DOI: 10.37943/17STXF5228

Daniar Aizhulov
Master, Lecturer, Institute of Cybernetics and Information Technology
daniar.aizhulov@gmail.com, orcid.org/0000-0001-5496-4639
Satbayev University, Kazakhstan

Maksat Kurmanseiit
Master, Researcher, Institute of Energy and Mechanical Engineering
kurmanseiit.maksat@gmail.com, orcid.org/0000-0002-5334-6537
Satbayev University, Kazakhstan

Nurlan Shayakhmetov
Master, Researcher, Institute of Energy and Mechanical Engineering
shayakhmetovn@gmail.com, orcid.org/0000-0002-7559-4371
Satbayev University, Kazakhstan

Madina Tungatarova
PhD, Dr. Eng., Associate Professor, Institute of Energy and Mechanical Engineering; Researcher,
madina.tungatarova@gmail.com, orcid.org/0000-0003-3138-3326
Satbayev University, Joldasbekov Institute of Mechanics and Engineering, Kazakhstan

Ainur Suleimenova
Master, Inventory planner
S_ain@mail.ru, orcid.org/0009-0006-0583-5869
Sandvik Mining and Construction, Kazakhstan

SYNTHETIC DATA GENERATION FOR ANN MODELING OF THE HYDRODYNAMIC PROCESSES OF IN-SITU LEACHING

Abstract: The work presents an approach to enhance the forecasting capabilities of In-Situ Leaching processes during both the production stage and early prognosis. ISL, a crucial method for resource extraction, demands rapid on-site forecasting to guide the deployment of new technological blocks. Traditional modeling techniques, though effective, are hindered by their computational demands and network throughput requirements, particularly when dealing with substantial datasets or remote computing needs. The integration of AI technologies, specifically neural networks, offers a promising opportunity for expedited calculations by leveraging the power of forward propagation through pretrained neural models. However, a critical challenge lies in transforming conventional numerical datasets into a format suitable for neural modeling. Furthermore, the scarcity of training data during the production phase, where vital parameters are concealed underground, poses an additional challenge in training AI models for In-Situ Leaching processes. This research addresses these challenges by proposing a methodology for generating training data tailored to the most resource-intensive Computational Fluid Dynamics problems encountered during modeling. Traditional numerical modeling techniques are harnessed to construct training datasets comprising input and corresponding expected output data, with a particular focus on varying well network patterns. Subsequent efforts are directed at the conversion of the acquired data into a format compatible with neural networks. The data is normalized to align with the data ranges stipulated by the activation functions employed within the neural network architecture. This preprocessing step ensures that the neural model can effectively learn from the generated data, facilitating accurate forecasting of In-Situ Leaching processes. An advantage of proposed technique lies in provision of large, reliable datasets to train neural network to predict hydrodynamic properties based on technological regimes currently active or expected on ISL site. A major implication of this approach lies in applicability of pre-trained AI technologies to forecast future or determine current hydrodynamic regime in the stratum circumventing cost deterministic simulations currently deployed at mining sites. Hence, innovative approach outlined in this paper holds promise for optimizing forecasting, allowing for quicker and more efficient decision-making in resource extraction operations while getting around the computational barriers associated with traditional methods.

Keywords: In-Situ Leaching modelling; neural networks; data preparation; data normalization; computational fluid dynamics; synthetic data generation.

Introduction

The In-Situ Leaching (ISL) method, widely used in uranium mining in Kazakhstan, involves injecting a leaching solution (usually sulfuric acid) into the subterrain, which dissolves crystallized minerals, thereby forming a productive solution that is brought to the surface for subsequent processing. The method allowed Kazakhstan to become a world leader in uranium production, accounting for as much as 40% of world share [1]. Among 15 mining sites active in Kazakhstan, ISL provides a safe, reliable and comparatively inexpensive way metal extraction. Major problem during mining operations arises due to the fact, that when a solution enters the subterrain, consequent processes are hidden below the surface, which hinders decision making process [2]. Therefore, simulating techniques as well as software packages have been under development to allow visualization, forecasting and decision making based on geological, physical and chemical modeling [3, 4].

The ISL process itself is implemented by drilling a network of injection and production wells. The network configuration is determined by the geological and hydrological properties of the subsoil, and, usually, has a row or hexagonal pattern [4, 5]. In the first case, the rows of production and injection wells alternate; in the latter case, the injection wells form a cell in the form of a hexagon with a production well at the center. The cells are grouped into a technological block, which are part of the deposit as a whole. The injection and extraction of the solution into and out of the formation is carried out through filters located along the well [6].

The efficiency of production directly depends on the decisions made regarding the choice of drilling patterns (well location), filter positioning, well flow rates and other parameters that directly affect the hydrodynamics and chemistry of the leaching process [7]. These decisions are made depending on the specific geological and hydrogeological conditions in each block [8].

Decision making can be based on various techniques, as well as on the results of modeling the ISL process. Modeling of the ISL process consists of the following stages:

- geological modelling;
- hydrodynamic modelling;
- modeling of the kinetics of chemical processes of reacting components;
- economic assessment.

