Науковий вісник НЛТУ України
Scientific Bulletin of UNFU
https://nv.nltu.edu.ua
https://doi.org/10.36930/40310614
Article received 29.11.2021 p.
Article accepted 29.12.2021 p.
UDC 004.8:032.26

ISSN 1994-7836 (print)
ISSN 2519-2477 (online)

@ Correspondence author
Yu. A. Lukashchuk
urijlukas@gmail.com

Ю. А. Лукащук
Національний Університет "Львівська Політехніка", м. Львів, Україна

ІМІТАЦІЙНА МОДЕЛЬ ОБЧИСЛЕННЯ ВАГОВИХ КОЕФІЦІЄНТІВ ДЛЯ НЕЙРОПОДІБНОГО ШИФРУВАННЯ ТА ДЕШИФРУВАННЯ ДАНИХ

Розроблено імітаційну модель обчислення вагових коефіцієнтів для заданої архітектури нейромережі. Ця архітектура задається розрядністю вхідного повідомлення та розрядністю входів. Проаналізовано останні дослідження та публікації щодо актуальності проблеми криптографічного шифрування та дешифрування даних у реальному часі. Для роботи було обрано метод сингулярного розкладу матриці на противагу методу головних компонентів. Передусім це було зроблено задля того, щоб мати змогу працювати із вхідною матрицею довільної розмірності, коли ж метод головних компонентів передбачає, що вхідна матриця є квадратною. Розроблена імітаційна модель ґрунтується на вдосконаленому методі сингулярного розкладу матриці, а для знаходження власних значень і власних векторів використано метод обертання Якобі. Практичною цінністю є те, що імітаційна модель забезпечує швидке обчислення коефіцієнтів для заданої архітектури нейромережі. Також розроблено гнучкий користувацький інтерфейс, який дає змогу користувачу зрозуміло та детально ознайомитись із роботою алгоритму. У розробленій аплікації можна переглянути кожен крок розрахунків, що дає змогу перевірити правильність виконання на кожному з етапів. Програмне забезпечення розроблено у середовищі Visual Studio 2019 та за допомогою мови програмування C#. Для матричних операцій підключено бібліотеку Accord.Math. Також продемонстровано роботу розробленої імітаційної моделі. Для прикладу обрано вхідне повідомлення із розрядністю – 16 та розрядністю входу – 2. Опираючись на ці дані, можна зазначити, що вхідна матриця матиме розмірність 8×2. Саме ж вхідне повідомлення задається користувачем. Як результат розраховано матрицю вагових коефіцієнтів розмірністю 2×2. Надалі ця матриця буде використовуватись під час шифрування та дешифрування вхідного повідомлення. Однак для повідомлення є вимоги, зокрема –розрядність самого повідомлення, а також розрядність входів мають бути такими ж самими, як для знаходження матриці вагових коефіцієнтів.

Ключові слова: нейромережа; SVD; нейроелементи; власні вектори; власні значення; матричні перетворення; метод обертання Якобі.

Вступ / Introduction

При передачі важливої інформації та дистанційному управлінні мобільними робототехнічними комплексами, безпілотними літальними апаратами та різноманітними мобільними транспортними системами важливою задачею є забезпечення криптографічного захисту. Розв'язання такої задачі потребує розроблення нових методів та алгоритмів криптографічного захисту, орієнтованих на ефективну програмно-апаратну реалізацію у реальному часі зі забезпеченням обмежень щодо габаритів, енергоспоживання та вартості. Одним із шляхів забезпечення таких вимог є використання автоасоціативної нейромережі прямого поширення, яка навчається на основі методу головних компонент. Особливістю таких нейромережах є можливість наперед обчислити вагові коефіцієнти та використати таблично-алгоритмічний метод для реалізації криптографічних засобів нейроподібного шифрування та дешифрування даних. Ключами під час нейроподібного шифрування та дешифрування даних є структури нейроподібної мережі (кількість нейронів, кількість входів і їх розрядність) та матриці вагових коефіцієнтів.

Тому актуальною проблемою є обчислення матриці вагових коефіцієнтів для заданої архітектури нейроподібної мережі.

Об'єкт дослідження – обчислення матриць вагових коефіцієнтів для заданих структур нейроподібних мереж.

