УДК 004.032.26

С.О. ДОВГИЙ
Інститут телекомунікацій і глобального інформаційного простору НАН України, Київ, Україна, e-mail: *pryjmalnya@gmail.com*.

М.О. ЗОЗЮК
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна, e-mail: *maksym.zoziuk@gmail.com*.

Д.В. КОРОЛЮК
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»; Інститут математики НАН України; Інститут телекомунікацій та глобального інформаційного простору НАН України, Київ, Україна; лабораторія цифрових інновацій Міждисциплінарної кафедри ЮНЕСКО з біотехнології та біоетики Римського університету Тор Вергата, Італія, e-mail: *dimitri.koroliouk@ukr.net*.

АДАПТИВНА СИСТЕМА ГЛИБОКОГО НАВЧАННЯ ДЛЯ ДОСЛІДЖЕННЯ ДАНИХ ЗАГАЛЬНОГО ТИПУ

Анотація. Представлено підхід до аналізу великої кількості даних, між якими немає чіткого зв'язку. Він слугує для того, щоб встановити наявність хоча б якогось зв'язку, і у разі встановлення його наявності визначити також ступінь цього зв'язку. Продемонстровано можливість автоматичного оброблення даних та автоматичної зміни параметрів систем глибокого навчання для формування платформи глибокого навчання під час аналізу параметрів даних. Показано, як ця система задовольнятиме потреби фахівців в інших галузях, які ніколи безпосередньо не використовували системи глибокого навчання. За допомогою відомого набору даних MNIST встановлено, що з використанням окремих параметрів цих даних можна визначити їхній вплив на точність передбачення системи глибокого навчання.

Ключові слова: нейронна мережа, автоматичне оброблення даних, інформативність вибірки, система прогнозування, методи оброблення.

ВСТУП

Нині у світі є велика кількість даних, які мають нелінійну залежність між різними параметрами. Наприклад, медико-біологічні [1–3] або астрономічні [4–6] дані, для яких залежності між параметрами одного об'єкта є неясними, і які можна встановити тільки за допомогою нелінійних методів апроксимації.

Іншою проблемою є потреба у розумінні взаємозалежностей даних та їхньої сутності. До того ж, потрібно мати навички оброблення цих даних та роботи з системами глибокого навчання. Зазвичай, організовують групи, до складу яких входять фахівці у різних галузях. Вони спільно ставлять задачу та виконують процес дослідження з використанням систем глибокого навчання.

У зв'язку з потребою у простих рішеннях для осіб, які не є фахівцями у галузі штучного інтелекту, створено доволі потужні системи типу ChatGPT [7] і Microsoft Bing [8]. Всі ці системи, як правило, призначені для швидкого доступу до інформації без використання звичайних пошукових систем. На жаль, вони не завжди дають змогу використати весь потенціал глибокого навчання для розв'язання проблем, особливо у тому разі, коли потрібно скористатися можливостями глибокого навчання для аналізу великої кількості даних.

Ще однією проблемою є труднощі у розумінні характеру впливу одних параметрів на інші у разі однорідних досліджуваних об'єктів, або специфічних залежностей, розглянутих дослідниками цих параметрів [9–11]. Є різні методи вивчення впливів одних параметрів на інші [12–15].

У розд. 1 описано автоматичну систему глибокого навчання. У розд. 2 представлено можливості застосування цієї системи у наукових цілях, а саме для вста-

новлення наявності взаємозв'язку між параметрами та визначення якості передбачення. У розд. 3 представлено дослідження класичного набору даних MNIST та розглянуто правильність підготовки даних для автоматичної системи.

1. АВТОМАТИЧНА СИСТЕМА ГЛИБОКОГО НАВЧАННЯ

Не всі науковці вміють застосовувати системи глибокого навчання та обробляти дані відповідно до встановлених практик. Тому виникає питання щодо можливості створення такої системи, яка б використовувала типи даних, кількість параметрів, кількість унікальних значень для кожного параметра, найменше/найбільше значення тощо. На основі інформації про ці властивості бази даних можна автоматично обробити та підготувати вхідні дані за допомогою системи автоматичного оброблення даних у потрібному для нейронної мережі форматі.