Geological modeling is the process of countering the ore body, determining the lithological and filtration properties of the formation [9]. The process typically involves geostatistical calculations to interpolate values in the interwell space from known data obtained through geophysical surveys [6]. Hydrodynamic modeling is usually carried out by numerical calculation based on conservation laws and Darcy's law, which describes the movement of fluids in porous medium [10]. Modeling the kinetics of chemical processes is usually based on the law of mass action [10]. Economic assessment is carried out by determining the operating time of the unit, operating and capital costs [11].

The work [12] highlights the results of mathematical modeling to create a geological and technological model of uranium ISL, as well as the use of mathematical modeling to solve geotechnological and environmental problems. The paper [13] details the application of a 3D reactive transport approach using the Hytec code at an operational scale in Kazakhstan. Hytec technology uses computing clusters to solve mass transport problems, in particular to solve chemical equations for most geochemical reactions, including aqueous complexation, redox, dissolution/precipitation and sorption. The technology also has a hydrodynamic module that describes the filtration of solutions during ISL.

Mass transport modeling is computationally intensive for 3D problems [14]. The work [14] considers a classic approach to increasing computational performance by converting sequential computations on central processors into parallel computations on high-performance graphics processors. The non-stationarity of the hydrodynamic processes of the ISL leads to the need to carry out calculations with each change in production modes. On average, the operation of a process unit can last several years, while changes in hydrodynamic conditions can occur daily, and, in some cases, hourly. Hydrodynamic modeling involves calculating the pressure distribution in the reservoir, which is based on solving elliptic differential equations. Calculating such tasks for each block on medium-power machines can take up to two days, depending on: the operating time of the block, the dimensions of the computational grid, the frequency of changes in flow rates at wells, etc.

Acceleration of resource-intensive calculations can potentially be achieved through the use of neural network modeling. The effective use of neural networks to speed up modeling tasks has been demonstrated in solving biological problems [15].

The authors of this article suggest that a properly trained neural network model will significantly speed up the process of hydrodynamic modeling during ISL. The neural network is trained on a set of input and output data, however, due to the complexity of the problem of calculating pressure fields, a significant amount of training data is required. However, there is no information available regarding the actual pressure distribution throughout the entire area under consideration during the production process. The collection of training data, nevertheless, can be achieved by generation done through deterministic traditional calculations based on conservation and Darcy laws.

The authors of the current work propose a technique for generating synthetic data to form the input layer of neural networks in order to simulate the process of uranium mining using the ISL method.

The paper discusses a background of application of machine learning approaches for simulating physical processes in “Methods section”; an approach proposed by authors to generate training datasets in “Generation of synthetic data” section. The latter section discusses conventional method of hydrodynamic simulation, which was used to generate training data, its processing, and splitting to input and output datasets for learning purposes.

Methods

Currently, machine learning (ML) methods are used in the field of mining, in particular in processes associated with drilling, blasting, logistics, processing and transportation of enriched metals, etc. [16]. In the mining industry, ML is used at various stages of operation, from exploration to final reclamation (Figure 1) [17].

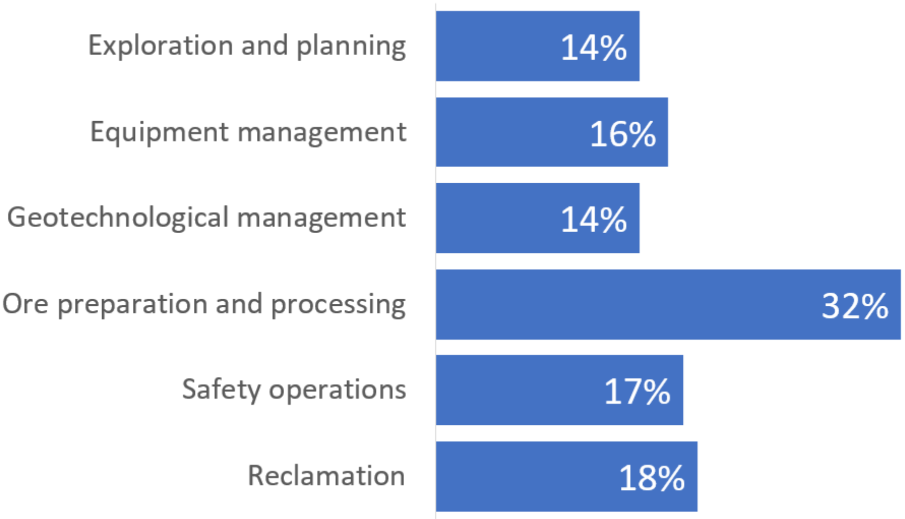

Figure 1. Share of published articles on the use of ML at various stages of field exploitation [17]

Increasing demand for raw materials, complex geological structures of ores and declining ore grades necessitate high-quality mineralogical analysis for efficient mining operations. Mineralogical data plays a critical role in estimating production duration and solving problems in ore exploration, mining and processing. Recent advances in mineral processing technology enable the use of sophisticated instrumentation and machine learning to increase the efficiency of design and operational workflows. Machine learning coupled with accurate data collection allows to diagnose and optimize various enterprise parameters in real time to improve recovery and energy efficiency. Integrating machine learning can: provide quantitative assessments of relationships between process units, optimize entire process, and identify performance issues [Minerals].