Предмет дослідження – методи, моделі та алгоритми обчислення матриць вагових коефіцієнтів для заданих архітектур нейроподібних мереж.

Мета роботи – розробити імітаційну модель обчислення вагових коефіцієнтів для криптографічного нейроподібного шифрування та дешифрування даних.

Для досягнення зазначеної мети визначено такі основні *завдання дослідження*:

- проаналізувати останні дослідження та публікації;
- удосконалити метод сингулярного розкладу матриці та адаптації його до задачі обчислення вагових коефіцієнтів для нейроподібних мереж;
- розробити імітаційну модель обчислення вагових коефіцієнтів для криптографічного нейроподібного шифрування та дешифрування даних.

Наукова новизна отриманих результатів дослідження – розроблено імітаційну модель обчислення ва-

Інформація про автора:
Лукащук Юрій Андрійович, аспірант, кафедра автоматизованих систем управління. **Email:** urijlukas@gmail.com;
https://orcid.org/0000-0002-8933-8635

Цитування за ДСТУ: Лукащук Ю. А. Імітаційна модель обчислення вагових коефіцієнтів для нейроподібного шифрування та дешифрування даних. Науковий вісник НЛТУ України. 2021, т. 31, № 6. С. 92–96.
Citation APA: Lukashchuk, Yu. A. (2021). Imitation model for calculating weights for neuro-like data encryption/decryption. *Scientific Bulletin of UNFU, 31*(6), 92–96. https://doi.org/10.36930/40310614

гових коефіцієнтів для криптографічного нейроподібного шифрування та дешифрування даних, яка ґрунтується на вдосконаленому методі сингулярного розкладу матриці та забезпечує обчислення матриці вагових коефіцієнтів для заданої архітектури нейроподібної мережі.

*Практична значущість результатів дос*лідження – використання розробленої імітаційної моделі забезпечує швидке обчислення вагових коефіцієнтів для заданої архітектури нейроподібної мережі.

Аналіз останніх досліджень та публікацій. У роботах [1, 7, 8, 9, 14, 15] проаналізовано основні шляхи розвитку бортових засобів криптографічного захисту передачі даних у реальному часі, який показав, що перспективним шляхом розвитку є використання нейромережевих методів для шифрування та дешифрування даних [6, 10, 13].

Аналіз публікацій [3, 4, 11] показує, що нейромережевий криптографічний захист даних переважно реалізується програмним шляхом. Основним недоліком такої реалізації є складність забезпечення режиму реального часу та обмежень, які висуваються до бортових систем, щодо маси, габаритів, енергоспоживання та вартості.

У роботах [3, 9, 14] розглянуто шляхи адаптації автоасоціативної нейронної мережі з неітераційним навчанням до задач криптографічного захисту даних. Особливістю функціонування такої нейромережі є можливість попереднього обчислення вагових коефіцієнтів і їх подальше використання під час шифрування та дешифрування даних. У роботі [15] показано, що для нейромережевого криптографічного шифрування та дешифрування даних використовуються ключі, до складу яких входять архітектура нейромережі та матриці вагових коефіцієнтів.

Аналіз робіт показує [1, 7, 8, 12], що для попереднього обчислення вагових коефіцієнтів застосовується метод головних компонент, який використовує систему власних векторів, які відповідають власним значенням коваріаційної матриці вхідних даних. У роботі [2] показано, що модель послідовних геометричних перетворень знаходиться в основі неітеративного методу навчання автоасоціативної нейронної мережі для операцій шифрування-дешифрування даних, для якої величини змінних, отримані внаслідок навчання мережі, перераховуються у відповідні вагові коефіцієнти міжнейронних зв'язків.

Матеріали та методи дослідження. Першим важливим кроком є вибір архітектури нейроподібної мережі для криптографічного шифрування та дешифрування даних. Архітектура нейроподібної мережі визначається кількістю нейроелементів N, кількість входів k і розрядність входів m. Кількість нейронних елементів визначають за такою формулою:

$$N = \frac{n}{m}, \tag{1}$$

де: n – розрядність повідомлення, m – розрядність входів. Вхідні повідомлення, які шифруються, можуть мати різну розрядність n та ділитися на різну кількість входів k, які дорівнюють кількості нейроелементів N. Від значення розрядності повідомлення n та кількості входів k залежить архітектура нейромережі.