Зауважимо, що всі дані для глибокого навчання нейронних мереж є специфічними до задач, які для цих даних формулюються. Велика кількість параметрів впливає на якість систем глибокого навчання, і до кожної проблеми є свій підхід. З іншого боку, якщо розуміти (наприклад, для нейромереж), що майже всі дані у кінцевому підсумку є просто числами від 0 до 1 (практично всі системи глибокого навчання використовують або повнозв'язний або частково зв'язаний нейрон), то можна поставити задачу зі створення автоматичної системи цих перетворень. Інакше кажучи, всі дані проходять через різні процедури, але кінець кінцем вони є раціональними числами від 0 до 1. Зазвичай краще за все системи глибокого навчання працюють тоді, коли використовують діапазон вхідних і вихідних значень від 0 до 1. Якщо для різних видів даних потрібно виконувати різні процедури, то на вході такої системи слід просто одразу встановити перевірку типу даних і автоматичний вибір процедури. Якщо тип даних однаковий, але є підтипи, здійснюють ту саму перевірку на підтип даних і відповідну процедуру. Нижче наведено способи автоматичного вибору процедур для різних типів даних та автоматичного підбору параметрів систем глибокого навчання залежно від характеристик даних.

Автоматична система оброблення даних. Метою всіх процедур є присвоєння кожній унікальній величині або числу нового унікального значення, яке складатиметься з чисел у діапазоні від 0 до 1. Процедури кодування інформації бувають різними. Все залежить від способу представлення цієї інформації. Якщо вона має вигляд графічних зображень, то результатом кодування зазвичай є набори чисел (тензори) від 0 до 1 для кожного індексу в тензорі. Якщо це послідовність чисел (звукові файли), то результатом кодування є вектори чисел для кожного зразка. У найпростішому випадку всі дані можна представити у вигляді таблиці, де кожен стовпець — це параметр, а кожен рядок — це елемент вибірки для систем глибокого навчання. У кожному стовпці параметр можна представити або в числовому вигляді, або у символьному. Для використання цих даних у системах глибокого навчання всі дані у стовпцях потрібно привести до значень у діапазоні від 0 до 1. У разі представлення у числовому вигляді можна використовувати такі види масштабування: абсолютне максимальне масштабування, мінімально-максимальне масштабування, нормалізація, стандартизація, надійне масштабування. Якщо потрібно, щоб усі значення були в діапазоні від 0 до 1, то найчастіше використовують мінімально-максимальне масштабування та нормалізацію.

Для мінімально-максимального масштабування формула нового значення має такий вигляд:

$$new.value = \frac{current.value - min.value.parameter}{max.value.parameter - min.value.parameter}. \qquad (1)$$

У разі застосування нормалізації замість мінімального значення використовують середнє значення параметра:

$$new.value = \frac{current.value - mean.value.\ parameter}{max.value.\ parameter - min.value.\ parameter}. \qquad (2)$$

Інакше кажучи, є тільки дві процедури, за допомогою яких можна кодувати числові дані у дані діапазону від 0 до 1.

Іншим типом даних є символьні дані. Тут вирішальну роль відіграє кількість унікальних значень для кожного параметра. Основні процедури кодування для символьних даних у числові є такими: кодування за мітками або порядкове кодування, OneHot кодування, кодування ефектів, геш-кодування, бінарне кодування, кодування Base-N та цільове кодування. Всі вони є зручними для різних ситуацій. У розглянутій в цій статті ситуації потрібні тільки процедури, на виході яких отримують числові дані в діапазоні від 0 до 1, як-от: OneHot кодування, геш-кодування та бінарне кодування.

У процедурі OneHot кодування кожна унікальна змінна стає вектором, який буде заповнений нулями і матиме кількість елементів, що дорівнює кількості унікальних значень для параметра. При цьому лише в окремій позиції буде одиниця (рис. 1).

У разі геш-кодування здійснюється схожа процедура, проте довжина кожного нового вектора не буде дорівнювати кількості унікальних значень. Вона буде меншою, і кожне унікальне значення буде мати свій «унікальний» вектор за алгоритмом гешування. Великим недоліком такої процедури є те, що не можна відновити початкові дані на основі геш-кодування.

Ще однією процедурою є бінарне кодування. Бінарне кодування схоже на OneHot кодування, але відрізняється тим, що в кожній позиції вектора може бути одиниця (рис. 2).

Колір		**Red**	**Green**	**Blue**
Red		1	0	0
Green	→	0	1	0
Blue		0	0	1
Green		0	1	0

Рис. 1. Приклад OneHot кодування

Стан	**Значення**
Idle	000
r1	001
r2	010
r3	011
r4	100
c	101
p1	110
p2	111

Рис. 2. Приклад бінарного кодування

Як випливає з порівняння видів процедур, найбільш універсальним є OneHot кодування. Будь-яку кількість унікальних значень деякого параметра можна конвертувати у значення в діапазоні від 0 до 1.

