Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)
Кафедра Інформатики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

ДОСЛІДЖЕННЯ ТА ВИКОРИСТАННЯ BIG DATA ДЛЯ ПРОГНОЗУВАННЯ РЕЗУЛЬТАТІВ СПОРТИВНИХ ПОДІЙ НА ОСНОВІ ІСТОРИЧНИХ СТАТИСТИЧНИХ ДАНИХ
(тема)

Виконав:
здобувач 2 року навчання,
групи ІНФм-24-1
Середа І. А.
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

Науковий керівник доц. Руденко Д. О.
(посада, прізвище, ініціали)

Допускається до захисту

Завідувач кафедри інформатики _____ Кобилін О. А.
(підпис) (прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту

Кафедра Інформатики

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«____»_____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Середі Іллі Андрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та використання Big Data для прогнозування результатів спортивних подій на основі історичних статистичних даних

затверджена наказом університету від 14 листопада 2025 року № 1045Ст
2. Термін подання здобувачем роботи до екзаменаційної комісії 08 грудня 2025 р.
3. Вихідні дані до роботи технології обробки великих даних, методи машинного навчання для прогнозування спортивних результатів, літературні джерела щодо застосування алгоритмів ML у спортивній аналітиці, інструменти для попередньої обробки історичних футбольних даних, програмні засоби для реалізації прогнозних моделей Python, Scikit-learn, XGBoost, CatBoost, PyTorch, методи калібрування ймовірностей, технології побудови API FastAPI, React, TypeScript, набори даних Англійської Прем'єр-ліги з платформ Football-Data, Kaggle та Opta, допоміжні діаграми, графіки та статистичні матеріали, результати навчання та тестування моделей, синтетично згенеровані дані для моделювання динаміки матчу.

4. Перелік питань, що потрібно опрацювати в роботі
1. Аналіз методів роботи з великими даними та машинного навчання у спортивній аналітиці.
2. Дослідження статистичних, просторових і часових показників футбольних матчів та визначення ключових ознак, що впливають на результат гри..
3. Формування вибірки та проведення попередньої обробки даних.
4. Порівняння та оцінка різних моделей машинного навчання і вибір найефективнішої.
5. Розробка веб-застосунку для інтерактивного прогнозування футбольних матчів та інтеграція моделі у клієнтсько-серверну архітектуру.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) актуальність прогнозування спортивних результатів, об'єкт і мета дослідження, постановка задачі; блок-схема обробки великих даних; діаграми попередньої обробки та нормалізації показників; ілюстрації формування вибірки й побудови ознак; приклад синтетичних хвилинних даних; схема архітектури веб-застосунку; інтерфейс головної сторінки та графік зміни ймовірності перемоги; сторінка статистики з аналітичними графіками; теплові карти дій гравців; порівняльні графіки ефективності моделей; підсумкові діаграми точності та перспективи розвитку системи.

6. Консультанти розділів роботи (п.6 включається до завдання 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

№ з/п	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	29.09.2025	
2	Аналіз завдання, підбір літератури	30.09.25-07.10.25	
3	Аналіз літератури з досліджуваної проблеми	08.10.25-14.10.25	
4	Особливості методів прогнозування спортивних подій	15.10.25-27.10.25	
5	Дослідження методів прогнозування спортивних подій	15.10.25-27.10.25	
6	Програмна реалізація	28.10.25-05.11.25	
7	Обґрунтування отриманих результатів	06.11.25-11.11.25	
8	Оформлення пояснювальної записки	12.11.25-24.11.25	
9	Перевірка на нормоконтроль	04.12.25	
10	Перевірка на плагіат	05.12.25	
11	Рецензування	06.12.25	
12	Підготовка презентації та доповіді	10.12.25	
13	Занесення роботи в електронний архів	17.12.25	
14	Попередній захист кваліфікаційної роботи	17.12.25	

Дата видачі завдання 29 вересня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ доц. Руденко Д. О
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи: 80 с., 6 табл., 14 рис., 32 джерела.

ПРОГНОЗУВАННЯ, СПОРТИВНА АНАЛІТИКА, REACT, МАШИННЕ НАВЧАННЯ, ЛОГІСТИЧНА РЕГРЕСІЯ, ВИПАДКОВИЙ ЛІС, ГРАДІЄНТНЕ ПІДСИЛЕННЯ, МОДЕЛЬ РИЗИКІВ, ПУАССОНІВСЬКА МОДЕЛЬ, АНСАМБЛЕВІ МЕТОДИ, ВІЗУАЛІЗАЦІЯ ДАНИХ, PYTHON, TYPESCRIPT.

Об’єктом дослідження є процес аналітичного прогнозування результатів спортивних подій на основі великих даних.

Предметом дослідження є методи та алгоритми машинного навчання для оцінювання ймовірностей спортивних результатів у динаміці матчу.

Метою дослідження є розроблення інтерактивної системи аналізу великих даних для прогнозування результатів спортивних матчів з використанням алгоритмів машинного навчання.

Використано методи логістичної регресії, випадкового лісу, градієнтного підсилення, пуассонівської моделі та моделі ризиків.

Наукова новизна роботи полягає у створенні інтегрованої системи, яка поєднує машинне навчання та аналітичні методи у єдиному середовищі великих даних для прогнозування спортивних результатів.

Взаємозв’язок з іншими роботами результати можуть бути використані в системах аналітики спортивних ставок.

Рекомендацією щодо використання результатів роботи є програмний комплекс придатний для розширення та підключення до спортивних баз даних.

У результаті дослідження розроблено веб-сайт з двошаровою архітектурою, реалізовано набір моделей машинного навчання, інтегровано інтерфейс для динамічного прогнозування та візуалізації результатів спортивних матчів у режимі реального часу.

ABSTRACT

Explanatory note to the qualification work: 80 pages, 6 tables, 14 figures, 32 sources.

PREDICTION, SPORTS ANALYTICS, REACT, MACHINE LEARNING, LOGISTIC REGRESSION, RANDOM FOREST, GRADIENT BOOSTING, HAZARD MODEL, POISSON MODEL, ENSEMBLE, DATA VISUALIZATION, PYTHON, TYPESCRIPT.

The object of the research is the process of analytical prediction of sports event outcomes based on Big Data.

Subject of research is machine learning methods and algorithms for estimating the probabilities of sports results in the dynamics of a match.

The aim of the research is to develop an interactive Big Data analytics system for predicting sports match outcomes using machine learning algorithms, stream processing, and real-time result visualization.

The research applies methods such as Logistic Regression, Random Forest, Gradient Boosting, Poisson, and Hazard models. was built; and model training and comparison were performed.

Scientific novelty of the work lies in the creation of an integrated system that combines machine learning and analytical methods within a unified Big Data environment for predicting sports results

Interconnection with other works is that the results can be applied in sports betting analytics systems.

Recommendations for using the results of the work is that the developed software system can be extended and connected to real sports databases.

As a result of the research, a web-based application with a two-tier architecture was developed. It includes a set of machine learning models, a module for dynamic prediction, and an interface for real-time visualization of sports match outcomes.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

LogLoss – Logarithmic Loss Function (логарифмічна функція втрат)

Brier Score – Brier Score (метрика якості ймовірнісних прогнозів)

xG – Expected Goals (очікувані голи)

ELO – система рейтингової оцінки команди/гравця на основі ймовірностей результатів матчів

EPL – English Premier League (Англійська Прем’єр-ліга)

ВСТУП

У сучасному світі цифрова трансформація охопила практично всі сфери нашого життя – починаючи з освіти та економіки й закінчуючи спортом та розвагами. Обсяги даних, що генеруються щодня, зростають у геометричній прогресії, а їхнє ефективне використання стає ключовим чинником успішності будь-якої галузі. Особливо ця ефективність стосується сфери спорту, де аналітика даних перетворилась на невід'ємну складову процесів управління, підготовки спортивних трансляцій, тактики на гру та прогнозування можливих результатів. Технології великих даних відкривають принципово нові можливості для обробки, використання, аналізу та візуалізації великих масивів даних, що дозволяє нам глибше та точніше розуміти ймовірності того чи іншого результату спортивної події, що значно підвищує точність прогнозів.

Актуальність цього дослідження пояснити дуже легко, бо вона обумовлена тим, що сфера спортивної аналітики та загалом спортивний напрямок розвиваються дуже швидко і тепер вже просто неможливо уявити собі перегляд будь-якої футбольної події без додаткової статистики чи інформації. Варто також додати, що за багатовікову історію існування спорту було вже зіграно безліч матчів, інформацію про хід та результати яких потрібно десь зберігати та якось аналізувати. При таких великих обсягах даних стає досить очевидно, що стандартні методи обробки інформації вже не справляються з тим, щоб виявити закономірності та якісно дати прогноз на матч. В такій ситуації на допомогу приходить технологія великих даних, яка здатна поєднати в собі всі задачі по зберіганню, обробці та інтелектуальному аналізу даних різного виду. Така потужна технологія відкриває двері для розробки інноваційних моделей, які можуть в подальшому використовуватись спортивними клубами, різними аналітичними сервісами і, звісно, на спортивних трансляціях.

Сьогодні активно розвивається сфера використання даних для прогнозування та аналізу спортивних подій. Інтерес до таких технологій

стрімко зростає як у наукових колах, так і в комерційних структурах. Відомі компанії, такі як Opta, Stats Perform, Sportradar та ряд інших аналітичних платформ, займаються систематичним збором та обробкою величезних обсягів інформації про спортивні змагання. Отримані дані стають основою для глибокого аналізу результатів команд, тренерських стратегій, індивідуальних показників спортсменів, а також для створення моделей прогнозування результатів матчів.

Висока точність таких прогнозів забезпечується використанням сучасних технологій обробки даних – від класичних статистичних методів і регресійного аналізу до алгоритмів машинного навчання, нейронних мереж і методів кластеризації, які дозволяють виявляти приховані закономірності в спортивній статистиці та використовувати їх для аналітичних висновків.

Попри значні досягнення, ще залишаються невирішеними низка важливих наукових питань. Серед таких питань – вибір оптимальних алгоритмів прогнозування та аналізу для різних видів спорту, визначення найважливіших та найінформативніших ознак, обробка зашумлених даних, а також інтеграція різних джерел інформацію, таких як відео, дані з сенсорів гравців та соціальні мережі. Крім того, одним з найважливіших питань цієї сфери є швидкість обробки та аналізу, бо будь-яка затримка може коштувати великих грошей компаніям, тому вони завжди стараються знаходити нові й нові алгоритми для прогнозування та ефективного аналізу. Усі ці аспекти зумовлюють необхідність подальших досліджень у сфері використання технології великих даних для прогнозування результатів спортивних подій.

Наукова задача, що розглядається в роботі, полягає у розробці та дослідженні ефективності різних видів моделей для прогнозування результатів спортивних подій на основі історичних статистичних даних із застосуванням такої технології як аналіз великих даних. Для вирішення такої комплексної задачі потрібно провести дослідження для аналізу вже існуючих методів обробки великих масивів даних, створити свою тестову модель прогнозування результатів, а також виявити оптимальні та неоптимальні алгоритми для

виконання умов поставленої задачі. Також потрібно провести порівняльний аналіз таких алгоритмів та їх результатів на основі тестових даних.

Актуальність роботи полягає у необхідності розробки ефективних та якісних методів для виконання аналізу, обробки та прогнозування результатів спортивних подій на основі історичних статистичних даних з використанням технології аналізу великих даних. В сучасні часи спортивна аналітика стала невід'ємною частиною будь-якої спортивної трансляції та вже просто неможливо уявити якісний перегляд матчу без демонстрації цікавих статистичних даних. Через це точність прогнозів є дуже важливим та навіть критичним аспектом, бо від цього залежить довіра до аналітичного сервісу та в подальшому – прибуток від такої аналітики.

Технологія опрацювання великих даних у цій сфері дозволяє суттєво підвищити точність прогнозів, виявляти приховані закономірності та залежності, а також формувати нові підходи до аналізу статистичної інформації. Використання методів інтелектуального аналізу даних разом із технологіями машинного навчання відкриває можливість створення адаптивних моделей, здатних реагувати на такі показники, як поточний рахунок, хвилина матчу, видалені гравці і так далі. Таким чином, дослідження в цій сфері є важливим не лише з теоретичної точки зору, а й з практичної, бо наявність алгоритму ефективного аналізу та прогнозування результатів може бути важливим стартом для створення будь-якого бізнесу, пов'язаного зі спортивною аналітикою та статистикою.

1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ПРОГНОЗУВАННЯ РЕЗУЛЬТАТІВ СПОРТИВНИХ ПОДІЙ

1.1 Аналіз сучасних методів прогнозування результатів спортивних подій та приклади їх практичного застосування

1.1.1 Модель Пуассона з імітацією Монте-Карло

Перший метод, який варто розглянути, це модель Пуассона з імітацією Монте-Карло [1]. Цей підхід часто використовується в спортивній аналітиці, особливо коли необхідно оцінити, скільки голів команда може забити протягом матчу. У наукових працях та серед аналітиків ставок ця модель вважається однією з найнадійніших для прогнозування результатів у футболі, оскільки вона точно відображає статистичну природу гри та дозволяє отримати реалістичні оцінки ймовірності.

Суть цього підходу полягає в тому, що кількість голів, забитих командою, розглядається як випадкова величина, що підпорядковується закону Пуассона. Основним параметром є інтенсивність (λ), яка показує середню кількість голів, забитих командою за певний час. Вона розраховується за формулою:

$$\lambda = \frac{G}{M}, \tag{1.1}$$

де G – середня кількість голів, що забила команда за певний період часу;

M – кількість зіграних матчів за цей же період часу.

Імовірність того, що команда заб'є k голів, визначається за формулою:

$$P(k;\lambda) = \frac{\lambda^k e^{-\lambda}}{k!}. \tag{1.2}$$

У цьому рівнянні $P(k,\lambda)$ показує імовірність забити саме k голів при середній інтенсивності λ.

Першим недоліком саме класичної версії моделі Пуассона варто виділити її неможливість підлаштовуватись та враховувати змінні, окрім рахунку та поточної хвилини. Такі змінні, які не можна розрахувати цифрами, наприклад, психологічний стан, підтримку фанатів, яка жене команду вперед, травми, втому і так далі. Сенс в тому, що футбол і спорт загалом не є чимось статичним, що можна легко розрахувати лише на основі рахунку або поточної хвилини. Інші фактори теж потрібно враховувати, якщо потрібно, щоб система видавала точний та надійний результат майже для будь-яких змагань.

Ще одним недоліком такого методу є те, що модель не дозволяє робити багаторазові симуляції для виявлення закономірностей та збільшення точності прогнозу. Без багаторазових симуляцій система буде доволі часто помилятись та робити неправильні висновки.

Саме тому в сучасній спортивній аналітиці класична модель Пуассона часто поєднується з моделюванням Монте-Карло. Такий підхід дозволяє «зіграти» матч тисячі разів у віртуальному середовищі, що дає змогу оцінити розподіл можливих результатів. Можна використовувати механізм, реалізований у вигляді багаторазових симуляцій – до 30000 повторень, які враховують поточну хвилину гри, рахунок і рейтинги команд за системою ELO.

Після запуску симуляцій система підраховує, скільки разів команда виграла, зіграла внічию або програла. Це формує розподіл ймовірностей для трьох основних результатів – перемога, нічия та поразка.

Отримані дані дозволяють зробити більш обґрунтований прогноз, враховуючи реальну ситуацію на полі – хто грає вдома, який рахунок і скільки часу залишилося до кінця матчу. Такий підхід поєднує математичну строгість із випадковістю спорту, яка моделюється завдяки великій кількості симуляцій. В результаті метод забезпечує гнучкі та досить точні прогнози, навіть коли історичних даних недостатньо для побудови повної статистичної моделі.

Додатково, використання стохастичних симуляцій робить систему здатною адаптуватися до нетипових сценаріїв гри, підсилюючи її стійкість до шуму та непередбачуваних подій.

1.1.2 Модель ризиків

Ще одним підходом, який є логічним продовженням моделі Пуассону з імітацією Монте-Карло, є модель ризиків [2]. Ця модель особлива тим, що вона здатна враховувати інтенсивність гри в залежності від рахунку. Наприклад, якщо гостьова команда веде в рахунку 1:0, то вона зменшить кількість своїх атак, щоб зберегти рахунок, через що імовірність зміни рахунку на 2:0 зменшується. Так само і навпаки, якщо домашня команда програє 1:0, то вона почне ще більше атакувати, щоб відіграти ся, завдяки чому імовірність нічийного результату зросте.

Тобто якщо попередня модель вважала інтенсивність за сталу змінну, модель ризиків розраховує інтенсивність динамічно в залежності від поточного рахунку та ймовірної поведінки команда в залежності від ситуації на полі. Математично інтенсивність в цій моделі можна виразити так:

$$\lambda_h' = \lambda_h \cdot e^{-a \cdot d}, \lambda_a' = \lambda_a \cdot e^{a \cdot d}, \tag{1.3}$$

де λ_h', λ_a' – скориговані інтенсивності для домашньої та гостьової команд;

$d = G_h - G_a$ – різниця у рахунку;

a – коефіцієнт чутливості, який зазвичай лежить у межах від 0,1 до 0,2.

