

Ю. В. Москаленко, аспірант

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського», м. Київ

СЕРЕДОВИЩЕ МОДЕЛЮВАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧІ КЛАСТЕРИЗАЦІЇ

Задача кластеризації розв'язується в багатьох прикладних галузях. Досягнути її швидкого та достатньо точного розв'язання можна за допомогою спеціальних нейронних мереж — карт самоорганізації Кохонена. Цей тип нейронних мереж постійно вдосконалюється як на рівні алгоритмів, так і на рівні програмного забезпечення. Тому доцільно створювати спеціальні програмні інструменти, що дозволяють в однакових умовах провести навчання та безпосередньо обчислювальні експерименти з розв'язання задачі кластеризації для коректного порівняльного аналізу отриманих результатів. Другим завданням такого середовища є визначення оптимального програмного інструментарію розв'язання прикладних задач технічної діагностики, наприклад, пошуку аномалій, класифікації сигналу з втратами.

У роботі обґрунтовано обрання існуючих програмних реалізацій карт самоорганізації SOM (self-organizing map) та відповідних їм багатосарових перцептронів MLP (multilayer perceptron) для розв'язання саме задачі класифікації. Всі обрані програмні реалізації є вільнодоступними та розповсюджуються з відкритою ліцензією.

Визначено параметри SOM та MLP, на які може впливати експериментатор. Обрано критерії порівняння реалізацій SOM.

Наведено архітектуру середовища моделювання і представлено функціонал його компонентів.

Для демонстрації застосування запропонованого середовища приведено розв'язання класичних тестових задач машинного навчання. Це дозволило провести коректне порівняння результатів обчислювальних експериментів, а також ефективності програмних реалізацій як за базовим, так і за оптимізованим алгоритмами.

Ключові слова: нейронні мережі, карти самоорганізації Кохонена, SOM, GSOM, ESOINN.

Вступ. Безперервний автоматичний аналіз сигналів необхідний для штатного функціонування технічних систем. Для його реалізації широко використовуються спеціальні засоби розпізнавання, насамперед, нейронні мережі. Для розв'язання в реальному часі задач класте-

ризації сигналу, наприклад, при розпізнаванні передаварійних режимів роботи обладнання, доцільно використовувати нейронні мережі Кохонена — SOM (Self-organizing map). Ці мережі дозволяють розпізнавати вхідний сигнал змінного розміру і проводити навчання на обмеженій вибірці. Розв'язання кожної практичної задачі потребує пошуків архітектури нейронної мережі SOM та налаштування її параметрів, тому створення фреймворка для порівняння якості кластеризації актуальне та має практичну значущість.

Основні вдосконалення SOM можна умовно поділити на оптимізацію за критеріями швидкості навчання та точності роботи. Оскільки в робочому режимі контролю обладнання використовується навчена мережа, подальші дослідження були спрямовані на підвищення долі коректної кластеризації SOM. Найвідомішими сучасними оптимізаціями за критерієм точності є алгоритми: GSOM (Growing Self-Organizing Maps) [1], SOINN (Self-Organizing Incremental Neural Network) [2], ESOINN (Enhanced self-organizing incremental neural network) [3].

Програмні реалізації мереж вищезазначених алгоритмів є вільнодоступними та розповсюджуються за відкритою ліцензією. Вони представлені на ресурсі Github [4]. Всі ці реалізації розраховані на кластеризацію неперервного сигналу за дискретними значеннями через сталі проміжки часу. Однак в технічній діагностиці є задачі, в яких неможливо отримати такий сигнал, наприклад, при розв'язанні задач пошуку аномалій або розпізнаванні сигналу з втратами. Для подібних задач необхідно розробити нові алгоритми, доповнені відносно базового алгоритму SOM, і, відповідно, провести їх валідацію. Приклад такої розробки наведено в статті [5] для задачі класифікації за характеристичним сигналом з випадковими втратами на основі карт самоорганізації.