Слід зазначити, що числові значення для параметра також можна конвертувати за допомогою не числового, а категорійного кодування Ця процедура може бути набагато ефективнішою під час роботи з системами глибокого навчання, оскільки кількість вхідних значень не завжди, але в деяких випадках визначає час ефективного навчання систем глибокого навчання. Інакше кажучи, якщо кількість унікальних значень для числового параметра не перевищує ~10–20 значень, то є сенс використовувати саме категорійне кодування (наприклад, OneHot кодування). Зазвичай ці цифри приблизні, але зі збільшенням кількості унікальних значень числового параметра вже є сенс використовувати

числове кодування, оскільки час навчання систем глибокого навчання буде збільшуватися нелінійно, а якість навчання буде або незмінною або мало змінюватися порівняно з витраченим часом.

Автоматична система підбору параметрів систем глибокого навчання залежно від типу даних. Вид системи навчання залежить від типу даних. Це може бути повнозв'язна, рекурентна, згорткова нейромережа, машина Больцмана та інші гібридні системи. Якщо розглядають графічні зображення з великою роздільною здатністю, то використовують згорткові нейромережі. Для часових рядів застосовують рекурентні системи. Тут розглянуто найпростіший випадок, коли дані представлено у вигляді таблиць параметрів.

Основними функціями активації для таких даних є ReLU, сигмоїдна або логістична функція активації та softmax. Усі вони призначені для різних нейронів: функцію ReLU використовують для внутрішніх та початкових нейронів, а сигмоїдну та softmax — для вихідних нейронів. Сигмоїдна активація підходить, як правило, тоді, коли є тільки один вихідний нейрон, тобто одне значення від 0 до 1. Функцію softmax застосовують тоді, коли є велика кількість вихідних нейронів і правильним значенням вибирають найбільше значення з-поміж усіх. Тому в цьому випадку все визначається тільки кількістю унікальних значень на виході. Якщо параметр, який прогнозують, є числовим, то, найімовірніше, потрібно встановити сигмоїдну активацію. Якщо параметр є символьним, то слід застосувати функцію softmax.

Зауважимо, що найкращим універсальним оптимізатором є Adam. Під час створення розширеної версії автоматичної системи підбору параметрів глибокого навчання можна надати можливість користувачу змінювати ці параметри вручну та експериментувати.

Є декілька варіантів вибору функції втрат. Можна скористатися MSE (функцією середньоквадратичної похибки), бінарною крос-ентропією та категорійною крос-ентропією. Оскільки потрібно, щоб усі дані набували значень від 0 до 1, то підходять лише бінарна та категорійна крос-ентропія. Бінарну крос-ентропію вибирають, коли на виході є тільки одне значення, а категорійну — коли більше одного (за аналогією із softmax та сигмоїдною функцією).

Теорема Цибенка. Універсальна теорема апроксимації. З теореми Цибенка [16] випливає, що одного внутрішнього шару нейромережі достатньо для апроксимації залежностей будь-якої складності. Тому можна вибрати невелику кількість шарів, які будуть прийнятними для різних типів параметрів (наприклад, 2–3 шари залежно від кількості наявних параметрів — стовпців).

Кількість нейронів у кожному шарі потрібно встановлювати залежно від кількості вхідних та вихідних нейронів. Стандартний підхід передбачає встановлення кількості нейронів у внутрішніх шарах у межах кількості початкових та вихідних нейронів. Найкращим буде підхід від більшого до меншого. Наприклад, якщо кількість вхідних нейронів становитиме 100, а вихідних — 5, то кількість нейронів у внутрішніх шарах становитиме 75 і 25, відповідно. Якщо кількість вихідних даних становить 50, а кількість вхідних досі становить 100, то розміри внутрішніх шарів становитимуть 85 і 65. Якщо кількість вхідних даних становить 5, а вихідних — 100, то кількість нейронів у внутрішніх шарах становитиме 25 і 75 тощо. У такий спосіб легко автоматизувати цей процес, і якість навчання нейронної мережі не буде сильно залежати від відхилень у той чи інший бік від ідеально можливої конфігурації параметрів нейромережі (за якої точність прогнозування буде максимально можливою). Цю конфігурацію поки встановлюють експериментально. Кількість епох безпосередньо залежить від розміру партії навчання (batch size) у разі використання оптимізатора Adam.

Розмір партії прямо залежить від розміру вибірки даних, тобто кількості елементів (рядків) у таблиці даних. Якщо розмір партії великий, то може знадобитися дуже велика кількість епох, і навпаки, якщо розмір партії малий, то протягом невеликої кількості епох можна здійснити ефективне навчання нейронної мережі. Отже, можна корелювати ці два параметри (розмір партії і кількість епох) з величиною вибірки. Дуже малий розмір партії завжди буде зумовлювати обмеження в часі, яке є відносно неефективним, бо дає той самий результат для більшого розміру. Простим підходом може бути автоматичне встановлення розміру партії на рівні 1–5 % відсотків від розміру даних навчання. Кількість епох встановлюють у 2–5 разів більшою за розмір партії. Також, щоб уникнути перенавчання та витрат часу, варто встановити процедуру зупинення процесу навчання залежно від якості навчання системи.