Основним методом під час дослідження, а надалі під час розроблення є метод сингулярного розкладу матриці (англ. *Singular Value Decomposition*, далі SVD).

Сингулярний розклад матриці – розкладання прямокутної матриці певного типу. Має широке застосування, через свою наочну геометричну інтерпретацію у вирішенні багатьох прикладних завдань. Переформулювання сингулярного розкладання, так зване розкладання Шмідта, має додатки в квантовій теорії інформації, наприклад, у заплутаності. SVD схожий на метод головних компонентів (МГК), але є більш загальним. МГК передбачає, що вхідна матриця є квадратною, SVD не має такого припущення.

Загальна формула SVD така:

$$A = UDV^T, \tag{2}$$

де: A – матриця вхідних даних Nxn; U – ліва сингулярна матриця NxN, стовпці містять власні вектори матриці AA^T; D – діагональна матриця Nxn, що містить сингулярні (власні) значення; V – права сингулярна матриця nxn, стовпці містять власні вектори матриці A^TA.

Для розрахунку власних значень та власних векторів використовували метод обертання Якобі.

Метод обертання Якобі – це ітераційний метод обчислення власних значень і власних векторів симетричної матриці (процес відомий як діагоналізація). Він названий на честь Карла Густава Якоба Якобі, який вперше запропонував цей метод в 1846, але набув поширення тільки в 1950-х роках з появою комп'ютерів.

Суть алгоритму зводиться до того, щоб для заданої матриці $S = S^{(0)}$ побудувати послідовність ортогональних подібних матриць $S^{(1)}$, $S^{(2)}$,…, $S^{(m)}$, які сходяться до діагональної матриці, на діагоналях якої стоять власні значення матриці S. Для побудови цієї послідовності використовується спеціально підібрана матриця обертання J_i, така, що норма наддіагональної частини

$$\left\| A^{(i)} \right\|_{off} = \sqrt{\sum_{1 \le j < \kappa \le v} \left(a_{jk}^{(i)} \right)^2} \tag{3}$$

зменшується при кожному двосторонньому обертанні матриці

$$A^{(i+1)} = J_i^T \cdot A^{(i)} \cdot J_i . \tag{4}$$

Отже, для розрахунку матриці U на вхід методу Якобі передається результат добутку AA^T. А для знаходження матриці V – результат добутку A^TA. Для того щоб знайти матрицю D достатньо взяти власні значення, які були знайдені при розрахунку матриці U чи матриці V. І розмістити їх на головній діагоналі.

Після знаходження матриць U, V та D відбувається розрахунок вагових коефіцієнтів. За основу взято формулу із джерела [4]

$$Aw = UD , \tag{5}$$

де: A – вхідна матриця розмірністю Nxn; w – матриця вагових коефіцієнтів розмірністю nxn; матриці U та D беруться з результату роботи SVD.

Тепер виражаємо матрицю вагових коефіцієнтів

$$w = A^{-1} \cdot UD . \tag{6}$$

Матрицю A^{-1} можна розписати за формулою, запропонованою у джерелі [4]

$$A^{-1} = VD^{-1} \cdot U^T . \tag{7}$$

Підставивши (7) у (6), отримаємо формулу розрахунку вагових коефіцієнтів

$$w = VD^{-1} \cdot U^T \cdot UD \tag{8}$$

Варто зазначити, що в разі, коли матриця D буде прямокутна, буде розраховуватись псевдоінверсна матриця, яка має назву ще псевдообернена матриці, тобто

узагальненої оберненої матриці. Тепер для того, щоб зашифрувати вхідні дані, достатньо скористатися формулою із джерела [13]

$$\bar{y} = w\bar{a}\,, \qquad (9)$$

де: y – вектор результуючої матриці із зашифрованими даними; w – матриця вагових коефіцієнтів, a – вектор вхідної матриці.

Результати дослідження та їх обговорення / Research results and their discussion

Результатом роботи є розроблене програмне забезпечення для розрахунку вагових коефіцієнтів, які використовуються для шифрування вхідного повідомлення. Для реалізації поставленої задачі обрано мову програмування C# і середовище розробки – Visual Studio 2019.

