DOI 10.31891/2307-5732-2020-289-5-120-125
УДК 004.9

М.П. КОМАР
Західноукраїнський національний університет

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ІНТЕЛЕКТУАЛЬНОЇ ОБРОБКИ ТА АНАЛІЗУ ВЕЛИКИХ ДАНИХ

В статті розроблено методологію створення інтелектуальних систем обробки та аналізу великих даних в умовах невизначеності, неповноти та нечіткості інформації, яка базується на використанні наступних принципів: горизонтальної масштабованості, відмовостійкості, локальності даних, стійкості до помилок у даних, адаптивності, еволюційності.

Запропоновані принципи покладені в основу для розроблення моделей і методів інтелектуальної обробки та аналізу великих даних: методу відновлення відсутніх даних, який створює додаткові значення даних на основі функціональних залежностей та правил асоціації та додає ці значення до наявних навчальних даних; методу прийняття рішень в процесі експлуатації системи на базі нечіткої логіки, який забезпечує автономну роботу процесу обробки та аналізу великих даних без будь-якого втручання користувача; методу навчання глибоких нейронних мереж на основі об'єднання генетичного алгоритму і нейронної мережі для знаходження оптимальних параметрів нейронної мережі; методу паралельного навчання глибоких нейронних мереж на основі розбиття навчальної вибірки на підвибірки і паралельного навчання кожної підвибірки на окремій копії моделі нейронної мережі.

Для інформаційної підтримки запропонованих моделей і методів розроблено інформаційну технологію інтелектуальної обробки та аналізу великих даних. Запропоновану технологію реалізовано в рамках інтелектуальної системи обробки та аналізу великих даних. Дослідження результатів застосування розробленої системи запропоновано провести на основі прогнозування результатів футбольних матчів.

Ключові слова: інформаційна технологія, інтелектуальна система, обробки та аналізу великих даних, глибока нейронна мережа, нечітка логіка, генетичний алгоритм, паралельне навчання, прогнозування.

M.P. KOMAR
West Ukrainian National University, Ternopil, Ukraine

INFORMATION TECHNOLOGY OF INTELLIGENT PROCESSING AND ANALYSIS BIG DATA

The methodology for creating intelligent systems for processing and analyzing big data in conditions of uncertainty, incompleteness and vagueness of information developed in the article, based on the use of the following principles: horizontal scalability, fault tolerance, data locality, resistance to data errors, adaptability, evolution.

The proposed principles are the basis for the development of models and methods of intelligent processing and analysis of big data: the method of recovering missing data, which creates additional data values based on functional dependencies and association rules and adds these values to existing training data; the method of decision-making in the operation of the system on the basis of fuzzy logic, which provides autonomous operation of the process of processing and analysis of large data without any user intervention; a method of learning deep neural networks based on the combination of genetic algorithm and neural network to find the optimal parameters of the neural network; method of parallel learning of deep neural networks based on the division of the training sample into sub-samples and parallel training of each sub-sample on a separate copy of the neural network model.

For information support of the offered models and methods the information technology of intellectual processing and the analysis of big data is developed. The proposed technology is implemented within the intelligent system of big data processing and analysis. Research on the results of the application of the developed system is proposed to be conducted on the basis of forecasting the results of football matches.

Keywords: information technology, intelligent system, big data processing and analysis, deep neural network, fuzzy logic, genetic algorithm, parallel learning, forecasting.

Вступ. Постановка проблеми

На сьогодні глибоке навчання (Deep Learning) і великі дані (Big Data) є одними з найбільш гарячими тенденціями в швидко зростаючому цифровому світі [1]. Існують різні визначення терміну Big Data [2–6], проте в даній роботі звертаємо увагу на експоненціальне зростання і широке поширення цифрових даних, які важко, або навіть неможливо обробляти і аналізувати за допомогою звичайних програмних засобів і технологій. Ці обмеження призвели до еволюції технологій навколо великих даних. Великі дані, що визначаються швидким зростанням обсягу, різноманітності та швидкості передачі даних, зазвичай мають справу з неструктурованими даними, які потребують великого пакетного аналізу або аналізу в режимі реального часу.