Апробування нових розробок SOM потребує створення умов для їх навчання і проведення коректного порівняльного аналізу результатів тестування з існуючими програмними реалізаціями. Крім цього для розв'язання кожної прикладної задачі доцільно проводити моделювання за параметрами їх алгоритмів для обрання оптимального в кожному поточному випадку. Для розробки вдосконалених алгоритмів SOM та оптимізацій розв'язання прикладних задач на основі існуючих розробок необхідно створити середовище моделювання карт самоорганізації Кохонена з вбудованими апробованими реалізаціями алгоритмів SOM і можливістю підключення нових розробок.

Мета: розробити середовище моделювання нейронних мереж Кохонена для порівняльного аналізу існуючих програмних реалізацій SOM при розв'язанні задачі кластеризації та доведення коректності нових модифікованих алгоритмів.

Для досягнення поставленої мети вирішено такі **задачі:**

- 1) визначити критерії моделювання в середовищі;

- 2) розробити архітектуру середовища моделювання SOM, інтерфейси та інший допоміжний функціонал, зокрема бази даних та відповідну програмну реалізацію;
- 3) провести обчислювальні експерименти в запропонованому середовищі моделювання.

Постановка задачі моделювання для нейронної мережі SOM.

При створенні середовища необхідно врахувати, що SOM виконує тільки задачу кластеризації. Для кінцевого визначення належності сигналу до певного класу необхідно додатково розв'язувати задачу розділення класів. Найчастіше для цього застосовують додаткову нейронну мережу — багатошаровий перцептрон — MLP (multilayer perceptron).

Таким чином задача розпізнавання розв'язується поетапно. Спочатку SOM навчається за поданими характеристиками. Після цього двовірна матриця значень решітки W_{som} перетворюється в одноірний вектор-стовпчик \bar{X}_{mlp} , який далі подається на вхід MLP. Навчання MLP проводиться за прикладами, отриманими на заздалегідь навченій SOM. На рисунку 1 зображено етапи розв'язання задачі класифікації з визначенням вхідної і вихідної інформації.

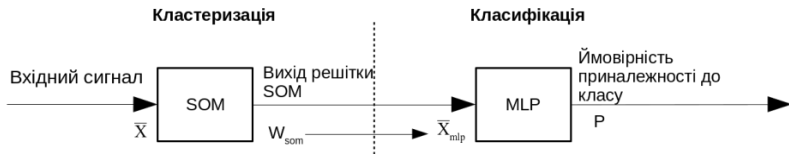


Рис. 1. Етапи навчання комплексу нейромереж

Середовище моделювання SOM може бути застосоване у двох випадках:

- 1) для обрання оптимальної реалізації карти самоорганізації при розв'язанні поточної прикладної задачі;
- 2) для верифікації та коректного порівняння нових розробок SOM з існуючими.

Тому при проектуванні середовища моделювання було проведено аналіз існуючих реалізацій SOM, як з традиційними так і з оптимізованими алгоритмами функціонування.

Середовище повинно мати можливість використовувати існуючі та зручно додавати нові розробки нейромереж Кохонена та інших типів.

Для вбудови нових мереж передбачено обгортки — wrappers, які взаємодіють з усіма мережами, в тому числі і з тими реалізаціями, які імплементовано на інших мовах програмування.

В середовище вбудовано такі різновиди SOM: GSOM [6], SOINN [7], ESOINN [8], SOMbase, SOMmod [5], Encog [9], NeuroPH [10].

Мережі Encog, NeuroPH є апробованими реалізаціями базового алгоритму SOM. Мережі GSOM, SOINN, ESOINN є одними з найефективніших алгоритмів оптимізованих карт самоорганізації. Мережі SOMbase та SOMmod є прикладами доданих мереж з відповідно базовим та оптимізованим алгоритмом. Ці мережі використано для розв'язання задачі розпізнавання вхідного сигналу з втратами.

В таблиці 1 наведено назви всіх SOM, які включені в систему моделювання, та відповідні їм MLP.

Для середовища було розроблено власний багатошаровий перцептрон — MLPbase. Однак завдяки різним програмним реалізаціям SOM, цю мережу не можна використовувати на всіх картах самоорганізації. Тому для реалізацій Encog та NeuroPH використано MLP з їхніх бібліотек.