Вхідними даними для програми є n – розрядність повідомлення, m – розрядність входів та саме повідомлення (рис. 1). Після введення даних програма розраховує кількість нейроелементів і входів. На основі введених і отриманих даних утворюється вхідна матриця. За цю операцію відповідає кнопка *Generate matrix*.

Наступним кроком відбувається знаходження матриць U, V та D за допомогою алгоритму SVD. Нагадаємо, що сингулярний розклад матриці (сингулярне подання матриці, SVD) – один з важливих методів розкладання матриці з дійсними або комплексними числами. Є узагальненням власного розкладання матриці невід'ємно визначеної нормальної матриці (наприклад, симетричної матриці з додатними власними значеннями) на матрицю розміру $m \times n$ як узагальнення полярного розкладу. За цю опцію відповідає кнопка *Calculate SVD*. Як було описано вище, для розрахунку власних значень та власних векторів використовується метод обертання Якобі. Результат проведених розрахунків продемонстровано на рис. 2-4. Справа на рис. 2,*a* показано результат методу Якобі над результатом добутку матриць AA^T. На рис. 2,*b* продемонстровано результат методу Якобі над результатом добутку A^TA.

Матриця D складається із власних значень, розміщених на головній діагоналі (див. рис. 3,*a*). Далі відбувається знаходження матриці Z, яка буде використовуватись для перевірки роботи SVD. Ця матриця є результатом множення матриці U на матрицю D.

На рис. 4 зображено матрицю, яка є результатом множення матриці Z на матрицю V. За рахунок неї можна переконатись у правильності роботи алгоритму. Оскільки ця матриця має бути рівною вхідній матриці, хоча і з певною похибкою.

Нарешті при натисканні на кнопку *Calculate weights* відбувається розрахунок вагових коефіцієнтів за формулою (8). Для перевірки правильності пророблених розрахунків використовується формула (5).