Recent years have seen significant application of deep learning, a form of machine learning, in a number of areas. Unlike many other machine learning techniques, deep learning naturally takes advantage of automatic pattern discovery from data combined with modeling frameworks [18].

The use of neural networks in modeling physical processes provides advantages such as their ability to capture complex nonlinear relationships, adaptability to different types of data, learning from production data, automatic feature extraction, and parallel processing capabilities. However, a number of disadvantages include the requirement for a significant amount of training data, limited interpretability, limitations in extrapolation, and the complexity of the initial configuration [19].

The Physics-Informed Neural Networks (PINNs) approach has been widely used, which is a type of approximation of universal functions described by partial differential equations, and includes in the learning process knowledge of the physical laws governing a given dataset [20]. In particular, DeepXDE technology, developed by Lu's group at the University of Pennsylvania, is used to solve forward and inverse problems, as well as partial differential equations. Vivid examples are: solving the Poisson, Laplace, Heimholtz equations, etc. [21]. PINNs combine data-driven deep learning techniques with the governing equations of a physical system to simultaneously fit data and enforce fundamental laws of physics. This is achieved by incorporating physics constraints into the loss function, allowing the neural network to optimize its weights and biases to minimize both fitting errors and violations of the underlying physics.

In neural networks, the input layer is the initial vector that receives and processes input data. It consists of neurons (also known as nodes or units), the number of which is equal to the size of the input data. Each neuron in the input layer represents a function or attribute of the input data. For example, when solving problems on a computational grid, the values of the input layer can be the values at the grid nodes and/or boundary and/or initial conditions.

The main function of the input layer is to pass input data forward through the network by applying weights and biases to the input values and passing them on to subsequent layers, often called hidden layers. These hidden layers then perform calculations and transformations on this data to learn and extract relevant features, which ultimately results in network output, which may be classification, regression, or some other tasks depending on the architecture and purpose of the network.

The current work describes the process of generating input layer data and final data for network training. The input layer is formed by normalizing and vectorizing the initial pressure distribution in the reservoir, while the final data is calculated by traditional numerical modeling. In other words, a set of synthetic data is generated to train a neural network to solve the problem of determining reservoir pressure at given well flow rates during uranium ISL.

Generation of synthetic data (3D domain state before and after)

A two-dimensional area of the ISL technological block is considered, with specified positions of production and injection wells (Figure 2).

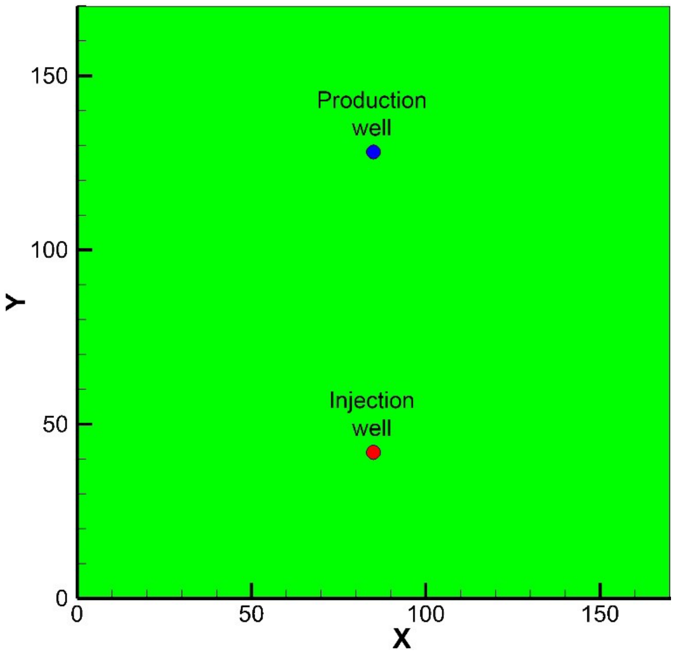

Figure 2. The domain under consideration

The area and dimension of the grid are fixed. Each well has a corresponding flow rate in m^3/day. Grid nodes corresponding to wells have hydraulic head values determined by the following equation:

$$h_{well} = \frac{q_{well}}{\Delta V K_f} \tag{1}$$

where h_{well} is the hydraulic head at the injection or production well, q_{well} is the well flow rate, ΔV is the volume of the well assembly, K_f is the rock filtration coefficient. In all other nodes, the pressure is zero.

At the moment, the neural network will be trained according to the positions of the wells; accordingly, the filtration characteristics were considered as homogeneous and isotropic.

