

О. Г. ЗІНОВ'ЄВА

старший викладач кафедри комп'ютерних наук
Таврійський державний агротехнологічний університет
імені Дмитра Моторного
ORCID: 0000-0003-3760-8952

ОГЛЯД ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РОБОТИ З НЕЙРОННИМИ МЕРЕЖАМИ

Одними з найважливіших областей досліджень та розробок сучасної кібернетики є галузі машинного навчання, розпізнавання образів та комп'ютерного зору. Прискорення темпів розвитку технологій інформаційного суспільства, розвиток робототехніки, розвиток концепцій «розумний дім» та «розумне місто», розвиток інтернету речей та систем штучного інтелекту визначають цій галузі особливе місце в сучасному науковому знанні. Багато прикладних завдань у практиці сучасного програмування використовуються методи збору даних, кластеризації і класифікації, методи статистичного вивода. У повсякденне життя, як і в корпоративне, і в промислове середовище починають впроваджуватися технології, що поступово стирають межу між реальним і віртуальним простором, що вимагає нового якісного рівня повсюдно впроваджуваних технологій розпізнавання, чия сфера застосування останніми роками виросла колосально: завдання розпізнавання, що вважались найскладнішими раніше. сьогодні цілодобово вирішуються мобільними пристроями пересічних громадян. Комп'ютеризовані простори з вираженою топологією, такі як «розумний дім» рядового користувача, розрахована на багато користувачів доповнена і розрахована на багато користувачів віртуальна реальності різного ступеня занурення, ускладнюється штучний інтелект в комп'ютерних іграх різного призначення вимагають нових ідей і підходів, нового рівня точності і швидкості розпізнавання.

Ця стаття присвячена порівняльному аналізу деяких програмних інструментів глибокого навчання, яких останнім часом з'явилося безліч [1]. До таких інструментів відносяться програмні бібліотеки, розширення мов програмування, а також самостійні мови, що дозволяють використовувати готові алгоритми створення та навчання нейромережевих моделей. Існуючі інструменти глибокого навчання мають різний функціонал та вимагають від користувача різного рівня знань та навичок. Правильний вибір інструмента – важливе завдання, що дозволяє досягти необхідного результату за найменший час і з меншою витратою сил.

Ключові слова: машинне навчання, нейронні мережі, бібліотеки.

O. G. ZINOVIEVA

Senior Lecturer at the Department of Computer Sciences
Dmytro Motornyi Tavria State Agrotechnological University
ORCID: 0000-0003-3760-8952

OVERVIEW OF SOFTWARE FOR WORKING WITH NEURAL NETWORKS

One of the most important areas of research and development in modern cybernetics are the fields of machine learning, pattern recognition, and computer vision. The acceleration of the pace of development of information society technologies, the development of robotics, the development of the "smart home" and "smart city" concepts, the development of the Internet of Things and artificial intelligence systems determine this field's special place in modern scientific knowledge. Many applied tasks in the practice of modern programming use methods of data collection, clustering and classification, methods of statistical inference. In everyday life, as well as in the corporate and industrial environment, technologies are beginning to be introduced that gradually erase the border between real and virtual space, which requires a new qualitative level of universally implemented recognition technologies, whose scope of application has grown enormously in recent years: recognition tasks that were considered the most difficult before. today are solved around the clock by the mobile devices of ordinary citizens. Computerized spaces with a pronounced topology, such as the "smart home" of an ordinary user, designed for many users augmented and designed for many users of virtual reality of various degrees of immersion, artificial intelligence in computer games for various purposes require new ideas and approaches, new level of accuracy and speed of recognition

This article is devoted to a comparative analysis of some deep learning software tools, many of which have appeared recently [1]. Such tools include software libraries, extensions of programming languages, as well as independent languages that allow the use of ready-made algorithms for creating and training neural network models. Existing deep learning tools have different functionality and require different levels of knowledge and skills from the user. Choosing the right tool is an important task that allows you to achieve the desired result in the shortest time and with less effort.

Key words: machine learning, neural networks, libraries.

