УДК 378.016:004.85

Горошко Ю. В.

ORCID 0000-0001-9290-7563

Доктор педагогічних наук, професор,
завідувач кафедри інформатики і обчислювальної техніки
Національного університету «Чернігівський колегіум» імені Т. Г. Шевченка
(м. Чернігів, Україна) E-mail: horoshko_y@ukr.net

Цибко Г. Ю.

ORCID 0000-0002-1861-3003
Researcher ID AAC-6021-2021

Кандидат педагогічних наук, доцент,
доцент кафедри інформатики і обчислювальної техніки
Національного університету «Чернігівський колегіум» імені Т. Г. Шевченка
(м. Чернігів, Україна) E-mail: a.tsb@ukr.net

Костюченко А. О.

ORCID 0000-0002-6178-6444

Кандидат педагогічних наук,
старший викладач кафедри інформатики і обчислювальної техніки
Національного університету «Чернігівський колегіум» імені Т. Г. Шевченка
(м. Чернігів, Україна) E-mail: kost_andrey@ukr.net

ТЕХНОЛОГІЇ ОПРАЦЮВАННЯ ВЕЛИКИХ ДАНИХ У НАВЧАННІ ІНФОРМАТИЧНИХ ДИСЦИПЛІН

У статті розглядаються основні поняття науки про великі дані (Data Science). Проаналізовано технології машинного навчання (Machine Learning) та інтелектуального аналізу даних (Data Mining) та відповідні вільно поширювані програмні засоби і джерела даних, доцільні для застосування у навчальному процесі. Запропоновано елементи методики навчання основ Data Science майбутніх учителів інформатики і фахівців з комп'ютерних наук.

Мета. *Відібрати доцільні вільно поширювані інструменти Data Mining, та розробити окремі компоненти методики навчання дисциплін циклу фахової підготовки майбутніх учителів інформатики і фахівців з комп'ютерних наук.*

Методологія. *Вивчення та аналіз наукових публікацій, навчально-методичних видань, порівняльний аналіз програмного забезпечення, узагальнення досвіду фахівців в галузі освіти і комп'ютерних наук, моделювання і синтез компонентів методики навчання, системний підхід до навчання інформатики.*

Наукова новизна. *Відібрано доцільні вільно поширювані інструменти Data Mining, та розроблено окремі компоненти методики навчання майбутніх учителів інформатики і фахівців з комп'ютерних наук.*

Висновки. *У роботі розглянуто основні концепції і терміни сучасних технологій опрацювання і аналізу великих даних Data Science, Machine Learning, Data Mining. Проаналізовано алгоритми машинного навчання, зроблено огляд джерел навчальних даних, відібрано вільно поширювані інструменти та наведено методичні підходи до навчання Data Mining і Machine Learning майбутніх учителів інформатики і фахівців з комп'ютерних наук. Зазначені методичні підходи до навчання сучасних методів і засобів опрацювання великих даних спрямовані на формування у майбутніх фахівців спеціальних компетентностей, необхідних для ефективного моделювання, проектування, розробки, супроводу, упровадження і навчання інформаційних технологій у професійній діяльності. Таке формування може бути проведене при вивченні дисциплін «Інформаційно-комунікаційні технології», «Програмування мовою Python», «Основи штучного інтелекту та інтелектуального аналізу даних», «Вступ в Data Science та машинне навчання», що актуалізує тематику вказаних курсів.*

Ключові слова*: великі дані, машинне навчання, компетентності, KNIME, Python.*

DOI: 10.5281/zenodo.4769225

Постановка проблеми. *Актуальність роботи.* Питання, пов'язані з опрацюванням великих даних та машинним навчанням, набувають дедалі більшого значення в найрізноманітніших сферах науки, суспільства, економіки. Таке опрацювання тісно пов'язане з дослідженнями у галузі штучного інтелекту. Алгоритми штучного інтелекту розробляються і широко використовуються такими IT-гігантами, як Google, Microsoft, Apple, Amazon та іншими. В сучасні системи на чіпі (як мобільні, так уже і для звичайних комп'ютерів) вбудовано блоки, пов'язані з обчисленнями в царині елементів штучного інтелекту. Ці блоки використовуються для так званої цифрової фотографії, розпізнавання облич, інтелектуального пошуку та інших завдань. Тому важливим є питання розгляду вище окреслених питань у відповідних інформатичних дисциплінах.

Аналіз останніх досліджень та публікацій. У зв'язку з великим зацікавленням цією темою існує багато досліджень щодо огляду та прикладів застосування data mining. Наприклад, у роботі [1] здійснено огляд основних архітектур і методів машинного навчання для аналізу великих даних. У роботі [2] розглянуто основні задачі (класифікація і регресія, пошук асоціативних правил, кластеризація) і принципи роботи базових алгоритмів Data Mining у контексті їх використання для різноманітних досліджень у галузі освіти (Educational Data Mining). Багато досліджень з цієї сфери стосуються бізнес-аналітики, наприклад роботи [3, 4]. Існують також вітчизняні методичні рекомендації і посібники щодо навчання Data Science [5, 6].

Мета. Галузь Data Mining є дуже широкою, вона розвивається швидкими темпами, в ній існує багато інструментів, концепцій, джерел даних, підходів до опрацювання даних. Виникає актуальна проблема виокремити коло тих концепцій цієї галузі та інструментів для практичного застосування цих концепцій, які можуть увійти до змісту навчальних дисциплін циклу фахової підготовки майбутніх учителів інформатики та фахівців з комп'ютерних наук. Тому вбачається необхідним відібрати доцільні вільно поширювані інструменти Data Mining, та розробити окремі компоненти методики навчання.