The Mass Conservation Law in the context of problems in continuum mechanics can be formulated using a partial differential equation. The change in fluid density (denoted as *ρ*) with time (*t*), accompanied by flow ($\vec{v}$), is described by the following equation:

$$\frac{\partial(\phi\rho)}{\partial t} + div(\rho\vec{v}) = \frac{q_{inj}}{dV}\int_{-\infty}^{+\infty} \delta\left(x - x_{inj}, y - y_{inj}\right) dxdy + \frac{q_{prod}}{dV}\int_{-\infty}^{+\infty} \delta\left(x - x_{prod}, y - y_{prod}\right)\ dxdy \qquad (2)$$

where, ρ is the fluid density, t is time, $\vec{v}$ is the Darcy filtration rate, and δ is the Dirac delta function.

It is assumed that all fluids in the rock, including groundwater and injected solutions, are considered incompressible, which means $\frac{d\rho}{dt} = 0$. Therefore, the equation can be simplified to the following form:

$$div(\vec{v}) = \frac{q_{inj}}{dV}\int_{-\infty}^{+\infty} \delta\left(x - x_{inj}, y - y_{inj}\right) dxdy + \frac{q_{prod}}{dV}\int_{-\infty}^{+\infty} \delta\left(x - x_{prod}, y - y_{prod}\right)\ dxdy \qquad (3)$$

Flow in a porous medium at low speeds is described by the Darcy's Law:

$$\vec{v} = \vec{u}\phi = -K_f\overrightarrow{grad}\,h \qquad (4)$$

where $\vec{v}$ is the Darcy filtration rate, $\vec{u}$ is the absolute flow velocity.

Substituting the equation (4) for (3), the following equation is obtained:

$$div\left(-K_f\overrightarrow{grad}\,h\right) = \frac{q_{inj}}{dV}\int_{-\infty}^{+\infty} \delta\left(x - x_{inj}, y - y_{inj}\right) dxdy + \frac{q_{prod}}{dV}\int_{-\infty}^{+\infty} \delta\left(x - x_{prod}, y - y_{prod}\right)\ dxdy \qquad (5)$$

The equation (5) is an elliptic equation and is usually solved using iterative methods in continuum mechanics problems.

The solution of this second-order partial differential equation is carried out by semi-implicit upper and lower relaxation schemes. In this case, for even iteration steps, upper relaxation is applied, and for odd iteration steps, lower relaxation is applied:

$$h_{i,j}^{n+1} = (1 - \Delta\tau)h_{i,j}^{n} + \Delta\tau \cdot \frac{\frac{K_{f_{i,j}}h_{i+1,j}^{n} + K_{f_{i-1,j}}h_{i-1,j}^{n+1}}{\Delta x^2} + \frac{K_{f_{i,j}}h_{i,j+1}^{n} + K_{f_{i,j-1}}h_{i,j-1}^{n+1}}{\Delta y^2} + \frac{q_{i,j}}{dV}}{\frac{K_{f_{i,j}} + K_{f_{i-1,j}}}{\Delta x^2} + \frac{K_{f_{i,j}} + K_{f_{i,j-1}}}{\Delta y^2}} \qquad (6)$$

for n=2k+1:

$$h_{i,j}^{n+1} = (1 - \Delta\tau)h_{i,j}^{n} + \Delta\tau \cdot \frac{\frac{K_{f_{i,j}}h_{i+1,j}^{n+1} + K_{f_{i-1,j}}h_{i-1,j}^{n}}{\Delta x^2} + \frac{K_{f_{i,j}}h_{i,j+1}^{n+1} + K_{f_{i,j-1}}h_{i,j-1}^{n}}{\Delta y^2} + \frac{q_{i,j}}{dV}}{\frac{K_{f_{i,j}} + K_{f_{i-1,j}}}{\Delta x^2} + \frac{K_{f_{i,j}} + K_{f_{i,j-1}}}{\Delta y^2}} \qquad (7)$$

Thus, if we take into account that the final result was obtained in *N* iterations, for a specific well configuration, h^0 is the data set for the input layer, h^N.

Numerous data grids have been generated for the purposes of training the neural network. Positions of both injection and production wells were determined at random within the area of the computational domain. Hydraulic head values at the grid nodes where wells reside were set to 7 and -7 meters for production and injection wells respectively, which corresponds flow rates 5 and -5 m^3/hour.