Для того, щоб ефективно обробляти великі обсяги даних при прийнятних часових затратах, необхідні особливі технології. Сьогодні методи машинного навчання [7-10], зокрема глибокого навчання [11–13] разом з досягненнями в області обчислювальної потужності, відіграють важливу роль у аналітиці великих даних. Глибокі нейронні мережі [14–17] мають велику ефективність нелінійного перетворення і представлення даних в порівнянні з традиційними нейронними мережами.

Перспективним напрямом є інтеграція великих потоків даних із моделями глибокого навчання та розробка на цій основі технологій обробки та аналізу великих даних. Метою дослідження є розробка моделей, методів та інформаційної технології інтелектуальної обробки та аналізу великих даних.

1. Методологія створення інтелектуальних систем обробки та аналізу великих даних

Методологія створення інтелектуальних систем обробки та аналізу великих даних в умовах невизначеності, неповноти та нечіткості інформації базується на використанні наступних принципів:

- горизонтальної масштабованості;

- відмовостійкості;
- локальності даних;
- стійкості до помилок у даних;
- адаптивності;
- еволюційності.

Запропоновані принципи покладені в основу для розроблення моделей та методів інтелектуальної обробки та аналізу великих даних:

1. Моделі і методу відновлення відсутніх даних, який створює додаткові значення даних на основі функціональних залежностей та правил асоціації та додає ці значення до наявних навчальних даних, що, в свою чергу, дало змогу підвищити ефективність подальшого аналізу даних.

2. Методу прийняття рішень в процесі експлуатації системи на базі нечіткої логіки, який забезпечує автономну роботу процесу обробки та аналізу великих даних без будь-якого втручання користувача, що дозволяє підвищити швидкодію та унеможливити виникнення помилок користувача при роботі з тим чи іншим набором великих даних та дає можливість обробляти та аналізувати потокові дані в режимі реального часу.

3. Методу навчання глибоких нейронних мереж на основі об'єднання генетичного алгоритму і нейронної мережі для знаходження оптимальних параметрів нейронної мережі, що дало можливість здійснювати аналіз даних вже на ранніх етапах роботи генетичного алгоритму, тобто паралельно з його роботою. Ця можливість обумовлена додаванням в базу даних мереж, що мають мінімальну середньоквадратичну помилку, на кожному кроці генетичного алгоритму.

4. Методу паралельного навчання глибоких нейронних мереж за рахунок розбиття навчальної вибірки на підвибірки і паралельного навчання кожної підвибірки на окремій копії моделі нейронної мережі, що дозволило збільшити швидкість навчання та зменшити використання пам’яті графічних процесорів.

2. Архітектура інтелектуальної системи обробки та аналізу великих даних

Запропоновану технологію реалізовано в рамках інтелектуальної системи обробки та аналізу великих даних. Компоненти архітектури системи BDDL представлено на рис. 1.

Користувацький інтерфейс. Це інтерфейс, який відображає інформацію з різних компонентів запропонованої системи, а також дозволяє користувачу виконувати різні дії, такі як вибір даних, попередня обробка даних та навчання глибоких нейронних мереж.

Модуль управління. Він діє як посередник між користувацьким інтерфейсом та іншими компонентами системи. Цей модуль інтерпретує дії користувача та викликає відповідний модуль. По-перше, він викликає модуль формування та вибору набору даних для обробки потоку великих даних, який вибраний користувачем у користувацькому інтерфейсі для подальшої обробки. Далі модуль управління перенаправляє дії користувача до одного з модулів попередньої обробки даних чи модуля відновлення відсутніх даних для виконання відповідних операцій над даними, які допомагають перетворити дані у необхідний формат для наступного кроку. Пізніше модуль управління надсилає попередньо оброблений набір даних модулю навчання глибоких нейронних мереж, а той в свою чергу, при необхідності, - модулю паралельного навчання.

Після того, як користувач закінчить навчання моделі, йому надається можливість записати всі виконані кроки в базу даних результатів. Вибрані файли, усі операції попередньої обробки даних, параметри навчання моделі та вибрана модель глибокого навчання записуються у файл JSON. В майбутньому той самий експеримент користувача з усім набором даних можна виконати автоматично, використовуючи ці записані кроки. Значення кожного кроку користувача витягуються з файлу JSON і виконуються у фоновому режимі.