Постановка проблеми

В останні роки методи глибокого навчання – нейронні мережі – дозволили досягнути великих успіхів в таких галузях, як комп'ютерний зір, обробка природної мови, обробка аудіо. Нейронні мережі використовуються для розв'язання складних задач, які потребують аналітичних обчислень, аналогічних тим, які виконує людський мозок. Однією з причин широкого застосування глибоких нейронних мереж є та, що мережа автоматично виділяє з даних важливі ознаки, необхідні для розв'язання задачі. Бібліотеки нейронних мереж є важливими складовими сучасних моделей глибокого навчання. Бібліотеки є важливими програмними інструментами, які прискорюють та спрощують роботу з алгоритмами, створення та навчання нейронних мереж. Однак на даний момент існує велика кількість бібліотек, на вибір яких доведеться витратити багато часу. У цій статті було розглянуто основні програмні модулі для роботи з нейронними мережами. Дослідження показало, що вибір бібліотеки залежатиме від низки умов: обмеження обчислювальних потужностей, операційної системи, мови програмування, і які дані потрібно обробляти. Результати дослідження розширюють знання про бібліотеки для нейронних мереж.

В даний час машинне навчання є областю наукових досліджень, що активно розвивається. Це з можливістю швидко збирати і обробляти дані, і навіть з можливістю виявлення цих даних різних законів, якими протікають різні процеси. Коли визначити такий закон є досить складним, використовують глибоке навчання. У цій роботі наводиться порівняння бібліотек глибокого навчання таких як Caffe, Pylearn2, PyTorch, Theano та TensorFlow.

Аналіз останніх досліджень і публікацій

Зростання популярності глибоких нейронних мереж, яке відбувається в останні роки можна пояснити різними факторами. По-перше, зросла продуктивність комп'ютерів, що дозволило навчати глибокі нейронні мережі значно швидше та з більш високою точністю [4]. Раніше тих обчислювальних потужностей не вистачало для навчання якоїсь складної мережі, яка застосовувалась для рішення практичних завдань. По-друге, був накоплений великий обсяг даних, який необхідний для навчання глибоких нейронних мереж. До того ж розроблені методи, які дозволяють швидко і якісно навчати мережі, що складаються зі ста та більше шарів [5]. Ці ознаки привели до прогресу в навчанні нейронних мереж та їх практичному використанню.

Формулювання мети дослідження

Мета даного дослідження – розглянути різні методи, що використовуються для глибокого навчання нейронних мереж, виділити галузі застосування таких засобів, а також показати нові шляхи та тенденції їх розвитку.

Викладення основного матеріалу дослідження

Існує безліч програмних засобів для вирішення завдань глибокого навчання, які надають широкі можливості для створення, навчання та застосування нейронних мереж у різних галузях.

Програмні бібліотеки DeepLearnToolbox, Theano, Pylearn2, Deepnet, Torch, Darch реалізують найширший спектр методів глибокого навчання. Розробники надають можливості для створення пов'язаних нейромереж (fully connected neural network, FC NN), згорткових нейронних мереж (convolutional neural network, CNN), автокодувальників (autoencoder, AE) та обмежених машин Больцмана (restricted Boltzmann machine, RBM). Простота інших бібліотек, таких як Caffe, nnForge, Cuda CNN та інші, незважаючи на те, що вони мають меншу функціональність, у деяких випадках допомагає досягти більшої продуктивності [7].

Для програмування нейронних мереж в даний час найчастіше використовується мова Python завдяки безлічі бібліотек з набором вбудованих математичних функцій. Python – проста у вивченні мова, яка активно використовується в шкільній та університетській освіті. Через це він набув популярності не лише в промисловому програмуванні, але й серед професіоналів, які використовують програмування як інструмент дослідження. Істотним недоліком Python є його низька продуктивність, яку можна подолати, написавши критичні частини програмного забезпечення на скомпільованій мові (зазвичай це C або C++) або за допомогою перекладача cython [5]. Багато бібліотек машинного навчання також написані двома або більше мовами. Частина програмного забезпечення, яка обробляє основну частину обчислень, зазвичай реалізована на C або C++ (основна частина або бекенд). Розглянемо деякі з найбільш поширених бібліотек.

Бібліотека Caffe – це відкрита бібліотека глибокого навчання, яка розроблена групою вчених з університету Берклі. Вона спроектована для ефективної реалізації конволюційних нейронних мереж та глибоких нейронних мереж у загальному. Однією з ключових особливостей Caffe є його швидкодія, що робить її популярним вибором для багатьох дослідників та розробників.