Таким чином, можна впевнено казати, що модель ризиків ефективно враховує емоційні та тактичні аспекти поведінки команд, що робить дану модель більш наближеною для реальної гри, ніж до статичної симуляції. Завдяки цій моделі можна отримати не лише імовірність фінального результату матчу, а й імовірність зміни рахунку впродовж кожної хвилини матчу. Крім того, динамічний характер цієї моделі дає змогу відстежувати, як навіть незначні зміни у грі миттєво впливають на прогнозовані події. Така система дозволяє робити не лише прогнози, а й цікаві для користувача візуальні симуляції матчів станом на кожну хвилину. Збільшення функціональних можливостей моделі без втрати точності завжди є бажаним результатом.

1.1.3 Логістична регресія з ізотонічним калібруванням

Ще одним методом є логістична регресія з ізотонічною калібровкою, яка відноситься до класичних алгоритмів машинного навчання. Перевагою цього алгоритму є здатність знаходити імовірність результату на основі певних ознак або факторів, від яких може залежати результат. Загальна формула при використанні цієї моделі має наступний вигляд:

$$P(y=1 \mid x) = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\ldots+\beta_n x_n)}}, \tag{1.4}$$

$$\mathrm{P}(\mathrm{y}=k \mid x) = \frac{e^{\beta_k^T x}}{\sum_j e^{\beta_j^T x}}, \tag{1.5}$$

де x – вектор вхідних ознак (різниця в рахунку, час до завершення матчу, різниця в рейтингу команд, фактор домашнього поля тощо);

β_k – набір вагових коефіцієнтів для кожного результату.

Після завершення навчання модель досить часто проходить процес ізотонічного калібрування, який коригує отримані імовірності, роблячи їх значно ближчими до реальної статистики. Цей процес дозволяє уникнути перенасичення «уточнених» прогнозів, що трапляється, коли алгоритм переоцінює впевненість у власних результатах й прогнозах.

Процедура ізотонічного калібрування полягає в побудові монотонної функції, яка краще відповідає емпіричним даним. Завдяки цьому досягається більша точність прогнозів, особливо в тих випадках, коли дані містять сторонній шум або мають нерівномірний розподіл результатів в навчальній виборці.

Ізотонічне калібруванням також є важливою при вирівнюванні дисбалансу між класами – наприклад, коли перемог набагато більше, ніж нічиїх або поразок, що є майже постійним явищем для спортивної аналітики та

статистики. Модель, яка не має калібрування, може значно «завищувати» впевненість у певних результатах, неправильно інтерпретуючи імовірність як абсолютні істини. Калібрування ж дає можливість досягти каліброваної впевненості, наприклад, коли прогноз моделі зі значенням 0,7 дійсно означає близько 70% шансів на певний результат події.

Якщо підсумовувати переваги такого підходу, то він добре поєднує простоту інтерпретації та достатньо велику точність. Також цей підхід дає можливість ефективно використовувати модель логістичної регресії в системах, де має велике значення саме швидкість прогнозів та наочність результатів. Саме через це логістична регресія з ізотонічним калібруванням є одним із найпопулярніших методів у професійній спортивній аналітиці, фінансових прогнозах та інших галузях, де потрібні швидкі та відносно точні прогнози.

1.1.4 Випадковий ліс

Випадковий ліс – ансамблевий алгоритм, який поєднує величезну кількість можливих дерев рішень. Основна ідея цього методу полягає в тому, що замість створення однієї великої моделі будується велика кількість дерев, кожне з яких «навчається» на випадковій підмножині даних і наборі ознак.

Кожне дерево формує власний прогноз, а кінцевий результат визначається шляхом усереднення (у завданнях регресії) або голосування (у завданнях класифікації). Це зменшує вплив випадкових коливань у даних і підвищує стійкість до перенавчання, що є характерною рисою окремих дерев.

Імовірність результату для заданих параметрів визначається формулою:

$$P(y = k \mid x) = \frac{1}{T} \sum_{t=1}^{T} P_t(y = k \mid x), \tag{1.6}$$

де T – кількість дерев у моделі;

$P_t(y = k \mid x)$ – передбачення окремого дерева.

На етапі навчання метод випадкового лісу створює декілька десятків або сотен дерев, кожне з яких отримує випадкову підвибірку даних за допомогою методу Bootstrap Aggregation (bagging). Це значить, що деякі приклади можуть і, ймовірно, будуть використанні по декілька разів, а інші можуть взагалі не потрапити до навчання певного дерева. Наявність такого підходу сприяє різноманітності дерев, завдяки чому їх об'єднання дає стабільний та узагальнений результат.

Крім цього, під час побудови кожного дерева на всіх розгалуженнях розглядається не весь можливий набір вхідних ознак, а лише їх випадкова підмножина. Цей процес ще більше зменшує кореляцію між деревами та сприяє уникненню перенавчання.

Варто також відмітити ще одну важливу перевагу методу випадковий ліс, якою є можливість оцінки важливості ознак. Алгоритм працює таким чином, що аналізує, наскільки кожна змінна зменшує невизначеність у процесі побудови дерев рішень, і таким чином визначає, які фактори мають найбільший вплив на визначення кінцевого результату.

У випадку прогнозування саме спортивних подій це дозволяє визначити, що, наприклад, поточний рахунок або залишок часу до завершення матчу мають значно більший вплив на результат, ніж домашня чи гостьова гра або номінальна історична сила однієї або іншої команди. Це також підсилює інтерпретованість моделі та робить її результати більш зрозумілими.

Окрім відносно високої точності, метод також має при собі ще одну корисну властивість – стійкість до стороннього шуму в даних та можливих похибок. У реальних наборах спортивної статистики майже неможливо уникнути хибних даних або взагалі відсутність інформації про якусь метрику, але завдяки усередненню по великій кількості дерев рішень модель залишається стабільною та точною. Це робить її корисною для задач, в яких дані збираються з різних джерел – наприклад, історичних статистичних даних, спортивних API або стрімінгових систем.

До недоліків методу відноситься певна втрата інтерпретованості – окреме дерево можна легко пояснити, але сукупність сотень дерев перетворюється на «чорну скриньку». Однак саме для задач спортивної аналітики це не є критичним недоліком, оскільки головною метою є отримання точного прогнозу, а не пояснення всіх проміжних залежностей.

Таким чином, випадковий ліс поєднує в собі гнучкість, високу точність і стійкість до перенавчання, що робить його одним з найефективніших інструментів для прогнозування результатів спортивних подій у рамках задач роботи з велики об’ємами даних. Його використання в комплексі з іншими методами, такими як логістична регресія чи модель динамічної інтенсивності, дозволяє отримати більш збалансовану систему, здатну навчатися на великих наборах даних і адекватно реагувати на поточні зміни у ігровій ситуації.

1.2 Аналіз літературних джерел щодо апробації методів прогнозування результатів спортивних подій

Розвиток технології роботи з великими об’ємами даних суттєво змінив підходи до аналізу спортивних даних. Якщо раніше, наприклад, основою для прогнозів виступали класичні статистичні моделі, засновані на обмежених наборах параметрів (кількість ударів, відсоток володіння м’ячом, кількість голів тощо), то тепер в центрі уваги дослідників знаходяться масштабні багатовимірні дані, які охоплюють практично всі аспекти гри – від динаміки рухів гравців до психологічних і поведінкових факторів для кожного гравця. У такому підході особливу увагу приділяють пошуку прихованих закономірностей, які неможливо виявити за допомогою традиційних методів.

Велика кількість публікацій, які вийшли за останні п’ять років, присвячена саме поєднанню статистичних методів із алгоритмами машинного навчання для отримання більш точних і гнучких прогнозів для будь-яких спортивних подій.

1.2.1 Загальні тенденції розвитку досліджень у спортивній аналітиці

За інформацією аналітичних звітів провідних аналітичних компаній у сфері спорту, таких як Opta, Stats Perform та Sportradar, обсяг даних, що збирається під час кожного футбольного матчу, перевищує розмір в кілька мільйонів записів. До цих даних відноситься інформація про переміщення гравців, швидкість, передачі та їх точність, удари, показник xG, єдиноборства, помилки, а також різні сторонні фактори: стадіон, погода, час доби тощо.

Обробка таких масивів інформації вимагає не лише статистичних методів, а й складних алгоритмів машинного навчання та систем обробки потокових даних у реальному часі.

Наукові праці Dixon & Coles від 1997 року [3] та Maher від 1982 року [4] стали основою класичних моделей прогнозування футбольних результатів, у яких використовувався розподіл Пуассона для моделювання імовірної кількості голів. Надалі ці підходи були вдосконалені за рахунок урахування залежностей між командами, домашньої переваги та поточного фізичного та психологічного стану гравців.

Починаючи з 2010-х років наукова спільнота продовжує активно вводити принципи аналітики великих даних у спортивну сферу. Зокрема, у роботах Bunker & Thabtah від 2019 року [5] та Baboota & Kaur від 2018 року [6] описано побудову моделей прогнозування результатів матчів із використанням таких алгоритмів, як випадковий ліс, Support Vector Machines (SVM) та градієнтний бустинг, які показуються помітно кращий результат, ніж класичні методи.

1.2.2 Сучасні дослідження з використанням великих даних у спорті

Упродовж останніх п’яти років наукова спільнота демонструє стрімке зростання інтересу до застосування технологій аналітики великих даних у сфері спорту. Якщо раніше дослідження були зосереджені переважно на побудові

статистичних моделей на основі кількості голів, передач чи ударів, то сьогодні акцент робиться на інтеграції великих обсягів даних різної природи – історичних, просторових, часових та контекстних. Це дозволяє не лише передбачати результати змагань, а й розуміти приховані закономірності у грі, поведінкові патерни гравців і вплив зовнішніх чинників на кінцевий рахунок.

Сучасні роботи, опубліковані у 2024–2025 рр., демонструють активне використання гібридних підходів, які поєднують статистичні методи та алгоритми машинного навчання. Наприклад, у дослідженні «Data-driven prediction of soccer outcomes using enhanced machine and deep learning techniques» (2024), яке спеціально присвячене порівнянню різних методів прогнозування, представлено комплексний підхід з багатоступеневою обробкою даних та відбором ознак. У роботі також запропоновано ансамблеву модель, що поєднує класичні методи, як-от логістична регресія, із сучасними архітектурами нейронних мереж. [7]. Авторами підкреслено, що ефективність прогнозу суттєво зростає тоді, коли модель має доступ до даних у реальному часі – таких як рахунок на певній хвилині матчу чи кількість небезпечних моментів. Це підтверджує, що великі дані дають змогу адаптувати математичну модель до змін під час гри.

Інше дослідження – «A Hybrid Machine Learning Framework for Soccer Match Outcome» пропонує гібридну систему, де поєднано кілька алгоритмів для зменшення похибок прогнозування [8]. Науковці довели, що комбінація моделей, зокрема випадковий ліс і градієнтного бустингу, показує більш стабільні результати, ніж окремі методи. Такий підхід вважається одним із ключових напрямів сучасної спортивної аналітики, адже він дозволяє зменшити вплив «шумистих» або неповних даних, які трапляються в реальних наборах.

Особливу увагу останнім часом приділяють і новим архітектурам глибокого навчання. У праці «Evaluating soccer match prediction models: a deep learning approach and feature optimization for gradient-boosted trees» дослідники випробували модель TabNet, яка здатна автоматично виділяти найбільш релевантні ознаки під час навчання [9]. Порівняння показало, що така нейронна

мережа може перевершувати ансамблеві методи, якщо дані мають складну структуру з численними залежностями. Це підкреслює тенденцію до використання алгоритмів, здатних самостійно навчатися на багатовимірних наборах без ручної селекції ознак.

Іншою важливою темою у сучасних дослідженнях є коректне калібрування моделей. Як показано у систематичному огляді «A Systematic Review of Machine Learning in Sports Betting» від 2024 року, висока точність класифікації ще не гарантує реалістичних імовірностей [10]. Автори доводять, що саме калібрування, тобто узгодження між передбаченими й фактичними частотами подій, є ключовим чинником у побудові надійних систем прогнозування. Такий підхід особливо актуальний у сфері ставок, де від коректності ймовірностей безпосередньо залежить фінансовий результат. Нові технологічні напрями також проникають у спортивну аналітику. У 2023 р. було опубліковано дослідження «Predicting football match outcomes with machine learning approaches», у якому запропоновано квантову нейронну мережу для обробки багатовимірних наборів даних [11]. Автори вважають, що квантові алгоритми можуть у перспективі підвищити швидкість та якість прогнозування, особливо коли йдеться про великі історичні бази матчів. Попри експериментальний характер, цей напрям демонструє поступове розширення меж застосування великих даних у спорті – від класичних моделей до інноваційних квантових обчислень.

Практичне застосування подібних рішень активно підтримують комерційні компанії. Платформи Opta, Stats Perform, Sportradar та StatsBomb створюють багатомільйонні бази ігрових подій, які аналізуються в реальному часі. Наприклад, система «Bayes Live Odds» від Sportradar використовує методи ймовірнісного моделювання для динамічного оновлення коефіцієнтів під час матчу, а аналітична платформа «Opta Vision» комбінує машинне навчання з комп’ютерним зором для визначення ключових моментів гри. Ці приклади демонструють, що наукові досягнення швидко інтегруються у практичні системи, створюючи замкнений цикл «наука – дані – прогноз – рішення».

1.2.3 Узагальнення результатів аналізу сучасних досліджень

Проведений аналіз літературних джерел за останні п’ять років показує, що застосування великих даних у спортивній аналітиці стало невід’ємною частиною сучасних досліджень та комерційних рішень. Еволюція цього напряму відбувається швидкими темпами – від простих статистичних моделей, які спиралися лише на історичні результати, до складних інтегрованих систем, що враховують контекст подій, динаміку матчу, фізичний стан гравців і навіть зовнішні чинники, такі як погодні умови чи географічне розташування стадіону.

У більшості сучасних робіт підкреслюється важливість комбінування різних підходів – традиційної статистики та алгоритмів машинного навчання. Якщо класичні моделі, наприклад Пуассонівські, забезпечують стабільну математичну основу, то методи на зразок випадкового лісу, градієнтного бустингу чи нейронних мереж дозволяють виявляти приховані нелінійні зв’язки між змінними. Така синергія стала одним із ключових напрямів розвитку спортивної аналітики. Особливо перспективними вважаються ансамблеві моделі, які поєднують кілька алгоритмів і формують кінцевий прогноз шляхом усереднення або голосування.

Сучасні дослідження також демонструють рух у напрямку інтеграції багатоджерельних даних. В аналітику матчів активно залучаються телеметричні системи GPS, трекінг позицій, відеоаналіз і дані про біометрику спортсменів. Це вимагає застосування складних інфраструктур обробки даних – розподілених баз, потокових систем та хмарних обчислень. Саме тому у сфері роботи з великими даними для спорту дедалі частіше говорять про необхідність побудови цілісних екосистем, які поєднують збір, обробку, аналіз і візуалізацію даних у єдиному середовищі.

Узагальнюючи, можна стверджувати, що сучасний етап розвитку спортивної аналітики характеризується переходом від експериментальних досліджень до практичних рішень, здатних масштабуватися й застосовуватися

на рівні професійних ліг і аналітичних сервісів. Робота з великими даними сьогодні виступає не просто інструментом збору інформації, а повноцінною платформою для прийняття рішень. Вона формує основу для нових інтелектуальних систем, які не лише прогнозують результат, а й допомагають аналітикам і навіть уболівальникам краще розуміти логіку гри.

1.3 Постановка задачі дослідження

У сучасному спорті точність прогнозів має велике значення – від неї залежать тренування, тактика та навіть фінансові рішення. Завдяки розвитку технологій роботи з великими даними стало можливо збирати й аналізувати величезні масиви даних про матчі, команди та гравців, що дало поштовх новим методам прогнозування.

Такі моделі поєднують статистику та машинне навчання, дозволяючи враховувати не лише цифри, а й контекст гри, форму гравців і психологічні чинники. Проте досі постає питання: які методи забезпечують найкраще співвідношення точності, швидкодії та зрозумілості?

Мета дослідження – порівняти ефективність різних підходів до прогнозування спортивних результатів і дослідити, як їх можна поєднати в єдину аналітичну систему, здатну враховувати динаміку та контекст подій.

Серед методів, які досліджуються, виділяються такі підходи:

– пуассонівська модель із симуляцією Монте-Карло, що дозволяє відтворити ймовірнісну природу спортивних подій через багаторазові симуляції можливих результатів;

– логістична регресія як базовий метод класифікації, що дає інтерпретовані результати;

– випадковий ліс і градієнтний бустинг, здатні моделювати складні нелінійні залежності у великому обсязі даних;

– модель ризику, яка враховує часову структуру подій і зміни темпу гри;

– ансамблеві методи, що поєднують кілька алгоритмів для підвищення стабільності прогнозів.

Таке поєднання підходів дозволяє порівняти моделі не лише за точністю передбачення, а й за поведінкою в різних умовах – наприклад, при зміні рахунку, хвилини матчу чи різниці у рейтингах команд.

Завдання дослідження передбачають:

– провести теоретичний аналіз сучасних методів прогнозування результатів спортивних подій, особливо тих, що базуються на великих даних;

– розробити математичні моделі для обраних методів і реалізувати їх у єдиному середовищі для порівняння результатів;

– виконати серію експериментів на історичних або синтетичних наборах даних, змінюючи ключові параметри (рахунок, хвилина, ELO);

– порівняти отримані результати за допомогою статистичних метрик (LogLoss, Brier Score, Accuracy) та визначити сильні й слабкі сторони кожного методу;

– оцінити можливість використання комбінованого підходу для підвищення стабільності прогнозів.

Особливість цього дослідження полягає в інтеграції кількох типів моделей – від класичних статистичних до сучасних ML-алгоритмів – у спільному експериментальному середовищі.