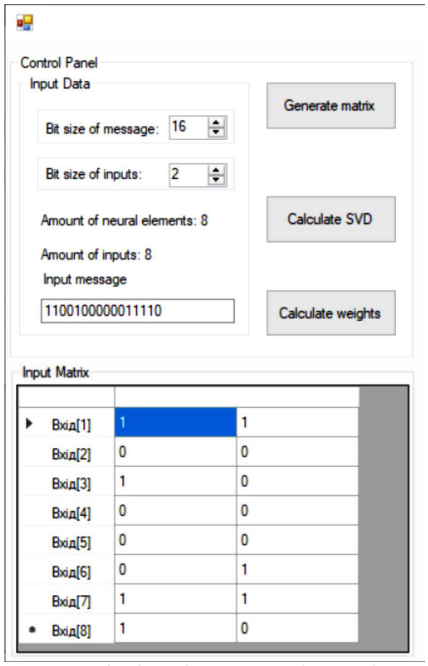

Рис. 1. Вхідні дані програми / Input data for the program

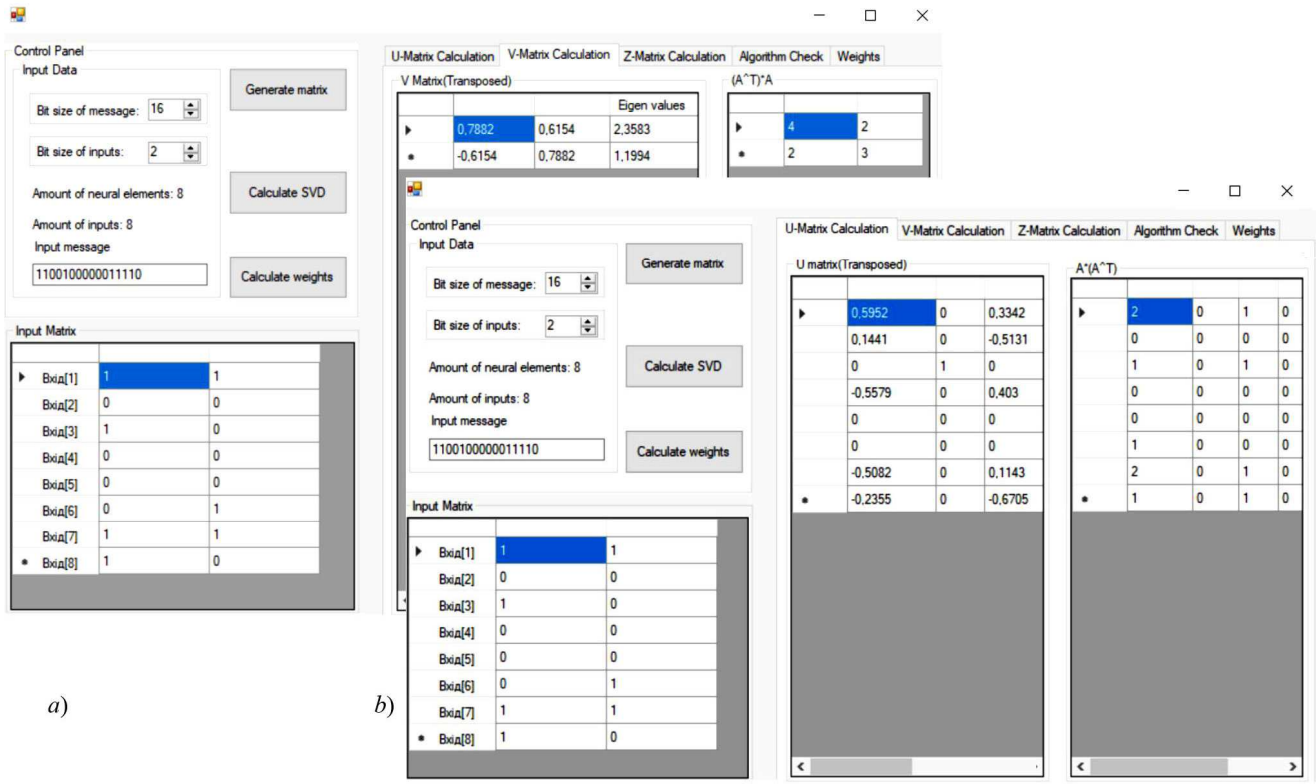

Рис. 2. Розрахунок матриці U / Calculation of the U matrix

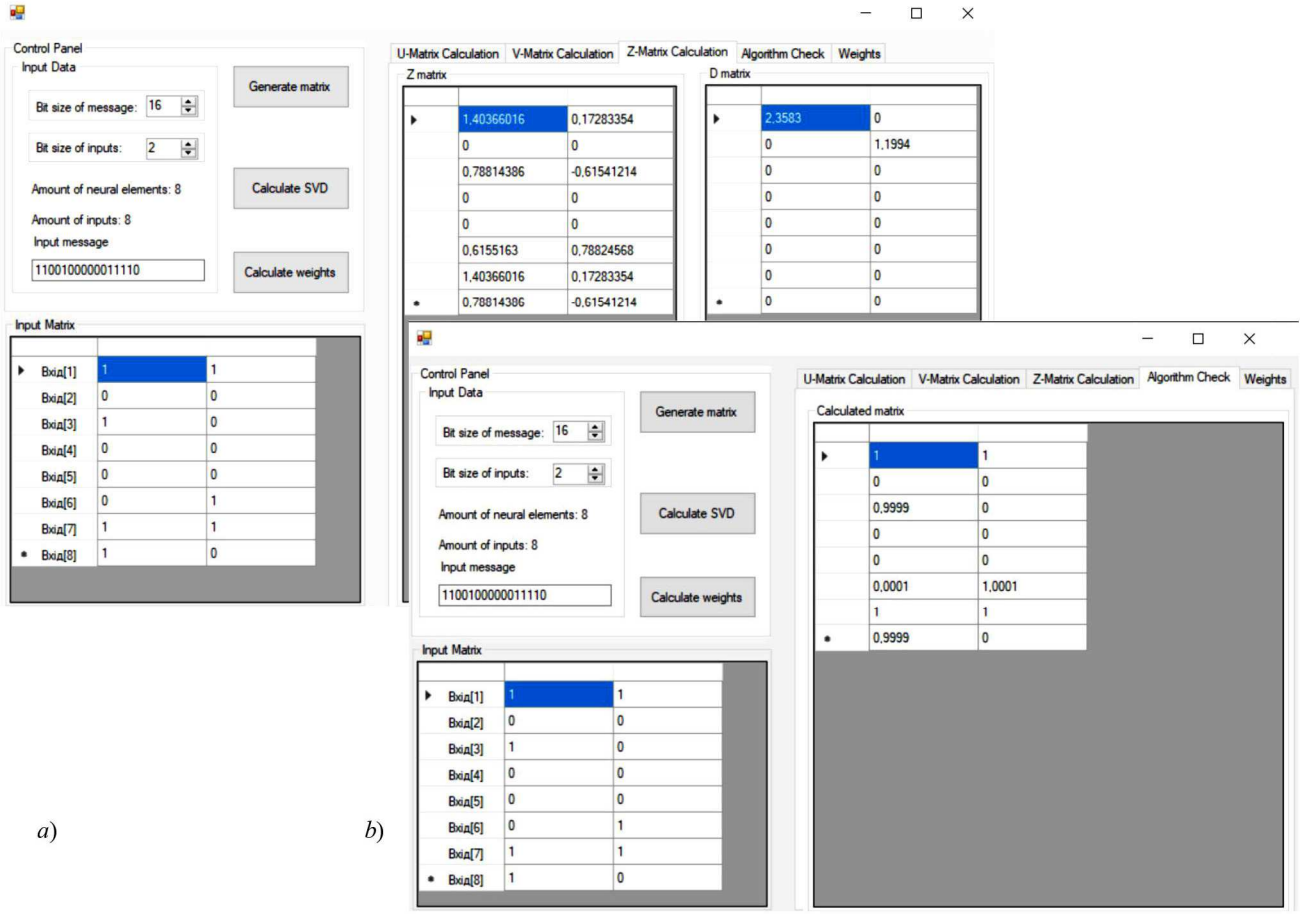

Рис. 3. Знаходження матриці D та розрахунок матриці Z / Finding the D matrix and calculating the Z matrix (*a*); матриця перевірки / Verification matrix (*b*)