Caffe реалізована з використанням мови програмування C++, є обгортки на Python та MATLAB. Офіційно підтримувані операційні системи – Linux і OS X, також неофіційний порт на Windows. Caffe використовує бібліотеку BLAS (ATLAS, Intel MKL, OpenBLAS) для векторних та матричних обчислень. Поряд з цим, до зовнішніх залежностей входять glog, gflags, OpenCV, protobuf, boost, leveldb, nappy, hdf5, lmdb. Для прискорення обчислень Caffe може бути запущена на GPU з використанням базових можливостей технології CUDA або бібліотеки примитивів глибокого навчання cuDNN.

Розробники Caffe підтримують можливість створення, навчання та тестування повністю пов'язаних та згорткових нейромереж. Вхідні дані та перетворення описуються поняттям шару. Залежно від формату зберігання можна використовувати такі типи шарів вихідних даних:

- DATA – визначає шар даних у форматі leveldb та lmbd;
- HDF5_DATA – шар даних у форматі hdf5;
- IMAGE_DATA – простий формат, який передбачає, що у файлі наведено список зображень із зазначенням мітки класу та інші.

У процесі навчання моделей застосовують різні методи оптимізації. Розробники Caffe надають реалізацію низки методів:

- стохастичний градієнтний спуск (Stochastic Gradient Descent, SGD);
- алгоритм з адаптивною швидкістю навчання (adaptive gradient learning rate algorithm, AdaGrad);
- прискорений градієнтний спуск Нестерова (Nesterov's Accelerated Gradient Descent, NAG).

У бібліотеці Caffe топологія нейромереж, вихідні дані та спосіб навчання задаються за допомогою конфігураційних файлів у форматі prototxt. Файл містить опис вхідних даних (тренувальних та тестових) та шарів нейронної мережі.

Бібліотека Caffe використовується в багатьох дослідницьких лабораторіях, компаніях та навчальних закладах для розробки та застосування нейронних мереж для різних завдань у галузі штучного інтелекту та комп'ютерного зору.

Бібліотека Pylearn2 – це бібліотека для машинного навчання та глибокого навчання, яка розроблена на мові програмування Python. Вона є продовженням і покращенням попередньої бібліотеки Pylearn, і має багато функцій, спрямованих на полегшення створення та навчання глибоких нейронних мереж

У Pylearn2 підтримується можливість створення повністю пов'язаних та згорткових нейромереж, різних видів автокодувальників (Contractive Auto-Encoders, Denoising Auto-Encoders) та обмежених машин Больцмана (Gaussian RBM, the spike-and-slab RBM). Передбачено кілька функцій помилки: крос-ентропія (cross-entropy), логарифмічна правдоподібність (log-likelihood). Є такі методи навчання:

- пакетний градієнтний узвіз (Batch Gradient Descent, BGD);
- стохастичний градієнтний узвіз (Stochastic Gradient Descent, SGD);
- нелінійний метод сполучених градієнтів (Nonlinear conjugate gradient descent, NCG).

У бібліотеці Pylearn2 нейромережі задаються за допомогою їх опису в конфігураційному файлі YAML. YAML-файли є зручним та швидким способом серіалізації об'єктів, оскільки вона розроблена з використанням методів об'єктно-орієнтованого програмування

PyTorch – це бібліотека для машинного навчання та глибокого навчання, яка розроблена компанією Facebook. Вона широко використовується для створення та навчання нейронних мереж, включаючи звичайні нейронні мережі, згорткові мережі, рекурентні мережі та трансформери. Ось деякі ключові особливості бібліотеки PyTorch:

- а) динамічний граф обчислень: однією з основних особливостей PyTorch є його підтримка динамічних графів обчислень. це дозволяє реалізувати складні моделі з умовними операторами та зворотним розповсюдженням помилок без необхідності визначення графа попередньо, як у tensorflow;
- б) простота використання: PyTorch має простий та інтуїтивно зрозумілий інтерфейс, що дозволяє легко створювати, навчати та використовувати нейронні мережі;
- в) гнучкість та розширюваність: вона має модульну архітектуру, яка дозволяє легко розширювати функціональність та додавати нові модулі;
- г) підтримка GPU: PyTorch підтримує виконання обчислень на графічних процесорах (gpu), що дозволяє прискорити навчання моделей;
- д) інтеграція з іншими бібліотеками: вона може бути легко інтегрована з іншими бібліотеками для машинного навчання та глибокого навчання, такими як TensorFlow, що розширює її можливості та робить її більш потужним інструментом для розробки нейронних мереж.