Методологія. Вивчення та аналіз наукових публікацій, навчально-методичних видань, порівняльний аналіз програмного забезпечення, узагальнення досвіду фахівців в галузі освіти і комп'ютерних наук, моделювання і синтез компонентів методики навчання, системний підхід до навчання інформатики.

Наукова новизна. Відібрано доцільні вільно поширювані інструменти Data Mining, та розроблено окремі компоненти методики навчання майбутніх учителів інформатики і фахівців з комп'ютерних наук.

Результати дослідження. Для початку розглянемо основні терміни, використовувані в цій галузі знань. Data Science – це наука про дані на стику різних дисциплін: математика і статистика; інформатика та комп'ютерні науки; бізнес і економіка. Термін Data Science запропонував Вільям Клівленд, професор університету Пердью, один із найвідоміших фахівців у статистиці, візуалізації даних і машинному навчанні. Клівленд стверджував, що наука про дані базується на таких галузях: 1) статистична теорія; 2) статистичні моделі; 3) статистичні методи і методи машинного навчання; 4) алгоритми для статистичних методів і методів машинного навчання і оптимізації; 5) обчислювальні системи для аналізу даних; 6) живий аналіз даних, результати якого оцінюються за результатами, а не за методологією і системами, які використовувалися. Узагальнюючи, можна визначити Data Science як дисципліну, що поєднує в собі різні напрямки статистики, інтелектуальний аналіз даних (data mining), машинне навчання і застосування СУБД для вирішення складних завдань, пов'язаних з обробленням даних [7].

Вікіпедія [8] визначає Data Mining як процес напівавтоматичного аналізу великих баз даних з метою пошуку корисних фактів. Задачі, що виникають у сфері Data Mining, зазвичай поділяють на задачі класифікації, моделювання та прогнозування.

Машинне навчання – це підрозділ штучного інтелекту, який розглядає побудову алгоритмів, що можуть навчатися на наявних даних [9].

Задача машинного навчання, як правило, полягає в побудові такої моделі, за якою будуть ефективно описуватися наявні дані і визначатися достовірні прогнози у певній предметній галузі.

Такими прогнозами можуть бути відповіді на питання: яка емоція у людини на зображенні? Чи є сенс зараз вкладатися в купівлю акцій? Чи вступлять студенти до магістратури?

Проблеми машинного навчання досліджуються досить тривалий час, але донедавна вони мали суто теоретичний характер. Проте на даний час застосування алгоритмів машинного навчання вже стало поширеною практикою.

Існують алгоритми «навчання» 3-х видів.

Алгоритми «навчання з учителем». За цим алгоритмом розв'язують ряд прикладів, кожен з яких складається з двох частин: 1) значення, що подають на вхід; 2) значення, яке повинно бути на виході.

У процесі «навчання» параметри алгоритму модифікуються таким чином, щоб за вхідними даними отримувати вихідні значення, максимально наближені до бажаних.

Алгоритми «навчання з заохоченням». В прикладах, що пропонуються розв'язати за допомогою таких алгоритмів, не вказують бажане вихідне значення, але після проходження кожного прикладу виставляється оцінка, як було виконане завдання, добре чи погано.

Алгоритми «навчання без вчителя». За допомогою алгоритму розв'язують набір прикладів (без бажаного значення на виході) і в процесі їх опрацювання в системі відбуваються певні процеси самоорганізації, що призводять до модифікації параметрів так, що за допомогою алгоритму стає можливим розв'язати певну задачу [10].

Найбільш поширеними методами машинного навчання для задач класифікації є штучні нейронні мережі (artificial neural network), логістична регресія, метод опорних векторів (Support Vector Machine – SVM) та випадковий ліс (random forest) або множина вирішальних дерев.

Штучна нейронна мережа – математична модель, а також її програмна або апаратна реалізація, побудована за принципом організації та функціонування біологічних нейронних мереж – мереж нервових клітин живого організму. Це поняття виникло у ході вивчення процесів, що перебігають у мозку, зокрема під час спроби моделювання цих процесів [11].

Логістична регресія – статистичний регресійний метод моделювання залежності між векторною змінною та скаляром (вихідним значенням). Цей метод є узагальненням методу лінійної регресії з використанням softmax функції (логістичної функції для багатовимірного випадку) і застосовується, коли залежна змінна може набувати лише скінченну множину значень [12].

Метод опорних векторів – категорія універсальних мереж прямого поширення – запропонований в 1963 р. Вапніком [13]. Метод SVM набув поширення для класифікації, регресії та ідентифікації новизни.

Random forest – один з поширених методів машинного навчання, що полягає у використанні ансамблю дерев рішень [14]. Застосовується для задач класифікації, регресії і кластеризації. Дерево рішень будується на основі навчальної вибірки з використанням поняття інформаційної ентропії.

Для побудови моделей машинного навчання необхідно мати вибірки різноманітних даних. На даний час існує досить багато легальних способів отримання чужих великих даних для розв'язування власних задач користувача. Зокрема, можна виокремити такі три способи:

•Використання готових наборів даних (datasets).

•Інтернет-портали і спільноти з Data Science та Machine Learning.

•Використання відкритих даних з веб-платформ, що надають різноманітну статистику.

Серед готових наборів даних на міжнародному рівні найбільш популярними можна вважати:

Kaggle (https://www.kaggle.com/). Платформа корпорації Google для опрацювання даних в різних сферах. Якість розміщених даних може досить сильно відрізнятися і потребувати очищення та нормалізації вибірок, проте всі вони абсолютно безкоштовні. Окрім того, можна завантажити власні бази даних.