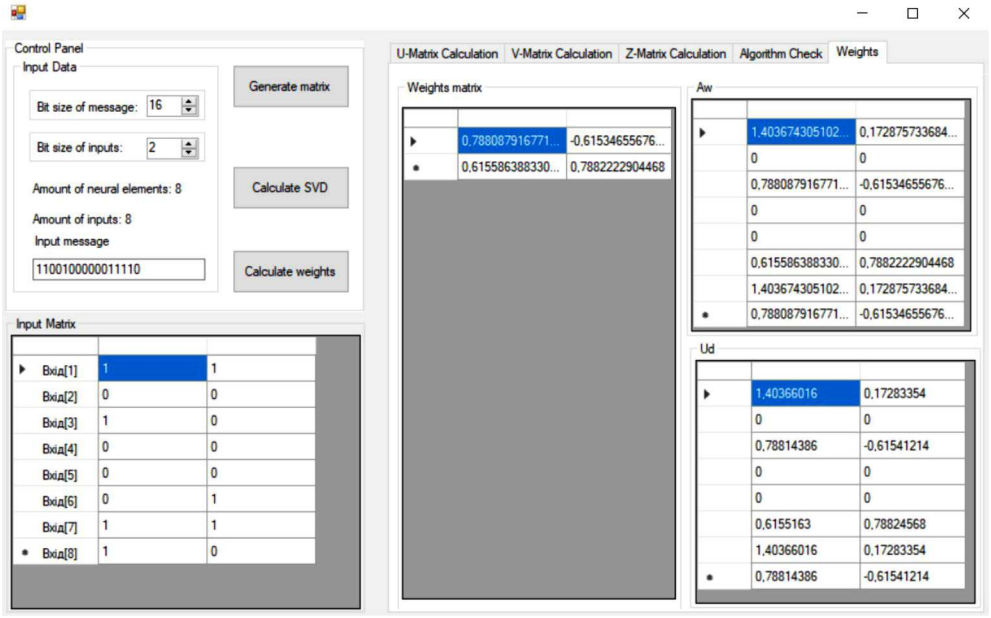

Рис. 4. Матриця вагових коефіцієнтів / Matrix of weights

Обговорення результатів дослідження. Для прикладу використовувалось 16-розрядне повідомлення з розрядністю входів – 2. Опираючись на ці дані, розраховувались кількість нейроелементів та кількість входів – 8.