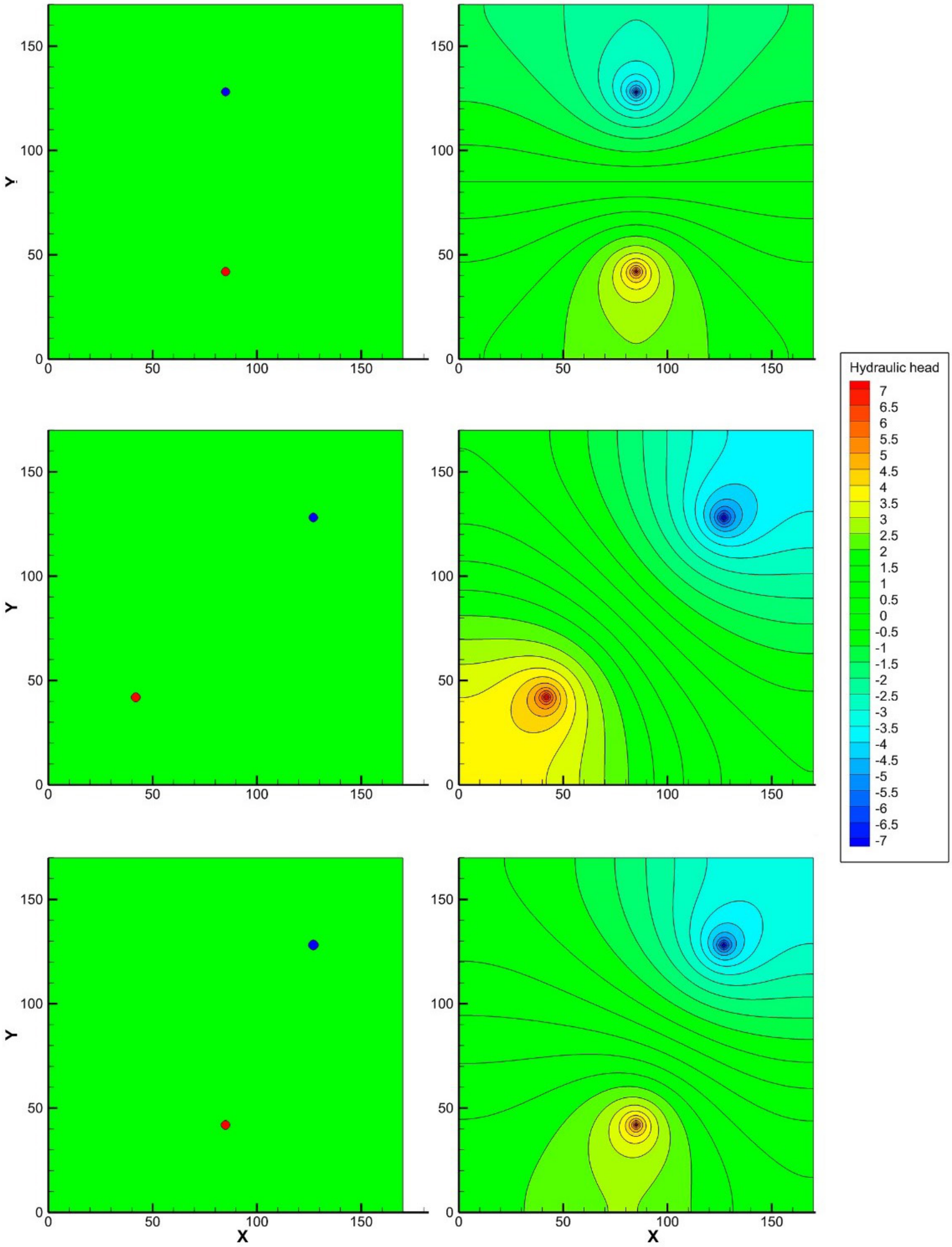

Figure 3. Some examples of generated grids

The dimensions of the grid are fixed, and in current case were equal to 170x170 for the area equal to 170x170 meters, i.e., the are of each cell in the grid is equal to 1 m^2. Boundary conditions were set to no-flow Neuman.

Let's consider a neural network with the number of layers L. The layers have n_i neurons, where $i = \overline{1, L}$ is the layer number. The number of neurons in the input and output layers of a single one: $n_1 = n_L$, and, in the context of the current task, will be equal to the number of nodes in the computational grid.

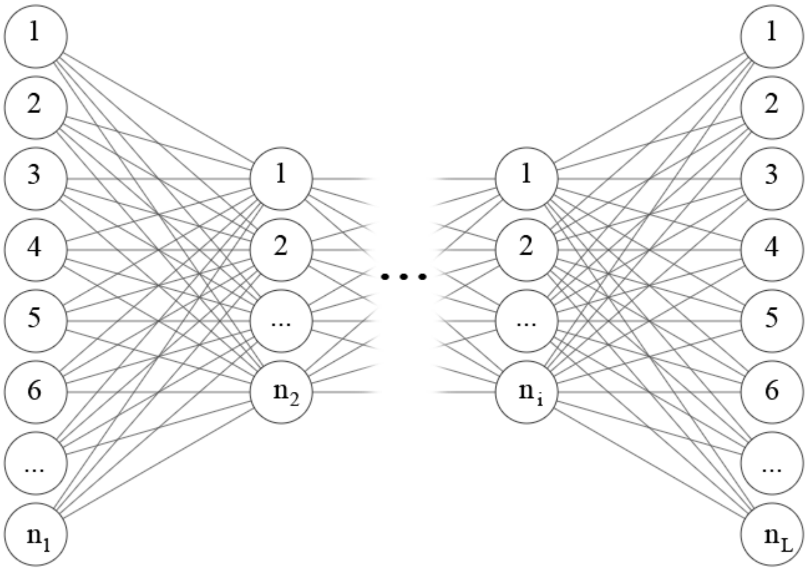

Figure 4. Neural network configuration example

The model includes the weight values w between all neighboring layers, the bias values b and the activation function f. Thus, the activation of each neuron is calculated using the following formula:

$$a_j^i = f(\textstyle\sum_k w_{jk}^i a_k^{i-1} + b_j^i) \tag{8}$$

Regression type problems in neural network modeling use specific activation functions that transform data in a certain range (usually from -1 to 1). For example, the hyperbolic tangent activation function is well suited for regression problems:

$$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1 \tag{9}$$

Data preparation is an important and critical step preceding the training process and has a significant impact on the efficiency and accuracy of the neural model. The main stages are: collecting and sampling data; integration of data from different sources and data processing. The last stage includes vectorization and data normalization [22].

In the current work, data collection is carried out based on the results of hydrodynamic modeling. Data sets h^0 and h^N are reduced to one-dimensional arrays L_l and E_o of the input layer and expected results, respectively.

According to the above, the process of training a neural network requires normalization of input data and arrays containing expected results to determine the loss function. There are a number of normalization algorithms for data processing: Min-Max Normalization, Decimal Scaling Normalization, Z-Score Normalization, Median Normalization, Sigmoid Normalization, and Tanh estimators (Figure 5). Min-Max Normalization scales data to a fixed range (usually between 0 and 1) by subtracting the minimum value and dividing by the range. Decimal Scaling Normalization shifts decimal points of data values to achieve normalization without altering the distribution. Z-Score Normalization rescales data to have a mean of 0 and a standard deviation of 1 by subtracting the mean and dividing by the standard deviation. Median Normalization adjusts data based on the median value, dividing each value by the median to center the distribution around 1. Sigmoid Normalization applies a sigmoid function to squash data between 0 and 1, often used in logistic regression to handle outliers. Tanh Normalization applies the hyperbolic tangent function to map data to the range [-1, 1], useful for handling data with negative values.

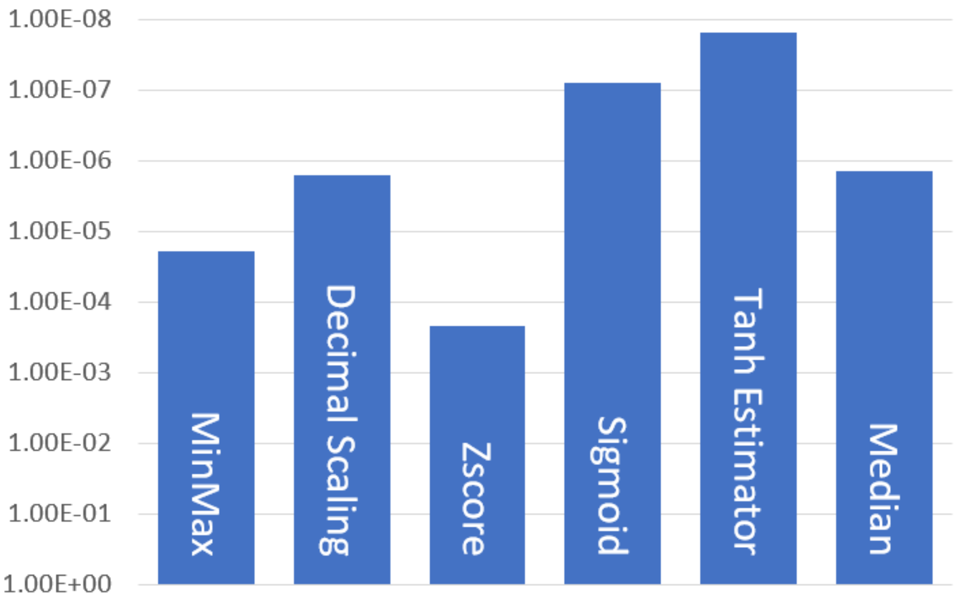

Figure 5. MSE errors when using various data normalization methods [23]

For the current task, when using an activation function that produces values in the range [-1; 1], where -1 will correspond to the minimum flow rate at the wells, and 1 to the maximum, taking into account the study of the effectiveness of various normalization methods in neural network modeling conducted in [23], it is reasonable to use the Min-Max algorithm to normalize hydrodynamic data. The reason behind choosing Min-Max algorithm is to scale their data to a fixed range between -1 and 1 and maintain the relative relationships between the original data points. This is to be able to reconvert data obtained through AI prediction to an original format.

Using Min-Max normalization, the input layer data is prepared using the following formula:

$$x_{normalized} = \frac{(high-low)*(x-min(I_l))}{max(I_l)-min(I_l)} \quad (10)$$

for each x value from the L_l array, where $high$ = 1 and low = -1.