Модуль формування та вибору набору даних. Цей модуль надає можливості вибору різних потоків великих даних із різних джерел даних. Він дозволяє користувачеві вибрати різні типи даних, такі як текст, зображення, відео, звук, числові ряди та ін. Він також обробляє великі потоки даних з різних джерел даних, таких як розподілена файлова система Hadoop (HDFS) для пакетних даних та Apache Storm та Apache Spark для потоків даних у реальному режимі часу. Основні функції модуля формування та вибору набору даних:

- дозволяє відбирати різні типи даних;
- дозволяє сформувати потоки даних з різних джерел даних;
- дозволяє користувачам вибрати файл або кілька файлів, щоб виконати наступні дії.

Потім вибрані файли даних використовуються як вхідні дані для одного з модулів попередньої обробки даних (текстових чи зображень).

Модуль попередньої обробки текстових даних. Він надає користувачеві бібліотеку операцій для виконання маніпуляцій та перетворень із вибраними текстовими даними. Використовуючи попередню обробку даних, користувач може перетворити необроблені дані в нормальний набір даних у певному форматі. Він включає такі задачі, як очищення даних, нормалізація, трансформація, зменшення розмірності тощо. Модуль попередньої обробки текстових даних підтримує наступні операції з даними:

1. Очищення даних. Основною метою цієї функції є видалення невідповідностей, надлишкових та нерелевантних даних. Це дозволяє видалити відсутні значення, зашумлені дані та виправити невідповідні дані.

2. Інтеграція даних. Ця операція дозволяє інтегрувати дані з декількох джерел (файлів даних) в єдиний набір даних шляхом об’єднання значень атрибутів.

3. Зменшення даних. Це дозволяє згенерувати стиснуту версію всього набору даних використовуючи метод головних компонент чи автоенкодер.

4. Перетворення даних. Це дозволяє перетворити дані у формат, який вимагають моделі глибокого навчання. Це передбачає нормалізацію - перетворення значення атрибута (числовий атрибут) у заданий діапазон, агрегацію - поєднання атрибутів в один атрибут та узагальнення - заміна атрибутів нижчого рівня або примітивних (необроблених) даних концепціями вищого рівня.