2. ПОБУДОВА АВТОМАТИЧНОЇ СИСТЕМИ ДЛЯ ДОСЛІДЖЕННЯ ДАНИХ

За допомогою зазначеного підходу можна автоматично встановити вихідні та вхідні дані нейромережі, не переймаючись питанням оброблення цих даних. Так, прекрасним шляхом є завантаження даних з одночасним вибором вихідних і вхідних даних користувачем. Після цього здійснюються всі автоматизовані процеси з оброблення даних та навчання нейромережі з параметрами, встановленими автоматично залежно від даних, вибраних як вхідні або вихідні.

Використання вихідних параметрів нейромережі для аналізу даних. Вихідні параметри нейромережі — це втрати (loss) та точність прогнозування (accuracy). Втрати зазвичай не дають нічого суттєвого для розуміння загальної якості прогнозування і використовуються для конкретних задач. Для оцінювання роботи нейромережі застосовують точність прогнозування.

Щоб скористатися цією величиною для аналізу даних, потрібно розуміти, як вона пов'язана з характеристикою даних. Припустимо, наприклад, що на виході є параметр D, а на вході — набір параметрів A, B, C. Параметр D може бути числовим або символьним. У першому випадку є два варіанти здійснення трансформації даних — за допомогою масштабування MinMax та кодування OneHot. Якщо застосовують MinMax, то на виході буде лише одне значення. Система визначення точності працює тільки з суто точними значеннями. Це означає, що якщо на виході будуть приблизні значення, то це буде неправильним значенням, що дасть у загальну точність прогнозування негативний внесок. Однак, якщо перевірити точність за кожним елементом, то вона може бути дуже високою. Інакше кажучи, потрібно сформувати алгоритм (систему), яка визначає точність прогнозування нейронної мережі на основі округлених її вихідних значень та значень, з якими потрібно порівняти ці округлені значення. Отже, потрібно округлювати всі числові значення до якогось знаку, перш ніж встановлювати їх як вихідні значення нейромережі. Якщо трансформацію числових значень здійснюють за допомогою OneHot, то кожне значення вважають унікальним. Як вже зазначено, OneHot для числових даних краще використовувати тоді, коли кількість унікальних значень не перевищує деякого значення. Тому, тут виникає та сама ситуація, що й у випадку, коли параметр D розглядають, як символьний.

У другому випадку (коли параметр D є символьним) універсальною трансформацією є кодування OneHot. На виході нейромережі буде вектор, який має довжину, що дорівнює кількості унікальних значень для параметра D. Серед усіх значень, які входять у вектор, потрібно вибрати те значення, яке є найбільшим. Наприклад, вектор [0.12, 0.5, 0.42, 0.7] буде перетворений у [0, 0, 0, 1] і у такий спосіб буде здійснюватися порівняння з правильним значенням.

Важливо розуміти, як правильно аналізувати точність прогнозування. Аналіз точності прогнозування прямо залежить від кількості унікальних значень для параметра. Наприклад, якщо унікальні значення параметра D — це [0, 1], і нейромережа навчена прогнозувати цей параметр з точністю 50 %, то це означає, що немає жодних залежностей всередині даних або нейромережа погано спроєктована. Інакше кажучи, мінімально корисну точність можна розрахувати за простим співвідношенням:

$$accuracy = \frac{1}{number\ of\ \ unique\ values} \cdot 100\ \%. \qquad (3)$$

Отже, можна стверджувати, що чим більша різниця між мінімальною точністю та реальною, тим більша залежність між параметром D та параметрами A, B, C.

За допомогою автоматичного встановлення вхідних та вихідних параметрів можна визначати ступінь залежності вихідних параметрів від вхідних. Так, наприклад, завдяки побудові нейромережі, яка б давала точність прогнозування параметра D залежно від параметрів A, B, C по черзі, можна встановити наявність залежності та її ступінь. До того ж, використовуючи статистичні дані в самих параметрах (належність різних унікальних значень (їхньої кількості) до того чи іншого виходу вихідного параметра D) можна встановити самі залежності.

Використання вихідних параметрів для визначення якості прогнозування. На основі точності прогнозування можна використовувати саму нейромережу для прогнозування на нових даних. Шляхом оцінювання різниці між мінімальною корисною точністю та реальною точністю можна формувати або нові дані, або нові моделі для прогнозування. Автоматизувати цей процес нескладно, бо весь процес оброблення даних та формування параметрів нейромережі виконуються автоматично. Отже, фахівці різних галузей, які не мають змоги будувати власні системи штучного інтелекту, отримають можливість скористатися автоматичною системою, завантаживши в неї власні, правильно представлені дані.

