніж двома класами. У нашому випадку ми маємо 3 класи для нашого результату (H, D, A).

Однак для цього існують розумні розширення логістичної регресії. У логістичній регресії "один проти решти" (OVR) для кожного класу навчається окрема модель, яка прогнозує, чи належить спостереження до цього класу, чи ні (таким чином, це робить проблему бінарної класифікації). Припускається, що кожна проблема класифікації (наприклад, клас H чи ні) є незалежною.

Набір даних розділено у співвідношенні 4:1 для навчання та тестування. Після першого етапу були отримані наступні результати:

Логістична регресія один проти всіх класифікаторів

 Модель навчена за 0,804028 секунди

Інформація про навчання:

F1 Оцінка:0.6612824278022515

Accuracy:0.6612824278022515

Зроблено прогнозів за 0.001457 секунд

Статистика тестів:

F1 Оцінка: 0.6633109619686801

Ассурасу:0.6633109619686801

Зроблено прогнозів за 0.000830 секунд

Наївний Байєсівський класифікатор Госсена

Модель навчена за 0.016375 секунди

Інформація про навчання:

F1 Оцінка: 0.6289769946157612

Ассурасу:0.6289769946157612

Зроблено прогнозів за 0.007448 секунд

Показники тесту:

F1 Оцінка:0.6061856823266219

Ассурасу:0.6061856823266219

Зробив прогнозів за 0.001650 секунд

Класифікатор випадкових лісів

Модель навчена за 1.326726 секунд

Інформація про навчання:

F1 Оцінка: 0.9999300748199427

Ассурасу:0.9999300748199427

Зроблено прогнозів за 0.243328 секунд

Показники тесту:

F1 Оцінка: 0.6554809843400448

Ассурасу:0.6554809843400448

Зробив прогнози за 0.111957 секунд

Приклад 4.1 - Перший етап навчання

За допомогою логістичної регресії та випадкового лісу ми отримали 65% точності, а за допомогою наївного Байєса - 60%.

Після багатьох доопрацювань, ось фінальні результати:

Логістична регресія один проти всіх класифікаторів

Модель навчена за 0.714509 секунд

Інформація про навчання:

F1 Оцінка:0.6815789473684211

Ассурасу:0.6815789473684211

Зроблено прогнозів за 0.001436 секунд

Тестові показники:

F1 Оцінка: 0.7092105263157895

Ассурасу:0.7092105263157895

Зроблено прогнозів за 0.000857 секунд

Наївний байєсівський класифікатор Госсена

Модель навчена за 0.013822 секунди

Інформація про навчання:

F1 Оцінка: 0.6398026315789473

Ассурасу:0.6398026315789473

Зроблено прогнозів за 0.007501 секунди

Показники тесту:

F1 Оцінка: 0.6526315789473685

Ассурасу:0.6526315789473685

Зроблено прогнозів за 0.002132 секунди

Класифікатор випадкових лісів

Модель навчена за 1.527298 секунд

Інформація про навчання:

F1 Оцінка: 0.999671052631579

Ассурасу:0.999671052631579

Зроблено прогнозів за 0.267743 секунди

Статистика тестів:

F1 Оцінка: 0.6907894736842105

Ассурасу:0.6907894736842105

Зробив прогнози за 0.124963 секунди

Приклад 4.2 - Другий етап навчання

Додавання "Домашніх пострілів" та "Виїзних пострілів" дійсно допомогло

дещо підвищити точність.

ВИСНОВОК

У роботі проаналізовано методи класифікації нейронних мереж. Проаналізовано архітектури, типи нейронних мереж та їх активаційні функції. На основі аналізу обрано архітектуру нейронної мережі - багатошаровий перцептрон з сигмоїдальною функцією активації та модель LSTM.

Розроблено інтелектуальну систему прогнозування результатів спортивних змагань, яка може допомогти командним тренуванням та фізичним вправам. Використання розсувного вікна дозволило забезпечити своєчасність прогнозування спортивної гри.

Результат футбольних матчів може бути змодельований на основі попередньої гри команд-суперників та їх рейтингу у відповідних лігах, а також з використанням результатів попередніх матчів на основі ковзного вікна.