Об’єкт дослідження – процес прогнозування результатів спортивних подій на основі великих даних, що охоплює статистичні та машинні методи аналізу матчів, командних характеристик і динаміки гри.

2 ТЕОРЕТИЧНІ ОСНОВИ ВИКОРИСТАННЯ ВЕЛИКИХ ДАНИХ У СПОРТИВНІЙ АНАЛІТИЦІ

2.1 Поняття та характеристики великих даних

У сучасному інформаційному світі поняття великих даних охоплює не лише колосальні обсяги інформації, а й цілу філософію її обробки, аналізу та зберігання. Йдеться про дані, які настільки масштабні та різноманітні, що традиційні підходи вже не можуть впоратися з їхньою обробкою. Поява цього терміну безпосередньо пов’язана зі стрімким зростанням кількості інформації, яку щодня створюють люди, цифрові пристрої, соціальні мережі, сенсори та організації.

За оцінками аналітиків IDC щорічний обсяг згенерованих у світі даних перевищує 120 зетабайт і ця цифра продовжує збільшуватись у геометричній прогресії. Такі масштаби давно перевищили можливості класичних баз даних і звичних методів обчислень. Сучасні цифрові системи генерують дані з такою швидкістю, що їх опрацювання потребує принципово нових алгоритмів. Саме тому дослідники дедалі частіше звертаються до розподілених обчислень та інтелектуальних систем аналізу. У результаті виникла потреба у зовсім нових підходах, які б поєднували високу швидкодію, гнучке масштабування та здатність до паралельної обробки інформації. Саме так сформувалася сучасна парадигма обробки великих даних – як фундамент для ефективної роботи з даними у будь-якій сфері, де важливі швидкість, точність та здатність бачити закономірності там, де раніше панував хаос.

Поняття великих даних традиційно характеризують через модель «5Vs», яка описує п’ять основних властивостей великомасштабних даних: Volume, Velocity, Variety, Veracity та Value, що представлено на рисунку 2.1.

Ця концепція допомагає зрозуміти, що сила великих даних полягає не лише в їх кількості, а й у швидкості надходження, різноманітті джерел, достовірності та тій практичній цінності, яку вони здатні створювати.

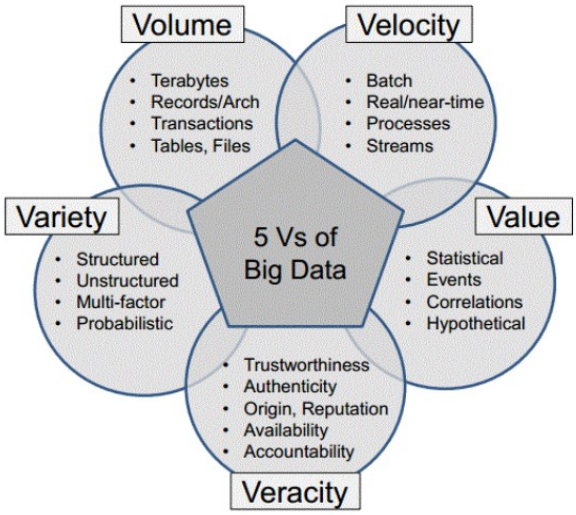

Рисунок 2.1 – Модель 5Vs

Volume (обсяг) – це головна характеристика, яка підкреслює, що йдеться про гігантські масиви інформації. Дані можуть надходити з соціальних мереж, сенсорів пристроїв, відеоспостереження, спортивних датчиків чи телеметричних систем. У контексті спортивної аналітики це означає щосекундну фіксацію координат гравців, швидкостей, ударів, пасів та інших показників, які формують мільйони записів під час одного матчу.

Velocity (швидкість) відображає темп генерації та передачі даних. Сучасні системи збирають і аналізують інформацію в реальному часі – наприклад, під час трансляції спортивного поєдинку або на тренуванні команди. Затримка в обробці навіть на кілька секунд може призвести до втрати актуальності аналітичних висновків, що критично для спортивних аналітичних платформ.

Variety (різноманітність) означає, що дані мають різні формати та структури – текстові файли, відео, аудіо, зображення, таблиці чи потоки датчиків. У спорті це поєднання телеметричних показників, відеозаписів, публікацій у соцмережах та історичної статистики з баз даних матчів. Інтеграція таких джерел вимагає спеціалізованих інструментів, здатних аналізувати неструктуровані та напівструктуровані дані.

Veracity (достовірність) – це ступінь точності та надійності даних. У сфері великих даних часто виникають проблеми із шумами, помилками чи

неповнотою інформації. Наприклад, у спортивних даних датчики можуть фіксувати помилкові координати або втрачати сигнал. Тому одним із ключових завдань аналітики є фільтрація та очищення даних, щоб підвищити їхню якість та достовірність результатів.

Value (цінність) – найважливіша характеристика великих даних, що підкреслює корисність отриманих даних для прийняття управлінських чи стратегічних рішень. У спорті це можливість виявити закономірності, які допомагають покращити тактику, оцінити форму гравців або підвищити точність прогнозування результатів матчів.

Розвиток аналізу великих даних став можливим завдяки появі масштабованих інфраструктур обробки інформації. У 2000-х роках провідні технологічні компанії створили нові моделі зберігання та паралельних обчислень – насамперед Hadoop, MapReduce та Spark. Вони дозволили розподіляти обробку даних між сотнями вузлів у кластері, значно зменшуючи час виконання операцій. Подальший перехід до хмарних технологій (AWS, Google Cloud, Azure) зробив таку обробку масово доступною, а це відкрило можливість застосовувати великі дані в комерційних та наукових сферах, зокрема в аналітиці спорту.

Ключовим аспектом стала поява Data Lakes – сховищ, що дозволяють зберігати дані в сирому вигляді для подальшої аналітики. Завдяки цьому аналітики та дослідники отримують доступ до повної картини подій без втрати деталей, які раніше могли ігноруватися під час передобробки. Сучасні платформи такі як Apache Kafka, Flink, Snowflake та Databricks забезпечують можливість обробляти стрімінгові дані в реальному часі, що особливо актуально для живих спортивних трансляцій.

Типова архітектура великих даних включає чотири основні рівні:

– збір даних, що полягає в отриманні інформації з різних джерел (сенсори, API, відео, соціальні мережі);

– зберігання з використанням розподілених систем на зразок HDFS, Cassandra, MongoDB чи ElasticSearch;

– обробка та аналіз через застосування фреймворків Apache Spark, Hadoop, Storm, а також бібліотек машинного навчання (PyTorch, TensorFlow, Scikit-learn);

– візуалізація та презентація для перетворення аналітичних результатів на зрозумілі графіки й діаграми, це можуть бути засоби на зразок Tableau, Power BI або Plotly.

У сучасних системах роботи з великими даними важливу роль відіграє паралельна та розподілена обробка, яка дозволяє виконувати аналіз мільйонів записів одночасно. Це особливо актуально для спортивних застосувань, де кожна секунда ігрового часу генерує тисячі нових подій. Крім того, активно використовуються технології стрімінгової аналітики (Streaming Analytics), які забезпечують оновлення результатів у режимі on-line.

Попри вражаючі можливості, технологія великих даних має низку викликів. Серед них можна визначити забезпечення конфіденційності даних, адже аналітика включає персональні показники спортсменів. Оскільки дані надходять із різних платформ, виникає проблема якості та узгодженості використовуваних джерел. Крім того, є проблеми масштабованості через те, що не всі системи здатні працювати зі зростанням навантаження. Із застосуванням глибоких нейронних мереж, які діють як «чорна скринька», інтерпретованість результатів може мати сумнівний характер.

Також актуальним є питання енергоспоживання й екологічності обчислень: обробка петабайтів інформації вимагає потужних серверів, а це породжує значне енергетичне навантаження. Тому в сучасних дослідженнях зростає інтерес до «Green Big Data» – енергоефективних підходів до зберігання та обробки інформації.

У сфері спорту обробка великих даних дозволяє переосмислити підходи до аналізу гри. Завдяки об’єднанню величезних масивів історичних і поточних даних аналітики можуть виявляти тонкі патерни поведінки команд, оцінювати ймовірності результатів та формувати адаптивні моделі прогнозування. Технології машинного та глибокого навчання, які працюють на основі великих

даних, здатні здійснювати динамічний аналіз у реальному часі – наприклад, визначати імовірність голу на кожній хвилині матчу чи оцінювати ризик травмування гравців під час навантаження.

2.2 Джерела та типи спортивних даних

Розвиток спортивної аналітики тісно пов’язаний із тим, наскільки різноманітні й якісні дані вдається зібрати під час змагань, тренувань чи дослідницьких спостережень. Сьогодні спорт уже давно перестав бути лише змаганням майстерності – він став високотехнологічною системою, де майже кожен рух спортсмена можна виміряти, зафіксувати та перетворити на дані. Завдяки цьому з’явилася можливість не просто рахувати голи чи секунди, а глибше розуміти логіку гри, аналізувати закономірності й навіть передбачати майбутні результати. Дані поступово перетворилися на справжній стратегічний ресурс: вони допомагають тренерам приймати точніші рішення, аналітикам – будувати прогнози, а вболівальникам – по-новому бачити улюблену команду.

У контексті великих даних спортивна інформація поділяється на кілька основних типів залежно від способу отримання, структури та призначення. Найчастіше виділяють три великі категорії: історичні (ретроспективні), оперативні (реального часу) та контекстуальні дані.

Історичні дані – це статистика минулих матчів, сезонів або турнірів. Вони включають результати ігор, кількість голів, передач, ударів, втрат, штрафів, а також середні показники команд і окремих гравців. Такі дані зазвичай структуровані та зберігаються в базах типу SQL або NoSQL. Саме вони лежать в основі більшості моделей прогнозування, зокрема пуассонівських і регресійних.

Оперативні або потокові дані – це інформація, яка збирається безпосередньо під час гри чи тренування, буквально в реальному часі. Її забезпечують спеціальні трекінгові системи, сенсори та технології

комп’ютерного зору. Вони фіксують координати гравців на полі, швидкість руху, частоту серцевих скорочень, інтенсивність навантаження, кількість спринтів і навіть мікропаузи у діях спортсменів. Такий тип даних надходить із величезною швидкістю, тому перед використанням потребує попередньої фільтрації та узагальнення, аби зменшити «шум» і зробити інформацію придатною для аналізу.

Контекстуальні дані допомагають побачити гру ширше – не лише через статистику, а й через обставини, які можуть вплинути на результат. Сюди входять погодні умови, розташування стадіону, емоційний стан команди, кількість уболівальників на трибунах, а також активність у соціальних мережах перед або після матчу. Іноді саме такі фактори, на перший погляд другорядні, стають вирішальними для підсумку зустрічі. Тому сучасні аналітичні системи все частіше враховують подібні показники, поєднуючи «сухі» цифри з живим контекстом реальної гри.

Таким чином, класифікація спортивних даних є необхідним етапом підготовки інформації до подальшого аналізу. Вона дозволяє не лише впорядкувати великі обсяги відомостей, а й створити передумови для побудови комплексних систем аналітики, які поєднують точність математичних методів із багатовимірністю реального спортивного процесу.

Джерела даних у спортивній аналітиці надзвичайно різноманітні та залежать від рівня організації змагань, доступних технологій і виду спорту. Умовно їх можна поділити на такі основні групи:

– офіційна статистика та бази даних спортивних організацій;

– трекінгові системи та сенсорні технології;

– відеоаналітика та комп’ютерний зір;

– відкриті дані та спортивні API;

– соціальні мережі та медіа-платформи.

Офіційна статистика є найнадійнішим джерелом даних, яке надають федерації, ліги або аналітичні агентства (наприклад, FIFA, NBA, UEFA, Opta Sports, Stats Perform). У таких базах містяться детальні зведення про матчі,

команди, суддів, тактичні схеми та показники ефективності гравців. Дані з офіційних джерел мають високу достовірність і найчастіше використовуються для навчання прогнозних моделей та формування офіційної статистики.

Трекінгові системи – джерела оперативних даних, що забезпечують збір інформації про фізичний стан і переміщення спортсменів у реальному часі. Більшість професійних клубів використовують GPS-трекери та сенсори, які закріплюються на формі або взутті гравців. Вони фіксують швидкість, прискорення, частоту кроків, навантаження на суглоби та інші біомеханічні показники. До таких систем належать Catapult Sports, STATSports, Polar Team Pro, Wimu Pro. Зібрані дані допомагають контролювати рівень підготовки, аналізувати ефективність тренувань і запобігати травмам.

Відеоаналітика є однією з найсучасніших системи, що використовують камери високої роздільної здатності та алгоритми машинного навчання для автоматичного розпізнавання подій матчу. Вони визначають координати гравців, траєкторії руху м’яча, типи передач, удари, втрати чи жести суддів. Прикладами таких технологій є Hawk-Eye та Opta Vision, які поєднують відеоаналіз із нейронними мережами, забезпечуючи формування даних у режимі реального часу [12].

Варто також враховувати дані з відкритих джерел та спортивних API бо це набори статистики, які доступні дослідникам і розробникам через публічні інтерфейси. Наприклад, платформи Football-Data.org, SportsDB, API-Football, NBA Stats API дозволяють отримати дані про склади команд, календар матчів, результати й турнірні таблиці. Такі ресурси широко використовуються у наукових дослідженнях та при створенні експериментальних моделей, однак вимагають додаткової перевірки через можливі розбіжності з офіційними джерелами.

Серед нових джерел інформації можна виділити соціальні мережі, які є новим, але дедалі впливовішим джерелом спортивних даних. Повідомлення в соцмережах (Twitter/X, Instagram, Reddit, Telegram), реакції фанатів, дописи гравців і тренерів створюють інформаційне тло, що може впливати на

психологічний стан команди. Методи аналізу тональності (Sentiment Analysis) дають змогу виявляти зв’язок між громадськими настроями та результатами матчів, що розширює аналітичні можливості у прогнозуванні спортивних подій.

Залежно від джерела походження дані можуть бути представлені в різних форматах. Так, класичні показники матчів і статистика гравців можуть відображатися в табличних даних як то CSV, Excel, SQL. Показники з сенсорів або трекінгових систем формують стрімінгові потоки на кшталт JSON, Kafka або MQTT. Візуальні дані (відео, зображення) застосовуються у комп’ютерному зорі та глибокому навчанні [13, 14, 15]. Текстові дані у вигляді трансляції або новини в соцмережі – для NLP-аналізу та емоційних індикаторів [16].

Для ефективної роботи з ними використовують спеціалізовані сховища даних (Data Lakes, Warehouses) і фреймворки типу Apache Spark, PyTorch, TensorFlow, що забезпечують можливість масштабної обробки інформації.

Важливою особливістю спортивних даних є їхня неоднорідність і часті помилки вимірювання. Дані з різних джерел можуть мати різні часові формати, структуру або одиниці вимірювання. Тому ключовим етапом роботи з ними є очищення, нормалізація та синхронізація. Також важливо зазначити, що досить велика кількість спортивних даних йде з відео та зображень гри, тому також важливо ефективно працювати зі зображеннями для аналізу [17, 18, 19].

Наприклад, координати, отримані з GPS-трекерів, часто потребують корекції через затримки сигналу, а текстові дані із соціальних мереж містять велику кількість «шуму» у вигляді неінформативних коментарів або спаму.

Без якісної підготовки такі дані можуть призвести до викривлених висновків навіть у найточніших моделях машинного навчання.

Кожен тип даних виконує свою роль у системі великих даних для спорту. Історичні дані формують основу для навчання моделей, оперативні дані забезпечують адаптацію прогнозів у реальному часі, а контекстуальні додають аналітичній системі «людський вимір» через врахування зовнішніх факторів та динаміки ігрового середовища.

Поєднання всіх трьох типів створює багатовимірний підхід до аналізу гри, де рішення базуються не лише на статистиці, а й на поведінкових та середовищних чинниках. Саме це робить великі дані у спорті унікальним інструментом, що дозволяє досягати глибшого розуміння процесів і приймати більш обґрунтовані рішення.

Таке поєднання різних джерел і форматів даних змінює сам підхід до аналізу. Якщо раніше спортивна аналітика спиралася лише на цифри – рахунок, удари, володіння м’ячем, то зараз вона охоплює цілий контекст події. Сучасні алгоритми здатні поєднувати статистику з емоційними реакціями гравців, поведінкою команди у стресових ситуаціях, навіть з атмосферою стадіону чи підтримкою уболівальників. Це дозволяє розглядати гру не як набір окремих дій, а як складну динамічну систему, у якій кожен фактор може вплинути на результат.

Крім того, інтеграція різних типів даних відкриває можливість створення глибших моделей прогнозування. Наприклад, історичні дані визначають закономірності, оперативні забезпечують актуальність, а контекстуальні дозволяють урахувати фактори, які не вимірюються безпосередньо, але мають значення. Завдяки цьому аналітика стає точнішою, а прогнози – ближчими до реальних сценаріїв. У результаті технологія великих даних перетворюється не просто на інструмент збору інформації, а на основу для стратегічного мислення у спорті, де кожне рішення підкріплене фактами, досвідом і контекстом гри.

2.3 Методи обробки великих обсягів даних

У сучасній спортивній аналітиці обсяг даних стрімко зростає: одна гра здатна генерувати сотні мільйонів подій, які надходять з камер трекінгу, GPS-сенсорів, телеметрії, медичних датчиків та інших джерел. Для роботи з такими обсягами інформації недостатньо класичних підходів – необхідні технології обробки даних, які забезпечують масштабованість та високу швидкість аналізу.