Таблиця 1

Відповідність нейромереж SOM та MLP в середовищі

№	Мережа SOM	Мережа MLP
1	GSOM	MLPbase
2	SOINN	MLPbase
3	ESOINN	MLPbase
4	Encog	Мережа із Encog
5	NeuroPH	Мережа із NeuroPH
6	SOMbase	MLPbase
7	SOMmod	MLPbase

Для навчання комплексу передбачено задання таких параметрів:

- 1) коефіцієнт швидкості навчання (learning rate) для SOM;
- 2) коефіцієнт швидкості навчання (learning rate) для MLP;
- 3) кількість епох до зменшення коефіцієнту швидкості навчання для SOM;
- 4) коефіцієнт зменшення швидкості навчання для SOM;
- 5) величина зменшення коефіцієнта навчання MLP при виході його на плато (Learning Rate Scheduler).

Перші два параметри впливають на швидкість навчання. Третій та четвертий — потрібні для більш точного навчання SOM, п'ятий — для коректного навчання MLP. Для визначення третього та четвертого параметру необхідно проводити експерименти з дослідження помилки на тестовій вибірці при повному циклі навчання.

Ефективність реалізації SOM для розв'язання поточної задачі визначається за такими критеріями:

- час навчання SOM (T_{SOM});
- частка коректної класифікації на комплексі SOM та MLP (D);
- час роботи на певному наборі даних (T).

Архітектура середовища. Основна проблема полягає в тому, що потрібно поєднувати існуючі реалізації нейронних мереж на різних мовах програмування. Тому в архітектурі використано модулі — обгортки (wrappers) для кожної завантаженої SOM.

На рисунку 2 відображено архітектуру середовища. На рисунку виокремлено курсивом всі компоненти, які реалізовано в даному проєкті. Реалізації нейронних мереж реалізованих на інших мовах програмування, виокремлено сірим кольором. Середовище реалізовано на мові C++ з використанням бібліотеки Qt для взаємодії з базами даних та JNI для взаємодії з мовою Java.

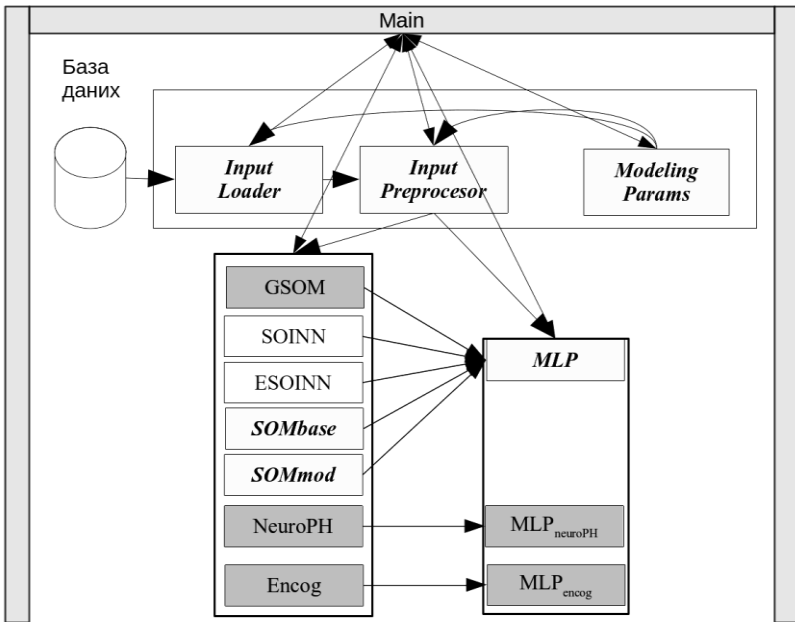


Рис. 2. Архітектура середовища моделювання карт самоорганізації

Середовище реалізовано у вигляді застосунку, який складається з таких основних модулів:

- база даних — містить тестові та навчальні вибірки;
- модуль InputLoader — завантажує навчальну вибірку з бази даних або файлової системи. Модуль після завантаження передає всю вибірку в модуль InputPreprocessor;
- модуль InputPreprocessor — обробляє навчальну вибірку. Модуль дозволяє проводити експерименти з втрат, спотворення сигналу, внесення аномалій в сигнал а також формує постановки задачі

класифікації та кластеризації в робочому режимі. При навчанні приклади подаються безпосередньо на обрану реалізацію SOM;

- модуль NNComplex — реалізує інтерфейси над різними реалізаціями нейромереж (SOM та MLP). Для мереж GSOM, NeuroPH, Encog зроблено обгортки для взаємодії додатку з реалізаціями цих мов;
- модуль ModelingParams — містить параметри навчання, такі як швидкість навчання MLP та SOM та інші;
- модуль main — реалізує основний функціонал, який безпосередньо взаємодіє з усіма модулями, проводить навчання та збір статистики.

Кожна з реалізацій SOM має відповідну їй реалізацію MLP, яка використовується автоматично.

Обчислювальні експерименти. Апробацію розробленого середовища проведено на таких задачах:

- класифікація ірисів Фішера [11];
- розпізнавання рукописних цифр за базою даних MNIST [12].

Ці задачі є класичними тестами для нейронних мереж.

Іриси Фішера — це задача класифікації ірисів трьох видів по чотирьом характеристикам:

- довжина зовнішньої частки оцвітини;
- ширина зовнішньої частки оцвітини;
- довжина внутрішньої частки оцвітини;
- ширина внутрішньої частки оцвітини.

Набір даних з класифікації ірисів містить 150 прикладів.

MNIST — база даних рукописних цифр розмірністю 28x28 пікселів. Цей набір даних містить 70000 прикладів.

В даних обчислювальних експериментах ставилася задача порівняння ефективності реалізації базових та оптимізованих SOM.

Для розв'язання було послідовно навчено всі нейронні мережі за такою послідовністю:

1. Завантаження вибірки з файлу або бази даних.

Завантажується з розподіленням на тренувальну та валідаційну вибірки.

2. Навчання SOM.

SOM навчається з початковим кроком навчання. Через певну кількість епох проводиться зменшення кроку навчання та повторення цього циклу до кінця навчання SOM.

3. Навчання MLP.

Після завершення навчання SOM, відбувається навчання MLP. Після кожної епохи відбувається перевірка точності роботи MLP. У випадку, якщо точність на валідаційній вибірці не зростає 5 епох поспіль, зменшується коефіцієнт швидкості навчання та процес продо-

вжується з новими параметрами. Якщо точність на валідаційній вибірці не зростає 12 епох — навчання MLP завершується.

В таблиці 2 зведено параметри навчального комплексу, які застосовувалися у задачах.

Таблиця 2

Параметри навчання комплексу

Параметри навчання комплексу	Набори даних	
	Іриси Фішера	MNIST
Коефіцієнт швидкості навчання SOM	0.2	0.2
Коефіцієнт швидкості навчання MLP	0.2	0.2
Кількість епох до зменшення коефіцієнту швидкості навчання для SOM2	50	30
Коефіцієнт зменшення швидкості навчання для SOM	5	5
Коефіцієнт зменшення коефіцієнта навчання MLP при виході його на плато	5	5

Для оцінювання результатів експериментів було обрано такі критерії:

- частка коректної класифікації;
- час навчання;
- час обробки набору даних.

Останні два критерії визначають ефективність розв'язання задачі. Результати експериментів зведено до таблиці 3.

Таблиця 3

Результати розв'язання задач

Реалізація SOM	GSOM	SOINN	ESOINN	SOMbase	SOMmode	ENCOG	NeuroPH
Критерій оцінювання							
Класифікація ірисів Фішера							
Час навчання SOM (T_{SOM}), сек.	3,2	3,1	3,1	1,2	1,2	1,3	1,3
Точність класифікації (D), %	92	91	92	86	86	86	86
Час роботи (T), сек.	5,5	6,1	5,4	3,1	3,1	3,3	3,3
MNIST							
Час навчання SOM (T_{SOM}), сек.	211	197	195	173	169	175	178
Точність класифікації (D), %	95.4	95.6	96.1	90.1	90.1	90.2	90.1
Час роботи (T), сек.	270	220	210	140	140	152	154

Оскільки іриси Фішера мають обмежений набір даних, представлення всіх прикладів цього набору здійснювалось 10000 разів.