Dataset Search (https://datasetsearch.research.google.com/). База даних наповнюється різноманітними міжнародними організаціями, зокрема, такими як Всесвітня Організація Охорони Здоров'я, Statista, Гарвард та ін. На цьому ресурсі набори даних згруповані за різними ознаками.

Відкриті набори даних Microsoft Azure (https://docs.microsoft.com/en-us/azure/azure-sql/public-data-sets). Вони містять урядові дані США, інші статистичні та наукові дані, а також дані з онлайн сервісів користувачів Microsoft.

Інтернет-портали і спільноти з Data Science та Machine Learning. Зокрема досить популярними вибірками для вивчення машинного навчання вважаються наступні:

Titanic (https://www.kaggle.com/biswajee/titanic-dataset) – структурована вибірка про пасажирів, що вижили після аварії Титаніка.

Boston (http://lib.stat.cmu.edu/datasets/boston). Набір даних містить ціни на нерухомість в Бостоні.

Дані смертей та битв з гри престолів (https://www.kaggle.com/mylesoneill/game-of-thrones). Об'єднання трьох джерел даних, що базуються на інформації з серії книг «Гра престолів».

Розпізнавання квітів (https://www.kaggle.com/alxmamaev/flowers-recognition). Містить 4242 зображення квітів.

Перелік ML-дата сетів з Вікіпедії

(https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research). Містить класичні набори даних, які досить часто використовуються для написання наукових статей.

Відкриті дані з веб-платформ. Відкриті дані можуть надходити з різних джерел, серед яких державні органи та наукові співтовариства.

Єдиний державний веб-портал відкритих даних (https://data.gov.ua/). Український урядовий веб-сайт, який розпочав роботу у квітні 2014 року.

Knoema (https://knoema.com/). Набори даних щодо міжнародних організацій, подій та ситуацій.

•Зрозуміло, що всі наведені вище методи роботи з великими даними базуються на використанні відповідного програмного забезпечення. Серед усього різноманіття вказаних програмних засобів зосередимось на тих, що є вільно поширюваними і кросплатформенними.

•*KNIME*

•Перша відібрана нами програма – KNIME (Konstanz Information Miner). Це вільно поширювана платформа з відкритим кодом для всіх етапів аналізу даних, що включають: читання даних з різних джерел, перетворення і фільтрацію, власне аналіз, візуалізацію та експорт [15]. Перевагою цієї платформи є велика бібліотека реалізованих алгоритмів, в тому числі data mining і машинного навчання, зрозумілий графічний інтерфейс користувача, відсутність вимоги наявності навичок програмування. Саме використання цієї програми, на нашу думку, є педагогічно доцільним для початкового ознайомлення студентів з основами Data Mining.

•Широке використання KNIME на практиці розпочалося у 2006 році. Платформа застосовується у таких галузях як фармація, CRM-аналітика, бізнес-аналітика, аналіз текстових та фінансових даних тощо.

Розглянемо більш детально інтерфейс цієї програми. Документ, що опрацьовується у KNIME, називається робочим процесом (workflow). Він складається з вузлів (node) з'єднаних між собою. Вузол приймає вхідний набір даних, певним чином опрацьовує його і передає на вихід.

Всі вузли KNIME зібрані у репозиторій, де вони розподілені за категоріями, утворюючи дерево вузлів.

Наприклад, категорія IO містить підкатегорії Читання (Read), Запис (Write), Інші (Other), Обробка файлів (File Handling), Кеш (Cache) та інші.

В інтерфейсі KNIME широко використовується Drag-and-Drop. З репозиторію дані можна перетягти на workflow, також за допомогою перетягування можна з'єднати вузли workflow між собою. Зрозуміло, що кожен вузол потребує налаштування. На рисунку 1 можна побачити приклад workflow, що реалізує приклад нейронної мережі під назвою багатошаровий перцептрон (MLP - multilayer perceptron).

Бібліотеки Python. Іншим відібраним нами інструментом для використання машинного навчання є мова Python, оскільки для неї розроблені різноманітні бібліотеки, в котрих вже визначені основні структури даних для нейронних мереж; серед яких найбільш відомими є бібліотеки Tensorflow, Torch, Theano та Keras. [16, 17]. Для більш зручної роботи, як інтегроване середовище розробки можна використовувати Jupyter Notebook (https://jupyter.org/), який є потужним інструментом для розробки та подання проектів Data Science в інтерактивному вигляді. Вибір цього інструменту зумовлено тим, що для студентів НУЧК імені Т.Г. Шевченка Python є базовою мовою програмування.

Лабораторні роботи. Формування у студентів компетентностей щодо здійснення інтелектуального аналізу даних може відбуватися в процесі виконання розглянутих нижче циклів лабораторних робіт, які передбачають використання вільно поширюваних програмних засобів і джерел даних з відкритим доступом.

У зв'язку з суттєвим оновленням і розширенням шкільного курсу інформатики виникає необхідність відповідним чином модернізувати освітні компоненти програм підготовки майбутніх учителів інформатики. Пропоновані елементи методики доцільно включити до курсів «Інформаційно-комунікаційні технології» та «Програмування мовою Python». Вони також можуть знайти застосування у навчанні майбутніх фахівців з комп'ютерних наук у складі курсів «Основи штучного інтелекту та інтелектуального аналізу даних» та «Вступ в Data Science та машинне навчання».

KNIME

Перша лабораторна робота присвячена формуванню компетентностей щодо інтерфейсу програми. Студенти ознайомлюються з різними вузлами, їх призначенням та налаштуванням.