Основою сучасних рішень є розподілена обробка даних, де великий масив розбивається на частини та аналізується паралельно на декількох вузлах. Такий підхід дозволяє прискорити обчислення у десятки разів і зменшити навантаження на окремі сервери. Принцип роботи подібних систем демонструє архітектурна схема Hadoop–Spark, що можна побачити на рисунку 2.2.

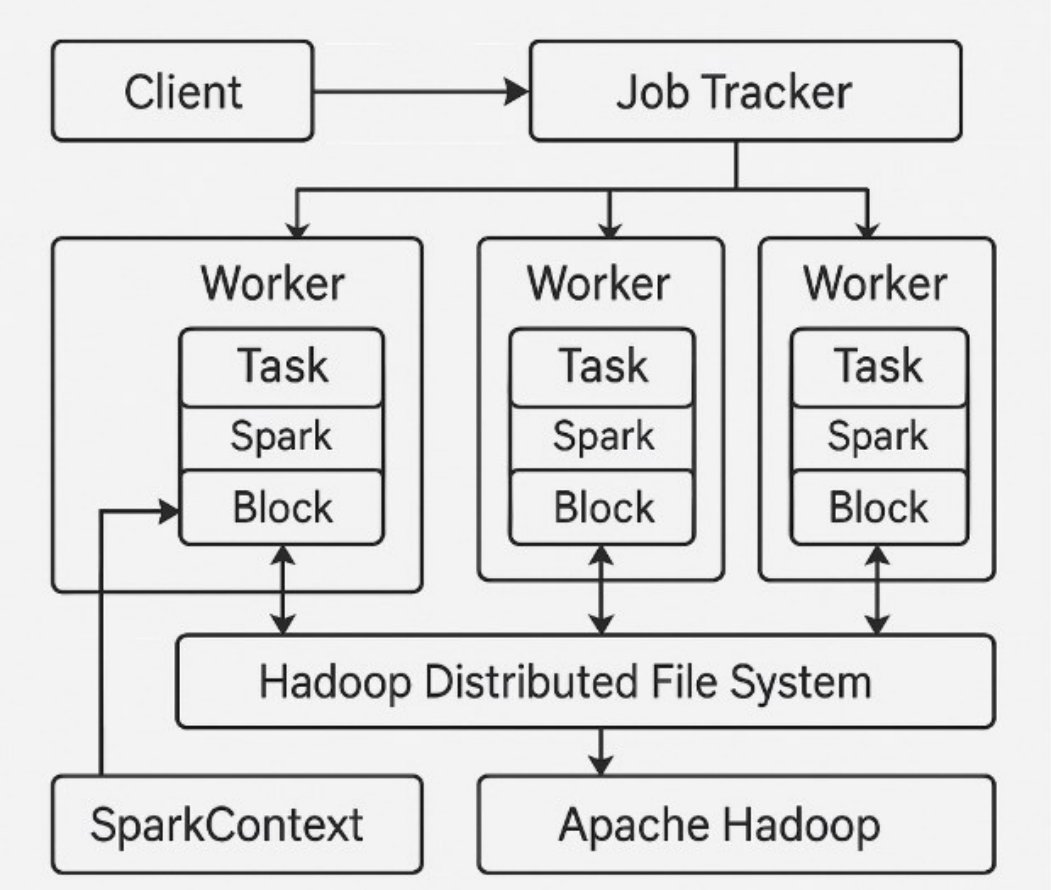

Рисунок 2.2 – Архітектурна схема Hadoop-Spark системи

Перед використанням у моделюванні дані проходять етап ETL-очищення: видалення дублікатів, стандартизацію форматів, синхронізацію часових позначок, узгодження між різними джерелами. Навіть одна помилкова мітка часу може викривити аналіз: програма може «побачити» втрату м’яча замість точного пасу або хибно оцінити швидкість гравця. Тому очищення даних – один з найважливіших етапів, що визначає точність подальших моделей.

Однак у спорті важливе не лише опрацювання історичних даних – критичним є аналіз у режимі реального часу, коли система реагує на події

матчу миттєво. Під час гри надходять дані про швидкість гравців, переміщення, навантаження, тиск, частоту пульсу, володіння м’ячем, перебіг атак. Такі дані передаються як неперервний потік, тому використовується потокова обробка. Її логіку демонструє рисунок 2.3.

Рисунок 2.3 – Схема потокової обробки даних в реальному часі

Після підготовки дані переходять до етапу аналітики та машинного навчання. Сучасні спортивні системи використовують регресійні моделі, дерева рішень, ансамблеві методи та нейронні мережі для пошуку закономірностей у грі. Аналітика дозволяє оцінити ймовірність голу, визначити сильні та слабкі сторони команд, прогнозувати динаміку подій на полі. У складніших випадках застосовуються моделі глибокого навчання, які аналізують відео, визначають положення гравців у просторі, будують trajectories та heatmap-карти зон активності.

Завершальним етапом є візуалізація даних, яка перетворює складні числові таблиці на доступні графіки, дашборди, мережеві діаграми передач, карти дій на полі або heatmap-моделі. Подібні інструменти дозволяють тренерам та аналітикам інтерпретувати результати за секунди, не заглиблюючись у послідовності чисел.

У результаті всі елементи – розподілена обробка, потокова аналітика, ETL-очищення, машинне та глибоке навчання, візуалізація та хмарні технології – формують єдину екосистему, яка перетворює необроблені дані на структуровані знання. Такий підхід робить спортивну аналітику максимально

точною, динамічною та практично корисною як у роботі команд, так і у дослідженнях, моделюванні та прогнозуванні результатів матчів.

2.4 Використання машинного навчання у спортивному прогнозуванні

Машинне навчання сьогодні стало невід'ємною частиною спортивної аналітики. Його активно застосовують як великі професійні клуби, так і незалежні аналітики, які займаються прогнозуванням результатів матчів. У спорті, де результат залежить від десятків факторів – від фізичної форми гравців до настрою команди чи навіть погодних умов, – здатність системи самостійно вчитися на даних дає величезну перевагу. Саме тому машинне навчання поступово витісняє традиційні статистичні підходи, які не завжди враховують складну динаміку спортивних подій [20].

Процес машинного навчання у спортивному прогнозуванні складається з низки послідовних етапів – від збору та очищення даних до моделювання, калібрування та візуалізації результатів. Кожен компонент цього конвеєра відіграє критичну роль у точності прогнозів: помилки на ранніх етапах можуть призвести до некоректних висновків на виході. Загальну структуру цього процесу та взаємозв'язок між його частинами подано у вигляді блок-схеми, що дозволяє наочно побачити логіку роботи ML-системи. Узагальнену послідовність етапів можна побачити на рисунку 2.4**.**

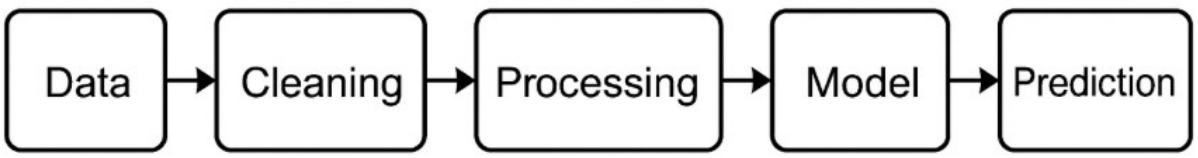

Рисунок 2.4 – Схема роботи ML-системи

Суть машинного навчання полягає в тому, що комп’ютерна програма не діє за заздалегідь заданими формулами, а «вчиться» на реальних прикладах. Алгоритм аналізує великі масиви історичних даних – результати минулих матчів, індивідуальні показники спортсменів, стратегії команд – і поступово виявляє закономірності, які потім використовуються для прогнозування майбутніх подій. Наприклад, система може навчитися визначати, за яких умов команда з більшою ймовірністю переможе: коли грає вдома, має вищий відсоток володіння м’ячем або демонструє стабільну гру останні кілька турів [21].

Процес створення прогнозної моделі складається з кількох етапів. Спочатку аналітики збирають дані з різних джерел – офіційної статистики, трекінгових сенсорів, відеоаналізу чи відкритих API. Далі відбувається очищення даних: видаляються дублікати, виправляються пропуски, узгоджується формат показників. Це необхідно, бо навіть невелика помилка може спотворити результат навчання моделі. Лише після цього дані подаються в алгоритм, який «вчиться» на їхній основі робити власні висновки.

Для прогнозування спортивних подій використовують різні алгоритми машинного навчання. Найпростіші базуються на статистичних підходах – наприклад, логістичній регресії, яка дозволяє оцінити ймовірність перемоги тієї чи іншої команди [22]. У складніших випадках застосовують дерева рішень або ансамблеві методи, що поєднують результати кількох моделей. Вони дають змогу враховувати одразу багато чинників – як об’єктивних (кількість ударів, фолів, пасів), так і контекстуальних (місце проведення матчу, попередня втома команди, травми гравців).

Окрему групу становлять нейронні мережі, які особливо добре працюють з великими та неструктурованими даними – наприклад, із відео. Такі системи можуть аналізувати ігрові моменти: визначати позиції гравців, швидкість м’яча, інтенсивність руху або тактику команд. На практиці це дозволяє робити не просто прогноз рахунку, а й оцінювати якість гри, ефективність окремих спортсменів чи ймовірність того, що конкретна атака призведе до голу [23].

У більшості реальних задач спортивної аналітики використовується не одна модель, а цілий ансамбль алгоритмів, які працюють паралельно з різними типами даних. Комбінування результатів логістичної регресії, дерев рішень, градієнтного бустингу, а також аналітичних моделей на кшталт пуассонівської чи моделі ризиків дозволяє зменшити похибки та врахувати більше факторів, ніж це можливо в межах одного підходу. Схему інтеграції окремих моделей у єдину систему комплексного прогнозування наведено на рисунку 2.5, де показано, як поєднані моделі формують узгоджене ймовірнісне передбачення.

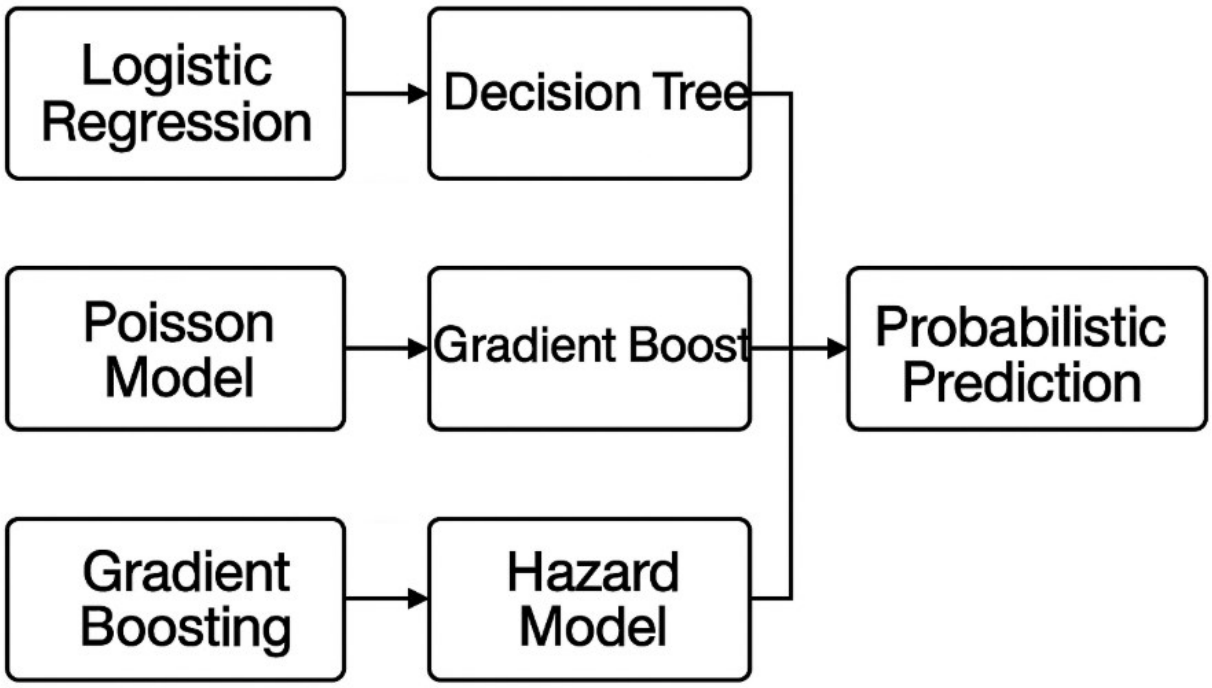

Рисунок 2.5 – Схема ансамблю моделей прогнозування

Машинне навчання застосовують і в медико-фізичній аналітиці. Сучасні спортивні організації збирають величезні масиви фізіологічних даних: пульс, частоту дихання, швидкість відновлення, кількість сну. На основі цих показників алгоритми можуть визначати, коли спортсмен перевтомився, прогнозувати ризик травми або пропонувати режим навантаження. Наприклад, якщо модель «бачить», що пульс і швидкість відновлення за межами норми, вона може попередити тренера та персонал, що гравцю потрібен відпочинок.

Ще один цікавий напрям – аналіз поведінки фанатів і психологічного стану команд. Завдяки методам обробки природної мови (Natural Language Processing) можна аналізувати публікації у соціальних мережах, новини, коментарі та тональність повідомлень. Це допомагає зрозуміти, у якому емоційному стані перебуває команда – під тиском критики чи, навпаки, у стані підйому після серії перемог. Такі фактори, хоч і здаються другорядними, можуть суттєво впливати на мотивацію гравців і, відповідно, на результат.

Сучасні системи прогнозування рідко спираються лише на один алгоритм. Найчастіше використовуються так звані ансамблеві моделі, які поєднують кілька підходів одночасно. Наприклад, одна модель аналізує статистику попередніх матчів, інша – фізичний стан гравців, а третя – погодні умови. Після цього результати об’єднуються, і формується загальний прогноз. Це дозволяє підвищити точність, оскільки кожен алгоритм «підстраховує» інший.

Перевагою машинного навчання є також здатність постійно оновлюватися. Після кожного нового матчу або тренування модель отримує нові дані й автоматично вдосконалюється. Це робить систему гнучкою – вона швидко реагує на зміни у складі команди, появу нових гравців або зміну тактики. Такі «самонавчальні» моделі дозволяють не просто аналізувати минуле, а динамічно підлаштовуватися під теперішні тенденції.

Щоб результати були зручними для сприйняття, прогнози зазвичай візуалізують. Аналітики створюють інтерактивні графіки, карти дій гравців, теплові зони активності або динаміку зміни ймовірності перемоги. Це дає змогу тренерам і менеджерам не лише бачити цифри, а й розуміти їх у контексті гри. Наприклад, за допомогою таких візуалізацій можна швидко зрозуміти, з яких зон команда найчастіше атакує, у яких моментах найчастіше допускає помилки або як змінюється стратегія протягом матчу.

Загалом використання машинного навчання у спортивному прогнозуванні стало справжнім проривом. Воно поєднує точність математичних розрахунків із гнучкістю адаптації до нових умов. Алгоритми не замінюють тренера чи аналітика, але значно розширюють їхні можливості. Завдяки цьому спорт

перетворюється на більш «розумну» сферу, де рішення ґрунтуються не на інтуїції, а на реальних даних. І саме в цьому – головна цінність сучасної спортивної аналітики, яка поєднує науку, технології та живу динаміку гри.

2.5 Аналіз існуючих рішень і систем прогнозування у спорті

Упродовж останнього десятиліття спортивна аналітика зробила величезний стрибок уперед – від ручних підрахунків до складних систем прогнозування, які використовують машинне навчання, нейронні мережі та обробку великих даних. Сучасні рішення вже не обмежуються простим аналізом статистики. Вони здатні враховувати безліч чинників – фізичний стан гравців, емоційний фон, погодні умови, тип суперника чи навіть поведінку фанатів у соціальних мережах. Саме завдяки такій комплексності спортивні аналітичні платформи перетворилися на повноцінні інструменти стратегічного планування.

Однією з найвідоміших і найстаріших систем є Opta Sports (нині – частина компанії Stats Perform). Вона збирає детальну статистику з матчів більш ніж 60 футбольних ліг світу. Система фіксує кожен дотик до м’яча, передачу, удар, перехоплення та створює на їх основі аналітичні моделі. Дані Opta активно використовуються тренерськими штабами, телетрансляторами та навіть букмекерськими компаніями. На базі цієї статистики будуються прогнози результатів матчів, очікуваних голів (xG), імовірності перемоги або поразки, а також рейтинги гравців за ефективністю [24].

Ще одним потужним рішенням є StatsBomb, яке пропонує розширені моделі аналізу дій гравців. На відміну від традиційних систем, що враховують лише факт події, StatsBomb аналізує контекст – положення гравців, напрямок передач, інтенсивність пресингу та позиційні зони. Ця система активно застосовується у футбольних клубах Англії, Іспанії, Німеччини та Італії, а також у наукових дослідженнях з тактичного аналізу гри.

Окрім безпосередньо спортивного використання, подібні системи дедалі частіше інтегруються у фінансову та управлінську діяльність клубів. На основі аналітичних моделей формуються рішення щодо трансферної політики, оцінки ринкової вартості гравців, планування бюджету та навіть прогнозування відвідуваності матчів. Це демонструє, що аналітика перестала бути лише «помічником тренера» – вона поступово перетворюється на інструмент управління бізнесом, де точні дані можуть безпосередньо впливати на фінансові результати.