Фреймворк створено на базі Torch, бібліотеки для мови Lua, яка призначена для математичних розрахунків та машинного навчання. Але його особливість – дотримання стилю Python і філософії програмування цією мовою.

PyTorch набуває все більшої популярності серед дослідників та розробників завдяки своїй гнучкості, простоті використання та потужним можливостям для розробки та навчання нейронних мереж.

Бібліотека Theano – це відкрита бібліотека для машинного навчання та глибокого навчання, розроблена на мові програмування Python. Вона дозволяє розробляти та оптимізувати математичні вирази, що використовуються в машинному навчанні, зокрема, для створення та навчання нейронних мереж. Ось деякі ключові особливості бібліотеки Theano

Бібліотека реалізована мовою Python, підтримується на операційних системах Windows, Linux та Mac OS. До складу Theano входить компілятор, який перекладає математичні вирази, написані мовою Python в ефективний код C або CUDA.

Theano надає базовий набір інструментів конфігурації нейромереж та їх навчання. Можлива реалізація багатошарових пов'язаних мереж (Multi-Layer Perceptron), згорткових нейромереж (CNN), рекурентних нейронних мереж (Recurrent Neural Networks, RNN), автокодувальників та обмежених машин Больцмана. Також передбачені

різні функції активації, зокрема сигмоїдальна, softmax-функція, крос-ентропія. У процесі навчання використовується пакетний градієнтний спуск (Batch SGD).

Бібліотека TensorFlow – це відкрита бібліотека для машинного навчання та глибокого навчання, розроблена компанією Google. Вона надає широкий спектр інструментів для розробки та навчання нейронних мереж, включаючи звичайні шаровані нейронні мережі, згорткові нейронні мережі, рекурентні нейронні мережі та багатошарові перцептрони.

Ключові особливості бібліотеки TensorFlow:

а) гнучкість: TensorFlow надає різні інструменти та інтерфейси для створення та налаштування різноманітних типів моделей нейронних мереж, що робить його гнучким для різних завдань машинного навчання;

б) швидкість та ефективність: TensorFlow використовує оптимізовані обчислення для роботи з великими обсягами даних та великими моделями нейронних мереж. Він може використовувати GPU та TPU для прискорення обчислень;

в) візуалізація: TensorFlow має інструменти для візуалізації графів обчислень, що допомагає розуміти та аналізувати структуру моделей;

г) модульність: TensorFlow побудований з урахуванням модульності, що дозволяє використовувати його як від індивідуальних компонентів, так і в якості повноцінного фреймворку для машинного навчання;

д) підтримка різних мов програмування: Ви можете використовувати TensorFlow з різними мовами програмування, такими як Python, C++, Java та інші.

TensorFlow підтримує кілька мов програмування. Головний з них – це Python. Крім того, є окремі пакети для C/C++, Golang та Java. А ще форк TensorFlow.js для виконання коду на стороні клієнта, в браузері, на JavaScript

TensorFlow використовується в різних галузях, включаючи комп’ютерний зір, обробку природної мови, генетику, медицину та інші, і вона залишається однією з найпопулярніших бібліотек для глибокого навчання у світі.

Порівняння бібліотек для роботи з нейронними мережами, такими як TensorFlow, PyTorch, та інші, може бути корисним для визначення найбільш підходящого інструменту для конкретного проекту.

Основні критерії, за якими порівнюються бібліотеки:

а) швидкодія бібліотек для нейронних мереж може бути розглянута у контексті навчання моделей та інференції (використання моделей для передбачень);

б) легкість використання може включати як простоту встановлення та конфігурування, так і зручний інтерфейс програмування;

в) документація та підтримка;

г) гнучкість та розширюваність – важливі характеристики, які дозволяють легко використовувати та розширювати функціонал для вирішення різноманітних завдань у галузі машинного навчання та глибокого навчання;

д) інтеграція з іншими бібліотеками може бути ключовим аспектом в розвитку складних програмних продуктів, особливо в галузі машинного навчання та глибокого навчання, де використовуються різноманітні інструменти та технології

В таблиці 1 наведені порівняльні характеристики бібліотек Caffe, Pylearn2, PyTorch, Theano та TensorFlow за різними критеріями.