Доцільність методів перевірено на прикладах.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Аксак Н.Г., Сердюк Н.М., Овчаренко Є. Модель системи віддаленого моніторингу та прогнозування стану здоров'я працівника в процесі виробничої діяльності. *Інформаційні технології та системи*. Матеріали Міжнародної науково-практичної конференції: тези доповідей, 9-10 квітня 2020 р. – Х.: ХНЕУ імені Семена Кузнеця, 2020. – С.3.
2. Овчаренко Є. Прийняття рішень за допомогою багатоагентних систем. *Методи та засоби обчислювального інтелекту*: матеріали XXIV міжнар. молод. форуму, м. Харків, 7-9 квіт. 2020 р. Харків, 2020. С. 173-174.
3. Аксак Н., Овчаренко Є. Огляд багатоагентних систем для електронного навчання. *Інформаційні технології в сучасному світі: дослідження молодих вчених*. Матеріали Міжнародної науково-практичної конференції молодих учених, аспірантів та студентів: тези доповідей, 17 – 18 лютого 2022 р. – Х.: ХНЕУ імені Семена Кузнеця, 2022. – С.3.
4. Сучасні інформаційні технології і системи: монографія / В. П. Бурдаєв, Н. Г. Аксак, М. В. Кушнарьов та ін.; за заг. ред . В. С. Пономаренка. - Харків: Вид. «Стиль-іздат», 2021. - 182 с...
5. H. Li Analysis on the construction of sports match prediction model using neural network *Soft Computing* (2020), pp. 1-11
6. H. Xu, P. Wu, J. Wei, Z. Yang, J. Wang A meta-path-based recurrent model for next poi prediction with spatial and temporal contexts *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*, Springer (2019), pp. 219-235
7. S. Jeon, B. Hong, V. Chang Pattern graph tracking-based stock price prediction using big data *Future Generat. Comput. Syst.*, 80 (2018), pp. 171-187
8. J. Luo, J. Li, L. Jiao, J. Cai On the effective parallelization and near-optimal deployment of service function chains *IEEE Trans. Parallel Distr. Syst.*, 32 (5) (2021), pp. 1238-1255

9. J. Cai, Z. Huang, L. Liao, J. Luo, W.-X. Liu Appm: adaptive parallel processing mechanism for service function chains *IEEE Transactions on Network and Service Management*, 18 (2) (2021), pp. 1540-1555
10. M.A. Rahman A deep learning framework for football match prediction *SN Applied Sciences*, 2 (2) (2020), p. 165
11. Y.J. Park, H.S. Kim, D. Kim, H. Lee, S.B. Kim, P. Kang A deep learning-based sports player evaluation model based on game statistics and news articles *Knowl. Base Syst.*, 138 (2017), pp. 15-26
12. A.H. Montoye, M. Begum, Z. Henning, K.A. Pfeiffer Comparison of linear and non-linear models for predicting energy expenditure from raw accelerometer data *Physiol. Meas.*, 38 (2) (2017), p. 343
13. W.R. Johnson, J. Alderson, D. Lloyd, A. Mian Predicting athlete ground reaction forces and moments from spatio-temporal driven cnn models *IEEE (Inst. Electr. Electron. Eng.) Trans. Biomed. Eng.*, 66 (3) (2018), pp. 689-694
14. L. Wu, Z. Yang, Q. Wang, M. Jian, B. Zhao, J. Yan, C.W. Chen Fusing motion patterns and key visual information for semantic event recognition in basketball videos *Neurocomputing*, 413 (2020), pp. 217-229
15. J. Huang, N. Li, T. Li, S. Liu, G. Li, Spatial-temporal context-aware online action detection and prediction, *IEEE Transactions on Circuits and Systems for Video Technology*.
16. M. Woźniak, M. Wiczorek, J. Siłka, D. Połap Body pose prediction based on motion sensor data and recurrent neural network *IEEE Transactions on Industrial Informatics*, 17 (3) (2020), pp. 2101-2111
17. M. Keshtkar Langaroudi, M. Yamaghani Sports result prediction based on machine learning and computational intelligence approaches: a survey *Journal of Advances in Computer Engineering and Technology*, 5 (1) (2019), pp. 27-36
18. N. Watson, S. Hendricks, T. Stewart, I. Durbach Integrating machine learning and decision support in tactical decision-making in rugby union *J. Oper. Res. Soc.* (2020), pp. 1-12