В американських видах спорту популярністю користуються Hudl, Catapult і ProZone. Вони орієнтовані переважно на аналіз відео та фізичних показників. Наприклад, Catapult використовує сенсори, що закріплюються на тілі спортсмена, і збирає дані про швидкість, прискорення, навантаження, відстань і пульс. Ці показники потім поєднуються з відеоаналізом, що дозволяє не лише оцінювати гру після матчу, а й у режимі реального часу контролювати стан гравців. Такі системи стали незамінними у підготовці до змагань, адже дозволяють запобігати травмам і оптимізувати тренувальні програми.

Велике поширення отримали також відкриті системи, що базуються на публічних наборах даних. Серед них варто відзначити Football-Data. org, API-Football, Sportradar, The SportsDB. Вони надають відкритий доступ до статистики матчів, результатів, складів, турнірних таблиць і навіть історичних архівів. Такі ресурси стали основою для численних дослідницьких проєктів, і стартапів, у яких створюються експериментальні моделі прогнозування результатів. Наприклад, за допомогою цих API можна побудувати систему, яка порівнює кілька сезонів поспіль і визначає закономірності у виступах команд.

Окрему нішу займають рішення, що базуються на штучному інтелекті. Платформи IBM Watson Analytics та Microsoft Azure Sports Analytics використовують глибоке навчання для прогнозування результатів змагань, оцінки ефективності гравців і планування стратегії. IBM Watson, зокрема, застосовується у тенісі – система аналізує дані матчів «Вімблдону», створюючи прогнози щодо ймовірного переможця та форму спортсменів у реальному часі.

У бейсболі, баскетболі та американському футболі активно розвиваються власні аналітичні екосистеми. Система Statcast, створена MLB (Вища бейсбольна ліга США), фіксує кожен рух м’яча та гравців за допомогою високошвидкісних камер і радарів. Зібрані дані використовуються для побудови моделей траєкторії польоту, сили удару, швидкості реакції та багатьох інших параметрів. Подібним чином у баскетболі функціонує Second Spectrum, яка автоматично розпізнає дії гравців на полі, створює теплові карти активності та навіть генерує відеоогляди з аналітичними коментарями.

Цікаво, що частина аналітичних систем орієнтована не лише на команди, а й на звичайних користувачів. Наприклад, додатки Kickdex, WhoScored або SofaScore використовують відкриті джерела даних і алгоритми машинного навчання, щоб створювати персональні прогнози для вболівальників. Вони враховують статистику гравців, форму команд, домашню перевагу, склад суперників і навіть тип покриття поля. Для багатьох фанатів такі прогнози стали інструментом не лише розваги, а й глибшого розуміння гри.

Проте, попри всі досягнення, сучасні системи прогнозування мають і певні обмеження. Більшість із них залежать від якості вхідних даних. Якщо статистика неповна або містить помилки, модель може видавати хибні результати. Крім того, машинні алгоритми не враховують «людський фактор» – психологічний стан гравця, конфлікти в команді, зміну тренера або натхнення після перемоги, які складно виміряти чисельно.

3 РЕАЛІЗАЦІЯ МОДЕЛІ ПРОГНОЗУВАННЯ НА ОСНОВІ ТЕХНОЛОГІЇ ВЕЛИКИХ ДАНИХ

3.1 Технічне середовище для обробки великих даних

3.1.1 Аналіз існуючих рішень для роботи з великими даними

У сучасній спортивній аналітиці кількість даних, що накопичуються, зростає експоненціально. Щодня генерується не лише підсумковий рахунок матчу, а й детальна ігрова статистика, позиційні координати гравців, швидкість переміщення, показники навантаження, інформація про заміни, травми, а також реакції вболівальників у соціальних мережах. Додатково враховуються зовнішні фактори – тип покриття поля, погодні умови, час доби чи навіть географічне розташування стадіону. Об'єднавши ці дані з історичними результатами за кілька сезонів, можна отримати типовий випадок роботи з великими даними – великий обсяг, різноманітність джерел, швидкість надходження та постійну потребу у фільтрації та нормалізації.

Для побудови моделі прогнозування результатів спортивних подій надзвичайно важливо обрати адекватне технічне середовище. Саме від нього залежить швидкість обчислень, масштабованість та інтеграція з інструментами машинного навчання. Вибір середовища визначає, чи зможе система швидко оновлювати показники після кожного туру, чи обмежиться періодичними пакетними обрахунками.

Архітектура типової системи великих даних охоплює послідовність взаємопов'язаних етапів, кожен із яких виконує власну функцію у процесі аналітики спортивних даних:

– джерела даних (Data Sources) формують первинні потоки інформації, що надходять із баз даних, відкритих API, сенсорів гравців, систем моніторингу матчів або соціальних мереж;

– збір та збереження (Data Storage) для накопичення даних у розподіленому середовищі, наприклад, базах Cassandra і Parquet;

– пакетна обробка (Batch Processing) застосовується для періодичного аналізу великих обсягів історичної інформації, наприклад, результатів минулих сезонів;

– потокова обробка (Stream Processing) використовується для обробки подій у реальному часі, зокрема під час активних матчів або оновлення коефіцієнтів;

– аналітичне сховище (Analytical Data Store) акумулює узагальнені дані, які потім використовуються для побудови моделей машинного навчання;

– аналітика та звітність (Analytics and Reporting) забезпечують створення прогнозів, візуалізацій, статистичних панелей і звітів для кінцевих користувачів.

Всі етапи з'єднані між собою двома потоками – пакетним і потоковим, які координуються підсистемою Orchestration, що забезпечує узгоджене керування всіма процесами у межах екосистеми великих даних, саме це і можна побачити на рисунку 3.1 [25].

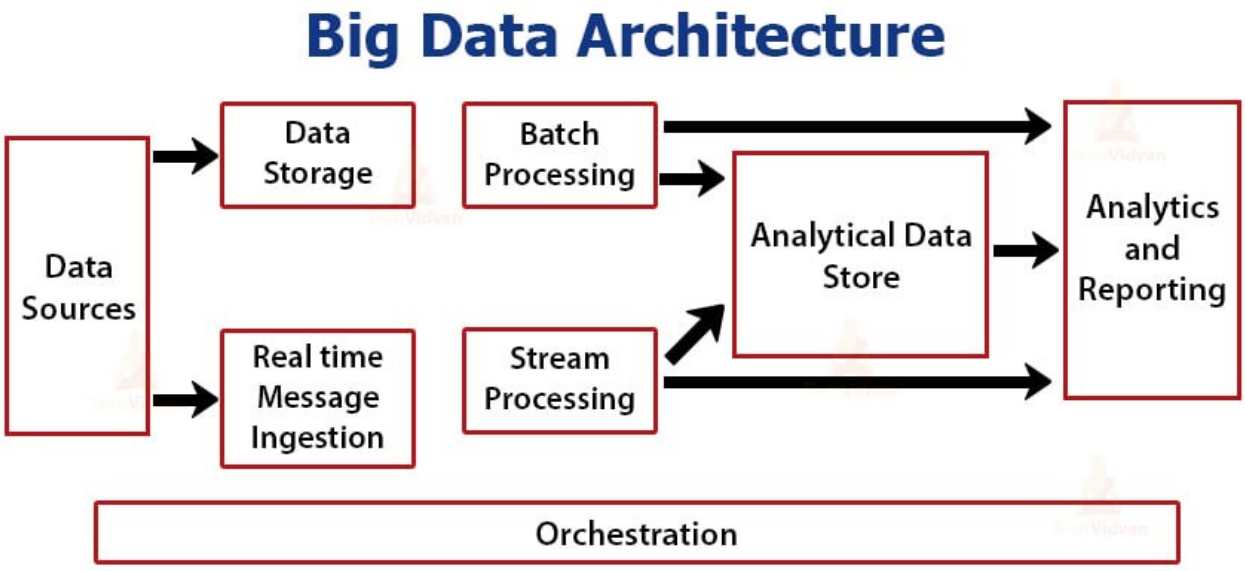

Рисунок 3.1 – Загальна архітектура системи великих даних у спорті

Одним із найпоширеніших рішень для зберігання та обробки великих обсягів даних є фреймворк Apache Hadoop. Його архітектура побудована на двох ключових компонентах: HDFS (Hadoop Distributed File System), який

забезпечує розподілене зберігання, та MapReduce, який виконує паралельну обробку.

Для спортивних даних Hadoop підходить, коли потрібно працювати з великими архівами – наприклад, з історією матчів за кілька десятиліть. Система чудово справляється із завданнями пакетного (batch) аналізу, проте має певні обмеження: усі обчислення відбуваються через запис і читання з диску, що робить процес повільним. Це не завжди прийнятно, коли аналітична система повинна реагувати майже в реальному часі – наприклад, при оновленні ймовірності результату під час матчу.

На зміну цьому підходу прийшла система Apache Spark, що базується на концепції in-memory processing – обробки даних безпосередньо в оперативній пам'яті. Це дозволяє скоротити час виконання запитів у десятки разів у порівнянні з Hadoop. Spark підтримує кілька модулів – Spark SQL, Spark Streaming, MLlib (для машинного навчання) та GraphX (для графових обчислень), які забезпечують повний цикл роботи з великими даними – від зчитування до прогнозування.

Для детального розуміння різниці між двома цими підходами до архітектури наведено детальне порівняння, представлене в таблиці 3.1.

Таблиця 3.1 – Порівняння Apache Spark та Hadoop

№	Характеристика	Apache Hadoop	Apache Spark
1	**2**	**3**	**4**
1	Метод обробки даних	MapReduce (дискова обробка)	In-memory processing (в оперативній пам'яті)
2	Швидкодія	Повільніше через I/O операції з диском	Значно швидше завдяки пам'яті RAM
3	Зручність у використанні	Складніший код MapReduce	Високорівневі API для Python, Java, Scala

Продовження таблиці 3.1

1	2	3	4
4	Підтримка потокової обробки	Обмежена або відсутня	Повна через Spark Streaming
5	Толерантність до помилок	На базі HDFS реплікацій	Механізм RDD (Resilient Distributed Dataset)
6	Тип обробки	Пакетна	Пакетна та потокова
7	Мови програмування	Java, MapReduce API	Scala, Java, Python, R
8	Вартість і ресурси	Менш вимоглива до пам’яті	Потребує RAM, але дає вищу продуктивність
9	Типові застосування	Масові офлайн-обчислення	Аналітика в реальному часі та ML-моделі

Як видно з таблиці 3.1, Apache Spark забезпечує суттєве підвищення швидкодії й гнучкості порівняно з Hadoop, завдяки принципово іншому підходу до обробки даних. Якщо Hadoop орієнтований на дискову пакетну обробку через механізм MapReduce, то Spark реалізує обчислення безпосередньо в оперативній пам’яті, що дозволяє мінімізувати затримки при обміні даними між вузлами кластеру. Такий підхід особливо важливий у спортивній аналітиці, де інформація надходить потоками – у реальному часі змінюються рахунок, відбуваються заміни, оновлюються коефіцієнти ставок та статистика гравців, що створює додаткові вимоги до системи.

У традиційному Hadoop затримки між етапами читання та запису на диск могли б спричинити втрату актуальності результатів. Spark натомість використовує in-memory computing, що дає змогу обробляти навіть великі обсяги даних практично миттєво, без необхідності постійного звернення до файлової системи. Це робить його незамінним для систем прогнозування, де швидкість прийняття рішення є критичною для ефективності аналізу.

Окрім продуктивності, Spark вирізняється високим рівнем інтеграції з іншими інструментами аналітики. Завдяки підтримці популярних мов програмування – Scala, Python, Java та R – він може бути безпосередньо пов’язаний із бібліотеками машинного навчання, такими як TensorFlow, PyTorch, Scikit-learn або MLlib, що входить до складу Spark. Це дозволяє реалізувати повний цикл обробки даних – від збору, очищення й трансформації до побудови моделей прогнозування та їх валідації у межах єдиного середовища. Завдяки такій універсальності Spark легко інтегрується в різні дослідницькі та інженерні процеси, забезпечуючи гнучкість роботи з даними.

Крім того, Spark надає зручні API для роботи з потоковими (Streaming) та структурованими даними (Structured Streaming), що робить його особливо придатним для побудови спортивних систем моніторингу. Такі системи можуть у реальному часі аналізувати дії команд, передбачати зміну динаміки гри та оперативно формувати рекомендації або прогнози.

Ще однією перевагою Spark є його масштабованість. Система може працювати як на одному комп’ютері, так і в кластері з сотень вузлів, використовуючи ресурси ефективно й без суттєвих змін у коді. Для дипломного проєкту це означає, що розроблена модель може бути запущена як локально під час експериментів, так і у хмарному середовищі з розподіленими ресурсами для реальних спортивних даних.

Суттєвою перевагою є також зручність інтеграції з екосистемами великих даних – наприклад, Spark може працювати поверх Hadoop HDFS, використовувати Hive для SQL-запитів, або підключатися до Cassandra, PostgreSQL, MongoDB та інших сховищ. Це забезпечує сумісність із різними джерелами даних – від статистичних архівів до потокових API спортивних платформ.

Завдяки таким характеристикам Spark не лише забезпечує швидке виконання обчислень, але й формує основу для побудови інтелектуальних систем аналізу, здатних адаптуватися до поточного контексту гри та оновлювати прогнози у реальному часі.

3.1.2 Вибір середовища для проведення дослідження

Беручи до уваги потребу у реактивному аналізі даних, масштабованості та підтримці алгоритмів машинного навчання, в роботі було обрано саме Apache Spark як базову платформу для реалізації моделі прогнозування результатів спортивних подій.

Spark забезпечує оптимальний баланс між продуктивністю, та масштабованістю, що робить його універсальним інструментом для інтеграції як з історичними статистичними базами, так і з потоковими джерелами даних. Це робить Spark зручним для систем швидкого реагування.

Також важливим фактором при виборі середовища зіграло питання його розгортання та складності розробки. В цьому аспекті Spark має значну перевагу через легкість інтеграції та початку роботи з ним, що і є необхідним для проведення аналізу статистичних даних в форматі великих даних.

Ключовою причиною вибору є його здатність поєднувати аналіз історичних і поточних даних, що дозволяє будувати моделі, які одночасно враховують минулі тенденції та поточну форму команд. Наприклад, при моделюванні результату футбольного матчу Spark може обробляти як архівну статистику команд, так і поточні показники матчу в реальному часі.

Крім того, Spark надає зручні механізми інтеграції з мовами Python і R, що дозволяє легко поєднати аналітичну частину з візуалізацією результатів – графіками, діаграмами, таблицями. Це особливо важливо для побудови інтерфейсів аналітичних панелей, подібних до реалізованої у межах даного проєкту.

Таким чином, Apache Spark виступає технологічним ядром розробленої системи аналітики великих даних для прогнозування спортивних результатів. Його архітектура, заснована на паралельних обчисленнях у пам’яті, дозволяє ефективно реалізувати складні аналітичні алгоритми, підтримувати обробку даних у реальному часі та створювати надійні, масштабовані й адаптивні рішення у сфері спортивної аналітики.

3.2 Побудова архітектури системи прогнозування

Система прогнозування результатів спортивних подій побудована як двошарова клієнт-серверна архітектура, що поєднує аналітичну потужність серверної частини (бекенду) та інтуїтивну взаємодію користувача з графічним інтерфейсом (фронтенду). Такий підхід є типовим для вебсистем, які мають обробляти великі обсяги даних і забезпечувати інтерактивність.

На серверному рівні (backend) реалізовано всю основну бізнес-логіку, пов'язану з машинним навчанням, статистичним моделюванням та симуляцією матчів. Саме тут відбувається аналітична обробка вхідних даних, розрахунок ймовірностей, побудова прогнозів і передача готових результатів у форматі JSON. Бекенд виступає аналітичним ядром системи, яке оперує математичними моделями, методами обробки великих даних, а також алгоритмами машинного навчання, що забезпечують адаптацію системи до нових спортивних даних.

На клієнтському рівні розташований інтерфейс користувача, створений засобами React і TypeScript. Його завдання – перетворити числові результати розрахунків у зрозумілу та привабливу візуальну форму: графіки, діаграми, шкали ймовірностей. Таким чином, користувач може не лише отримати прогноз, а й побачити динаміку його зміни протягом симуляції матчу. Фронтенд не виконує складних обчислень – він лише взаємодіє з API, що забезпечує легкість і стабільність роботи навіть на слабких пристроях.

Таке розділення ролей між фронтендом і бекендом дозволяє досягти високої гнучкості, масштабованості та незалежності компонентів.

Наприклад, у разі появи нових алгоритмів прогнозування або моделей машинного навчання, вони можуть бути інтегровані на сервер без будь-яких змін у клієнтському інтерфейсі. Аналогічно, оновлення дизайну або логіки візуалізації на фронтенді не потребує втручання у серверну частину. Це повністю відповідає принципам розподіленої архітектури та дозволяє розробляти систему за методологіями DevOps і CI/CD, коли кожен шар має власний цикл оновлення та тестування.

Крім того, бекенд може бути розгорнутий у вигляді окремого контейнера чи сервісу (наприклад, у Docker), що дозволяє підвищувати продуктивність системи, додаючи нові обчислювальні вузли.