Таблиця 1

Порівняння бібліотек за різними критеріями

Критерій	Caffe	Pylearn2	PyTorch	Theano		TensorFlow
Модель програмування	Статичний граф обчислень	Не визначено	Динамічний граф обчислень	Статичний граф обчислень		Статичний граф обчислень
Швидкодія	++	+	+	++		++
Легкість використання	+	+	++	+		+
Документація та підтримка	++	+	+	+		++
Гнучкість та розширюваність	+	++	++	++		++
Інтеграція з іншими бібліотеками	+	-	+	+		+

Примітки:

++: Дуже високий рівень підтримки або функціональності.

+: Посередній рівень підтримки або функціональності.

-: Низький рівень підтримки або функціональності.

Ці бібліотеки мають різні особливості та сильні сторони, тому вибір між ними може залежати від конкретних потреб вашого проекту, вашого досвіду та інших факторів. Наприклад, Caffe може бути корисним для швидкого прототипування моделей у задачах комп’ютерного зору, тоді як PyTorch може бути популярним серед дослідників

та шукачів індивідуальності за рахунок його динамічного графу обчислень та легкості використання. TensorFlow, з іншого боку, може бути відмінним вибором для великих проєктів з глибоким навчанням завдяки своїй широкій підтримці та високій ефективності.

Висновки

На основі порівняння різних бібліотек можна зробити ряд висновків. Майже всі бібліотеки, крім PyTorch, показують приблизно однаковий час навчання. У випадку PyTorch довший час навчання можна пояснити підтримкою динамічного обчислювального графіка, який, очевидно, вимагає додаткових обчислювальних витрат. У свою чергу, TensorFlowlibrary показала середній результат точності, поступаючись PyTorch і Theano.

Список використаної літератури

1. Новотарський М.А., Нестеренко Б.Б. Штучні нейронні мережі: обчислення. Праці Інституту математики НАН України. Т. 50. Київ : Ін-т математики НАН України, 2004. 408 с.
2. Кононюк А.Ю. Нейронні мережі і генетичні алгоритми. Київ : «Корнійчук», 2008. 446 с.
3. M. Gupta Madan, Jin Liang & Homma Noriyasu. Gupta Statistic and Dynamic Neural Networks : From Fundamentals to Advanced Theory. USA : IEEE Press, 2003. 722 p.
4. Ciresan D.C., Meier U., Gambardella L.M. Schmidhuber J. Deep, Big, Simple Neural Nets for Handwritten Digit Recognition. Neural Computation. 2010. vol. 22, no. 12. pp. 3207–3220.
5. He K., Zhang X., Ren S., et al. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (Las Vegas, NV, USA, 27–30 June 2016), 2016. pp. 770–778.
6. Hinton, G.E.: Learning Multiple Layers of Representation. In: Trends in Cognitive Sciences. Vol. 11. pp. 428–434. (2007)

References

1. Novotarskyi M.A. & Nesterenko B.B. (2004) Shtuchni neironni merezhi: obchyslennia [Artificial neural networks: calculations]. *Pratsi Instytutu matematyky NAN Ukrainy*. Kyiv : Incnbnet matematyky NAN Ukrainy [In Ukrainian]
2. Kononiuk A.Iu. (2008) Neironi merezhi i henetychni alhorytmy [Neural networks and genetic algorithms]. Kyiv : «Korniichuk» [In Ukrainian]
3. Madan M. Gupta, Jin Liang & Homma Noriyasu/(2003) Statistic and Dynamic Neural Networks: From Fundamentals to Advanced Theory / M. Gupta Madan, Jin Liang, Homma Noriyasu. USA : IEEE Press,
4. Ciresan D.C., Meier U., Gambardella L.M. & Schmidhuber J. (2010) Deep, Big, Simple Neural Nets for Handwritten Digit Recognition. Neural Computation. vol. 22, no. 12.
5. He K., Zhang X., Ren S., et al. (2016) Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (Las Vegas, NV, USA, 27–30 June 2016).
6. Hinton, G.E. (2007) Learning Multiple Layers of Representation. In: Trends in Cognitive Sciences. Vol. 11.