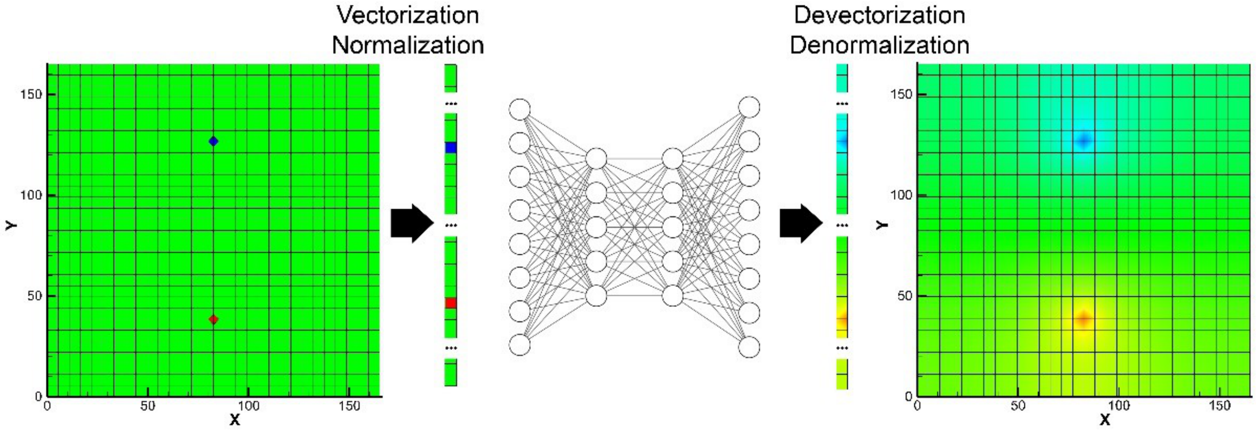

Figure 6. Schematic representation of data preparation for neural network

Model is to be trained on a fixed grid size, although the assumption is that grids of different dimensions will be resized to fit the node amount of input and output layers. This can be achieved through ordinary interpolation.

Eventual efficiency advantage of pre-trained neural network over conventional method can be estimated through Big-O notation of computational complexity.

Conclusion

Current modelling techniques provide a set of tools to determine the dynamics of the ISL process during production stage as well as during early prognosis. The continuous development of the deposit, requires a quick forecast capability on site, before launching new technological blocks. Traditional methods, however, are computationally expensive and might require introduction of cluster computing, which will be impossible to implement for every deposit or would require high throughput of the network in case of remote computing, due to large size of datasets.

An introduction of AI technologies such as neural networks might significantly decrease calculation times via forward propagation through a trained neural model. However, steps are required to convert a traditional numerical method dataset into the form appropriate for neural modelling.

An absence of training data available during production with necessary parameters being hidden underground, leads to a challenge of training an AI model.

In current work, a technique for generating training data for the most resource intensive CFD problems arising during ISL modeling has been proposed. Traditional numerical modelling has been used to generate training datasets with input and expected outputs. In this case, specifically for varying well network patterns.

Further work has been conducted to convert obtained data into a structure suitable for neural network. The data has been further normalized to correspond to data ranges imposed by the activation functions used in the network.

Neural networks, once trained, can dramatically cut computation times through forward propagation. In other words, calculation would only go forward through a configured network once. In contrast, when using conventional methods, this would require solving an elliptical equation iteratively for each time the hydrodynamic regimes change, which occurs daily over several ears of a technological block operation.

Next step of current scientific research would be to train the network and to obtain predictions with sufficient accuracy.

Acknowledgements

The work has been funded by the Ministry of Science and Higher Education of Kazakhstan through the Program of targeted financing BR18574136.

References

[1] World Uranium Mining Production. (2024). https://world-nuclear.org/information-library/nuclear-fuel-cycle/mining-of-uranium/world-uranium-mining-production.aspx

[2] Collet, A., Regnault, O., Ozhogin, A., Imantayeva, A., & Garnier, L. (2022). Three-dimensional reactive transport simulation of Uranium in situ recovery: Large-scale well field applications in Shu Saryssu Bassin, Tortkuduk deposit (Kazakhstan). *Hydrometallurgy*, 211(105873), 105873. https://doi.org/10.1016/j.hydromet.2022.105873

[3] van der Lee, J., De Windt, L., Lagneau, V., & Goblet, P. (2003). Module-oriented modeling of reactive transport with HYTEC. *Computers & Geosciences*, 29(3), 265–275. https://doi.org/10.1016/s0098-3004(03)00004-9

[4] Laurent, G., Izart, C., Lechenard, B., Golfier, F., Marion, P., Collon, P., Laurent, T., Jean-Jacques, R., & Lev, F. (2019). Numerical modelling of column experiments to investigate in-situ bioleaching

as an alternative mining technology. *Hydrometallurgy*, 188, 272–290. https://doi.org/10.1016/j.hydromet.2019.07.002

[5] Li, G., & Yao, J. (2024). A review of in situ leaching (ISL) for uranium mining. *Mining*, 4(1), 120–148. https://doi.org/10.3390/mining4010009

[6] Poezhaev, I.P., Polinovskiy, K.D., & Gorbatenko, O.A. (2017). Uranium Geotechnology: A Training Manual. *Almaty: Insitute of High Technology Kazatomprom*.