Алгоритм автоматичної системи оброблення даних та підбору параметрів нейромережі має вигляд, наведений на рис. 3.

Звичайно, мають бути можливості ручного встановлення всіх процесів оброблення даних (вибір трансформації), всіх параметрів нейромережі (гіперпараметри, тип нейромережі тощо). Однак, для цього потрібно мати хоча б початкове розуміння всіх процесів або час на експериментальне дослідження даних для різних варіантів.

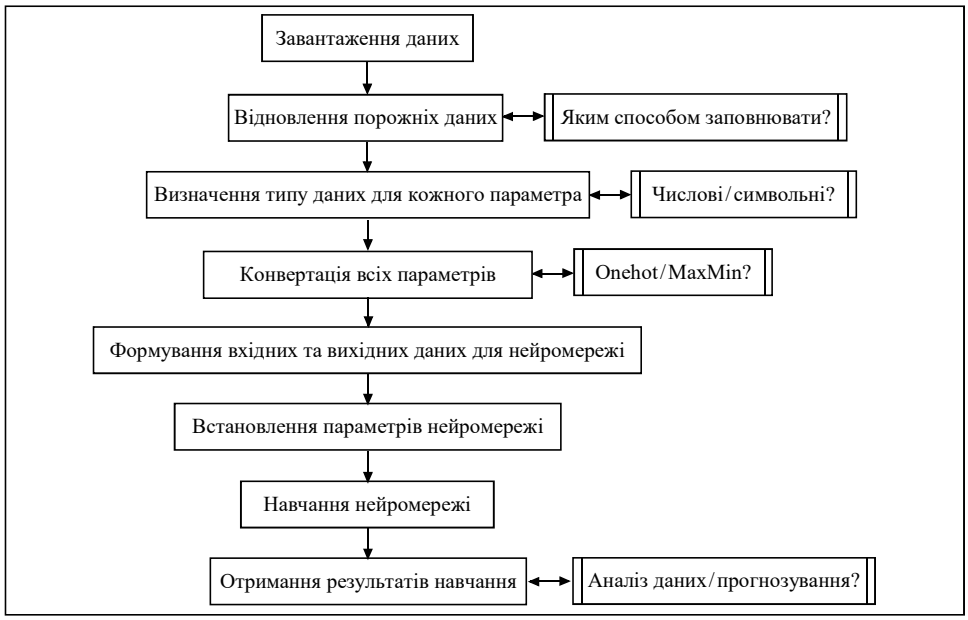


Рис. 3. Алгоритм автоматичної системи глибокого навчання для табличних даних

3. ДОСЛІДЖЕННЯ НАБОРУ ДАНИХ MNIST ТА АНАЛІЗ ПРАВИЛЬНОСТІ ПІДГОТОВКИ ДАНИХ ДЛЯ АВТОМАТИЧНОЇ СИСТЕМИ

Для дослідження інформативності наборів даних розглянемо автоматичну систему підготовки даних та параметрів нейромережі на прикладі набору даних MNIST [17]. Кожен елемент у цьому наборі є вектором довжиною 785 елементів. Його перший елемент — це число від 0 до 9, яке визначає, що цей вектор представляє певну рукописну цифру. Всі інші елементи — це числа від 0 до 255. По суті вони визначають інтенсивність пікселя у зображенні цифри. Якщо конвертувати цей вектор у матрицю розміром 28×28 і залежно від чисел сформувати зображення цієї матриці, то це буде рукописне число від 0 до 9. Щоб розв'язати задачу прогнозування числа для кожного елемента MNIST, потрібно завантажити файл в автоматичну систему та вказати, який саме параметр прогнозувати (нас цікавить число від 0 до 10) та які брати за основу (всі інші числа в діапазоні від 0 до 255). Інакше кажучи, слід вибрати перший елемент з 785 як вхід, а всі інші — як вихід. Як зазначено у розд. 2, метод кодування інформації визначають залежно від кількості унікальних значень для кожного параметра. Тут для чисел від 0 до 9 застосовано OneHot, бо кількість унікальних значень невелика. Для всіх інших параметрів (позицій у векторі) використано MinMax.

Для того, щоб повністю розкрити можливості автоматичної системи, можна після завантаження файлу встановити тільки частину (з 784) пікселів для прогнозування числа від 0 до 9. У цій роботі представлено один із прикладів для трьох випадків прогнозування числа від 0 до 9 залежно тільки від частини пікселів, а саме, для пікселів від 0 до 261, від 261 до 522 та від 522 до 783, як показано на рис. 4.