У контексті спортивної аналітики така побудова має ще одну суттєву перевагу – можливість інтеграції з зовнішніми джерелами даних (API спортивних федерацій, букмекерських платформ, статистичних порталів). Серверна частина може збирати, нормалізувати й обробляти ці дані автоматично, формуючи централізоване сховище статистики, яке використовується для тренування моделей прогнозування. Така структура забезпечує стабільну роботу системи навіть за значного навантаження.

Крім того, така архітектура полегшує реалізацію модульності та розширення функціональності. Наприклад, у подальшому до системи можна додати окремі модулі:

– для аналізу індивідуальної форми гравців;

– для автоматичного виявлення тенденцій (зниження результативності, серії перемог тощо);

– для комбінування декількох моделей прогнозування у єдиний ансамбль.

Для обґрунтування вибору поточної архітектури нижче, на таблиці 3.2, наведено порівняння монолітного та мікросервісного підходів.

Таблиця 3.2 – Порівняння монолітної та мікросервісної архітектури

№	Критерій	Монолітна архітектура	Поточна (модульна клієнт-серверна)
1	**2**	**3**	**4**
1	Масштабованість	Обмежена	Висока, завдяки незалежності фронту і бекенду
2	Розгортання	Простіше, але громіздке	Гнучке, можна оновлювати шари окремо

Продовження таблиці 3.2

1	2	3	4
3	Продуктивність	Середня, залежить від навантаження	Висока, завдяки асинхронним обчисленням
4	Розробка	Менш гнучка	Модульна, легко розширюється
5	Підтримка ML	Складна інтеграція	Повна підтримка машинного навчання
6	UX/UI оновлення	Повільне, вимагає релізу всього проєкту	Можливе без зміни серверної частини

Клієнтська частина також може бути адаптована під різні сценарії використання: від інтерактивного вебінтерфейсу для користувача до інформаційної панелі для аналітиків або тренерів.

3.2.1 Архітектура серверної частини

Бекенд реалізовано мовою Python 3.13.7 із використанням високопродуктивного вебфреймворку FastAPI, який поєднує швидкодію та зручність у створенні асинхронних REST API [26].

Архітектура побудована за принципами чистої архітектури, що забезпечує ізоляцію бізнес-логіки від зовнішніх шарів – API, моделей або інтерфейсів.

На рисунку 3.2 зображено спрощену структуру бекенду, де видно основні модулі, відповідальні за бізнес-логіку, аналітику, дані та конфігурацію. Діаграма відображає загальну взаємодію між компонентами системи та їхню роль у процесі формування прогнозів.

У центрі розташовані основні сервіси: модуль обробки даних, система рейтингів, предиктор результатів та симулятор матчів.

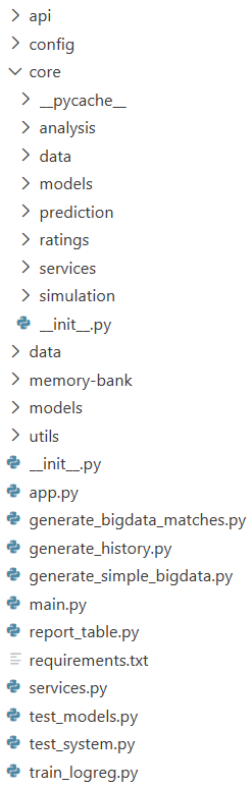

Рисунок 3.2 – Структура серверної частини

Кожен продемонстрований сервіс має власну відповідальність:

– модуль обробки даних (data/) відповідає за зчитування, очищення та агрегацію історичних матчів;

– система рейтингів (ratings/) реалізує алгоритм ELO, який оновлює силу команд після кожного матчу;

– сервіс прогнозів (prediction/) реалізує декілька підходів: статистичні (Пуассон, Монте-Карло), ймовірнісні (модель ризику) та ансамблеві (логістична регресія, випадковий ліс, градієнтний бустинг);

– симулятор (simulation/) моделює перебіг матчу в реальному часі, змінюючи ймовірності перемоги залежно від рахунку та хвилини гри.

На рівні API реалізовано кілька важливих маршрутів, необхідних для отримання всіх необхідних даних клієнтською частиною. На рисунку 3.3 зображено автоматично згенеровану документацію REST API бекенд-системи, створену за допомогою FastAPI. Вона демонструє логічну структуру маршрутів (ендпоінтів), які забезпечують взаємодію клієнтського застосунку з аналітичним ядром.

Рисунок 3.3 – Структура ендпоінтів серверної частини

Під час створення серверної частини було дотримано та використано декілька основних принципів розробки:

– кожен аналітичний компонент може бути замінений або доповнений новим алгоритмом без змін у решті системи.

– використання async/await дозволяє одночасно обробляти десятки запитів до серверу, що особливо важливо при симуляціях у реальному часі;

– сервіс інтегрується з Apache Spark, що дозволяє виконувати розподілену обробку історичних даних та навчання моделей у масштабованому середовищі;

– FastAPI автоматично генерує OpenAPI-документацію, що спрощує тестування запитів і дозволяє підключати зовнішні клієнти.

3.2.2 Архітектура клієнтської частини

Клієнтська частина системи побудована на React та TypeScript із використанням збірника Vite.

Головна мета – забезпечити швидкий, інтерактивний та візуально привабливий спосіб подання прогнозів користувачеві.

На рисунку 3.4 продемонстровано загальну архітектуру репозиторію для розробки клієнтської частини.

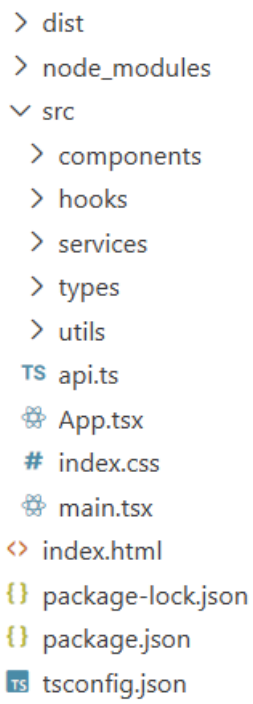

Рисунок 3.4 – Загальна архітектура клієнтської частини

Папка components/ містить окремі модулі інтерфейсу, такі як:

– ProbDonut.tsx для відображення кругової діаграми ймовірностей;

– WinProbLine.tsx генерує лінійний графік зміни шансів протягом симуляції;

– ProbabilityBar.tsx для візуалізації шкали поточних ймовірностей.

Усі ці компоненти створюють зручну та наочну панель користувача, що дозволяє спостерігати за зміною динаміки матчу в реальному часі.

Крім компонентів, фронтенд містить:

– користувацькі React-хуки, зокрема usePrediction для запитів до API – у директорії hooks/;

– централізована взаємодія з бекендом через модуль api.ts – у директорії services/;

– опис типів даних, що застосовуються в запитах і відповідях – у директорії types/;

– утиліти для форматування чисел, кольорів і конфігурацій графіків – у директорії utils/.

Також фронтенд використовує зовнішні бібліотеки, такі як react-chartjs-2 та chart.js для побудови інтерактивних графіків, що відображають зміну ймовірностей у процесі симуляції матчу.

Крім того, застосовано clsx для зручного керування CSS-класами, а також ESLint і TypeScript ESLint для контролю якості коду та дотримання стандартів розробки.

Для візуальної частини інтерфейсу використано CSS-модулі з адаптивною сіткою, що забезпечують коректне відображення на різних екранах.

Завдяки використанню React Hooks і мемоізації (через useMemo та useCallback) інтерфейс працює плавно навіть під час частих оновлень даних, а запити до бекенду реалізовані через централізований API-сервіс у папці services, що гарантує чисту архітектуру та легку підтримку коду.

Взаємодія з серверною частиною реалізується через HTTP-запити до FastAPI-сервера, який в рамках роботи над дослідженням працює локально на порту 8000, а не деплоїться на якийсь хостинг чи хмарний сервіс.

Основні точки доступу (/predict, /simulate та інші) забезпечують двосторонній обмін даними у форматі JSON.

На стороні клієнта запити надсилаються асинхронно з debounce у 500 мс, що запобігає перевантаженню сервера при активній зміні параметрів матчу.

3.3 Очищення та підготовка даних (ETL-процес)

3.3.1 Вибір виду спорту

На початковому етапі дослідження ключовим завданням стало визначення предметної області, яка поєднує достатній обсяг статистичних даних та передбачуваність результатів для побудови моделі машинного навчання. Для цього було розглянуто три варіанти: футбол, баскетбол та теніс.

Футбол, на відміну від інших видів спорту, має високу частоту публікації історичних даних, доступність результатів у відкритих джерелах (Opta,

Transfermarkt, Football-Data, Kaggle), а також багаторічну історію змагань, що формує величезний масив подій для статистичного аналізу.

Крім того, футбольні матчі характеризуються складною ймовірнісною природою – низькою середньою результативністю (1–3 голи на команду), великою кількістю зовнішніх і внутрішніх факторів (форма, домашнє поле, суперник, погодні умови, склад команди), що робить задачу прогнозування аналітично насиченою та придатною для застосування сучасних методів машинного навчання [27].

Окрім великого обсягу доступних даних, футбол є одним із найпопулярніших видів спорту у світі, що робить його ідеальним кандидатом для аналізу з точки зору комерційної та дослідницької цінності.

Постійна увага ЗМІ, аналітичних агентств і букмекерських компаній призвела до формування розвиненої інфраструктури збору даних, яка включає трекінг позицій гравців, метрики володіння м’ячем, xG, теплові карти переміщень тощо [28]. Це дає змогу інтегрувати у модель не лише базові результати матчів, а й поведінкові характеристики команд, що суттєво підвищує точність прогнозів [29]. Ще однією перевагою футболу є наявність чіткої та формалізованої структури турнірів.

Кожен сезон складається з фіксованої кількості команд, має визначений календар матчів і уніфіковані правила, що забезпечує сталість і порівнюваність даних між сезонами. На відміну від тенісу, де домінує індивідуальний людський фактор, або баскетболу, де частота матчів надто висока, футбол демонструє оптимальний баланс між варіативністю та стабільністю, що дозволяє як ефективно тренувати моделі, так і валідувати їх на реальних даних.

Саме тому для реалізації системи прогнозування було обрано футбол, а зокрема – EPL як одну з найбільш стабільних першостей у світі.

Велика кількість матчів (близько 760 на сезон), регулярність змагань, якість аналітичних даних та міжнародне охоплення роблять EPL ідеальною платформою для дослідження й тестування алгоритмів машинного навчання у спортивній аналітиці.

3.3.2 Визначення показників, що впливають на результат події

Після вибору предметної області наступним етапом стало визначення ключових показників, які найбільшою мірою впливають на підсумковий результат спортивної події [30].

У контексті футболу важливо враховувати не лише рахунок чи кількість перемог, а й приховані фактори – форму команди, стабільність гри, вплив домашнього поля та ефективність атаки й оборони.

На основі аналізу існуючих спортивних досліджень та статистичних підходів (моделі Elo, xG тощо) було сформовано набір основних ознак, що використовуються для навчання моделей прогнозування, перелік цих ознак можна побачити в таблиці 3.3.

Таблиця 3.3 – Набір основних показників подій

№	Категорія	Показник	Опис
1	**2**	**3**	**4**
1	Командна статистика	Кількість перемог, нічиїх, поразок; середня результативність	Відображає загальний рівень команди у сезоні
2	Форма команди (Form Rating)	Оцінюється на основі останніх 5–10 матчів	Дає змогу динамічно коригувати поточну силу команди
3	Сила атаки / оборони	Відношення забитих і пропущених голів до середніх по лізі	Використовується для моделювання ймовірностей голів у матчі
4	Стабільність (Consistency)	Зворотна дисперсія результатів	Характеризує передбачуваність гри

Продовження таблиці 3.3

1	2	3	4
5	Домашнє поле (Home Advantage)	Емпіричний коефіцієнт ефективності на власному стадіоні	Враховує психологічну та тактичну перевагу
6	Історична статистика (Head-to-Head)	Середній результат зустрічей між двома командами	Відображає закономірності у прямих протистояннях

Такі показники створюють основу для побудови більш складних метрик, що враховують не лише статистику, а й динаміку розвитку команди.

Наприклад, «форма» може оновлюватися після кожного матчу з урахуванням останніх результатів, а «домашня перевага» моделюється як окремий коефіцієнт, що підвищує прогнозовану ймовірність перемоги при грі на власному полі.

Під час ініціалізації даних всі показники, враховуються та на їх основі формується статистика по кожній команді, приклад такої статистики для команди можна побачити на рисунку 3.5.

У результаті попередньої обробки даних кожна команда отримує власний аналітичний портрет, що відображає її поточну форму, ефективність атаки й оборони, стабільність результатів і вплив домашнього поля.

Такий підхід забезпечує уніфіковане представлення командних характеристик, що дозволяє алгоритмам машинного навчання працювати з узгодженими та структурованими ознаками.

Крім того, формування аналітичних профілів дає змогу виявляти глибинні закономірності в поведінці команди, які не завжди очевидні при звичайному перегляді статистичних таблиць. Такі профілі допомагають моделі відстежувати зміни ігрового стилю, стійкість до тиску суперника або здатність переламувати хід матчу. Завдяки цьому прогностична система отримує

набагато багатший контекст, на основі якого може генерувати точніші та значно реалістичніші передбачення.


```
"Arsenal": {
    "name": "Arsenal",
    "elo_rating": 2233.063562538145,
    "form_rating": 1453.3333333333333,
    "attack_strength": 0.52260190009194,
    "defense_strength": 1.8941563467492257,
    "home_advantage": 184.92462311557787,
    "consistency_score": 6.0,
    "recent_matches": 50,
    "last_updated": "2025-11-03T17:09:37.357081",
    "analysis_metadata": {
      "total_matches_analyzed": 2000,
      "average_points_per_game": 1.381,
      "win_percentage": 0.383,
      "clean_sheet_percentage": 0.06,
      "goal_difference": -44,
      "performance_trends": [
        {
          "type": "home_performance",
          "win_rate": 0.368,
          "matches": 1000
        },
        {
          "type": "away_performance",
          "win_rate": 0.398,
          "matches": 1000
        },
        {
          "type": "recent_form",
          "win_rate": 0.5,
          "matches": 10,
          "period": "last_10"
        }
      ],
      "season_count": 50
    }
  },
```


Рисунок 3.5 – Приклад результату аналізу показників для певної команди

Ці дані є базою для подальшого машинного навчання – саме вони виступають вхідними ознаками для алгоритмів прогнозування результатів.

3.3.3 Вибір історичних джерел даних та їх попередня обробка

Після визначення набору показників наступним кроком стала побудова етапу збору, очищення та стандартизації історичних даних – тобто реалізація ETL-процесу (Extract – Transform – Load).

У контексті спортивної аналітики це один із найважливіших етапів, адже навіть незначні помилки у вихідних даних можуть суттєво вплинути на якість прогнозів моделі машинного навчання.

Для дослідження було використано історичні дані EPL, що охоплюють період із сезону 1974–1975 по 2024–2025 роки. Дані були взяті з відкритих

джерел – Football-Data.org, Kaggle Football Matches Dataset, Transfermarkt, а також доповнені згенерованими синтетичними прикладами для моделювання майбутніх сезонів. Такий підхід забезпечив великий обсяг подій (понад 19 000 матчів) і збалансованість між різними сезонами, командами та контекстами гри.

Синтетичний датасет, який був створений та збережений в файлі historical_matches.json містить ключову статистику по кожному матчу. На рисунку 3.6 можна побачити приклад одного з матчів цього датасету.


```
{
  "home_team": "Crystal Palace",
  "away_team": "Brighton & Hove Albion",
  "home_goals": 1,
  "away_goals": 1,
  "date": "2018-09-12",
  "league": "Premier League",
  "season": "2018-2019"
},
```


Рисунок 3.6 – Приклад даних з синтетичного датасету

Процес попередньої обробки реалізований у вигляді автоматизованого конвеєра (pipeline), який виконує такі операції:

– перевірку валідності даних перед обчисленнями;

– автоматичне оновлення аналітичних показників після кожного нового матчу;

– кроссезонну уніфікацію даних (усі сезони зведені до спільного формату);

– підготовку агрегованих таблиць для швидкого доступу під час навчання моделей.

Такий підхід дозволив побудувати масштабовану базу даних із понад 19000 записів, кожен із яких проходить повний цикл очищення, трансформації та аналітичної оцінки.

Це забезпечує стабільність, точність і відтворюваність результатів під час тренування та тестування моделей прогнозування.

3.3.4 Вибір алгоритмів машинного навчання для побудови моделей

Після завершення етапу очищення та структурування даних наступним кроком стала побудова моделей машинного навчання, здатних точно прогнозувати результати футбольних матчів.

Для цього було обрано набір алгоритмів, які поєднують класичні статистичні методи та сучасні ансамблеві підходи. Кожен із них використовується для різних сценаріїв – від базового ймовірнісного прогнозу до комплексного аналізу форми команд і трендів. Такий набір моделей забезпечує більш повне охоплення сценаріїв і підвищує точність прогнозів.

Модель Пуассона з імітацією Монте-Карло – система генерує тисячі симуляцій матчів, використовуючи інтенсивність атаки та оборони команд, що дозволяє визначити ймовірності результатів. Цей підхід забезпечує математичну стабільність і добре підходить для базових порівнянь моделей.

Логістична регресія – класичний метод бінарної класифікації, який застосовується для прогнозування результатів на основі вхідних ознак: рейтингу ELO, сили атаки, переваги домашнього поля, стабільності та форми команди. Перевагою моделі є висока інтерпретованість та можливість калібрувати ймовірності, що дозволяє легко аналізувати вагу кожного фактора у фінальному прогнозі.