Упродовж роботи вийшло розрахувати матрицю вагових коефіцієнтів, але зважаючи на результати добутків матриць помітно, що є певна похибка. Маємо на увазі результат, продемонстрований на рис. 3,*b*, а саме різниця між елементами матриць *Calculated Matrix* та *Input Matrix*. Calculated Matrix – це матриця, яка була розрахована за *формулою* (2). Основна її мета – це продемонструвати точність знайдених матриць *U*, *V* та *D*, які в подальшому використовуються для знаходження матриці вагових коефіцієнтів. У цьому випадку похибка вийшла 0,0001 і зумовлена вона матричними перетвореннями і обраним заокругленням в чотири знаки після коми при розрахунках. Однак, можна зазначити, що у цьому випадку ця похибка не мутує результуючу матрицю вагових коефіцієнтів. Результат і процес розрахунку вагових коефіцієнтів було порівняно із результа-

тами продемонстрованими у роботі [11]. У цій праці автори також знаходять матрицю вагових коефіцієнтів, однак для своїх розрахунків використовують метод головних компонентів.

У подальшому ця матриця буде використовуватись під час шифрування та дешифрування вхідного повідомлення. Однак для повідомлення будуть вимоги, щоб розрядність самого повідомлення, а також розрядність входів були такими ж самими, як при знаходженні матриці вагових коефіцієнтів.

Висновки / Conclusions

1. Розроблено імітаційну модель знаходження вагових коефіцієнтів для заданої архітектури нейромережі. Роботоздатність такої моделі продемонстровано на прикладі повідомлення *1100100000011110*. Розрядність повідомлення у цьому випадку – 16(n = 16), розрядність входів – 2(m = 2), кількість нейроелементів – 8(N = 8), кількість входів – 8(k = 8).

2. Удосконалено метод сингулярного розкладу матриці для знаходження матриці вагових коефіцієнтів. Це вдалося зробити внаслідок аналізу джерела [4] та формули (5)–(8).

3. Практична цінність полягає у тому, що використання розробленої імітаційної моделі забезпечує швидке обчислення вагових коефіцієнтів.