Розглядаються наступні вузли:

1. Вузли зчитування даних (Excel Reader (XLS), CSV Reader, File Reader)

2. Вузли перегляду (Interactive Table, Box Plot, Scatter Plot). Відображення даних у вигляді таблиці, відображення статистичних даних, точкової діаграми.

3. Вузли для маніпуляції з рядками (Concatenate, Row Filter). Об'єднання, відбір рядків.

4. Вузли маніпуляції для стовпців (Column Filter, Column Rename). Відбір та перейменування стовпців.

У другій лабораторній роботі студентам пропонується розробити нейронну мережу на основі багаторівневого перцептрона для розв'язування задачі класифікації. Студентам надається таблиця із згенерованими даними про ПІБ студентів та їх оцінками, отриманими під час навчання на бакалавраті. Також таблиця містить стовпчик «Вступ», який для кожного студента містить значення «Так» або «Ні» – результат вступу у магістратуру. Перед студентами ставиться задача: зібрати штучну нейронну мережу, завантажити таблицю з даними та розділити її на дві частини. Одну частину треба відправити на навчання нейронної мережі, а іншу – для перевірки роботи вже навченої мережі. У результаті роботи за мережею повинен бути сформований додатковий стовпчик «Передбачення «Вступ»«, також заповнений значеннями «Так» або «Ні». Далі студенти повинні провести статистичний аналіз стовпчиків «Вступ» та «Передбачення «Вступ»« для встановлення якості передбачення за нейронною мережею. Вигляд розробленої нейронної мережі можна побачити на рис. 1.

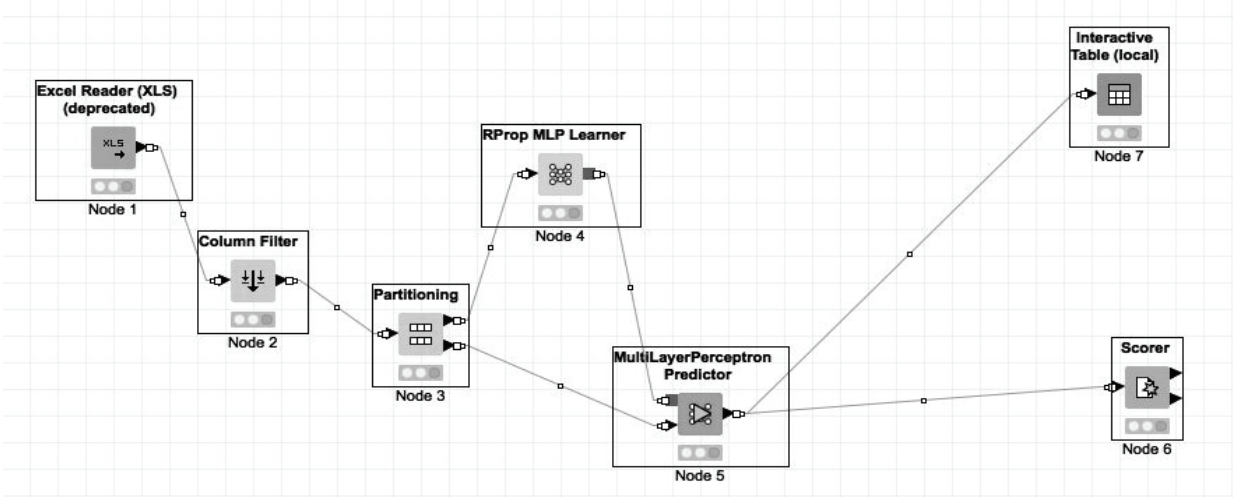

Рис. 1. Структура нейронної мережі

Бібліотеки Python

Мета цього циклу робіт – сформувати компетентності щодо обробки, аналізу та візуалізації великих даних з використанням бібліотек мови Python.

Перша лабораторна робота присвячена задачі розпізнавання рукописних цифр. Вказана задача є класичною для нейронних мереж. В якості вхідних даних можна запропонувати використати вже готову базу рукописних цифр MNIST (https://uk.wikipedia.org/wiki/MNIST_(база_даних)). Ця база містить 60000 зображень для навчання та 10000 зображень для тестування. Зображення в наборі мають розміром 28x28 px, колір задається числом від 0 до 255, як градація сірого. Для розв'язування задачі скористаємося мовою Python та бібліотекою Keras, за якою дослідникові надається певна свобода: можливість вибору кількості шарів, числа нейронів, типу шару та функції активації нейронної мережі. Для виконання завдання знадобляться такі модулі: `matplotlib`, `sklearn`, `tensorflow`, `numpy`, `itertools`. Фрагмент коду для підключення цих модулів виглядатиме так:

```
import numpy as np
import matplotlib.pyplot as plt
import gzip
from typing import List
from sklearn.preprocessing import OneHotEncoder
import tensorflow.keras as keras
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
import itertools
```

Наступним кроком є завантаження даних. Завантажити їх можна з ресурсу http://yann.lecun.com/exdb/mnist/. Цей набір розподілений по 4-х файлах: `train-images-idx3-ubyte.gz` – тренувальний набір зображень, `train-labels-idx1-ubyte.gz` – тренувальний набір міток, `t10k-images-idx3-ubyte.gz` – тестовий набір зображень, `t10k-labels-idx1-ubyte.gz` – тестовий набір міток. Для початку доцільно скористатися тільки тренувальним набором. Для його завантаження слід виконати команди:

```
%%bash
wget -q http://yann.lecun.com/exdb/mnist/train-images-idx3-ubyte.gz
wget -q http://yann.lecun.com/exdb/mnist/train-labels-idx1-ubyte.gz
```

Є деякі нюанси щодо отриманих даних, а саме: тренувальний набір міток `train-labels-idx1-ubyte` містить значення міток від 0 до 9, записаних однобайтовим числом. Проте на початку файлу міститься 8 байт метаданих, які можуть бути використані низькорівневими мовами програмування, тому їх можна пропустити. Для зчитування даних з файлу необхідно виконати наступні операції:

1. Відкрити файл, розпакувавши його з використанням бібліотеки gzip.
2. Прочитати весь масив байт в пам'ять.
3. Пропустити перші 8 байт.
4. Перебрати кожний байт і привести його до цілого числа.