Результати експериментів за класичними тестовими задачами підтверджують те, що модифіковані мережі, такі як і, ESOINN, GSOM є ефективнішими за реалізацію базового алгоритму функціонування.

Висновки:

1. Запропоновано концепцію порівняльного аналізу алгоритмів і відповідних реалізацій карт самоорганізації Кохонена для доведення коректності нових модифікованих алгоритмів SOM.
2. Представлено програмну реалізацію середовища моделювання нейронних мереж для розв'язання задачі кластеризації.
3. Наведено приклад обчислювальних експериментів з доведенням ефективності відомих реалізацій за базовим та модифікованими алгоритмами.

Список використаних джерел:

1. Cao M. Growing self-organizing map approach for semantic acquisition modeling / M. Cao, A. Li, Q. Fang, B. J. Kroger // 2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom). — 2013.
2. Furoo S. An incremental network for on-line unsupervised classification and topology learning / S. Furoo, O. Hasegawa // Neural Networks. — 2016. — Vol. 19. — P. 90–106.
3. Furoo S. An enhanced self-organizing incremental neural network for online unsupervised learning / S. Furoo, T. Ogura // Neural Networks. — 2007. — Vol. 20. — P. 893–903.
4. Github. — Access mode: <https://github.com>.
5. Shapovalova S. Increasing the share of correct clustering of characteristic signal with random losses in self-organizing maps / S. Shapovalova, Yu. Moskalenko // Eastern-European Journal of Enterprise Technologies. — 2019. — № 2/4 (98). — P. 13–21.
6. Growing Self-Organizing Map. — Access mode: <https://github.com/philippludwig/pygsom>.
7. Self-Organizing Incremental Neural Network. — Access mode: <https://github.com/fukatani/soinn>.
8. An enhanced self-organizing incremental neural network for online unsupervised learning. URL: <https://github.com/BeIBES/ESOINN>.
9. Encog Machine Learning Framework. — Access mode: <https://github.com/encog/encog-java-core>.
10. Neuroph — Java Neural Network Platform Neuroph. — Access mode: <https://github.com/neuroph/neuroph>.
11. Iris Data Set. — Access mode: <http://archive.ics.uci.edu/ml/datasets/Iris>
12. Handwritten digit database. — Access mode: <http://cis.jhu.edu/~sachin/-digit/digit.html>.

AN ARTIFICIAL NEURAL NETWORKS MODELING ENVIRONMENT FOR SOLVING A CLUSTERING TASK

The task of clustering is solving in various fields of application. In order to achieve a fast and sufficiently accurate clustering solution, it is possible to use special neural networks like Kohonen's self-organization card. This type of neural network is always improved both at the algorithm level and at the software level. So, it is necessary to create special software tools that provide the opportunity of training in the same conditions and quickly computational experiments to solve the clustering problem. And also to carry out a comparative analysis of the received results.

The second task of such software is to create examples of tasks in technical diagnostics, such as: search of abnormality, classification of signal with losses, and others.

This paper include such existing software implementations of SOM (self-organization map) and their respective MLP (multilayer perceptron) to solve precisely the classification problem.

All selected software implementations are freely available and spread under free license.

SOM and MLP parameters have been defined that may be influenced by the researcher.

Criteria for comparing SOM implementations have been selected.

This paper presents the architecture of the modeling environment and functionality of its components.

For the demonstration is taken the solution of classical problems of machine learning. It helps properly compare the results of computational experiments and to implement the effectiveness of software implementations on both basic and optimized algorithms.

Key words: *neural networks, self-organizing map, SOM, GSOM, ESOINN.*

Отримано: 14.08.2019