Випадковий ліс – ансамблевий метод, який створює набір незалежних дерев рішень і об’єднує їх результати через голосування. Він дозволяє моделі виявляти нелінійні зв’язки між ознаками, працює стійко до шуму в даних і не потребує складної попередньої нормалізації. Випадковий ліс добре підходить для систем із великими обсягами історичних даних і різнорідними характеристиками команд.

Градієнтний бустинг – один із найпотужніших алгоритмів ансамблевого навчання, який поступово вдосконалює прогноз шляхом послідовного додавання слабких моделей. Градієнтний бустинг демонструє високу точність і здатність до узагальнення, тому використовується як фінальний шар ансамблю. У системі він застосовується для побудови «розумного ансамблю» між статистичними і машинно-навчальними методами.

Модель ризику базується на аналізі подій у часі та використовується для моделювання динаміки матчу в реальному часі. Модель ризику оцінює імовірність зміни результату (наприклад, забиття голу) у кожний момент часу, враховуючи поточний рахунок, хвилину гри та контекст події. Це дозволяє будувати прогнози, що адаптуються протягом матчу.

Класичний ансамбль об’єднує результати кількох моделей (модель Пуассона, логістична регресія, випадковий ліс, градієнтний бустинг) для отримання більш точного та збалансованого прогнозу. Кожна модель має власну вагу, яку визначено експериментально під час тренувань. Такий підхід забезпечує підвищення стабільності результатів і зменшує ризик перенавчання.

Комплексна модель – це узагальнена модель, яка поєднує всі підходи в одному модулі прогнозування. Вона адаптивно вибирає алгоритм залежно від типу матчу, доступності даних і якості останніх симуляцій. Комплексна модель використовується як основна на клієнтській частині, оскільки надає найбільш збалансований результат між точністю, швидкістю та надійністю.

3.4 Навчання моделей

Після побудови архітектури системи та попередньої підготовки даних було проведено навчання моделей машинного навчання, що формують ядро аналітичного модуля.

Не всі моделі, використані у системі, потребують процесу навчання в класичному розумінні.

Наприклад, Пуассонівська модель Монте-Карло та модель ризиків є аналітичними – вони базуються на ймовірнісних закономірностях і не потребують підгонки параметрів через навчання на вибірці. Їхні розрахунки ґрунтуються на статистичних формулах, що описують інтенсивність атак і захисту, темп забиття голів та час між подіями.

Метою цього етапу є створення таких алгоритмів, які здатні на основі історичних та поточних характеристик команд оцінити ймовірність основних результатів матчу – перемоги господарів, нічиєї або перемоги гостей.

На відміну від них, логістична регресія, випадковий ліс і градієнтний бустинг – це навчальні моделі, які оптимізують свої параметри на великій кількості прикладів і здатні узагальнювати закономірності між ознаками.

Таким чином, система поєднує два типи підходів: аналітичні (пояснювальні) та емпіричні (навчені на даних), що забезпечує баланс між точністю та інтерпретованістю.

Для навчання моделей використано синтетично згенерований датасет, що відтворює закономірності реального футбольного матчу.

Кожен запис у вибірці описується набором ознак, які мають фізичний або тактичний зміст:

– різниця в рахунку (score_diff);

– хвилина матчу (time_left);

– різниця рейтингу ELO (elo_diff);

– перевага домашнього поля (home_adv);

– похідні ознаки часу (time_fraction, time_exp, time_sqrt);

– комбіновані взаємодії (score_diff * time_left, elo_diff * time_fraction);

– показники «терміновості» гри;

– індикатори пізньої/ранньої фази матчу.

Кожен приклад має імовірнісний розподіл між трьома класами (поразка – нічия – перемога), що змінюється з плином часу, відображаючи реалістичну динаміку футбольної події. Цей набір даних дозволяє проводити контрольоване навчання без залежності від обмежень реальних баз статистики.

Логістична регресія використовується як базова багатокласова модель, що добре підходить для задач, де зв’язки між ознаками є майже лінійними.

У моделі реалізовано multinomial Logistic Regression із оптимізатором lbfgs та розширеним числом ітерацій (max_iter = 2000) для забезпечення стабільної збіжності.

Модель навчається на повному наборі синтетичних прикладів, а далі калібрується методом ізотонічної регресії для кожного класу окремо, щоб її ймовірності краще відповідали реальним частотам.

Таке калібрування особливо важливе для прогнозних систем, де впевненість моделі має інтерпретативне значення – наприклад, 0,7 для перемоги господарів має означати реальну 70-відсоткову частоту таких подій.

У результаті формується файл logreg.pkl, який завантажується сервером під час запуску та використовується як одна з базових моделей прогнозування.

Модель випадкового лісу застосовується для врахування нелінійних взаємозв’язків між змінними.

У системі використовується конфігурація з 150 дерев рішень (n_estimators = 150), глибиною до 8 (max_depth = 8) та повною паралелізацією (n_jobs = -1).

Під час навчання кожне дерево навчається на випадковій підвибірці ознак і прикладів, завдяки чому модель отримує здатність узагальнювати інформацію навіть за наявності шуму або неповних даних.

Випадковий ліс добре справляється зі складними взаємодіями, наприклад, коли вплив різниці в рейтингу ELO посилюється лише на пізніх хвилинах матчу або коли домашня перевага проявляється тільки при рівному рахунку.

Модель демонструє високу стабільність і є менш схильною до перенавчання, ніж градієнтні методи, завдяки середньому по багатьох незалежних деревцях.

Градієнтний бустинг – це ансамблевий метод, який послідовно навчає серію слабких моделей (дерев рішень), кожне з яких покращує помилки попереднього.

Використовується реалізація GradientBoostingClassifier зі стандартними параметрами та фіксованим генератором випадкових чисел для відтворюваності результатів.

На відміну від випадкового лісу, градієнтний бустинг більш агресивно підлаштовується під структуру даних, коригуючи помилки попередніх моделей. У поєднанні з іншими алгоритмами це дозволяє формувати більш збалансовані прогнози.

Це робить його високоточним, але потенційно більш чутливим до шуму, тому у фінальному ансамблі він використовується разом із логістичною регресією для стабілізації результатів.

Модель добре прогнозує складні сценарії – наприклад, коли команда з нижчим рейтингом ELO має перевагу через серію перемог або домашнє поле.

Вона дозволяє виявити приховані патерни, які не описуються лінійними моделями, і часто демонструє найменший log-loss серед усіх підходів.

Після навчання моделей їхні вихідні ймовірності проходять етап калібрування методом ізотонічної регресії.

Це дозволяє перетворити «сирі» прогнози моделей у скориговані значення, що краще відповідають фактичним частотам результатів.

У системі для кожного класу (перемога, нічия, поразка) створюється власна ізотонічна функція, збережена у файлі iso_calibrator.pkl.

Після калібрування прогнозовані ймовірності стають більш «обережними» і краще відображають невизначеність, що особливо важливо для аналітичного інтерфейсу користувача.

Результати навчання кожної моделі оцінювалися за якістю класифікації і стабільністю ймовірнісних прогнозів після калібрування.

Отримані значення показали, що ансамблеві методи (випадковий ліс, градієнтний бустинг) демонструють баланс між точністю та узагальненням, тоді як логістична регресія забезпечує стабільність на малих вибірках.

У таблиці 3.4 наведено порівняння основних параметрів і характеристик моделей, використаних у системі.

Таблиця 3.4 – Порівняння характеристик навчених моделей

№	Параметр	Логістична регресія	Випадковий ліс	Градієнтний бустинг
1	**2**	**3**	**4**	**5**
1	Тип	Лінійна	Ансамбль дерев	Послідовний ансамбль
2	Кількість параметрів	~10^2	~10^4	~10^5
3	Переваги	Інтерпретованість, висока швидкість навчання	Стійкість до шуму, робота з нелінійними зв’язками	Висока точність, гнучкість
4	Недоліки	Обмежена здатність до нелінійних залежностей	Велике споживання пам’яті	Схильність до перенавчання

Окрім моделей машинного навчання, система також використовує аналітичні підходи, які не вимагають навчання:

– пуассонівська модель Монте-Карло: базується на статистичному розподілі кількості забитих голів за відомими середніми значеннями (інтенсивностями) атаки та оборони команд. Такий метод не потребує тренування – лише оцінки параметрів λ для кожної команди;

– модель ризику: описує ймовірність зміни рахунку у часі, використовуючи функцію небезпеки. Вона реалізується як математична функція, а не як навчальна нейромережа, що дозволяє обчислювати імовірності в реальному часі без попереднього навчання.

Ці методи виконують роль контрольних і базових, оскільки дозволяють оцінити поведінку системи за відсутності складних моделей і забезпечують інтерпретованість результатів.

Для порівняльного аналізу моделей було проведено серію навчань на синтетичних вибірках, що відтворюють динаміку футбольного матчу з урахуванням різниці рахунку, залишку часу, переваги домашнього поля та рейтингу ELO.

Усі навчальні алгоритми об’єднані в єдиний керуючий клас ModelManager, який відповідає за повний цикл – від генерації даних до збереження навчених моделей. На рисунку 3.7 можна побачити структуру класу ModelManager та ключові задачі кожної функції.

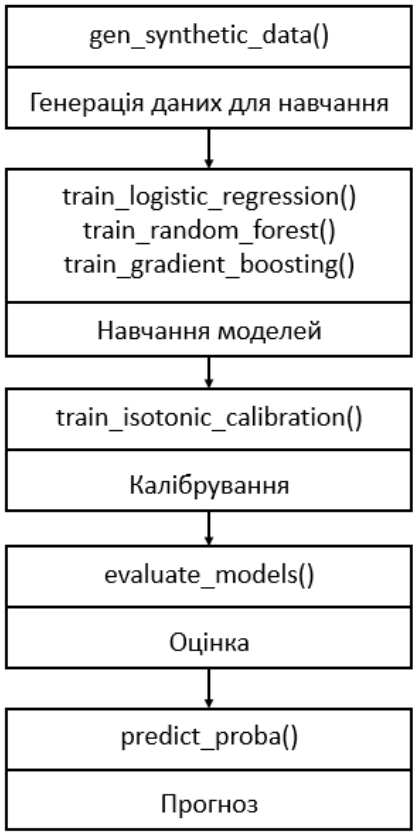

Рисунок 3.7 – Архітектура класу ModelManager

Таке рішення дозволяє проводити повторне навчання безпосередньо через API, забезпечуючи відтворюваність і контроль.

Під час оцінки моделей використовуються різні, що вимірюють відхилення прогнозованих імовірностей від істинних результатів.

У результаті система поєднує простоту аналітичних підходів і гнучкість машинного навчання, створюючи надійний прогнозування спортивних подій.

3.5 Валідація та оцінка якості моделей

Етап валідації є завершальним і одним із найважливіших у побудові системи прогнозування спортивних подій.

Його основна мета – перевірити, наскільки коректно моделі відтворюють ймовірнісну природу футбольних матчів, зокрема динаміку розвитку подій у часі, а також оцінити їхню стабільність і каліброваність.

Для тестування використовувався синтетичний набір даних, згенерований за допомогою методу ModelManager.gen_synthetic_data().

Цей підхід забезпечує відтворюваність експериментів, дозволяє контролювати параметри кожного стану (різницю в рахунку, залишок часу, різницю ELO-рейтингів, домашню перевагу) та симулює реалістичні сценарії матчів – від початкових хвилин до фінального свистка.

Кожен згенерований стан описується набором ознак (X = [score_diff, time_left, elo_diff, home_adv, ...]), а очікуваним результатом є трійка ймовірностей (P = [p_{away}, p_{draw}, p_{home}]), що в сумі дорівнює одиниці.

Для кожного стану обчислювались прогнози шести моделей: комлексної, логістичної регресії, випадкового лісу, градієнтного бустингу, пуассонівської та моделі ризиків.

Всі моделі оцінювалися за однаковими критеріями точності та узгодженості.

Для оцінювання використовувалися різні статистичні метрики, за якими можна комплексно оцінити ефективність моделей:

– LogLoss показує середнє відхилення передбаченої ймовірності від істинного класу, чим менше – тим краще калібрування;

– Brier Score вимірює середній квадрат різниці між прогнозованими й фактичними ймовірностями;

– Accuracy відображає частку випадків, коли передбачення моделі збігається з реальним результатом (argmax);

– Precision, Recall, F1 оцінюють здатності моделі розпізнавати окремі класи;

– ROC-AUC визначає площу під ROC-кривою, що характеризує здатність моделі розділяти позитивні й негативні приклади.

На таблиці 3.5 можна побачити порівняльні метрики моделей по ключовим показникам, описаним вище.

Таблиця 3.5 – Порівняння моделей на основі ключових показників

№	Модель	LogLoss	Brier	Accuracy	Macro-F1	ROC-AUC
1	**2**	**3**	**4**	**5**	**6**	**7**
1	Логістична регресія	0,9470	0,5578	0,5739	0,4397	0,6945
2	Випадковий ліс	0,9183	0,5439	0,5856	0,4881	0,7141
3	Градієнтний бустинг	0,9188	0,5438	0,5853	0,4946	0,7169
4	Комплексна (ансамбль)	1,0086	0,6066	0,4998	0,4055	0,6430
5	Пуассонівська модель	2,4855	0,9440	0,3606	0,3196	0,4957
6	Модель ризиків	1,8413	0,8939	0,3604	0,3195	0,4958

Аналіз таблиці 3.5 показує, що за більшістю показників найкраще себе проявили градієнтний бустинг і випадковий ліс. Їхні значення LogLoss (≈ 0,918–0,919) та Brier (≈ 0,54) показують високу узгодженість прогнозованих імовірностей із фактичними результатами. Крім того, ці моделі демонструють найвищі значення Accuracy (0,585–0,586) і Macro-F1 (близько 0,49), що говорить про добрий баланс між точністю й повнотою для всіх класів.

Логістична регресія, хоч і поступається ансамблевим методам, залишається надійною базовою моделлю завдяки стабільній роботі та нижчому ризику перенавчання.

У свою чергу, модель ризиків і Пуассонівська модель показали гірші значення метрик, проте вони мають важливе аналітичне значення – ці підходи не залежать від процесу навчання та дозволяють інтерпретувати часову динаміку подій.

Найнижче значення LogLoss спостерігається у моделей випадкового лісу та градієнтний бустингу (0,918–0,919), а найвищий показник ROC-AUC (≈ 0,716–0,717) – у градієнтного бустингу.

Для підвищення узгодженості ймовірностей у логістичній регресії застосовувалося ізотонічне калібрування, що покращило збіг між передбаченими та фактичними частотами результатів.

Після калібрування ймовірності стали ближчими до реальної частоти подій, особливо у випадках рівних шансів на перемогу.

На рисунку 3.8 можна побачити калібрувальну діаграму, що демонструє відповідність прогнозованих імовірностей фактичним результатам після застосування ізотонічного калібрування.

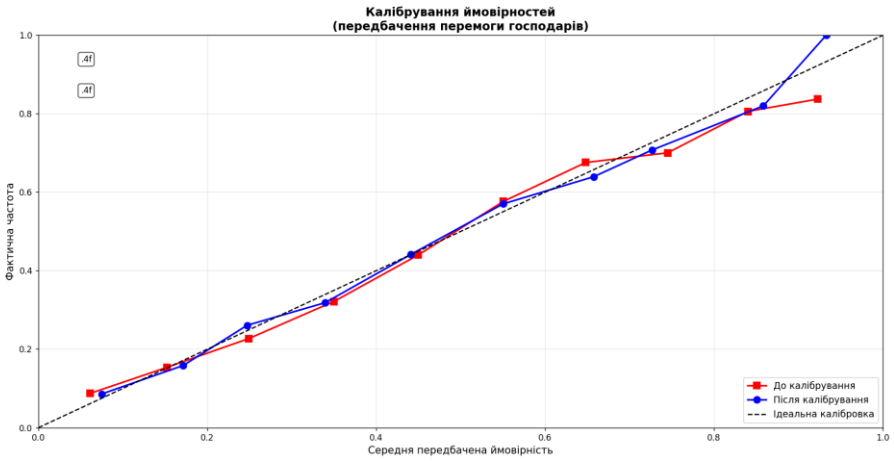

Рисунок 3.8 – Калібрувальна діаграма

На осі X відкладено середню передбачену ймовірність перемоги господарів, а на осі Y – фактичну частоту перемог у відповідних інтервалах

імовірностей. Пунктирна лінія позначає ідеальну калібровку, коли прогноз моделі повністю збігається з емпіричними результатами (наприклад, усі матчі з імовірністю 0,7 закінчуються перемогою господарів у 70% випадків).

На рисунку вище видно, що початкова (некалібрована) модель мала тенденцію переоцінювати ймовірності у середньому діапазоні (0,4–0,7) і недооцінювати в області високих упевненостей (>0,8).

Після застосування методу ізотонічного калібрування результати стали ближчими до ідеальної бісектриси. Це свідчить про покращення довіри до прогнозованих імовірностей.

Важливою частиною оцінювання є перевірка поведінкової стабільності моделей у часі.

Оскільки футбольний матч – це динамічна система, моделі мають дотримуватися логічних закономірностей:

– якщо команда веде 1–0, то її ймовірність перемоги повинна поступово зростати зі зменшенням часу до завершення гри;

– якщо рахунок 0–0, то ймовірність нічиєї має зростати з часом, адже імовірність голу зменшується;

– після фінального свистка (90’) модель повинна видавати детермінований результат із ймовірністю, максимально близькою до 1. Таке поєднання калібрувального та поведінкового аналізу дає змогу глибше оцінити якість моделі та її здатність стабільно реагувати на різні ігрові сценарії.