У представленій системі всі процеси встановлення гіперпараметрів та параметри оброблення даних для нейронної мережі виконано в автоматичному режимі. Кількість шарів, кількість нейронів у кожному шарі, величину партії (batch size), кількість епох, функції активації на всіх шарах, оптимізатор, метод розрахунку похибки встановлено на основі того, які дані подано на вхід та вихід нейронної мережі, а також способу, у який виконано перетворення даних у потрібний формат.

Результати експерименту представлено у вигляді точності прогнозування для випадків використання часткових даних про елементи MNIST, як показано на рис. 5.

По осі ординат точність становить від 0 до 1. Якщо врахувати формулу (3), то мінімальна точність становить 0.1 або 10 %. Інтерес становлять криві тестових даних. У випадку, коли враховано пікселі від 0 до 261 (рис. 5, *а*), маємо криву, яка асимптотично прямує до 0.77 (77 %). Для проміжку пікселів від 261 до 522 (рис. 5, *б*) маємо криву, яка асимптотично прямує до 0.92 (92 %). Для третього випадку (проміжок від 522 до 783 пікселів) (рис. 5, *в*) маємо криву, яка асимптотично прямує до 0.77 (77 %).

Очевидно, що середня частина пікселів несе набагато більше інформації про елемент, ніж верхня чи нижня. Зрозуміло також, що у разі збільшення кількості епох для верхньої і нижньої частини елемента точність буде збільшуватись, але час, витрачений на навчання, зростатиме нелінійно.

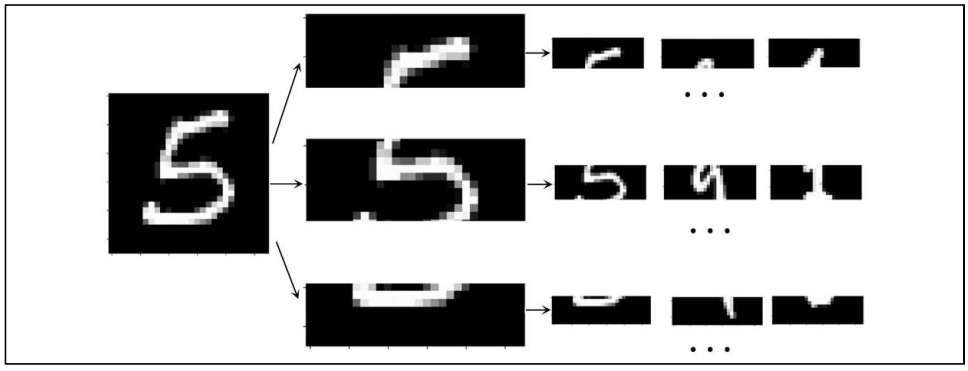
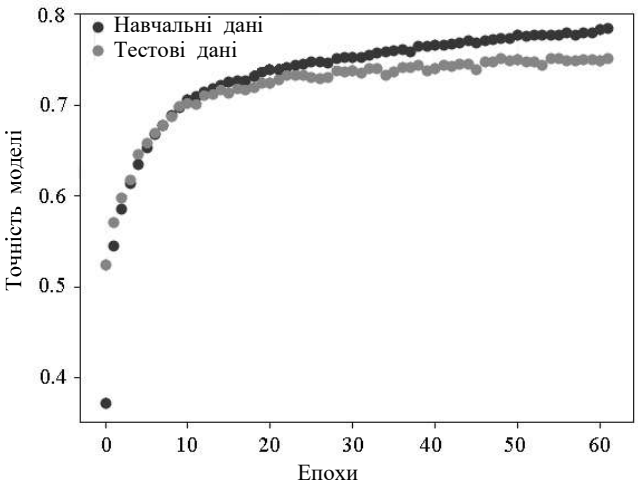
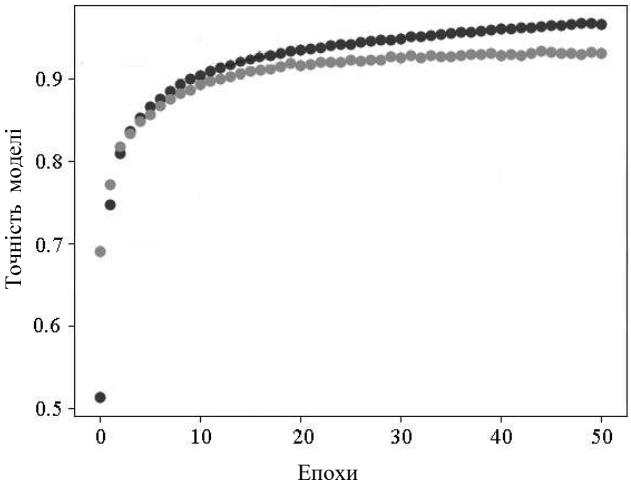