Програмний код виглядатиме так:

```
with gzip.open('train-labels-idx1-ubyte.gz') as train_l:
    train_f = train_l.read()
data_l = train_f[8:]
labels = [int(l_byte) for l_byte in data_l]
```

Отримання зображень дещо відрізняється від отримання міток. В файлі зображень перші 16 байт також містять метадані, котрі також можна пропустити. Як було зазначено раніше, кожне зображення подається у вигляді байтового масиву 28x28. Отже, потрібно читати по одному зображенню за раз і зберігати їх в масиві. Програмний код:

```
SIZE_OF_IMG = 28 * 28
imgs = []
with gzip.open('train-images-idx3-ubyte.gz') as train_i:
    train_i.read(16)
    for _ in range(60000):
        img = train_i.read(size=SIZE_OF_IMG)
        img_np = np.frombuffer(img, dtype='uint8') / 255
        imgs.append(img_np)
imgs = np.array(imgs)
imgs.shape
```

За останньою операцією виводиться розмірність отриманого масиву, для даного набору це буде (60000, 784). Тобто в масиві 60000 зображень, кожне з яких подається бітовим вектором розмірності SIZE_OF_IMG. Для виведення на екран окремих зображень слід описати функцію, в якій буде використана бібліотека matplotlib.

```
def plot_img(pixels: np.array):
    plt.imshow(pixels.reshape((28, 28)), cmap='gray')
    plt.show()
```

Окрім того, доцільно дещо перекодувати наявні мітки, зробивши їх векторами. Кожен вектор буде складатися з 10 цифр (оскільки є 10 міток), кожна позиція в якому буде відповідати цифрі від 0 до 9. Тобто за міткою 3 буде створено вектор [0 0 0 1 0 0 0 0 0 0], в якому на третій позиції (враховуючи нумерацію з нуля) міститиметься одиниця, що і визначатиме цифру 3. Зокрема, ці дані будуть передаватися на відповідні шари нейронних мереж.

```
l_np = np.array(labels).reshape((-1, 1))
encoder = OneHotEncoder(categories='auto')
labels_vector = encoder.fit_transform(l_np).toarray()
```

Перевірка правильності формування даних може бути здійснена за командою: `labels_vector`[346]

Буде отримано: `array([0., 0., 0., 0., 0., 0., 0., 0., 0., 1.])`

За командою: `plot_img(imgs[346])`

буде отримано: 

Можна бачити, що зображення з індексом 346 подає цифру 9, а 346 вектор містить одиницю на 9-й позиції (враховуючи нумерацію з нуля), тобто визначає цифру 9. Отже, дані вичитані правильно.

Для тренування та перевірки нейронної мережі слід розбити тренувальний набір на 2 частини: для тренування та для тестування.

```
X_train, X_test, y_train,
y_test = train_test_split(imgs, labels_vector)
```

Тепер власне можна перейти до тренування нейронної мережі. Перш за все необхідно створити модель та додати шари.

```
model = keras.Sequential()
```

На вхід першого шару нейронної мережі буде подаватися SIZE_OF_IMG значень – в розглядуваному прикладі це 784, кількість нейронів на виході дорівнюватиме 128 (units=128), крім того, як функцію активації вкажемо 'relu'.

```
model.add(keras.layers.Dense(input_shape=(SIZE_OF_IMG,), units=128,
activation='relu'))
```

У задачі з розпізнавання цифр є певна особливість. На виході передбачається отримання вказівки на певну цифру. Тому необхідно створити мережу, яка буде містити 10 виходів, в результаті на кожному з яких буде отримана деяка ймовірність того, що досліджуване зображення має відношення до певної цифри. Тому другий шар буде приймати дані від першого шару, кількість нейронів на виході буде рівна 10, а як функцію активації слід вказати 'softmax' (розподіл ймовірностей для кожного результату).

```
model.add(keras.layers.Dense(units=10, activation='softmax'))
```

Завершальним етапом створення моделі є підготовка її до роботи (так звана компіляція) з вказанням оптимізатора (він призначений для налаштування вагових коефіцієнтів розроблюваної мережі так, щоб наблизитися до точки з найменшими втратами) та метрику для оцінки.

```
model.compile(optimizer='sgd',
              loss='categorical_crossentropy',
              metrics=['accuracy'])
```

Для навчання моделі необхідно скористатися функцією fit().

```
model.fit(X_train, y_train, epochs=20, batch_size=128)
```

Тепер можна оцінити модель та переглянути, як вона працює, передавши до неї тестові набори даних.

```
scores = model.evaluate(X_test, y_test, verbose=0)
print(«Точність: %.2f%%» % (scores[1]*100))
```

У результаті отримано: Точність: 92.27%.

Отже, побудована нейронна мережа з ймовірністю 92% правильно визначає рукописні цифри.

Слід перевірити побудовану мережу за допомогою зображень з тестової вибірки. Для цього можна взяти випадкове зображення – наприклад картинку з індексом 856 та вивести її екран.

```
plot_img(X_test[856])
```

Далі доцільно пропустити зображення цифри через нейронну мережу і проаналізувати отриманий результат.

```
cifra_resuls = model.predict(X_test[1010].reshape((1, -1)))
```

Очікується, що кожне зображення буде відповідати лише одній цифрі. Проте відповідно до шару softmax буде отримано розподіл ймовірностей тієї чи іншої цифри.