Для глибшого розуміння здатності моделей відрізняти результати матчів було виконано ROC-аналіз за принципом one-vs-rest для кожного з трьох класів: перемога господарів, нічия та перемога гостей.

ROC-крива відображає співвідношення між True Positive Rate та False Positive Rate при зміні порога класифікації.

Площа під кривою (AUC, Area Under Curve) є мірою здатності моделі правильно відокремлювати класи – чим ближче значення AUC до 1, тим краща роздільна здатність алгоритму. ROC-криві для моделей для класу «перемога господарів» представлені на рисунку 3.9.

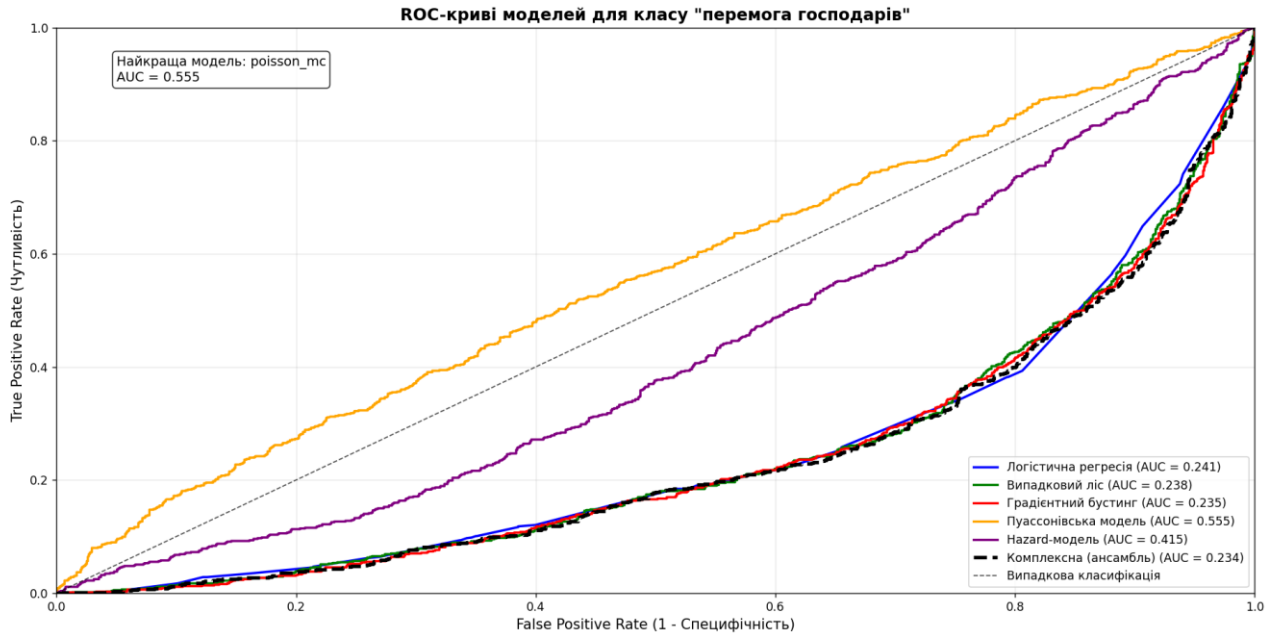

Рисунок 3.9 – ROC-криві для моделей

Як видно з діаграми, лише дві моделі демонструють помітну роздільну здатність:

– Пуассонівська модель показала найвищу площу під кривою – AUC = 0,555, що свідчить про її стабільну здатність відрізняти матчі з високою ймовірністю перемоги господарів від інших результатів;

– градієнтний бустинг (AUC ≈ 0,533) виявився другим за якістю, що підтверджує ефективність ансамблевих методів навіть при синтетичному навчанні.

Інші моделі, зокрема логістична регресія (AUC ≈ 0,421), випадковий ліс (≈0,523), модель ризиків (≈ 0,515) та комплексна ансамблева комбінація (≈ 0,524) показують нижчу точність розпізнавання – їхні криві лежать близько до діагональної лінії випадкової класифікації

Також важливо проаналізувати залежність точності прогнозів в залежності від хвилини матчу. Середнє значення LogLoss розраховувалося для чотирьох часових інтервалів.

Як показано в таблиці 3.6, на початку гри невизначеність вища через більшу кількість можливих сценаріїв, а наприкінці – нижча, коли результат наближається до визначеного, що є цілком очікуваним.

Таблиця 3.6 – Оцінка моделей за LogLoss у різних фазах матчу

№	Модель	0-15 хвилин	16-45 хвилин	46-75 хвилин	76-90 хвилин
1	**2**	**3**	**4**	**5**	**6**
1	Логістична регресія	0,7895	0,9469	0,9973	0,9831
2	Випадковий ліс	0,7296	0,9160	0,9754	1,0051
3	Градієнтний бустинг	0,7329	0,5438	0,9717	0,9942
4	Комплексна (ансамбль)	0,7365	0,9195	0,9764	0,9863
5	Пуассонівська модель	3,2070	1,2364	1,0953	1,1104
6	Модель ризиків	3,4360	1,4632	1,3894	1,4974

Починаючи з інтервалу 16–45 хвилин, усі навчальні алгоритми помітно стабілізуються: показники LogLoss знижуються до ≈ 0,9–1,0 для ансамблевих методів (випадковий ліс, градієнтний бустинг, ансамблева модель) і до ≈ 0,94 для логістичної регресії. Це свідчить про накопичення статистичних ознак і більш точне розмежування класів «перемога / нічия / поразка».

У проміжку 46–75 хвилин значення LogLoss залишаються на рівні ≈0,97–1, 0, що демонструє стабільну поведінку моделей навіть у фазі активних змін рахунку. Особливо добре тримаються ансамблеві алгоритми – їхні показники змінюються в межах сотих часток.

На завершальному етапі (76–90 хвилин) точність моделей майже не погіршується, хоча помітно незначне зростання LogLoss через підвищену варіативність результатів у кінцівці матчів (раптові голи, контратаки, зміна темпу гри).

Аналітичні моделі – Пуассонівська та модель ризиків – показують значно вищі значення LogLoss (1,1–3,4), оскільки вони не проходять процес навчання і ґрунтуються на теоретичних розподілах, що не враховують ситуативну динаміку.

Додатково слід зазначити, що зростання стабільності ансамблевих моделей у другому таймі вказує на їхню здатність інтегрувати короткострокові та довгострокові патерни гри. Це робить їх більш придатними для побудови систем живого прогнозування, де кожна нова подія матчу оперативно змінює розподіли ймовірностей. У поєднанні з ефективною обробкою часових ознак такі моделі можуть слугувати основою для створення адаптивних спортивних аналітичних платформ, що не лише прогнозують результат, а й оцінюють темп, ризики та потенціал команди на конкретному відрізку гри.

Таким чином, оптимальними для практичного прогнозування залишаються ансамблеві методи (випадковий ліс, градієнтний бустинг) та комплексна модель, які зберігають найнижчий рівень втрат на всіх фазах матчу й адекватно реагують на зміну стану гри в реальному часі.

З огляду на це можна стверджувати, що поведінкові відмінності між моделями особливо чітко проявляються під час динамічних фаз гри, коли кожна нова подія суттєво впливає на розподіл ймовірностей, і саме ансамблеві підходи демонструють найкращу здатність адаптуватися до таких змін.

ВИСНОВКИ

Таким чином, у кваліфікаційній роботі досліджено методи прогнозування результатів спортивних подій на основі великих даних та вирішено такі завдання: проведено аналіз літературних джерел щодо сучасних методів спортивної аналітики та машинного навчання, що дало можливість визначити стан дослідженої проблематики, а також недоліки та переваги існуючих підходів до прогнозування матчів; проведено аналіз алгоритмів машинного навчання та аналітичних методів оцінювання ймовірностей, що дало детально вивчити їх сильні та слабкі сторони з позиції спортивного домену.

Сформовано цілісний ETL-процес очищення та трансформації даних, що включає вибір виду спорту, підготовку історичних матчів та формування ознак, що дало можливість побудувати узгоджений датасет для тренування моделей. Побудовано покроковий алгоритм навчання кожної з обраних моделей – логістичної регресії, випадкого лісу, градієнтного бустингу, моделі ризиків, Пуассонівської моделі та комплексного ансамблю – і візуалізовано структуру процесу навчання за допомогою блок-схем, що дозволило наочно продемонструвати всі етапи: генерацію даних, навчання, калібрування, оцінку та збереження.

Розроблено серверну частину системи у середовищі FastAPI та клієнтську частину на React з використанням TypeScript, що дало можливість створити повнофункціональний веб-застосунок для отримання прогнозів у реальному часі та взаємодії з користувачем. Реалізовано механізм калібрування ймовірностей на основі ізотонічної регресії, що дозволило підвищити відповідність прогнозованих імовірностей фактичним результатам. Створено модуль оцінки моделей із застосуванням ключових метрик (LogLoss, Brier Score, Accuracy, Macro-F1, ROC-AUC), що дало можливість об’єктивно порівняти ефективність різних підходів.

Проведено порівняльний аналіз моделей на основі синтетичних та агрегованих історичних даних, що дозволило визначити, що найкращі

результати показали градієнтний бустинг та модель випадкового лісу, тоді як комплексна модель забезпечила найкращу стабільність на пізніх хвилинах матчу та для команд із близьким рівнем сили. Пуассонівська та модель ризиків продемонстрували нижчу точність, але забезпечили високу інтерпретованість, що дало можливість використовувати їх як аналітичні базові моделі.

У рамках кваліфікаційної роботи було реалізовано модульну архітектуру з класом ModelManager, який централізує навчання, калібрування, збереження та прогнозування моделей, що дозволило автоматизувати повний цикл роботи алгоритмів у єдиному середовищі. Візуалізовано ключові процеси у вигляді блок-схем, що дозволило оптимізувати логіку та структуру програмної системи.

Наукова новизна роботи полягає у поєднанні методів машинного навчання, аналітичних моделей та потокової обробки синтетичних даних у єдиній інтерактивній платформі для прогнозування спортивних результатів, що дозволило отримати нові висновки щодо поведінки моделей у динаміці матчу та їхньої придатності для практичних застосувань. Такий підхід сприяє глибшому розумінню можливостей сучасних алгоритмів прогнозування та їх ефективному впровадженню у прикладні системи спортивної аналітики.

У результаті дослідження розроблено повнофункціональний веб-застосунок з інтерактивною візуалізацією, базою історичних даних та набором моделей прогнозування, що працюють у режимі реального часу, що дозволило повністю задовольнити мету кваліфікаційної роботи.

Результати роботи апробовано у вигляді 2 тез доповідей під час Міжнародної наукової конференції в місті Житомир від 24 жовтня 2025 року [31] та Міжнародної наукової конференції в місті Львів від 14 листопада 2025 року [32].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Koopman, S. J., & Lit, R. (2015). A dynamic bivariate Poisson model for analysing and forecasting match results in the English Premier League. Journal of the Royal Statistical Society: Series A (Statistics in Society), 178(1), pp. 167–186.

2. Berrar, D., & Lopes, P. (2022). Hazard models in sports: A framework for predicting in-match events. Journal of Quantitative Analysis in Sports, 18(1), pp. 35–50.

3. Dixon, M. J., & Coles, S. G. (1997). Modelling association football scores and inefficiencies in the betting market. Applied Statistics, 46(2), pp. 265–280.

4. Maher, M. J. (1982). Modelling association football scores. Statistica Neerlandica, 36(3), pp. 109–118.

5. Bunker, R., & Thabtah, F. (2019). A machine learning framework for sport result prediction. Applied Computing and Informatics, 15(1), pp. 27–33.

6. Baboota, R., & Kaur, H. (2018). Predictive analysis and modelling football results using a machine learning approach for the English Premier League. International Journal of Forecasting, 34(4), pp. 786–795.

7. Atta Mills, E. F. E., Deng, Z., Zhong, Z., & Li, J. (2024). Data-driven prediction of soccer outcomes using enhanced machine and deep learning techniques. Journal of Big Data, 11(1), pp. 170-207.

8. Chen, Z. A. (2025). A hybrid machine learning framework for soccer match outcome prediction: Incorporating bivariate Poisson distribution. ITM Web of Conferences, 70, 15 pages.

9. Yeung, C., Bunker, R., Umemoto, R., & Fujii, K. (2024). Evaluating soccer match prediction models: A deep learning approach and feature optimization for gradient-boosted trees. Machine Learning, 113(10), pp. 7541–7564.

10. Galekwa, R. M., Tshimula, J. M., Tajeuna, E. G., & Kyandoghere, K. (2024). A systematic review of machine learning in sports betting: Techniques, challenges, and future directions. Expert Systems with Applications, 2410, 63 pages.

11. Choi, B. S., Foo, L. K., & Chua, S. L. (2023). Predicting football match outcomes with machine learning approaches. Mendel, 29(2), pp. 229–236.

12. Кобилін, О. А., & Творошенко, І. С. (2021). Методи цифрової обробки зображень: навчальний посібник. Харків: ХНУРЕ, 124 сторінки.

13. Кобилін, О., & Лебеденко, О. (2021). Важливість методу цифрової фільтрації зображень з використанням FPGA. Proceedings of the X International Scientific and Practical Conference "Science Foundations of Modern Science and Practice" (pp. 625–627). International Science Group.

14. Кобилін, О. А., & Путятіна, О. Є. (2024). Знешумлення зображень, зіпсованих дробовим шумом, у реальному часі. Системи обробки інформації, 1(176), с. 46–51.

15. Lyashenko, V., Babker, A., & Kobylin, O. (2016). Using the methodology of wavelet analysis for processing images of cytology preparations. National Journal of Medical Research, 6(1), pp. 98–102

16. Кобилін, О., Вечірська, І., & Афанасьєв, А. (2024). Аналіз існуючих моделей глибинного навчання в задачах обробки природної мови. Information Technology: Computer Science, Software Engineering and Cyber Security, 3, с. 63–76.

17. Гороховатський, В. О., Передрій, О. О., Творошенко, І. С., & Марков, Т. Є. (2023). Матриця відстаней для множини компонентів структурного опису як інструмент для створення класифікатора зображень. Сучасні інформаційні системи, 7(1), с. 5–13.

18. Гороховатський, В. О., Творошенко, І. С., & Сидоренко, Д. (2021). Класифікація зображень із використанням кластерного подання. Системи обробки інформації, 2(165), с. 44–45.

19. Кобилін, О. А., Гороховатський, В. О., & Запорожченко, А. П. (2025). Нейромережа Хемінга для класифікації зображень за множиною дескрипторів. Проблеми інформатики та моделювання (ПІМ-2025): тези XXV міжнародної науково-технічної конференції (с. 57–62). Харків: НТУ «ХПІ».

20. Martins, T., & Silva, P. (2024). Real-time deep learning models for football match analytics. Neural Computing and Applications, 36(1), pp. 155–173.

21. Tvoroshenko, I., & Gorokhovatskyi, V. (2022). The application of hybrid intelligence systems for dynamic data analysis. International Journal of Engineering and Information Systems (IJEIS), 6(2), pp. 40–48.

22. Гороховатський, В. О., Творошенко, І. С., & Чмутов, Ю. В. (2022). Застосування систем ортогональних функцій для формування простору ознак у методах класифікації зображень. Сучасні інформаційні системи, 6(3), с. 5-12.

23. StatsBomb. Introduction to xG models and football analytics. URL: https://statsbomb.com/education/xg (дата звернення 14.10.2025).

24. Гороховатський, В., & Творошенко, І. (2021). Методи інтелектуального аналізу та оброблення даних. URL: https://openarchive.nure.ua/server/api/core/bitstreams/2e55d639-52fd-48d9-b7b7-14989f49f291/content (дата звернення: 13.10. 2025).

25. Tvoroshenko, I., & Maksimenko, H. (2021). Research of regression and modular testing of web applications. Science, Theory and Practice: Abstracts of the IV International Scientific and Practical Conference, pp. 406–411.

26. FastAPI Documentation. Building high-performance ML APIs with FastAPI. URL: https://fastapi.tiangolo.com/ (дата звернення 08.10.2025).

27. Гороховатський, В. О., & Творошенко, І. С. (2022). Аналіз багатовимірних даних за описом у формі множини компонент: монографія. Харків: ХНУРЕ, 124 сторінки.

28. Кобилін, О., Вечірська, І., & Кравченко, О. (2024). Порівняння нейронних мереж типу RNN та LSTM. Information Technology: Computer Science, Software Engineering and Cyber Security, 3, с. 97–107.

29. Kobylin, I., & Nikolaichuk, A. (2024). Monitoring and diagnosing faults in online mode using time series data. Системи обробки інформації, 3(178), pp. 27–32.

30. Tvoroshenko, I. S., & Maksimenko, H. (2021). To the question of analysis of existing mechanisms of web application testing. Science, Theory and Practice: Abstracts of the IV International Scientific and Practical Conference, pp. 403–409.

31. Середа, І. А. (2025). Огляд сучасних методів прогнозування результатів спортивних подій. Збірник наукових праць з матеріалами VII міжнародної конференції (с. 240–245). Житомир, Україна.

32. Середа, І. А. (2025). Порівняльний аналіз Apache Spark та Hadoop для обробки спортивних даних. Збірник наукових праць з матеріалами X міжнародної конференції (с. 415–418). Львів, Україна.