References

1. Arvandi, M., Wu, S., Sadeghian, A., Melek, W. W., & Woungang, I. (2006). Symmetric cipher design using recurrent neural networks. Proceedings of the IEEE International Joint Conference on Neural Networks, 2039–2046.
2. Chang, A. X. M., Martini, B., & Culurciello, E. (2015). Recurrent neural networks hardware implementation on FPGA: arXiv preprint arXiv:1511.05552.
3. Chi, Zhang, Wei, Zou, Liping, Ma, & Zhiqing, Wang. (2020). Biologically inspired jumping robots: A comprehensive review, Robotics and Autonomous Systems, vol. 124.
4. Diamantaras, K. I., & Kung, S. Y. (1996). Principal Component Neural Networks. Theory and Applications (Wiley, 1996), 270 p.
5. Rabyk, V., Tsmots, I., Lyubun, Z., & Skorokhoda, O. (2020). Method and Means of Symmetric Real-time Neural Network Data Encryption. 2020 IEEE 15th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2020 – Proceedings, 1, 47–50.
6. Riznik, O. Ia., Tkachenko, R. O., & Kinash, Iu. Ye. (2019). Neiromerezheva tekhnologiia zakhistu ta peredachi danikh u realʹnomu chasi z vikoristanniam shumopodibnikh kodiv. Innovatciini tekhnologii u rozvitku suchasnogo suspilʹstva: zbirnik tez dopovidei mizhnarodnoi naukovo-praktichnoi konferentcii (Lʹviv, 18–19 kvitnia 2019 r.), 19–23. [In Ukrainian].
7. Sagar, V., & Kumar, K. A. (2014). Symmetric Key Cryptographic Algorithm Using Counter Propagation Network (CPN). Proceedings of the 2014 ACM International Conference on Information and Communication Technology for Competitive Strategies.
8. Shihab, K. A. (2006). Backpropagation neural network for computer network security. *Journal of Computer Science*, vol. 2, no. 9, 710–715.
9. Śledź, S., Ewertowski, M. W., & Piekarczyk, J. (2021). Applications of unmanned aerial vehicle (UAV) surveys and Structure from Motion photogrammetry in glacial and periglacial geomorphology. Geomorphology 2021, 378 p.
10. Tcimbal, Iu. V. (2018). Neiromerezhevii metod simetrichnogo shifruvannia danikh. Visnik Natcionalʹnogo universitetu "Lʹvivsʹka politekhnika". Seriia: Informatciini sistemi ta merezhi, 901, 118–122. [In Ukrainian].
11. Tcmotc, I. G., Rabik, V. G., & Lukashhuk, Iu. A. (2021). Rozroblennia mobilnikh zasobiv neiropodibnogo kriptografichnogo shifruvannia ta deshifruvannia danikh u realnomu chasi. Visnik Natcionalnogo universitetu "Lvivska politekhnika". Seriia: Informatciini sistemi ta merezhi, 9, 84–95. [In Ukrainian].
12. Tsmots, I., Tsymbal, Y., Khavalko, V., Skorokhoda, O., & Tesluyk, T. (2018). Neural-Like Means for Data Streams Encryption and Decryption in Real Time. Processing of the 2018 IEEE 2nd International Conference on Data Stream Mining and Processing, DSMP 2018, 438–443.
13. Tsmots, I., Tsymbal, Y., Skorokhoda, O., & Tkachenko, R. (2019). Neural-like methods and hardware structures for real-time data encryption and decryption. Kompiuterni nauki ta informatciini tekhnologii (CSIT-2019): materiali XIV Mizhnarodnoi naukovo-tekhnichnoi konferentcii, 17–20 veresnia 2019, Lʹviv, Ukraina, 248–253. [In Ukrainian].
14. Verma, A., & Ranga, V. (2020). Security of RPL based 6LoWPAN Networks in the Internet of Things: A Review. IEEE Sens. J., 20, 5666–5690.
15. Volna, E., Kotyrba, M., Kocian, V., & Janosek, M. (2012). Cryptography Based On Neural Network. Proceedings of the 26th European Conference on Modeling and Simulation, 386–391.

Yu. A. Lukashchuk

Lviv Polytechnic National University, Lviv, Ukraine

IMITATION MODEL FOR CALCULATING WEIGHTS FOR NEURO-LIKE DATA ENCRYPTION/DECRYPTION

The article presents the results of research of an initiation model for calculating weights for neuro-like data encryption/ decryption. In course of research a simulation model for calculating weights for a given neural network architecture has been developed. This architecture is specified by the bit size of the input message and the bit size of the inputs. An analysis of recent research and publications on the relevance of the problem of cryptographic encryption and real-time data decryption was conducted. The method of singular value decomposition was chosen for the work in contrast to the principal component analysis algorithm. This was primarily done in order to be able to work with an input matrix of arbitrary dimension when the principal component analysis algorithm assumes that the input matrix is square. The developed simulation model is based on the improved method of singular value decomposition, and the Jacobi rotation method was used to find the eigenvalues and eigenvectors. The practical value is that the simulation model provides a quick calculation of the coefficients for given neural network architecture. A flexible user interface has also been developed, which allows the user to clearly and in detail get acquainted with the operation of the algorithm. In the developed application, you can view each step of the calculations, which in turn allows checking the correctness of execution at each stage. The software was developed in Visual Studio 2019 and using the C# programming language. The Accord.Math library was connected for matrix operations. Also, the work of the developed simulation model is demonstrated. As an example was used an input message with bit size – 16 and input bit size – 2. Based on these data, the dimension of the input matrix will be 8×2. The incoming message is set by the user. As a result, a matrix of 2×2 weights was calculated. In the future, this matrix will be used to encrypt and decrypt the incoming message. However, there are requirements for the message, namely that the bit size of the message itself, as well as the bit size of the inputs, must be the same as when finding the matrix of weights.

Keywords: neural network; SVD; neuroelements; eigenvectors; eigenvalues; matrix transformations; Jacobi rotation method.