19. D. Strnad, A. Nerat, Š. Kohek Neural network models for group behavior prediction: a case of soccer match attendance *Neural Comput. Appl.*, 28 (2) (2017), pp. 287-300
20. C. Li, D. Song, D. Tao Multi-task recurrent neural networks and higher-order markov random fields for stock price movement prediction: multi-task rnn and higher-order mrfs for stock price classification *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2019), pp. 1141-1151
21. M. Chen, W.-Z. Li, L. Qian, S.-L. Lu, D.-X. Chen Next poi recommendation based on location interest mining with recurrent neural networks *J. Comput. Sci. Technol.*, 35 (2020), pp. 603-616
22. M. Oytun, C. Tinazci, B. Sekeroglu, C. Acikada, H.U. Yavuz Performance prediction and evaluation in female handball players using machine learning models *IEEE Access*, 8 (2020), pp. 116321-116335
23. C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, C.-W. Lin Deep learning on image denoising: an overview *Neural Network.*, 131 (2020), pp. 251-275
24. Y. Li, H. Ma, L. Wang, S. Mao, G. Wang, Optimized content caching and user association for edge computing in densely deployed heterogeneous networks, *IEEE Trans. Mobile Comput.*.doi:10.1109/TMC.2020.3033563.
25. W. Zaremba, I. Sutskever, O. Vinyals, Recurrent Neural Network Regularization, arXiv preprint arXiv:1409.2329.
26. Y. Zhao, R. Yang, G. Chevalier, R.C. Shah, R. Romijnders Applying deep bidirectional lstm and mixture density network for basketball trajectory prediction *Optik*, 158 (2018), pp. 266-272
27. W. Wei, H. Wu, H. Ma An autoencoder and lstm-based traffic flow prediction method *Sensors*, 19 (13) (2019), p. 2946
28. Y. Li, C. Liao, Y. Wang, C. Wang Energy-efficient optimal relay selection in cooperative cellular networks based on double auction *IEEE Trans. Wireless Commun.*, 14 (8) (2015), pp. 4093-4104

29. Y. Yan, J. Ren, G. Sun, H. Zhao, J. Han, X. Li, S. Marshall, J. Zhan Unsupervised image saliency detection with gestalt-laws guided optimization and visual attention based refinement *Pattern Recogn.*, 79 (2018), pp. 65-78
30. Y. Li, S. Xia, B. Cao, Q. Liu, et al., Lyapunov optimization based trade-off policy for mobile cloud offloading in heterogeneous wireless networks, *IEEE Transactions on Cloud Computing*.doi:10.1109/TCC.2019.2938504.
31. M. Zhang, V. Palade, Y. Wang, Z. Ji Attention-based word embeddings using artificial bee colony algorithm for aspect-level sentiment classification *Inf. Sci.*, 545 (2021), pp. 713-738
32. J. Guo, X. Nie, Y. Ma, K. Shaheed, I. Ullah, Y. Yin Attention based consistent semantic learning for micro-video scene recognition *Inf. Sci.*, 543 (2021), pp. 504-516
33. K. Wu, J. Wu, L. Feng, B. Yang, R. Liang, S. Yang, R. Zhao An attention-based cnn-lstm-bilstm model for short-term electric load forecasting in integrated energy system *International Transactions on Electrical Energy Systems* (2020), Article e12637
34. Y. Xie, R. Liang, Z. Liang, C. Huang, C. Zou, B. Schuller Speech emotion classification using attention-based lstm *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27 (11) (2019), pp. 1675-1685
35. S. Zhang, G. Wu, J. Gu, J. Han Pruning convolutional neural networks with an attention mechanism for remote sensing image classification *Electronics*, 9 (8) (2020), p. 1209
36. M. Sangiorgio, F. Dercole Robustness of lstm neural networks for multi-step forecasting of chaotic time series, *Chaos Solitons & Fractals*, 139 (2020), p. 110045
37. J. Stübinger, B. Mangold, J. Knoll Machine learning in football betting: prediction of match results based on player characteristics *Appl. Sci.*, 10 (1) (2020), p. 46
38. X.-m. Yu, W.-z. Feng, H. Wang, Q. Chu, Q. Chen An attention mechanism and multi-granularity-based bi-lstm model for Chinese q&a system *Soft Computing*, 24 (8) (2020), pp. 5831-5845
39. Z. Zhu, Y. Wang, X. Zhou, L. Yang, G. Meng, Z. Zhang Swav: a web-based visualization browser for sliding window analysis *Sci. Rep.*, 10 (1) (2020), pp. 1-4