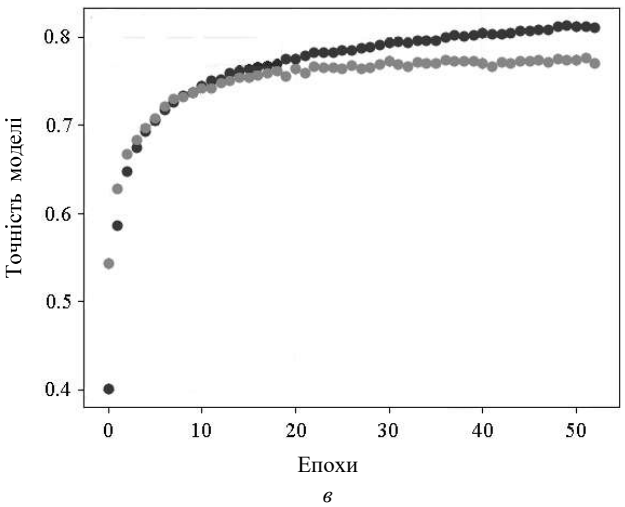
Рис. 4. Розділення кожного елемента на три частини



а



б



в

Рис. 5. Графіки залежності точності прогнозування від кількості епох: для пікселів від 0 до 261 (*а*), для пікселів від 261 до 522 (*б*), для пікселів від 522 до 783 (*в*)

Дані для завантаження в автоматичну систему потрібно було підготувати у такий спосіб: кожна колонка — це параметр досліджуваного об'єкта (у цьому випадку рукописна цифра). Перший рядок — це назви параметрів, а кожен наступний — це елемент даних. Насправді автоматизація процесу оброблення даних, а саме завантаження різних форматів файлів тощо, не є серйозною перешкодою для якісного використання глибокого навчання. Для табличних даних, формат представлення даних, описаний у цій статті, здається найбільш логічним та універсальним, незважаючи на суттєві труднощі, що можуть виникнути під час постановки задачі. Деякі завдання не завжди вдається розв'язати у цей автоматичний спосіб. Приклад з набором даних MNIST є демонстрацією можливостей цього підходу. Особливо це важливо для дослідників об'єктів, які не мають змоги створити власну систему оброблення даних про цей об'єкт та систему штучного інтелекту. Автоматичні системи можуть стати у пригоді дослідникам медико-біологічних, економічних, астрономічних об'єктів. Оскільки в цих галузях є багато об'єктів з неясним зв'язками між параметрами, системи штучного інтелекту дадуть змогу зрозуміти суть цих зв'язків.

У перспективі ці автоматичні системи можна розширити для оброблення графічних та часових даних. Інакше кажучи, йдеться про автоматичне оброблення цих даних і побудову системи глибокого навчання з використанням процесу згортки та поняття рекурентності. Також зауважимо, що є тенденція до використання різних варіантів згорткової та рекурентної нейронних мереж залежно від типу, кількості та унікальності даних.

ВИСНОВКИ

Запропоновано новий підхід до розв'язання проблеми, пов'язаної з автоматичним обробленням табличних даних та автоматичною побудовою системи глибокого навчання залежно від унікальності, типу та кількості даних. Продемонстровано, що для великої кількості досліджуваних об'єктів цей підхід може бути ефективним, оскільки час, витрачений на побудову всіх систем вручну, може виявитися невиправдано тривалим. Цей підхід є зручним для дослідників, які не мають досвіду роботи з системами глибокого навчання і працюють з великою кількістю досліджуваних даних. Показано, що така автоматична система є актуальною для аналізу впливу параметрів на досліджувані об'єкти. Запропоновано алгоритм роботи цієї автоматичної системи у разі представлення даних у вигляді таблиць параметрів.

Проведено експеримент з набором даних MNIST для перевірки роботи запропонованої системи. Зрозуміло, що цей приклад є простим, але водночас ефективним з погляду демонстрації можливостей системи. Показано, що параметри, які знаходиться у проміжку від 261 до 522 пікселів (у середній частині), більше впливають на кінцевий результат — розпізнання рукописного числа. Пікселі з інших проміжків дають приблизно однакову точність (77 %).

СПИСОК ЛІТЕРАТУРИ

1. Shaikh A.A., Doss A.N., Subramanian M., Jain V., Naved M., Mohiddin Md.K. Major applications of data mining in medical. *Materials Today: Proceedings*. 2022. Vol. 56, Pt. 4. P. 2300–2304. https://doi.org/10.1016/j.matpr.2021.11.642.
2. Behrad F., Abadeh M.S. An overview of deep learning methods for multimodal medical data mining. *Expert Systems with Applications*. 2022. Vol. 200. Article 117006. https://doi.org/10.1016/j.eswa.2022.117006.
3. Taranath N.L., Roopashree H.R., Yogeesh A.C., Darshan L.M., Subbaraya C.K. Medical decision support system using data mining. In: Predictive Modeling in Biomedical Data Mining and Analysis. Roy S., Goyal L.M., Balas V.E., Agarwal B., Mittal M. (Eds.). Academic Press, 2022. P. 49–64. https://doi.org/10.1016/B978-0-323-99864-2.00007-X.
4. Brescia M., Longo G. Astroinformatics, data mining and the future of astronomical research. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment.* 2013. Vol. 720. P. 92–94, https://doi.org/10.1016/j.nima.2012.12.027.