Вивівши отриманий розподіл:

```
array([[1.5865073e-05,  5.4601705e-06,  9.2759860e-01,  5.6979667e-02,
2.5663927e-09,     2.6977782e-06,     1.4370844e-08,     5.4974902e-07,
1.5396825e-02, 4.7750814e-07]], dtype=float32)
```

можна бачити, що 2-й індекс (нумерація починається з нуля) дійсно найближче до одиниці (0,92), а це означає, що зображення з більшою долею ймовірності було визначене як двійка.

Якщо дослідникові потрібно не бачити отриманий розподіл, а відразу отримати результуючу цифру, то замість методу predict можна скористатися методом predict_classes і отримати 2.

```
cifra_resuls = model.predict_classes(X_test[866].reshape((1, -1)))
cifra_resuls[0]
```

Отже, маємо навчену модель розпізнавання рукописних цифр.

Далі студентам можна запропонувати самостійно виконати такі завдання та порівняти результати навчання нейронної мережі – як змінюється точність та час навчання:

– завантажити тестовий набір зображень та міток (t10k-images-idx3-ubyte.gz, t10k-labels-idx1-ubyte.gz) та перевірити побудовану модель;

– скомпілювати модель з оптимізатором, який використовує алгоритм Адама (optimizer='adam');

– додати більше шарів з поступовим зменшенням вихідних нейронів;

– додати шар виключення для попередження перенавчання, який випадковим чином усуває з'єднання між шарами (model.add(Dropout(0.2) – відкине 20% існуючих з'єднань;

– додати шар пакетної нормалізації, яка нормалізує вихідні дані, що будуть надходити в наступний шар (model.add(BatchNormalization()));

– спробувати використати RandomForestClassifier (з бібліотеки scikit-learn) замість нейронної мережі.

Ще одним цікавим завданням може бути створення студентами зображень із власними рукописними цифрами і перевірка їх на побудованій нейронній мережі. Для цього перш за все необхідно підготувати зображення рукописних цифр. Ці зображення мають бути чорно-білими і мати розмір 28x28 px.

Далі слід скористатися наступною функцією розпізнавання зображення з файлу [18]:

```
def recognition_img(model, img_file):
    IMG_SIZE = 28
    img = keras.preprocessing.image.load_img(img_file,
    target_size=(IMG_SIZE, IMG_SIZE), color_mode='grayscale')
    img_arr = np.expand_dims(img, axis=0)
    img_arr = 1 - img_arr/255.0
    img_arr = img_arr.reshape((1, IMG_SIZE*IMG_SIZE))
    result = model.predict_classes([img_arr])
    return result[0]
```

При виклику функції recognition_img, як параметрами необхідно передати побудовану модель та ім'я файлу з рукописною цифрою (recognition_img(model, 'digit.png')).

Запропоновані студенту додаткові завдання є змістом другої лабораторної роботи.

Висновки. У роботі розглянуто основні концепції і терміни сучасних технологій опрацювання і аналізу великих даних Data Science, Machine Learning, Data Mining. Проаналізовано алгоритми машинного навчання, зроблено огляд джерел навчальних даних, відібрано вільно поширювані інструменти та наведено методичні підходи до навчання Data Mining і Machine Learning майбутніх учителів інформатики і фахівців з комп'ютерних наук. Зазначені методичні підходи до навчання сучасних методів і засобів опрацювання великих даних спрямовані на формування у майбутніх фахівців спеціальних компетентностей, необхідних для ефективного моделювання, проектування, розробки, супроводу, упровадження і навчання інформаційних технологій у професійній діяльності. Таке формування може бути проведене при вивченні дисциплін «Інформаційно-комунікаційні технології» та «Програмування мовою Python», що актуалізує тематику вказаних курсів, приводить їх до реалій значно більшого обсягу компетентностей випускника середньої школи, що вивчав інформатику з 5-го класу. Для майбутніх фахівців з комп'ютерних наук зазначені підходи доцільно застосувати під час навчання курсів «Основи штучного інтелекту та інтелектуального аналізу даних» та «Вступ в Data Science та машинне навчання».

Постійне вдосконалення і розширення кола технологій роботи з великими даними і сфер їх застосування відкриває широкі перспективи для подальших досліджень у галузі їх теоретичних, прикладних і методичних аспектів. Враховуючи важливість питань, розглянутих у статті, доцільно у майбутньому вдосконалити освітню програму підготовки учителів інформатики, ввівши в неї освітній компонент, пов'язаний з Data Mining.