[7] Shayakhmetov, N.M., Aizhulov, D.Y., Alibayeva, K.A., Serovajsky, S., & Panfilov, I. (2020). Application of hydrochemical simulation model to determination of optimal well pattern for mineral production with In-Situ Leaching. *Procedia Computer Science*, 178, 84-93. https://doi.org/10.1016/j.procs.2020.11.010

[8] IAEA. (2001). *Manual of acid in situ uranium mining technology*. Vienna: IAEA.

[9] Aizhulov, D., Tungatarova, M., & Kaltayev, A. (2022). Streamlines Based Stochastic Methods and Reactive Transport Simulation Applied to Resource Estimation of Roll-Front Uranium Deposits Exploited by In-Situ Leaching. *Minerals*, 12(10), 1209. https://doi.org/10.3390/min12101209

[10] Aizhulov, D.Y., Shayakhmetov, N.M., & Kaltayev, A. (2018). Quantitative model of the formation mechanism of the rollfront uranium deposits. *Eurasian Chemico-Technological Journal*, 20(3), 213-221. https://doi.org/10.18321/ectj724

[11] Shayakhmetov, N.M., Alibayeva, K.A., Kaltayev, A., & Panfilov, I. (2023) Enhancing uranium in-situ leaching efficiency through the well reverse technique: A study of the effects of reversal time on production efficiency and cost. *Hydrometallurgy*, 219, 106086. https://doi.org/10.1016/j.hydromet.2023.106086

[12] Li, H., Tang, Z., & Xiang, D. (2024). Study on numerical simulation of reactive-transport of groundwater pollutants caused by acid leaching of uranium: A case study in Bayan-Uul area, northern China. *Water*, 16(3), 500. https://doi.org/10.3390/w16030500

[13] Regnault, O., Lagneau, V., & Fiet, N. (2014). 3D Reactive Transport simulations of Uranium In Situ Leaching: Forecast and Process Optimization. *Uranium - Past and Future Challenges*, 725-730. https://doi.org/10.1007/978-3-319-11059-2_83

[14] Tungatarova, M.S., Kurmanseiit, M.B., & Shayakhmetov, N.M. (2020). GPU Accelerated Modeling of In-Situ Leaching Process and Streamline Based Reactive Transport Simulation. *Procedia Computer Science*, 178, 145-152. https://doi.org/10.1016/j.procs.2020.11.016

[15] Wang, S., Fan, K., Luo, N., Cao, Y., Wu, F., Zhang, C., Heller, K.A., & You, L. (2019). Massive computational acceleration by using neural networks to emulate mechanism-based biological models. *Nature Communications*, 10(1), 4354. https://doi.org/10.1038/s41467-019-12342-y

[16] Fu, Y., & Aldrich, C. (2020). Deep Learning in Mining and Mineral Processing Operations: A Review. *IFAC PapersOnLine*, 53(2), 11920–11925. https://doi.org/10.1016/j.ifacol.2020.12.712

[17] Azhari, F., Sennersten, C.C., Lindley, C.A., & Sellers, E. (2023). Deep learning implementations in mining applications: a compact critical review. *Artificial Intelligence Review*. https://doi.org/10.1007/s10462-023-10500-9

[18] Jooshaki, M., Nad, A., & Michaux, S. (2021). A Systematic Review on the Application of Machine Learning in Exploiting Mineralogical Data in Mining and Mineral Industry. *Minerals 11*, Article 816. https://doi.org/10.3390/min11080816

[19] Aggarwal, C.C. (2023). Neural Networks and Deep Learning. *Springer*, (2).

[20] Karniadakis, G.E., Kevrekidis, I.G., Lu L., Perdikaris, P., Wang, S. & Yang, L. (2021). Physics-informed machine learning. *Nature Reviews Physics*, 6 (3), 422-440. https://doi.org/10.1038/s42254-021-00314-5

[21] Lu, L., Meng, X., Mao, Z., & Karniadakis, G. (2021). DeepXDE: A deep learning library for solving differential equations. *SIAM Review*, 63(1), 208–228. https://doi.org/10.1137/19M1274067

[22] Yu, L., Wang, S. & Lai, K.K. (2007). Foreign-exchange-rate forecasting with artificial neural networks. *Springer*.

[23] Bhanja, S., & Das, A. (2021). Deep Neural Network for Multivariate Time-Series Forecasting. In Bhattacharjee, D., Kole, D.K., Dey, N., Basu, S., Plewczynski, D. *Proceedings of International Conference on Frontiers in Computing and Systems. Advances in Intelligent Systems and Computing*, 1255. https://doi.org/10.1007/978-981-15-7834-2_25