5. Settino M., Ruffolo A., La Regina F. “Mining” the sky from data mining. *Acta Astronautica*. 2006. Vol. 59, Iss. 6. P. 499–502. https://doi.org/10.1016/j.actaastro.2006.03.006.
6. Chen Y., Kong R., Kong L. Applications of artificial intelligence in astronomical big data. In: Big Data in Astronomy. Kong L., Huang T., Zhu Y., Yu S. (Eds.). Elsevier, 2020. P. 347–375. https://doi.org/10.1016/B978-0-12-819084-5.00006-7.
7. Carlbring P., Hadjistavropoulos H., Kleiboer A., Andersson G. A new era in Internet interventions: The advent of Chat-GPT and AI-assisted therapist guidance. *Internet Interventions*. 2023. 100621. https://doi.org/10.1016/j.invent.2023.100621.
8. Thelwall M., Sud P. Webometric research with the Bing Search API 2.0. *Journal of Informetrics.* 2012. Vol. 6, Iss. 1. P. 44–52. https://doi.org/10.1016/j.joi.2011.10.002.
9. Antonello F., Baraldi P., Shokry A., Zio E., Gentile U., Serio L. A novel association rule mining method for the identification of rare functional dependencies in Complex Technical Infrastructures from alarm data. *Expert Systems with Applications*. 2021. Vol. 170. 114560. https://doi.org/10.1016/j.eswa.2021.114560.
10. He Y.-L., Ou G.-L., Fournier-Viger P., Huang J.Z., Suganthan P.N. A novel dependency-oriented mixed-attribute data classification method. *Expert Systems with Applications.* 2022. Vol. 199. 116782. https://doi.org/10.1016/j.eswa.2022.116782.
11. Di Martino F., Loia V., Sessa S. Fuzzy transforms method and attribute dependency in data analysis. *Information Sciences*. 2010. Vol. 180, Iss. 4. P. 493–505. https://doi.org/10.1016/j.ins.2009.10.012.
12. Koroliouk D.V., Korolyuk V.S. Filtration of stationary Gaussian statistical experiments. *J. Math. Sci.* 2018. Vol. 229, Iss. 1. P. 30–35. https://doi.org/10.1007/s10958-018-3660-0.
13. Koroliouk D.V. Binary statistical experiments with persistent nonlinear regression. *Theor. Probability and Math. Statist.* 2015. Vol. 91. P. 71–80. https://doi.org/10.1090/tpms/967.
14. Koroliouk D., Koroliuk V.S., Nicolai E., Bisegna P., Stella L., Rosato N. A statistical model of macromolecules dynamics for Fluorescence Correlation Spectroscopy data analysis. *Statistics, Optimization & Information Computing.* 2016. Vol. 4, N 3. P. 233–242. https://doi.org/10.19139/soic.v4i3.219.
15. Koroliouk D.V. Multivariate statistical experiments with persistent non-linear regression and equilibrium. *Theor. Probability and Math. Statist.* 2016. Vol. 92. P. 71–79. https://doi.org/10.1090/tpms/983.
16. Cybenko G.V. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*. 1989. Vol. 2, N 4. P. 303–314.
17. Deng L. The MNIST database of handwritten digit images for machine learning research [best of the Web]. *IEEE Signal Processing Magazine.* 2012. Vol. 29, N 6. P. 141–142. https://doi.org/10.1109/MSP.2012.2211477.

S. Dovgyi, M. Zoziuk, D. Koroliouk

AN ADAPTIVE DEEP LEARNING SYSTEM FOR EXPLORING GENERAL DATA

Abstract. The authors present an approach to the analysis of a large amount of data that have no clear interconnection, but the nature of such a connection, if it exists, needs to be established. The possibility of automatic data processing and automatic change of parameters of the deep learning system for the formation of a deep learning platform during the analysis of data parameters is shown. It is also shown how such a system solves the needs of specialists from other fields who have never used deep learning systems. With the help of the well-known MNIST data set, it is established that with the use of individual parameters, it is possible to change their influence on the accuracy of prediction.

Keywords: neural network, automatic data processing, sampling informativeness, forecasting system, processing methods.

Надійшла до редакції 12.05.2023