References

1. Аксютина Е. М., Белов Ю. С. Обзор архитектур и методов машинного обучения для анализа больших данных. *Электронный журнал : наука, техника и образование.* 2016, №1 (5). С. 132–139. URL: http://nto-journal.ru/uploads/articles/0b9bd6d9833003ed0d6f9bb16fab81f1.pdf (дата звернення: 05.02.2021).
   Aksyutina E. M., Belov Yu. S. (2016). Obzor arhitektur i metodov mashinnogo obucheniya dlya analiza bolshih dannyh [Overview of architecture and machine learning methods for big d ata analysis]. *Elektronnyiy zhurnal : nauka, tehnika i obrazovanie – Electronic journal : Science, Technology and Education*, 1 (5), 132–139. Retrieved from : http://nto-journal.ru/uploads/articles/0b9bd6d9833003ed0d6f9bb/16fab81f1.pdf.
2. Ковальчук Ю. О. Пошук, отримання й аналіз даних в освіті: сучасний стан і перспективи розвитку. *Інформаційні технології і засоби навчання*, 2015. Том 50. № 6. С. 152–164. DOI: 10.33407/itlt.v50i6. 1284
   Kovalchuk, Yurii (2016). Poshuk, otrymannia y analiz danykh v osviti: suchasnyi stan i perspektyvy rozvytku [Data mining in education : current state and perspectives of development]. *Informatsiini tekhnolohii i zasoby navchannia – Information Technologies and Learning Tools*, 50, 152–164. DOI: 10.33407/ itlt.v50i6.1284.
3. Кузьміна О. М. Застосування методів інтелектуального аналізу даних у бізнес-середовищі. URL : https://ir.vtei.edu.ua/g.php?fname=25802.pdf (дата звернення: 05.02.2021).
   Kuzmina, O. M. Zastosuvannia metodiv intelektualnoho analizu danykh u biznes-seredovyshchi [Application of data mining methods in the business environment]. Retrieved from : https://ir.vtei.edu.ua/g.php?fname=25802.pdf.
4. Пронін С. В., Усиченко О. Ю. Аналіз інструментів машинного навчання для аналізу великих масивів даних. URL : http://publications.ntu.edu.ua/avtodorogi_i_stroitelstvo/108/67.pdf (дата звернення : 05.02.2021).
   Pronin, S.V., Usychenko, O.Iu. Analiz instrumentiv mashynnoho navchannia dlia analizu velykykh masyviv danykh [Analysis of machine learning tools for analysis of large data sets]. Retrieved from : http://publications.ntu.edu.ua/avtodorogi_i_stroitelstvo/108/67.pdf.
5. Конспект лекцій з дисципліни «Інтелектуальний аналіз даних» для студентів спеціальності 8.04030301 «Системний аналіз і управління». Дніпропетровськ, 2014 рік. 50 с. URL : https://sau.nmu.org.ua/ua/osvita/metod/magistr/Intellectual_data_analysis/(Lecture)_NMU_SAU.pdf (дата звернення: 05.02.2021).
   Konspekt lektsii z dystsypliny «Intelektualnyi analiz danykh» dlia studentiv spetsialnosti 8.04030301 «Systemnyi analiz i upravlinnia» [Summary of lectures on the subject «Data Mining» for students majoring in 8.04030301 «Systems Analysis and Management»]. Dnipropetrovsk, 2014. Retrieved from : URL:https://sau.nmu.org.ua/ua/osvita/metod/magistr/Intellectual_data_analysis/(Lecture)_NMU_SAU.pdf.
6. Могильний С. Б. Машинне навчання з використанням мікрокомп'ютерів: навч.-метод. посіб. / за ред. О. В. Лісового та ін. Київ, 2019. 226 с.

Mohylnyi, S. B. (2019). Mashynne navchannia z vykorystanniam mikrokompiuteriv: navch.-metod. posib [Machine learning using microcomputers : training manual]. Kyiv.