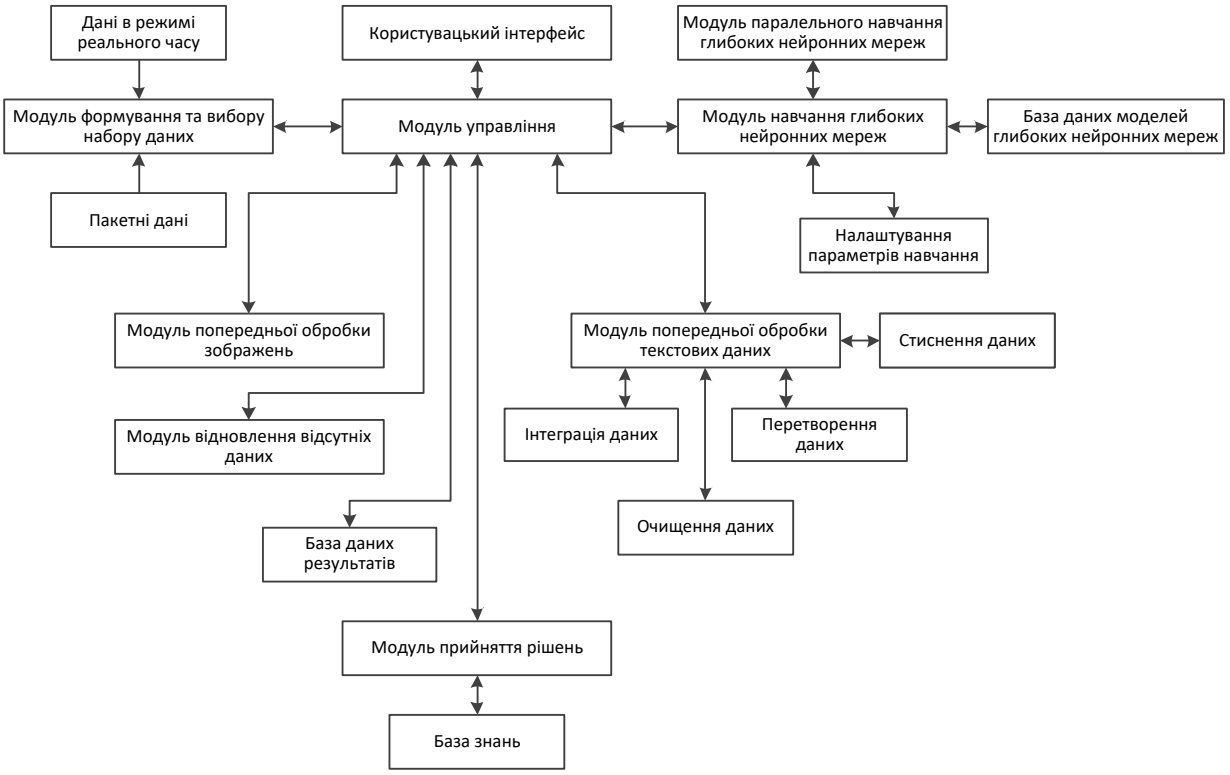

Рис. 1. Архітектура системи BDDL

Результатом попередньої обробки є набір даних, який можна використовувати як остаточний навчальний набір для моделей глибокого навчання.

Модуль попередньої обробки зображень. Після вибору файлів зображень для подальшої обробки система надає користувачеві різні операції, такі як обрізання зображення, обертання зображення, регулювання яскравості, насиченості та ін. Для виконання цих операцій на зображеннях доцільно використовувати фреймворк Tensorflow.

Модуль відновлення відсутніх даних. Його суть полягає в обробці структурованих та напівструктурованих даних на основі ієрархії об'єктів, а також набору функціональних залежностей та розробки правил асоціації. Це питання дуже важливе для інтерфейсів великих даних, оскільки більша частина інформації доступна в напівструктурованому вигляді. Запропонований підхід створює додаткові значення даних за допомогою доменних та функціональних залежностей на основі декількох методів обчислення та додає ці значення до наявних навчальних даних. Правильність обчислених значень перевіряється на класифікаторі, побудованому на вихідному наборі даних.

Модуль навчання глибоких нейронних мереж. Він надає користувачеві можливість вибрати з бази даних модель глибокого навчання для виконання різних задач, таких як класифікація, прогнозування та рекомендації та ін. Для навчання моделей цей модуль використовує попередньо оброблений набір даних з одного з модулів попередньої обробки. Він також надає користувачеві можливість вибору значень для різних параметрів для оптимізації процесу навчання.

Основними задачами модуля навчання глибоких нейронних мереж:

- дозволяє користувачам вибирати значення різних параметрів навчання, таких як кількість епох, розмір партії, функція активації, швидкість навчання, крок, кількість шарів, кількість нейронів у відповідному шарі;
- дозволяє користувачам вибирати різні моделі глибокого навчання для різних задач, таких як класифікація, прогнозування, рекомендації та ін.

Користувач може також змінювати значення параметрів навчання за замовчуванням. Для підвищення ефективності навчання глибоких нейронних мереж можна використати генетичні алгоритми. Генетичний алгоритм – адаптивний метод пошуку, який все частіше використовується для вирішення задач функціональної оптимізації. Він заснований на генетичних процесах біологічних організмів: біологічні популяції розвиваються впродовж декількох поколінь, підкоряючись законам природного відбору і принципу «виживає найбільш пристосований». Початкові покоління наборів параметрів (особин) для генетичного алгоритму визначаються випадковим чином. Далі, найбільш пристосованими особинами

вважаються набори, навчені на яких мережі дають мінімальні помилки. Нове покоління особин виходить шляхом схрещування найбільш пристосованих особин попереднього покоління і мутації.

Модуль паралельного навчання. За необхідності, можна використати переваги паралельного навчання глибоких нейронних мереж. Основною проблемою при глибокому навчанні є висока ресурсоємність навіть невеликих нейронних мереж. Реальний об'єм пам'яті, потрібний для мережі середнього розміру ResNet-50, що має 26 мільйонів вагових параметрів і обчислює близько 16 мільйонів операцій при прямому проході, складає майже 8 Гб оперативної пам'яті. Даний модуль дозволяє використовувати хмарні платформи 1Cloud, Azure, Google Cloud, де можна замовити віртуальні сервери потрібної конфігурації з підтримкою GPU та без неї на потрібний час.

Модуль прийняття рішень. Цей модуль дозволяє приймати рішення в процесі експлуатації системи на базі нечіткої логіки використовуючи правила нечіткого висновку Мамдані. Побудовані правила зберігаються в базі знань. Модуль прийняття рішень забезпечує автономну роботу процесу обробки та аналізу великих даних без будь-якого втручання користувача, що дозволяє підвищити швидкодію та унеможливити виникнення помилок користувача при роботі з тим чи іншим набором великих даних та дає можливість обробляти та аналізувати потокові дані в режимі реального часу, що є надзвичайно актуальним при побудові сучасних систем аналізу мережевого трафіку з метою виявлення вторгнень чи інших систем, наприклад, систем Інтернету Речей.

3. Дослідження результатів застосування розробленої системи

Дослідження результатів застосування розробленої системи запропоновано провести на основі прогнозування результатів футбольних матчів англійської Прем'єр-ліги. Для опису сильних сторін футбольних команд можна використовувати багато різних показників. Підбір показників, що формують рейтинг футбольної команди, є важливою задачею. Необхідно вибирати такі показники, які мають високий ступінь інформації та значення для опису колективу. Найбільш значущими показниками є місце в турнірній таблиці, кількість очок за вибраний інтервал часу, кількість забитих голів за вибраний інтервал часу, кількість пропущених голів тощо.

Розглянемо показники, які було обрано для формування навчальної вибірки:

- S – кількість ударів команди;
- ST – кількість ударів команди в площину воріт;
- C – кількість кутових ударів команди;
- F – кількість фолів команди;
- CS – кількість жовтих та червоних карточок команди;
- GS – кількість забитих командою голів;
- HTGS – кількість забитих командою голів в першому таймі;
- GC – кількість пропущених командою голів;
- HTGC – кількість пропущених командою голів в першому таймі.

Наведені показники можна завантажити з Football-Data.co.uk [18] і включати інформацію про всі ігри Прем'єр-ліги з 2002 року. Для формування вхідних даних не використовується інформація лише про останній матч, оскільки її недостатньо для побудови повної інформації про умови команд. Замість цього використовуємо останні 35 збігів та обчислюємо агреговані статистичні показники. Такі показники є більш інформативними і можуть бути використані для оцінки поточного стану команди. Агрегація вхідного шаблону виконується шляхом усереднення кожного параметра для 35, 15, 10, 5 збігів. І так по кожному показнику.

Крім того, розраховуємо стандартне відхилення для 35 збігів і додаємо його до шаблону даних. Стандартне відхилення дає інформацію про стабільність команди і дуже важливе для прогнозування результату.

Після агрегування отримуємо вхідний вектор, що складається із 108 значень - 54 значення для обох команд. Крім того, вхідні дані нормуються за такими статистичними показниками: середнє значення та середнє відхилення. Це дозволяє отримати більш стабільну модель прогнозування.

Для прогнозування спортивних результатів була використана еластична нейронна мережа. Вибір цієї структури нейронної мережі обумовлений тим, що вона здатна робити хороші прогнози, використовуючи не повні дані. Архітектура нейронної мережі для прогнозування результатів матчів еластичну мережу з архітектурою, показаною на рис. 2.

Вхідний шар містить 108 нейронів. Розмірність вхідного шаблону визначає кількість вхідних нейронів. Мережа має три приховані шари зі 128, 64 та 32 нейронами у кожному шарі відповідно та 3 нейронами у вихідному шарі. Така мережа, яка містить кілька прихованих шарів, називається Deep Elastic Net.

Підготовлений шаблон (див. рисунок 3.5) надходить на вхід мережі. Далі три приховані шари з функціями активації LeakyReLU виконують обчислення на вхідному шаблоні. Три вихідні нейрони відображають результати обчислень, інтерпретовані в наступній формі: виграш - нічия - програш. Перший нейрон виходу відповідає за перемогу господарів, другий нейрон - за нічию в матчі, а третій нейрон - за перемогу команди гостей. Використовується функція активації softmax для нейронів вихідного шару.