7. Моторин Р. М. Роль статистики у підготовці фахівців з дослідженням даних (Data Science). *Нові джерела та методи поширення даних у статистиці: матеріали XVIII Міжнародної науково-практичної конференції з нагоди Дня працівників статистики.* Київ. «Інформаційно-аналітичне агентство», 2020. С. 103–106.
Motoryn, R.M. (2020). Rol statystyky u pidhotovtsi fakhivtsiv z doslidzhenniam danykh (Data Science). [The role of statistics in the training of specialists in data research] (Data Science). *Novi dzherela ta metody poshyrennia danykh u statystytsi: materialy XVIII Mizhnarodnoi naukovo-praktychnoi konferentsii z nahody Dnia pratsivnykiv statystyky – New sources and methods of data dissemination in statistics. Materials of the XVIII International scientific-practical conference on the occasion of the Day of Statistics*. Kyiv. «Informatsiino-analitychne ahentstvo», 103–106.
8. Дописувачі Вікіпедії «Добування даних». *Українська Вікіпедія.* URL: https://uk.wikipedia.org/wiki/Добування_даних (дата звернення : 13.02.2021).
Dobuvannia danykh [Data Mining]. Retrieved from : https://uk.wikipedia.org/wiki/Добування_даних.
9. Лавренюк М. С., Новиков О. М. Огляд методів машинного навчання для класифікації великих обсягів супутникових даних. *Системні дослідження та інформаційні технології.* Київ, 2018. № 1. С. 52–71.
Lavreniuk, M. S., Novykov, O. M. (2018). Ohliad metodiv mashynnoho navchannia dlia klasyfikatsii velykykh obsiahiv suputnykovykh danykh [Overview of machine learning methods for classifying large amounts of satellite data]. *Systemni doslidzhennia ta informatsiini tekhnolohii – Systems research and information technology.* Kyiv, 1, 52–71.
10. Горошко Ю. В. Інформаційне моделювання у підготовці учителів математики та інформатики: Навчально-методичний посібник для студентів. Чернігів : Видавець Лозовий В. М., 2012. 368 с.
Horoshko, Y V. (2012). Informatsiine modeliuvannia u pidhotovtsi uchyteliv matematyky ta informatyky : Navchalno-metodychnyi posibnyk dlia studentiv [Information modeling in the training of future teachers of mathematics and computer science : training manual]. Chernihiv : Vydavets Lozovyi V.M.
11. Чередниченко А. О., Шура Н. О. Застосування штучних нейронних мереж як дієвого механізму прийняття ефективних управлінських рішень на підприємстві. *Глобальні та національні проблеми економіки*. Миколаїв, 2015. Випуск 4. С. 628–630.
Cherednychenko, A. O., Shura, N. O. (2015). Zastosuvannia shtuchnykh neironnykh merezh yak diievoho mekhanizmu pryiniattia efektyvnykh upravlinskykh rishen na pidpryiemstvi [The use of artificial neural networks as an effective mechanism for making effective management decisions in the enterprise]. *Hlobalni ta natsionalni problemy ekonomiky – Global and national economic problems.* Mykolaiv, Ukraine, 4, 628–630.
12. Кузнєцова Н. В., Бідюк П. І. Інформаційна технологія аналізу фінансових даних на основі інтегрованого методу. *Системні дослідження та інформаційні технології*, 2011. № 1. С. 22–33.
Kuznietsova, N.V., Bidiuk, P.I. (2011). Informatsiina tekhnolohiia analizu finansovykh danykh na osnovi intehrovanoho metodu [Information technology of financial data analysis based on an integrated method]. *Systemni doslidzhennia ta informatsiini tekhnolohii – Systems research and information technology,* 1, 22–33.
13. Haykin S. Neural networks and learning machines. Upper Saddle River. NJ, USA : Pearson, 2009. Vol. 3. 938 p.
14. Pirotti F., Sunar F., Piragnolo M. Benchmark of machine learning methods for classification of a Sentinel-2 image. International Archives of the photogrammetry, Remote Sensing & Spatial Information Sciences. 2016. Vol. 41. P. 335–340.
15. Обзор Knime Analytics Platform – open source системы для анализа данных. URL : https://habr.com/ru/post/320500/ (дата звернення: 05.02.2021).
Obzor Knime Analytics Platform – open source systemy dlia analyza dannykh [Review of Knime Analytics Platform – open source systems for data analysis]. Retrieved from : https://habr.com/ru/post/320500/.
16. Грас Дж. Data Science. Наука о данных с нуля. СПб. БХВ-Петербург, 2019. 337 с.
Joel Grus (2019). Nauka o dannyih s nulya [Data Science from Scratch]. St. Petersburg. BHV-Peterburg.
17. Андреас Мюллер, Сара Гвидо. Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными. Москва. Вильямс, 2020. 480 с.
Andreas Myuller, Sara Gvido. (2020). Vvedenie v mashinnoe obuchenie s pomoschyu Python. *Rukovodstvo* dlya spetsialistov po rabote s dannyimi [Introduction to Machine Learning with Python A Guide for Data Scientists]. Moscow. Vilyams.
18. Машинное зрение на Python. Обучаем нейросеть распознавать цифры. URL: https://medium.com/@enduranceprog/machine-vision-digits-94eb258c6ff8 (дата звернення: 09.02.2021).
Mashinnoe zrenie na Python. Obuchaem neyroset raspoznavat tsifryi [Machine vision by Python. We train the neural network to recognize numbers] Retrieved from : https://medium.com/@enduranceprog/machine-vision-digits-94eb258c6ff8.

Horoshko Y.

ORCID 0000-0001-9290-7563

Doctor of Pedagogical Sciences, Professor,
Head of Department of Computer Science and Engineering,
T. H. Shevchenko National University «Chernihiv Colehium»
(Chernihiv, Ukraine) E-mail: horoshko_y@ukr.net

Tsybko H.

ORCID 0000-0002-1861-3003
Researcher ID AAC-6021-2021

PhD in Pedagogical Sciences, Associate Professor,
Associate Professor of Department
of Computer Science and Engineering,
T. H. Shevchenko National University «Chernihiv Colehium»
(Chernihiv, Ukraine) E-mail: a.tsb@ukr.net

Kostiuchenko A.

ORCID 0000-0002-6178-6444

PhD in Pedagogical Sciences,
Senior Lecturer of Department
of Computer Science and Engineering,
T. H. Shevchenko National University «Chernihiv Colehium»
(Chernihiv, Ukraine) E-mail: kost_andrey@ukr.net

BIG DATA PROCESSING TECHNOLOGIES IN COMPUTER SCIENCE TEACHING

The article considers basic concepts of data science. The technologies of Machine Learning and Data Mining and the corresponding free software and data sources suitable for use in the educational process are analyzed. Elements of the methodology of teaching the basics of Data Science to future computer science teachers and computer scientists are offered.

Article's purpose *is to select appropriate free Data Mining tools and develop some components of the methodology of teaching disciplines of the cycle of professional training of future computer science teachers and computer scientists.*

Methodology. *Study and analysis of scientific publications, educational and methodical publications, comparative analysis of software, generalization of experience of specialists in education and computer science, modeling and synthesis of components of teaching methods, a systematic approach to teaching computer science.*

Scientific novelty. *Appropriate freely available Data Mining tools have been selected, and some components of the methodology for teaching future computer science teachers and computer scientists have been developed.*

Conclusions. *The main concepts and terms of modern technologies of processing and analysis of big data such as Data Science, Machine Learning, Data Mining are considered in the article. Machine learning algorithms are analyzed, sources of educational data are reviewed, freely distributed tools are selected, and methodological approaches to training Data Mining and Machine Learning of future computer science teachers and computer science specialists are presented. These methodological approaches to teaching modern methods and tools for processing large data are aimed at forming in future professionals the special competencies needed for effective modeling, design, development, maintenance, implementation and training of information technology in professional activities. Such formation can be carried out at studying courses «Information and communication technologies», «Programming in Python», «Fundamentals of artificial intelligence and data mining», «Introduction to Data Science and machine learning», which actualizes the subjects of these courses.*

Keywords*: Big Data, machine learning, competences, KNIME, Python.*

Стаття надійшла до редакції 15.02.2021

Рецензент: доктор педагогічних наук, професор ***О. М. Торубара***