Відмінною особливістю Elastic Net є те, що вона використовує регуляризацію L1, L2. У той час як регуляризація L1 (також відома як Регресія Лассо) використовується для вибору параметрів, регуляризація L2 (також відома як Ridge Regression) здійснює контроль над переналагодженням мережі (переналагодження означає зростання модельних коефіцієнтів) у процесі навчання.

Для тестування системи, розробленої для прогнозування результатів футбольних матчів англійської Прем'єр-ліги, було використано набір даних, що складається з 5018 шаблонів (тобто набір містить інформацію про всі ігри з 2002 року). Для дострокового завершення навчального процесу було використано перевірочний набір розміром 15% від навчального набору.

Для вивчення Deep Elastic Net були обрані такі параметри:

- коефіцієнт для регуляризації L1 = 0,002;
- коефіцієнт для регуляризації L2 = 0,0005;
- алгоритм навчання - SGD (стохастичний градієнтний спуск) з кроком, рівним 0,01;
- розмір мініпартії дорівнює 64.

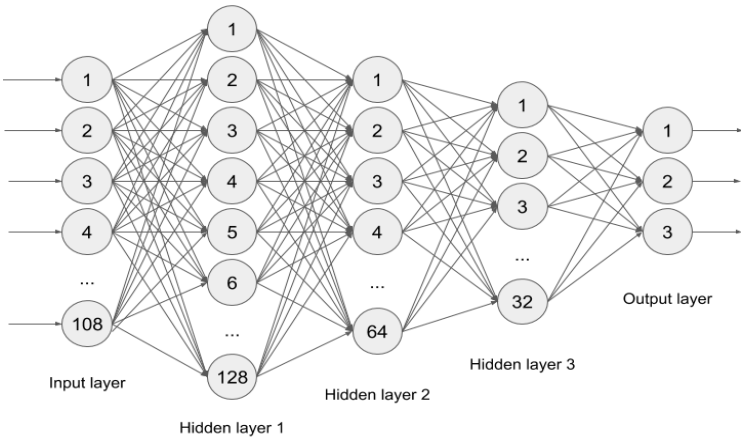

Рис. 2. Структура розробленої мережі Deep Elastic для прогнозування результатів футбольних матчів

Процес навчальння нейронної мережі займає приблизно 5 хвилин на наступній конфігурації ПК: графічний процесор NVidia 1070TI, процесор Xeon e5-2680 v2, оперативна пам’ять 32 ГБ.

Навчена система була протестована на останніх 350 матчах Прем’єр-ліги, які не були включені до навчальних та перевірочних наборів.

Результати прогнозування представлено в табл. 1.

Таблиця 1

Результати прогнозування

Команди та остаточний рахунок матчу	Перемога команди господарів, %	Нічия в матчі, %	Перемога команди гостей, %
West Ham vs Southampton (score 3:0)	41,6	24,7	33,7
Huddersfield vs Man United (score 1:1)	24,3	15,4	60,2
Leicester vs Arsenal (score 3:0)	30,3	43,5	26,2
Burnley vs Man City (score 0:1)	14,2	19,1	66,7
Fulham vs Cardiff (score 1:0)	48,3	27,5	24,3
Wolves vs Fulham (score 1:0)	72,7	17,4	9,9

Система показала точність прогнозування на наборі даних 61,14%. Використовуючи результати прогнозування розробленої системи, користувачі можуть зробити ставку на ту чи іншу команду в майбутньому матчі.

Висновки

На основі інтеграції великих потоків даних із моделями глибокого навчання розроблено інформаційну технологію інтелектуальної обробки та аналізу великих даних, яка передбачає вибір даних з різних архітектур великих даних, попередню обробку та відновлення відсутніх даних, навчання глибоких нейронних мереж на основі генетичного алгоритму для знаходження оптимальних параметрів нейронної мережі, паралельне навчання моделей глибоких нейронних мереж та прийняття рішень на основі нечіткої логіки, що дозволило підвищити ефективність обробки та аналізу великих даних і забезпечити автономну роботу процесу в порівнянні з відомими інформаційними технологіями.

Запропоновану технологію реалізовано в рамках інтелектуальної системи обробки та аналізу великих даних. Дослідження результатів застосування розробленої системи проведено на основі прогнозування результатів футбольних матчів. Пропонований підхід базується на глибокій еластичній мережі і може навчатися на наборі даних з обмеженим відкритим доступом. Експериментальні дослідження показали точність прогнозування на наборі даних 61,14%. Використовуючи результати прогнозування розробленої системи, користувачі можуть зробити ставку на ту чи іншу команду в майбутньому матчі.

Систему можна вдосконалити за допомогою використання більш детального та повного набору даних, який може бути забезпечений платними ресурсами та розробки більш складної структури глибокої нейронної мережі.

References

1. Chen X.-W. Big Data Deep Learning / X.-W. Chen, X. Lin // IEEE Access. – 2014. – Vol. 2. – P. 514-525.
2. Lynch C. Big data: science in the petabyte era / C. Lynch // Nature. – 2008. – Vol. 455. – P. 1-50.
3. Jean-Pierre D. Big Data for the Enterprise / D. Jean-Pierre // Oracle. http://Big Datawithoracle-521307.pdf [Access 18.08.2020].
4. Великі дані (Big Data). https://rb.ru/howto/chto-takoe-big-data [Access 18.08.2020].
5. The National Security Agency: Missions, Authorities, Oversight and Partnerships. http://www.nsa.gov [Access 18.08.2020].
6. Big Data: The Next Frontier for Innovation, Competition, and Productivity. McKinsey Global Institute. http://www.mckinsey.com/insights/business_technology/big_data_the_next_frontier_for_innovation [Access 18.08.2020].

7. Lin J. Large-scale machine learning at twitter / J. Lin, A. Kolcz // Proc. ACM SIGMOD Scottsdale Arizona USA. – 2012. – P. 793-804.

8. Smola A. An architecture for parallel topic models / A. Smola, S. Narayanamurthy // Proc. VLDB Endowment. – 2010. – Vol. 3, No. 1, pp. 703–710.

9. Ng A. et al. Map-reduce for machine learning on multicore // Proc. Adv. Neural Inf. Procees. Syst. – 2006. – Vol. 19, pp. 281–288.

10. Panda B. MapReduce and its application to massively parallel learning of decision tree ensembles / B. Panda, J. Herbach, S. Basu, and R. Bayardo // In Scaling Up Machine Learning: Parallel and Distributed Approaches. Cambridge, U.K.: Cambridge Univ. Press, 2012.

11. Crego E. Big data and deep learning: Big deals or big delusions / E. Crego, G. Munoz, and F. Islam.? Business. http://www.huf_ngtonpost.com/george-munoz-frank-islamand-ed-crego/big-data-and-deep-learnin_b_3325352.html [Access 19.08.2020].

12. Bengio Y. Modeling high-dimensional discrete data with multi-layer neural networks / Y. Bengio, S. Bengio // In Proc. Adv. Neural Inf. Process. Syst. – 2000. - Vol. 12., pp. 400–406.

13. Marc'Aurelio Ranzato Y. Sparse feature learning for deep belief networks / Y. Marc'Aurelio Ranzato, L. Boureau, Y. LeCun // In Proc. Adv. Neural Inf. Process. Syst. – 2007. – Vol. 20., pp. 1185–1192.

14. Hinton G.E. A fast learning algorithm for deep belief nets / G. E. Hinton, E.S. Osindero, Y. Teh // Neural Computation. – 2006. – Vo1. 18. – pp. 1527–1554.

15. Hinton G. Reducing the dimensionality of data with neural networks / G. Hinton, R. Salakhutdinov // Science. – 2006. – Vo1. 313 (5786). – pp. 504–507.

16. Hinton G.E. A practical guide to training restricted Boltzmann machines / G.E. Hinton // Machine Learning Group, University of Toronto. – 2010 (Tech. Rep. 2010-000).

17. LeCun Y. Deep learning / Y. LeCun, Y. Bengio, G. Hinton // Nature. – 2015. – Vol. 521 (7553). – pp. 436–444.

18. Football-Data. http://www.football-data.co.uk [Access 20.08.2020].

Надійшла / Paper received : 14.11.2020 Надрукована/Printed :27.11.2020